



UNIVERSIDAD  
DE LOS ANDES  
VENEZUELA



AKADEMIA

tesis universitaria en cd rom



Consejo de Publicaciones

**Universidad de Los Andes**  
**Facultad de Ciencias Económicas y Sociales**  
**Instituto de Estadística Aplicada y Computación**

**Sistema Automatizado de Registro,  
Control y Análisis de Inventarios  
(SARCAI) para PROULA,  
Planta de Medicamentos**

**Versión 2.0**

Autor: Ernesto Antonio Ponsot Balaguer

Tutor: Víctor Márquez

Asesor: Lionel Castro

Trabajo presentado como Proyecto de Grado para optar al **Título de *Magister Scientae en Estadística*** que otorga el Instituto de Estadística Aplicada y Computación de la Facultad de Ciencias Económicas y Sociales - Universidad de Los Andes

Mérida, Diciembre de 1997

Título de la obra: **Sistema Automatizado de Registro, Control y Análisis de Inventarios (SARCAI) para PROULA, Planta de Medicamentos**

Autor: Ernesto A. **Ponsot** Balaguer

Editado por el Consejo de Publicaciones de la Universidad de Los Andes  
Av. Andrés Bello, antiguo CALA. La Parroquia.  
Mérida, estado Mérida. Venezuela  
Telefax (+58274) 2713210, 2712034, 2711955  
e-mail cpula@ula.ve  
<http://www.ula.ve/cp>

1ª edición en CD-ROOM. 2007

Reservados todos los derechos

© Ernesto A. Ponsot B.

Diseño de portada: INNOVA. Diseño y Tecnología, C.A.

Mérida, Venezuela, 2007

A mis dos Morellas ... y mi Marina,  
manantiales tibios de fuerza de voluntad.

## **Agradecimientos**

La lucha del hombre por forjarse y forjarle a los suyos un futuro mejor, no tiene valor si no se acompaña de las ideas, del pensamiento, de la experiencia, que será su legado. Agradezco en primer término a la Universidad de Los Andes, fuente de la luz del saber, en la que me formé como ingeniero, me formé como maestro en Estadística y me formo todos los días como profesor y ser humano. Es gracias al trabajo que en ella se desarrolla, libre, sin ataduras y barreras, que he podido alcanzar esta meta. Agradezco también a sus hombres, mi padre, fuente de inspiración y amor por la Universidad, Víctor Márquez, tutor de este trabajo y amigo en la verdadera acepción de la palabra, Lionel Castro, sin cuya disposición a hacer de la Planta de Medicamentos un “Proyecto Universitario” este trabajo no habría sido posible, a Giampaolo Orlandoni y Segundo Quiroz, profesores que siempre han estado dispuestos a escuchar mis planteamientos cumpliendo a cabalidad su papel. Finalmente agradezco a la Escuela de Estadística y al Instituto de Estadística Aplicada y Computación, en particular a mis colegas profesores, por las facilidades que he recibido para la realización de este proyecto.

# Contenido

<b>Contenido</b>	<b>Pág.</b>
Índice de Tablas del SARCAI v. 2.0	5
Índice de Consultas del SARCAI v. 2.0	7
Índice de Paneles del SARCAI v. 2.0	10
<b>INTRODUCCIÓN</b>	<b>11</b>
Convenciones Tipográficas.	14
Instalación	15
<b>CAPÍTULO I. El Sistema de Inventarios de la Planta</b>	<b>16</b>
I.1. PROULA y la Planta de Medicamentos	17
I.2. El Sistema de Inventarios	19
<b>CAPÍTULO II. Diseño de la Base de Datos</b>	<b>22</b>
II.1. Diagrama Referencial del SARCAI	24
II.2. Diseño Detallado de la Base de Datos	31
II.2.1. Tablas	32
II.2.2. Consultas	73
II.3. Resumen	124
<b>CAPÍTULO III. Estructura de Paneles del Sistema</b>	<b>125</b>
III.1. Panel de Control Principal	126
III.1.1. Panel de Clientes	129
III.1.2. Panel de Proveedores	133
III.1.3. Panel de Almacén de Insumos	135
III.1.4. Panel de Producción	144
III.1.5. Panel de Almacén de Productos Terminados	154
III.1.6. Panel de Estadísticas Generales	165
III.2. Resumen	166
<b>CAPÍTULO IV Modelos</b>	<b>167</b>
IV.1. Modelos EOQ	169
IV.1.1. Modelo Teórico	170
IV.1.2. Formulario para los Modelos EOQ	174
IV.1.3. Descripción	175
IV.1.4. Código Fuente	178
IV.2. Modelos de Programación	189
IV.2.1. Modelo Teórico	190
IV.2.2. Formulario para los Modelos de Programación	195
IV.2.3. Descripción	196
IV.2.4. Ejemplo de la Hoja de Cálculo	200
IV.2.5. Código Fuente	202
<b>Conclusiones y Recomendaciones</b>	<b>219</b>
<b>Bibliografía</b>	<b>222</b>
<b>Anexos</b>	

## Índice de Tablas del SARCAI v. 2.0

Nombre de la Tabla	Pág.
Accionistas de Clientes .....	32
Cargos .....	32
Ciudades de Estados .....	33
Clientes .....	33
Condiciones.....	34
Contactos en Clientes.....	35
Contactos en Proveedores .....	35
Conversiones.....	35
Cotizaciones.....	36
Demanda Intercambio Excel.....	36
Demanda Intercambio Excel Modelo .....	37
Demanda por Fechas Tabla.....	37
Departamentos .....	38
Destinos.....	38
Documentación .....	38
Elementos del Panel de control.....	39
Entradas de Insumos 3ros .....	39
Escalas de Precios .....	39
Estados Civiles.....	40
Estados de Insumos.....	40
Estados de Países .....	41
Facturas .....	41
Figuras Jurídicas .....	42
Fórmulas .....	42
Gráfico de Productos Estrella T1 .....	44
Gráfico de Productos Estrella T2 .....	44
Grupos de Clientes.....	44
Insumos.....	44
Insumos en Fórmulas .....	45
Insumos en Órdenes .....	45
Insumos en Órdenes de Producción.....	46
Llegadas de Insumos al Almacén .....	47
Magnitudes Físicas.....	48
Meses del Año.....	48
Modelos de Programación .....	48
Modelos EOQ .....	49
Modelos Resultados Excel.....	51

Notas .....	51
Notas en Cotizaciones.....	51
Notas en Facturas.....	52
Órdenes de Compra.....	52
Órdenes de Producción .....	53
Países.....	55
Pedidos de Clientes .....	55
Personas .....	56
Planes de Producción .....	57
Precios de Productos .....	58
Presentaciones.....	58
Productos Cotizados a Clientes.....	59
Productos Demandados.....	59
Productos en Modelos P.....	60
Productos en Planes .....	60
Productos Facturados a Clientes .....	61
Productos Genéricos .....	62
Productos Pedidos de Clientes .....	62
Productos Terminados .....	62
Productos Terminados y Escalas de Precios.....	63
Proveedores.....	64
Razones de Salidas.....	65
Referencias Bancarias de Clientes .....	65
Referencias Bancarias de Personas.....	66
Referencias Comerciales de Clientes.....	66
Referencias Comerciales de Personas.....	66
Revisiones de Fórmulas .....	67
Salidas del Almacén de Insumos .....	67
Sectores Económicos .....	68
Sucursales de Clientes.....	69
Sucursales de Proveedores.....	69
Teléfonos de Clientes.....	69
Teléfonos de Proveedores .....	70
Tenencia de Locales.....	70
Tipos de Clientes.....	71
Tipos de Insumos .....	71
Tipos de Órdenes de Pedido .....	72
Tipos de Proveedores .....	72
Tipos de Salidas .....	72
Unidades de Medición .....	72
Vehículos .....	73



# Índice de Consultas del SARCAI v. 2.0

<b>Nombre de la Consulta</b>	<b>Pág.</b>
Accionistas de Clientes Consulta.....	75
Cantidad Despachada en Almacén I.....	76
Cantidad Despachada en Almacén T.....	76
Cantidad Facturada en Almacén T.....	77
Cantidad Ordenada en Almacén I.....	77
Cantidad que ha Salido de Insumo por Lote.....	78
Cantidad que ha Salido de Insumo por Lote 0.....	78
Cantidad Recibida de Insumo por Orden.....	78
Cantidad Recibida de Insumo por Orden 0.....	79
Cantidad Recibida en Almacén I.....	79
Cantidad Total Despachada en Almacén I.....	79
Cantidad Total Desechada en Almacén I.....	80
Cantidad Total Despachada en Almacén T.....	80
Cantidad Total Facturada en Almacén T.....	81
Cantidad Total Facturada por Pedido.....	81
Cantidad Total Ordenada en Almacén I.....	81
Cantidad Total Pedida en Almacén T.....	82
Cantidad Total Producida en Almacén T.....	82
Cantidad Total Recibida en Almacén I.....	83
Cantidades en Almacén I.....	83
Clientes Consulta.....	83
Clientes y Facturas G.....	84
Clientes y Facturas S.....	84
Clientes y Facturas V.....	85
Clientes y Pedidos G.....	85
Clientes y Pedidos S.....	86
Clientes y Pedidos V.....	86
Clientes, Pedidos y Facturas G.....	87
Clientes, Pedidos y Facturas S.....	87
Clientes, Pedidos y Facturas V.....	87
Contactos en Clientes Consulta.....	88
Contactos en Proveedores Consulta.....	88
Conversión de Insumos.....	88
Demanda Bimensual.....	89
Demanda Mensual.....	89
Demanda por Fechas.....	89
Demanda por Fechas 0.....	90
Demanda por Fechas Creación.....	90
Demanda Promedio Bimensual.....	91
Demanda Promedio Mensual.....	91

Demanda Promedio Trimestral .....	91
Demanda Trimestral.....	92
Estado de Pedidos .....	92
Existencia Real en Almacén I.....	93
Existencia Real en Almacén I 1 .....	93
Existencia Real en Almacén I 2 .....	94
Existencia Real en Almacén T.....	94
Existencia Total Real an Almacén I.....	95
Existencia Total Real en Almacén T .....	96
Explosión de Materiales.....	97
Explosión de Materiales 1.....	97
Explosión de Materiales 2.....	97
Facturas Totales Despachadas .....	98
Facturas Totales entre Fechas .....	99
Facturas y Pedidos consultas .....	99
Fórmulas Consulta .....	99
Fórmulas y Productos Terminados .....	100
Gráfico de Productos Estrella .....	100
Gráfico de Productos Estrella 1 .....	101
Gráfico de Productos Estrella 2 .....	101
Insumos en Órdenes Consulta.....	101
Insumos Ordenados y Recibidos.....	101
Lista de Precios Vigente en Fecha Buscada .....	102
Lista Sencilla de Clientes.....	102
Lista Sencilla de Productos Terminados.....	103
Lista Sencilla de Proveedores .....	103
Monto Global de Facturas.....	103
Monto Global de Pedidos.....	104
Órdenes de Producción Consulta.....	104
Pedidos Totales entre Fechas .....	104
Planes de Producción Globales.....	105
Planes de Producción Globales 1 .....	105
Planes de Producción Mensuales .....	106
Planes de Producción Mensuales RC.....	106
Precios de Insumos .....	107
Precios de Productos Hasta Fecha Buscada.....	107
Precios de Productos Hasta Fecha Buscada 1.....	107
Productos Despachados entre Fechas .....	108
Productos Pedidos entre Fechas.....	108
Productos Pedidos y Despachados entre Fechas.....	109
Productos Pedidos y Facturados .....	110
Productos Pedidos y Facturados 1 .....	110
Productos Pedidos y Precios .....	111
Productos Pedidos y Precios 1 .....	111
Productos Pedidos y Precios 2 .....	112
Productos Pedidos y Precios 3 .....	112

Productos Terminados e Insumos .....	112
Productos Terminados y Fórmulas Consulta .....	113
Productos y Escalas consulta .....	113
Proveedores Consulta.....	114
Resumen de Órdenes de Producción.....	114
Salidas por Lote en Almacén I.....	114
Subtotal Kg y Lt Insumos en Fórmulas .....	115
Subtotal Kg y Lt Insumos en Ordenes de Prod.....	115
Total Utilizado en Producción por Insumo .....	116
Total Utilizado en Producción por Lote.....	116
Totales de Productos Facturados a Clientes.....	116
Últimas Fechas de Precios de Insumos .....	117
Últimas Fechas de Precios de Productos .....	117
Últimos Precios de Insumos Consulta .....	117
Últimos Precios de Productos Consulta.....	117
Unidades Convertibles .....	118
Unidades de Concentración .....	118
Unidades de Presentación .....	119
Unidades de Tiempo .....	119
Ventas Anuales D .....	119
Ventas Anuales G .....	120
Ventas Anuales P.....	120
Ventas Mensuales D .....	120
Ventas Mensuales G .....	121
Ventas Mensuales P.....	121
Ventas Semestrales D .....	121
Ventas Semestrales G .....	122
Ventas Semestrales P.....	122
Ventas Trimestrales D.....	123
Ventas Trimestrales G.....	123
Ventas Trimestrales P .....	123

# Índice de Paneles del SARCAI V. 2.0

<b>Nombre del Panel</b>	<b>Pág.</b>
Panel de Control Principal .....	126
Panel de Clientes.....	129
Panel de Informes sobre Clientes.....	132
Panel de Proveedores .....	133
Panel de Informes sobre Proveedores .....	134
Panel de Almacén de Insumos .....	135
Panel de Insumos .....	136
Panel de Órdenes de Compra.....	138
Panel de Llegadas al Almacén de Insumos.....	140
Panel de Estadísticas sobre Insumos.....	143
Panel de Producción.....	144
Panel de Fórmulas.....	146
Panel de Productos Terminados.....	148
Panel de Órdenes de Producción.....	150
Panel de Planes de Producción .....	152
Panel de Almacén de Productos Terminados.....	154
Panel de Cotizaciones .....	156
Panel de Pedidos .....	158
Panel de Facturas .....	160
Panel de Estadísticas sobre Productos Terminados .....	162
Panel de Estadísticas Generales .....	165

## Introducción

Es conocido que no se puede construir un edificio sin sentar primero sus bases. Esta afirmación es la guía central del presente trabajo. Por supuesto, aquí no se trata de cemento, ladrillos, madera ni vigas, se trata del edificio de la *información*, isomorfo con aquél, pero de naturaleza intangible, virtual.

La estadística es tal vez el elemento constructivo más importante y difundido de este edificio. Ella compila datos, los procesa científicamente y permite extraer conclusiones, inferencias para tomar decisiones. Es la materialización cuantitativa del esfuerzo de las organizaciones por conservar y aprender de su propia historia y de la de otros. Sin embargo, sus intentos serán falaces si las bases del edificio no están bien construidas, si el insumo fundamental de la estadística, los datos, mienten, engañan, no están disponibles a tiempo o su integridad está en tela de juicio.

Esta reflexión es de importancia capital para entender por qué un proyecto que podría estar catalogado dentro de las ciencias de la Computación, se ha desarrollado en el ámbito de la Estadística. El asunto es que no se puede pensar en las organizaciones como objetos estáticos. Las organizaciones son dinámicas, cambian con el paso del tiempo, surgen cada día nuevos elementos de juicio, nuevos datos. Un análisis estadístico realizado en un momento particular de la vida de la organización, será válido para este instante y tal vez no para otro, en cuanto las condiciones, los supuestos, los datos, pueden cambiar. La complejidad que representa el manejo de grandes volúmenes de información generada por una empresa del mundo de hoy, hace infactible el empleo continuo de la estadística como herramienta de apoyo al proceso decisional operativo y directivo cotidiano, a no ser que la organización disponga en primer lugar de un Sistema de Información soportado por una Base de Datos bien construida, en la cual quede automáticamente registrada la historia en términos de datos. Si adicionalmente se incorporan al Sistema de Información algoritmos de alto nivel que automaticen los cómputos complejos

requeridos por el análisis estadístico, se tiene una herramienta fundamental y cada vez más refinada, que enriquece la eficiencia de la empresa e incrementa su productividad.

Así pues, este trabajo es el primer intento por echar las bases del edificio de la información en la Planta de Medicamentos de PROULA (y con muy pocos cambios, en cualquier empresa productora de fármacos), especialmente pensando en uno de sus subsistemas más importantes: El Sistema de Inventarios.

La intención del proyecto es construir un Sistema de Información y la Base de Datos que lo soporte y además integrar algoritmos de cómputo que automaticen el cálculo numérico de algunos modelos (que podrían ser estocásticos) planteados a partir de los propios datos presentes en la base de datos. Hasta ahora, el diseño del Sistema de Información y la Base de Datos ha requerido un gran esfuerzo, y debido a factores externos -fundamentalmente deficiencias presupuestarias en la Planta de Medicamentos- no ha sido posible implantarlo en las instalaciones de la organización, no sólo para que fuese probado, sino para la obtención de datos en medios computacionales que permitieran estudiar modelos complejos y posteriormente, programar las soluciones apropiadas. Esta idea sigue en mente (y prosigue el trabajo asociado con ella), sin embargo, seguir esperando la consecución de datos, demorando la entrega del proyecto, haría que el mismo dejara de cumplir su misión como Proyecto de Grado para optar al título de *Magister Scientae* en Estadística. Así pues, en lo que respecta a los modelos programados, sólo se incluyen dos tipos: Uno que comprende las distintas variantes aplicadas al modelo del Tamaño Económico del Lote, y otro que resuelve un modelo de Programación Lineal para la Planeación de la Producción, ambos, en esencia, no probabilísticos, aunque basados en estadísticas de la demanda, obtenidas y proyectadas al futuro desde y hacia la base de datos del sistema (en línea y automáticamente).

SARCAI versión 2.0 es un producto de desarrollo tecnológico, un conjunto de programas para el computador, que resuelven el control y análisis de los datos relacionados con el inventario. Se ha decidido sacar a la luz pública la segunda versión del sistema (saltando la primera) sólo para enfatizar que el software empleado para su construcción ha variado desde el primer intento. En efecto, la primera versión del producto fue realizada en Microsoft ACCESS, versión 7.0 para Windows 95, y durante el proceso de desarrollo surgió la última versión de este producto que se

denomina Microsoft ACCESS 97 (de la familia OFFICE 97) para Windows 95. Se decidió entonces transformar lo hecho hasta el momento llevándolo a la nueva versión de ACCESS, ya que ésta incorpora varias mejoras (soporte para la Internet, mayor y mejor integración con componentes como EXCEL o WORD, VISUAL BASIC para aplicaciones como lenguaje principal de todos sus componentes, el núcleo de resolución de consultas SQL ahora desarrollado para plataformas de 32 bits, etc.). Así pues, SARCAI versión 2.0 está desarrollado en ACCESS 97 para Windows 95, utiliza VISUAL BASIC para Aplicaciones (un lenguaje de programación orientado a objetos y conducido por eventos) como lenguaje nativo, hace uso extensivo del JET SET SQL de 32 Bits de Microsoft para la resolución de consultas y utiliza los manejadores ODBC de 32 Bits del Windows 95 para comunicar y recibir datos a y desde otras aplicaciones, principalmente EXCEL 97. La documentación está disponible desde el propio sistema con la incorporación de hipervínculos a los documentos en formato html de Internet. En fin, podría decirse que este sistema, además de realizar su papel de gestor de información, es una buena muestra académica de la programación en entornos gráficos operativos (como el Windows 95) sacando provecho al máximo de las herramientas integradoras que dan al usuario del software gran versatilidad y facilidad de uso.

Este documento, contenedor de toda la información relativa al diseño y desarrollo del proyecto, está organizado de la siguiente forma: El Capítulo I, denominado “El Sistema de Inventarios de La Planta”, presenta el sistema de Inventarios de PROULA – Planta de Medicamentos, desde el punto de vista conceptual y la problemática asociada con los inventarios en general. El Capítulo II, denominado “Diseño de la Base de Datos”, muestra el diseño de la Base de Datos, núcleo del Sistema Automatizado de Registro, Control y Análisis de Inventarios SARCAI, tanto en el aspecto general como en aspectos particulares. El Capítulo III, denominado “Estructura de Paneles del Sistema”, presenta las distintas opciones implementadas y dispuestas formando una secuencia de paneles que definen las distintas secciones. El Capítulo IV, denominado “Modelos”, muestra con detalle el funcionamiento de dos de las tres alternativas disponibles en esta versión, incorporadas en la opción Estadísticas Generales, para el análisis de la demanda y modelado del inventario (según el Tamaño Económico del Lote y según Un Modelo de Programación Lineal). El Capítulo V, denominado “Conclusiones y Recomendaciones”, presenta los comentarios finales, reflexiones, conclusiones y recomendaciones surgidas del proceso de diseño y programación, que condensan la experiencia del autor y proponen líneas de acción a

seguir en el futuro. El cuerpo principal del texto concluye con secciones dedicadas a presentar la bibliografía utilizada en la realización del proyecto. Adicionalmente se incluyen varios anexos a los que se hará referencia en el texto y que forman parte integrante del documento: El Anexo 1 denominado “Formularios Básicos de Datos y Código Fuente que los Controla”, muestra los distintos Formularios (o ventanas de trabajo) del SARCAI, con excepción de aquellos dedicados a los Modelos (que se exponen con detalle en el Capítulo IV), se muestra además en este anexo el código que hace funcionar a cada formulario. El Anexo 2 denominado “Informes del SARCAI y Detalles de Programación” muestra los detalles (código fuente y definiciones) de los distintos informes que se producen a partir de la Base de Datos del sistema. El Anexo 3 denominado “Ejemplos de Informes SARCAI V. 2.0” muestra en orden alfabético todos los informes que entrega el sistema, producidos con datos ficticios pero útiles para formarse una idea de su operación y funcionalidad. Por último, el Anexo 4 denominado “Macros y Código Fuente del Módulo General del SARCAI v. 2.0”, muestra los listados y detalles del código programado para realizar ciertas funciones de servicio general.

## Convenciones Tipográficas

Antes de continuar con la lectura, es conveniente conocer algunas convenciones tipográficas que se utilizan en el texto. Estas son:

1. En letra *itálica* o **negrita** se resaltan palabras importantes o nombres de paneles de opciones del sistema.
2. En letra tipo `arial` pequeña se escriben las referencias bibliográficas y demás notas a pié de página.
3. Los nombres de tablas y campos de tablas se encierran entre corchetes ([ ]).
4. El código de programa Visual Basic y SQL se escribe en letra `Courier New` de tamaño 10, equiespaciada, excepto en el texto de las tablas.
5. Para dar a entender el nombre genérico de un parámetro o variable necesaria en el código programado, se encierra éste entre signos de menor y mayor (<>).



6. Las opciones del sistema se escriben con letra tipo Tahoma de tamaño 10.
7. Las teclas del teclado o combinaciones de ellas se presentan entre llaves ({ }).
8. Los demás tipos son utilizados de manera bien conocida: fuentes mayores para títulos, negritas y centrado para nombres de figuras y demás, comillas para citas textuales o palabras de sentido especial, etc.

## **Instalación**

El Sistema Automatizado de Registro, Control y Análisis de Inventarios, versión 2.0, es un producto de software diseñado para funcionar bajo la familia de sistemas operativos Microsoft Windows 95 (incluidos Windows NT Server y WorkStation). Por razones de seguridad se ha previsto que el proceso de instalación del software sea conducido por personal calificado y autorizado por el autor. Sin embargo, a título informativo, pueden describirse los siguientes aspectos generales: (1) el sistema ha de residir en una estructura de directorios C:\DATOS\PROULA 97\ para el archivo principal de la base de datos SARCAI.MDB, C:\DATOS\PROULA 97\HOJAS\ para las hojas de cálculo Excel 97 necesarias y C:\DATOS\PROULA 97\DOCUMENTOS\ para la documentación. Por su tamaño, los dos primeros componentes se entregan comprimidos (en formato PKZIP) en un disco flexible 3½”, y la documentación en otro del mismo formato. La instalación completa del software requiere la existencia de Office 97, por supuesto Access y especialmente Excel, y la configuración y entonamiento especiales de algunos de estos componentes.

# CAPÍTULO I

## El Sistema de Inventarios de la Planta

En cualquier actividad productiva, uno de los puntos críticos que debe ser considerado dentro de la planeación es el manejo de inventarios. El inventario es el conjunto de bienes “en reserva” con que cuenta la organización para protegerse de la escasez futura de esos bienes, luego, la manutención del inventario parte del principio de que se presentará escasez de los bienes (que no es otra cosa que la no tenencia de ellos cuando se les necesite).

Obviamente, si todas las fases del proceso productivo, desde la obtención de los insumos hasta la venta, fuesen perfecta y determinísticamente conocidas (esto es, sabido exactamente lo que se demanda, cuándo se demanda, el tiempo que se demoran los insumos en llegar, los tiempos de retardo en la producción, etc.), la manutención de inventarios no sería necesaria, podría entonces emplearse lo que los autores modernos han denominado “Producción Justo a Tiempo”<sup>1</sup> (en inglés JIT, Just In Time), que promueve la reducción drástica de los inventarios con tendencia a desaparecerlos del todo, de forma que cada bien necesario para la producción esté en el lugar requerido justo antes de que sea necesitado por la tarea. Sin embargo, aunque este “ideal” está avalado por razones de peso (sobre todo aquellas que se desprenden del Control Total de la Calidad), representa en ocasiones un reto prácticamente imposible de salvar.

Por ejemplo, en nuestra sociedad, donde la libre competencia tiene raíces nominales mas no efectivas, donde la organización de las empresas para la competencia leal y libre no ha sido una preocupación valedera, donde el poder adquisitivo de los ciudadanos fluctúa de una manera en

---

<sup>1</sup> TAHA, Hamdy A. “Investigación de Operaciones. V Edición”. Alfaomega Grupo Editor. México – 1995, p.622.

gran cantidad de casos inesperada, donde las reglas del juego de mercado cambian constantemente, pensar en que será posible la producción JIT es por demás arriesgado. No significa esto que no se deba tener en mente la reducción del inventario como objetivo loable, lo que significa es que en nuestra realidad, la reducción total de inventarios para la producción es prácticamente imposible con los niveles de desarrollo actuales.

Los inventarios no representarían mayor problema a no ser porque consumen recursos de la organización, tiempo, espacio, dinero. La inversión necesaria para mantener inventarios puede llegar a niveles muy altos y es claro que en vista de que se trata de un elemento del proceso productivo, que requiere del encadenamiento de capitales, los cuales no rendirán frutos de forma clara e inmediata, es lógico que las organizaciones se preocupen por **controlar** y **optimizar** este aspecto de su actividad.

La mayor contribución metodológica al tema de inventarios ha sido aportada por la idea de sistemas. En efecto, el inventario, tal como se dijo, es el conjunto de bienes acumulado, sin embargo, lo verdaderamente importante para la organización no es controlar el inventario, sino controlar y optimizar el *sistema de inventarios* de la organización. El sistema de inventarios es algo más que los bienes acumulados, es el conjunto de procesos, decisiones, políticas, actividades, etc. organizados con la finalidad de proteger el proceso productivo de la escasez (y por lo tanto garantizar la permanencia en el tiempo de la organización).

Este trabajo pretende proveer una herramienta técnica computarizada, materializada en un sistema de información, que utiliza los lineamientos de bases de datos relacionales para apoyar el manejo del sistema de inventarios, en particular de la Planta de Medicamentos de PROULA.

## **I.1. PROULA y la Planta de Medicamentos**

PROULA, C. A., empresa promotora de la Universidad de Los Andes, es por definición “catalizadora de todos los procesos inherentes a la promoción y puesta en marcha de proyectos

rentables, en especial aquellos que tienen como sustento la capacidad científica y tecnológica de la institución”<sup>2</sup>.

La Planta de Medicamentos, ubicada en el sector Llano de la Alegría, en las inmediaciones de la ciudad de Lagunillas, en el estado Mérida, es uno de los componentes del proyecto y su objetivo es “la investigación, desarrollo, producción y comercialización de medicamentos genéricos, medicamentos de marca, productos naturales, productos veterinarios, cosméticos, fluidoterápicos y productos químicos”<sup>3</sup>.

La operación de la Planta de Medicamentos comenzó formalmente hace algo más de dos años gracias a la adquisición y recuperación de maquinaria de laboratorio con un préstamo de 86 millones de bolívares<sup>4</sup> gestionado entre PROULA y el Consejo de Fomento de la Universidad de Los Andes. En la actualidad se encuentra en la primera etapa de producción, concentrada en la elaboración de sólidos, líquidos y productos no estériles para el mercado de medicamentos genéricos local y nacional.

La Planta de Medicamentos no cuenta hasta el momento con ningún tipo de sistema computarizado para el manejo de la información. Sin embargo, está vista la necesidad de inversiones en esta línea (más allá del procesamiento electrónico de textos, hojas de cálculo y otras), tanto en equipos como en programas, infraestructura de redes y personal. El software producido en este trabajo representa el primer producto de este tipo para la planta de medicamentos y se espera que le sigan otros esfuerzos similares, integrados a lo aquí expuesto y compatibles con este trabajo. Hay que mencionar por otra parte que como no se trata de un producto que esté en la “línea de ventas” o que genere rentabilidad directa, ha sido difícil llegar a esta etapa de desarrollo, ya que escasea la disposición (posiblemente por las limitantes de recursos) a invertir en informática, tal vez con la idea de que estas inversiones consumen recursos que podrían ser empleados en la producción. Es necesario decir que, aunque efectivamente existe una curva de utilidad de los proyectos de sistemas de información según la cual es posible medir la ganancia que éstos producen a la organización en función de las

---

<sup>2</sup> Presentación PROULA C.A. División Medicamentos, bíptico, Lagunillas – Mérida.

<sup>3</sup> Op. cit. Presentación PROULA C.A.

<sup>4</sup> Heras, A. “Alcance. La Planta de Medicamentos va al Cuartel”. ULA – Mérida, Mayo 1996, p.24.

inversiones necesarias, las organizaciones modernas requieren un control efectivo sobre sus datos para competir exitosamente en un mercado no subvencionado y la Planta debe prepararse aceleradamente para este escenario si se desea que sobreviva.

## I.2. El Sistema de Inventarios

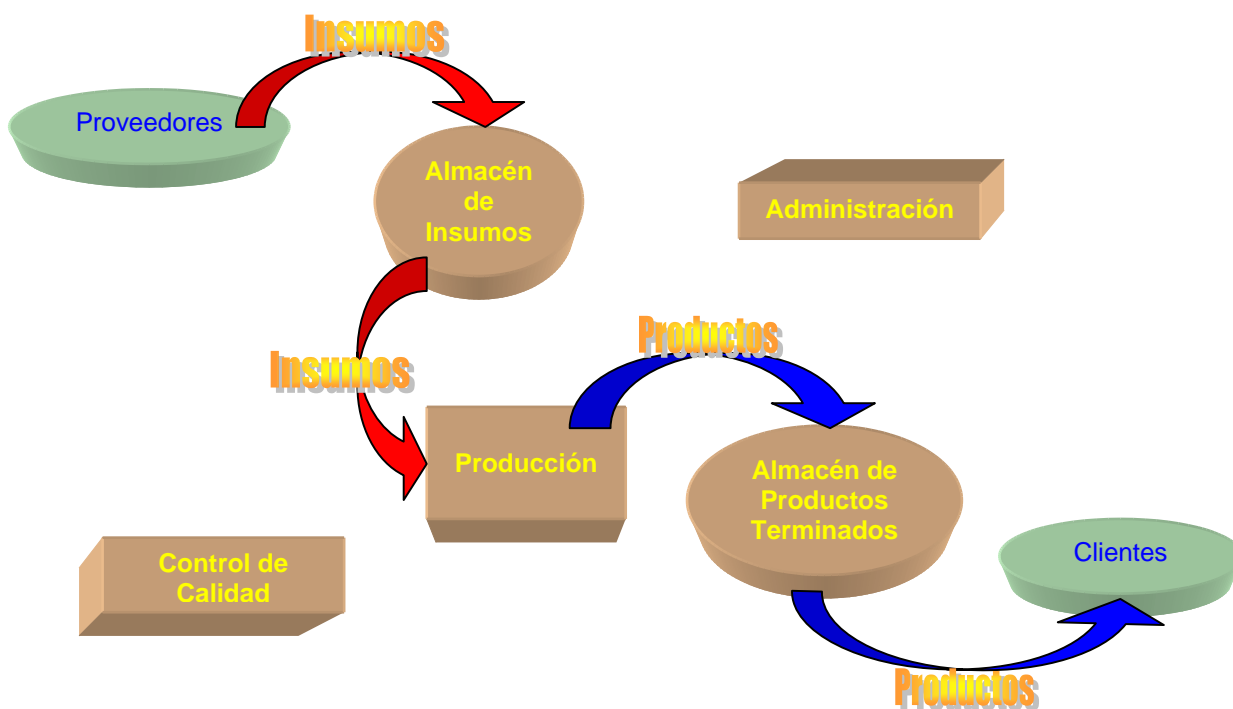
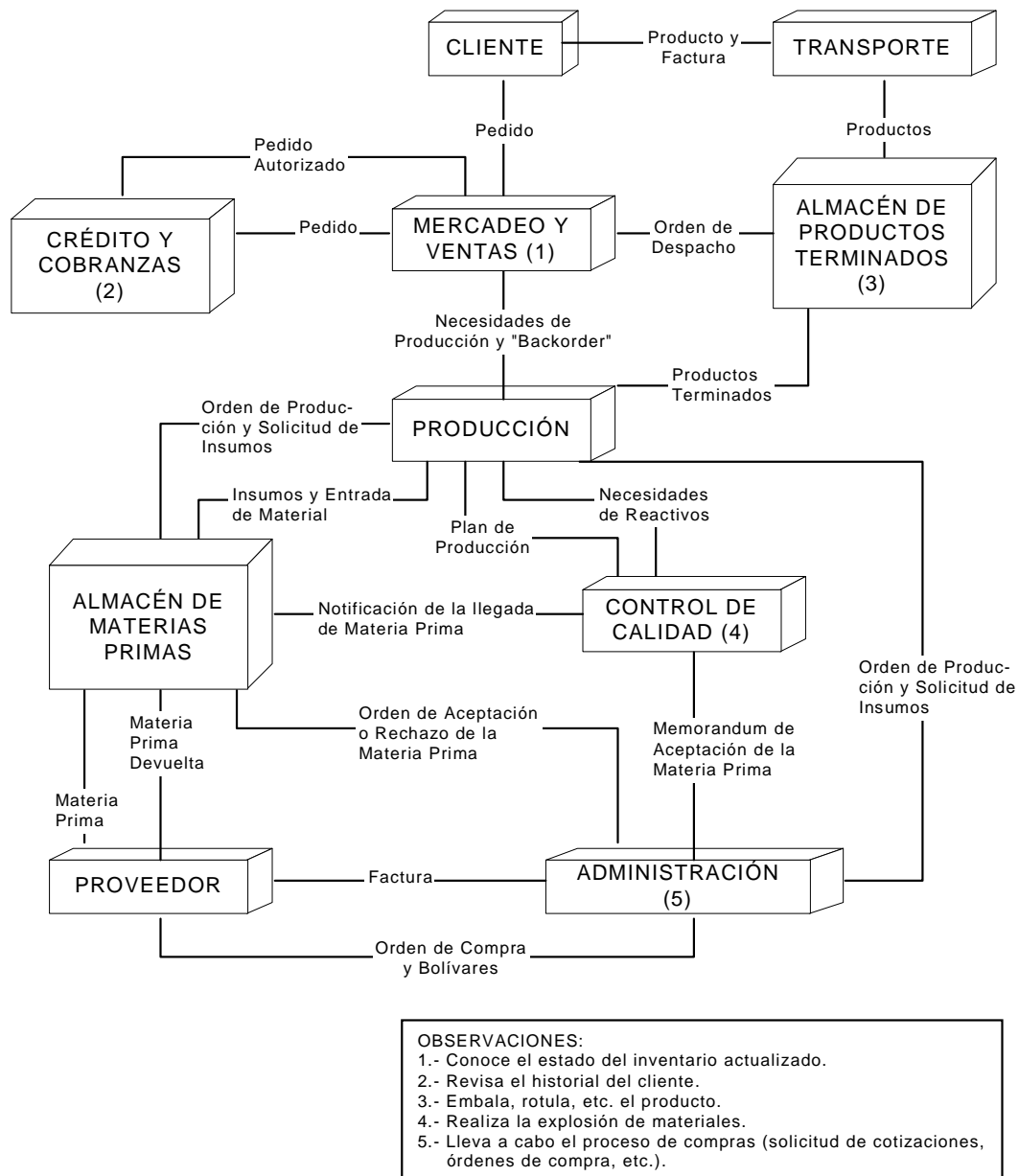


Figura 1. Sistema de Inventarios de la Planta de Medicamentos

Como puede observarse en la Figura 1, existen en la planta dos inventarios: el Inventario de Insumos y el Inventario de Productos Terminados. Ambos se relacionan a través de la Producción (que transforma los insumos en productos terminados). La mayoría de los procesos productivos que se implementan en la planta se basan en la combinación, en condiciones química o farmacológicamente especiales, de diversos insumos para la elaboración de medicamentos aptos para el consumo humano. Así, Producción solicita al Almacén de Insumos lo requerido, procesa y envasa lo que éste le entrega de acuerdo con una fórmula farmacéutica y elabora un producto que luego es entregado al Almacén de Productos Terminados para ser vendido a terceros.

El Almacén de Insumos se surte con las compras realizadas a los Proveedores y se consume al entregar lo adquirido a la Producción. El Almacén de Productos Terminados se surte con la Producción y se consume al entregar (en venta) lo producido a un Cliente. La Administración fiscaliza los aspectos contables de la operación y Control de Calidad fiscaliza los aspectos cualitativos relacionados con la calidad de los insumos comprados y productos terminados.



**Figura 2. Procesos Relacionados con el Inventario en la Planta**

La Figura 2 muestra este proceso con más de detalle y permite visualizar el flujo de documentos entre los distintos actores de la organización en lo relativo al sistema de inventarios.

La mayoría de los documentos y controles mostrados en la Figura 2 se encuentran implementados en SARCAI, bien sea como elementos diferenciados especialmente o bien como datos integrados dentro de estos elementos.

A medida que el lector avance en la lectura de este documento, irá descubriendo los conceptos asociados con cada parte del proceso, desde el punto de vista de los datos. El sistema de información pretende (en su primera versión terminada) ser un espejo del funcionamiento de la organización imitando en forma virtual lo descrito en la figura. En futuras versiones será posible ahondar perfeccionando algunos procesos de información a medida que la organización madura en su funcionamiento.

## CAPÍTULO II

### Diseño de la Base de Datos

Una base de datos es simplemente un contenedor eficiente de datos que responde a un modelo de datos. El avance vertiginoso de las ciencias de la computación en materia de bases de datos, ha puesto al alcance de las organizaciones herramientas para controlar sus datos, basadas en lo que se denomina tecnología de bases de datos. Hace algún tiempo atrás, los analistas de sistemas pensaban en el sistema de información (o en los sistemas, en plural), como entes complejos (generalmente en manos de ‘expertos’) que llegaban a convertirse en el objetivo, más que en un medio, para la organización. Efecto ampliamente documentado en la literatura especializada de este enfoque “divisionista” es la proliferación, dentro de las empresas, de sistemas de información especializados en tareas concretas, pero sin ninguna posibilidad de conexión o consolidación de datos y pobremente documentados. Cada vez que surgía una nueva necesidad de información o cambiaba alguna de las existentes, el personal de informática desarrollaba un nuevo sistema (desde cero) para hacerle frente.

La tecnología de bases de datos ha cambiado la forma de ver el problema de la información. Los datos, en sí mismos, tienen ahora valor ya no solamente como insumos de la información, sino por ellos mismos. Los analistas han cambiado el enfoque y han notado que un almacén bien previsto de datos es el insumo más importante para cualquier nueva necesidad de información que se presente (un poco en el sentido de los inventarios). Así pues, en el enfoque de bases de datos - que utiliza este proyecto -, antes de pensar en el sistema de información global, o en las distintas aplicaciones programadas, se piensa en los datos y se propone un modelo de datos (por supuesto, a partir del estudio y análisis de la organización), que no necesariamente responde a los aspectos funcionales de la empresa, sino más bien a los aspectos semánticos o de significado de



los datos mismos. El paradigma de esta relativamente novedosa forma de pensar es la **independencia** entre los distintos niveles de una base de datos. Así, los aspectos físicos (el hardware) de la base de datos se procuran independientes de los aspectos conceptuales o lógicos (el modelo de datos) y éstos se procuran independientes de los aspectos de vistas (distintas visuales u ópticas a través de las cuales el usuario interactúa con la base de datos). El punto principal es que lo verdaderamente importante es la conservación de los datos independientemente de los programas que acceden a ellos. Con esto en mente, es claro que una base de datos puede crecer sin traumas a medida que la organización crece, ya que este proceso no involucra desechar lo programado hasta el momento, sólo será necesario agregar nuevos elementos de datos a la base y los programas (independientes) que accederán a ellos. Adicionalmente, este contenedor de datos **centraliza** desde el punto de vista conceptual (no necesariamente físico, no necesariamente en un solo computador) los datos, evitando la redundancia excesiva, garantizando al máximo la integración y la integridad de los mismos y asegurando la propagación de las correcciones o modificaciones en línea a los largo de los sistemas que comparten la base de datos.

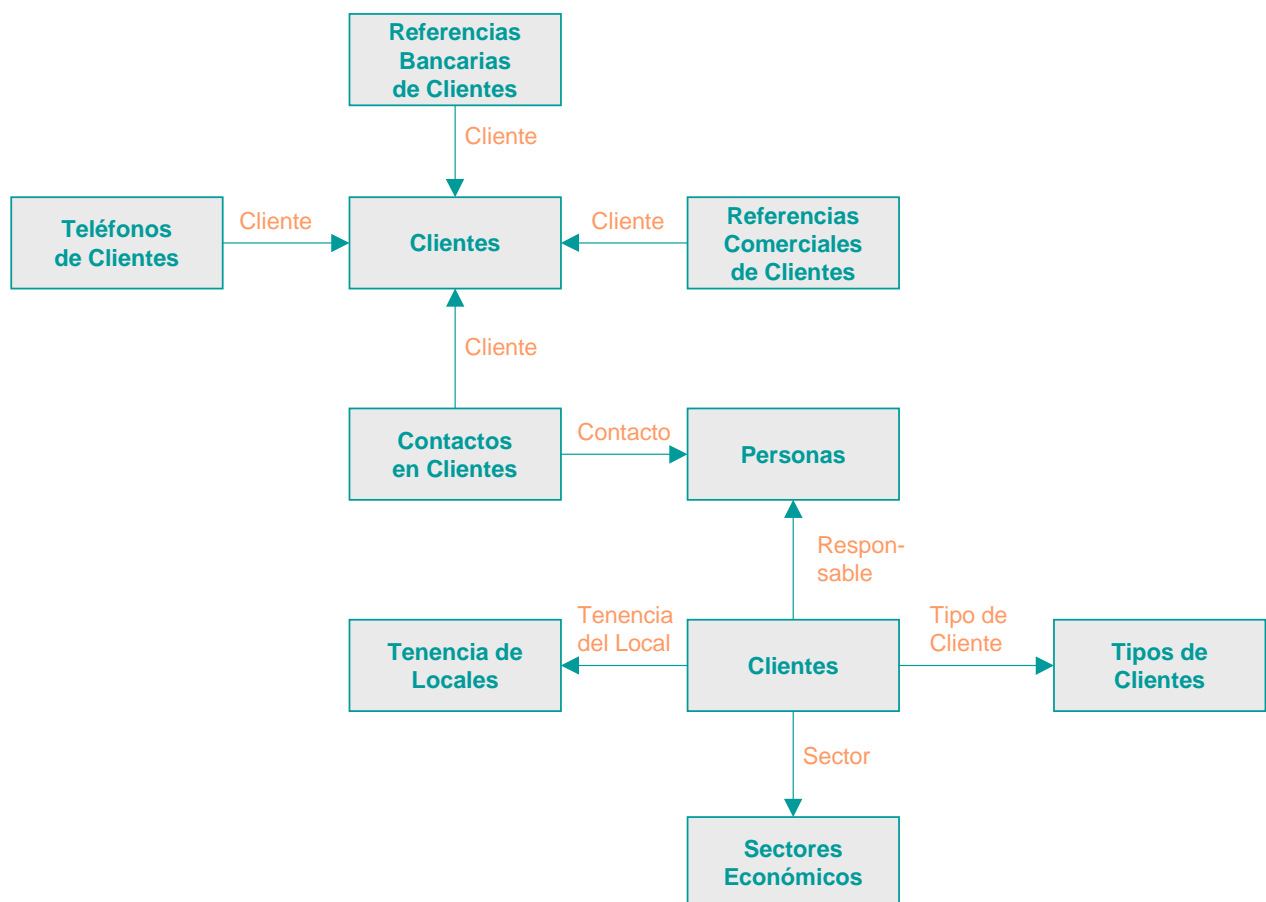
En particular, SARCAI versión 2.0 está centrado en un modelo de base de datos de la Planta de Medicamentos. Este modelo responde a los lineamientos del Modelo Relacional de Bases de Datos, uno de cuyos manejadores más modernos es el ACCESS de Microsoft.

El Modelo Relacional propone que todos los elementos de la base de datos pueden ser vistos como relaciones (en el sentido matemático). Una relación o tabla es una estructura de datos que se vislumbra con una cabecera, donde se muestran los distintos tipos de datos o atributos que engloba la relación y un conjunto de tuplas o registros que contienen valores asociados con cada atributo, en cada tupla relacionados con el mismo objeto o individuo. Entonces, una relación (o tabla, como las llama el ACCESS) puede ser vista como un conjunto de atributos (o campos) y un conjunto de tuplas (o registros). Al tratarse de conjuntos, es claro que el orden en que se dispongan los campos o los registros no es relevante. Consecuentemente, la localización de un valor específico para un campo, dentro de un registro, dentro de una tabla, ocurre en el modelo relacional no por su posición, sino por su contenido.

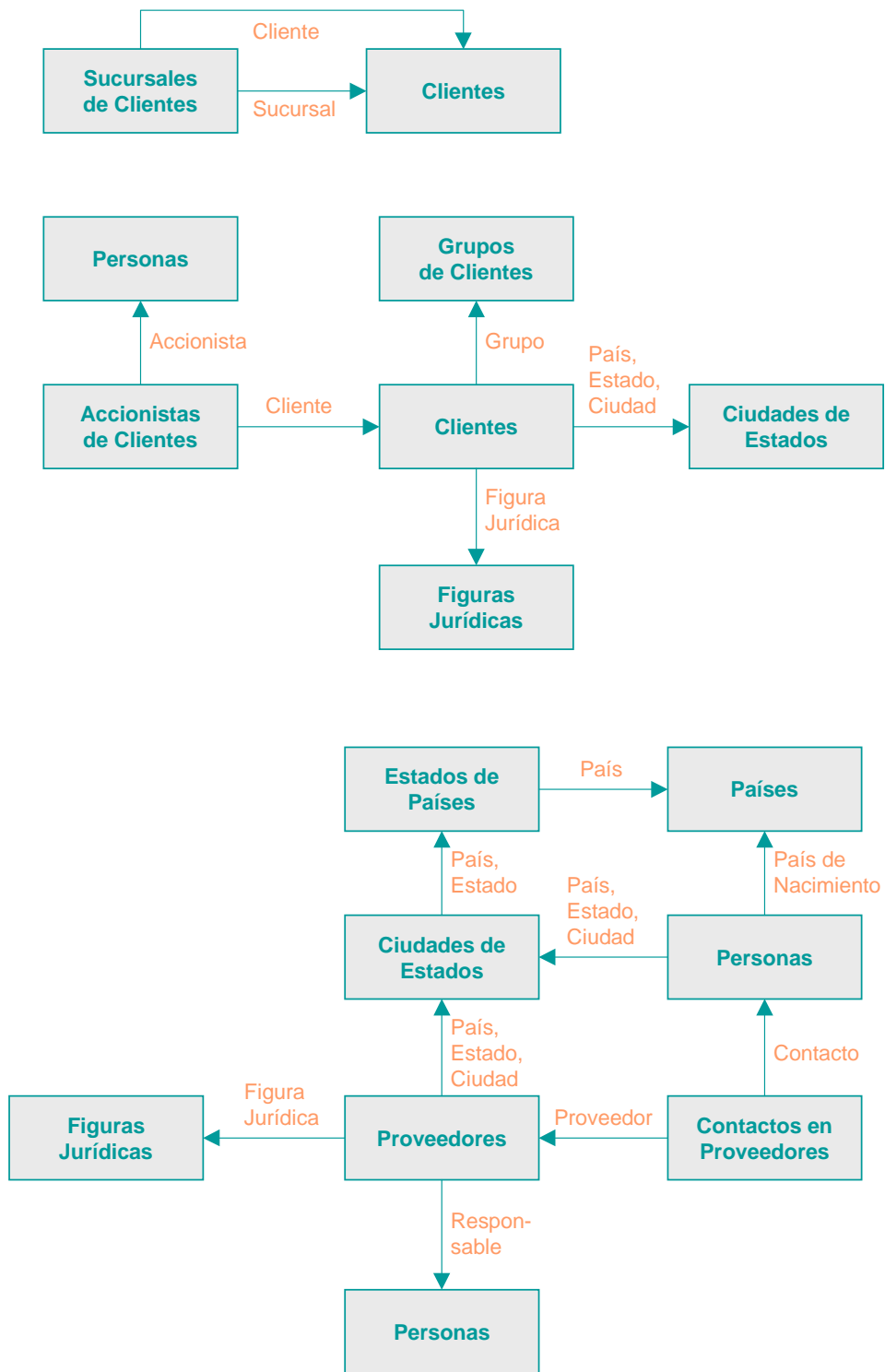
Otro elemento importante para el modelo relacional de bases de datos tiene que ver con la determinación de claves primarias. Una clave primaria es el mínimo conjunto de atributos de la relación, cuyos valores, al combinarse, hacen posible identificar de forma única a cada tupla de la relación.

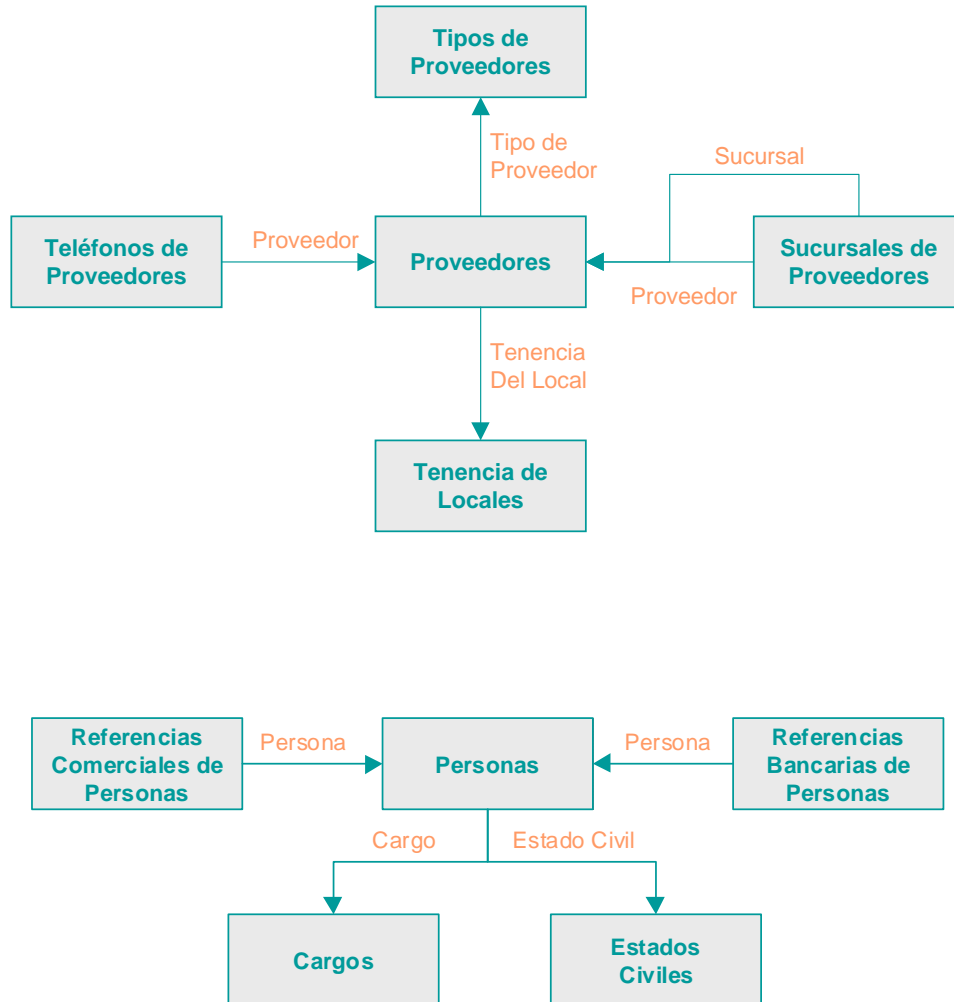
Entonces, una base de datos relacional queda determinada con el diseño de la cabecera de todas y cada una de sus relaciones (o tablas), las claves primarias de cada relación y las asociaciones entre distintas relaciones, establecidas al incorporar claves primarias de algunas como atributos de otras (claves ajenas).<sup>5</sup>

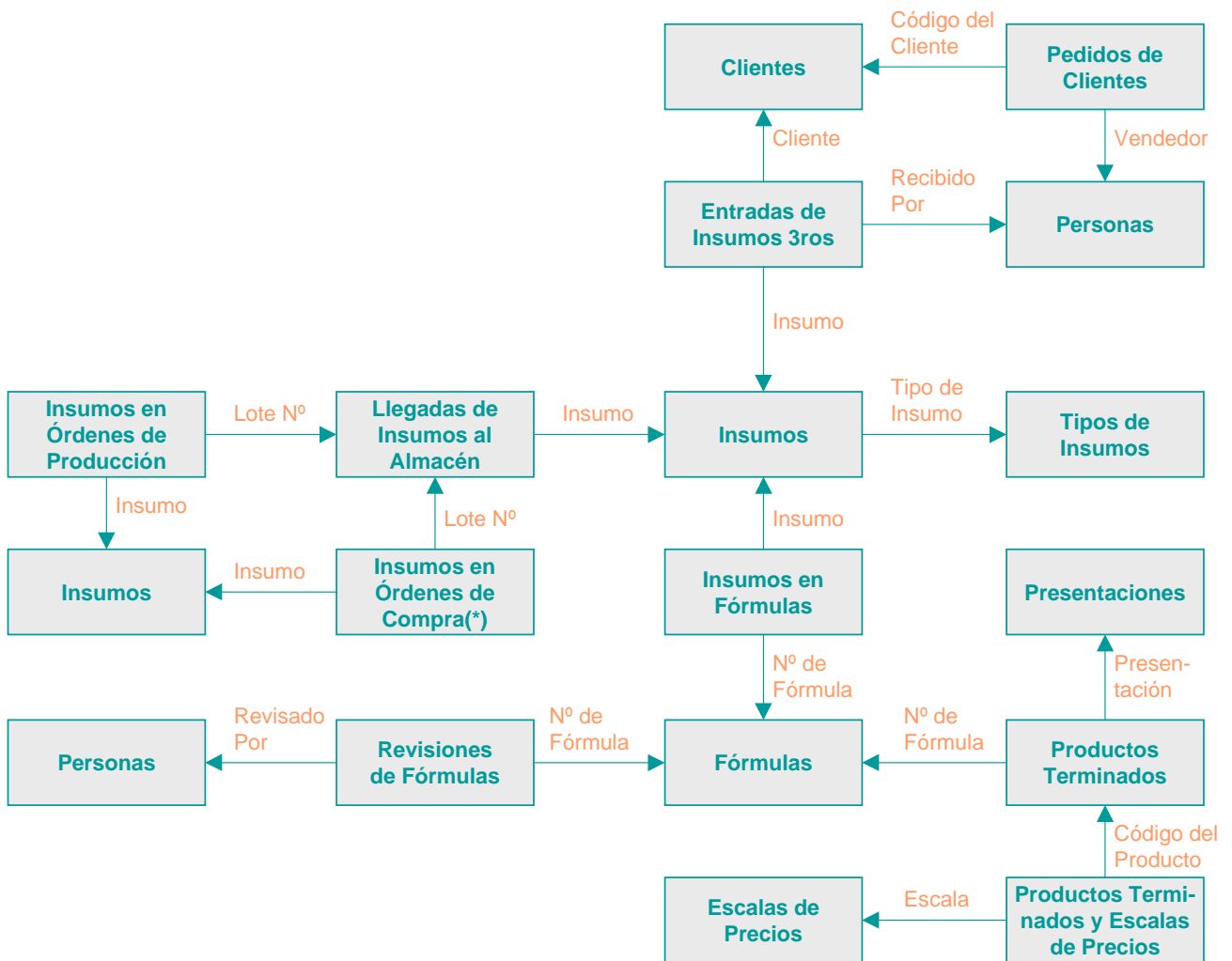
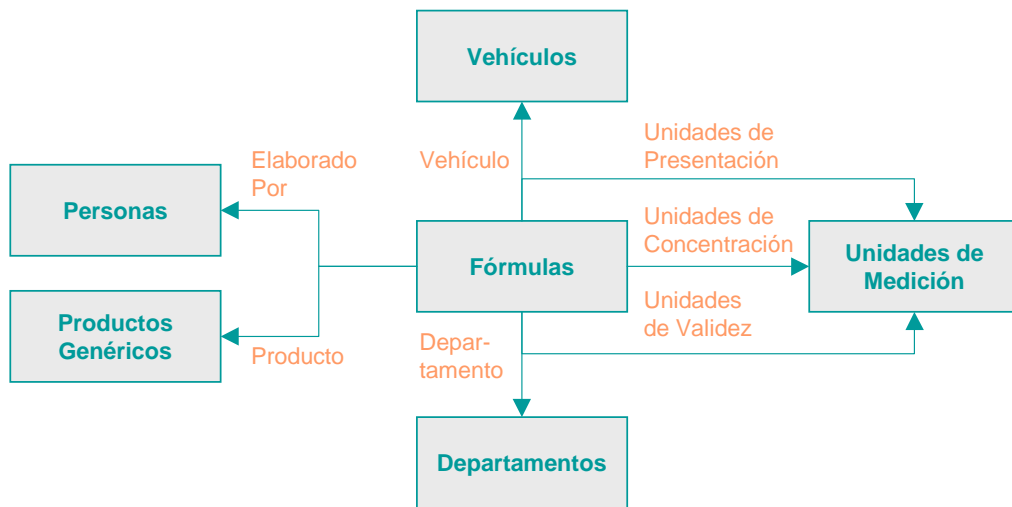
## II.1. Diagrama Referencial del SARCAI

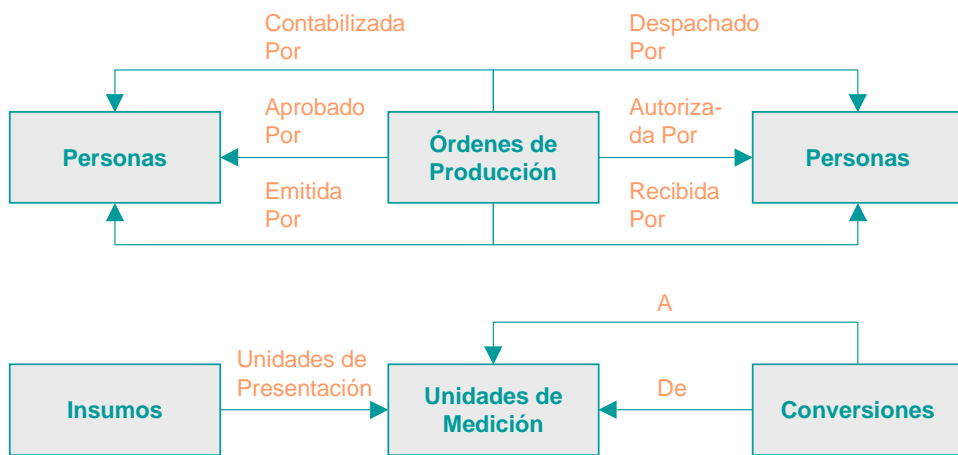
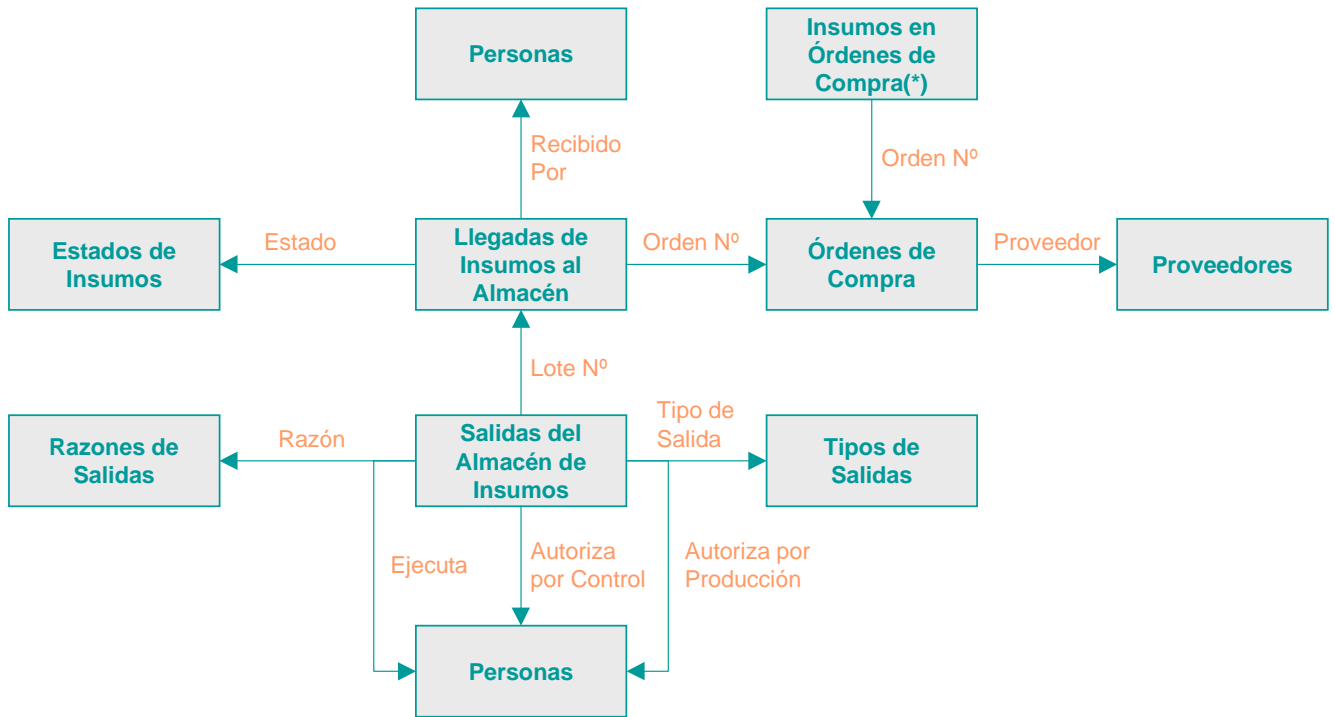


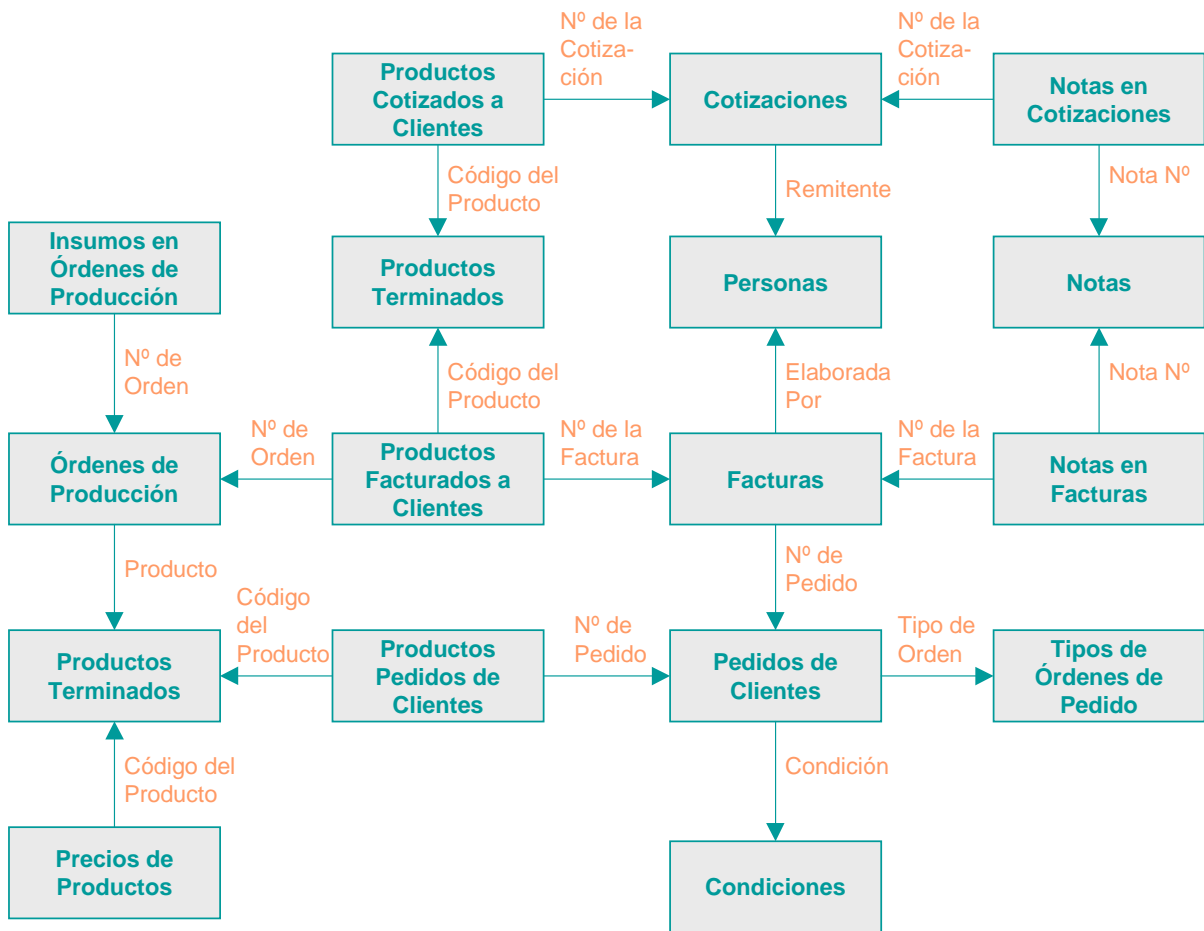
<sup>5</sup> Una discusión amplia del tema puede ser consultada en DATE, C.J. "Introducción a los Sistemas de Bases de Datos. Volumen 1. V Edición". Addison – Wesley Iberoamericana. México, 1993.











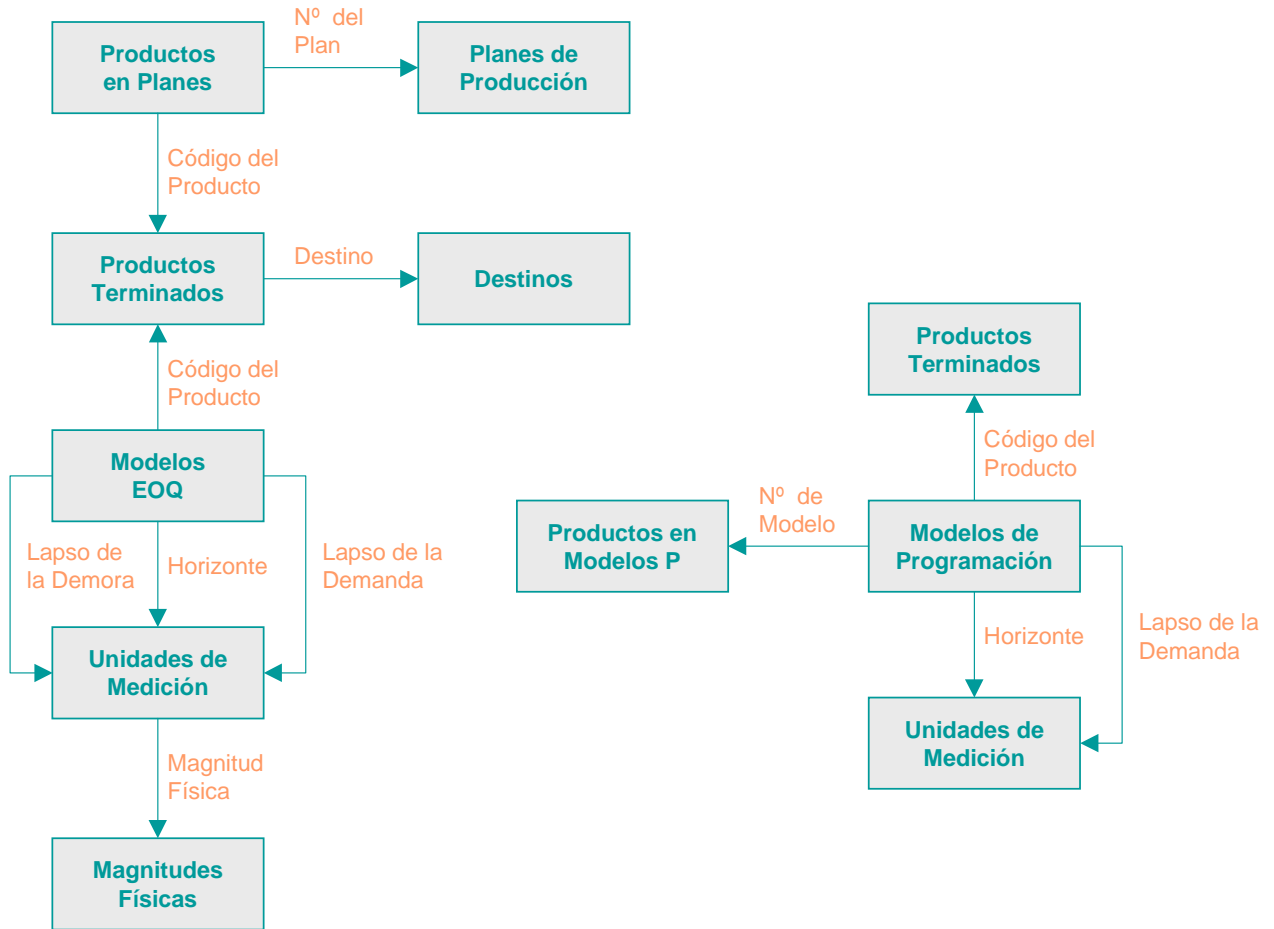


Figura 3. Diagrama Referencial del SARCAI

Un diagrama referencial como el que se muestra es la herramienta del Modelo Relacional de Bases de Datos para presentar las distintas relaciones entre las tablas que componen el diseño lógico de la base. En él, cada tabla es representada como un rectángulo en el que se incluye su nombre, y las relaciones entre ellas se representan por medio de flechas que parten de la tabla referencial y llegan a la tabla referenciada a través del campo que materializa la asociación. Así, por ejemplo, la sección:





indica que la tabla [Productos Facturados a Clientes], contiene un campo [N° de la Factura], que es en realidad conceptualmente la clave primaria de la tabla [Facturas]. Mediante esta combinación (denominada Clave Ajena), en una tupla de la tabla [Productos Facturados a Clientes] se conoce con exactitud todo lo que contenga la tabla [Facturas], ya que se conoce el valor de su clave primaria. Este diseño se utiliza en el ejemplo para definir una factura y los datos básicos asociados con ella en la tabla [Facturas] y definir los productos relativos a esa factura y sus cantidades en la tabla [Productos Facturados a Clientes] enlazada a través del [N° de la Factura] con la primera. En consecuencia, dado un número de Factura, se conoce de [Facturas], todo lo relativo al documento (fecha, cliente, etc.) y de [Productos Facturados a Clientes], todos aquellos productos que componen el documento.

El orden en que se muestren las tablas en este diagrama es totalmente irrelevante y se ha procurado conservar al máximo la lógica de presentación de los elementos del diagrama, aun cuando el criterio que prevalece es el de economía de espacio.

El lector podrá reconocer en el diagrama todas las asociaciones implementadas en SARCAI, omitiendo los detalles. A continuación, entonces, se presenta el diseño detallado de la base de datos.

## II.2. Diseño Detallado de la Base de Datos

A continuación se presentan las tablas y consultas, en orden alfabético, que componen la base de datos del SARCAI. Para las tablas se muestra el nombre, los campos que la componen, el tipo de datos de cada uno, el tamaño y el origen de la fila o restricciones (en algunos casos, la instrucción SQL que ejecuta el sistema para enlazar el campo en cuestión –clave ajena– con otra tabla o consulta). Se subrayan los campos que conforman la clave primaria de cada relación (sinónimo de tabla). La mayoría de las tablas y campos han sido nombrados mnemotécnicamente, por lo que no se supone requieran explicaciones adicionales, sin embargo, donde se crea conveniente se explicará algún elemento de interés. Las instrucciones `select` que aparecen en las tablas son también consultas diseñadas en la base de datos, pero de fácil comprensión, pues invocan otra tabla o conjunto de tablas y muestran parte de sus campos con la finalidad de que el usuario pueda escoger de entre estos valores y mantener así la regla de

Integridad Referencial. Posteriormente se incluyen las distintas consultas (el código SQL, una explicación de su funcionamiento y objetivos) que contiene esta versión del sistema. No hay que perder de vista que con la operación efectiva del SARCAI es posible que surjan muchas otras interrogantes que den lugar a que en futuras versiones se amplíe considerablemente el número de estas consultas a la base de datos.

## II.2.1. Tablas

### ACCIONISTAS DE CLIENTES

Campo	Tipo	Tamaño	Origen de la fila / Restricciones
<u>Cliente</u>	Texto	10	SELECT DISTINCTROW [Clientes].[Código del Cliente], [Clientes].[Nombre o Razón Social] FROM [Clientes];
<u>Accionista</u>	Texto	20	SELECT DISTINCTROW [Personas].[Cédula de Identidad o Clave], [Personas].[Apellidos], [Personas].[Nombres] FROM [Personas];

Esta tabla es una relación entre los clientes y las personas que se registran como accionistas de los mismos. Se establece tipeando el código del cliente ([Cliente]) y el código de la persona que es accionista de la empresa –en caso de ser pertinente– llamado aquí [Accionista]. Esta tabla permite registrar cualquier número de accionistas para un mismo cliente y, para todos los clientes. Los accionistas de los clientes se editan a través del subformulario [Accionistas] que se encuentra en el formulario [Clientes]. Se trata conceptualmente de personas, que pueden ser definidas en el mismo formulario, presionando el botón [Editar Personas] (ver Anexo 1, Figuras 2 y 3, pág. 5 y 6).

### CARGOS

Campo	Tipo	Tamaño	Origen de la fila / Restricciones
<u>Cargo</u>	Texto	60	

Contiene todos los cargos que puede tener una persona, sea o no empleado de La Planta. Antes de poder utilizar el dato sobre un cargo en alguna otra tabla, es necesario que esté definido aquí. Los cargos se editan a través del formulario [Cargos] (ver Anexo 1, Figura 1, pág. 5) al cual se accede presionando el botón [Editar Cargos] del formulario [Personas] (ver Anexo 1, Figura 38, pág. 76).

### CIUDADES DE ESTADOS

Campo	Tipo	Tamaño	Origen de la fila / Restricciones
País	Texto	50	SELECT DISTINCTROW [Países].[País] FROM [Países];
Estado	Texto	50	SELECT DISTINCTROW [Estados de Países].[Estado], [Estados de Países].[País] FROM [Estados de Países];
Ciudad	Texto	50	
Código Telefónico	Texto	10	

En esta tabla se definen las ciudades, inmersas dentro de los estados, inmersos a su vez en los países, que se utilizan en algunas otras tablas como parte de la dirección de las empresas o personas. También se registra aquí el código telefónico (supuesto único) de cada ciudad. Para poder registrar una ciudad, es necesario que esté registrado previamente el estado y el país donde se encuentra, en las tablas [Estados de Países] y [Países]. Las ciudades se editan a través del subsubformulario [Ciudades] inmerso dentro del subformulario [Estados], el cual a su vez se encuentra en el formulario [Países] (ver Anexo 1, Figura 29, pág. 50). Todos aquellos formularios que tengan elementos relacionados con países, estados y ciudades, contienen un botón [Editar Países, Estados y Ciudades] con el que se llega al formulario Países mencionado.

### CLIENTES

Campo	Tipo	Tamaño	Origen de la fila / Restricciones
Código del Cliente	Texto	10	
Nombre o Razón Social	Texto	80	
CI o RIF	Texto	20	
NIT	Texto	20	
Dirección	Texto	100	
País	Texto	50	SELECT DISTINCTROW [Países].[País] FROM [Países];
Estado	Texto	50	SELECT DISTINCTROW [Estados de Países].[Estado], [Estados de Países].[País] FROM [Estados de Países];
Ciudad	Texto	50	SELECT DISTINCTROW [Ciudades de Estados].[Ciudad], [Ciudades de Estados].[Estado], [Ciudades de Estados].[País] FROM [Ciudades de Estados];
Tenencia del Local	Texto	15	SELECT DISTINCTROW [Tenencia de Locales].[Tenencia] FROM [Tenencia de Locales];
Figura Jurídica	Texto	50	SELECT DISTINCTROW [Figuras Jurídicas].[Figura Jurídica] FROM [Figuras Jurídicas];
Nº del RC	Texto	50	
Fecha del RC	Fecha /Hora	8	
Duración del RC	Texto	50	
Tipo de Cliente	Texto	50	SELECT DISTINCTROW [Tipos de Clientes].[Tipo de Cliente] FROM [Tipos de Clientes];
Grupo	Texto	50	SELECT DISTINCTROW [Grupos de Clientes].[Grupo] FROM [Grupos de Clientes];
Sector	Texto	50	SELECT DISTINCTROW [Sectores Económicos].[Sector]

Campo	Tipo	Tamaño	Origen de la fila / Restricciones
			FROM [Sectores Económicos];
Observaciones	Memo	-	
Responsable	Texto	20	SELECT DISTINCTROW [Personas].[Cédula de Identidad o Clave], [Personas].[Apellidos], [Personas].[Nombres] FROM [Personas];
Fecha de Revisión	Fecha /Hora	8	

Esta tabla contiene los datos relativos a los clientes de la organización. Cada uno debe recibir un [Código del Cliente] que será su identificación única en todo el sistema. El campo [CI o RIF] debe recibir una cadena de caracteres que contendrá la Cédula de Identidad, en caso de que el cliente sea una persona natural, o el RIF en caso de que sea una persona jurídica o empresa. El campo [NIT] debe recibir una cadena de caracteres que represente el número de identificación tributaria que tiene el cliente. El campo [N° del RC] se refiere al número del Registro de Comercio que identifica al cliente (en caso de ser pertinente). [Fecha del RC] y [Duración del RC] son datos adicionales del Registro de Comercio. Los campos [País], [Estado], [Ciudad], [Tenencia del Local] (por ejemplo, si tiene local propio o alquilado), [Figura Jurídica] (por ejemplo, si se trata de una persona natural o jurídica), [Tipo de Cliente] (clasificación a ser definida por la organización), [Grupo] (también a ser definido), [Sector] (por ejemplo público, privado o mixto) y [Responsable] (persona encargada de mantener la relación con el cliente), requieren que exista el registro previo, en la tabla correspondiente, de la característica general que seleccionará el usuario utilizando la instrucción SQL. [Fecha de Revisión] se refiere a la última fecha en que fueron revisados los datos del cliente. Los clientes se editan a través del formulario [Clientes] (ver Anexo 1, Figuras 2 y 3, pág. 5 y 6). Todos aquellos formularios que tengan elementos relacionados con clientes contienen un botón [Editar Clientes] con el que se llega a este formulario.

#### CONDICIONES

Campo	Tipo	Tamaño	Origen de la fila / Restricciones
Condición	Texto	20	

En esta tabla se definen las distintas condiciones de venta que pueden presentarse, por ejemplo, crédito o contado. Las condiciones se editan a través del formulario [Condiciones] (ver Anexo 1, Figura 4, pág. 10) al cual se accede desde el formulario [Pedidos de Clientes] (ver Anexo 1, Figuras 30 y 31, pág. 52 y 53) presionando el botón [Editar Condiciones].

**CONTACTOS EN CLIENTES**

<b>Campo</b>	<b>Tipo</b>	<b>Tamaño</b>	<b>Origen de la fila / Restricciones</b>
<u>Cliente</u>	Texto	10	SELECT DISTINCTROW [Clientes].[Código del Cliente], [Clientes].[Nombre o Razón Social] FROM [Clientes];
<u>Contacto</u>	Texto	20	SELECT DISTINCTROW [Personas].[Cédula de Identidad o Clave], [Personas].[Apellidos], [Personas].[Nombres] FROM [Personas];

En esta tabla se registran las personas con las que la organización se entiende al tratar con cada cliente. El campo [Cliente] es una clave ajena que hace referencia a la tabla [Clientes] por el [Código del Cliente] y [Contacto] hace lo propio con la tabla [Personas] y [Cédula de Identidad o Clave]. Los contactos de los clientes se editan a través del subformulario [Contactos] que se encuentra en el formulario [Clientes]. Se trata conceptualmente de personas, que pueden ser definidas en el mismo formulario presionando el botón [Editar Personas] (ver Anexo 1, Figuras 2 y 3, pág. 5 y 6).

**CONTACTOS EN PROVEEDORES**

<b>Campo</b>	<b>Tipo</b>	<b>Tamaño</b>	<b>Origen de la fila / Restricciones</b>
<u>Proveedor</u>	Texto	10	SELECT DISTINCTROW [Proveedores].[Código del Proveedor], [Proveedores].[Nombre o Razón Social] FROM [Proveedores];
<u>Contacto</u>	Texto	20	SELECT DISTINCTROW [Personas].[Cédula de Identidad o Clave], [Personas].[Apellidos], [Personas].[Nombres] FROM [Personas];

Como en la anterior, en esta tabla se registran las personas con las que la organización se entiende al tratar con cada proveedor. Los campos tienen una connotación como la de antes. Los contactos en proveedores se editan a través del subformulario [Contactos] que se encuentra en el formulario [Proveedores]. Se trata conceptualmente de personas, que pueden ser definidas en el mismo formulario presionando el botón [Editar Personas] (ver Anexo 1, Figuras 43 y 44, pág. 86 y 87).

**CONVERSIONES**

<b>Campo</b>	<b>Tipo</b>	<b>Tamaño</b>	<b>Origen de la fila / Restricciones</b>
<u>De</u>	Texto	10	SELECT DISTINCTROW [Unidades Convertibles].[Unidades] FROM [Unidades Convertibles];
<u>A</u>	Texto	10	SELECT DISTINCTROW [Unidades Convertibles].[Unidades] FROM [Unidades Convertibles];
Ponderador	Número (simple)	4	

Esta tabla permite definir conversiones entre las distintas unidades de medición utilizadas a lo largo del sistema. El campo [De] acepta la unidad de medida cuya conversión se desea definir, el

campo [A] acepta la unidad de medida a donde se desea convertir y el campo [Ponderador] acepta el número, que multiplicado por la cantidad de la unidad [De] transforma esta unidad en la unidad [A]. Por ejemplo, si se deseara definir la forma de conversión de kilogramos a gramos, debería tipearse en [De], Kg. en [A], g. y en [Ponderador], 0.001, pues para convertir Kg. a g. se debe multiplicar la cantidad en kilogramos por 0.001. Todas las conversiones que utiliza el sistema deben estar definidas previamente en esta tabla, incluyendo las conversiones de una unidad a sí misma (que requieren un ponderador igual a 1). Las conversiones se editan a través del formulario [Conversiones] (ver Anexo 1, Figura 5, pág. 11), al cual se accede donde sea pertinente presionando el botón [Editar Conversiones].

#### COTIZACIONES

<b>Campo</b>	<b>Tipo</b>	<b>Tamaño</b>	<b>Origen de la fila / Restricciones</b>
Nº de la Cotización	Número (largo) Autoincremento	4	
Código del Cliente	Texto	50	SELECT DISTINCTROW [Clientes].[Código del Cliente], [Clientes].[Nombre o Razón Social], [Clientes].[CI o RIF] FROM [Clientes];
Fecha de Emisión	Fecha/Hora	8	
Fecha de Envío	Fecha/Hora	8	
Remitente	Texto	50	SELECT DISTINCTROW [Personas].[Cédula de Identidad o Clave], [Personas].[Apellidos], [Personas].[Nombres] FROM [Personas];

Guarda los datos relativos a las cotizaciones o presupuestos entregados a los clientes que así lo solicitan. Cada cotización recibe automáticamente un número único secuencial que la identifica. Entre otras cosas se registra el cliente ([Código del Cliente]) al que se le está redactando la cotización. El campo [Remitente] hace referencia al empleado de La Planta que elabora el documento. En la tabla [Productos Cotizados a Clientes], que se definirá posteriormente, se incluyen entonces los bienes que se cotizan. Las cotizaciones se editan a través del formulario [Cotizaciones] (ver Anexo 1, Figura 6, pág. 12).

#### DEMANDA INTERCAMBIO EXCEL

<b>Campo</b>	<b>Tipo</b>	<b>Tamaño</b>	<b>Origen de la fila / Restricciones</b>
Nº de Modelo	Número (largo)	4	SELECT DISTINCTROW [Modelos de Programación].[Nº de Modelo], [Modelos de Programación].[Código del Producto] FROM [Modelos de Programación];
Período	Número (entero)	2	
Demanda	Número	4	

<b>Campo</b>	<b>Tipo</b>	<b>Tamaño</b>	<b>Origen de la fila / Restricciones</b>
	(largo)		
Costo de Producción	Moneda	8	
Costo de Mantener	Moneda	8	
Costo de Escasez	Moneda	8	

Esta tabla registra los cálculos que realiza el algoritmo de Modelos de Producción que luego Excel recupera e incluye en el modelo de programación lineal que pronostica el comportamiento del inventario. [N° de Modelo] es una clave ajena que enlaza esta tabla con la tabla [Modelos de Programación], donde se definen los parámetros de cada modelo ensayado. En base a los parámetros del modelo, como el horizonte de planeación, los costos asociados o la inflación, se calculan los demás campos. Aun cuando estos campos son por definición calculados, el usuario puede editarlos si le parece que alguna de las estimaciones no es apropiada. Esta tabla no es una exigencia del modelo conceptual de la base de datos, se necesita sólo para cálculos (Ver Capítulo III).

#### DEMANDA INTERCAMBIO EXCEL MODELO

<b>Campo</b>	<b>Tipo</b>	<b>Tamaño</b>	<b>Origen de la fila / Restricciones</b>
Límite de Producción	Número (largo)	4	
Inventario Inicial	Número (largo)	4	
N° de Registros	Número (entero)	2	

Al igual que antes, esta tabla se utiliza para que Excel tome los datos del Modelo de Producción que necesita para el cálculo. En este caso, los datos no se refieren a la demanda, sino a restricciones generales del modelo. Esta tabla tampoco es una exigencia del modelo conceptual de la base de datos, se necesita sólo para cálculos. En general contendrá un solo registro por lo que no es necesario definir una clave primaria. (Ver Capítulo III).

#### DEMANDA POR FECHAS TABLA

<b>Campo</b>	<b>Tipo</b>	<b>Tamaño</b>	<b>Origen de la fila / Restricciones</b>
Producto	Texto	10	
Fecha Base	Fecha/Hora	8	
Demanda Puntual	Número (doble)	8	
Demanda en Bs.	Número (doble)	8	

Calculada a partir de una consulta SQL, donde se resumen algunos datos necesarios para la realización de gráficos e informes, relacionados con la demanda de productos. Tampoco es una exigencia del modelo conceptual de la base de datos, se necesita sólo para cálculos. Esta tabla

se edita automáticamente en base a los parámetros introducidos por el usuario en el formulario [Pedir Fechas para la Demanda] (ver Anexo 1, Figura 34, pág. 64).

#### DEPARTAMENTOS

Campo	Tipo	Tamaño	Origen de la fila / Restricciones
<u>Departamento</u>	Texto	20	

En ésta se definen los distintos departamentos de fármacos presentes en la organización, por ejemplo, sólidos, líquidos, antibióticos, etc. Todos los departamentos han de definirse aquí antes de poder incluirse en otras secciones de la base de datos. Los departamentos se editan a través del formulario [Departamentos] (ver Anexo 1, Figura 7, pág. 15), al cual se accede desde el formulario [Fórmulas] (ver Anexo 1, Figuras 18 y 19, pág. 25 y 26) presionando el botón [Editar Departamentos].

#### DESTINOS

Campo	Tipo	Tamaño	Origen de la fila / Restricciones
<u>Destino</u>	Texto	20	

Aquí se definen los destinos para los que se fabrican los fármacos, por ejemplo, venta, muestra médica, Ministerio de Sanidad y Asistencia Social (MSAS), etc. Todos ellos han de colocarse aquí antes de poder incluirse en otras secciones. Los destinos se editan a través del formulario [Destinos] (ver Anexo 1, Figura 8, pág. 15), al cual se accede desde el formulario [Productos Terminados] (ver Anexo 1, Figura 42, pág.82) presionando el botón [Editar Destinos].

#### DOCUMENTACIÓN

Campo	Tipo	Tamaño	Origen de la fila / Restricciones
<u>Documento</u>	Texto	255	
<u>Dirección</u>	Hipervínculo	-	

Esta tabla contiene el nombre y la dirección (en formato de Internet), de la documentación en línea que el usuario puede consultar cuando opere con el sistema. Esta es una tabla no editable, el usuario la utiliza sólo cuando desea una referencia a un documento de los redactados para SARCAI y da clic sobre el hipervínculo. Los documentos se muestran a partir del formulario [Documentación] (ver Anexo 1, Figura 9, pág. 16).



**ELEMENTOS DEL PANEL DE CONTROL**

<b>Campo</b>	<b>Tipo</b>	<b>Tamaño</b>	<b>Origen de la fila / Restricciones</b>
<u>SwitchboardID</u>	Número (largo)	4	
<u>ItemNumber</u>	Número (entero)	2	
ItemText	Texto	255	
Command	Número (entero)	2	
Argument	Texto	50	

Access define y llena de datos automáticamente esta tabla. En ella se guardan las distintas opciones de los paneles que componen el sistema, realizados con la herramienta Administrador de Panel de Control, que incluye software. Bajo ninguna circunstancia se debe modificar el contenido de esta tabla.

**ENTRADAS DE INSUMOS 3ROS**

<b>Campo</b>	<b>Tipo</b>	<b>Tamaño</b>	<b>Origen de la fila / Restricciones</b>
<u>Lote N°</u>	Número (largo) Autoincremento	4	
Fecha	Fecha/Hora	8	
Cliente	Texto	10	SELECT DISTINCTROW [Clientes].[Código del Cliente], [Clientes].[Nombre o Razón Social] FROM [Clientes];
Insumo	Texto	10	SELECT DISTINCTROW [Insumos].[Código de Insumo], [Insumos].[Nombre] FROM [Insumos];
Cantidad	Moneda	8	>=0
Estado	Texto	15	SELECT DISTINCTROW [Estados de Insumos].[Estado] FROM [Estados de Insumos];
Observaciones	Memo	-	
Recibido Por	Texto	50	SELECT DISTINCTROW [Personas].[Cédula de Identidad o Clave], [Personas].[Apellidos], [Personas].[Nombres] FROM Personas;

Aquí se registran los datos correspondientes a cada entrada de insumos que serán empleados en la producción por encargo de terceras personas (clientes). Cada entrada se lotifica con un número secuencial que se establece automáticamente. Los campos [Cliente], [Insumo], [Estado] y [Recibido Por] son claves ajenas a sus respectivas tablas maestras, por lo tanto deben contar con las definiciones apropiadas antes de poder insertarse en esta tabla. Las entradas de insumos a terceros se editan a través del formulario [Entradas de Insumos 3ros] (ver Anexo 1, Figuras 10 y 11, pág. 16 y 17).

**ESCALAS DE PRECIOS**

<b>Campo</b>	<b>Tipo</b>	<b>Tamaño</b>	<b>Origen de la fila / Restricciones</b>
<u>Escala</u>	Número (largo)	4	

Campo	Tipo	Tamaño	Origen de la fila / Restricciones
	Autoincremento		
Descripción	Texto	50	
Límite Inferior	Número (largo)	4	>=0
Límite Superior	Número (largo)	4	>=0
% Descuento	Número (simple)	4	>=0 Y <=1

Aquí se pueden definir todas las escalas de precios, por cantidad, que se desee. En realidad, el concepto que se utiliza en esta tabla es la aplicación de un porcentaje de descuento al precio de cada producto. Para definir una escala de precios se asigna primero una clave numérica secuencial automáticamente, se establece entonces una cadena de caracteres que describa o nombre la escala, se colocan las cantidades del artículo por encima y por debajo de las cuales, rige el descuento y se tipea el descuento como un número porcentual. Esta tabla se relacionará luego con los precios de los productos normalizando las políticas de descuento de la organización, de acuerdo con la cantidad que compre cada cliente. Las escalas de precios se editan a través del formulario [Escala de Precios] (ver Anexo 1, Figura 12, pág. 19) al cual se accede desde el formulario [Productos Terminados] (ver Anexo 1, Figura 42, pág.82) presionando el botón [Editar Escalas de Precios].

#### ESTADOS CIVILES

Campo	Tipo	Tamaño	Origen de la fila / Restricciones
Estado Civil	Texto	15	

Permite la definición de los distintos estados civiles de las personas, por ejemplo, Casado, Soltero, Viudo, etc. Los estados civiles se editan a través del formulario [Estados Civiles] (ver Anexo 1, Figura 13, pág. 19), al cual se accede desde el formulario [Personas] (ver Anexo 1, Figura 38, pág.76) presionando el botón [Editar Estados Civiles].

#### ESTADOS DE INSUMOS

Campo	Tipo	Tamaño	Origen del Registro / Restricciones
Estado	Texto	15	
¿Es Utilizable?	Sí/No	1	

Aquí es posible definir los distintos estados en que puede encontrarse un insumo (fundamentalmente utilizados cuando se registra la llegada del insumo). Adicionalmente para cada estado, el usuario completa un campo de tipo lógico donde se informa si el estado mencionado permite que el insumo se utilice o, por el contrario, lo prohíbe. Por ejemplo, un

posible estado es En Cuarentena, el cual no permite la utilización del insumo. Los estados de insumos se editan a través del formulario [Estados de Insumos] (ver Anexo 1, Figura 14, pág. 20) al cual se accede desde donde se les necesite presionando el botón [Editar Estados de Insumos].

#### ESTADOS DE PAÍSES

Campo	Tipo	Tamaño	Origen del Registro / Restricciones
País	Texto	50	SELECT DISTINCTROW [Países].[País] FROM [Países];
Estado	Texto	50	

Aquí se definen todos los estados de interés inmersos dentro de los países (previamente definidos) para efectos de dirección o números telefónicos en otras secciones de la base de datos. En esta tabla se contempla la posibilidad de que en países distintos existan estados con el mismo nombre, sin embargo, la combinación país-estado no se puede repetir. Los estados se editan a través del subformulario [Estados] inmerso en el formulario [Países] (ver Anexo 1, Figura 29, pág. 50). Todos aquellos formularios que tengan elementos relacionados con países, estados y ciudades, contienen un botón [Editar Países, Estados y Ciudades] con el que se llega al formulario Países mencionado.

#### FACTURAS

Campo	Tipo	Tamaño	Origen del Registro / Restricciones
Nº de Factura	Número (largo) Autoincremento	4	
Nº de Pedido	Número (largo)	4	SELECT DISTINCTROW [Pedidos de Clientes].[Nº de Pedido], [Pedidos de Clientes].[Código del Cliente], [Pedidos de Clientes].[Fecha], [Pedidos de Clientes].[Vendedor], [Pedidos de Clientes].[Condición] FROM [Pedidos de Clientes];
Fecha de Emisión	Fecha/Hora	8	
Elaborada Por	Texto	50	SELECT DISTINCTROW [Personas].[Cédula de Identidad o Clave], [Personas].[Apellidos], [Personas].[Nombres] FROM [Personas];
Fecha de Despacho	Fecha/Hora	8	

En esta tabla se registra la información básica de cada factura redactada. Una factura es el documento que acompaña la venta de mercancía. Se maneja como la constancia de que se realizó una venta, lo que se transó y el monto que debe cancelar el cliente por la transacción. Adicionalmente, la elaboración de facturas y la colocación de las fechas de emisión y despacho afectan automáticamente los niveles del inventario (por lo que deben ser utilizadas con cuidado). El campo [Nº de Factura] es colocado por el sistema automáticamente para cada una de ellas;

toda factura debe estar precedida por un pedido cuyo número se registra en el campo [N° de Pedido]; esta relación es crucial para el inventario, pues no se tratan de la misma forma las ventas (facturadas) que la demanda (los pedidos). No todo pedido ha de ser facturado, pero toda factura ha de partir de un pedido. Esta tabla actúa conjuntamente con [Productos Facturados a Clientes], donde se detallan los productos pedidos y facturados a los clientes. Las facturas se editan a través del formulario [Facturas] (ver Anexo 1, Figuras 15 y 16, pág. 20 y 21).

#### FIGURAS JURÍDICAS

Campo	Tipo	Tamaño	Origen del Registro / Restricciones
Figura Jurídica	Texto	50	

Aquí se definen las distintas figuras jurídicas de clientes o proveedores posibles. Por ejemplo, Persona Natural, Persona Jurídica y otras. Las figuras jurídicas se editan a través del formulario [Figuras Jurídicas] (ver Anexo 1, Figura 17, pág. 24), al cual se accede desde donde se le necesite presionando el botón [Editar Figuras Jurídicas].

#### FÓRMULAS

Campo	Tipo	Tamaño	Origen del Registro / Restricciones
N° de Fórmula	Texto	10	
Producto	Texto	30	SELECT DISTINCTROW [Productos Genéricos].Producto FROM [Productos Genéricos];
Vehículo	Texto	20	SELECT DISTINCTROW [Vehículos].[Vehículo] FROM [Vehículos];
Unidades de Presentación	Texto	10	SELECT DISTINCTROW [Unidades de Presentación].[Unidades], [Unidades de Presentación].[Magnitud Física] FROM [Unidades de Presentación];
Concentración	Moneda	8	>=0
Unidades de Concentración	Texto	10	SELECT DISTINCTROW [Unidades de Concentración].[Unidades], [Unidades de Concentración].[Magnitud Física] FROM [Unidades de Concentración];
Departamento	Texto	20	SELECT DISTINCTROW [Departamentos].[Departamento] FROM [Departamentos];
Fecha de Emisión	Fecha/Hora	8	
Validez	Moneda	8	>=0
Unidades de Validez	Texto	10	SELECT DISTINCTROW [Unidades de Medición].[Unidades], [Unidades de Medición].[Magnitud Física] FROM [Unidades de Medición];
Registro Sanitario	Texto	30	
Técnica de	Memo	-	

<b>Campo</b>	<b>Tipo</b>	<b>Tamaño</b>	<b>Origen del Registro / Restricciones</b>
Elaboración			
Tamaño del Lote	Moneda	8	>=0
Elaborado por	Texto	20	SELECT DISTINCTROW [Personas].[Cédula de Identidad o Clave], [Personas].[Apellidos], [Personas].[Nombres] FROM [Personas];

Una fórmula es el centro del proceso productivo. Siguiendo las fórmulas se elaboran los productos. Esta tabla permite describir cada fórmula asignándole en primer lugar un [N° de Fórmula] único. El campo [Producto] es una clave ajena que referencia los productos genéricos de donde se debe escoger uno que sea la base de la fórmula. El [Vehículo] no es aplicable en todos los casos, sólo es pertinente cuando se trata de jarabes y similares; el usuario decide en cada fórmula si incorpora o no este dato. El campo [Unidades de Presentación] recibe la información de la presentación del fármaco, por ejemplo, tabletas, kilogramos, frascos, etc. La [Concentración] es un número que describe esta variable química para el fármaco. Las [Unidades de Concentración] son las unidades de medición en que se expresa la concentración, por ejemplo, 50 mg., 250 ml., etc. El [Departamento] expresa el tipo de fármaco que se está formulando, por ejemplo, Sólidos, Líquidos, Antibióticos, etc. La [Validez] es un número que expresa el tiempo de duración que tendrá el medicamento formulado. [Unidades de Validez] aceptan unidades de medición del tiempo que acompañan a la validez del producto, por ejemplo, 2 Años, 18 Meses, etc. [Registro Sanitario] es una cadena de caracteres donde se puede insertar el número de registro sanitario de la fórmula. El campo [Técnica de Elaboración] es una cadena de caracteres amplia (aproximadamente admite hasta 32 Kb. de texto) donde el usuario describe detalladamente la técnica como se debe elaborar el producto formulado. El [Tamaño del Lote] es un dato numérico que dice la cantidad de unidades (de presentación) del fármaco que deben ser producidas en cada corrida de producción, por ejemplo, 100000 tabletas, 25000 frascos, etc. El campo [Elaborado Por] admite una clave de [Personas] que representa a la persona que elaboró la fórmula (y la registró) por primera vez. Esta tabla trabaja conjuntamente con [Insumos en Fórmulas] donde se listan los distintos insumos y cantidades necesarias para completar el producto formulado. Las fórmulas se editan a través del formulario [Fórmulas] (ver Anexo 1, Figuras 18 y 19, pág. 25 y 26).

**GRÁFICO DE PRODUCTOS ESTRELLA T1**

<b>Campo</b>	<b>Tipo</b>	<b>Tamaño</b>	<b>Origen del Registro / Restricciones</b>
<u>Producto</u>	Texto	255	
Total Ventas	Número (doble)	8	

Esta tabla es necesaria para la construcción del informe que muestra los principales productos de acuerdo con las ventas. No pertenece al conjunto de las tablas básicas (que conforman el modelo de la base de datos).

**GRÁFICO DE PRODUCTOS ESTRELLA T2**

<b>Campo</b>	<b>Tipo</b>	<b>Tamaño</b>	<b>Origen del Registro / Restricciones</b>
<u>Producto</u>	Texto	255	
Total Ventas	Número (doble)	8	

Al igual que la anterior, esta tabla es necesaria para la construcción del informe que muestra los principales productos de acuerdo con las ventas. Sus datos se calculan automáticamente a partir de consultas SQL elaboradas con base en el formulario [Pedir Fechas para Ventas por Productos] (ver Anexo 1, Figura 36, pág. 71).

**GRUPOS DE CLIENTES**

<b>Campo</b>	<b>Tipo</b>	<b>Tamaño</b>	<b>Origen del Registro / Restricciones</b>
<u>Grupo</u>	Texto	50	

Aquí se definen los distintos grupos de clientes que el usuario tenga a bien formar. Se utilizan posteriormente estos grupos en la tabla [Clientes] y se producen gráficos e informes asociados con este agrupamiento. Los grupos de clientes se editan a través del formulario [Grupos de Clientes] (ver Anexo 1, Figura 20, pág. 30), al cual se accede desde el formulario [Clientes] (ver Anexo 1, Figuras 2 y 3, pág. 5 y 6) presionando el botón [Editar Grupos].

**INSUMOS**

<b>Campo</b>	<b>Tipo</b>	<b>Tamaño</b>	<b>Origen del Registro / Restricciones</b>
<u>Código de Insumo</u>	Texto	10	
Nombre	Texto	80	
Tipo de Insumo	Texto	50	SELECT DISTINCTROW [Tipos de Insumos].[Tipo de Insumo] FROM [Tipos de Insumos];
Presentación Mínima	Moneda	8	
Unidades de Presentación	Texto	10	SELECT DISTINCTROW [Unidades de Presentación].[Unidades], [Unidades de Presentación].[Magnitud Física] FROM [Unidades de

Campo	Tipo	Tamaño	Origen del Registro / Restricciones
			Presentación];
Condiciones de Entrega	Memo	-	

En esta tabla se definen los distintos insumos que se utilizan en la producción. Cada insumo recibe del usuario un código único. El [Tipo de Insumo] es una clave ajena que referencia los distintos tipos de insumos definidos, por ejemplo, Principio Activo, Excipiente y otros. La [Presentación Mínima] se acompaña de las [Unidades de Presentación] para indicar en qué condiciones se puede obtener el insumo en el mercado, por ejemplo, 1 Kg., 200 g. etc. El campo [Condiciones de Entrega] permite describir en que condiciones se espera sea entregado el insumo, generalmente para efectos de control de calidad. Los insumos se editan a través del formulario [Insumos] (ver Anexo 1, Figura 21, pág. 34) al cual se accede desde donde se le necesite presionando el botón [Editar Insumos].

#### INSUMOS EN FÓRMULAS

Campo	Tipo	Tamaño	Origen del Registro / Restricciones
Nº de Fórmula	Texto	50	SELECT DISTINCTROW [Fórmulas].[Nº de Fórmula], [Fórmulas].[Producto], [Fórmulas].[Vehículo], [Fórmulas].[Concentración], [Fórmulas].[Unidades de Concentración] FROM [Fórmulas];
Código de Insumo	Texto	50	SELECT DISTINCTROW Insumos.[Código de Insumo], Insumos.Nombre FROM Insumos ORDER BY Insumos.Nombre;
Cantidad	Moneda	8	>0

Esta tabla opera conjuntamente con [Fórmulas] para la definición de la cantidad del insumo necesaria, para producir el tamaño del lote estándar establecido en la fórmula. Para una misma fórmula se deben definir todos los distintos insumos necesarios y las cantidades de cada uno, medidas en las unidades de presentación del insumo. Así, [Nº de Fórmula] es una clave ajena que referencia a [Fórmulas] y [Código de Insumo] lo es, referenciando a [Insumos]. Los insumos en fórmulas se editan a través del subformulario [Insumos] que se encuentra en el formulario [Fórmulas]. Pueden ser definidos en el mismo formulario presionando el botón [Editar Insumos] (ver Anexo 1, Figuras 18 y 19, pág. 25 y 26).

#### INSUMOS EN ÓRDENES

Campo	Tipo	Tamaño	Origen del Registro / Restricciones
Orden Nº	Número (largo)	4	SELECT DISTINCTROW [Órdenes de Compra].[Nº], [Órdenes de Compra].[Proveedor], [Órdenes de Compra].[Fecha] FROM [Órdenes de Compra];

<b>Campo</b>	<b>Tipo</b>	<b>Tamaño</b>	<b>Origen del Registro / Restricciones</b>
<u>Insumo</u>	Texto	10	SELECT DISTINCTROW [Insumos].[Código de Insumo], [Insumos].[Nombre] FROM [Insumos];
Precio Unitario	Moneda	8	>0
Cantidad	Moneda	8	>0

Esta tabla opera en conjunto con [Órdenes de Compra]. Se refiere a los insumos que se ordena comprar a los proveedores en cada orden. El campo [Orden N°] es una clave ajena que hace referencia a la tabla [Órdenes de Compra] (que contiene previamente los datos generales de la orden), el campo [Insumo] es una clave ajena que hace referencia a la tabla [Insumos] a través del [Código de Insumo], el campo [Precio Unitario] permite al usuario definir el precio de cada insumo que está comprando en la orden y la [Cantidad] expresa el número de unidades (de presentación) del insumo que se compran. Los insumos en órdenes se editan a través del subformulario [Insumos Solicitados] que se encuentra en el formulario [Órdenes de Compra]. Pueden ser definidos en el mismo formulario presionando el botón [Editar Insumos] (ver Anexo 1, Figura 26, pág. 42).

#### INSUMOS EN ÓRDENES DE PRODUCCIÓN

<b>Campo</b>	<b>Tipo</b>	<b>Tamaño</b>	<b>Origen del Registro / Restricciones</b>
<u>N° de Orden</u>	Número (largo)	4	SELECT DISTINCTROW [Órdenes de Producción].[N° de Orden], [Órdenes de Producción].[Producto] FROM [Órdenes de Producción];
<u>Código de Insumo</u>	Texto	50	SELECT DISTINCTROW Insumos.[Código de Insumo], Insumos.Nombre FROM Insumos ORDER BY Insumos.Nombre;
<u>Lote N°</u>	Número (largo)	4	SELECT DISTINCTROW [Compras de Insumos].[Lote N°], [Compras de Insumos].[Fecha] FROM [Compras de Insumos];
Cantidad a Utilizar	Moneda	8	
Fecha del Surtido	Fecha/Hora	8	

En este caso se opera en conjunto con [Órdenes de Producción]. Se refiere a los insumos que se solicitan para iniciar una corrida de producción de un artículo. El campo [N° de Orden] es una clave ajena que hace referencia a la tabla [Órdenes de Producción] (que contiene previamente los datos generales de la orden), el campo [Código de Insumo] es una clave ajena que hace referencia a la tabla [Insumos], el campo [Lote N°] registra el lote del que se espera sea surtido el insumo solicitado. Este lote se establece en la llegada del insumo al almacén de insumos. La [Cantidad a Utilizar] muestra el número de unidades (de presentación) del insumo que se solicitan y [Fecha del Surtido] registra la fecha en la cual se recibe el insumo del almacén. La



estructura de esta tabla permite que para una misma orden se registren varios insumos iguales pero tomados de distintos lotes. La razón estriba en que es necesario controlar el lote (y aspectos como la potencia) del insumo que se utiliza en la producción. Los insumos en órdenes de producción se editan a través del subformulario [Insumos Requeridos] que se encuentra en el formulario [Órdenes de Producción] (ver Anexo 1, Figuras 27 y 28, pág. 45 y 46). El sistema los define automáticamente a partir de la fórmula, una vez que el usuario selecciona el producto a fabricar, por lo cual no hace falta editar los insumos en este formulario.

#### LLEGADAS DE INSUMOS AL ALMACÉN

Campo	Tipo	Tamaño	Origen del Registro / Restricciones
Lote N°	Número (largo) Autoincremento	4	
Fecha	Fecha/Hora	8	
Orden N°	Número (largo)	4	SELECT DISTINCTROW [Órdenes de Compra].[N°], [Órdenes de Compra].[Proveedor] FROM [Órdenes de Compra];
Insumo	Texto	10	SELECT DISTINCTROW [Insumos].[Código de Insumo], [Insumos].[Nombre] FROM [Insumos];
Cantidad	Moneda	8	>=0
Potencia	Número (simple)	4	
Estado	Texto	15	SELECT DISTINCTROW [Estados de Insumos].[Estado] FROM [Estados de Insumos];
Observaciones	Memo	-	
Recibido Por	Texto	50	SELECT DISTINCTROW [Personas].[Cédula de Identidad o Clave], [Personas].[Apellidos], [Personas].[Nombres] FROM [Personas];

Aquí se registran los arribos de insumos para la producción, provenientes de los proveedores y mediando una orden de compra. Cada vez que ocurre una llegada, automáticamente se lotifica con un número secuencial. Este número será el identificador único de cada llegada. El campo [Orden N°] acepta sólo números válidos de la tabla [Órdenes de Compra], donde se supone fue solicitado el insumo que está llegando. [Insumo] registra el [Código del Insumo] que llega, [Cantidad] registra el número de unidades (de presentación) que llegan del insumo (inferior o igual a lo solicitado), [Potencia] registra un porcentaje, calculado por Control de Calidad, indicador de la pureza del insumo -donde es aplicable-, [Estado] permite registrar el estado en que se encuentra el insumo (por ejemplo, generalmente, al llegar es En Cuarentena, y una vez analizado por Control de Calidad, el estado puede pasar a En Almacén). Aunque una llegada, por lo general, incluye varios insumos distintos, en esta tabla se registra cada uno de los insumos que

llegan por separado (para efectos de control y lotificación). Es delicada la información que se registre aquí, pues afecta directamente el nivel de inventario. Las llegadas de insumos al almacén se editan a través del formulario [Llegadas de Insumos al Almacén] (ver Anexo 1, Figuras 22 y 23, pág. 36 y 37).

#### MAGNITUDES FÍSICAS

Campo	Tipo	Tamaño	Origen del Registro / Restricciones
Magnitud Física	Texto	15	

Esta tabla registra las distintas magnitudes que se utilizan en el sistema. Una magnitud física es el grupo al que pueden pertenecer las unidades de medida, por ejemplo, Tiempo, Peso, Volumen, Unidades, etc. Las magnitudes físicas se editan a través del formulario [Magnitudes Físicas] (ver Anexo 1, Figura 24, pág. 41), al cual se accede desde el formulario [Unidades de Medición] (ver Anexo 1, Figura 53, pág. 93) presionando el botón [Editar Magnitudes Físicas].

#### MESES DEL AÑO

Campo	Tipo	Tamaño	Origen del Registro / Restricciones
Código	Número (Byte)	1	
Mes	Texto	10	

Esta es una tabla accesoria utilizada para diagramar algunos informes y gráficos relacionados con pedidos y ventas. Se definen aquí los meses del año en la forma, por ejemplo, 1 : Enero, 2 : Febrero, etc.

#### MODELOS DE PROGRAMACIÓN

Campo	Tipo	Tamaño	Origen del Registro / Restricciones
Nº de Modelo	Número (largo) Autoincremento	4	
Código del Producto	Texto	10	SELECT DISTINCTROW [Productos Terminados].[Código del Producto], [Productos Terminados].[Nº de Fórmula] FROM [Productos Terminados];
Límite de Producción	Número (largo)	4	
Descripción	Memo	-	
Desde	Fecha/Hora	8	
Horizonte	Texto	50	SELECT DISTINCTROW [Unidades de Medición].[Unidades], [Unidades de Medición].[Magnitud Física] FROM [Unidades de Medición];
¿Utilizando Historia?	Sí/No	1	

<b>Campo</b>	<b>Tipo</b>	<b>Tamaño</b>	<b>Origen del Registro / Restricciones</b>
Historia Desde	Fecha/Hora	8	
Historia Hasta	Fecha/Hora	8	
Lapso de la Demanda	Texto	50	SELECT DISTINCTROW [Unidades de Medición].[Unidades], [Unidades de Medición].[Magnitud Física] FROM [Unidades de Medición];
% de Exceso o Defecto	Moneda	8	
Costo U Estimado	Moneda	8	
Costo U de Mantener	Moneda	8	
Costo U de Escasez	Moneda	8	
% Anual de Inflación	Moneda	8	
Inventario Inicial	Número (largo)	4	
Costo del Plan	Moneda	8	

Esta tabla registra los datos básicos (o parámetros) necesarios para el conjunto de modelos de programación lineal desarrollados con miras a la predicción de los niveles de inventario en un horizonte de planeación. El campo [N° del Modelo] es la clave primaria de la relación y por tanto es único, esto se traduce en la posibilidad de almacenar un número variable de modelos (del mismo tipo) donde, en cada caso, cambian los parámetros; esta facilidad permite al usuario “jugar” con los modelos de programación hasta conseguir el plan de producción que más convenga. [Código del Producto] especifica el producto sobre el cual se desea analizar el modelo (En esta versión del sistema, sólo es posible modelar un producto a la vez; en futuras versiones, una de las líneas de investigación a desarrollar es el modelado de múltiples productos a la vez). Desde el punto de vista de la base de datos (objeto del presente capítulo), el resto de los datos no ofrecen nuevas complicaciones, sin embargo, se abordará una explicación más detallada en el Capítulo IV.

#### MODELOS EOQ

<b>Campo</b>	<b>Tipo</b>	<b>Tamaño</b>	<b>Origen del Registro / Restricciones</b>
<u>N° de Modelo</u>	Número (largo) Autoincremento	4	
Código del Producto	Texto	10	SELECT DISTINCTROW [Productos Terminados].[Código del Producto], [Productos Terminados].[N° de Fórmula] FROM [Productos Terminados];
Descripción	Memo	-	
Desde	Fecha/Hora	8	
Horizonte	Texto	50	SELECT DISTINCTROW [Unidades de Medición].[Unidades], [Unidades de Medición].[Magnitud Física] FROM [Unidades de Medición];
¿Utilizando Historia?	Sí/No	1	

<b>Campo</b>	<b>Tipo</b>	<b>Tamaño</b>	<b>Origen del Registro / Restricciones</b>
Historia Desde	Fecha/Hora	8	
Historia Hasta	Fecha/Hora	8	
Demanda en el Período	Número (largo)	4	
Lapso de la Demanda	Texto	50	SELECT DISTINCTROW [Unidades de Medición].[Unidades], [Unidades de Medición].[Magnitud Física] FROM [Unidades de Medición];
DS de la Demanda	Moneda	8	
CV de la Demanda	Moneda	8	
Conclusión	Texto	50	
Demora	Moneda	8	
Lapso de la Demora	Texto	50	SELECT DISTINCTROW [Unidades de Medición].[Unidades], [Unidades de Medición].[Magnitud Física] FROM [Unidades de Medición];
Costo U Estimado	Moneda	8	
Costo U de Ordenar	Moneda	8	
Costo U de Mantener	Moneda	8	
¿Hay Escasez?	Sí/No	1	
Costo U de Escasez	Moneda	8	
Tamaño Óptimo del Lote	Número (largo)	4	
Ciclo Óptimo	Moneda	8	
Faltante Óptimo	Número (largo)	4	
Máximo Nivel	Número (largo)	4	
Reorden	Número (largo)	4	
Ciclos Antes	Número (entero)	2	
Costo del Plan	Moneda	8	

Como en la tabla anterior, aquí se registran los parámetros requeridos para el conjunto de modelos del tipo Tamaño Económico del Lote (en inglés Economic Order Quantity -EOQ-). Estos modelos han sido, desde los inicios de la teoría de inventarios, una familia ampliamente utilizada en la predicción del comportamiento de estos sistemas. El campo [N° del Modelo] nuevamente es la clave primaria de la relación y otorga la posibilidad de almacenar un número variable de modelos cambiando los parámetros. Aquí también el usuario puede “jugar” con los modelos hasta conseguir el plan de producción más razonable. [Código del Producto] especifica el producto sobre el cual se desean los pronósticos (en futuras versiones, una de las líneas de investigación a desarrollar es el modelado EOQ de múltiples productos a la vez). Como antes, en el Capítulo IV se abordará una explicación detallada del funcionamiento y significado de cada dato.

**MODELOS RESULTADOS EXCEL**

<b>Campo</b>	<b>Tipo</b>	<b>Tamaño</b>	<b>Origen del Registro / Restricciones</b>
<u>Nº de Modelo</u>	Número (doble)	8	
<u>Período</u>	Número (doble)	8	
A Producir	Número (doble)	8	
A Mantener	Número (doble)	8	
A Demorar	Número (doble)	8	

Esta tabla es en realidad una tabla vinculada del Excel. Una tabla vinculada es aquella que no existe físicamente en la base de datos, existe en otra aplicación (en este caso, Excel, formando parte de un libro de hojas de cálculo). Así, la información contenida aquí se utiliza en la presentación de los resultados obtenidos por el Excel y en la optimización del modelo de programación de la producción. Mayores detalles serán expuestos en el Capítulo IV.

**NOTAS**

<b>Campo</b>	<b>Tipo</b>	<b>Tamaño</b>	<b>Origen del Registro / Restricciones</b>
<u>Nota Nº</u>	Número (largo) Autoincremento	4	
Contenido	Memo	-	

Aquí, el usuario puede escribir las distintas notas o anotaciones que puede eventualmente desear incluir en los documentos que produce el sistema (como cotizaciones o facturas). Estas notas seguramente contendrán advertencias o informaciones adicionales que dar a los clientes. Cada nota recibe un número y se redacta su contenido (a medida que se van necesitando). Las notas se editan a través del formulario [Notas] (ver Anexo 1, Figura 25, pág. 41), al cual se accede desde donde se requiera presionando el botón [Editar Notas].

**NOTAS EN COTIZACIONES**

<b>Campo</b>	<b>Tipo</b>	<b>Tamaño</b>	<b>Origen del Registro / Restricciones</b>
<u>Nº de la Cotización</u>	Número (largo)	4	SELECT DISTINCTROW [Cotizaciones].[Nº de la Cotización], [Cotizaciones].[Código del Cliente], [Cotizaciones].[Remitente] FROM [Cotizaciones];
<u>Nota Nº</u>	Número (largo)	4	SELECT DISTINCTROW [Notas].[Nota Nº] FROM [Notas];

En esta tabla se definen las distintas notas que se desea aparezcan en cada cotización. El [Nº de la Cotización] representa la cotización en cuestión y, para cada uno, [Nota Nº] representa el número de la nota en la tabla [Notas]. Una cotización puede entonces recibir más de una nota. Las notas en cotizaciones se editan a través del subformulario [Notas] que se encuentra en el

formulario [Cotizaciones]. Las notas pueden ser definidas en el mismo formulario presionando el botón [Editar Notas] (ver Anexo 1, Figura 6, pág. 12).

#### NOTAS EN FACTURAS

Campo	Tipo	Tamaño	Origen del Registro / Restricciones
Nº de Factura	Número (largo)	4	SELECT DISTINCTROW [Facturas].[Nº de Factura], [Facturas].[Nº de Pedido], [Facturas].[Fecha de Emisión] FROM [Facturas];
Nota N°	Número (largo)	4	SELECT DISTINCTROW [Notas].[Nota N°] FROM [Notas];

Al igual que antes se definen las distintas notas que se desea aparezcan en la factura. El [Nº de la Factura] representa el recibo en cuestión y, para cada uno, [Nota N°] representa el número de la nota en la tabla [Notas]. La factura también puede recibir más de una nota. Las notas en facturas se editan a través del subformulario [Notas] que se encuentra en el formulario [Facturas]. Las notas pueden ser definidas en el mismo formulario presionando el botón [Editar Notas] (ver Anexo 1, Figuras 15 y 16, pág. 20 y 21).

#### ÓRDENES DE COMPRA

Campo	Tipo	Tamaño	Origen del Registro / Restricciones
Nº	Número (largo) Autoincremento	4	
Proveedor	Texto	10	SELECT DISTINCTROW [Proveedores].[Código del Proveedor], [Proveedores].[Nombre o Razón Social] FROM [Proveedores];
Fecha	Fecha/Hora	8	
Cargos %	Número (simple)	4	

Aquí se registran los datos básicos de cada orden de compra, emitida a los proveedores, para la adquisición de insumos. El campo [Nº] representa el número de la orden de compra que se establece automática y secuencialmente. El campo [Proveedor] es una clave ajena que hace referencia a la tabla [Proveedores] y registra el proveedor al cual se le solicitan los insumos contenidos en la presente orden (por supuesto, la lista de los insumos ordenados se encuentra en la tabla [Insumos en Órdenes], presentada antes). El campo [Fecha] indica la fecha de emisión de la orden, que podrá ser posteriormente cotejada con la fecha de llegada para efectos estadísticos. El campo [Cargos %] permite incluir un número porcentual que se aplicará al valor total de la orden de compra y se añadirá al monto comprometido, generalmente por causas de fletes o

cargos administrativos que aplica el proveedor. Las órdenes de compra se editan a través del formulario [Órdenes de Compra] (ver Anexo 1, Figura 26, pág. 42).

### ÓRDENES DE PRODUCCIÓN

Campo	Tipo	Tamaño	Origen del Registro / Restricciones
Nº de Orden	Número (largo) Autoincremento	4	
Fecha de Emisión	Fecha/Hora	8	
Producto	Texto	10	SELECT DISTINCTROW [Productos Terminados y Fórmulas Consulta].[Código del Producto], [Productos Terminados y Fórmulas Consulta].[Producto], [Productos Terminados y Fórmulas Consulta].[Destino], [Productos Terminados y Fórmulas Consulta].[Vehículo], [Productos Terminados y Fórmulas Consulta].[Presentación], [Productos Terminados y Fórmulas Consulta].[Cantidad en Presentación], [Productos Terminados y Fórmulas Consulta].[Unidades de Presentación], [Productos Terminados y Fórmulas Consulta].[Concentración], [Productos Terminados y Fórmulas Consulta].[Unidades de Concentración], [Productos Terminados y Fórmulas Consulta].[Departamento] FROM [Productos Terminados y Fórmulas Consulta];
¿De Producción?	Sí/No	1	
¿De Acondicionamiento?	Sí/No	1	
Requerido Para	Fecha/Hora	8	
Cantidad	Moneda	8	
Emitida Por	Texto	50	SELECT DISTINCTROW [Personas].[Cédula de Identidad o Clave], [Personas].[Apellidos], [Personas].[Nombres] FROM [Personas];
Autorizada Por	Texto	50	SELECT DISTINCTROW [Personas].[Cédula de Identidad o Clave], [Personas].[Apellidos], [Personas].[Nombres] FROM [Personas];
Despachado Por	Texto	50	SELECT DISTINCTROW [Personas].[Cédula de Identidad o Clave], [Personas].[Apellidos], [Personas].[Nombres] FROM [Personas];
Recibida Por	Texto	50	SELECT DISTINCTROW [Personas].[Cédula de Identidad o Clave], [Personas].[Apellidos], [Personas].[Nombres] FROM [Personas];
Contabilizada Por	Texto	50	SELECT DISTINCTROW [Personas].[Cédula de Identidad o Clave], [Personas].[Apellidos], [Personas].[Nombres] FROM [Personas];
Observaciones	Memo	-	
Lote Nº	Texto	10	
Fecha de Inicio	Fecha/Hora	8	
Fecha de Término	Fecha/Hora	8	
Horas Hombre	Moneda	8	
Rendimiento	Moneda	8	

<b>Campo</b>	<b>Tipo</b>	<b>Tamaño</b>	<b>Origen del Registro / Restricciones</b>
Fecha de Vencimiento	Fecha/Hora	8	
Cantidad de Muestras	Moneda	8	
Horas de Análisis	Moneda	8	
Protocolo de Análisis	Texto	50	
Fecha de Aprobación	Fecha/Hora	8	
Aprobado Por	Texto	50	SELECT DISTINCTROW [Personas].[Cédula de Identidad o Clave], [Personas].[Apellidos], [Personas].[Nombres] FROM [Personas];
Producción Efectiva	Moneda	8	

Esta tabla contiene la información básica de cada orden de producción (el centro del sistema). Cada registro se identifica con un número único decidido por el sistema secuencialmente. El campo [Producto] referencia una combinación de las tablas [Fórmulas] y [Productos Terminados] para hacer más explicativa la lista de productos de entre los cuales el usuario escoge el objetivo de la orden, sin embargo, en el nivel lógico, este campo contiene simplemente un código de producto terminado, el cual se ordena producir mediante el documento. Los campos [¿De Producción?] y [¿De Acondicionamiento?] son campos de tipo lógico que indican si la orden se emite para comenzar uno de los procesos productivos de la empresa y/o si se emite para acondicionar el producto en cuestión (el acondicionamiento, generalmente comprende el envasado, etiquetado y así por el estilo). [Requerido Para] expresa la fecha en la cual se necesitan los insumos solicitados en la orden para comenzar la producción (como el lector ya supondrá, tales insumos se definen en la tabla [Insumos en Órdenes de Producción]). [Cantidad] expresa la cantidad, medida en unidades de presentación del producto -por ejemplo, tabletas o frascos-, que se piensa producir de acuerdo con la orden. Los campos que hacen referencia a la tabla [Personas] identifican a los empleados de la planta que intervienen en la elaboración y producción de la orden. El campo [Lote N°] es de suma importancia para el sistema; este dato debe ser seleccionado por el usuario una vez concluida la corrida de producción que implica la orden; la lógica indica que debe ser decidido de común acuerdo entre Producción y Control de Calidad y se espera que se convierta en una identificación adicional, en lo sucesivo, para cada producto. Las fechas ([Fecha de Inicio] y [Fecha de Término]) registran los momentos en los que se comienza la corrida de producción y aquellos en que se termina. Afectan directamente algunos reportes del sistema, sobre todo relacionados con la existencia a la fecha, de los productos. Los datos subsiguientes de la tabla son anexados, al término de la producción, por Control de Calidad o la persona responsable de la producción, fundamentalmente miden cualitativamente los



resultados del proceso. El campo [Producción Efectiva] es vital para el sistema, ya que registra, al término de cada corrida de producción relacionada con la orden, el número de unidades (de presentación) realmente producidas (que puede diferir considerablemente de lo planificado) en el proceso; este dato es en realidad el que se considera para efectos del cómputo de las existencias. Las órdenes de producción se editan a través del formulario [Órdenes de Producción] (ver Anexo 1, Figuras 27 y 28, pág. 45 y 46).

#### PAÍSES

Campo	Tipo	Tamaño	Origen del Registro / Restricciones
País	Texto	50	
Código Telefónico	Texto	10	

Aquí se definen los países que se utilizarán luego conjuntamente con los estados y las ciudades para efectos del registro de direcciones y números telefónicos. Se supone que cada país tiene un sólo código telefónico. Los países se editan a través del formulario [Países] (ver Anexo 1, Figura 29, pág. 50). Todos aquellos formularios que tengan elementos relacionados con países, estados y ciudades, contienen un botón [Editar Países, Estados y Ciudades] con el que se llega a este formulario.

#### PEDIDOS DE CLIENTES

Campo	Tipo	Tamaño	Origen del Registro / Restricciones
Nº de Pedido	Número (largo) Autoincremento	4	
Código del Cliente	Texto	50	SELECT DISTINCTROW [Clientes].[Código del Cliente], [Clientes].[Nombre o Razón Social], [Clientes].[CI o RIF] FROM [Clientes];
Fecha	Fecha/Hora	8	
Vendedor	Texto	50	SELECT DISTINCTROW [Personas].[Cédula de Identidad o Clave], [Personas].[Apellidos], [Personas].[Nombres] FROM [Personas];
Condición	Texto	20	SELECT DISTINCTROW [Condiciones].[Condición] FROM [Condiciones];
Tipo de Orden	Texto	30	SELECT DISTINCTROW [Tipos de Órdenes de Pedido].[Tipo de Orden] FROM [Tipos de Órdenes de Pedido];
Nº de Orden	Texto	20	
Observaciones	Memo	-	

Esta tabla registra la información básica de cada pedido recibido del cliente. Quizá una de las restricciones más importantes impuestas por SARCAI a la organización tiene que ver con los

pedidos. En efecto, en la actualidad, la organización no tiene un control estricto de los pedidos que le son ordenados y en consecuencia no domina la información relativa a la demanda de sus productos (que no debe confundirse con las ventas). En el sistema, toda venta debe estar precedida por un pedido; por otro lado, no todos los pedidos logran convertirse en ventas (bien sea porque el cliente se arrepiente de comprar o, lo que es mucho más frecuente, porque la planta carece de existencias para la venta). Como seguramente se imaginará el lector, para efectos de pronosticar y planificar el comportamiento futuro del sistema, lo verdaderamente importante es el pedido (no la factura) si se piensa que uno de los objetivos primarios es justamente disminuir el número de veces en que ocurre escasez del producto, aumentando o planificando mejor en el tiempo, la producción. Así pues, la información de los pedidos es fundamental para el estudio de la demanda de cada ítem. Aquí, cada pedido recibe un número único secuencial que lo identifica, registrado en el campo [N° de Pedido]. Se registra referencialmente el cliente, también el vendedor (que para todos los efectos es quien logra “arrancar” al cliente el pedido), la condición, por ejemplo, si se espera que la factura sea cancelada a crédito o a contado, el tipo de orden (alguna forma de clasificación a ser definida por el usuario) y el número que la identifica por parte del cliente. El trabajo con los pedidos se comparte entre esta tabla y la tabla [Insumos Pedidos de Clientes] que contiene la lista detallada de lo solicitado. Los pedidos de clientes se editan a través del formulario [Pedidos de Clientes] (ver Anexo 1, Figuras 30 y 31, pág. 52 y 53).

## PERSONAS

Campo	Tipo	Tamaño	Origen del Registro / Restricciones
Cédula de Identidad o Clave	Texto	20	
Apellidos	Texto	25	
Nombres	Texto	25	
Fecha de Nacimiento	Fecha/Hora	8	
País de Nacimiento	Texto	50	SELECT DISTINCTROW [Países].[País] FROM [Países];
Sexo	Sí/No	1	
Estado Civil	Texto	15	SELECT DISTINCTROW [Estados Civiles].[Estado Civil] FROM [Estados Civiles];
N° de Hijos	Número (entero)	2	>=0
Dirección de Habitación	Texto	100	
Dirección de Trabajo	Texto	100	
País	Texto	50	SELECT DISTINCTROW [Países].[País] FROM [Países];
Estado	Texto	50	SELECT DISTINCTROW [Estados de Países].[Estado], [Estados de Países].[País]

Campo	Tipo	Tamaño	Origen del Registro / Restricciones
			FROM [Estados de Países];
Ciudad	Texto	50	SELECT DISTINCTROW [Ciudades de Estados].[Ciudad], [Ciudades de Estados].[Estado], [Ciudades de Estados].[País] FROM [Ciudades de Estados];
Cargo	Texto	60	SELECT DISTINCTROW [Cargos].[Cargo] FROM [Cargos];
Empleado de PROULA?	Sí/No	1	

En esta tabla se definen todas las personas que intervienen en las distintas secciones del sistema, ya sean empleados de La Planta o no. Los datos de esta tabla han de ser familiares para el lector, por lo que no deberían requerir mayores explicaciones. Nótese sin embargo que para los efectos del inventario se podría decir que lo único pertinente en este momento es la identificación de la persona y si es no empleado de la organización, no obstante, se han incluido algunos otros datos con la finalidad de sugerir una de las vías de crecimiento del sistema: el subsistema de personal. Las personas se editan a través del formulario [Personas] (ver Anexo 1, Figura 38, pág. 76) al cual se accede, donde sea requerido, presionando el botón [Editar Personas].

#### PLANES DE PRODUCCIÓN

Campo	Tipo	Tamaño	Origen del Registro / Restricciones
Nº del Plan	Número (largo) Autoincremento	4	
Descripción	Memo	-	
Fecha de Inicio	Fecha/Hora	8	
Fecha de Culminación	Fecha/Hora	8	

Esta tabla almacena los datos básicos de los distintos planes de producción que es posible formular. Trabaja conjuntamente con la tabla [Productos en Planes] (donde se definen específicamente los productos que se espera producir en el plan). Pueden, dado que existe un identificador único de cada plan ([Nº del Plan]), definirse un sinnúmero de planes para la producción de uno o varios productos. El campo [Descripción] es capaz de recibir una cadena larga de caracteres donde se explique en qué consiste el plan. Los campos [Fecha de Inicio] y [Fecha de Culminación] admiten dos fechas que indican para cuándo se planifica el comienzo y la culminación del plan. Esta tabla puede ser actualizada por el usuario, sin embargo, también es actualizada automáticamente cuando se decide la implementación de alguno de los modelos de Programación o EOQ. En estos casos, el sistema automáticamente elabora el plan de producción calculando fechas, cantidades y la descripción, que incluye la propia descripción del modelo

empleado. Los planes de producción se editan a través del formulario [Planes de Producción] (ver Anexo 1, Figura 39, pág. 79), al cual también se accede desde los modelos.

#### PRECIOS DE PRODUCTOS

Campo	Tipo	Tamaño	Origen del Registro / Restricciones
<u>Código del Producto</u>	Texto	10	SELECT DISTINCTROW [Productos Terminados].[Código del Producto], [Productos Terminados].[Nº de Fórmula], [Productos Terminados].[Presentación] FROM [Productos Terminados];
<u>Fecha</u>	Fecha/Hora	8	
Precio Unitario de Venta	Moneda	8	>0

Aquí se almacenan los distintos precios que van adjudicándoseles a los productos terminados a lo largo del tiempo. [Código del Producto] referencia la tabla [Productos Terminados] de donde se selecciona el producto que recibirá un nuevo precio de venta. [Fecha] es el campo que registra la fecha en que se está decidiendo el nuevo precio (se supone que un mismo producto no puede recibir dos precios distintos en una misma fecha). [Precio Unitario de Venta] es un número que representa el precio del producto. En vista de la posibilidad de “recordar” los distintos precios de los productos a todo lo largo de su historia, se cuenta, gracias a esta tabla, con la serie de tiempo de los precios de cada producto (y posteriormente será posible estudiar su evolución). Los precios de productos se editan a través del subformulario [Precios] que se encuentra en el formulario [Productos Terminados] (ver Anexo 1, Figura 42, pág. 82).

#### PRESENTACIONES

Campo	Tipo	Tamaño	Origen del Registro / Restricciones
<u>Presentación</u>	Texto	40	

Esta tabla permite definir las distintas presentaciones de los productos terminados, por ejemplo, frascos de vidrio, frascos plásticos, estuches, etc. No debe confundirse con las unidades de presentación de los insumos. Las presentaciones se editan a través del formulario [Presentaciones] (ver Anexo 1, Figura 40, pág. 81), al cual se accede desde el formulario [Productos Terminados] (ver Anexo 1, Figura 42, pág. 82) presionando el botón [Editar Presentaciones].

**PRODUCTOS COTIZADOS A CLIENTES**

<b>Campo</b>	<b>Tipo</b>	<b>Tamaño</b>	<b>Origen del Registro / Restricciones</b>
<u>Nº de la Cotización</u>	Número (largo)	4	SELECT DISTINCTROW [Cotizaciones].[Nº de la Cotización], [Cotizaciones].[Código del Cliente], [Cotizaciones].[Fecha de Emisión] FROM [Cotizaciones];
<u>Código del Producto</u>	Texto	50	SELECT DISTINCTROW [Productos Terminados].[Código del Producto], [Productos Terminados].[Nº de Fórmula], [Productos Terminados].[Presentación] FROM [Productos Terminados];
Cantidad Cotizada	Número (simple)	4	>0

Esta tabla trabaja mancomunadamente con la tabla [Cotizaciones]. Aquí se definen en detalle los distintos productos que le son cotizados al cliente. Así, el campo [Nº de la Cotización] referencia a la tabla [Cotizaciones] y toma de ella el número correspondiente, el campo [Código del Producto] referencia a [Productos Terminados] e indica el producto que se incluye en la cotización y [Cantidad Cotizada] representa la cantidad del producto que se ofrece. Nótese que el precio del producto cotizado no está presente en esta tabla; la razón es que, para conservar la integridad de los datos, el usuario que realiza la cotización no puede escoger el precio que coloca al producto, el mismo se toma automáticamente de la base de datos de acuerdo con la fecha en que se elabora la cotización y cotejando la lista de precios vigente para ese momento. Los productos cotizados a clientes se editan a través del subformulario [Productos Cotizados] que se encuentra en el formulario [Cotizaciones]. Los productos pueden definirse en el mismo formulario presionando el botón [Editar Productos] (ver Anexo 1, Figura 6, pág. 12).

**PRODUCTOS DEMANDADOS**

<b>Campo</b>	<b>Tipo</b>	<b>Tamaño</b>	<b>Origen del Registro / Restricciones</b>
<u>Producto</u>	Texto	10	SELECT [Lista Sencilla de Productos Terminados].[Código del Producto], [Lista Sencilla de Productos Terminados].Nombre FROM [Lista Sencilla de Productos Terminados];

Esta tabla no pertenece al conjunto básico, se utiliza con la finalidad de que el usuario exponga los productos para los cuales desea listar algunos informes y gráficos relacionados con la demanda. El campo [Producto] referencia a la tabla [Productos Terminados] (aunque la instrucción SQL extrae los datos de un resumen de esta tabla denominado [Lista Sencilla de

Productos Terminados]). Esta tabla se llena con información introducida en el formulario [Pedir Fechas para la Demanda] (ver Anexo 1, Figura 34, pág. 64).

#### PRODUCTOS EN MODELOS P

Campo	Tipo	Tamaño	Origen del Registro / Restricciones
<u>Nº de Modelo</u>	Número (largo)	4	SELECT DISTINCTROW [Modelos de Programación].[Nº de Modelo], [Modelos de Programación].[Código del Producto] FROM [Modelos de Programación];
<u>Período</u>	Número (entero)	2	
Demanda	Número (largo)	4	
Costo de Producción	Moneda	8	
Costo de Mantener	Moneda	8	
Costo de Escasez	Moneda	8	
A Producir	Número (largo)	4	
A Mantener	Número (largo)	4	
A Demorar	Número (largo)	4	

Aquí se guardan los datos en detalle que serán utilizados en la resolución de cada modelo de programación de la producción (puede verse también la tabla [Modelos de Programación]). Adicionalmente, una vez resuelto el modelo, las cantidades óptimas a producir, a almacenar y a demorar, se devuelven del Excel y se incorporan automáticamente a la tabla. Los costos que se representan se toman de [Modelos de Programación] y se les aplica tanto el porcentaje de exceso o defecto como el índice inflacionario estimados por el usuario, de período en período. Vale decir que el usuario puede no obstante modificar estos datos y ajustarlos de acuerdo con su experiencia, para la elaboración del plan de producción (esto es, los datos de esta tabla se calculan automáticamente, pero son editables). Mayores detalles pueden verse en el Capítulo IV.

#### PRODUCTOS EN PLANES

Campo	Tipo	Tamaño	Origen del Registro / Restricciones
<u>Nº del Plan</u>	Número (largo)	4	SELECT DISTINCTROW [Planes de Producción].[Nº del Plan], [Planes de Producción].[Fecha de Inicio], [Planes de Producción].[Fecha de Culminación] FROM [Planes de Producción];
<u>Código del Producto</u>	Texto	10	SELECT DISTINCTROW [Productos Terminados].[Código del Producto], [Productos Terminados].[Nº de Fórmula], [Productos Terminados].[Cantidad en Presentación], [Productos Terminados].[Destino] FROM [Productos Terminados];
Fecha	Fecha/Hora	8	
Cantidad	Moneda	8	>=0

Esta tabla opera con la tabla [Planes de Producción]. Aquí se definen los distintos productos, la cantidad de cada uno y la fecha en que se producirán, para cada plan de producción que haya sido previamente almacenado por el usuario. Los productos en planes se editan a través del subformulario [Plan de Producción] que se encuentra en el formulario [Planes de Producción]. Los productos pueden definirse en el mismo formulario presionando el botón [Editar Productos Terminados] (ver Anexo 1, Figura 39, pág. 79).

#### PRODUCTOS FACTURADOS A CLIENTES

Campo	Tipo	Tamaño	Origen del Registro / Restricciones
<u>Nº de Factura</u>	Número (largo)	4	SELECT DISTINCTROW [Facturas].[Nº de Factura], [Facturas].[Nº de Pedido], [Facturas].[Fecha de Emisión] FROM [Facturas];
<u>Código del Producto</u>	Texto	50	SELECT DISTINCTROW [Productos Terminados].[Código del Producto], [Productos Terminados].[Nº de Fórmula], [Productos Terminados].[Presentación] FROM [Productos Terminados];
<u>Nº de Orden</u>	Número (largo)	4	SELECT DISTINCTROW [Órdenes de Producción].[Nº de Orden], [Órdenes de Producción].[Fecha de Emisión], [Órdenes de Producción].[Producto] FROM [Órdenes de Producción];
Cantidad Facturada	Número (simple)	4	>0

Aquí se almacenan los datos detallados sobre los productos que se incluyen en la factura (definida en la tabla [Facturas]). Los campos [Nº de Factura], [Código del Producto] y [Nº de Orden] representan la clave primaria de la relación. Este último referencia la tabla [Órdenes de Producción], lo que implica para cada factura la posibilidad de incluir un mismo producto varias veces proveniente de distintas órdenes de producción. La razón de esta decisión de diseño tiene que ver con la necesidad de controlar, en cada producto vendido, el origen exacto e incluso el origen de los insumos utilizados para su elaboración. Por lo demás, los campos se explican por sí mismos. Tal vez lo que merezca algún comentario adicional es que los precios (como se explicó para el caso de las cotizaciones) se toman automáticamente de acuerdo con la fecha de la factura y vigencia del precio de los productos. Los productos facturados a clientes se editan a través del subformulario [Productos Facturados] que se encuentra en el formulario [Facturas]. Los productos son automáticamente tomados una vez que el usuario selecciona el pedido asociado con la factura, por lo que no es necesario editarlos en este formulario (ver Anexo 1, Figuras 15 y 16, pág. 20 y 21).

**PRODUCTOS GENÉRICOS**

<b>Campo</b>	<b>Tipo</b>	<b>Tamaño</b>	<b>Origen del Registro / Restricciones</b>
<u>Producto</u>	Texto	30	

Aquí se almacena la información de los productos genéricos, utilizados principalmente para la elaboración de la fórmula del producto. Los productos genéricos se editan a través del formulario [Productos Genéricos] (ver Anexo 1, Figura 41, pág. 81), al cual se accede desde el formulario [Fórmulas] (ver Anexo 1, Figuras 18 y 19, pág. 25 y 26) presionando el botón [Editar Productos Genéricos].

**PRODUCTOS PEDIDOS DE CLIENTES**

<b>Campo</b>	<b>Tipo</b>	<b>Tamaño</b>	<b>Origen del Registro / Restricciones</b>
<u>Nº de Pedido</u>	Número (largo)	4	SELECT DISTINCTROW [Pedidos de Clientes].[Nº de Pedido], [Pedidos de Clientes].[Código del Cliente], [Pedidos de Clientes].[Fecha], [Pedidos de Clientes].[Vendedor] FROM [Pedidos de Clientes];
<u>Código del Producto</u>	Texto	50	SELECT DISTINCTROW [Productos Terminados].[Código del Producto], [Productos Terminados].[Nº de Fórmula], [Productos Terminados].[Presentación] FROM [Productos Terminados];
Cantidad Pedida	Número (simple)	4	>0

Esta tabla funciona de forma equivalente, aplicada al caso de los pedidos, que sus similares en cotizaciones y facturas, por lo que sobran más explicaciones. Los productos pedidos de clientes se editan a través del subformulario [Productos Solicitados] que se encuentra en el formulario [Pedidos de Clientes]. Los productos pueden definirse en el mismo formulario presionando el botón [Editar Productos Terminados] (ver Anexo 1, Figuras 30 y 31, pág. 52 y 53).

**PRODUCTOS TERMINADOS**

<b>Campo</b>	<b>Tipo</b>	<b>Tamaño</b>	<b>Origen del Registro / Restricciones</b>
<u>Código del Producto</u>	Texto	10	
<u>Nº de Fórmula</u>	Texto	10	SELECT DISTINCTROW [Fórmulas].[Nº de Fórmula], [Fórmulas].[Producto], [Fórmulas].[Vehículo], [Fórmulas].[Concentración], [Fórmulas].[Unidades de Concentración] FROM [Fórmulas];
<u>Presentación</u>	Texto	40	SELECT DISTINCTROW [Presentaciones].[Presentación] FROM [Presentaciones];
Cantidad en Presentación	Moneda	8	>0
<u>Destino</u>	Texto	50	SELECT DISTINCTROW [Destinos].[Destino] FROM [Destinos];
<u>Punto de Reorden</u>	Moneda	8	



<b>Campo</b>	<b>Tipo</b>	<b>Tamaño</b>	<b>Origen del Registro / Restricciones</b>
Costo Unitario	Moneda	8	
Fotografía	Objeto OLE	-	

En esta tabla se definen los productos terminados. Cada uno recibe un código de producto que lo identifica en el sistema. La relación más importante de esta tabla es la que hace referencia a la fórmula. La idea es que varios productos diferentes (por ejemplo, en cuanto al destino o la presentación final), pueden compartir la misma fórmula. Los campos [Presentación] y [Cantidad en Presentación] operan conjuntamente para especificar la presentación final del producto. La cantidad se refiere al número de unidades de presentación básicas (expresadas en la fórmula) que se aglutinan en la presentación final (por ejemplo, tabletas en un frasco) para conformar el producto que será vendido. [Destino] permite almacenar un registro enlazado de la tabla [Destinos] que indica el tipo de transacción para la que está pensado el producto, por ejemplo, Venta, Muestra Médica u otros. El campo [Punto de Reorden] es el nivel del inventario del producto por debajo del cual la organización debe reaprovisionarse (es urgente la producción de un nuevo lote). Este valor puede ser introducido de acuerdo con la experiencia del usuario, o bien puede ser automáticamente fijado por el sistema, cuando se ejecuta un modelo del tipo EOQ. La intención es que al implementarse este sistema será mucho más fácil realizar políticas de revisión continua del inventarios, en las cuales, con una simple alarma disparada automáticamente, se ordene la producción de aquel producto cuya existencia esté por debajo del punto de reorden. El campo [Costo Unitario] debe ser introducido por el usuario de acuerdo con los cómputos que tenga la administración. Sirve para realizar estudios de rentabilidad y se espera, en futuras versiones, que sea posible determinarlo automáticamente a partir del conocimiento profundo de los distintos procesos productivos y sus costos asociados (incorporando un subsistema de costos). El campo [Fotografía] puede recibir un objeto gráfico cualquiera que muestre una fotografía digital del producto. La utilidad de este campo es que se cuenta entonces con un catálogo (posiblemente a emplearse para efectos publicitarios), con arte final (o casi) que está siempre actualizado con los cambios a la base. Los productos terminados se editan a través del formulario [Productos Terminados] (ver Anexo 1, Figura 42, pág. 82).

#### PRODUCTOS TERMINADOS Y ESCALAS DE PRECIOS

<b>Campo</b>	<b>Tipo</b>	<b>Tamaño</b>	<b>Origen del Registro / Restricciones</b>
<u>Código del Producto</u>	Texto	10	SELECT DISTINCTROW [Productos

Campo	Tipo	Tamaño	Origen del Registro / Restricciones
			Terminados].[Código del Producto], [Productos Terminados].[Nº de Fórmula] FROM [Productos Terminados];
<u>Escala</u>	Número (largo)	4	SELECT DISTINCTROW [Escala] de Precios].[Escala], [Escala] de Precios].[Descripción], [Escala] de Precios].[Límite Inferior], [Escala] de Precios].[Límite Superior], [Escala] de Precios].[% Descuento] FROM [Escala] de Precios];

La forma implementada en SARCAI para decidir descuentos en la facturación a clientes se basa en esta tabla. Aquí se asocian los productos con las escalas de precios (que como se explicó definen porcentajes de descuento de acuerdo con la cantidad comprada). Se optó por esta idea, pues es la más sistemática y equitativa como política de descuentos. En futuras versiones del software podrán incorporarse nuevos métodos. Esta decisión es importante, pues la conformación de los precios afecta directamente la elaboración de los informes relativos a las existencias en el inventario. El funcionamiento de la tabla es similar a las anteriormente expuestas de su tipo, por lo que se omiten mayores explicaciones. Los productos terminados y escalas de precios se editan a través del subformulario [Escala de Descuentos por Cantidad] que se encuentra en el formulario [Productos Terminados]. Las escalas pueden definirse en el mismo formulario presionando el botón [Editar Escalas de Precios] (ver Anexo 1, Figura 42, pág. 82).

#### PROVEEDORES

Campo	Tipo	Tamaño	Origen del Registro / Restricciones
<u>Código del Proveedor</u>	Texto	10	
Nombre o Razón Social	Texto	80	
CI o RIF	Texto	20	
Dirección	Texto	100	
País	Texto	50	SELECT DISTINCTROW [Países].[País] FROM [Países];
Estado	Texto	50	SELECT DISTINCTROW [Estados de Países].[Estado], [Estados de Países].[País] FROM [Estados de Países];
Ciudad	Texto	50	SELECT DISTINCTROW [Ciudades de Estados].[Ciudad], [Ciudades de Estados].[Estado], [Ciudades de Estados].[País] FROM [Ciudades de Estados];
Tenencia Del Local	Texto	15	SELECT DISTINCTROW [Tenencia de Locales].[Tenencia] FROM [Tenencia de Locales];
Figura Jurídica	Texto	50	SELECT DISTINCTROW [Figuras Jurídicas].[Figura Jurídica] FROM [Figuras Jurídicas];
Nº del RC	Texto	50	
Fecha del RC	Fecha/Hora	8	
Duración del RC	Texto	50	

<b>Campo</b>	<b>Tipo</b>	<b>Tamaño</b>	<b>Origen del Registro / Restricciones</b>
Tipo de Proveedor	Texto	50	SELECT DISTINCTROW [Tipos de Proveedores].[Tipo de Proveedor] FROM [Tipos de Proveedores];
Responsable	Texto	20	SELECT DISTINCTROW [Personas].[Cédula de Identidad o Clave], [Personas].[Apellidos], [Personas].[Nombres] FROM [Personas];
Fecha de Revisión	Fecha/Hora	8	

Esta tabla opera de manera muy similar a la tabla [Clientes], por lo que la mayoría de los datos pueden ser entendidos repasando lo escrito para aquel caso. Tal vez lo que se deba aclarar es que se trata de proveedores, en su mayor parte empresas, cuyo objetivo en el sistema es el surtido de insumos para la producción y no la recepción de la labor de ventas de la empresa. Los proveedores se editan a través del formulario [Proveedores] (ver Anexo 1, Figuras 43 y 44, pág. 86 y 87).

#### RAZONES DE SALIDAS

<b>Campo</b>	<b>Tipo</b>	<b>Tamaño</b>	<b>Origen del Registro / Restricciones</b>
<u>Razón</u>	Texto	100	

En esta tabla se listan las distintas razones que pueden aducirse en cada salida que ocurra de insumos. Se opera aquí conjuntamente con la tabla [Salidas del Almacén de Insumos]. Estas salidas no son las convencionales (que ocurren cuando se consume el insumo en la producción), sino aquellas que se producen por otras causas, por ejemplo, obsolescencia, robo o deficiencias cualitativas. Para efectos estadísticos, esta tabla será importante en futuros análisis de calidad. Las razones de salidas se editan a través del formulario [Razones de Salidas] (ver Anexo 1, Figura 45, pág. 90), al cual se accede desde el formulario [Llegadas de Insumos al Almacén] (ver Anexo 1, Figuras 22 y 23, pág. 36 y 37) presionando el botón [Editar Razones de Salidas].

#### REFERENCIAS BANCARIAS DE CLIENTES

<b>Campo</b>	<b>Tipo</b>	<b>Tamaño</b>	<b>Origen del Registro / Restricciones</b>
<u>Cliente</u>	Texto	10	SELECT DISTINCTROW [Clientes].[Código del Cliente], [Clientes].[Nombre o Razón Social] FROM [Clientes];
<u>Nombre del Banco</u>	Texto	60	
<u>Nº de Cuenta</u>	Texto	50	
<u>Dirección y Teléfonos</u>	Memo	-	
<u>Montos de Créditos</u>	Moneda	8	>=0

Aquí se registra la información de cuentas bancarias de los clientes, parte de los datos que completa el expediente de cada uno. Los nombres de los campos se corresponden fielmente con

su significado, por lo que se omiten más explicaciones. Las referencias bancarias de clientes se editan a través del subformulario [Referencias Bancarias] que se encuentra en el formulario [Clientes] (ver Anexo 1, Figuras 2 y 3, pág. 5 y 6).

#### REFERENCIAS BANCARIAS DE PERSONAS

<b>Campo</b>	<b>Tipo</b>	<b>Tamaño</b>	<b>Origen del Registro / Restricciones</b>
<u>Persona</u>	Texto	20	SELECT DISTINCTROW [Personas].[Cédula de Identidad o Clave], [Personas].[Apellidos], [Personas].[Nombres] FROM [Personas];
<u>Nombre del Banco</u>	Texto	60	
<u>Nº de Cuenta</u>	Texto	50	
<u>Dirección y Teléfonos</u>	Memo	-	
<u>Montos de Créditos</u>	Moneda	8	>=0

La descripción de esta tabla es similar a la anterior, aplicada aquí no a los clientes, sino a las personas. Las referencias bancarias de personas se editan a través del subformulario [Referencias Bancarias] que se encuentra en el formulario [Personas] (ver Anexo 1, Figura 38, pág. 76).

#### REFERENCIAS COMERCIALES DE CLIENTES

<b>Campo</b>	<b>Tipo</b>	<b>Tamaño</b>	<b>Origen del Registro / Restricciones</b>
<u>Cliente</u>	Texto	10	SELECT DISTINCTROW [Clientes].[Código del Cliente], [Clientes].[Nombre o Razón Social] FROM [Clientes];
<u>Nombre del Comercio</u>	Texto	60	
<u>Dirección y Teléfonos</u>	Memo	-	
<u>Montos de Créditos</u>	Moneda	8	>=0

Aquí se registra la información de créditos comerciales que son incorporados como referencia a los clientes. Su significado es similar al de las referencias bancarias, por lo que sobran más detalles. Las referencias comerciales de clientes se editan a través del subformulario [Referencias Comerciales] que se encuentra en el formulario [Clientes] (ver Anexo 1, Figuras 2 y 3, pág. 5 y 6).

#### REFERENCIAS COMERCIALES DE PERSONAS

<b>Campo</b>	<b>Tipo</b>	<b>Tamaño</b>	<b>Origen del Registro / Restricciones</b>
<u>Persona</u>	Texto	10	SELECT DISTINCTROW [Personas].[Cédula de Identidad o Clave], [Personas].[Apellidos], [Personas].[Nombres] FROM [Personas];
<u>Nombre del Comercio</u>	Texto	60	
<u>Dirección y Teléfonos</u>	Memo	-	
<u>Montos de Créditos</u>	Moneda	8	>=0

Aquí se trata la misma idea que antes, pero aplicada a las personas. Las referencias comerciales de personas se editan a través del subformulario [Referencias Comerciales] que se encuentra en el formulario [Personas] (ver Anexo 1, Figura 38, pág. 76).

#### REVISIONES DE FÓRMULAS

Campo	Tipo	Tamaño	Origen del Registro / Restricciones
<u>Nº de Fórmula</u>	Texto	10	SELECT DISTINCTROW [Fórmulas].[Nº de Fórmula], [Fórmulas].[Producto], [Fórmulas].[Vehículo], [Fórmulas].[Concentración], [Fórmulas].[Unidades de Concentración] FROM [Fórmulas];
<u>Fecha de Revisión</u>	Fecha/Hora	8	
Revisado Por	Texto	20	SELECT DISTINCTROW [Personas].[Cédula de Identidad o Clave], [Personas].[Apellidos], [Personas].[Nombres] FROM [Personas];
Observaciones	Memo	-	

En esta tabla se almacenan los datos registrados cuando ocurre una revisión de cada fórmula (actividad que por definición corresponde realizar al farmaceuta patrocinante del medicamento, cada cierto tiempo). De esta manera se registra la fórmula objeto de revisión, la fecha cuando se da tal revisión, quién la revisa y las observaciones que se le hacen. Si después de implementada una fórmula, el farmaceuta decide alterar de algún modo la versión original, el procedimiento debería ser efectuar la alteración modificando los datos en [Fórmulas] o en [Insumos en Fórmulas] y luego describir la modificación en esta tabla, indicando fecha y razones de la alteración. De esta manera se cuenta con un historial de cómo cambia la fórmula a través del tiempo. Las revisiones de fórmulas se editan a través del subformulario [Revisiones de Fórmulas] que se encuentra en el formulario [Fórmulas]. Las personas que la revisan pueden definirse en el mismo formulario presionando el botón [Editar Personas] (ver Anexo 1, Figuras 18 y 19, pág. 25 y 26).

#### SALIDAS DEL ALMACÉN DE INSUMOS

Campo	Tipo	Tamaño	Origen del Registro / Restricciones
<u>Lote Nº</u>	Número (largo)	4	SELECT DISTINCTROW [Llegadas de Insumos al Almacén].[Lote Nº], [Llegadas de Insumos al Almacén].[Fecha], [Llegadas de Insumos al Almacén].[Orden Nº], [Llegadas de Insumos al Almacén].[Insumo], [Llegadas de Insumos al Almacén].[Cantidad] FROM [Llegadas de Insumos al Almacén];
<u>Fecha</u>	Fecha/Hora	8	

<b>Campo</b>	<b>Tipo</b>	<b>Tamaño</b>	<b>Origen del Registro / Restricciones</b>
<u>Tipo de Salida</u>	Texto	50	SELECT DISTINCTROW [Tipos de Salidas].[Tipo de Salida] FROM [Tipos de Salidas];
<u>Razón</u>	Texto	50	SELECT DISTINCTROW [Razones de Salidas].[Razón] FROM [Razones de Salidas];
Cantidad	Moneda	8	>=0
Autoriza por Producción	Texto	50	SELECT DISTINCTROW [Personas].[Cédula de Identidad o Clave], [Personas].[Apellidos], [Personas].[Nombres] FROM [Personas];
Autoriza por Control	Texto	50	SELECT DISTINCTROW [Personas].[Cédula de Identidad o Clave], [Personas].[Apellidos], [Personas].[Nombres] FROM [Personas];
Ejecuta	Texto	50	SELECT DISTINCTROW [Personas].[Cédula de Identidad o Clave], [Personas].[Apellidos], [Personas].[Nombres] FROM [Personas];
Observaciones	Memo	-	

Aquí se registran los datos básicos de cada salida que, por razones distintas a la producción, ocurra en el almacén de insumos. El campo [Lote N°] se refiere al lote del cual sale el insumo en cuestión (hay que recordar que para un insumo, existen varios lotes -a medida que van llegando las cantidades compradas-). [Fecha] registra el momento en que ocurre la salida (o se detecta). [Tipo de Salida] registra la razón macro de la salida, por ejemplo, robo u obsolescencia. [Razón] registra con mayor detalle la causa de la salida. [Cantidad] es un número medido en unidades de presentación del insumo, que indica las unidades que salen. [Autoriza por Producción], [Autoriza por Control] y [Ejecuta] son campos que esperan recibir claves de personas que realizan el control de las salidas, autorizándolas o ejecutándolas (cuando se trate de destrucción por vencimiento o descomposición). Las salidas del almacén de insumos se editan a través del subformulario [Salidas del Almacén de Insumos] que se encuentra en el formulario [Llegadas de Insumos al Almacén]. Las razones, tipos de salidas y personas pueden definirse en el mismo formulario presionando respectivamente los botones [Editar Razones de Salidas], [Editar Tipos de Salidas] y [Editar Personas] (ver Anexo 1, Figuras 22 y 23, pág. 36 y 37).

#### SECTORES ECONÓMICOS

<b>Campo</b>	<b>Tipo</b>	<b>Tamaño</b>	<b>Origen del Registro / Restricciones</b>
<u>Sector</u>	Texto	50	

En esta tabla se definen los distintos sectores de la economía a los que pueden pertenecer los clientes, por ejemplo, Sector Público, Sector Privado, Mixto u otros. Los sectores económicos se editan a través del formulario [Sectores Económicos] (ver Anexo 1, Figura 46, pág. 90), al cual

se accede desde el formulario [Clientes] (ver Anexo 1, Figuras 2 y 3, pág. 5 y 6) presionando el botón [Editar Sectores].

#### SUCURSALES DE CLIENTES

Campo	Tipo	Tamaño	Origen del Registro / Restricciones
<u>Cliente</u>	Texto	10	SELECT DISTINCTROW [Lista Sencilla de Clientes].[Código del Cliente], [Lista Sencilla de Clientes].[Nombre o Razón Social] FROM [Lista Sencilla de Clientes];
<u>Sucursal</u>	Texto	10	SELECT DISTINCTROW [Lista Sencilla de Clientes].[Código del Cliente], [Lista Sencilla de Clientes].[Nombre o Razón Social] FROM [Lista Sencilla de Clientes];

Esta es una relación de tipo especial. En la jerga de bases de datos, a este tipo de relaciones se les conoce como “autorreferenciales”. El sentido de esta tabla es relacionar los clientes consigo mismos. El campo [Cliente] hace referencia a la tabla [Clientes] y el campo [Sucursal] también. En consecuencia, el supuesto sobre el que se construye esta relación es que una sucursal de cualquier cliente es también un cliente y, por lo tanto, ha de ser definida primero como tal. Esta tabla permite crear tantas sucursales como sea necesario y permite a la larga estudiar el comportamiento de clientes del tipo consorcio o fusionados. Las sucursales de clientes se editan a través del subformulario [Sucursales] que se encuentra en el formulario [Clientes] (ver Anexo 1, Figuras 2 y 3, pág. 5 y 6).

#### SUCURSALES DE PROVEEDORES

Campo	Tipo	Tamaño	Origen del Registro / Restricciones
<u>Proveedor</u>	Texto	10	SELECT DISTINCTROW [Lista Sencilla de Proveedores].[Código del Proveedor], [Lista Sencilla de Proveedores].[Nombre o Razón Social] FROM [Lista Sencilla de Proveedores];
<u>Sucursal</u>	Texto	10	SELECT DISTINCTROW [Lista Sencilla de Proveedores].[Código del Proveedor], [Lista Sencilla de Proveedores].[Nombre o Razón Social] FROM [Lista Sencilla de Proveedores];

La explicación de esta tabla es totalmente análoga a la anterior, en este caso pensando en los proveedores. Las sucursales de proveedores se editan a través del subformulario [Sucursales] que se encuentra en el formulario [Proveedores] (ver Anexo 1, Figuras 43 y 44, pág. 86 y 87).

#### TELÉFONOS DE CLIENTES

Campo	Tipo	Tamaño	Origen del Registro / Restricciones
<u>Cliente</u>	Texto	10	SELECT DISTINCTROW [Clientes].[Código del Cliente], [Clientes].[Nombre o Razón Social] FROM [Clientes];

<b>Campo</b>	<b>Tipo</b>	<b>Tamaño</b>	<b>Origen del Registro / Restricciones</b>
<u>Teléfono</u>	Texto	15	
Otro Código o Celular	Texto	10	
¿Es Fax?	Sí/No	1	

En esta tabla se registran los distintos números telefónicos (sea el cual sea la cantidad de ellos) de los clientes. Tal vez el único campo que requiera una explicación adicional es [Otro Código o Celular]. Este campo cubre la eventualidad de que dado un país, un estado y una ciudad que contienen la información sobre los códigos de área telefónicos, haga falta todavía especificar algún otro código adicional (como sería el caso de los teléfonos celulares). Los teléfonos de clientes se editan a través del subformulario [Teléfonos] que se encuentra en el formulario [Clientes] (ver Anexo 1, Figuras 2 y 3, pág. 5 y 6).

#### TELÉFONOS DE PROVEEDORES

<b>Campo</b>	<b>Tipo</b>	<b>Tamaño</b>	<b>Origen del Registro / Restricciones</b>
<u>Proveedor</u>	Texto	10	SELECT DISTINCTROW [Proveedores].[Código del Proveedor], [Proveedores].[Nombre o Razón Social] FROM [Proveedores];
<u>Teléfono</u>	Texto	15	
Otro Código o Celular	Texto	10	
¿Es Fax?	Sí/No	1	

La estructura y descripción del funcionamiento de esta tabla es la misma que antes, aplicada al caso de los proveedores en lugar de los clientes. Los teléfonos de proveedores se editan a través del subformulario [Teléfonos] que se encuentra en el formulario [Proveedores] (ver Anexo 1, Figuras 43 y 44, pág. 86 y 87).

#### TENENCIA DE LOCALES

<b>Campo</b>	<b>Tipo</b>	<b>Tamaño</b>	<b>Origen del Registro / Restricciones</b>
<u>Tenencia</u>	Texto	15	

Aquí se definen las posibilidades de tenencia de los locales (de clientes y proveedores) que generalmente son dos: Propio o Alquilado, pero eventualmente podrían ser más. Las tenencias de locales se editan a través del formulario [Tenencia de Locales] (ver Anexo 1, Figura 47, pág. 91), al cual se accede desde donde se requiera presionando el botón [Editar Tenencias de Locales].



**TIPOS DE CLIENTES**

<b>Campo</b>	<b>Tipo</b>	<b>Tamaño</b>	<b>Origen del Registro / Restricciones</b>
Tipo de Cliente	Texto	50	
Descuento	Número (simple)	4	

Esta tabla ha cambiado desde el diseño original. Registra los distintos tipos de clientes (a ser definidos por la organización). El cambio ocurrió con la incorporación del campo [Descuento]. Este campo se ha añadido para registrar un número porcentual, que representa el descuento que podrá en un futuro recibir un cliente de acuerdo con el tipo al que pertenezca (otra modalidad de descuentos). Esta modalidad de descuento no se encuentra soportada por la actual versión del SARCAI, sin embargo, en futuras versiones podrá estarlo. Los tipos de clientes se editan a través del formulario [Tipos de Clientes] (ver Anexo 1, Figura 48, pág. 91), al cual se accede desde el formulario [Clientes] (ver Anexo 1, Figuras 2 y 3, pág. 5 y 6) presionando el botón [Editar Tipos de Clientes].

**TIPOS DE INSUMOS**

<b>Campo</b>	<b>Tipo</b>	<b>Tamaño</b>	<b>Origen del Registro / Restricciones</b>
Tipo de Insumo	Texto	20	
¿Se Controla Potencia?	Sí/No	1	

Esta tabla permite definir los distintos tipos de insumo con que trata la empresa (normalmente Principios Activos, Excipientes, Material de Empaque, Vehículos, etc.). Es de importancia capital, pues aquí el usuario informa al sistema sobre si para un tipo de insumo ha de ser controlada la Potencia (medida de pureza del insumo) o no. Si se controla la potencia del insumo, los cálculos de las cantidades requeridas para la producción que utilice el insumo en cuestión se ven afectados por el ponderador de potencia (calculado cuando llega el insumo para cada nuevo lote), de lo contrario, las cantidades a emplear en la producción no se afectan. Generalmente se controla la potencia de los Principios Activos y no, por ejemplo, de los Materiales de Empaque. Los tipos de insumos se editan a través del formulario [Tipos de Insumos] (ver Anexo 1, Figura 49, pág. 91), al cual se accede desde el formulario [Insumos] (ver Anexo 1, Figura 21, pág. 34) presionando el botón [Editar Tipos de Insumos].

**TIPOS DE ÓRDENES DE PEDIDO**

<b>Campo</b>	<b>Tipo</b>	<b>Tamaño</b>	<b>Origen del Registro / Restricciones</b>
<u>Tipo de Orden</u>	Texto	30	

Aquí se definen los posibles tipos de órdenes de pedido (que deben ser decididos por la organización). Los tipos de órdenes de pedido se editan a través del formulario [Tipos de Órdenes de Pedido] (ver Anexo 1, Figura 50, pág. 92), al cual se accede desde el formulario [Pedidos de Clientes] (ver Anexo 1, Figuras 30 y 31, pág. 52 y 53) presionando el botón [Editar Tipos de Órdenes].

**TIPOS DE PROVEEDORES**

<b>Campo</b>	<b>Tipo</b>	<b>Tamaño</b>	<b>Origen del Registro / Restricciones</b>
<u>Tipo de Proveedor</u>	Texto	50	

Aquí se definen los posibles tipos de proveedores con que cuenta la organización. Esta clasificación está abierta con la finalidad de decidir si se hará en base a la calidad del proveedor o a la magnitud del negocio que se realiza con él. Los tipos de proveedores se editan a través del formulario [Tipos de Proveedores] (ver Anexo 1, Figura 51, pág. 92), al cual se accede desde el formulario [Proveedores] (ver Anexo 1, Figuras 43 y 44, pág. 86 y 87) presionando el botón [Editar Tipos de Proveedores].

**TIPOS DE SALIDAS**

<b>Campo</b>	<b>Tipo</b>	<b>Tamaño</b>	<b>Origen del Registro / Restricciones</b>
<u>Tipo de Salida</u>	Texto	50	

Funciona con la tabla [Salidas del Almacén de Insumos], registrándose aquí todos los tipos posibles de salidas de un insumo por razones distintas a la producción; por ejemplo, robo, vencimiento, etc. Los tipos de salidas se editan a través del formulario [Tipos de Salidas] (ver Anexo 1, Figura 52, pág. 92), al cual se accede desde el formulario [Llegadas de Insumos al Almacén] (ver Anexo 1, Figuras 22 y 23, pág. 36 y 37) presionando el botón [Editar Tipos de Salidas].

**UNIDADES DE MEDICIÓN**

<b>Campo</b>	<b>Tipo</b>	<b>Tamaño</b>	<b>Origen del Registro / Restricciones</b>
<u>Unidades</u>	Texto	10	

Magnitud Física	Texto	15	SELECT DISTINCTROW [Magnitudes Físicas].[Magnitud Física] FROM [Magnitudes Físicas];
-----------------	-------	----	--

Aquí se registran todas las unidades de medición con que trabaja el sistema. Se registra además la magnitud física (tiempo, peso, volumen, unidades, etc.) a la que pertenece la unidad de medición. Por ejemplo, Kg. se registra perteneciendo a la magnitud Peso, Lts. a Volumen, etc. Es muy importante para el sistema un manejo consciente de esta tabla, pues es la base de varios cálculos que se requieren en informes y gráficos, e incluso en los modelos. Al menos deben estar registradas, y de esta forma particular, las unidades Kg., g., Lts., Unidades, Días, Semanas, Quincenas, Meses, Bimestres, Trimestres, Semestres, Años y Bienios. Las unidades de medición se editan a través del formulario [Unidades de Medición] (ver Anexo 1, Figura 53, pág. 93), al cual se accede desde donde se requiera, presionando el botón [Editar Unidades de Medición].

#### VEHÍCULOS

Campo	Tipo	Tamaño	Origen del Registro / Restricciones
Vehículo	Texto	20	

Esta tabla, última en el orden alfabético de la base de datos, registra los distintos vehículos (como alcohol, agua y otros) que eventualmente serán utilizados formando parte de los productos. Este dato se necesita en algunos casos dentro de la tabla [Fórmulas] y forma parte del nombre genérico de cada producto terminado. Los vehículos se editan a través del formulario [Vehículos] (ver Anexo 1, Figura 54, pág. 94), al cual se accede desde el formulario [Fórmulas] (ver Anexo 1, Figuras 18 y 19, pág. 25 y 26) presionando el botón [Editar Vehículos].

### II.2.2. Consultas

Es de aclarar que una comprensión cabal de lo que resta del capítulo requiere algún dominio del lenguaje de consulta formal de bases de datos relacionales SQL (Structured Query Language)<sup>6</sup>. Este es el estándar para la extracción y consolidación de información en bases de datos relacionales y, por supuesto, ACCESS hace un uso intensivo del mismo (por cierto, una de las razones de mayor peso para su escogencia como la plataforma de software de este trabajo). Tratándose de un estándar que con pocos o ningún cambio permite compilar y ejecutar el código fuente en diversas plataformas computacionales, lo que añade versatilidad al producto SARCAI,

pues aminora el esfuerzo de migración de la base de datos a otros sistemas operativos como, por ejemplo, UNIX. Dado que no es la intención del presente trabajo ahondar en los detalles propios del SQL, sólo se expondrán sobre el particular los siguientes aspectos generales:

1. SQL es un lenguaje de consulta, no un lenguaje de programación (esto es, carece de algunas de las estructuras básicas de la programación).
2. Se trata de un lenguaje “no procedimental” en el sentido de que el programador expresa lo que desea, pero no cómo ha de obtenerse lo que desea.
3. Puede utilizarse en cuatro grandes áreas: como lenguaje de consulta propiamente dicho como lenguaje de definición de datos, como lenguaje de acciones sobre los datos y como lenguaje de consultas con funciones de dominio agregado.
4. Como lenguaje de consulta propiamente dicho, utiliza la sentencia `SELECT <Campos> - FROM <Tablas o Consultas> - WHERE <Condición Lógica>;` (con sus variantes). Esta sintaxis se corresponde con el significado de las palabras clave en inglés, es decir, “Seleccione algunos campos (o uno, o todos), tomados de una lista de tablas (puede ser una sola), donde los registros cumplan la condición lógica especificada y muéstrellos (en el mismo formato de una tabla)”.
5. Como lenguaje de definición de datos utiliza principalmente la instrucción `CREATE` para establecer la estructura física de cada tabla de la base de datos. Aunque sería posible haber creado de esta forma las tablas del sistema, el procedimiento utilizado en SARCAI para la creación de las tablas fue distinto. Se usó el asistente en la vista “Tablas”, “Ver Diseño”, del ACCESS, por lo que no se encontrarán sentencias SQL de este tipo.
6. Como lenguaje de acciones sobre los datos, es capaz de realizar inserciones de registros nuevos (sentencia `INSERT`), actualizaciones de registros (sentencia `UPDATE`) y eliminaciones (sentencia `DELETE`). Todas estas sentencias se utilizan en el trabajo.

---

<sup>6</sup> Ver DATE. Op. Cit. p.117

7. Como lenguaje de consultas con funciones de dominio agregado, utiliza la instrucción `SELECT <Campos y Cómputos sobre los grupos de registros> - FROM <Tablas o Consultas> - WHERE <Condición Lógica> - GROUP BY <Criterio de Agrupamiento o Campos>` (con sus variantes). Este tipo de consultas resultan en una síntesis de la información contenida en la base de datos, generalmente expresada en una función estadística como, por ejemplo, la suma, promedio, varianza, conteo, etc. de los valores de un campo, para un grupo de registros que cumple con cierta condición.
8. El SQL: puede ser anidado, esto es, una consulta puede ser el insumo de datos de otra e incluso puede formar parte de la cláusula `WHERE` o `FROM`.
9. Para garantizar que el interpretador del lenguaje “entiende correctamente” los nombres y la proveniencia de los objetos, se utilizan corchetes (`[]`) para aquellos nombres que incluyan caracteres especiales como espacios en blanco o acentos y se usa la sintaxis `<Nombre de la Tabla o Consulta>.<Nombre del Campo>` para nombrar los campos.
10. Con el objeto de brindar la posibilidad de que las condiciones lógicas incluyan operandos variables (cuyos valores puedan ser suministrados por el usuario o los programas), se pueden definir estos operandos como parámetros de la consulta utilizando la sentencia inicial `PARAMETERS <Nombre del Parámetro> <Tipo de Dato>;`.
11. En el sistema, la mayoría de los parámetros no son suministrados directamente por el usuario. Son valores que el usuario ha escrito en un cuadro de texto, dentro de un formulario preparado especialmente para tal fin (Ver Capítulo IV). Por esta razón, las variables llevan el nombre completo del objeto que las contiene, de la forma: `[Forms]![<Nombre del Formulario>]![<Nombre del Cuadro de Texto, dentro del Formulario>]`.

A continuación se presentan las consultas, principalmente utilizadas como insumos de datos para los formularios e informes, con una breve explicación en cada caso (la lectura de la instrucción SQL debería bastar para su comprensión).

#### ACCIONISTAS DE CLIENTES CONSULTA

```
SELECT DISTINCTROW [Accionistas de Clientes].Cliente, [Accionistas de Clientes].Accionista, Personas.Apellidos, Personas.Nombres
```

---

```
FROM Personas INNER JOIN [Accionistas de Clientes]
ON (Personas.[Cédula de Identidad o Clave] = [Accionistas de
  Clientes].Accionista);
```

---

Obtiene los Apellidos y Nombres de todos los accionistas de cada cliente. Realiza una “Reunión”<sup>7</sup> de las relaciones [Personas] y [Accionistas de Clientes] para aquellos registros que comparten el mismo valor de cédula o clave y accionista.

#### CANTIDAD DESPACHADA EN ALMACÉN I

---

```
PARAMETERS [Forms]![Pedir Dos Fechas]![FechaÚnica] DateTime;
SELECT Insumos.[Código de Insumo], IIf(IsNull([Nº de Orden]),0,[Nº de
  Orden]) AS [Orden de P Nº], IIf(IsNull([Lote Nº]),0,[Lote Nº]) AS
  [Nº Lote], IIf(IsNull([Insumos en Órdenes de Producción]![Cantidad a
  Utilizar]),0,[Insumos en Órdenes de Producción]![Cantidad a
  Utilizar]) AS [Cantidad Despachada], [Insumos en Órdenes de
  Producción].[Fecha del Surtido]
FROM Insumos LEFT JOIN [Insumos en Órdenes de Producción]
ON Insumos.[Código de Insumo] = [Insumos en Órdenes de
  Producción].[Código de Insumo]
WHERE ((([Insumos en Órdenes de Producción].[Fecha del
  Surtido])<=[Forms]![Pedir Dos Fechas]![FechaÚnica]));
```

---

Se muestran las cantidades despachadas del almacén de insumos de acuerdo con las órdenes de producción, tal que el surtido haya ocurrido antes o durante la fecha especificada.

#### CANTIDAD DESPACHADA EN ALMACÉN T

---

```
PARAMETERS [Forms]![Pedir Dos Fechas]![FechaÚnica] DateTime;
SELECT [Productos Facturados a Clientes].[Código del Producto],
  [Productos Facturados a Clientes].[Nº de Orden], Sum([Productos
  Facturados a Clientes].[Cantidad Facturada]) AS [Cantidad
  Despachada]
FROM Facturas INNER JOIN [Productos Facturados a Clientes]
ON Facturas.[Nº de Factura] = [Productos Facturados a Clientes].[Nº de
  Factura]
WHERE (((Facturas.[Fecha de Despacho])<=[Forms]![Pedir Dos
  Fechas]![FechaÚnica]))
GROUP BY [Productos Facturados a Clientes].[Código del Producto],
  [Productos Facturados a Clientes].[Nº de Orden];
```

---

<sup>7</sup> Producto cartesiano condicionado, de dos o más tablas.

Se totalizan las cantidades facturadas y despachadas a los clientes (del almacén de productos terminados) agrupando por código del producto y, dentro de éste, por número de la orden de producción, tal que la fecha del despacho de la mercancía haya sido anterior o igual a la fecha especificada.

#### CANTIDAD FACTURADA EN ALMACÉN T

---

```
PARAMETERS [Forms]![Pedir Dos Fechas]![FechaÚnica] DateTime;
SELECT [Productos Facturados a Clientes].[Código del Producto],
       [Productos Facturados a Clientes].[N° de Orden], Sum([Productos
       Facturados a Clientes].[Cantidad Facturada]) AS [Cantidad Facturada]
FROM Facturas INNER JOIN [Productos Facturados a Clientes]
ON Facturas.[N° de Factura] = [Productos Facturados a Clientes].[N° de
Factura]
WHERE (((Facturas.[Fecha de Emisión])<=[Forms]![Pedir Dos
Fechas]![FechaÚnica]))
GROUP BY [Productos Facturados a Clientes].[Código del Producto],
         [Productos Facturados a Clientes].[N° de Orden];
```

---

Se totalizan las cantidades facturadas a los clientes agrupando por código del producto y, dentro de éste, por número de la orden de producción, tal que la fecha del despacho de la mercancía haya sido anterior o igual a la fecha especificada.

#### CANTIDAD ORDENADA EN ALMACÉN I

---

```
PARAMETERS [Forms]![Pedir Dos Fechas]![FechaÚnica] DateTime;
SELECT Insumos.[Código de Insumo], [Insumos en Órdenes].[Orden N°] AS [N°
Orden], [Órdenes de Compra].Fecha, [Insumos en Órdenes].[Precio
Unitario], [Insumos en Órdenes].Cantidad AS [Cantidad Ordenada]
FROM [Órdenes de Compra] INNER JOIN (Insumos INNER JOIN [Insumos en
Órdenes] ON Insumos.[Código de Insumo] = [Insumos en
Órdenes].Insumo) ON [Órdenes de Compra].[N°] = [Insumos en
Órdenes].[Orden N°]
WHERE ((([Órdenes de Compra].Fecha)<=[Forms]![Pedir Dos
Fechas]![FechaÚnica]));
```

---

Se listan todas las requisiciones de insumos hechas a los proveedores mediante órdenes de compra, tal que la fecha de la orden es inferior o igual a la fecha especificada.

**CANTIDAD QUE HA SALIDO DE INSUMO POR LOTE**


---

```
SELECT [Cantidad que ha Salido de Insumo por Lote 0].[Lote N°],
       IIf(IsNull([SumaDeCantidad]),0,[SumaDeCantidad]) AS Salidas
FROM [Cantidad que ha Salido de Insumo por Lote 0];
```

---

Se trata de la consulta final que muestra la cantidad de cada insumo que ha salido del almacén de insumos por razones distintas a la producción. Utiliza una consulta previa como la fuente de datos (ver la consulta [Cantidad que ha Salido de Insumo por Lote 0]) y muestra, para cada lote de insumos, la cantidad que ha salido si ésta es positiva, y el número 0 en caso contrario.

**CANTIDAD QUE HA SALIDO DE INSUMO POR LOTE 0**


---

```
SELECT [Salidas del Almacén de Insumos].[Lote N°], Sum([Salidas del
       Almacén de Insumos].Cantidad) AS SumaDeCantidad
FROM [Salidas del Almacén de Insumos]
GROUP BY [Salidas del Almacén de Insumos].[Lote N°];
```

---

Se agrupan los registros por el número del lote y se totalizan, para cada uno de los lotes, las cantidades que han salido por razones distintas a la producción. Ciertos cómputos requieren que las consultas, fuentes de datos, no sean del tipo con funciones de dominio agregado, por lo tanto, esta consulta opera conjuntamente con una consulta adicional (que muestra algo muy similar) que sólo utiliza una proposición de selección (Ver la consulta [Cantidad que ha Salido de Insumo por Lote]).

**CANTIDAD RECIBIDA DE INSUMO POR ORDEN**


---

```
SELECT [Cantidad Recibida de Insumo por Orden 0].[Orden N°], [Cantidad
       Recibida de Insumo por Orden 0].Insumo, [Insumos en
       Órdenes].Cantidad AS Solicitado, IIf(IsNull([Ha llegado]),0,[Ha
       llegado]) AS Recibido, [Solicitado]-[Recibido] AS Resta
FROM [Cantidad Recibida de Insumo por Orden 0] INNER JOIN [Insumos en
       Órdenes]
ON ([Cantidad Recibida de Insumo por Orden 0].Insumo = [Insumos en
       Órdenes].Insumo) AND ([Cantidad Recibida de Insumo por Orden
       0].[Orden N°] = [Insumos en Órdenes].[Orden N°]);
```

---



Muestra las cantidades de insumos que se han recibido asociadas con cada orden de compra. Muestra además las cantidades que restan por recibirse de cada orden (sustrayendo lo recibido de lo ordenado).

#### CANTIDAD RECIBIDA DE INSUMO POR ORDEN 0

---

```
SELECT [Insumos en Órdenes].[Orden N°], [Insumos en Órdenes].Insumo,
       Sum([Llegadas de Insumos al Almacén].Cantidad) AS [Ha llegado]
FROM [Insumos en Órdenes] LEFT JOIN [Llegadas de Insumos al Almacén] ON
     ([Insumos en Órdenes].Insumo = [Llegadas de Insumos al
     Almacén].Insumo) AND ([Insumos en Órdenes].[Orden N°] = [Llegadas de
     Insumos al Almacén].[Orden N°])
GROUP BY [Insumos en Órdenes].[Orden N°], [Insumos en Órdenes].Insumo;
```

---

Agrupando por orden de compra y dentro de ésta por insumo, se totalizan las cantidades que han arribado al almacén provenientes de los proveedores.

#### CANTIDAD RECIBIDA EN ALMACÉN I

---

```
PARAMETERS [Forms]![Pedir Dos Fechas]![FechaÚnica] DateTime;
SELECT DISTINCTROW Insumos.[Código de Insumo], Iif(IsNull([Lote
N°]),0,[Lote N°]) AS [N° Lote], [Insumos Ordenados y
Recibidos].[Precio Unitario], [Insumos Ordenados y Recibidos].Fecha,
Iif(IsNull([Insumos Ordenados y Recibidos].[Cantidad]),0,[Insumos
Ordenados y Recibidos].[Cantidad]) AS [Cantidad Recibida]
FROM Insumos INNER JOIN ([Insumos en Órdenes] LEFT JOIN [Insumos
Ordenados y Recibidos] ON [Insumos en Órdenes].Insumo = [Insumos
Ordenados y Recibidos].Insumo) ON Insumos.[Código de Insumo] =
[Insumos en Órdenes].Insumo
WHERE ((([Insumos Ordenados y Recibidos].Fecha)<=[Forms]![Pedir Dos
Fechas]![FechaÚnica]));
```

---

Se presentan las cantidades de cada insumo y su precio unitario para aquellos de los que se ha recibido al menos una parte (cero en otro caso), tales que la fecha de la recepción en almacén es inferior o igual a una fecha dada.

#### CANTIDAD TOTAL DESPACHADA EN ALMACÉN I

---

```
SELECT [Cantidad Despachada en Almacén I].[Código de Insumo],
       Sum([Cantidad Despachada en Almacén I].[Cantidad Despachada]) AS
       [Cantidad Total Despachada]
```

---

---

```
FROM [Cantidad Despachada en Almacén I]
GROUP BY [Cantidad Despachada en Almacén I].[Código de Insumo];
```

---

Agrupando por insumo se totalizan las cantidades que han sido despachadas para la producción. Se utiliza como fuente de datos la consulta [Cantidad Despachada en Almacén I], la cual restringe el espacio de respuesta de acuerdo con una fecha dada, por lo tanto, el agrupamiento hereda el parámetro de la consulta fuente (esto quiere decir que en realidad se agrupan los registros cuya fecha de despacho es inferior o igual que la fecha dada).

#### **CANTIDAD TOTAL DESECHADA EN ALMACÉN I**

---

```
SELECT [Salidas por Lote en Almacén I].Insumo, Sum([Salidas por Lote en
    Almacén I].Salidas) AS [Cantidad Total Desechada]
FROM [Salidas por Lote en Almacén I]
GROUP BY [Salidas por Lote en Almacén I].Insumo;
```

---

Agrupando por insumo se totalizan las cantidades que han salido del almacén por razones distintas a la producción. Se utiliza como fuente de datos la consulta [Salidas por Lote en Almacén I], la cual restringe el espacio de respuesta de acuerdo con una fecha dada, por lo tanto, el agrupamiento hereda el parámetro de la consulta fuente (esto quiere decir que en realidad se agrupan los registros cuya fecha de salida es inferior o igual que la fecha dada).

#### **CANTIDAD TOTAL DESPACHADA EN ALMACÉN T**

---

```
SELECT [Cantidad Despachada en Almacén T].[Código del Producto],
    Sum([Cantidad Despachada en Almacén T].[Cantidad Despachada]) AS
    [Total Despachado]
FROM [Cantidad Despachada en Almacén T]
GROUP BY [Cantidad Despachada en Almacén T].[Código del Producto];
```

---

Agrupando por producto se totalizan las cantidades que han sido despachadas a los clientes (en el almacén de productos terminados). Como antes, se utiliza como fuente de datos la consulta [Cantidad Despachada en Almacén T], la cual restringe el espacio de respuesta de acuerdo con

una fecha dada. El agrupamiento hereda entonces el parámetro de la consulta fuente (se agrupan los registros cuya fecha de despacho es inferior o igual que la fecha dada).

#### **CANTIDAD TOTAL FACTURADA EN ALMACÉN T**

---

```
SELECT [Cantidad Facturada en Almacén T].[Código del Producto],
       Sum([Cantidad Facturada en Almacén T].[Cantidad Facturada]) AS
       [SumaDeCantidad Facturada]
FROM [Cantidad Facturada en Almacén T]
GROUP BY [Cantidad Facturada en Almacén T].[Código del Producto];
```

---

Agrupando por producto se totalizan las cantidades que han sido facturadas (no todas necesariamente despachadas) a los clientes en el almacén de productos terminados. Al igual que antes, se utiliza como fuente de datos la consulta previa que restringe el espacio de respuesta de acuerdo con una fecha dada. El agrupamiento hereda entonces el parámetro de la consulta fuente.

#### **CANTIDAD TOTAL FACTURADA POR PEDIDO**

---

```
SELECT DISTINCTROW Facturas.[N° de Pedido], [Productos Facturados a
       Clientes].[Código del Producto], Sum([Productos Facturados a
       Clientes].[Cantidad Facturada]) AS [SumaDeCantidad Facturada]
FROM Facturas INNER JOIN [Productos Facturados a Clientes] ON
       Facturas.[N° de Factura] = [Productos Facturados a Clientes].[N° de
       Factura]
GROUP BY Facturas.[N° de Pedido], [Productos Facturados a
       Clientes].[Código del Producto];
```

---

Agrupando de acuerdo con el número del pedido y dentro de éste por producto, se calcula el total facturado a los clientes en el almacén de productos terminados. La utilidad de la consulta está en que permitirá comparar lo solicitado por el cliente con lo facturado y/o lo despachado.

#### **CANTIDAD TOTAL ORDENADA EN ALMACÉN I**

---

```
SELECT [Cantidad Ordenada en Almacén I].[Código de Insumo], Sum([Cantidad
       Ordenada en Almacén I].[Cantidad Ordenada]) AS [Cantidad Total
       Ordenada]
FROM [Cantidad Ordenada en Almacén I]
GROUP BY [Cantidad Ordenada en Almacén I].[Código de Insumo];
```

---

Agrupando por insumo se totaliza la cantidad que ha sido ordenada en el almacén de insumos a los proveedores. Se hereda un parámetro fecha que restringe la respuesta a que la fecha de la orden de compra sea inferior o igual a una fecha dada.

#### CANTIDAD TOTAL PEDIDA EN ALMACÉN T

---

```
PARAMETERS [Forms]![Pedir Dos Fechas]![FechaÚnica] DateTime;
SELECT [Productos Pedidos de Clientes].[Código del Producto],
       Sum([Productos Pedidos de Clientes].[Cantidad Pedida]) AS
       [SumaDeCantidad Pedida]
FROM [Pedidos de Clientes] INNER JOIN [Productos Pedidos de Clientes] ON
     [Pedidos de Clientes].[N° de Pedido] = [Productos Pedidos de
     Clientes].[N° de Pedido]
WHERE ((([Pedidos de Clientes].Fecha)<=[Forms]![Pedir Dos
     Fechas]![FechaÚnica]))
GROUP BY [Productos Pedidos de Clientes].[Código del Producto];
```

---

Agrupando por productos se totalizan las cantidades que han sido pedidas por los clientes al almacén de productos terminados, considerando solamente aquellos pedidos ocurridos antes o durante una fecha dada.

#### CANTIDAD TOTAL PRODUCIDA EN ALMACÉN T

---

```
PARAMETERS [Forms]![Pedir Dos Fechas]![FechaÚnica] DateTime;
SELECT [Productos Terminados].[Código del Producto], Sum(Int([Producción
     Efectiva]/[Cantidad en Presentación])) AS [Total Producido]
FROM [Productos Terminados] INNER JOIN [Órdenes de Producción] ON
     [Productos Terminados].[Código del Producto] = [Órdenes de
     Producción].Producto
WHERE ((([Órdenes de Producción].Fecha de Aprobación)<=[Forms]![Pedir
     Dos Fechas]![FechaÚnica]))
GROUP BY [Productos Terminados].[Código del Producto];
```

---

Agrupando por productos se totalizan las cantidades efectivamente producidas (no las ordenadas) considerando su presentación final unitaria (frascos, cajas u otras) en el almacén de productos terminados. Se consideran solamente aquellas órdenes de producción aprobadas antes o durante una fecha dada.

**CANTIDAD TOTAL RECIBIDA EN ALMACÉN I**


---

```

SELECT [Cantidad Recibida en Almacén I].[Código de Insumo], Sum([Cantidad
  Recibida en Almacén I].[Cantidad Recibida]) AS [Cantidad Total
  Recibida]
FROM [Cantidad Recibida en Almacén I]
GROUP BY [Cantidad Recibida en Almacén I].[Código de Insumo];

```

---

Agrupando por insumo se totalizan las cantidades recibidas de cada uno en el almacén de insumos, provenientes de los proveedores. Se hereda un parámetro fecha que restringe la respuesta a que la recepción haya ocurrido antes o durante una fecha dada.

**CANTIDADES EN ALMACÉN I**


---

```

SELECT Insumos.[Código de Insumo], Insumos.[Tipo de Insumo],
  Insumos.Nombre, Insumos.[Unidades de Presentación], [Insumos en
  Órdenes].[Orden N°], [Insumos en Órdenes].[Precio Unitario],
  [Insumos en Órdenes].Cantidad AS [Cantidad Ordenada], [Órdenes de
  Compra].[Cargos %], [Llegadas de Insumos al Almacén].[Lote N°],
  [Llegadas de Insumos al Almacén].Fecha, [Llegadas de Insumos al
  Almacén].Cantidad AS [Cantidad Recibida]
FROM ([Órdenes de Compra] INNER JOIN (Insumos INNER JOIN [Insumos en
  Órdenes] ON Insumos.[Código de Insumo] = [Insumos en
  Órdenes].Insumo) ON [Órdenes de Compra].[N°] = [Insumos en
  Órdenes].[Orden N°]) INNER JOIN [Llegadas de Insumos al Almacén] ON
  ([Órdenes de Compra].[N°] = [Llegadas de Insumos al Almacén].[Orden
  N°]) AND (Insumos.[Código de Insumo] = [Llegadas de Insumos al
  Almacén].Insumo);

```

---

Se muestran datos consolidados de varias tablas, sobre las llegadas, precios, fechas y otros de los insumos en el almacén de insumos.

**CLIENTES CONSULTA**


---

```

SELECT DISTINCTROW Clientes.[Código del Cliente], Clientes.[Nombre o
  Razón Social], Clientes.[CI o RIF], Clientes.NIT,
  Clientes.Dirección, Clientes.País, Países.[Código Telefónico],
  Clientes.Estado, Clientes.Ciudad, [Ciudades de Estados].[Código
  Telefónico], Clientes.[Tenencia del Local], Clientes.[Figura
  Jurídica], Clientes.[N° del RC], Clientes.[Fecha del RC],
  Clientes.[Duración del RC], Clientes.[Tipo de Cliente],
  Clientes.Observaciones, Clientes.Responsable, Personas.Apellidos,
  Personas.Nombres, Clientes.Grupo, Clientes.Sector
FROM Personas RIGHT JOIN ([Ciudades de Estados] RIGHT JOIN (Clientes LEFT

```

---

---

```

JOIN Países ON Clientes.País = Países.País) ON ([Ciudades de
Estados].Ciudad = Clientes.Ciudad) AND ([Ciudades de Estados].Estado
= Clientes.Estado) AND ([Ciudades de Estados].País = Clientes.País))
ON Personas.[Cédula de Identidad o Clave] = Clientes.Responsable;

```

---

Se muestran datos consolidados de varias tablas sobre los clientes.

#### CLIENTES Y FACTURAS G

---

```

SELECT [Lista Sencilla de Clientes].Grupo, Sum([Facturas Totales
Despachadas].[Monto Total Surtido]) AS [Total Surtido]
FROM (([Pedidos de Clientes] INNER JOIN [Lista Sencilla de Clientes] ON
[Pedidos de Clientes].[Código del Cliente] = [Lista Sencilla de
Clientes].[Código del Cliente]) INNER JOIN Facturas ON [Pedidos de
Clientes].[N° de Pedido] = Facturas.[N° de Pedido]) INNER JOIN
[Facturas Totales Despachadas] ON Facturas.[N° de Factura] =
[Facturas Totales Despachadas].[N° de Factura]
WHERE ((([Pedidos de Clientes].Fecha) Between [Forms]![Pedir Fechas para
Ventas]![FechaInicial] And [Forms]![Pedir Fechas para
Ventas]![FechaFinal])) OR ((([Pedidos de
Clientes].Fecha)=[Forms]![Pedir Fechas para Ventas]![FechaInicial]
Or ([Pedidos de Clientes].Fecha)=[Forms]![Pedir Fechas para
Ventas]![FechaFinal]) Is Null))
GROUP BY [Lista Sencilla de Clientes].Grupo;

```

---

Se totalizan las cantidades (en unidades monetarias -Bs.-) surtidas a los clientes, agrupándolos por grupo de cliente, tales que hayan sido pedidas y surtidas entre dos fechas dadas como parámetros.

#### CLIENTES Y FACTURAS S

---

```

SELECT [Lista Sencilla de Clientes].Sector, Sum([Facturas Totales
Despachadas].[Monto Total Surtido]) AS [Total Surtido]
FROM ([Pedidos de Clientes] INNER JOIN [Lista Sencilla de Clientes] ON
[Pedidos de Clientes].[Código del Cliente] = [Lista Sencilla de
Clientes].[Código del Cliente]) INNER JOIN (Facturas INNER JOIN
[Facturas Totales Despachadas] ON Facturas.[N° de Factura] =
[Facturas Totales Despachadas].[N° de Factura]) ON [Pedidos de
Clientes].[N° de Pedido] = Facturas.[N° de Pedido]
WHERE ((([Pedidos de Clientes].Fecha) Between [Forms]![Pedir Fechas para
Ventas]![FechaInicial] And [Forms]![Pedir Fechas para
Ventas]![FechaFinal])) OR ((([Pedidos de
Clientes].[Fecha]]=[Forms]![Pedir Fechas para Ventas]![FechaInicial]
Or ([Pedidos de Clientes].[Fecha]]=[Forms]![Pedir Fechas para
Ventas]![FechaFinal]) Is Null))
GROUP BY [Lista Sencilla de Clientes].Sector;

```

---

Se totalizan las cantidades surtidas a los clientes (en unidades monetarias - Bs. -), agrupándolos por el sector al que pertenecen, tales que hayan sido pedidas y surtidas entre dos fechas dadas como parámetros.

#### CLIENTES Y FACTURAS V

---

```

SELECT [Pedidos de Clientes].Vendedor, Sum([Facturas Totales
    Despachadas].[Monto Total Surtido]) AS [Total Surtido]
FROM [Pedidos de Clientes] INNER JOIN (Facturas INNER JOIN [Facturas
    Totales Despachadas] ON Facturas.[N° de Factura] = [Facturas Totales
    Despachadas].[N° de Factura]) ON [Pedidos de Clientes].[N° de
    Pedido] = Facturas.[N° de Pedido]
WHERE ((([Pedidos de Clientes].Fecha) Between [Forms]![Pedir Fechas para
    Ventas]![FechaInicial] And [Forms]![Pedir Fechas para
    Ventas]![FechaFinal])) OR ((([Pedidos de
    Clientes].[Fecha])=[Forms]![Pedir Fechas para Ventas]![FechaInicial]
    Or ([Pedidos de Clientes].[Fecha])=[Forms]![Pedir Fechas para
    Ventas]![FechaFinal]) Is Null))
GROUP BY [Pedidos de Clientes].Vendedor;

```

---

Se totalizan las cantidades surtidas a los clientes (en unidades monetarias - Bs. -), agrupándolos por el vendedor que logró el pedido, tales que hayan sido pedidas y surtidas entre dos fechas dadas como parámetros.

#### CLIENTES Y PEDIDOS G

---

```

PARAMETERS [Forms]![Pedir Fechas para Ventas]![FechaInicial] DateTime,
    [Forms]![Pedir Fechas para Ventas]![FechaFinal] DateTime;
SELECT [Lista Sencilla de Clientes].Grupo, Sum([Pedidos Totales entre
    Fechas].[Monto Total Pedido]) AS [Total Pedido]
FROM ([Pedidos de Clientes] INNER JOIN [Lista Sencilla de Clientes] ON
    [Pedidos de Clientes].[Código del Cliente] = [Lista Sencilla de
    Clientes].[Código del Cliente]) INNER JOIN [Pedidos Totales entre
    Fechas] ON [Pedidos de Clientes].[N° de Pedido] = [Pedidos Totales
    entre Fechas].[N° de Pedido]
GROUP BY [Lista Sencilla de Clientes].Grupo
ORDER BY Sum([Pedidos Totales entre Fechas].[Monto Total Pedido]) DESC;

```

---

Se totalizan las cantidades pedidas por los clientes (en unidades monetarias - Bs. -), agrupándolos por grupo de cliente, tales que hayan sido pedidas y surtidas entre dos fechas dadas

como parámetros. La respuesta se ordena en forma decreciente (de mayor a menor) por el total pedido de cada grupo; se visualizan fácilmente entonces aquellos grupos de clientes que entre las fechas dadas han pedido un monto mayor en dinero.

#### CLIENTES Y PEDIDOS S

---

```
PARAMETERS [Forms]![Pedir Fechas para Ventas]![FechaInicial] DateTime,
           [Forms]![Pedir Fechas para Ventas]![FechaFinal] DateTime;
SELECT [Lista Sencilla de Clientes].Sector, Sum([Pedidos Totales entre
Fechas].[Monto Total Pedido]) AS [Total Pedido]
FROM ([Pedidos de Clientes] INNER JOIN [Lista Sencilla de Clientes] ON
[Pedidos de Clientes].[Código del Cliente] = [Lista Sencilla de
Clientes].[Código del Cliente]) INNER JOIN [Pedidos Totales entre
Fechas] ON [Pedidos de Clientes].[N° de Pedido] = [Pedidos Totales
entre Fechas].[N° de Pedido]
GROUP BY [Lista Sencilla de Clientes].Sector
ORDER BY Sum([Pedidos Totales entre Fechas].[Monto Total Pedido]) DESC;
```

---

Se totalizan las cantidades pedidas por los clientes (en unidades monetarias -Bs.-), agrupándolos por el sector a que pertenecen, tales que hayan sido pedidas y surtidas entre dos fechas dadas como parámetros. La respuesta se ordena en forma decreciente (de mayor a menor) por el total pedido de cada sector; se visualizan fácilmente entonces aquellos sectores de clientes que entre las fechas dadas han pedido un monto mayor en dinero.

#### CLIENTES Y PEDIDOS V

---

```
PARAMETERS [Forms]![Pedir Fechas para Ventas]![FechaInicial] DateTime,
           [Forms]![Pedir Fechas para Ventas]![FechaFinal] DateTime;
SELECT [Pedidos de Clientes].Vendedor, Sum([Pedidos Totales entre
Fechas].[Monto Total Pedido]) AS [Total Pedido]
FROM [Pedidos de Clientes] INNER JOIN [Pedidos Totales entre Fechas] ON
[Pedidos de Clientes].[N° de Pedido] = [Pedidos Totales entre
Fechas].[N° de Pedido]
GROUP BY [Pedidos de Clientes].Vendedor
ORDER BY Sum([Pedidos Totales entre Fechas].[Monto Total Pedido]) DESC;
```

---

Se totalizan las cantidades pedidas por los clientes (en unidades monetarias -Bs.-), agrupándolos por el vendedor que obtuvo el pedido, tales que hayan sido pedidas y surtidas entre dos fechas dadas. La respuesta se ordena en forma decreciente (de mayor a menor) por el total pedido a cada vendedor; se visualizan fácilmente entonces aquellos vendedores que entre las fechas dadas han logrado un mayor volumen de ventas (en dinero).



**CLIENTES, PEDIDOS Y FACTURAS G**


---

```
PARAMETERS [Forms]![Pedir Fechas para Ventas]![FechaInicial] DateTime,
[Forms]![Pedir Fechas para Ventas]![FechaFinal] DateTime;
SELECT [Clientes y Pedidos G].Grupo, [Clientes y Pedidos G].[Total
Pedido], [Clientes y Facturas G].[Total Surtido]
FROM [Clientes y Facturas G] INNER JOIN [Clientes y Pedidos G] ON
[Clientes y Facturas G].Grupo = [Clientes y Pedidos G].Grupo
ORDER BY [Clientes y Pedidos G].[Total Pedido] DESC;
```

---

Se muestran las cantidades pedidas por los clientes y surtidas (en unidades monetarias -Bs.-), previamente agrupadas en la consulta fuente por grupos, tales que hayan sido pedidas y surtidas entre dos fechas dadas. La respuesta se ordena en forma decreciente por el total pedido por cada grupo; se visualizan fácilmente entonces las diferencias entre lo pedido y lo surtido.

**CLIENTES, PEDIDOS Y FACTURAS S**


---

```
SELECT [Clientes y Pedidos S].Sector, [Clientes y Pedidos S].[Total
Pedido], [Clientes y Facturas S].[Total Surtido]
FROM [Clientes y Facturas S] INNER JOIN [Clientes y Pedidos S] ON
[Clientes y Facturas S].Sector = [Clientes y Pedidos S].Sector
ORDER BY [Clientes y Pedidos S].[Total Pedido] DESC;
```

---

Se muestran las cantidades pedidas por los clientes y surtidas (en unidades monetarias -Bs.-), previamente agrupadas en la consulta fuente por sectores, tales que hayan sido pedidas y surtidas entre dos fechas dadas. La respuesta se ordena en forma decreciente por el total pedido por cada sector; se visualizan fácilmente entonces las diferencias entre lo pedido y lo surtido.

**CLIENTES, PEDIDOS Y FACTURAS V**


---

```
SELECT Trim([Apellidos]) & ", " & Trim([Nombres]) AS Agente, [Clientes y
Pedidos V].[Total Pedido], [Clientes y Facturas V].[Total Surtido]
FROM Personas INNER JOIN ([Clientes y Pedidos V] INNER JOIN [Clientes y
Facturas V] ON [Clientes y Pedidos V].Vendedor = [Clientes y
Facturas V].Vendedor) ON Personas.[Cédula de Identidad o Clave] =
[Clientes y Pedidos V].Vendedor
ORDER BY [Clientes y Pedidos V].[Total Pedido] DESC;
```

---

Se muestran las cantidades pedidas por los clientes y surtidas (en unidades monetarias -Bs.-), previamente agrupadas en la consulta fuente por vendedor, tales que hayan sido pedidas y surtidas entre dos fechas dadas. La respuesta se ordena en forma decreciente por el total pedido a cada vendedor; se visualizan fácilmente las diferencias entre lo pedido y lo surtido.

**CONTACTOS EN CLIENTES CONSULTA**


---

```
SELECT DISTINCTROW [Contactos en Clientes].Cliente, Personas.[Cédula de
  Identidad o Clave], Personas.Apellidos, Personas.Nombres
FROM Personas RIGHT JOIN [Contactos en Clientes] ON Personas.[Cédula de
  Identidad o Clave] = [Contactos en Clientes].Contacto;
```

---

Se muestran apellidos y nombres de todas las personas con las cuales se entiende la organización cuando trata con cada cliente (los contactos).

**CONTACTOS EN PROVEEDORES CONSULTA**


---

```
SELECT DISTINCTROW Proveedores.[Código del Proveedor], Personas.[Cédula
  de Identidad o Clave], Personas.Apellidos, Personas.Nombres
FROM Proveedores INNER JOIN (Personas INNER JOIN [Contactos en
  Proveedores] ON Personas.[Cédula de Identidad o Clave] = [Contactos
  en Proveedores].Contacto) ON Proveedores.[Código del Proveedor] =
  [Contactos en Proveedores].Proveedor;
```

---

Se muestran apellidos y nombres de todas las personas con las cuales se entiende la organización cuando trata con cada proveedor (los contactos).

**CONVERSIÓN DE INSUMOS**


---

```
SELECT DISTINCTROW [Insumos en Fórmulas].[N° de Fórmula], [Insumos en
  Fórmulas].[Código de Insumo], [Insumos en
  Fórmulas].[Cantidad]*[Conversiones].[Ponderador] AS Cantidad,
  Conversiones.A, Insumos.Nombre
FROM (Insumos INNER JOIN Conversiones ON Insumos.[Unidades de
  Presentación] = Conversiones.De) INNER JOIN [Insumos en Fórmulas] ON
  Insumos.[Código de Insumo] = [Insumos en Fórmulas].[Código de
  Insumo]
WHERE (((Conversiones.A)="Kg." Or (Conversiones.A)="Lts."));
```

---

Convierte las cantidades de insumos, necesarias para la preparación de los productos en la fórmula, donde aplique, de cualquier unidad en se expresen originalmente a Kg. o a Lts., utilizando la tabla de conversiones. Esta consulta se utiliza en la totalización de Kg. o Lts. de insumos requeridos para la preparación de cualquiera de los productos terminados.

**DEMANDA BIMENSUAL**


---

```

SELECT Year([Fecha Base]) AS Año, IIf(Month([Fecha
    Base])<=2,1,IIf(Month([Fecha Base])<=4,2,IIf(Month([Fecha
    Base])<=6,3,IIf(Month([Fecha Base])<=8,4,IIf(Month([Fecha
    Base])<=10,5,6)))) AS Bimestre, Sum([Demanda por Fechas
    Tabla].[Demanda Puntual]) AS [SumaDeDemanda Puntual]
FROM [Demanda por Fechas Tabla]
GROUP BY Year([Fecha Base]), IIf(Month([Fecha
    Base])<=2,1,IIf(Month([Fecha Base])<=4,2,IIf(Month([Fecha
    Base])<=6,3,IIf(Month([Fecha Base])<=8,4,IIf(Month([Fecha
    Base])<=10,5,6)))));

```

---

Agrupando por año y dentro de éste creando y agrupando por bimestre (dos meses), se totaliza la demanda, medida en número de unidades pedidas (puntual), previamente calculada en [Demanda por Fechas Tabla].

**DEMANDA MENSUAL**


---

```

SELECT Year([Fecha Base]) AS Año, Month([Fecha Base]) AS Mes,
    Sum([Demanda por Fechas Tabla].[Demanda Puntual]) AS [SumaDeDemanda
    Puntual]
FROM [Demanda por Fechas Tabla]
GROUP BY Year([Fecha Base]), Month([Fecha Base]);

```

---

Agrupando por año y dentro de éste por mes, se totaliza la demanda, medida en número de unidades pedidas (puntual), previamente calculada en [Demanda por Fechas Tabla].

**DEMANDA POR FECHAS**


---

```

PARAMETERS Forms![Pedir Fechas para la Demanda]!FechaInicial DateTime,
    Forms![Pedir Fechas para la Demanda]!FechaFinal DateTime;
SELECT [Demanda por Fechas 0].Producto, [Lista Sencilla de Productos
    Terminados].Nombre, [Demanda por Fechas 0].[Fecha Base], [Demanda
    por Fechas 0].[Demanda Puntual], [Demanda por Fechas 0].[Demanda en
    Bs]
FROM [Demanda por Fechas 0] INNER JOIN [Lista Sencilla de Productos
    Terminados] ON [Demanda por Fechas 0].Producto = [Lista Sencilla de
    Productos Terminados].[Código del Producto]
WHERE ((([Demanda por Fechas 0].[Fecha Base]) Between [Forms]![Pedir
    Fechas para la Demanda]!FechaInicial And [Forms]![Pedir Fechas
    para la Demanda]!FechaFinal)) OR ((([Fecha Base]=[Forms]![Pedir
    Fechas para la Demanda]!FechaInicial) Or ([Fecha
    Base]=[Forms]![Pedir Fechas para la Demanda]!FechaFinal)) Is
    Null));

```

---

Dadas dos fechas (si se omiten, se calcula sin restricciones), se muestran los productos terminados y las demandas puntuales ocurridas, en unidades y en dinero, para aquellas transacciones que se hayan dado dentro de tales fechas.

#### DEMANDA POR FECHAS 0

---

```

SELECT [Productos Demandados].Producto, TransformaFecha([Fecha]) AS
  [Fecha Base], Sum([Productos Pedidos de Clientes].[Cantidad Pedida])
AS [Demanda Puntual], Sum([Productos Pedidos y Precios 3].[Monto del
Producto]) AS [Demanda en Bs]
FROM [Pedidos de Clientes] INNER JOIN (([Productos Demandados] INNER JOIN
[Productos Pedidos de Clientes] ON [Productos Demandados].Producto =
[Productos Pedidos de Clientes].[Código del Producto]) INNER JOIN
[Productos Pedidos y Precios 3] ON ([Productos Pedidos de
Clientes].[Código del Producto] = [Productos Pedidos y Precios
3].[Código del Producto]) AND ([Productos Pedidos de Clientes].[N°
de Pedido] = [Productos Pedidos y Precios 3].[N° de Pedido])) ON
[Pedidos de Clientes].[N° de Pedido] = [Productos Pedidos de
Clientes].[N° de Pedido]
GROUP BY [Productos Demandados].Producto, TransformaFecha([Fecha]);

```

---

Agrupando por productos y dentro de éstos quitando la sección “hora” de cada fecha, y agrupando (ver la Función TransformaFecha en los anexos), se totalizan las cantidades, puntuales y en dinero, demandadas.

#### DEMANDA POR FECHAS CREACIÓN

---

```

PARAMETERS Forms![Pedir Fechas para la Demanda]!FechaInicial DateTime,
  Forms![Pedir Fechas para la Demanda]!FechaFinal DateTime;
INSERT INTO [Demanda por Fechas Tabla] ( Producto, [Fecha Base], [Demanda
Puntual], [Demanda en Bs] )
SELECT [Demanda por Fechas 0].Producto, [Demanda por Fechas 0].[Fecha
Base], [Demanda por Fechas 0].[Demanda Puntual], [Demanda por Fechas
0].[Demanda en Bs]
FROM [Demanda por Fechas 0]
WHERE ((([Demanda por Fechas 0].[Fecha Base]) Between [Forms]![Pedir
Fechas para la Demanda]!FechaInicial And [Forms]![Pedir Fechas
para la Demanda]!FechaFinal)) OR ((([Fecha Base]=[Forms]![Pedir
Fechas para la Demanda]!FechaInicial) Or ([Fecha
Base]=[Forms]![Pedir Fechas para la Demanda]!FechaFinal)) Is
Null));

```

---

Esta consulta inserta nuevos registros a la base de datos. Llena la tabla [Demanda Por Fechas Tabla] que contiene, entre dos fechas señaladas, para cada producto, lo demandado en unidades y en dinero. Normalmente se ejecuta luego de vaciar el contenido previo de la tabla y se utiliza para la creación de algunos informes y gráficos.

#### DEMANDA PROMEDIO BIMENSUAL

---

```
SELECT [Demanda Bimensual].Bimestre AS Período, Avg([Demanda
    Bimensual].[SumaDeDemanda Puntual]) AS Promedio, Min([Demanda
    Bimensual].[SumaDeDemanda Puntual]) AS Mínima, Max([Demanda
    Bimensual].[SumaDeDemanda Puntual]) AS Máxima, StDev([Demanda
    Bimensual].[SumaDeDemanda Puntual]) AS DS
FROM [Demanda Bimensual]
GROUP BY [Demanda Bimensual].Bimestre;
```

---

Calcula la demanda promedio, mínima, máxima y su desviación estándar, agrupando por bimestres.

#### DEMANDA PROMEDIO MENSUAL

---

```
SELECT [Demanda Mensual].Mes AS Período, Avg([Demanda
    Mensual].[SumaDeDemanda Puntual]) AS Promedio, Min([Demanda
    Mensual].[SumaDeDemanda Puntual]) AS Mínima, Max([Demanda
    Mensual].[SumaDeDemanda Puntual]) AS Máxima, StDev([Demanda
    Mensual].[SumaDeDemanda Puntual]) AS DS
FROM [Demanda Mensual]
GROUP BY [Demanda Mensual].Mes;
```

---

Calcula la demanda promedio, mínima, máxima y su desviación estándar, agrupando por meses.

#### DEMANDA PROMEDIO TRIMESTRAL

---

```
SELECT [Demanda Trimestral].Trimestre AS Período, Avg([Demanda
    Trimestral].[SumaDeDemanda Puntual]) AS Promedio, Min([Demanda
    Trimestral].[SumaDeDemanda Puntual]) AS Mínima, Max([Demanda
    Trimestral].[SumaDeDemanda Puntual]) AS Máxima, StDev([Demanda
    Trimestral].[SumaDeDemanda Puntual]) AS DS
FROM [Demanda Trimestral]
GROUP BY [Demanda Trimestral].Trimestre;
```

---

Calcula la demanda promedio, mínima, máxima y su desviación estándar, agrupando por trimestres.

#### DEMANDA TRIMESTRAL

---

```
SELECT Year([Fecha Base]) AS Año, IIf(Month([Fecha
Base])<=3,1,IIf(Month([Fecha Base])<=6,2,IIf(Month([Fecha
Base])<=9,3,4))) AS Trimestre, Sum([Demanda por Fechas
Tabla].[Demanda Puntual]) AS [SumaDeDemanda Puntual]
FROM [Demanda por Fechas Tabla]
GROUP BY Year([Fecha Base]), IIf(Month([Fecha
Base])<=3,1,IIf(Month([Fecha Base])<=6,2,IIf(Month([Fecha
Base])<=9,3,4)));
```

---

Agrupando por año y dentro de éste creando y agrupando por trimestre, se totaliza la demanda, medida en número de unidades pedidas, previamente calculada en [Demanda por Fechas Tabla].

#### ESTADO DE PEDIDOS

---

```
SELECT DISTINCTROW [Productos Pedidos y Facturados 1].[N° de Pedido],
[Monto Global de Pedidos].[Monto Global del Pedido], [Pedidos de
Clientes].[Código del Cliente], Clientes.[Nombre o Razón Social],
[Pedidos de Clientes].Fecha, [Productos Pedidos y Facturados
1].[Factura N°] AS [N° de Factura], [Productos Pedidos y Facturados
1].[Fecha de Emisión], [Productos Pedidos y Facturados 1].[Fecha de
Despacho], [Productos Pedidos y Facturados 1].[Código del Producto],
[Productos Pedidos y Facturados 1].[Cantidad Pedida], [Productos
Pedidos y Facturados 1].[Cantidad Facturada Real] AS [Cantidad
Facturada], [Lista Sencilla de Productos Terminados].Nombre,
[Productos Pedidos y Facturados 1].[Precio Unitario de Venta],
[Productos Pedidos y Facturados 1].[% Descuento], [Productos Pedidos
y Facturados 1].[Precio con Descuento], [Productos Pedidos y
Facturados 1].[Monto del Producto]
FROM (Clientes INNER JOIN ([Lista Sencilla de Productos Terminados] INNER
JOIN ([Pedidos de Clientes] INNER JOIN [Productos Pedidos y
Facturados 1] ON [Pedidos de Clientes].[N° de Pedido] = [Productos
Pedidos y Facturados 1].[N° de Pedido]) ON [Lista Sencilla de
Productos Terminados].[Código del Producto] = [Productos Pedidos y
Facturados 1].[Código del Producto]) ON Clientes.[Código del
Cliente] = [Pedidos de Clientes].[Código del Cliente]) INNER JOIN
[Monto Global de Pedidos] ON [Pedidos de Clientes].[N° de Pedido] =
[Monto Global de Pedidos].[N° de Pedido];
```

---

Esta consulta consolida información proveniente de distintas tablas y consultas. Da un panorama general de cada pedido en cuanto a montos involucrados, cumplimiento del surtido y otros.

#### EXISTENCIA REAL EN ALMACÉN I

---

```
SELECT [Cantidad Recibida en Almacén I].[Código de Insumo],
       Insumos.Nombre, Insumos.[Tipo de Insumo], Insumos.[Unidades de
       Presentación], [Cantidad Recibida en Almacén I].[N° Lote], [Cantidad
       Recibida en Almacén I].[Cantidad Recibida], [Cantidad Recibida en
       Almacén I].Fecha, [Cantidad Despachada en Almacén I].[Orden de P
       N°], [Cantidad Despachada en Almacén I].[Cantidad Despachada],
       [Cantidad Despachada en Almacén I].[Fecha del Surtido],
       IIf(IsNull([Salidas]),0,[Salidas]) AS [Salidas Totales]
FROM (([Cantidad Recibida en Almacén I] LEFT JOIN [Cantidad Despachada en
Almacén I] ON ([Cantidad Recibida en Almacén I].[Código de Insumo] =
[Cantidad Despachada en Almacén I].[Código de Insumo]) AND
([Cantidad Recibida en Almacén I].[N° Lote] = [Cantidad Despachada
en Almacén I].[N° Lote])) LEFT JOIN Insumos ON [Cantidad Recibida en
Almacén I].[Código de Insumo] = Insumos.[Código de Insumo]) LEFT
JOIN [Salidas por Lote en Almacén I] ON [Cantidad Recibida en
Almacén I].[N° Lote] = [Salidas por Lote en Almacén I].[Lote N°];
```

---

Considerando un parámetro fecha heredado de consultas fuentes, se muestran las llegadas al almacén de insumos, consideradas por lote, para cada insumo presente en la base de datos, sin importar si ha o no recibido órdenes de compra (en caso negativo se suponen llegadas y lotes iguales a cero). Sirve de fuente de datos a la consulta siguiente.

#### EXISTENCIA REAL EN ALMACÉN I 1

---

```
SELECT [Existencia Real en Almacén I].[Código de Insumo], [Existencia
Real en Almacén I].Nombre, [Existencia Real en Almacén I].[Tipo de
Insumo], [Existencia Real en Almacén I].[Unidades de Presentación],
[Existencia Real en Almacén I].[N° Lote], [Existencia Real en
Almacén I].[Cantidad Recibida], [Existencia Real en Almacén
I].Fecha, [Existencia Real en Almacén I].[Salidas Totales],
Sum([Existencia Real en Almacén I].[Cantidad Despachada]) AS
[Cantidad Despachada Total]
FROM [Existencia Real en Almacén I]
GROUP BY [Existencia Real en Almacén I].[Código de Insumo], [Existencia
Real en Almacén I].Nombre, [Existencia Real en Almacén I].[Tipo de
Insumo], [Existencia Real en Almacén I].[Unidades de Presentación],
[Existencia Real en Almacén I].[N° Lote], [Existencia Real en
Almacén I].[Cantidad Recibida], [Existencia Real en Almacén
I].Fecha, [Existencia Real en Almacén I].[Salidas Totales];
```

---

Utilizando como fuente principal de datos la consulta anterior, se totalizan las cantidades despachadas para cada lote de insumo, agrupando apropiadamente los campos. Es una consulta intermedia que sirve de fuente de datos a la consulta siguiente.

### EXISTENCIA REAL EN ALMACÉN I 2

---

```

SELECT [Existencia Real en Almacén I 1].*, [Cantidad Recibida]-[Cantidad
  Despachada Total]-[Salidas Totales] AS [Existencia Real], [Insumos
  en Órdenes].[Precio Unitario]
FROM [Insumos en Órdenes] INNER JOIN ([Existencia Real en Almacén I 1]
  INNER JOIN [Llegadas de Insumos al Almacén] ON [Existencia Real en
  Almacén I 1].[Nº Lote] = [Llegadas de Insumos al Almacén].[Lote Nº])
  ON ([Insumos en Órdenes].Insumo = [Llegadas de Insumos al
  Almacén].Insumo) AND ([Insumos en Órdenes].[Orden Nº] = [Llegadas de
  Insumos al Almacén].[Orden Nº]);

```

---

Muestra la existencia real por lotes en el almacén de insumos, entendiendo por existencia real la totalización de las llegadas, restando despachos a producción y salidas por razones ajenas a la producción, para cada lote. También muestra el precio del insumo que regía para el momento indicado en la fecha dada (heredada como parámetro en las consultas anteriores).

### EXISTENCIA REAL EN ALMACÉN T

---

```

PARAMETERS [Forms]![Pedir Dos Fechas]![FechaÚnica] DateTime;
SELECT [Órdenes de Producción].Producto, [Órdenes de Producción].[Nº de
  Orden], [Lista Sencilla de Productos Terminados].Nombre, [Lista
  Sencilla de Productos Terminados].Departamento, Int([Producción
  Efectiva]/[Cantidad en Presentación]) AS [Cantidad Producida],
  [Cantidad Despachada en Almacén T].[Cantidad Despachada],
  IIf(IsNull([Cantidad Producida]),0,[Cantidad Producida])-
  IIf(IsNull([Cantidad Despachada]),0,[Cantidad Despachada]) AS
  [Existencia Real]
FROM [Productos Terminados] INNER JOIN ([Lista Sencilla de Productos
  Terminados] INNER JOIN ([Órdenes de Producción] LEFT JOIN [Cantidad
  Despachada en Almacén T] ON ([Órdenes de Producción].Producto =
  [Cantidad Despachada en Almacén T].[Código del Producto]) AND
  ([Órdenes de Producción].[Nº de Orden] = [Cantidad Despachada en
  Almacén T].[Nº de Orden])) ON [Lista Sencilla de Productos
  Terminados].[Código del Producto] = [Órdenes de
  Producción].Producto) ON [Productos Terminados].[Código del
  Producto] = [Órdenes de Producción].Producto
WHERE ((([Órdenes de Producción].[Fecha de Aprobación])<=[Forms]![Pedir
  Dos Fechas]![FechaÚnica]));

```

---



De manera similar a la mostrada en las tres consultas anteriores, aquí se calcula la existencia real de cada producto terminado, a la fecha dada, considerando únicamente producción (entradas) y ventas efectivas (salidas). En futuras versiones del sistema se planifica incorporar razones adicionales que afectan la existencia, como entradas de productos terminados por vías distintas a la producción (por ejemplo, reacondicionamiento y devoluciones) y salidas por vías distintas a las ventas (como destrucción u obsolescencia). No se consideraron en la presente versión otros tipos de salidas o entradas, ya que prácticamente no ocurren (la organización controla efectivamente esto, en palabras textuales, “aquí nada se pierde”, pues representan fallas graves y pérdidas), sin embargo, para mayor precisión, hay que dotar al sistema de esta posibilidad en el futuro inmediato, más que para precisar aún más los cálculos de las existencias en el almacén de productos terminados, para registrar y controlar salidas intermedias que podrían mejorar los procesos productivos.

#### EXISTENCIA TOTAL REAL AN ALMACÉN I

---

```

SELECT Insumos.[Código de Insumo], Insumos.Nombre, Insumos.[Tipo de
  Insumo], Insumos.[Unidades de Presentación], [Cantidad Total
  Ordenada en Almacén I].[Cantidad Total Ordenada], [Cantidad Total
  Recibida en Almacén I].[Cantidad Total Recibida], [Cantidad Total
  Despachada en Almacén I].[Cantidad Total Despachada], [Cantidad
  Total Desechada en Almacén I].[Cantidad Total Desechada],
  IIf(IsNull([Cantidad Total Recibida]),0,IIf(IsNull([Cantidad Total
  Despachada]),IIf(IsNull([Cantidad Total Desechada]),[Cantidad Total
  Recibida],[Cantidad Total Recibida]-[Cantidad Total
  Desechada]),IIf(IsNull([Cantidad Total Desechada]),[Cantidad Total
  Recibida]-[Cantidad Total Despachada],[Cantidad Total Recibida]-
  [Cantidad Total Despachada]-[Cantidad Total Desechada]))) AS
  [Existencia Total Real], IIf(IsNull([Cantidad Total
  Ordenada]),[Existencia Total Real],IIf(IsNull([Cantidad Total
  Recibida]),[Cantidad Total Ordenada],[Cantidad Total Ordenada]-
  [Cantidad Total Recibida]+[Existencia Total Real])) AS [Existencia
  Total Proyectada], [Últimos Precios de Insumos Consulta].[Precio
  Unitario]
FROM (((((Insumos LEFT JOIN [Cantidad Total Despachada en Almacén I] ON
  Insumos.[Código de Insumo] = [Cantidad Total Despachada en Almacén
  I].[Código de Insumo]) LEFT JOIN [Cantidad Total Ordenada en Almacén
  I] ON Insumos.[Código de Insumo] = [Cantidad Total Ordenada en
  Almacén I].[Código de Insumo]) LEFT JOIN [Cantidad Total Recibida en
  Almacén I] ON Insumos.[Código de Insumo] = [Cantidad Total Recibida
  en Almacén I].[Código de Insumo]) LEFT JOIN [Últimos Precios de
  Insumos Consulta] ON Insumos.[Código de Insumo] = [Últimos Precios
  de Insumos Consulta].Insumo) LEFT JOIN [Cantidad Total Desechada en
  Almacén I] ON Insumos.[Código de Insumo] = [Cantidad Total Desechada
  en Almacén I].Insumo;

```

---

Discriminando ahora por insumos (no por lotes, como antes) se calcula la existencia real y la proyectada, entendiendo por existencia real lo que ha llegado menos lo que ha sido despachado a producción, menos lo que ha salido por razones distintas a la producción, y por existencia proyectada lo que ha sido ordenado (que puede no haber llegado aún) menos lo despachado, menos las salidas. Todo esto verificando que las fechas claves de cada actividad, sean inferiores o iguales a una dada (heredada como parámetro).

#### EXISTENCIA TOTAL REAL EN ALMACÉN T

---

```

SELECT [Productos Terminados].[Código del Producto], [Cantidad Total
  Producida en Almacén T].[Total Producido], [Cantidad Total Pedida en
  Almacén T].[SumaDeCantidad Pedida] AS [Total Pedido], [Cantidad
  Total Facturada en Almacén T].[SumaDeCantidad Facturada] AS [Total
  Facturado], [Cantidad Total Despachada en Almacén T].[Total
  Despachado], IIf(IsNull([Total Producido]),0,[Total Producido])-
  IIf(IsNull([Total Despachado]),0,[Total Despachado]) AS [Existencia
  Real Total], IIf(IsNull([Total Producido]),0,[Total Producido])-
  IIf(IsNull([Total Despachado]),0,[Total Despachado])-
  IIf(IsNull([Total Pedido]),0,[Total Pedido]) AS [Existencia
  Proyectada Total], [Últimos Precios de Productos Consulta].[Precio
  Unitario de Venta]
FROM (((([Productos Terminados] LEFT JOIN [Cantidad Total Despachada en
  Almacén T] ON [Productos Terminados].[Código del Producto] =
  [Cantidad Total Despachada en Almacén T].[Código del Producto]) LEFT
  JOIN [Cantidad Total Producida en Almacén T] ON [Productos
  Terminados].[Código del Producto] = [Cantidad Total Producida en
  Almacén T].[Código del Producto]) LEFT JOIN [Cantidad Total
  Facturada en Almacén T] ON [Productos Terminados].[Código del
  Producto] = [Cantidad Total Facturada en Almacén T].[Código del
  Producto]) LEFT JOIN [Cantidad Total Pedida en Almacén T] ON
  [Productos Terminados].[Código del Producto] = [Cantidad Total
  Pedida en Almacén T].[Código del Producto]) LEFT JOIN [Últimos
  Precios de Productos Consulta] ON [Productos Terminados].[Código del
  Producto] = [Últimos Precios de Productos Consulta].[Código del
  Producto]);

```

---

De la misma forma que en la consulta anterior, se calculan las existencias reales y proyectadas, ahora para cada producto terminado y dada una fecha máxima a considerar. En este caso, la existencia real es lo producido menos lo despachado, y la existencia proyectada es lo producido menos lo pedido (que puede no haber sido despachado todavía), menos los despachado.

**EXPLOSIÓN DE MATERIALES**


---

```

SELECT [Planes de Producción].[N° del Plan], [Productos en
Planes].[Código del Producto], [Productos en Planes].Fecha,
[Productos Terminados e Insumos].[Código de Insumo], [Productos
Terminados e Insumos].[Cantidad para el Lote], [Cantidad]*[Cantidad
para el Lote]/[Tamaño del Lote] AS [Cantidad Para Producir]
FROM [Planes de Producción] INNER JOIN (([Productos en Planes] INNER JOIN
[Productos Terminados e Insumos] ON [Productos en Planes].[Código
del Producto] = [Productos Terminados e Insumos].[Código del
Producto]) INNER JOIN [Lista Sencilla de Productos Terminados] ON
[Productos en Planes].[Código del Producto] = [Lista Sencilla de
Productos Terminados].[Código del Producto]) ON [Planes de
Producción].[N° del Plan] = [Productos en Planes].[N° del Plan];

```

---

Para cada producto dentro de un plan de producción, se incluyen los insumos y las cantidades necesarias para producirlo, tomadas de las fórmulas. Es una consulta preliminar que sirve de fuente de datos a la próxima.

**EXPLOSIÓN DE MATERIALES 1**


---

```

SELECT [Explosión de Materiales].[N° del Plan], [Explosión de
Materiales].[Código de Insumo], Sum([Explosión de
Materiales].[Cantidad Para Producir]) AS [Cantidad Total Para
Producir]
FROM [Explosión de Materiales]
GROUP BY [Explosión de Materiales].[N° del Plan], [Explosión de
Materiales].[Código de Insumo];

```

---

Agrupando por planes y, dentro de éstos, por insumo, se totalizan (se consolidan los insumos necesarios para todos los productos) las cantidades necesarias para la producción del plan. Es una consulta intermedia que servirá para la próxima.

**EXPLOSIÓN DE MATERIALES 2**


---

```

PARAMETERS Forms![Pedir Número del Plan]![Exceso] IEEESingle,
Formularios![Pedir Número del Plan]![Plan] Long;
SELECT [Explosión de Materiales 1].[N° del Plan], [Planes de
Producción].Descripción, [Planes de Producción].[Fecha de Inicio],
[Planes de Producción].[Fecha de Culminación], [Explosión de
Materiales 1].[Código de Insumo], Insumos.Nombre, Insumos.[Tipo de
Insumo], Insumos.[Presentación Mínima], Insumos.[Unidades de
Presentación], [Explosión de Materiales 1].[Cantidad Total Para
Producir] AS [Cantidad a Utilizar], [Forms]![Pedir Número del

```

---

```

Plan]![Exceso]*[Cantidad a Utilizar] AS [Exceso en Unidades],
[Cantidad a Utilizar]+[Exceso en Unidades] AS [Total Requerido en
Unidades], IIf(IsNull([Existencia Total Real]),0,[Existencia Total
Real]) AS [Existencias a la Fecha en Unidades], [Últimos Precios de
Insumos Consulta].[Precio Unitario], [Total Requerido en
Unidades]*[Últimos Precios de Insumos Consulta]![Precio Unitario] AS
[Total Requerido en Bolívares], [Existencias a la Fecha en
Unidades]*[Últimos Precios de Insumos Consulta]![Precio Unitario] AS
[Existencias a la Fecha en Bolívares], [Total Requerido en
Unidades]-[Existencias a la Fecha en Unidades] AS [Necesidades
Adicionales en Unidades], IIf([Total Requerido en Bolívares]-
[Existencias a la Fecha en Bolívares]<0,0,[Total Requerido en
Bolívares]-[Existencias a la Fecha en Bolívares]) AS [Necesidades
Adicionales en Bolívares]
FROM ((([Explosión de Materiales 1] LEFT JOIN Insumos ON [Explosión de
Materiales 1].[Código de Insumo] = Insumos.[Código de Insumo]) LEFT
JOIN [Últimos Precios de Insumos Consulta] ON [Explosión de
Materiales 1].[Código de Insumo] = [Últimos Precios de Insumos
Consulta].Insumo) LEFT JOIN [Existencia Total Real an Almacén I] ON
[Explosión de Materiales 1].[Código de Insumo] = [Existencia Total
Real an Almacén I].[Código de Insumo]) INNER JOIN [Planes de
Producción] ON [Explosión de Materiales 1].[N° del Plan] = [Planes
de Producción].[N° del Plan]
WHERE ((([Explosión de Materiales 1].[N° del Plan])=[Formularios]![Pedir
Número del Plan]![Plan]));

```

Dado un porcentaje de exceso (estimación de necesidades adicionales para protección del proceso productivo), se calculan y presentan las existencias a la fecha dada (parámetro heredado), las necesidades para cada plan de cada insumo (tomadas de la consulta anterior) y los montos requeridos en bolívares para completar el plan, considerando que si la existencia menos lo requerido resulta positivo, no se necesitan recursos adicionales, y en caso contrario se estiman los recursos adicionales tomando como base el último precio registrado en la base de datos.

#### FACTURAS TOTALES DESPACHADAS

```

SELECT Facturas.[N° de Factura], [Monto Global de Facturas].[Monto Global
de la Factura] AS [Monto Total Surtido]
FROM Facturas INNER JOIN [Monto Global de Facturas] ON Facturas.[N° de
Factura] = [Monto Global de Facturas].[Factura N°]
WHERE (((Facturas.[Fecha de Despacho]) Is Not Null))
ORDER BY [Monto Global de Facturas].[Monto Global de la Factura] DESC;

```

Se muestran los montos (dinero) involucrados en cada factura, de forma decreciente, con la finalidad de que el usuario visualice más fácilmente los compromisos más grandes.

**FACTURAS TOTALES ENTRE FECHAS**


---

```

PARAMETERS [Forms]![Pedir Fechas para Ventas]![FechaInicial] DateTime,
           [Forms]![Pedir Fechas para Ventas]![FechaFinal] DateTime;
SELECT Facturas.[N° de Factura], [Monto Global de Facturas].[Monto Global
de la Factura] AS [Monto Total Surtido]
FROM Facturas INNER JOIN [Monto Global de Facturas] ON Facturas.[N° de
Factura] = [Monto Global de Facturas].[Factura N°]
WHERE (((Facturas.[Fecha de Despacho]) Between [Forms]![Pedir Fechas para
Ventas]![FechaInicial] And [Forms]![Pedir Fechas para
Ventas]![FechaFinal])) OR ((([Facturas].[Fecha de
Despacho])=[Forms]![Pedir Fechas para Ventas]![FechaInicial] Or
([Facturas].[Fecha de Despacho])=[Forms]![Pedir Fechas para
Ventas]![FechaFinal]) Is Null))
ORDER BY [Monto Global de Facturas].[Monto Global de la Factura] DESC;

```

---

De manera similar a la anterior, se muestran los montos involucrados en cada factura, sin embargo, a diferencia de aquella, el usuario puede restringir las fechas entre las cuales se listará la información.

**FACTURAS Y PEDIDOS CONSULTAS**


---

```

SELECT DISTINCTROW Facturas.[N° de Factura], Facturas.[Fecha de Emisión],
           Facturas.[Elaborada Por], [Pedidos de Clientes].*, Facturas.[Fecha
de Despacho]
FROM [Pedidos de Clientes] INNER JOIN Facturas ON [Pedidos de
Clientes].[N° de Pedido] = Facturas.[N° de Pedido];

```

---

Consolida la información de pedidos y facturas para que sea utilizada en el análisis comparativo de lo comprometido con lo cumplido.

**FÓRMULAS CONSULTA**


---

```

SELECT DISTINCTROW Fórmulas.[N° de Fórmula], Fórmulas.Producto,
           Fórmulas.Vehículo, Fórmulas.[Unidades de Presentación],
           Fórmulas.Concentración, Fórmulas.[Unidades de Concentración],
           Fórmulas.Departamento, Fórmulas.[Fecha de Emisión],
           Fórmulas.Validez, Fórmulas.[Unidades de Validez], Fórmulas.[Registro
Sanitario], Fórmulas.[Técnica de Elaboración], Fórmulas.[Tamaño del
Lote], Fórmulas.[Elaborado por], Personas.Apellidos,
           Personas.Nombres
FROM Personas RIGHT JOIN Fórmulas ON Personas.[Cédula de Identidad o
Clave] = Fórmulas.[Elaborado por]
ORDER BY Fórmulas.Producto;

```

---

Ordenando por productos se consolida alguna información de las fórmulas y quién las elabora. Esta consulta se utiliza en subformularios, subinformes (elementos dentro de formularios e informes) y otras consultas de más alto nivel.

#### FÓRMULAS Y PRODUCTOS TERMINADOS

---

```
SELECT DISTINCTROW Fórmulas.[N° de Fórmula], [Productos
Terminados].Presentación, [Productos Terminados].[Cantidad en
Presentación], [Productos Terminados].Destino
FROM Fórmulas INNER JOIN [Productos Terminados] ON Fórmulas.[N° de
Fórmula] = [Productos Terminados].[N° de Fórmula]
GROUP BY Fórmulas.[N° de Fórmula], [Productos Terminados].Presentación,
[Productos Terminados].[Cantidad en Presentación], [Productos
Terminados].Destino;
```

---

Consolida información de fórmulas y productos terminados considerando que la presentación, la cantidad en presentación y el destino del producto, pueden variar para una misma fórmula (razón del agrupamiento sin funciones de dominio agregado).

#### GRÁFICO DE PRODUCTOS ESTRELLA

---

```
SELECT [Productos Pedidos y Despachados entre Fechas].Nombre AS Producto,
Sum([Productos Pedidos y Despachados entre Fechas].[Total Ventas])
AS [Total Ventas]
FROM [Productos Pedidos y Despachados entre Fechas]
GROUP BY [Productos Pedidos y Despachados entre Fechas].Nombre
ORDER BY Sum([Productos Pedidos y Despachados entre Fechas].[Total
Ventas]) DESC;
```

---

Esta consulta opera en conjunto con la siguiente para la elaboración de un gráfico porcentual importante denominado “de productos estrella” (los mejores). Este gráfico requiere una partición del conjunto de los registros de ventas, en un número previamente especificado y el agrupamiento de los restantes. Para los productos escogidos, totaliza las ventas (entendidas como pedidos) entre fechas (parámetros heredados), organizados del “mejor” al “peor” producto, en términos de ventas.

**GRÁFICO DE PRODUCTOS ESTRELLA 1**


---

```
SELECT "Otros" AS Producto, Sum([Gráfico de Productos Estrella T2].[Total
Ventas]) AS [Total Ventas]
FROM [Gráfico de Productos Estrella T2];
```

---

En combinación con la consulta anterior, ahora se totalizan las ventas del resto de los productos excluidos antes, bajo el rótulo “Otros”.

**GRÁFICO DE PRODUCTOS ESTRELLA 2**


---

```
SELECT *
FROM [Gráfico de Productos Estrella T1]
UNION SELECT * FROM [Gráfico de Productos Estrella 1];
```

---

Utilizando las dos consultas anteriores se realiza la unión de ambos conjuntos de registros (ejemplo de consultas SQL anidadas) para conformar un solo conjunto que contenga lo relativo a los productos escogidos por el usuario y los restantes. Esta es la consulta final utilizada en el gráfico mencionado.

**INSUMOS EN ÓRDENES CONSULTA**


---

```
SELECT DISTINCTROW [Insumos en Órdenes].[Orden N°], [Órdenes de
Compra].Fecha, [Insumos en Órdenes].Insumo, Insumos.Nombre, [Insumos
en Órdenes].[Precio Unitario], [Insumos en Órdenes].Cantidad,
Insumos.[Unidades de Presentación], Insumos.[Condiciones de Entrega]
FROM [Órdenes de Compra] INNER JOIN (Insumos INNER JOIN [Insumos en
Órdenes] ON Insumos.[Código de Insumo] = [Insumos en
Órdenes].Insumo) ON [Órdenes de Compra].[N°] = [Insumos en
Órdenes].[Orden N°];
```

---

Consolida la información principal proveniente de las órdenes de compra y los insumos involucrados en cada orden (debe recordarse que por razones de diseño y normalización, estos datos se encuentran físicamente en tablas separadas).

**INSUMOS ORDENADOS Y RECIBIDOS**


---

```
SELECT [Llegadas de Insumos al Almacén].*, [Insumos en Órdenes].[Precio
Unitario]
FROM [Insumos en Órdenes] INNER JOIN [Llegadas de Insumos al Almacén] ON
```

---

---

```
([Insumos en Órdenes].Insumo = [Llegadas de Insumos al
Almacén].Insumo) AND ([Insumos en Órdenes].[Orden N°] = [Llegadas de
Insumos al Almacén].[Orden N°]);
```

---

Incluye, además de la información de cada insumo que arriba al almacén, el precio pautado por él en la orden de compra.

#### LISTA DE PRECIOS VIGENTE EN FECHA BUSCADA

---

```
SELECT DISTINCTROW [Precios de Productos].[Código del Producto], [Precios
de Productos].Fecha, [Lista Sencilla de Productos
Terminados].Nombre, [Lista Sencilla de Productos
Terminados].Departamento, [Precios de Productos].[Precio Unitario de
Venta], [Productos Terminados y Escalas de Precios].Escala, [Escalas
de Precios].[Límite Inferior], [Escalas de Precios].[Límite
Superior], [Escalas de Precios].[% Descuento], [Precio Unitario de
Venta]-[Precio Unitario de Venta]*[% Descuento] AS [Precio con
Descuento]
FROM [Escalas de Precios] INNER JOIN ((([Precios de Productos HASTA Fecha
Buscada1] INNER JOIN [Precios de Productos] ON ([Precios de
Productos HASTA Fecha Buscada1].MáxDeFecha = [Precios de
Productos].Fecha) AND ([Precios de Productos HASTA Fecha
Buscada1].[Código del Producto] = [Precios de Productos].[Código del
Producto])) INNER JOIN [Lista Sencilla de Productos Terminados] ON
[Precios de Productos].[Código del Producto] = [Lista Sencilla de
Productos Terminados].[Código del Producto]) INNER JOIN [Productos
Terminados y Escalas de Precios] ON [Lista Sencilla de Productos
Terminados].[Código del Producto] = [Productos Terminados y Escalas
de Precios].[Código del Producto]) ON [Escalas de Precios].Escala =
[Productos Terminados y Escalas de Precios].Escala;
```

---

Aplicando la política de descuentos por cantidad y considerando la lista de precios de cada producto, lista los distintos precios aplicables a los productos, vigentes en la fecha especificada por el usuario. Esta consulta es útil, pues permite conocer no sólo el precio actual de algún producto, sino también el precio en cualquier fecha pasada e incluso futura.

#### LISTA SENCILLA DE CLIENTES

---

```
SELECT DISTINCTROW Clientes.[Código del Cliente], Clientes.[Nombre o
Razón Social], Clientes.[Tipo de Cliente], Clientes.Grupo,
Clientes.Sector
FROM Clientes
ORDER BY Clientes.[Nombre o Razón Social];
```

---



Se toman los datos resumidos de cada cliente y se muestran por orden alfabético del nombre.

#### LISTA SENCILLA DE PRODUCTOS TERMINADOS

---

```
SELECT DISTINCTROW [Productos Terminados].[Código del Producto],
  [Fórmulas]![Producto] & " " & [Fórmulas]![Vehículo] & " " &
  [Fórmulas]![Concentración] & " " & [Fórmulas]![Unidades de
  Concentración] & " x " & [Productos Terminados]![Cantidad en
  Presentación] & " " & [Fórmulas]![Unidades de Presentación] AS
  Nombre, Fórmulas.Departamento, Fórmulas.[Tamaño del Lote],
  [Productos Terminados].[Punto de Reorden]
FROM Fórmulas INNER JOIN [Productos Terminados] ON Fórmulas.[N° de
  Fórmula] = [Productos Terminados].[N° de Fórmula];
```

---

La estructura del nombre de un producto terminado se relaciona con diversos campos combinados de las tablas [Fórmulas] y [Productos Terminados]. Esto obedece a la previsión de futuras necesidades de información, agrupada por los criterios lógicos que componen un producto farmacéutico (por ejemplo, presentaciones o vehículos). Se construye entonces el nombre, concatenando las diversas cadenas de texto pertinentes, con la finalidad de manejarse dentro del sistema con una sola cadena de caracteres.

#### LISTA SENCILLA DE PROVEEDORES

---

```
SELECT DISTINCTROW Proveedores.[Código del Proveedor],
  Proveedores.[Nombre o Razón Social]
FROM Proveedores
ORDER BY Proveedores.[Nombre o Razón Social];
```

---

Con la misma idea de las dos consultas anteriores, aquí se muestra sólo la información principal de cada proveedor (omitiendo u ocultando los detalles).

#### MONTO GLOBAL DE FACTURAS

---

```
SELECT [Productos Pedidos y Facturados 1].[Factura N°], Sum([Productos
  Pedidos y Facturados 1].[Monto Facturado]) AS [Monto Global de la
  Factura]
FROM [Productos Pedidos y Facturados 1]
```

---

---

```
WHERE ((([Productos Pedidos y Facturados 1].[Factura N°])<>0))
GROUP BY [Productos Pedidos y Facturados 1].[Factura N°]
ORDER BY Sum([Productos Pedidos y Facturados 1].[Monto Facturado]) DESC;
```

---

Agrupando por facturas se totalizan los montos en dinero, involucrados para cada una de ellas, ordenados del mayor al menor.

#### **MONTO GLOBAL DE PEDIDOS**

---

```
SELECT [Productos Pedidos y Precios 3].[N° de Pedido], Sum([Productos
Pedidos y Precios 3].[Monto del Producto]) AS [Monto Global del
Pedido]
FROM [Productos Pedidos y Precios 3]
GROUP BY [Productos Pedidos y Precios 3].[N° de Pedido];
```

---

El mismo concepto anterior, ahora aplicado a los pedidos en lugar de las facturas.

#### **ÓRDENES DE PRODUCCIÓN CONSULTA**

---

```
SELECT DISTINCTROW [Órdenes de Producción].[N° de Orden], [Órdenes de
Producción].[Fecha de Emisión], [Productos Terminados].Presentación,
[Productos Terminados].[Cantidad en Presentación], [Productos
Terminados].Destino, Fórmulas.Producto, Fórmulas.Vehículo,
Fórmulas.[Unidades de Presentación], Fórmulas.Concentración,
Fórmulas.[Unidades de Concentración]
FROM (Fórmulas INNER JOIN [Productos Terminados] ON Fórmulas.[N° de
Fórmula] = [Productos Terminados].[N° de Fórmula]) INNER JOIN
[Órdenes de Producción] ON [Productos Terminados].[Código del
Producto] = [Órdenes de Producción].Producto;
```

---

Consolida información de las órdenes de producción, fórmulas y productos terminados.

#### **PEDIDOS TOTALES ENTRE FECHAS**

---

```
PARAMETERS [Forms]![Pedir Fechas para Ventas]![FechaInicial] DateTime,
[Forms]![Pedir Fechas para Ventas]![FechaFinal] DateTime;
SELECT [Pedidos de Clientes].[N° de Pedido], [Monto Global de
Pedidos].[Monto Global del Pedido] AS [Monto Total Pedido]
FROM [Pedidos de Clientes] INNER JOIN [Monto Global de Pedidos] ON
[Pedidos de Clientes].[N° de Pedido] = [Monto Global de Pedidos].[N°
de Pedido]
WHERE ((([Pedidos de Clientes].Fecha) Between [Forms]![Pedir Fechas para
```

```

Ventas][FechaInicial] And [Forms][Pedir Fechas para
Ventas][FechaFinal])) OR ((([Pedidos de
Clientes].[Fecha]=[Forms][Pedir Fechas para Ventas][FechaInicial]
Or ([Pedidos de Clientes].[Fecha]=[Forms][Pedir Fechas para
Ventas][FechaFinal]) Is Null))
ORDER BY [Monto Global de Pedidos].[Monto Global del Pedido] DESC;

```

Muestra los montos de pedidos considerando sólo aquellos que están entre dos fechas dadas.

### PLANES DE PRODUCCIÓN GLOBALES

```

SELECT [Planes de Producción].[N° del Plan], [Productos en
Planes].[Código del Producto], Sum([Productos en Planes].Cantidad)
AS [Total Unidades]
FROM [Planes de Producción] INNER JOIN [Productos en Planes] ON [Planes
de Producción].[N° del Plan] = [Productos en Planes].[N° del Plan]
GROUP BY [Planes de Producción].[N° del Plan], [Productos en
Planes].[Código del Producto];

```

Agrupando los planes de producción y dentro de éstos cada producto, se totalizan las cantidades planificadas para ser producidas en cada caso. Hay que recordar que los planes de producción incluyen, para cada plan y producto, fechas distintas de producción.

### PLANES DE PRODUCCIÓN GLOBALES 1

```

PARAMETERS Formularios![Pedir Número del Plan]![Plan] Long;
SELECT [Lista Sencilla de Productos Terminados].Departamento, [Planes de
Producción Globales].[N° del Plan], [Planes de
Producción].Descripción, [Planes de Producción].[Fecha de Inicio],
[Planes de Producción].[Fecha de Culminación], [Planes de Producción
Globales].[Código del Producto], [Lista Sencilla de Productos
Terminados].Nombre, [Planes de Producción Globales].[Total
Unidades], [Últimos Precios de Productos Consulta].[Precio Unitario
de Venta] AS [Precio Estimado], [Total Unidades]*[Precio Unitario de
Venta] AS [Total Producción]
FROM (([Planes de Producción Globales] INNER JOIN [Lista Sencilla de
Productos Terminados] ON [Planes de Producción Globales].[Código del
Producto] = [Lista Sencilla de Productos Terminados].[Código del
Producto]) INNER JOIN [Últimos Precios de Productos Consulta] ON
[Planes de Producción Globales].[Código del Producto] = [Últimos
Precios de Productos Consulta].[Código del Producto]) INNER JOIN
[Planes de Producción] ON [Planes de Producción Globales].[N° del
Plan] = [Planes de Producción].[N° del Plan]
WHERE ((([Planes de Producción Globales].[N° del
Plan])=[Formularios]![Pedir Número del Plan]![Plan]));

```

Para un plan específico y utilizando como fuente de datos, entre otras, la consulta anterior, se consolida la información del plan incluyendo precios estimados y total de unidades involucradas.

#### PLANES DE PRODUCCIÓN MENSUALES

---

```

SELECT [Planes de Producción].[N° del Plan], [Lista Sencilla de Productos
Terminados].Departamento, [Productos en Planes].[Código del
Producto], [Lista Sencilla de Productos Terminados].Nombre,
[Productos en Planes].Fecha, [Productos en Planes].Cantidad AS
Unidades, IIf(Month([Fecha])<10,"0" & LTrim(Str(Month([Fecha]))) &
"):","(" & LTrim(Str(Month([Fecha]))) & ")") &
DLookUp("Mes","[Meses del Año]","Código=" & Str(Month([Fecha])) &
"-" & Right(Str(Year([Fecha])),2) AS [Año-Mes]
FROM ([Planes de Producción] INNER JOIN [Productos en Planes] ON [Planes
de Producción].[N° del Plan] = [Productos en Planes].[N° del Plan])
INNER JOIN [Lista Sencilla de Productos Terminados] ON [Productos en
Planes].[Código del Producto] = [Lista Sencilla de Productos
Terminados].[Código del Producto]
ORDER BY IIf(Month([Fecha])<10,"0" & LTrim(Str(Month([Fecha]))) &
"):","(" & LTrim(Str(Month([Fecha]))) & ")") &
DLookUp("Mes","[Meses del Año]","Código=" & Str(Month([Fecha])) &
"-" & Right(Str(Year([Fecha])),2);

```

---

Considerando cada fecha prevista en el plan de producción se construye un campo especial que extrae de ésta el año y el mes correspondiente, como primer paso para la obtención de totales mensuales.

#### PLANES DE PRODUCCIÓN MENSUALES RC

---

```

PARAMETERS [Formularios]![Pedir Número del Plan]![Plan] Long;
TRANSFORM Sum([Planes de Producción Mensuales].Unidades) AS [El Valor]
SELECT [Planes de Producción Mensuales].[N° del Plan], [Planes de
Producción Mensuales].Departamento, [Planes de Producción
Mensuales].[Código del Producto], [Planes de Producción
Mensuales].Nombre, Sum([Planes de Producción Mensuales].Unidades) AS
[Total de Unidades]
FROM [Planes de Producción Mensuales]
WHERE ((([Planes de Producción Mensuales].[N° del
Plan])=[Formularios]![Pedir Número del Plan]![Plan]))
GROUP BY [Planes de Producción Mensuales].[N° del Plan], [Planes de
Producción Mensuales].Departamento, [Planes de Producción
Mensuales].[Código del Producto], [Planes de Producción
Mensuales].Nombre
PIVOT [Planes de Producción Mensuales].[Año-Mes];

```

---

Esta es una consulta de referencias cruzadas. Una referencia cruzada es la transposición de una tabla donde los valores de algunos campos se transforman en nombres de campos, algunos nombres de campos en valores de registros, y como valores a determinar se totalizan en la encrucijada de filas y columnas algunos otros valores numéricos. Este tipo de consultas es muy útil para la presentación de resultados en forma gráfica o consolidada. En este caso, con base en la consulta anterior, se totalizan los requerimientos de cada plan de producción por plan y producto, sumando lo que esté planificado producirse en cada mes de cada año involucrado (los campos que ahora pasan a ser nombres de campos). Con esta consulta se elaboran informes de planes de producción.

#### PRECIOS DE INSUMOS

---

```
SELECT [Insumos en Órdenes].Insumo, [Insumos en Órdenes].[Precio
    Unitario], [Órdenes de Compra].Fecha
FROM [Órdenes de Compra] INNER JOIN [Insumos en Órdenes] ON [Órdenes de
    Compra].[N°] = [Insumos en Órdenes].[Orden N°]
ORDER BY [Insumos en Órdenes].Insumo, [Órdenes de Compra].Fecha;
```

---

Tiene el objetivo de producir la serie de precios de insumos tomando información de la fecha de las órdenes de compra y los insumos involucrados.

#### PRECIOS DE PRODUCTOS HASTA FECHA BUSCADA

---

```
PARAMETERS [Forms]![Pedir Dos Fechas]![FechaÚnica] DateTime;
SELECT DISTINCTROW [Precios de Productos].[Código del Producto], [Precios
    de Productos].Fecha, [Precios de Productos].[Precio Unitario de
    Venta]
FROM [Precios de Productos]
WHERE ((([Precios de Productos].Fecha)<=[Forms]![Pedir Dos
    Fechas]![FechaÚnica]));
```

---

Se muestran todos los precios de los productos terminados, fijados con anterioridad o durante la fecha especificada. Opera en conjunto con la consulta siguiente.

#### PRECIOS DE PRODUCTOS HASTA FECHA BUSCADA1

---

```
SELECT DISTINCTROW [Precios de Productos HASTA Fecha Buscada].[Código del
```

---

---

```

    Producto], Max([Precios de Productos HASTA Fecha Buscada].Fecha) AS
    MáxDeFecha
FROM [Precios de Productos HASTA Fecha Buscada]
GROUP BY [Precios de Productos HASTA Fecha Buscada].[Código del
Producto];

```

---

Agrupando por producto se encuentra el precio del producto terminado, vigente en la oportunidad próxima pasada más cercana a la fecha suministrada (que es un parámetro heredado de la consulta anterior). Es decir, se construye la lista de precios de los productos terminados, vigente en una fecha cualquiera dada.

#### PRODUCTOS DESPACHADOS ENTRE FECHAS

---

```

SELECT [Productos Pedidos y Facturados 1].[Código del Producto],
    Sum([Productos Pedidos y Facturados 1].[Cantidad Facturada Real]) AS
    [Total Unidades], Sum([Productos Pedidos y Facturados 1].[Monto
    Facturado]) AS [Total Bs]
FROM [Productos Pedidos y Facturados 1] INNER JOIN [Pedidos de Clientes]
    ON [Productos Pedidos y Facturados 1].[N° de Pedido] = [Pedidos de
    Clientes].[N° de Pedido]
WHERE ((([Pedidos de Clientes].Fecha) Between [Forms]![Pedir Fechas para
    Ventas por Productos]![FechaInicial] And [Forms]![Pedir Fechas para
    Ventas por Productos]![FechaFinal]) AND (([Productos Pedidos y
    Facturados 1].[Fecha de Despacho] Is Not Null)) OR ((([Pedidos de
    Clientes].[Fecha])=[Forms]![Pedir Fechas para Ventas por
    Productos]![FechaInicial] Or ([Pedidos de
    Clientes].[Fecha])=[Forms]![Pedir Fechas para Ventas por
    Productos]![FechaFinal]) Is Null))
GROUP BY [Productos Pedidos y Facturados 1].[Código del Producto]
ORDER BY Sum([Productos Pedidos y Facturados 1].[Monto Facturado]) DESC;

```

---

Agrupando los productos se totalizan las unidades y montos, involucrados en las facturas, de forma que se consideran sólo aquellos registros cuya fecha de despacho al cliente está entre dos parámetros dados (heredados de consultas fuente).

#### PRODUCTOS PEDIDOS ENTRE FECHAS

---

```

PARAMETERS [Forms]![Pedir Fechas para Ventas por
    Productos]![FechaInicial] DateTime, [Forms]![Pedir Fechas para
    Ventas por Productos]![FechaFinal] DateTime;
SELECT [Productos Pedidos y Precios 3].[Código del Producto],
    Sum([Productos Pedidos y Precios 3].[Cantidad Pedida]) AS [Total
    Unidades], Sum([Productos Pedidos y Precios 3].[Monto del Producto])

```

---

```

AS [Total Bs]
FROM [Pedidos de Clientes] INNER JOIN [Productos Pedidos y Precios 3] ON
  [Pedidos de Clientes].[N° de Pedido] = [Productos Pedidos y Precios
  3].[N° de Pedido]
WHERE ((([Pedidos de Clientes].Fecha) Between [Forms]![Pedir Fechas para
  Ventas por Productos]![FechaInicial] And [Forms]![Pedir Fechas para
  Ventas por Productos]![FechaFinal])) OR ((([Pedidos de
  Clientes].[Fecha])=[Forms]![Pedir Fechas para Ventas por
  Productos]![FechaInicial] Or ([Pedidos de
  Clientes].[Fecha])=[Forms]![Pedir Fechas para Ventas por
  Productos]![FechaFinal]) Is Null))
GROUP BY [Productos Pedidos y Precios 3].[Código del Producto]
ORDER BY Sum([Productos Pedidos y Precios 3].[Monto del Producto]) DESC;

```

De forma similar a la descrita antes se totalizan por pedido las unidades y montos de productos involucrados.

#### PRODUCTOS PEDIDOS Y DESPACHADOS ENTRE FECHAS

```

SELECT [Lista Sencilla de Productos Terminados].Nombre, [Lista Sencilla
  de Productos Terminados].Departamento, [Productos Pedidos entre
  Fechas].[Código del Producto], [Productos Pedidos entre
  Fechas].[Total Unidades] AS [Unidades Vendidas], [Productos Pedidos
  entre Fechas].[Total Bs] AS [Total Ventas], [Productos Despachados
  entre Fechas].[Total Unidades] AS [Unidades Despachadas], [Productos
  Despachados entre Fechas].[Total Bs] AS [Total Despachado],
  [Unidades Vendidas]-[Unidades Despachadas] AS [Unidades
  Insatisfechas], [Total Ventas]-[Total Despachado] AS [Total
  Insatisfecho]
FROM [Lista Sencilla de Productos Terminados] INNER JOIN ([Productos
  Despachados entre Fechas] INNER JOIN [Productos Pedidos entre
  Fechas] ON [Productos Despachados entre Fechas].[Código del
  Producto] = [Productos Pedidos entre Fechas].[Código del Producto])
  ON [Lista Sencilla de Productos Terminados].[Código del Producto] =
  [Productos Pedidos entre Fechas].[Código del Producto]
ORDER BY [Productos Pedidos entre Fechas].[Total Bs] DESC;

```

Utilizando como base las dos consultas anteriores se muestra como está la relación entre lo pedido por cada cliente y lo efectivamente despachado; se agrega un campo que calcula el número de unidades insatisfechas en esta relación. Esta consulta es de importancia capital, pues permite asignar prioridades e incluso estudiar la relación entre la labor de ventas y la labor de producción o despacho, lo que a su vez da la oportunidad de mejorar el servicio a los clientes.

Utilizando dos fechas como parámetros heredados, se puede obtener la información restringida a los lapsos establecidos por las fechas, y omitiéndolas se obtiene la información general.

### PRODUCTOS PEDIDOS Y FACTURADOS

---

```

SELECT DISTINCTROW [Pedidos de Clientes].[N° de Pedido], Facturas.[N° de
  Factura], [Totales de Productos Facturados a Clientes].[Código del
  Producto], [Productos Pedidos de Clientes].[Cantidad Pedida],
  [Totales de Productos Facturados a Clientes].[Total Facturado] AS
  [Cantidad Facturada]
FROM [Pedidos de Clientes] INNER JOIN ([Productos Pedidos de Clientes]
  INNER JOIN (Facturas INNER JOIN [Totales de Productos Facturados a
  Clientes] ON Facturas.[N° de Factura] = [Totales de Productos
  Facturados a Clientes].[N° de Factura]) ON [Productos Pedidos de
  Clientes].[Código del Producto] = [Totales de Productos Facturados a
  Clientes].[Código del Producto]) ON ([Pedidos de Clientes].[N° de
  Pedido] = [Productos Pedidos de Clientes].[N° de Pedido]) AND
  ([Pedidos de Clientes].[N° de Pedido] = Facturas.[N° de Pedido]);

```

---

De forma muy similar a la utilizada en la consulta anterior se comparan los productos pedidos por los clientes y los productos facturados por la empresa. En este caso, lo que se pretende medir, podría decirse, es la labor administrativa que procesa pedidos y facturas, de acuerdo con las existencias. Sin embargo, aun cuando puede haber existencia suficiente de productos, esta consulta ayuda a detectar demoras puramente administrativas.

### PRODUCTOS PEDIDOS Y FACTURADOS 1

---

```

SELECT [Productos Pedidos y Precios 3].[N° de Pedido], [Productos Pedidos
  y Precios 3].[Código del Producto], IIf(IsNull([Productos Pedidos y
  Facturados].[N° de Factura]),0,[Productos Pedidos y Facturados].[N°
  de Factura]) AS [Factura N°], [Productos Pedidos y Precios
  3].[Cantidad Pedida], IIf(IsNull([Cantidad Facturada]),0,[Cantidad
  Facturada]) AS [Cantidad Facturada Real], [Productos Pedidos y
  Precios 3].[Precio Unitario de Venta], [Productos Pedidos y Precios
  3].[% Descuento], [Productos Pedidos y Precios 3].[Precio con
  Descuento], [Productos Pedidos y Precios 3].[Monto del Producto],
  [Precio con Descuento]*[Cantidad Facturada Real] AS [Monto
  Facturado], Facturas.[Fecha de Emisión], Facturas.[Fecha de
  Despacho]
FROM ([Productos Pedidos y Facturados] RIGHT JOIN [Productos Pedidos y
  Precios 3] ON ([Productos Pedidos y Facturados].[Código del
  Producto] = [Productos Pedidos y Precios 3].[Código del Producto])
  AND ([Productos Pedidos y Facturados].[N° de Pedido] = [Productos
  Pedidos y Precios 3].[N° de Pedido])) LEFT JOIN Facturas ON
  [Productos Pedidos y Facturados].[N° de Factura] = Facturas.[N° de
  Factura];

```

---



Operando en conjunto con la consulta anterior se amplía la información mostrada para la relación entre productos pedidos y facturados, incluyendo los montos involucrados. Esto da otra perspectiva, ahora desde el punto de vista del flujo de caja comprometido, que auxilia en la labor administrativa.

### PRODUCTOS PEDIDOS Y PRECIOS

```

SELECT [Pedidos de Clientes].[N° de Pedido], [Pedidos de Clientes].Fecha,
       [Productos Pedidos de Clientes].[Código del Producto], [Precios de
       Productos].Fecha
FROM [Pedidos de Clientes] INNER JOIN ([Productos Pedidos de Clientes]
   INNER JOIN [Precios de Productos] ON [Productos Pedidos de
   Clientes].[Código del Producto] = [Precios de Productos].[Código del
   Producto]) ON [Pedidos de Clientes].[N° de Pedido] = [Productos
   Pedidos de Clientes].[N° de Pedido]
WHERE ((([Precios de Productos].Fecha)<=[Pedidos de Clientes].[Fecha]));

```

Muestra para una fecha dada los productos que han sido solicitados por los clientes y las fechas y precios de venta establecidos antes o durante la mencionada fecha. Opera en conjunto con las tres consultas siguientes.

### PRODUCTOS PEDIDOS Y PRECIOS 1

```

SELECT [Productos Pedidos y Precios].[N° de Pedido], [Productos Pedidos y
       Precios].[Pedidos de Clientes].Fecha, [Productos Pedidos y
       Precios].[Código del Producto], Max([Productos Pedidos y
       Precios].[Precios de Productos].Fecha) AS MaxFecha
FROM [Productos Pedidos y Precios]
GROUP BY [Productos Pedidos y Precios].[N° de Pedido], [Productos Pedidos
       y Precios].[Pedidos de Clientes].Fecha, [Productos Pedidos y
       Precios].[Código del Producto];

```

Encuentra de la consulta anterior la fecha próxima pasada más cercana a la fecha dada. Esto es, la fecha que, cotejada con la fecha del pedido, es aquella en que se fijó el precio de cada producto correspondiente involucrado en la transacción.

**PRODUCTOS PEDIDOS Y PRECIOS 2**


---

```

SELECT [Productos Pedidos y Precios 1].*, [Productos Pedidos de
  Clientes].[Cantidad Pedida], [Precios de Productos].[Precio Unitario
  de Venta]
FROM ([Productos Pedidos y Precios 1] INNER JOIN [Precios de Productos]
  ON ([Productos Pedidos y Precios 1].[Código del Producto] = [Precios
  de Productos].[Código del Producto]) AND ([Productos Pedidos y
  Precios 1].MaxFecha = [Precios de Productos].Fecha)) INNER JOIN
  [Productos Pedidos de Clientes] ON ([Productos Pedidos y Precios
  1].[Código del Producto] = [Productos Pedidos de Clientes].[Código
  del Producto]) AND ([Productos Pedidos y Precios 1].[N° de Pedido] =
  [Productos Pedidos de Clientes].[N° de Pedido]);

```

---

Encontrada la fecha de fijación del precio de venta de cada producto, que se corresponde con la fecha de cada pedido, se extrae entonces el precio de venta aplicable en cada caso.

**PRODUCTOS PEDIDOS Y PRECIOS 3**


---

```

SELECT [Productos Pedidos y Precios 2].[N° de Pedido], [Productos Pedidos
  y Precios 2].[Código del Producto], [Productos Pedidos y Precios
  2].[Cantidad Pedida], [Productos Pedidos y Precios 2].[Precio
  Unitario de Venta], [Escalas de Precios].[% Descuento], [Precio
  Unitario de Venta]-([Precio Unitario de Venta]*[% Descuento]) AS
  [Precio con Descuento], [Cantidad Pedida]*([Precio Unitario de
  Venta]-([Precio Unitario de Venta]*[% Descuento])) AS [Monto del
  Producto]
FROM [Escalas de Precios] INNER JOIN ([Productos Pedidos y Precios 2]
  INNER JOIN [Productos Terminados y Escalas de Precios] ON [Productos
  Pedidos y Precios 2].[Código del Producto] = [Productos Terminados y
  Escalas de Precios].[Código del Producto]) ON [Escalas de
  Precios].Escala = [Productos Terminados y Escalas de Precios].Escala
WHERE ((([Escalas de Precios].[Límite Inferior])<=[Cantidad Pedida]) AND
  (([Escalas de Precios].[Límite Superior])>=[Cantidad Pedida]));

```

---

Consolidados los datos de pedidos, productos y precios de venta aplicables (en las tres consultas anteriores) se calculan, con base en la política de descuentos por cantidad, los precios finales que han de aplicarse al cliente.

**PRODUCTOS TERMINADOS E INSUMOS**


---

```

SELECT [Productos Terminados].[Código del Producto], [Insumos en
  Fórmulas].[Código de Insumo], [Insumos en Fórmulas].Cantidad AS
  [Cantidad para el Lote]
FROM Insumos INNER JOIN ((Fórmulas INNER JOIN [Insumos en Fórmulas] ON

```

---

---

```

Fórmulas.[N° de Fórmula] = [Insumos en Fórmulas].[N° de Fórmula])
INNER JOIN [Productos Terminados] ON Fórmulas.[N° de Fórmula] =
[Productos Terminados].[N° de Fórmula]) ON Insumos.[Código de
Insumo] = [Insumos en Fórmulas].[Código de Insumo]
ORDER BY [Productos Terminados].[Código del Producto], [Insumos en
Fórmulas].[Código de Insumo];

```

---

Se listan las necesidades de cada insumo, contempladas en la formulación de los productos terminados.

#### PRODUCTOS TERMINADOS Y FÓRMULAS CONSULTA

---

```

SELECT DISTINCTROW [Productos Terminados].[Código del Producto],
[Productos Terminados].[N° de Fórmula], [Productos
Terminados].Presentación, [Productos Terminados].[Cantidad en
Presentación], [Productos Terminados].Destino, Fórmulas.Producto,
Fórmulas.Vehículo, Fórmulas.[Unidades de Presentación],
Fórmulas.Concentración, Fórmulas.[Unidades de Concentración],
Fórmulas.Departamento, Fórmulas.[Tamaño del Lote]
FROM Fórmulas INNER JOIN [Productos Terminados] ON Fórmulas.[N° de
Fórmula] = [Productos Terminados].[N° de Fórmula];

```

---

Se consolida información proveniente de las tablas [Productos Terminados] y [Fórmulas] para utilizarse en otras consultas y reportes.

#### PRODUCTOS Y ESCALAS CONSULTA

---

```

SELECT DISTINCTROW [Productos Terminados y Escalas de Precios].[Código
del Producto], [Escalas de Precios].Escala, [Escalas de
Precios].Descripción, [Escalas de Precios].[Límite Inferior],
[Escalas de Precios].[Límite Superior], [Escalas de Precios].[%
Descuento]
FROM [Escalas de Precios] INNER JOIN [Productos Terminados y Escalas de
Precios] ON [Escalas de Precios].Escala = [Productos Terminados y
Escalas de Precios].Escala;

```

---

Se consolida información de las escalas de precios aplicables a cada producto y los detalles de cada escala (debe recordarse que esta información se encuentra, por razones de diseño y normalización, en tablas aparte).

**PROVEEDORES CONSULTA**


---

```

SELECT DISTINCTROW Proveedores.[Código del Proveedor],
    Proveedores.[Nombre o Razón Social], Proveedores.[CI o RIF],
    Proveedores.Dirección, Proveedores.País, Países.[Código Telefónico],
    Proveedores.Estado, Proveedores.Ciudad, [Ciudades de
    Estados].[Código Telefónico], Proveedores.[Tenencia Del Local],
    Proveedores.[Figura Jurídica], Proveedores.[N° del RC],
    Proveedores.[Fecha del RC], Proveedores.[Duración del RC],
    Proveedores.[Tipo de Proveedor], Proveedores.Observaciones,
    Proveedores.Responsable, Personas.Apellidos, Personas.Nombres,
    Proveedores.[Fecha de Revisión]
FROM Personas RIGHT JOIN ([Ciudades de Estados] RIGHT JOIN (Países RIGHT
    JOIN Proveedores ON Países.País = Proveedores.País) ON ([Ciudades de
    Estados].Ciudad = Proveedores.Ciudad) AND ([Ciudades de
    Estados].Estado = Proveedores.Estado) AND ([Ciudades de
    Estados].País = Proveedores.País)) ON Personas.[Cédula de Identidad
    o Clave] = Proveedores.Responsable;

```

---

Se consolida alguna información de identificación relativa a los proveedores.

**RESUMEN DE ÓRDENES DE PRODUCCIÓN**


---

```

SELECT [Productos Terminados].[Código del Producto], [Lista Sencilla de
    Productos Terminados].Departamento, [Lista Sencilla de Productos
    Terminados].Nombre, [Productos Terminados].Presentación, [Órdenes de
    Producción].[N° de Orden], [Órdenes de Producción].[Fecha de
    Emisión], [Cantidad]/[Cantidad en Presentación] AS [Cantidad
    Ordenada], [Órdenes de Producción].[Fecha de Inicio], [Órdenes de
    Producción].[Fecha de Término], [Órdenes de Producción].[Fecha de
    Aprobación], Int([Producción Efectiva]/[Cantidad en Presentación])
    AS [Cantidad Producida]
FROM ([Productos Terminados] INNER JOIN [Lista Sencilla de Productos
    Terminados] ON [Productos Terminados].[Código del Producto] = [Lista
    Sencilla de Productos Terminados].[Código del Producto]) INNER JOIN
    [Órdenes de Producción] ON [Productos Terminados].[Código del
    Producto] = [Órdenes de Producción].Producto;

```

---

Se consolida y sintetiza alguna información conceptualmente relativa a las órdenes de producción.

**SALIDAS POR LOTE EN ALMACÉN I**


---

```

PARAMETERS [Forms]![Pedir Dos Fechas]![FechaÚnica] DateTime;
SELECT [Salidas del Almacén de Insumos].[Lote N°], [Llegadas de Insumos
    al Almacén].Insumo, Sum([Salidas del Almacén de Insumos].Cantidad)

```

---

---

```

AS Salidas
FROM [Llegadas de Insumos al Almacén] INNER JOIN [Salidas del Almacén de
Insumos] ON [Llegadas de Insumos al Almacén].[Lote N°] = [Salidas
del Almacén de Insumos].[Lote N°]
WHERE ((([Salidas del Almacén de Insumos].Fecha)<=[Forms]![Pedir Dos
Fechas]![FechaÚnica]))
GROUP BY [Salidas del Almacén de Insumos].[Lote N°], [Llegadas de Insumos
al Almacén].Insumo;

```

---

Agrupando por lotes de insumos recibidos en el almacén se totalizan las cantidades desechadas (salidas no imputables a la producción) de cada lote. Esta consulta se utiliza en los cómputos de existencia en el almacén de insumos.

#### **SUBTOTAL KG Y LT INSUMOS EN FÓRMULAS**

---

```

SELECT [Insumos en Fórmulas].[N° de Fórmula], Sum([Insumos en
Fórmulas]![Cantidad]*[Conversiones]![Ponderador]) AS Conversión,
Conversiones.A
FROM (Insumos INNER JOIN Conversiones ON Insumos.[Unidades de
Presentación] = Conversiones.De) INNER JOIN [Insumos en Fórmulas] ON
Insumos.[Código de Insumo] = [Insumos en Fórmulas].[Código de
Insumo]
GROUP BY [Insumos en Fórmulas].[N° de Fórmula], Conversiones.A
HAVING (((Conversiones.A)="Kg." Or (Conversiones.A)="Lts."));

```

---

Para efectos de incluir los totales de insumos en peso y volumen, principalmente en formularios e informes, se realizan los agrupamientos y suman las cantidades necesarias, convertidas de cualquier unidad a Kg. o Lts. Este es un requerimiento de la producción que sirve para tomarse una idea de la cantidad de insumo total requerida en el proceso productivo.

#### **SUBTOTAL KG Y LT INSUMOS EN ORDENES DE PROD**

---

```

SELECT [Insumos en Órdenes de Producción].[N° de Orden], Sum([Insumos en
Órdenes de Producción]![Cantidad a
Utilizar]*[Conversiones]![Ponderador]) AS Conversión, Conversiones.A
FROM (Insumos INNER JOIN Conversiones ON Insumos.[Unidades de
Presentación] = Conversiones.De) INNER JOIN [Insumos en Órdenes de
Producción] ON Insumos.[Código de Insumo] = [Insumos en Órdenes de
Producción].[Código de Insumo]
GROUP BY [Insumos en Órdenes de Producción].[N° de Orden], Conversiones.A
HAVING (((Conversiones.A)="Kg." Or (Conversiones.A)="Lts."));

```

---

Con una idea muy parecida a la esbozada en la consulta anterior se convierten las unidades y totalizan entorno a Kg. y Lts. ahora necesarios para cada orden de producción (que puede diferir de la necesaria en la fórmula debido a que cada orden propone una cierta cantidad de producto terminado a fabricarse, que puede diferir de lo establecido en la fórmula).

#### TOTAL UTILIZADO EN PRODUCCIÓN POR INSUMO

---

```
SELECT [Insumos en Órdenes de Producción].[N° de Orden], [Insumos en
    Órdenes de Producción].[Código de Insumo], Sum([Insumos en Órdenes
    de Producción].[Cantidad a Utilizar]) AS [Total Insumo en Orden]
FROM [Insumos en Órdenes de Producción]
GROUP BY [Insumos en Órdenes de Producción].[N° de Orden], [Insumos en
    Órdenes de Producción].[Código de Insumo];
```

---

Debido a que cada orden de producción puede utilizar de un mismo insumo cantidades provenientes de lotes diferentes, en esta consulta se totalizan estas cantidades mostrándose entonces lo utilizado del insumo (sin importar el lote del que provenga).

#### TOTAL UTILIZADO EN PRODUCCIÓN POR LOTE

---

```
SELECT [Insumos en Órdenes de Producción].[Lote N°], Sum([Insumos en
    Órdenes de Producción].[Cantidad a Utilizar]) AS [Total Utilizado]
FROM [Insumos en Órdenes de Producción]
GROUP BY [Insumos en Órdenes de Producción].[Lote N°];
```

---

De forma inversa a lo planteado en la consulta anterior se agrupan aquí los lotes de insumos y se totalizan las cantidades utilizadas en cada orden de producción.

#### TOTALES DE PRODUCTOS FACTURADOS A CLIENTES

---

```
SELECT [Productos Facturados a Clientes].[N° de Factura], [Productos
    Facturados a Clientes].[Código del Producto], Sum([Productos
    Facturados a Clientes].[Cantidad Facturada]) AS [Total Facturado]
FROM [Productos Facturados a Clientes]
GROUP BY [Productos Facturados a Clientes].[N° de Factura], [Productos
    Facturados a Clientes].[Código del Producto];
```

---

En vista de que un producto pedido por el cliente puede proceder, en una misma factura, de lotes diversos (pues un lote puede no alcanzar para surtir completo el mismo pedido), se totalizan las cantidades facturadas agrupando por factura y producto.

#### ÚLTIMAS FECHAS DE PRECIOS DE INSUMOS

---

```
SELECT [Insumos en Órdenes].Insumo, Max([Órdenes de Compra].Fecha) AS
MáxDeFecha
FROM [Órdenes de Compra] INNER JOIN [Insumos en Órdenes] ON [Órdenes de
Compra].[N°] = [Insumos en Órdenes].[Orden N°]
GROUP BY [Insumos en Órdenes].Insumo;
```

---

Agrupando para cada insumo se encuentran las últimas fechas en que fueron conocidos precios.

#### ÚLTIMAS FECHAS DE PRECIOS DE PRODUCTOS

---

```
SELECT [Precios de Productos].[Código del Producto], Max([Precios de
Productos].Fecha) AS MáxDeFecha
FROM [Precios de Productos]
GROUP BY [Precios de Productos].[Código del Producto];
```

---

Agrupando ahora para cada producto se encuentran las últimas fechas en que han sido fijados precios.

#### ÚLTIMOS PRECIOS DE INSUMOS CONSULTA

---

```
SELECT DISTINCTROW [Precios de Insumos].Insumo, [Precios de
Insumos].Fecha, [Precios de Insumos].[Precio Unitario]
FROM [Precios de Insumos] INNER JOIN [Últimas Fechas de Precios de
Insumos] ON ([Precios de Insumos].Fecha = [Últimas Fechas de Precios
de Insumos].MáxDeFecha) AND ([Precios de Insumos].Insumo = [Últimas
Fechas de Precios de Insumos].Insumo);
```

---

Establecidas las últimas fechas en que se conocen nuevos precios de insumos se encuentran los precios que se corresponden con estas fechas.

#### ÚLTIMOS PRECIOS DE PRODUCTOS CONSULTA

---

```
SELECT DISTINCTROW [Últimas Fechas de Precios de Productos].[Código del
Producto], [Últimas Fechas de Precios de Productos].MáxDeFecha AS
```

---

---

```

[Última Fecha], [Precios de Productos].[Precio Unitario de Venta]
FROM [Precios de Productos] INNER JOIN [Últimas Fechas de Precios de
Productos] ON ([Precios de Productos].[Código del Producto] =
[Últimas Fechas de Precios de Productos].[Código del Producto]) AND
([Precios de Productos].Fecha = [Últimas Fechas de Precios de
Productos].MáxDeFecha);

```

---

Establecidas las últimas fechas en que se fijan nuevos precios de productos terminados se encuentran los precios que se corresponden con estas fechas.

#### UNIDADES CONVERTIBLES

---

```

SELECT DISTINCTROW [Unidades de Medición].[Magnitud Física], [Unidades de
Medición].Unidades
FROM [Unidades de Medición]
GROUP BY [Unidades de Medición].[Magnitud Física], [Unidades de
Medición].Unidades
HAVING ((([Unidades de Medición].[Magnitud Física])<>"Unidades"))
ORDER BY [Unidades de Medición].[Magnitud Física], [Unidades de
Medición].Unidades;

```

---

Se listan aquellas magnitudes físicas y unidades de medición asociadas que pueden ser convertidas (no se contemplan conversiones que se apliquen a la magnitud “Unidades”).

#### UNIDADES DE CONCENTRACIÓN

---

```

SELECT DISTINCTROW [Unidades de Medición].[Magnitud Física], [Unidades de
Medición].Unidades
FROM [Unidades de Medición]
GROUP BY [Unidades de Medición].[Magnitud Física], [Unidades de
Medición].Unidades
HAVING ((([Unidades de Medición]![Magnitud Física])="Peso" Or ([Unidades
de Medición]![Magnitud Física])="Volumen"))
ORDER BY [Unidades de Medición].[Magnitud Física], [Unidades de
Medición].Unidades;

```

---

Se establecen aquellas magnitudes físicas que pueden considerarse para conformar la concentración de algún producto terminado (por ejemplo, la magnitud “Unidades” no puede estar entre ellas).



**UNIDADES DE PRESENTACIÓN**


---

```

SELECT DISTINCTROW [Unidades de Medición].[Magnitud Física], [Unidades de
Medición].Unidades
FROM [Unidades de Medición]
GROUP BY [Unidades de Medición].[Magnitud Física], [Unidades de
Medición].Unidades
HAVING ((([Unidades de Medición].[Magnitud Física])<>"Tiempo"))
ORDER BY [Unidades de Medición].[Magnitud Física], [Unidades de
Medición].Unidades;

```

---

Se establecen aquellas magnitudes físicas que pueden considerarse para conformar la presentación de algún producto terminado (por ejemplo, las magnitudes de tiempo no pueden estar entre ellas).

**UNIDADES DE TIEMPO**


---

```

SELECT DISTINCTROW [Unidades de Medición].Unidades
FROM [Unidades de Medición]
GROUP BY [Unidades de Medición].[Magnitud Física], [Unidades de
Medición].Unidades
HAVING ((([Unidades de Medición].[Magnitud Física])="Tiempo"))
ORDER BY [Unidades de Medición].Unidades;

```

---

Se aíslan aquellas unidades de medición del tiempo (días, semanas, meses, etc.).

**VENTAS ANUALES D**


---

```

SELECT Year([Fecha]) AS Año, Sum([Productos Pedidos y Precios 3].[Monto
del Producto]) AS [Monto del Producto]
FROM ([Productos Pedidos y Precios 3] INNER JOIN [Pedidos de Clientes] ON
[Productos Pedidos y Precios 3].[N° de Pedido] = [Pedidos de
Clientes].[N° de Pedido]) INNER JOIN [Lista Sencilla de Productos
Terminados] ON [Productos Pedidos y Precios 3].[Código del Producto]
= [Lista Sencilla de Productos Terminados].[Código del Producto]
WHERE ((([Lista Sencilla de Productos
Terminados].Departamento)=[Forms]![Pedir Año para resumen de
Ventas]![Departamento]))
GROUP BY Year([Fecha]);

```

---

Dado como parámetro un departamento (sólidos, líquidos u otros) y agrupando por años, se totalizan las ventas en bolívares logradas por dicho departamento.

**VENTAS ANUALES G**


---

```

SELECT Year([Fecha]) AS Año, Sum([Productos Pedidos y Precios 3].[Monto
del Producto]) AS [Monto del Producto]
FROM [Productos Pedidos y Precios 3] INNER JOIN [Pedidos de Clientes] ON
[Productos Pedidos y Precios 3].[N° de Pedido] = [Pedidos de
Clientes].[N° de Pedido]
GROUP BY Year([Fecha]);

```

---

Agrupando por años se encuentran las ventas totales (de todos los productos).

**VENTAS ANUALES P**


---

```

SELECT Year([Fecha]) AS Año, Sum([Productos Pedidos y Precios
3].[Cantidad Pedida]) AS [Cantidad Pedida], Sum([Productos Pedidos y
Precios 3].[Monto del Producto]) AS [Monto del Producto]
FROM [Productos Pedidos y Precios 3] INNER JOIN [Pedidos de Clientes] ON
[Productos Pedidos y Precios 3].[N° de Pedido] = [Pedidos de
Clientes].[N° de Pedido]
WHERE ((([Productos Pedidos y Precios 3].[Código del
Producto])=[Forms]![Pedir Año para resumen de Ventas]![Producto]))
GROUP BY Year([Fecha]);

```

---

Dado como parámetro un producto y agrupando por años se totalizan las ventas en bolívares logradas por dicho producto.

**VENTAS MENSUALES D**


---

```

PARAMETERS [Forms]![Pedir Año para resumen de Ventas]![Año] Short;
SELECT Month([Fecha]) AS Mes, Sum([Productos Pedidos y Precios 3].[Monto
del Producto]) AS [Monto del Producto]
FROM [Lista Sencilla de Productos Terminados] INNER JOIN ([Productos
Pedidos y Precios 3] INNER JOIN [Pedidos de Clientes] ON [Productos
Pedidos y Precios 3].[N° de Pedido] = [Pedidos de Clientes].[N° de
Pedido]) ON [Lista Sencilla de Productos Terminados].[Código del
Producto] = [Productos Pedidos y Precios 3].[Código del Producto]
WHERE (((Year([Fecha])=[Forms]![Pedir Año para resumen de Ventas]![Año])
AND (([Lista Sencilla de Productos
Terminados].Departamento)=[Forms]![Pedir Año para resumen de
Ventas]![Departamento])))
GROUP BY Month([Fecha]);

```

---

Dados como parámetros un departamento y un año, agrupando por mes, se totalizan las ventas en bolívares logradas por dicho departamento en el año.

**VENTAS MENSUALES G**


---

```

PARAMETERS [Forms]![Pedir Año para resumen de Ventas]![Año] Short;
SELECT Month([Fecha]) AS Mes, Sum([Productos Pedidos y Precios 3].[Monto
del Producto]) AS [Monto del Producto]
FROM [Productos Pedidos y Precios 3] INNER JOIN [Pedidos de Clientes] ON
[Productos Pedidos y Precios 3].[N° de Pedido] = [Pedidos de
Clientes].[N° de Pedido]
WHERE (((Year([Fecha]))=[Forms]![Pedir Año para resumen de
Ventas]![Año]))
GROUP BY Month([Fecha]);

```

---

Dado como parámetro un año, agrupando por mes, se totalizan las ventas logradas (para todos los productos).

**VENTAS MENSUALES P**


---

```

PARAMETERS [Forms]![Pedir Año para resumen de Ventas]![Año] Short;
SELECT Month([Fecha]) AS Mes, Sum([Productos Pedidos y Precios
3].[Cantidad Pedida]) AS [Cantidad Pedida], Sum([Productos Pedidos y
Precios 3].[Monto del Producto]) AS [Monto del Producto]
FROM [Productos Pedidos y Precios 3] INNER JOIN [Pedidos de Clientes] ON
[Productos Pedidos y Precios 3].[N° de Pedido] = [Pedidos de
Clientes].[N° de Pedido]
WHERE (((Year([Fecha]))=[Forms]![Pedir Año para resumen de Ventas]![Año])
AND (([Productos Pedidos y Precios 3].[Código del
Producto])=[Forms]![Pedir Año para resumen de Ventas]![Producto]))
GROUP BY Month([Fecha]);

```

---

Dados como parámetros un producto y un año, agrupando por mes, se totalizan las ventas en bolívares logradas por dicho producto en el año.

**VENTAS SEMESTRALES D**


---

```

PARAMETERS [Forms]![Pedir Año para resumen de Ventas]![Año] Short;
SELECT IIf(Month([Fecha]) Between 1 And 6,"1er. Sem.,"2do. Sem.") AS
Semestre, Sum([Productos Pedidos y Precios 3].[Monto del Producto])
AS [Monto del Producto]
FROM ([Productos Pedidos y Precios 3] INNER JOIN [Pedidos de Clientes] ON
[Productos Pedidos y Precios 3].[N° de Pedido] = [Pedidos de
Clientes].[N° de Pedido]) INNER JOIN [Lista Sencilla de Productos
Terminados] ON [Productos Pedidos y Precios 3].[Código del Producto]
= [Lista Sencilla de Productos Terminados].[Código del Producto]
WHERE (((Year([Fecha]))=[Forms]![Pedir Año para resumen de Ventas]![Año])
AND (([Lista Sencilla de Productos

```

---

```

Terminados].Departamento)=[Forms]![Pedir Año para resumen de
Ventas]![Departamento]))
GROUP BY IIf(Month([Fecha]) Between 1 And 6,"1er. Sem.,"2do. Sem.");

```

Dados como parámetros un departamento y un año, agrupando por semestre (1 y 2), se totalizan las ventas en bolívares logradas por dicho departamento en el año.

#### VENTAS SEMESTRALES G

```

PARAMETERS [Forms]![Pedir Año para resumen de Ventas]![Año] Short;
SELECT IIf(Month([Fecha]) Between 1 And 6,"1er. Sem.,"2do. Sem.") AS
Semestre, Sum([Productos Pedidos y Precios 3].[Monto del Producto])
AS [Monto del Producto]
FROM [Productos Pedidos y Precios 3] INNER JOIN [Pedidos de Clientes] ON
[Productos Pedidos y Precios 3].[N° de Pedido] = [Pedidos de
Clientes].[N° de Pedido]
WHERE (((Year([Fecha]))=[Forms]![Pedir Año para resumen de
Ventas]![Año]))
GROUP BY IIf(Month([Fecha]) Between 1 And 6,"1er. Sem.,"2do. Sem.);

```

Dado como parámetro un año, agrupando por semestre (1 y 2), se totalizan las ventas logradas (considerando todos los productos).

#### VENTAS SEMESTRALES P

```

PARAMETERS [Forms]![Pedir Año para resumen de Ventas]![Año] Short;
SELECT IIf(Month([Fecha]) Between 1 And 6,"1er. Sem.,"2do. Sem.") AS
Semestre, Sum([Productos Pedidos y Precios 3].[Cantidad Pedida]) AS
[Cantidad Pedida], Sum([Productos Pedidos y Precios 3].[Monto del
Producto]) AS [Monto del Producto]
FROM [Productos Pedidos y Precios 3] INNER JOIN [Pedidos de Clientes] ON
[Productos Pedidos y Precios 3].[N° de Pedido] = [Pedidos de
Clientes].[N° de Pedido]
WHERE (((Year([Fecha]))=[Forms]![Pedir Año para resumen de Ventas]![Año])
AND (([Productos Pedidos y Precios 3].[Código del
Producto])=[Forms]![Pedir Año para resumen de Ventas]![Producto]))
GROUP BY IIf(Month([Fecha]) Between 1 And 6,"1er. Sem.,"2do. Sem.);

```

Dados como parámetros un producto y un año, agrupando por semestre (1 y 2), se totalizan las ventas logradas por dicho producto en el año.

**VENTAS TRIMESTRALES D**


---

```

PARAMETERS [Forms]![Pedir Año para resumen de Ventas]![Año] Short;
SELECT IIf(Month([Fecha]) Between 1 And 3,"I",IIf(Month([Fecha]) Between
    4 And 6,"II",IIf(Month([Fecha]) Between 7 And 9,"III","IV"))) AS
    Trimestre, Sum([Productos Pedidos y Precios 3].[Monto del Producto])
    AS [Monto del Producto]
FROM [Lista Sencilla de Productos Terminados] INNER JOIN ([Productos
    Pedidos y Precios 3] INNER JOIN [Pedidos de Clientes] ON [Productos
    Pedidos y Precios 3].[N° de Pedido] = [Pedidos de Clientes].[N° de
    Pedido]) ON [Lista Sencilla de Productos Terminados].[Código del
    Producto] = [Productos Pedidos y Precios 3].[Código del Producto]
WHERE (((Year([Fecha])=[Forms]![Pedir Año para resumen de Ventas]![Año])
    AND (([Lista Sencilla de Productos
    Terminados].Departamento)=[Forms]![Pedir Año para resumen de
    Ventas]![Departamento])))
GROUP BY IIf(Month([Fecha]) Between 1 And 3,"I",IIf(Month([Fecha])
    Between 4 And 6,"II",IIf(Month([Fecha]) Between 7 And
    9,"III","IV")));

```

---

Dados como parámetros un departamento y un año, agrupando por trimestre (1, 2, 3 y 4), se totalizan las ventas en bolívares logradas por dicho departamento en el año.

**VENTAS TRIMESTRALES G**


---

```

PARAMETERS [Forms]![Pedir Año para resumen de Ventas]![Año] Short;
SELECT IIf(Month([Fecha]) Between 1 And 3,"I",IIf(Month([Fecha]) Between
    4 And 6,"II",IIf(Month([Fecha]) Between 7 And 9,"III","IV"))) AS
    Trimestre, Sum([Productos Pedidos y Precios 3].[Monto del Producto])
    AS [Monto del Producto]
FROM [Productos Pedidos y Precios 3] INNER JOIN [Pedidos de Clientes] ON
    [Productos Pedidos y Precios 3].[N° de Pedido] = [Pedidos de
    Clientes].[N° de Pedido]
WHERE (((Year([Fecha])=[Forms]![Pedir Año para resumen de
    Ventas]![Año])))
GROUP BY IIf(Month([Fecha]) Between 1 And 3,"I",IIf(Month([Fecha])
    Between 4 And 6,"II",IIf(Month([Fecha]) Between 7 And
    9,"III","IV")));

```

---

Dado como parámetro un año, agrupando por trimestre (1, 2, 3 y 4), se totalizan las ventas logradas (para todos los productos).

**VENTAS TRIMESTRALES P**


---

```

PARAMETERS [Forms]![Pedir Año para resumen de Ventas]![Año] Short;

```

---

---

```

SELECT IIf(Month([Fecha]) Between 1 And 3,"I",IIf(Month([Fecha]) Between
4 And 6,"II",IIf(Month([Fecha]) Between 7 And 9,"III","IV"))) AS
Trimestre, Sum([Productos Pedidos y Precios 3].[Cantidad Pedida]) AS
[Cantidad Pedida], Sum([Productos Pedidos y Precios 3].[Monto del
Producto]) AS [Monto del Producto]
FROM [Productos Pedidos y Precios 3] INNER JOIN [Pedidos de Clientes] ON
[Productos Pedidos y Precios 3].[N° de Pedido] = [Pedidos de
Clientes].[N° de Pedido]
WHERE (((Year([Fecha])=[Forms]![Pedir Año para resumen de Ventas]![Año])
AND (([Productos Pedidos y Precios 3].[Código del
Producto])=[Forms]![Pedir Año para resumen de Ventas]![Producto])))
GROUP BY IIf(Month([Fecha]) Between 1 And 3,"I",IIf(Month([Fecha])
Between 4 And 6,"II",IIf(Month([Fecha]) Between 7 And
9,"III","IV")));

```

---

Dados como parámetros un producto y un año, agrupando por trimestre (1, 2, 3 y 4), se totalizan las ventas logradas por dicho producto en el año.

### II.3. Resumen

En este capítulo se ha presentado un panorama detallado del diseño de la base de datos del SARCAI v.2.0. Especial énfasis se ha hecho en las explicaciones (sobre todo en el nivel conceptual) de los elementos fundamentales de tablas y consultas. La esperanza es que el lector tenga una visión clara del modelo de la base de datos relacional escogido para la solución del problema de información del inventario en la Planta de Medicamentos.

Se han descrito 78 tablas, 117 consultas y se ha mostrado un diagrama, de alto nivel, que muestra las relaciones entre los elementos de la base (hay que recordar que las consultas en el modelo relacional son tablas “virtuales” en el sentido de que existen desde el punto de vista lógico más no físico. Las consultas representan vistas de la base de datos que toman el material que presentan, de las tablas -que sí existen físicamente-).

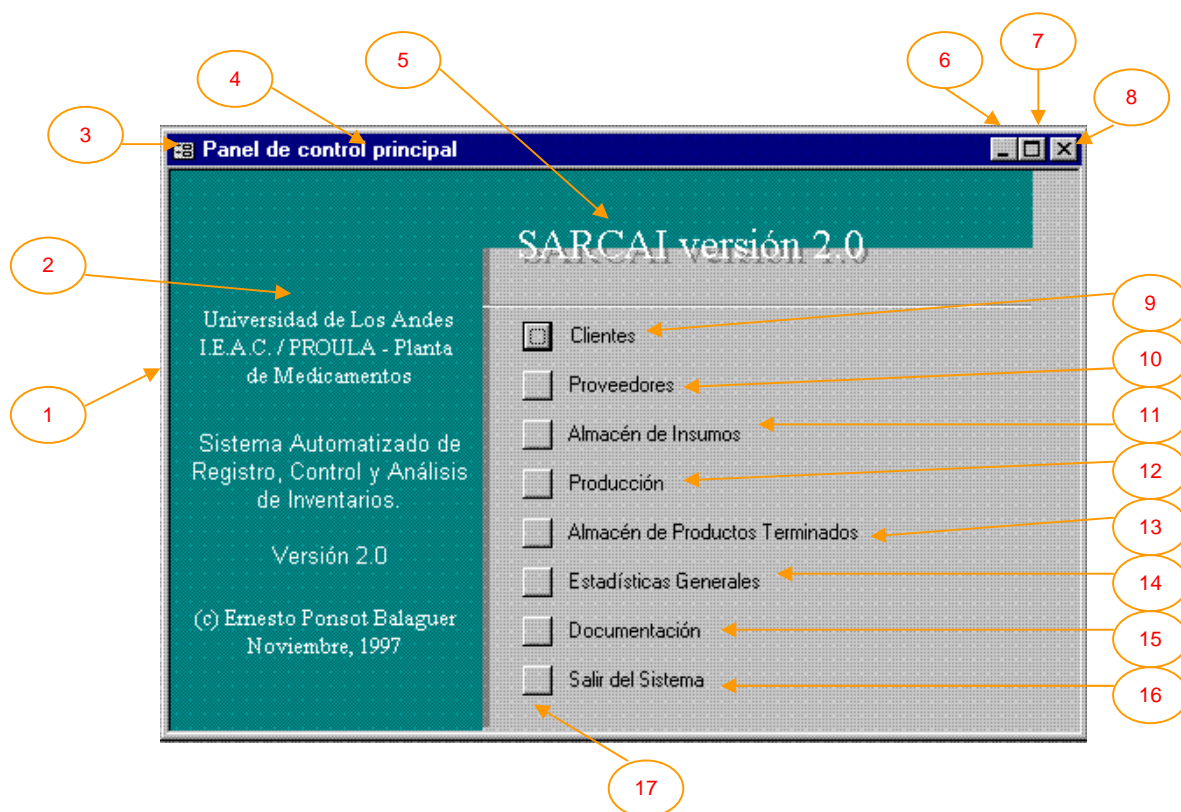
## CAPÍTULO III

### Estructura de Paneles del Sistema

Este capítulo trata de las distintas secciones presentes en SARCAI. La navegación por el sistema se realiza mediante opciones que al seleccionarse (dando clic al botón de la opción), invocan paneles con subopciones o ejecutan acciones diversas. Un panel es una ventana que tiene el papel de menú. El usuario verá siempre un mismo tipo de panel en el que cambian los botones, de acuerdo con la sección en que se encuentre. El sistema controla automáticamente la presentación de los paneles de forma tal que cuando hay movimiento de uno a otro, no se abre una nueva ventana (lo que consumiría muchos recursos de memoria), sino que cambian las opciones y el título de la ventana. Así pues, los paneles contienen la estructura lógica del sistema de información.

El primer panel se visualiza inmediatamente se haya cargado una sesión de trabajo con el sistema, y contiene el nivel más alto en la jerarquía de opciones.

### III.1. Panel de Control Principal



Es la primera ventana de la aplicación. Contiene las opciones primarias del sistema que reflejan los distintos aspectos que toca el software. Siguiendo la numeración impuesta en la gráfica, a continuación se explica cada elemento:

- 1) Marco de la ventana. Delimita el espacio que corresponde a la ventana mostrada. Si se acerca el puntero del ratón a este elemento, es posible cambiar sus dimensiones dando clic y arrastrando en cualquier dirección.
- 2) Área de identificación del sistema e información de Copyright ©. En este caso, todas las ventanas del Panel de Control tienen estos datos.



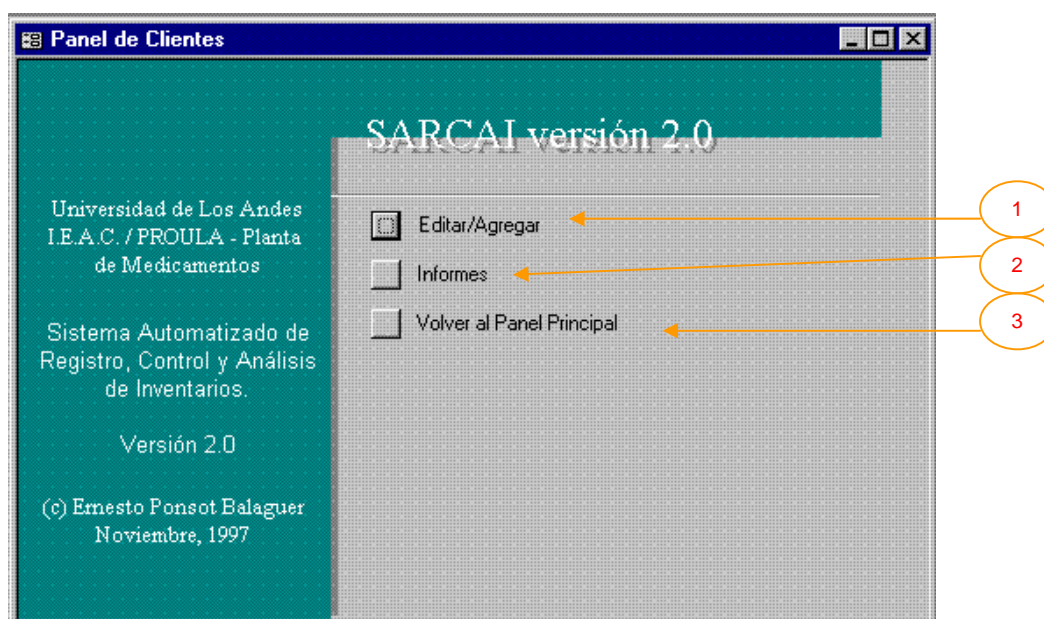
- 3) Despliega el menú de movimiento y tamaño de la ventana. Este menú es compartido por todas las ventanas en Windows 95 y su uso es una alternativa al uso de los objetos 6, 7 y 8 mostrados.
- 4) Título de la ventana. Muestra en cada caso el nombre que se le ha adjudicado a la ventana. Identifica el área del sistema con la que el usuario se dispone a trabajar. En este caso está mostrando que el usuario se encuentra en el *Panel de Control Principal*, presto a seleccionar alguna de las opciones primarias del sistema.
- 5) Área del título del sistema de información. Todos los paneles muestran el mismo título “SARCAI versión 2.0”, es decir, Sistema Automatizado de Registro, Control y Análisis de Inventarios, versión 2.0 (la versión 1.0 de SARCAI fue implementada para la versión 7.0 de Access y la versión que se presenta en este trabajo se programó con la nueva versión Access 97).
- 6) Botón de minimización de la ventana. Toda ventana en Windows 95 puede ser minimizada transformándola en un icono que se retira hacia la barra de inicio (en el escritorio) o hacia la parte inferior izquierda de la ventana que la contiene. Esta acción no cierra la ventana, sólo permite liberar espacio de pantalla con el objetivo de visualizar algún otro aspecto de la aplicación. La ventana principal de Access también puede ser minimizada dejando libre el espacio de pantalla para ejecutar, desde el escritorio de Windows, algún otro programa (como Excel, por ejemplo).
- 7) Botón de maximización de la ventana. Como antes, toda ventana en Windows 95 puede ser maximizada haciendo que ocupe el área máxima de pantalla. Para restablecer las dimensiones normales de la ventana, hay que volver a dar clic sobre el mismo icono, que ahora muestra varios cuadrados.
- 8) Botón de cierre de la ventana. Nuevamente, todas las ventanas de Windows muestran este botón. Dando clic en él, la ventana se cierra. La operación de cierre de la ventana libera el espacio en memoria que está siendo utilizado por ella y devuelve el control a la ventana “madre” desde donde se invocó. Generalmente, las operaciones que ameritan confirmación se activan cuando el usuario cierra una ventana. Por ejemplo, si el usuario debe confirmar

la salida de la aplicación, recibirá un mensaje de advertencia cuando trate de cerrar la ventana principal o raíz.

- 9) Clientes. Esta opción cede el control del sistema al área dedicada a los clientes. El enfoque del área de trabajo se mueve hacia el *Panel de Clientes*.
- 10) Proveedores. El enfoque del área de trabajo se mueve hacia el *Panel de Proveedores*.
- 11) Almacén de insumos. El enfoque del área de trabajo se mueve hacia el *Panel de Insumos*.
- 12) Producción. El enfoque del área de trabajo se mueve hacia el *Panel de Producción*.
- 13) Almacén de Productos Terminados. El enfoque del área de trabajo se mueve hacia el *Panel de Productos Terminados*.
- 14) Estadísticas generales. El enfoque del área de trabajo se mueve hacia el *Panel de Estadísticas Generales*.
- 15) Documentación. Esta opción abre el formulario [Documentación] (ver Anexo 1, Figura 9, pág. 16) desde donde pueden invocarse los contenidos del presente documento resumidos y organizados de manera un tanto diferente. La documentación muestra los conceptos relevantes que se utilizan y describe todos los objetos que el usuario debe manipular para operar con el sistema. Estos documentos se presentan como páginas “html” y por tanto se requiere la instalación previa de algún software de navegación en la Internet para poder visualizarlas. La ventaja de que la documentación se encuentre en formato html estriba en que automáticamente está disponible en toda la red de la organización (e incluso más allá).
- 16) Salir del sistema. Esta es la forma más segura de abandonar el trabajo con SARCAI. Siempre que termine la labor con el sistema, antes de desconectar el equipo o de comenzar la operación de algún otro programa, deberá utilizarse esta opción, pues es la única que garantiza el cierre de todos los archivos que estén abiertos y la actualización de la base de datos.
- 17) Botón de opción. Este es un objeto que el usuario encontrará frecuentemente en SARCAI (y en general en la mayoría de las aplicaciones Windows 95). La selección de cualquiera

de las opciones mostradas se realiza dando un clic sobre el botón de la opción correspondiente. De esta forma, el usuario, fácilmente se mueve de un lugar a otro en el sistema “apretando el botón apropiado” (con el puntero del ratón localizado sobre el botón y pulsando una vez el control izquierdo).

### III.1.1. Panel de Clientes



Cuando en el Panel de Control Principal se selecciona Clientes, lo que se obtiene es el despliegue de esta ventana. Un cliente es para SARCAI una persona o empresa a la cual la organización vende (o ha vendido e incluso venderá) sus productos. Los clientes son la razón de ser de la organización, ya que son los que generan la demanda. En el caso particular de los medicamentos, dado que el expendio de fármacos está especialmente regulado por la normativa legal, sobre todo cuando se enfocan las ventas al mayoreo (que es el caso), es imprescindible contar con información especial de los clientes. Adicionalmente, en lo que respecta al sistema de inventarios, es obvia la importancia que debe adjudicárseles, ya que mantener su “buena voluntad” hacia la empresa es un objetivo primario.

La gráfica contiene algunos elementos ya vistos y otros nuevos. Los elementos nuevos serán aquí, y de ahora en adelante, objeto de explicación. No se repetirán las explicaciones de aquellos elementos que ya hayan sido tratados en páginas anteriores.

Este panel es el primero en mostrar tres opciones que el usuario verá con mucha frecuencia en los paneles siguientes. Debido a que la mayor parte del trabajo con el sistema se dedica a la manipulación (entrada, modificación y eliminación) de datos, la mayoría de las secciones que contiene proveen herramientas para desarrollar esta actividad. La estructura de este panel se utilizará como ejemplo y las explicaciones dadas aquí servirán más adelante.

La actividad de introducción de datos nuevos, modificación de los ya existentes y eliminación de los no deseados, se engloba en SARCAI bajo las palabras Editar/Agregar. Siempre que se desee Editar/Agregar información a la base de datos, la acción desencadenada es la apertura de un **formulario**.

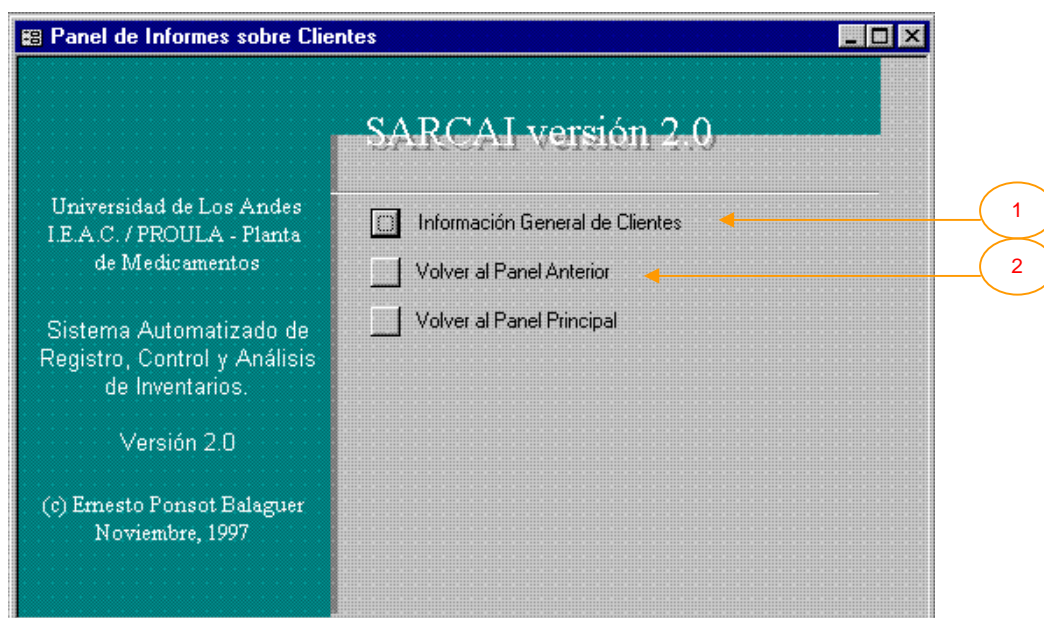
Un formulario es la ventana a través de la cual el usuario vislumbra los datos contenidos en la base de datos. Puede contener información de una sola tabla o de varias tablas a la vez. Estas tablas podrán estar relacionadas según se muestra en el modelo de datos del sistema (ver Capítulo II). Cada formulario representa entonces una “vista” distinta de la base de datos y muestra o solicita, mediante controles, la información de las tablas involucradas. También presenta controles que ejecutan acciones como la impresión de informes, llamadas a otros formularios emergentes e incluso ejecución de cómputos especiales y estimaciones.

Otra de las palabras o frases claves inmersas en SARCAI es **informes**. Un informe es un objeto resultante de la mezcla de información de diversa procedencia (generalmente realizada con una consulta SQL) que resulta en una serie de páginas con formato, preparadas especialmente para ser impresas o incorporadas a otros documentos. Al contrario de lo que ocurre con el formulario, el usuario no está en capacidad de interactuar con el informe. El informe sólo presenta los datos de la base de datos, relacionados convenientemente y dispuestos de tal manera que sean útiles para tomar decisiones. En mucho es cierto que los informes son el fin último del sistema, ya que son el resumen confiable de datos que necesita la organización para funcionar.

En el panel de clientes, los objetos nuevos son entonces:

- 1) Editar/Agregar. Es la opción que permite al usuario abrir el formulario [Clientes] (ver Anexo 1, Figuras 2 y 3, pág. 5 y 6), el cual contiene toda la información relativa al control de clientes de la planta. En este formulario, el usuario podrá crear un nuevo cliente e introducir sus datos, modificar los datos de un cliente ya creado o eliminar algún cliente que deba salir del sistema.
- 2) Informes. Esta opción presenta al usuario el panel donde podrá escoger los informes relacionados con el tema que está tratando.
- 3) Volver al panel principal. Esta opción no presenta una nueva ventana, es del tipo de las opciones que ejecuta una acción. En este caso, la acción ejecutada es el retorno al panel principal (desde donde fue llamado el panel de clientes) para que el usuario pueda seleccionar una opción de trabajo distinta.

### III.1.1.1. Panel de Informes sobre Clientes



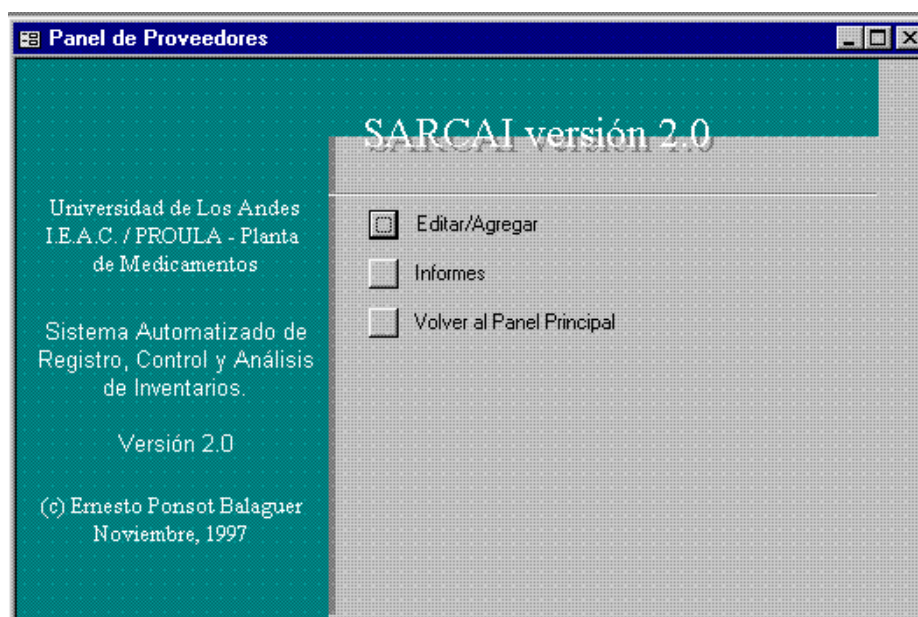
En este panel se presenta la opción de mostrar los informes disponibles, principalmente relacionados con los clientes de la organización. En esta versión del sistema sólo está disponible un informe directamente relacionado con los clientes titulado [Clientes], sin embargo, la estructura del sistema y el hecho de tener un panel dedicado a los informes en cada sección, permiten agregar fácilmente tantos informes como se desee. En futuras versiones de SARCAI, luego de la operación concienzuda del sistema, seguramente se agregarán nuevos informes, tanto a ésta como a otras secciones.

Los elementos novedosos en este panel, son:

- 1) Información General de Clientes. Opción que permite presentar un informe que contiene todos los clientes registrados en la base de datos, con todos sus datos. El informe presentado se denomina [Clientes] (ver Anexo 2, Punto 3, pág. 4 y Anexo 3, “Cliente:”).
- 2) Volver al panel anterior. Esta opción es similar a la opción de Volver al panel principal. La diferencia estriba en que se retorna el control al panel inmediatamente anterior (panel “madre”) y no al panel principal. Esto flexibiliza la navegación por los distintos paneles,

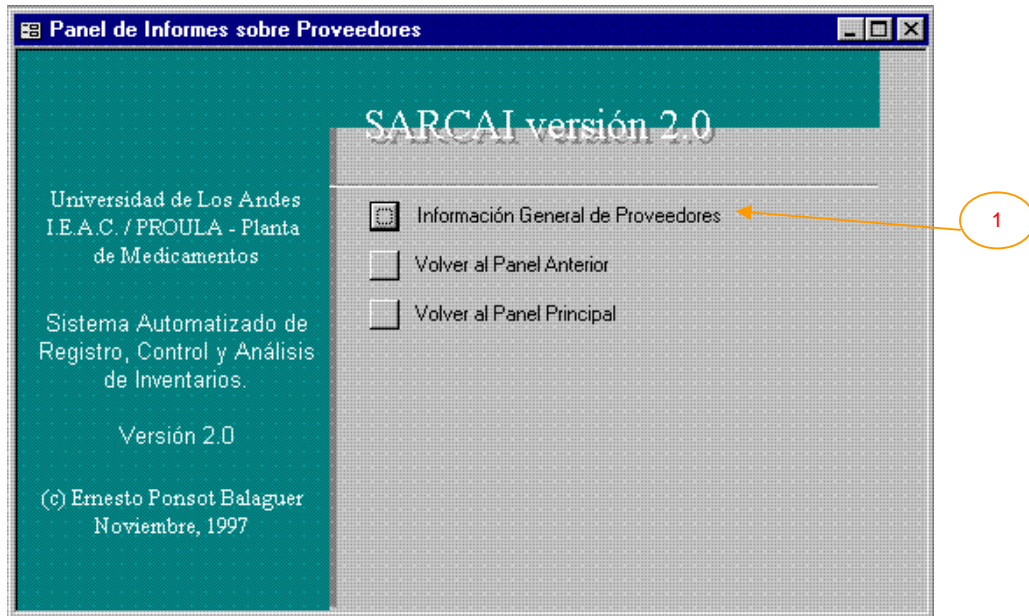
pues permite el trabajo con una sección del programa sin tener que moverse constantemente al panel principal. Claro está, si el usuario desea tocar alguna otra sección distinta de la actual, por ejemplo, operar con los pedidos, deberá retornar al panel principal y de allí tomar el camino correcto hasta llegar a la nueva sección de su preferencia.

### III.1.2. Panel de Proveedores



Se obtiene al seleccionar Proveedores en el panel principal. Presenta las distintas opciones relacionadas con los proveedores de la planta. Un proveedor en SARCAI se interpreta como una persona o empresa a la cual se le compran los insumos necesarios para la producción. Con los proveedores se establece una relación clave en lo que respecta al manejo de inventarios, ya que de la fiabilidad, calidad y completitud de los pedidos servidos por los proveedores depende en gran parte el nivel de inventario de insumos que la organización deba mantener y, por consiguiente, la inversión de capitales inmóviles dedicados al inventario. Las opciones presentes en este panel son similares, aplicadas al caso de los proveedores, que las explicadas para el Panel de Clientes. El formulario que se abre en la opción Editar/Agregar es en este caso [Proveedores] (ver Anexo 1, Figuras 43 y 44, pág. 86 y 87).

### III.1.2.1. Panel de Informes sobre Proveedores

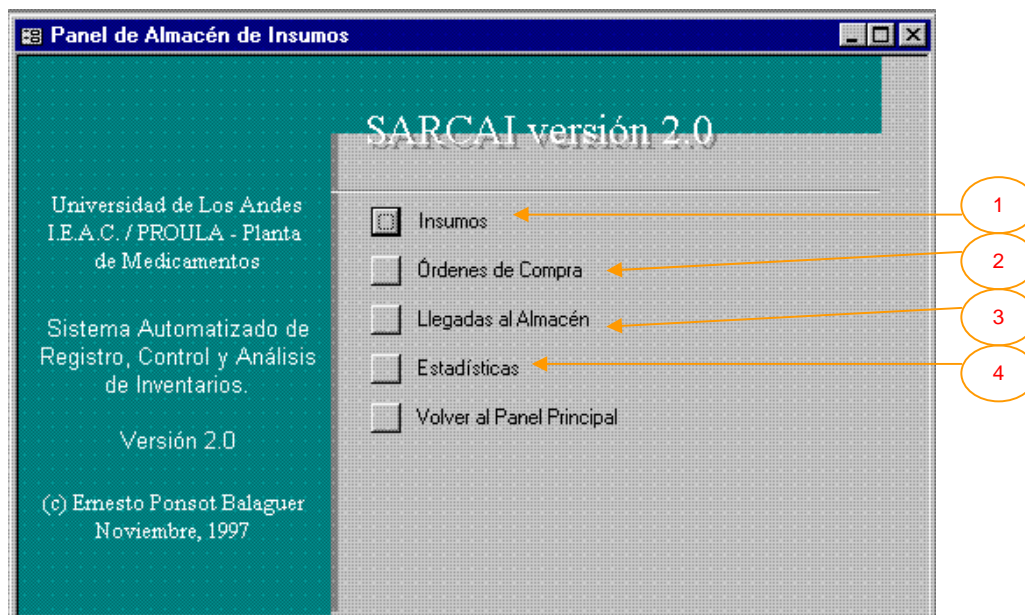


Se obtiene al seleccionar informes en el panel de proveedores.

- 1) Presenta el informe [Proveedores] (ver Anexo 2, Punto 54, pág. 60 y Anexo 3, “Proveedor:”), el cual lista todos los proveedores y sus datos.



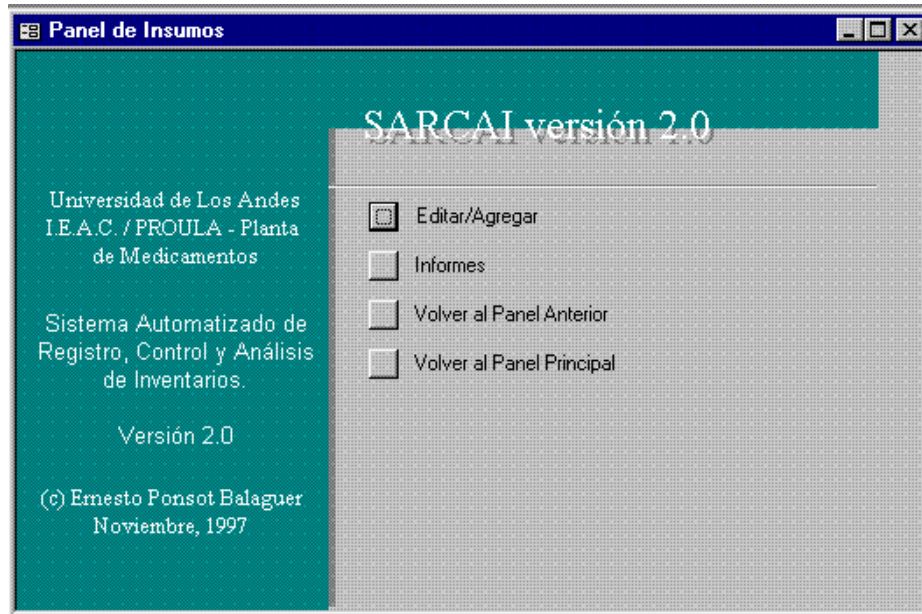
### III.1.3. Panel de Almacén de Insumos



El almacén de insumos es uno de los dos almacenes controlados en SARCAI. En este almacén, se trata con los insumos que se compran a los proveedores y luego se utilizan en el proceso productivo. Así, insumo es cualquier elemento, comprado a un proveedor, que se utiliza en alguno de los procesos productivos de la planta. Sobre el particular se presentan las opciones:

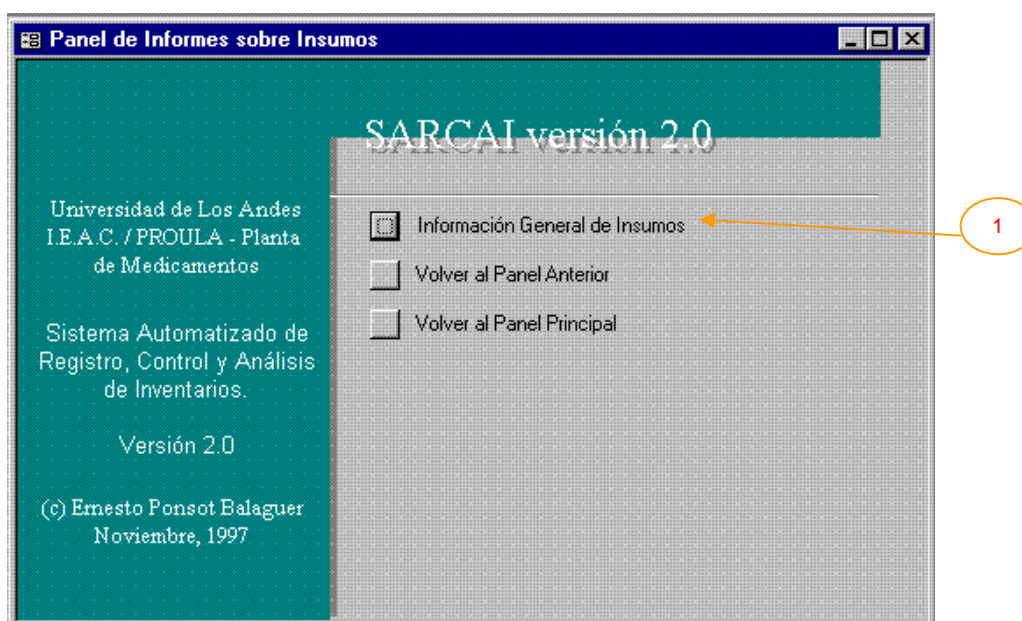
- 1) Insumos. Mueve el enfoque del sistema hacia el *Panel de Insumos*.
- 2) Órdenes de Compra. Mueve el enfoque del sistema hacia el *Panel de Órdenes de Compra*.
- 3) Llegadas al Almacén. Mueve el enfoque del sistema hacia el *Panel de Llegadas al Almacén de Insumos*.
- 4) Estadísticas. Mueve el enfoque del sistema hacia el *Panel de Estadísticas de Insumos*.

### III.1.3.1. Panel de Insumos



Se obtiene al seleccionar Insumos en el panel de almacén de insumos. Presenta las distintas subopciones relacionadas con la “creación” de los insumos en el sistema. Editar/Agregar abre el formulario [Insumos] (ver Anexo 1, Figura 21, pág. 34), donde el usuario puede definir o “crear” un insumo que luego será utilizado en la producción. El resto de las opciones no presenta novedad.

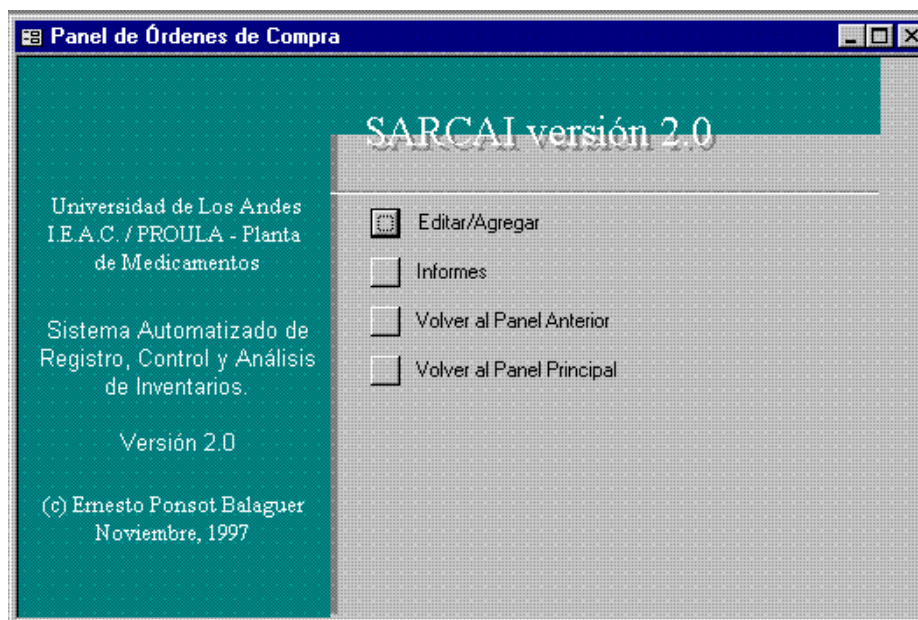
### III.1.3.1.1. Panel de Informes sobre Insumos



Este panel presenta las opciones para la obtención de informes relacionados con los insumos. En este momento:

- 1) Información General de Insumos. Abre el informe [Insumos] (ver Anexo 2, Punto 29, pág. 25 y Anexo 3, “Insumos”) que presenta un reporte de todos los insumos con sus respectivos datos.

### III.1.3.2. Panel de Órdenes de Compra

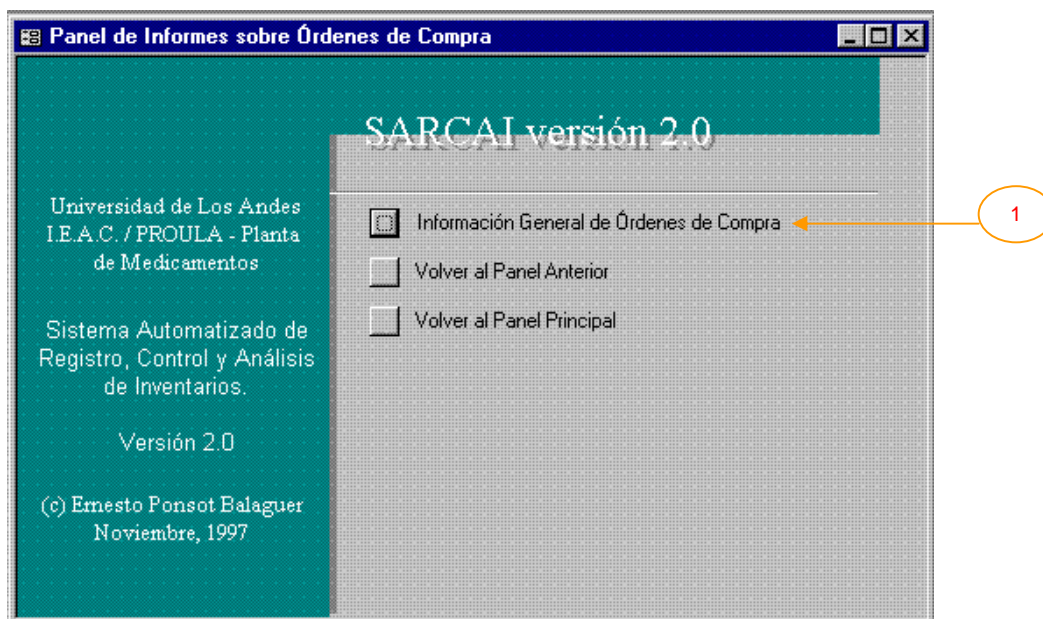


Una orden de compra es un documento emitido por la organización, donde se establece una transacción de adquisición de insumos al proveedor. Con la emisión de este documento, la empresa se compromete a cancelar el monto solicitado y se espera que el proveedor surta los insumos indicados en ella (en teoría, al instante, en la práctica, algún tiempo después una vez procesada la orden). Así, esta sección proporciona las herramientas para la elaboración e impresión de una orden de compra. Editar/Agregar permite al usuario abrir el formulario [Órdenes de Compra] (ver Anexo 1, Figura 26, pág. 42) y configurar la orden, esto es, establecer qué insumos son los solicitados y a qué precios se está ordenando comprarlos (se supone que el usuario conoce y ha negociado estos precios con el proveedor).

La emisión de una orden de compra es la primera acción que modifica directamente el estado del inventario de insumos. Cuando se emite una orden de compra al proveedor, el sistema actualiza la existencia esperada en almacén de los insumos involucrados en la transacción (esto es, el número de artículos del mismo tipo con los que se espera contar en el futuro inmediato). Esta y las demás opciones que afectan directamente al inventario son, en consecuencia, delicadas. La

modificación de sus datos o la eliminación de algún registro altera las cifras de existencia a futuro en el almacén de insumos.

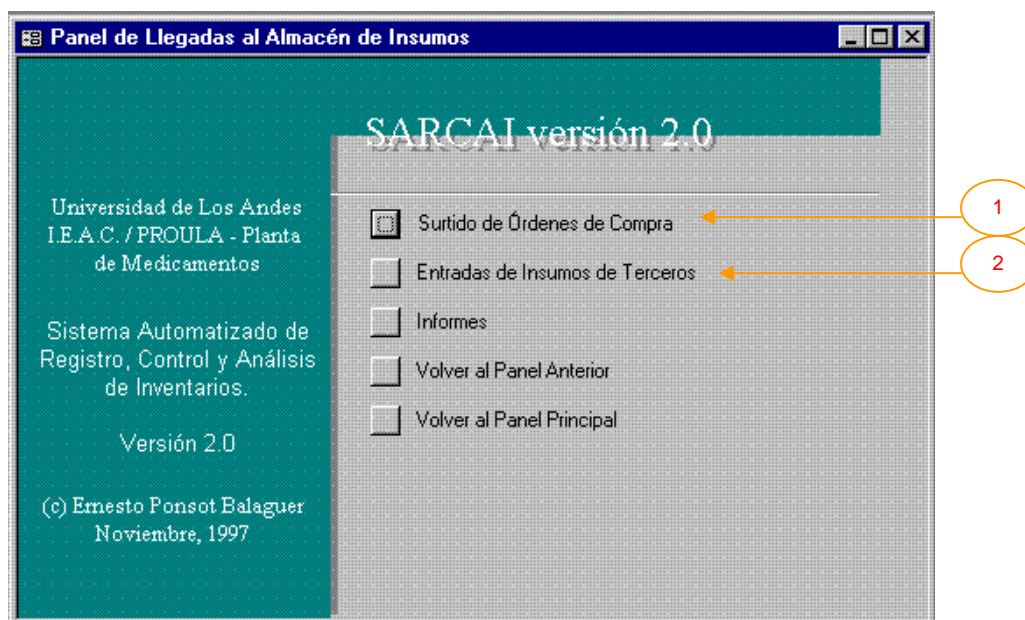
### III.1.3.2.1. Panel de Informes sobre Órdenes de Compra



Este panel presenta las opciones para la obtención de informes relacionados con las órdenes de compra. En este momento:

- 1) Información General de Órdenes de Compra. Abre el informe [Órdenes de Compra] (ver Anexo 2, Punto 38, pág. 46 y Anexo 3, “Orden de Compra”) que presenta un reporte de todos las órdenes emitidas hasta la fecha con sus respectivos datos.

### III.1.3.3. Panel de Llegadas al Almacén de Insumos



En la Planta de Medicamentos, los arribos de insumos para la producción ocurren en este momento por dos vías, dependiendo del tipo de producción en el que participarán. Una vía es la del surtido de órdenes de compra. Si es este el caso, se supone que las entradas al almacén de insumos serán utilizadas en la producción de productos terminados, que luego venderá directamente la organización, con su marca en ellos.

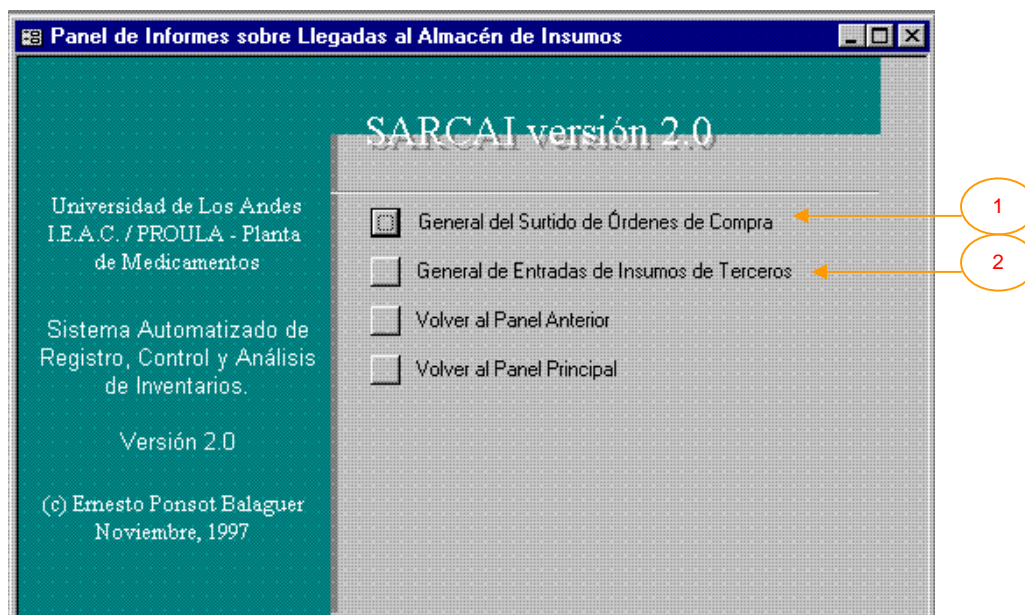
La otra vía es la entrada de insumos de terceras personas. Esta posibilidad se produce cuando el insumo no ha sido ordenado o comprado por la empresa. Es un cliente de la organización quien coloca los insumos en el almacén de la planta para que ésta los procese y los transforme en productos terminados que serán vendidos no por PROULA sino por el cliente, con su propia marca. Como conceptualmente, una vez que ingresan los insumos al almacén, el tratamiento desde el punto de vista del inventario es similar (sea cual sea la vía de ingreso), en esta versión de SARCAI no se han implementado más opciones relacionadas con la entrada de insumos de terceros que las presentes en este panel. En esta decisión también influyó el hecho de que, como estrategia de penetración en el mercado, el óptimo de toda planta productora de medicamentos es

producir para si misma (ya que obtiene valor agregado de los productos en todas las etapas del proceso de comercialización, lo que se traduce en una mayor rentabilidad), por lo tanto, esta actividad, realizada en la actualidad principalmente para mantener activa la planta sin tener que invertir capital en el inventario de insumos (aunque sí en la manutención y cuidado de los mismos), deberá ser muy marginal e incluso desaparecer del panorama de acción de la planta en el futuro, sobre todo una vez superadas las limitantes de flujo de caja para capital de trabajo.

Así pues, en este panel, el usuario encontrará herramientas para registrar todo lo concerniente a la llegada de insumos al almacén. Las opciones novedosas, son:

- 1) Surtido de Órdenes de Compra. Abre el formulario [Llegadas de Insumos al Almacén] (ver Anexo 1, Figuras 22 y 23, pág. 36 y 37) donde es posible registrar las llegadas físicas de insumos al almacén. Además, en vista de que el seguimiento de los insumos (particularmente en el caso de los fármacos), debe ser estricto y continuo a lo largo del proceso, a cada llegada se le asigna un número de lote distinto, aun cuando se trate de un insumo recibido en oportunidades anteriores. Entonces, este formulario no sólo permite registrar una entrada, sino también controlar lo que ocurre con el lote que ha entrado, pudiéndose incorporar información sobre las salidas por causas diversas (como daño o robo) relacionadas con cada lote.
- 2) Entradas de Insumos de Terceros. Abre el formulario [Entradas de Insumos 3ros] (ver Anexo 1, Figuras 10 y 11, pág. 16 y 17), que permite registrar insumos enviados por los clientes para la producción “por encargo”. No se encontrarán en esta versión del sistema más opciones que soporten esta actividad por las razones antes expuestas.

### III.1.3.3.1. Panel de Informes sobre Llegadas al Almacén de Insumos

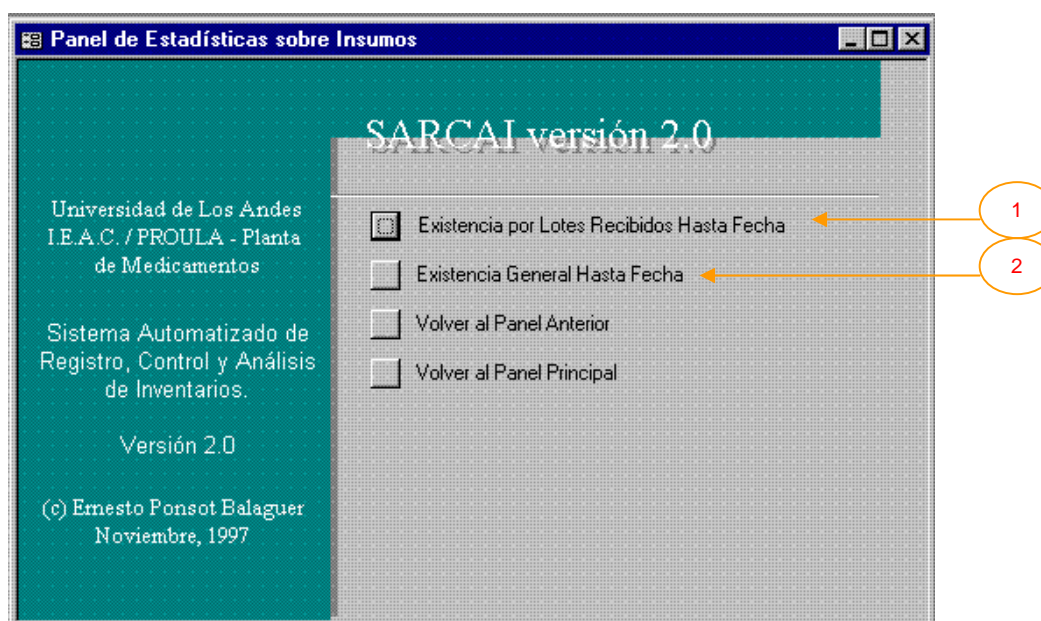


En este panel se presentan dos informes relacionados con las llegadas:

- 1) General del Surtido de Órdenes de Compra. Abre el informe [Llegadas de Insumos al Almacén] (ver Anexo 2, Punto 34, pág. 27 y Anexo 3, “Llegadas de Insumos al Almacén”) que presenta todos los registros de llegadas de insumos, con sus datos, hasta la fecha.
- 2) General de Entradas de Insumos de Terceros. Abre el informe [Entradas de Insumos 3ros] (ver Anexo 2, Punto 15, pág. 14 y Anexo 3, “Entradas de Insumos de Terceros”) que presenta todos los registros de insumos de terceros recibidos hasta la fecha.



### III.1.3.4. Panel de Estadísticas sobre Insumos



En el caso del almacén de insumos se presentan dos estadísticas fundamentales, actualizadas automáticamente con los datos presentes para el momento, en la base de datos. Estas estadísticas tienen relación con la existencia o disponibilidad de los insumos en la planta. En futuras versiones del sistema podrán incorporarse muchas otras opciones a este panel. Lo novedoso en este panel, es:

- 1) Existencia por Lotes Recibidos Hasta Fecha. Esta opción realiza dos acciones. En primer lugar se invoca el formulario [Solicitud de Fechas] (ver Anexo 1, Figura 33, pág. 61) con la finalidad de que el usuario introduzca la fecha (en el formato día / mes / año) hasta la cual desea se compute la existencia por lotes. A continuación se abre el informe [Existencia Real en Almacén I] (ver Anexo 2, Punto 17, pág. 15 y Anexo 3, “Existencias Por Lotes Recibidos en Almacén de Insumos”) que muestra todos los lotes y cuál era, o es, la cantidad que existía, o existe en la fecha deseada. Para el cómputo de la existencia en determinada fecha se toman en cuenta las llegadas al almacén de insumos, las salidas del mismo y el consumo realizado en el proceso productivo. Esta estadística es realmente flexible, pues no sólo permite conocer la existencia de un lote en la actualidad, sino que

virtualmente se puede conocer cuál era la existencia en cualquier instante del pasado, con lo que podría construirse la serie de tiempo de la variable existencia del lote.

- 2) Existencia General Hasta Fecha. Esta opción, al igual que la anterior, realiza dos acciones. Nuevamente se invoca el formulario [Solicitud de Fechas] (ver Anexo 1, Figura 33, pág. 61) y a continuación se abre el informe [Existencia Total Real en Almacén I] (ver Anexo 2, Punto 19, pág. 17 y Anexo 3, “Existencias en Almacén de Insumos”) que muestra el estado, en el instante del tiempo deseado, de todos los insumos (ahora sin considerarlos discriminados en lotes). Como la anterior, esta estadística es flexible, pues permite conocer cuál era la existencia de un insumo en cualquier instante del pasado, con lo que podría construirse la serie de tiempo de la variable existencia del insumo.

### III.1.4. Panel de Producción



El corazón de toda empresa manufacturera es la producción. Esta actividad puede ser dissociada, debido a la complejidad que representa, del tratamiento de inventarios (podría manejarse como una caja negra), sin embargo, no puede ser olvidada, ya que es la razón de ser del inventario de insumos y la fuente de entradas al almacén de productos terminados. En SARCAI se le ha dado

un espacio para hacer más versátil el manejo de los inventarios; esto quiere decir que se soportan algunas acciones (no todas) relacionadas con la producción.

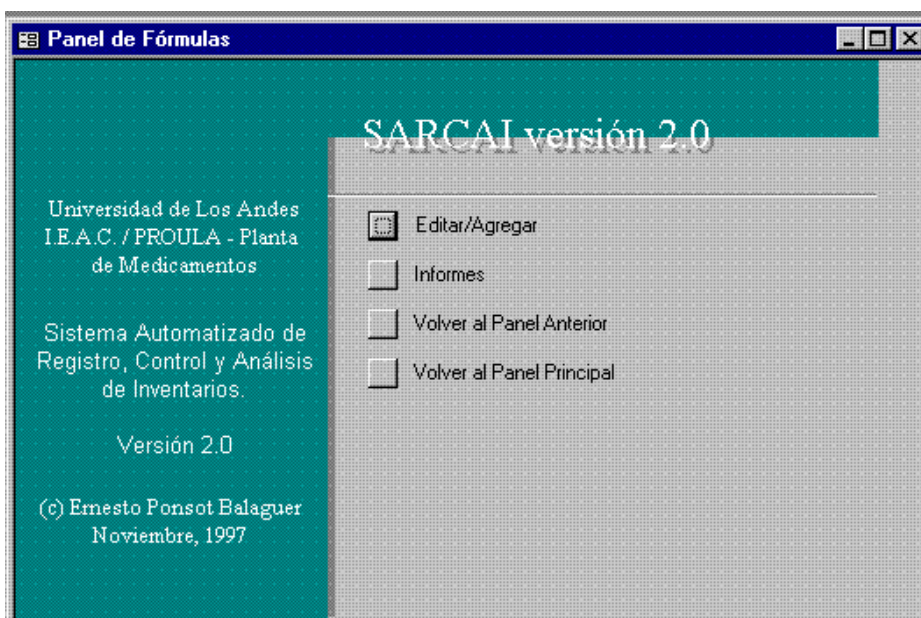
La planta de Medicamentos opera sus distintos procesos productivos, partiendo de una fórmula. La fórmula es la especificación de características elementales y necesidad de insumos para la elaboración de un producto (en este caso, un medicamento general). De una misma fórmula pueden eventualmente obtenerse varios productos terminados, que se diferencian en el destino previsto (venta, muestra médica, etc.). Un producto terminado es entonces la aplicación de una fórmula en la materialización de un elemento que será objeto de transacción. La producción parte de la impresión de una orden de producción, un documento donde se especifica lo que se quiere producir, la cantidad, cuándo debe comenzarse la producción y los insumos necesarios. Normalmente, la elaboración de los distintos productos obedece un plan de producción, esto es, la determinación a futuro de los productos que deberán ser producidos en las cantidades fijadas y en las fechas previstas.

Lo relativo a la producción afecta automáticamente el estado de los inventarios. Una orden de producción rebaja las existencias reales o esperadas en el almacén de insumos e incrementa las existencias reales o esperadas en el almacén de productos terminados. Las ediciones de una fórmula y un producto terminado no afectan automáticamente el cálculo de las existencias, a menos que varíen las cantidades o se eliminen registros. Estas operaciones, por lo tanto, son en extremo delicadas para el sistema. Los planes de producción pueden ser alterados tanto como se desee, sin afectar en lo más mínimo los inventarios (se espera que si la Planta cumple con el plan, por supuesto serán afectados en el futuro, pero no automáticamente). Una demostración de haber alcanzado un alto grado de control y automatización sería que una vez definido el plan, fuera éste el que controlara la emisión de órdenes de producción sin intervención del usuario, empero los requerimientos para esta etapa futura (que podrá incorporarse fácilmente al sistema), van más allá de los aspectos computacionales y se ubican en el campo de la organización interna, dependiente en este momento en extremo de los recursos monetarios con que se cuenta. Esta idea puede ser llevada a su máxima expresión si una vez logrado el control que permita la emisión automática de órdenes de producción, es posible incluso la emisión automática de órdenes de compra (insumos) en la fecha apropiada.

La gráfica muestra las opciones relacionadas con la producción, donde pueden verse:

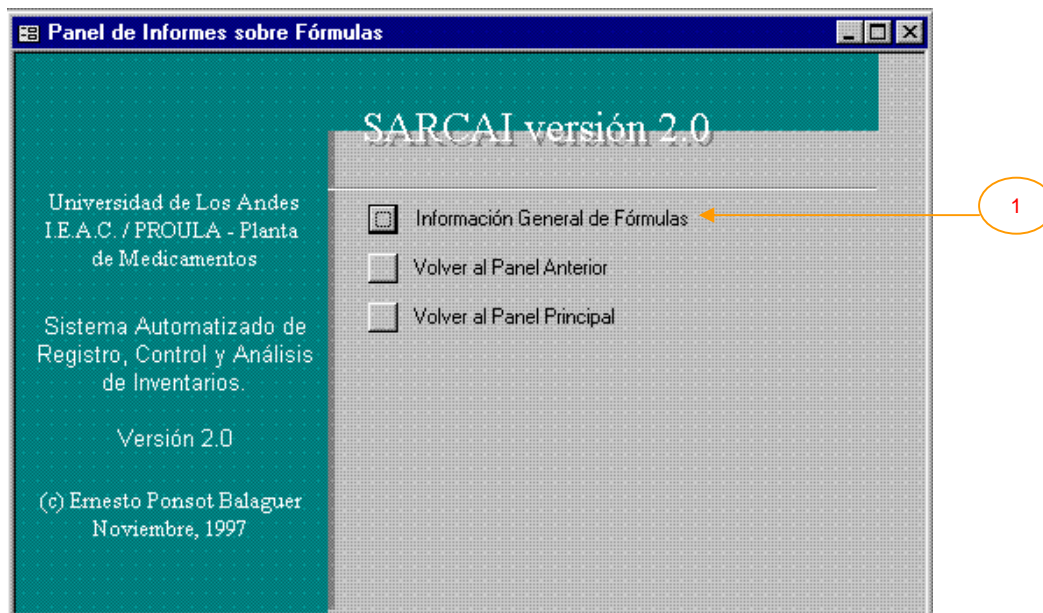
- 1) Fórmulas. Transfiere el control al *Panel de Fórmulas*.
- 2) Productos Terminados. Transfiere el control al *Panel de Productos Terminados*.
- 3) Órdenes de Producción. Transfiere el control al *Panel de Órdenes de Producción*.
- 4) Planes de Producción. Transfiere el control al *Panel de Planes de Producción*.

### III.1.4.1. Panel de Fórmulas



Presenta las opciones (varias veces mencionadas) para la operación con las fórmulas de medicamentos. Editar/Agregar transfiere el control al formulario [Fórmulas] (ver Anexo 1, Figuras 18 y 19, pág. 25 y 26) que permite la introducción de los datos.

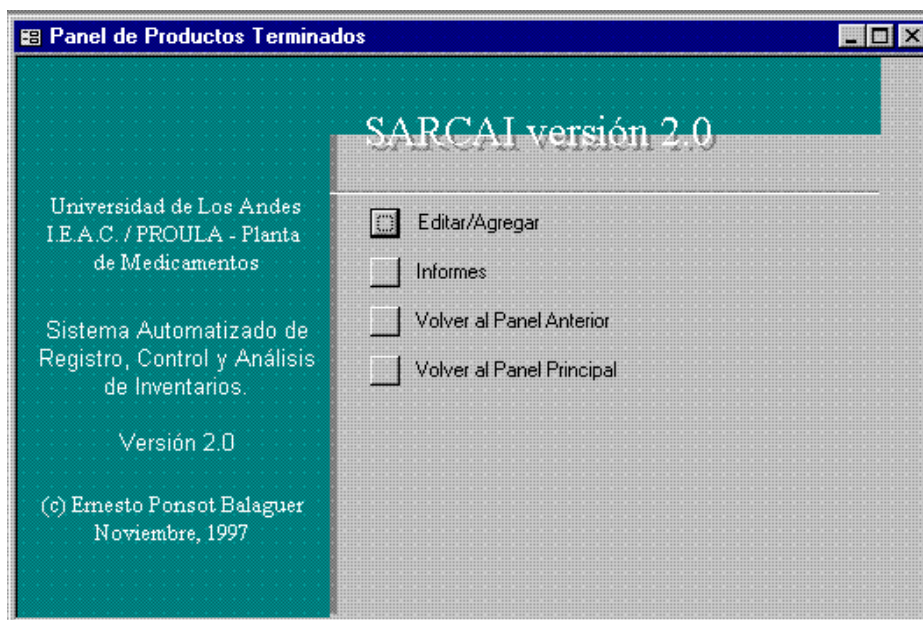
### III.1.4.1.1. Panel de Informes sobre Fórmulas



Este panel presenta las opciones dedicadas a los informes relacionados con las fórmulas. En este momento:

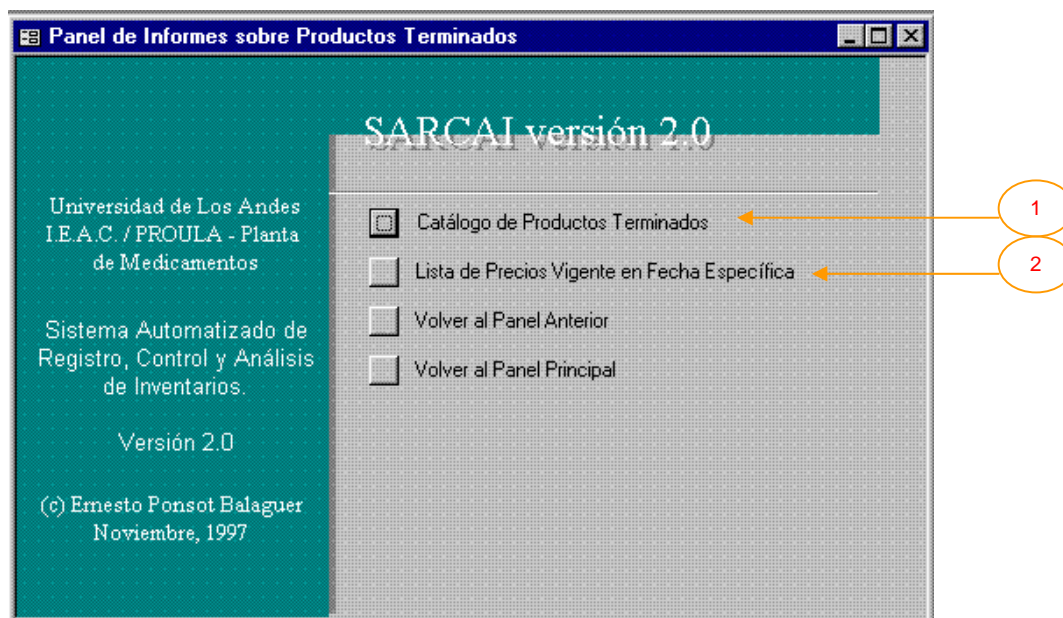
- 1) Información General de Fórmulas. Abre el informe [Fórmulas] (ver Anexo 2, Punto 23, pág. 22 y Anexo 3, “Fórmula:”) que lista todos los datos de cada fórmula.

### III.1.4.2. Panel de Productos Terminados



Este panel representa la sección de productos terminados. En este caso, Editar/Agregar transfiere el control al formulario [Productos Terminados] (ver Anexo 1, Figura 42, pág. 82) que permite la definición de un nuevo producto transable o la modificación de los datos de uno ya definido.

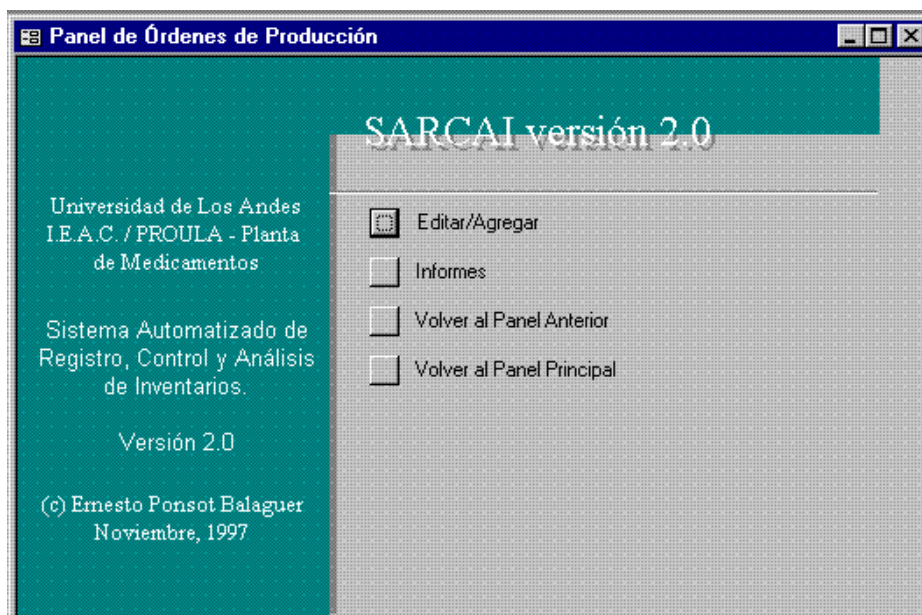
### III.1.4.2.1. Panel de Informes sobre Productos Terminados



Este panel presenta las opciones dedicadas a los informes relacionados con los productos que fabrica la Planta de Medicamentos. En este momento:

- 1) Catálogo de Productos Terminados. Esta opción muestra el informe [Productos Terminados] (ver Anexo 2, Punto 51, pág. 59 y Anexo 3, “Producto Terminado”), el cual lista todos los datos de cada producto terminado definido en la base de datos.
- 2) Lista de Precios Vigente en Fecha Específica. Esta opción realiza dos acciones. Se invoca el formulario [Solicitud de Fechas] (ver Anexo 1, Figura 33, pág. 61) y a continuación se abre el informe [Lista de Precios Vigente en Fecha Buscada] (ver Anexo 2, Punto 32, pág. 26 y Anexo 3, “Lista de Precios Vigente El:”) que muestra, para cada producto, los precios vigentes en la fecha que haya introducido el usuario.

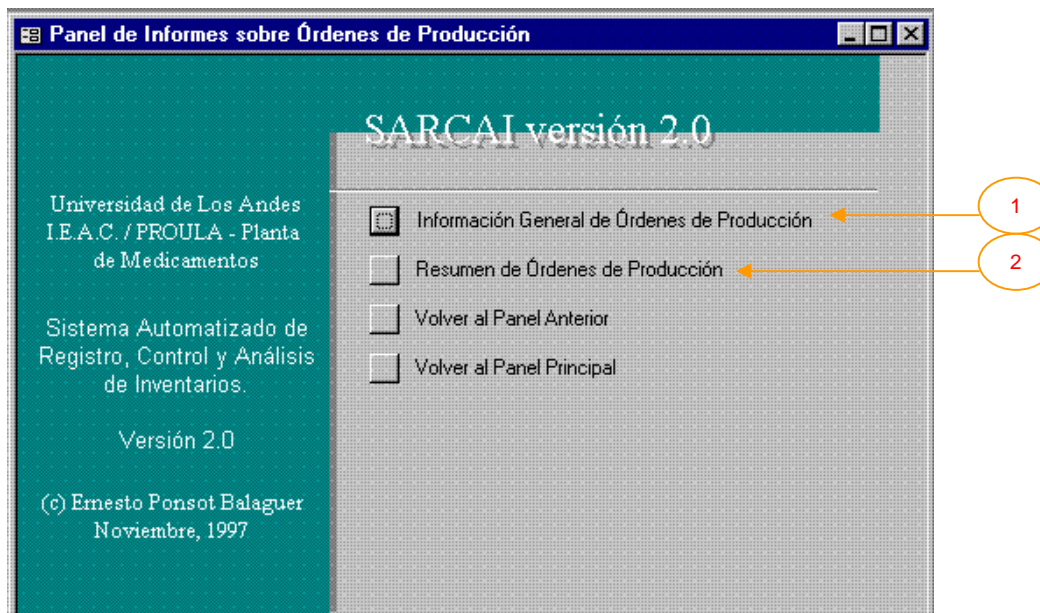
### III.1.4.3. Panel de Órdenes de Producción



Este panel muestra las opciones relativas a las órdenes de producción. En este caso, Editar/Agregar transfiere el control al formulario [Órdenes de Producción] (ver Anexo 1, Figuras 27 y 28, pág. 45 y 46) que permite la diagramación de una nueva orden o la modificación de los datos de una ya definida.



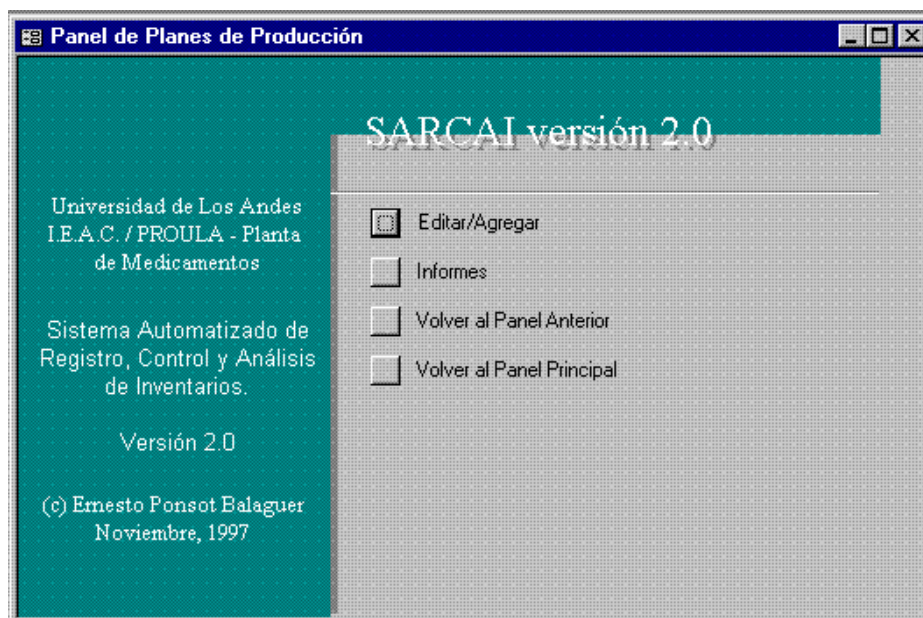
### III.1.4.3.1. Panel de Informes sobre Órdenes de Producción



Este panel presenta las opciones dedicadas a los informes relacionados con las órdenes de producción. En esta versión:

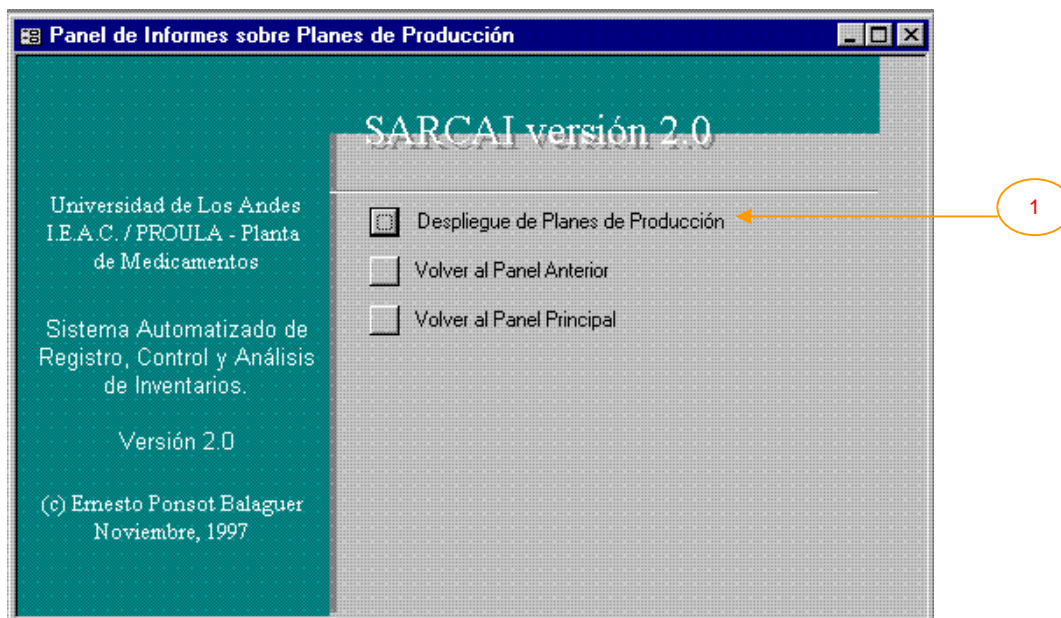
- 1) Información General de Órdenes de Producción. Esta opción muestra el informe [Órdenes de Producción] (ver Anexo 2, Punto 39, pág. 47 y Anexo 3, “Orden de Producción”), el cual lista todos los datos de cada una de las órdenes contenidas en la base de datos.
- 2) Resumen de Órdenes de Producción. Esta opción abre el informe [Resumen de Órdenes de Producción] (ver Anexo 2, Punto 58, pág. 60 y Anexo 3, “Resumen de Órdenes de Producción”) que oculta parte de la información de la orden para obtener una lista más concisa.

### III.1.4.4. Panel de Planes de Producción



Este panel muestra las opciones relativas a los planes de producción. Editar/Agregar transfiere el control al formulario [Planes de Producción] (ver Anexo 1, Figura 39, pág. 79) que permite la configuración de un nuevo plan o la modificación de los existentes.

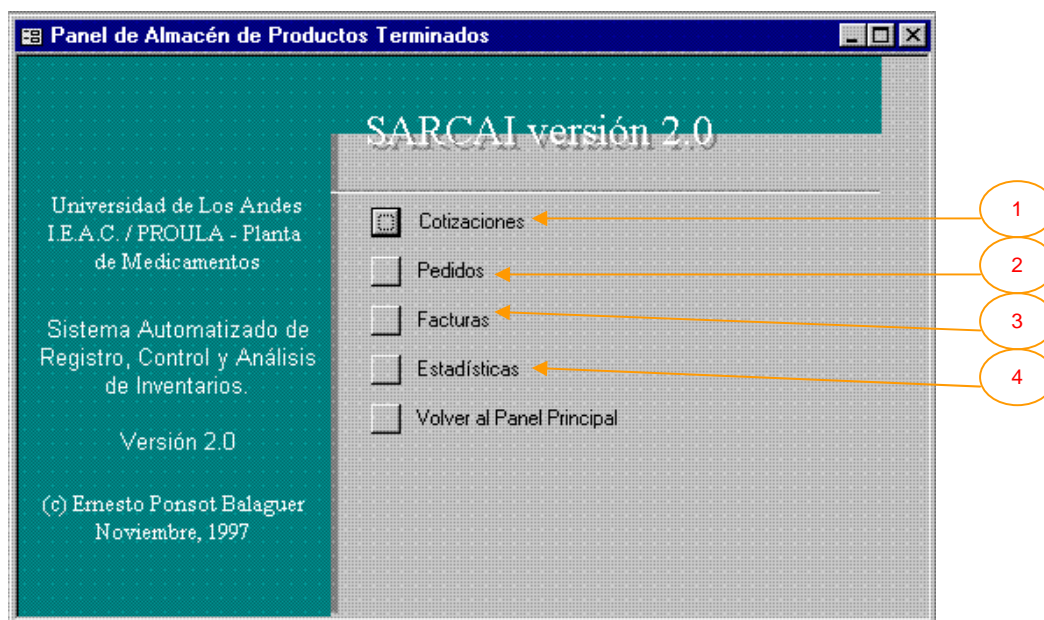
### III.1.4.4.1. Panel de Informes sobre Planes de Producción



Presenta las opciones dedicadas a los informes relacionados con planes de producción. En esta versión:

- 1) Despliegue de Planes de Producción. Esta opción realiza dos acciones. Se invoca el formulario [Pedir Número del Plan] (ver Anexo 1, Figura 37, pág. 74), que solicita un número de plan y permite escoger varios informes. A continuación, y dependiendo de las opciones seleccionadas por el usuario, es posible abrir los informes [Planes de Producción Mensuales], [Planes de Producción Globales 1] y [Explosión de Materiales] (ver respectivamente Anexo 2, Punto 45, pág. 49; Punto 44, pág. 49; Punto 21, pág. 20 y Anexo 3, “Plan de Producción por Mes”, “Plan de Producción Global” y “Requerimientos de Materia Prima e Insumos”) que muestran, el primero, las cantidades a producir en el plan, agrupadas por mes, el segundo, los datos generales del plan, y el tercero, los insumos necesarios para el plan.

### III.1.5. Panel de Almacén de Productos Terminados



La culminación de cualquiera de los procesos productivos es un producto terminado. Ahora bien, cuando se emprende una “corrida” de producción, normalmente se obtiene un número elevado de productos terminados del mismo tipo (por razones de costos, debe aprovecharse al máximo la capacidad productiva de la empresa). Esto crea la necesidad de almacenar, en principio, todo el resultante de la producción. En la Planta de Medicamentos, el lugar donde se almacenan estos bienes se denomina almacén de productos terminados. El sistema provee varias opciones para el control de este almacén.

Una cotización es un documento que no compromete realmente las existencias en el almacén de productos terminados, sin embargo toma información de aquél para hilvanar una oferta coherente que será presentada al cliente, indicando los precios de los productos al momento de su elaboración. En la actualidad, no todas las transacciones de ventas son precedidas por una cotización, por lo tanto, la mayor parte de ellas se elaboran a título informativo. Su inclusión en el sistema pretende contribuir a simplificar la emisión del documento sin concederle poderes reales sobre el inventario, no obstante, podría pensarse en medir la labor de ventas, por ejemplo,

calculando qué porcentaje de las cotizaciones y, dentro de éstas, de los productos cotizados, se convierten efectivamente en pedidos, pero esto es materia del futuro. Por otra parte, un pedido, podría decirse, es el principal documento del sistema. Representa el combustible que mueve los engranajes productivos y administrativos. Paradójicamente, el origen de un pedido se encuentra fuera de las instalaciones de la empresa, así que por ahora (esto puede cambiar en el futuro), la sección de pedidos del sistema se dedica no a la elaboración del pedido, sino a la transcripción de los datos del mismo, entregados por el vendedor, quien a su vez los recibe del cliente. El pedido es entonces una lista de productos terminados que el cliente se compromete a comprarle a la empresa. Afecta directamente las existencias esperadas en el almacén de productos terminados, en cuanto que compromete (al menos moralmente) a la Planta como suplidora del cliente. Las facturas son documentos donde se incluyen los productos que se espera entregar al cliente en venta, los precios a los que se venden y los montos totales que deben ser cancelados. La simple emisión de la factura, no compromete directamente el inventario; lo compromete, ahora en las existencias reales, la inclusión de una fecha de despacho asociada con la factura.

Debe mencionarse que pedidos y facturas tratan cualitativamente aspectos distintos del mercadeo. Un pedido representa en realidad una venta, entendida como el logro de un vendedor que involucra los recursos del cliente. Una factura, que está en el sistema indivisiblemente asociada a un pedido, representa la concreción de la transacción en un hecho de intercambio que puede resultar, por razones imputables a la empresa, en cantidades inferiores (o incrementos de costos) a las del pedido. En consecuencia, la información de pedidos y facturas es en realidad pertinente para medir el funcionamiento del sistema (una relación deseable podría ser facturar, todo lo que se pide). El apoyo a esta labor de análisis se implementa en la opción Estadísticas.

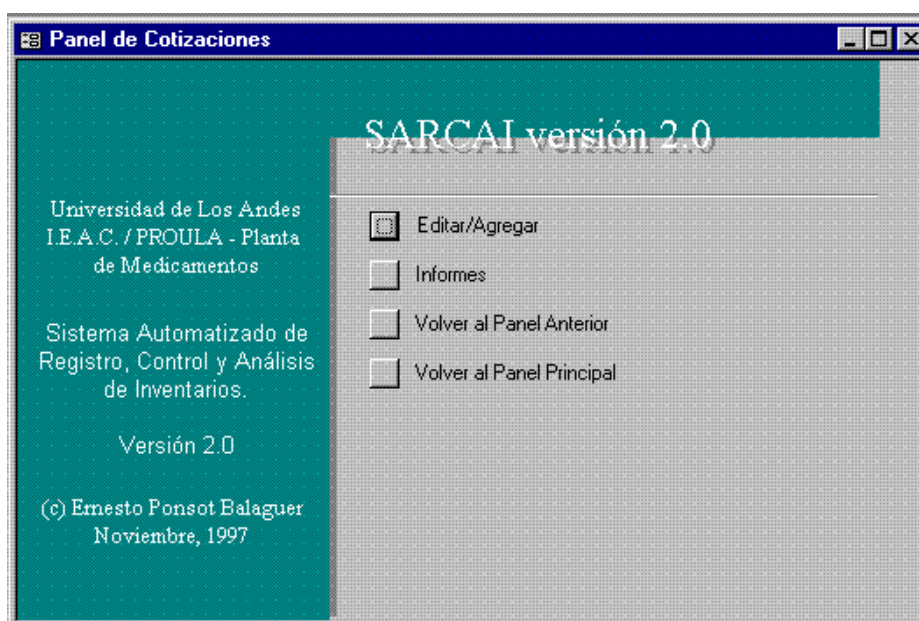
Debe mencionarse también que la demanda generada (pedidos) y la demanda satisfecha (facturas) de productos terminados implica (está positiva y completamente correlacionada con) la demanda de insumos, ya que los procesos productivos de la Planta de Medicamentos son todos del tipo que transforman los insumos (con mínima pérdida) en productos terminados. Por lo tanto, podría decirse que cuando se demanda un producto, en realidad lo que se está demandando es un conjunto de insumos en cantidades conocidas. La implicación de esta idea es fundamental: ejercer un control efectivo, tratando de adelantarse a la demanda de productos terminados, determina automáticamente un control efectivo sobre la demanda de insumos. Por esta razón, las

opciones (incluidas en Estadísticas Generales o Modelos) se dedican al almacén de productos terminados y no al almacén de insumos.

La figura muestra las opciones relacionadas con el almacén de productos terminados:

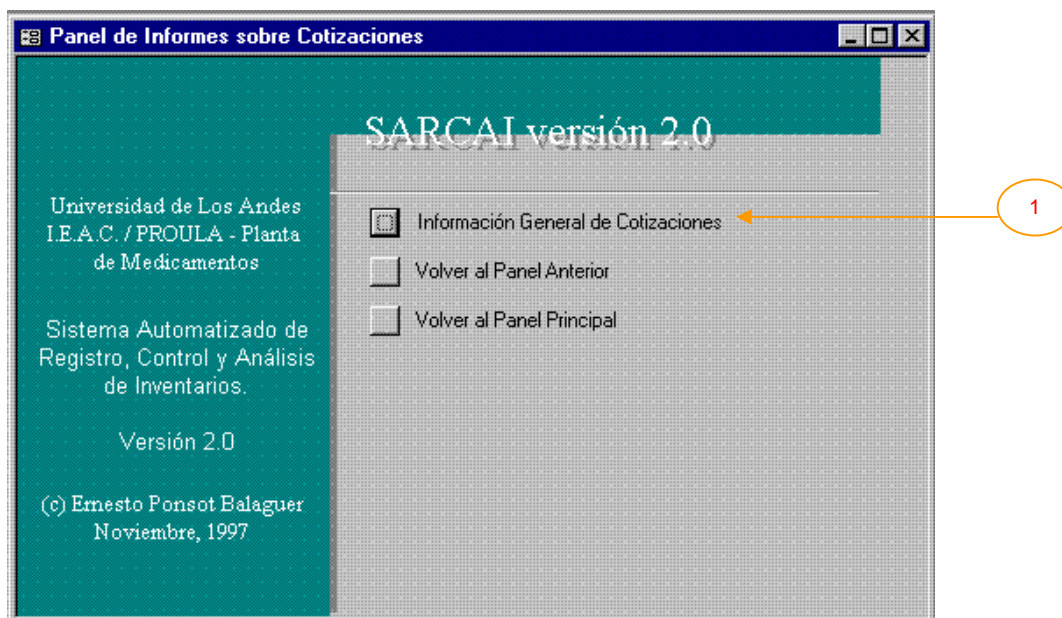
- 1) Cotizaciones. Transfiere el control al *Panel de Cotizaciones*.
- 2) Pedidos. Transfiere el control al *Panel de Pedidos*.
- 3) Facturas. Transfiere el control al *Panel de Facturas*.
- 4) Estadísticas. Transfiere el control al *Panel de Estadísticas*.

### III.1.5.1. Panel de Cotizaciones



Muestra las opciones relativas a las cotizaciones. Editar/Agregar transfiere el control al formulario [Cotizaciones] (ver Anexo 1, Figura 6, pág. 12), donde es posible elaborar una nueva oferta o modificar una existente.

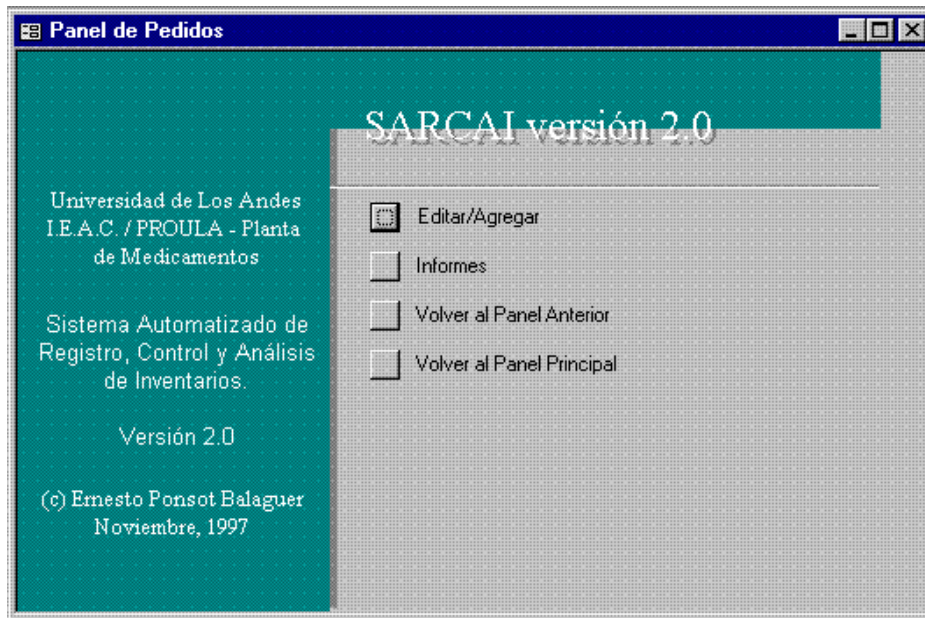
### III.1.5.1.1. Panel de Informes sobre Cotizaciones



Presenta las opciones dedicadas a los informes relacionados con cotizaciones. En esta versión:

- 1) Información General de Cotizaciones. Invoca el formulario [Cotizaciones] (ver Anexo 2, Punto 11, pág. 10 y Anexo 3, “Cotización”), que contiene toda la información registrada de cada cotización.

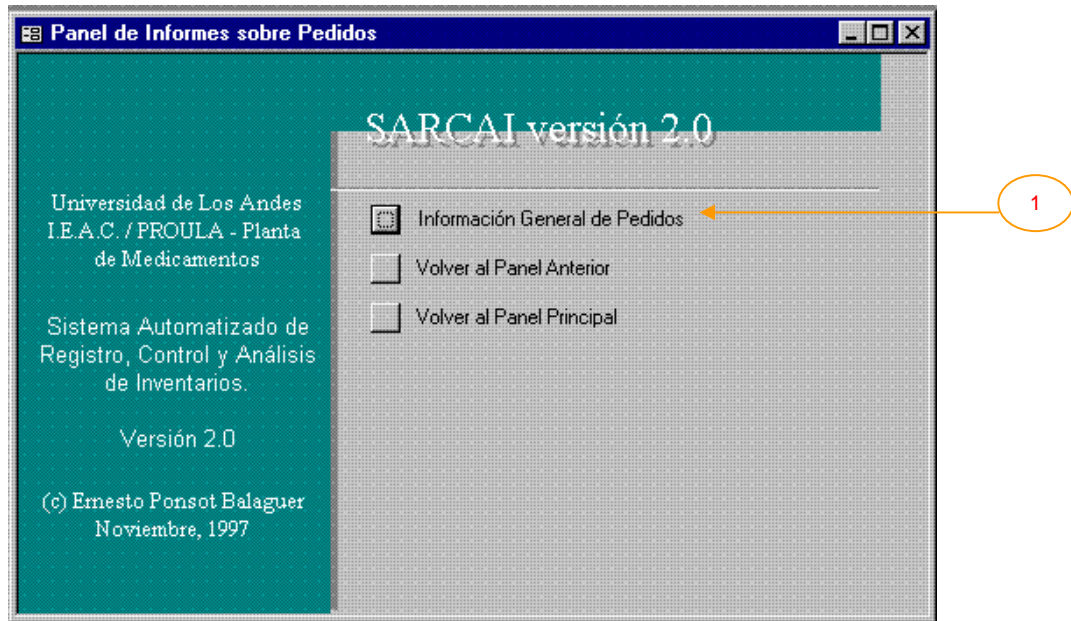
### III.1.5.2. Panel de Pedidos



Muestra las opciones relativas a los pedidos. Editar/Agregar transfiere el control al formulario [Pedidos de Clientes] (ver Anexo 1, Figuras 30 y 31, pág. 52 y 53) donde se transcribe un nuevo pedido o se modifica un existente.



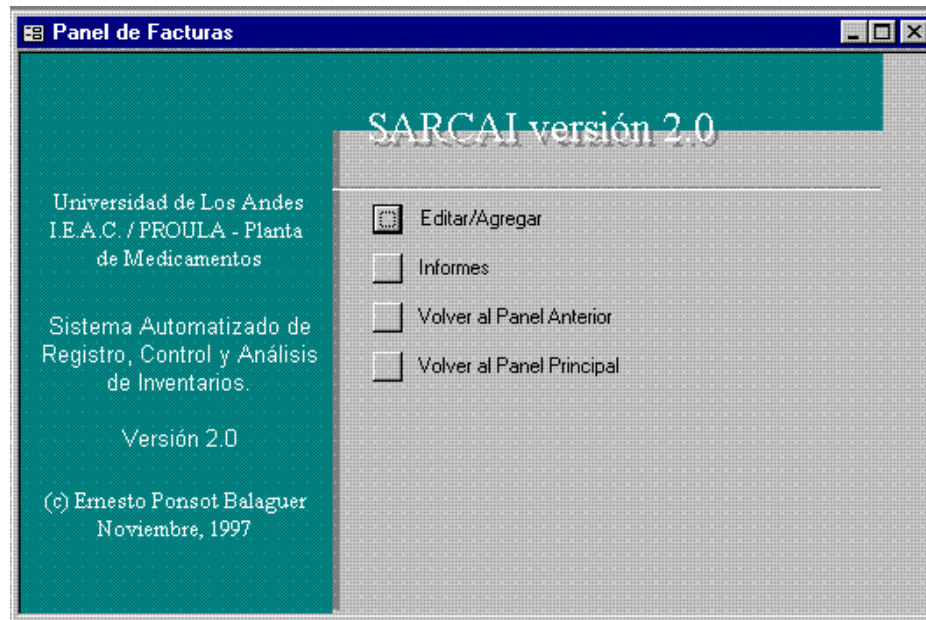
### III.1.5.2.1. Panel de Informes sobre Pedidos



Presenta las opciones dedicadas a los informes de pedidos. En este momento:

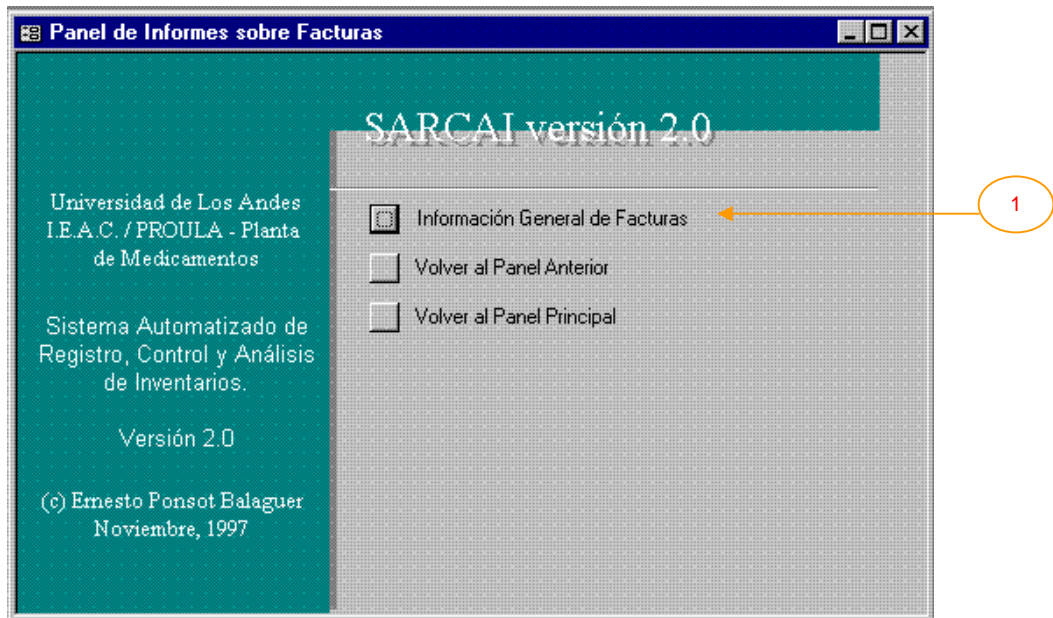
- 1) Información General de Pedidos. Invoca el informe [Pedidos de Clientes] (ver Anexo 2, Punto 41, pág. 48 y Anexo 3, “Pedido del Cliente”) que contiene toda la información de cada pedido.

### III.1.5.3. Panel de Facturas



Muestra las opciones relativas a las facturas. Editar/Agregar transfiere el control al formulario [Facturas] (ver Anexo 1, Figuras 15 y 16, pág. 20 y 21) donde se elabora una nueva factura, a partir de un pedido, o se modifica una existente.

### III.1.5.3.1. Panel de Informes sobre Facturas



Presenta las opciones dedicadas a los informes de facturas. En esta versión:

- 1) Información General de Facturas. Llama al informe [Facturas] (ver Anexo 2, Punto 22, pág. 21 y Anexo 3, “Factura”), que contiene toda la información de cada factura.

### III.1.5.4. Panel de Estadísticas sobre Productos Terminados



En este panel se presentan las opciones que consolidan y resumen información relacionada con las existencias y las ventas en el almacén de productos terminados. En esta versión, puede seleccionarse de entre las siguientes:

- 1) Existencia por Órdenes de Producción Hasta Fecha. Realiza dos acciones, en primer término abre el formulario [Solicitud de Fechas] (ver Anexo 1, Figura 33, pág. 61), donde el usuario debe introducir una fecha que será el límite superior para filtrar los registros del informe resultante, luego, se despliega el informe [Existencia Real en Almacén T] (ver Anexo 2, Punto 18, pág. 16 y Anexo 3, “Existencia Real de Productos Terminados”), que contiene la información de existencias de productos, organizada por órdenes de producción.
- 2) Existencias Generales Hasta Fecha. También realiza dos acciones, primero abre el formulario [Solicitud de Fechas] (ver Anexo 1, Figura 33, pág. 61), luego se despliega el informe [Existencia Total Real en Almacén T] (ver Anexo 2, Punto 20, pág. 18 y Anexo 3,

- “Existencias en Almacén de Productos Terminados”), que contiene la información de existencias de productos, sin importar la orden de producción de la que provengan.
- 3) Estado de Pedidos. Abre el informe [Estado de Pedidos] (ver Anexo 2, Punto 16, pág. 14 y Anexo 3, “Estado de Pedidos”), que presenta las cantidades pedidas y las satisfechas (despachadas) para cada uno de los pedidos. Esto permite visualizar el grado de cumplimiento de los pedidos.
  - 4) Distribución de Pedidos y Ventas. Realiza dos acciones, primero abre el formulario [Pedir Fechas para Ventas] (ver Anexo 1, Figura 35, pág. 68) de forma tal que se pueden introducir dos fechas y escoger el tipo de informe que se desea, luego, considerando sólo los registros cuyas fechas de pedido están entre las dos introducidas antes, se despliegan, dependiendo de la opción elegida, los informes [Clientes, Pedidos y Facturas G], que contiene información (texto y gráficos) sobre pedidos y despachos, organizada por grupos de clientes [Clientes, Pedidos y Facturas S], que contiene información (texto y gráficos) sobre pedidos y despachos, organizada por sectores de clientes, [Clientes, Pedidos y Facturas V], que contiene información (texto y gráficos) sobre pedidos y despachos, organizada por vendedores (ver respectivamente Anexo 2, Punto 5, pág. 4, Punto 6, pág. 6, Punto 7, pág. 8 y Anexo 3, “Pedidos y Ventas por Grupos de Clientes”, “Pedidos y Ventas por Sectores de Clientes” y “Pedidos y Ventas por Vendedor”).
  - 5) Ventas por Productos. Realiza dos acciones. Primero abre el formulario [Pedir Fechas para Ventas por Productos] (ver Anexo 1, Figura 36, pág. 71), de forma tal que se pueden introducir dos fechas y escoger el tipo de informe que se desea, luego, considerando sólo los registros cuyas fechas de pedido están entre las dos introducidas antes, se despliegan, dependiendo de la opción elegida, los informes [Productos Pedidos y Despachados entre Fechas], que contiene información sobre pedidos y despachos organizada por productos, [Productos Pedidos y Despachados entre Fechas D], la misma información organizada por departamentos, [Gráfico de Productos Estrella], que contiene una gráfica de torta que muestra los “mejores” productos (en cuanto a las ventas) para las fechas dadas y [Gráfico de Productos Vendidos Depto], que contiene una gráfica de torta con información porcentual de las ventas en cada departamento (ver respectivamente Anexo 2, Punto 49,

pág. 56, Punto 50, pág. 57, Punto 26, pág. 22, Punto 27, pág. 23 y Anexo 3, “Relación de Ventas de Productos”, “Relación de Ventas de Productos por Departamentos”, “Ventas de Productos Destacados” y “Ventas de Productos por Departamentos”).

- 6) Resúmenes de Ventas. Realiza dos acciones. Primero abre el formulario [Pedir Año para Resumen de Ventas] (ver Anexo 1, Figura 32, pág. 56), que solicita una serie de valores con los cuales será construido cada informe, luego, considerando los datos introducidos, construye y despliega el informe apropiado que en general presenta información consolidada por años, semestres, trimestres y meses, de pedidos y ventas, organizada por productos o departamentos con textos, gráficos o ambos (ver Anexo 3, “Ventas Anuales” como ejemplo, ya que todos tienen la misma estructura). En orden de aparición, los informes presentados son los siguientes:

<b>Informe</b>	<b>Ver en Anexo 2</b>
[Ventas Anuales G]	Punto 72, pág. 65
[Ventas Anuales D]	Punto 71, pág. 64
[Ventas Anuales P]	Punto 73, pág. 66
[Ventas Semestrales G]	Punto 78, pág. 73
[Ventas Semestrales D]	Punto 77, pág. 72
[Ventas Semestrales P]	Punto 79, pág. 74
[Ventas Trimestrales G]	Punto 81, pág. 77
[Ventas Trimestrales D]	Punto 80, pág. 76
[Ventas Trimestrales P]	Punto 82, pág. 78
[Ventas Mensuales G]	Punto 75, pág. 69
[Ventas Mensuales D]	Punto 74, pág. 68
[Ventas Mensuales P]	Punto 76, pág. 70

### III.1.6. Panel de Estadísticas Generales



Este panel se dedica a la presentación de opciones relacionadas con la demanda de productos terminados. Se trata de la sección dedicada a los pronósticos con modelos cuyas dos últimas opciones serán ampliamente discutidas en el siguiente Capítulo. Las opciones son:

- 1) Estudio de la Demanda de Productos Terminados. Realiza dos acciones, en primer término abre el formulario [Pedir Fechas para la Demanda] (ver Anexo 1, Figura 34, pág. 64) donde el usuario puede introducir dos fechas entre las cuales desea listar la demanda, escoger el producto que desea visualizar y seleccionar si prefiere la demanda expresada en unidades o en dinero. Luego se despliega alguno de los informes [Demanda por Fechas] (general), [Demanda por Fechas GU] y [Demanda por Fechas GB], que contiene la información de pedidos de productos, organizada por fechas, formando una serie de tiempo. Esta información puede ser desplegada en forma de texto (como una lista) o en forma gráfica (como un gráfico de líneas en función del tiempo). Si se selecciona la gráfica y más de un producto, el sistema despliega las series de tiempo superpuestas para los productos seleccionados y puede verse entonces el comportamiento de las series en conjunto (ver

respectivamente Anexo 2, Punto 12, pág. 11, Punto 14, pág. 13, Punto 13, pág. 12 y Anexo 3, “Demanda de Productos Terminados por Fechas”). En futuras versiones del sistema, se piensa adicionar a esta sección el modelado de series de tiempo de la demanda (muy probablemente utilizando ARIMA).

- 2) Modelo del Tamaño Económico del Lote. Transfiere el control al formulario [Modelos EOQ], donde se puede configurar un modelo del tipo Tamaño Económico del Lote (que supone demanda constante del producto), para calcular la cantidad fija que deberá ordenarse óptimamente (ver el siguiente Capítulo).
- 3) Modelo de Programación de la Producción. Transfiere el control al formulario [Modelos de Programación], donde se puede configurar un modelo del tipo de Programación Lineal, para la minimización de la función de costos del sistema: producción–demora–inventario, que supone demanda conocida no constante del producto en el período (ver el siguiente Capítulo).

### **III.2. Resumen**

Se han presentado los distintos paneles o ventanas de opciones que componen el sistema. Esta información debe combinarse con la que se presenta en los demás capítulos y anexos, ya que es allí donde se encuentran las características detalladas de los formularios e informes que se invocan desde los paneles. El sistema se compone de 31 paneles, cada uno con por lo menos tres opciones, y tocan principalmente cinco aspectos: Clientes y Proveedores, Almacén de Insumos, Producción, Almacén de Productos Terminados y Estadísticas Generales, siguiendo la lógica del modelo mostrado en el Capítulo I, figura 1.



# CAPÍTULO IV

## Modelos

Este capítulo presenta dos familias de modelos programados en SARCAI v. 2.0: la primera familia de modelos tomando como base la idea del Tamaño Económico del Lote (EOQ) y la segunda, tomado un Modelo de Programación Lineal para la Planificación de la Producción y el Inventario. Se trata de familias de modelos, ya que el usuario tiene la posibilidad de alterar los parámetros de cada uno y obtener así un modelo distinto de tipo similar.

Ambos modelos tienen un objetivo común, aunque utilizan distintas vías para el planteamiento del problema: los modelos EOQ se basan en la deducción de la fórmula del tamaño económico del lote, bajo el supuesto de demanda constante y calculada para obtener la cantidad a reordenar que minimiza el costo total del sistema de inventarios; el modelo de Programación de la Producción utiliza una función objetivo con los costos de escasez, producción e inventario, sometida a distintas restricciones, y resuelve un problema de programación lineal para obtener los valores que minimizan la función de costos (supuesta lineal). Así, el objetivo común es determinar “científicamente” una política óptima para el manejo de los inventarios (en esta versión estudiando un solo producto a la vez, -univariante-).

Ambos modelos son en esencia determinísticos, sin embargo, para la obtención de los datos iniciales de cada uno se ha utilizado la información extraída directamente de la base de datos, la cual ha recibido un tratamiento estadístico sencillo para determinar principalmente la demanda promedio, desviaciones estándares y coeficientes de variación, con lo que se sugieren acciones al usuario.

Los resultados que arrojan estos modelos deberán ser tomados con precaución y validados con datos reales antes de poder tomar decisiones importantes basadas en ellos, ya que se trata de aproximaciones sencillas al problema de inventario que en situaciones complejas, pueden no ser aplicables. Lo que se sugiere es tomar una idea de los modelos sin que ésta sustituya la opinión experta o el sentido común.

El lector debe comprender que la idea integradora que representa la inclusión de estos modelos en el sistema, tomando la información necesaria para su ejecución directamente de la base de datos, se basa en que, a medida que la organización utilice intensivamente y durante un largo período el sistema, éste recaudará más datos, con los cuales las estimaciones de los modelos serán cada vez más cercanas a la realidad (aspecto garantizado por la teoría estadística). Entonces, en las primeras de cambio no se pueden esperar estimaciones muy buenas, sin embargo, a medida que pase el tiempo y se acumulen datos en la base de datos, éstas irán mejorando.

En cualquier caso, sea que se adopte una política sugerida por alguno de los modelos o no, ésta deberá monitorearse cuidadosamente para verificar su aplicabilidad.

En el capítulo se presentarán en orden el modelo teórico utilizado, el formulario que lo representa en el sistema, la descripción del formulario y, por último, el código fuente programado que lo controla. Tal como se ha mencionado antes, los datos son ficticios y se utilizan sólo para ejemplificar.

## IV.1. Modelos EOQ

Esta familia de modelos se crearon y utilizaron con éxito en la primera mitad del siglo XX y, hasta nuestros días, conservan intacta la preferencia de aquellos quienes tienen a su cargo el manejo de los inventarios en las organizaciones. La principal razón por la cual son ampliamente utilizados en el tratamiento del problema de inventarios, estriba en su simplicidad. En efecto, los modelos de esta familia son en esencia aproximaciones sencillas, fáciles de manejar, implantar y comprender, y entregan resultados funcionales para tomar decisiones en una gran proporción de los casos.

El modelo general tratado aquí fue deducido a partir de los siguientes supuestos:

1. El único proveedor del inventario de productos terminados de la planta (que es el inventario controlado) es el proceso productivo de la misma y la tasa de reaprovisionamiento del inventario es infinita. Esto quiere decir que todo lo que se solicita a producción se entrega al inventario en un mismo punto del tiempo. Este supuesto es razonable, ya que la capacidad productiva de la empresa, considerada por lotes, al menos en los rubros que maneja actualmente, es suficientemente grande.
2. No existen precios con descuentos por cantidad. Esto es razonable, pues la empresa siempre procura fabricar el máximo posible de productos tratando de aprovechar su capacidad instalada y reducir, tanto como sea posible, los costos.
3. La tasa a la que se demandan los productos a la empresa es constante en el tiempo. Esto significa que sea cual sea el período que se considere en el horizonte de tiempo, la demanda del producto será la misma, por lo tanto, el nivel del inventario está representado por una línea recta, con pendiente negativa, igual a la demanda.
4. Se acepta que el proveedor puede demorarse en las entregas, pero el tiempo que demora es una constante conocida por la empresa.
5. Se acepta que la empresa puede incurrir en escasez como parte de su política de inventarios.

6. Los costos involucrados, del producto, de ordenar un pedido, de mantener los bienes en el inventario y de escasez de bienes, son constantes conocidas por la empresa en el horizonte de tiempo. El costo de colocar una orden de producción (que en este caso es el reaprovisionamiento) no depende del número de unidades a producir.

### IV.1.1. Modelo Teórico<sup>1</sup>

Las variables que intervienen en este modelo, son:

Variable	Nombre	Descripción
$Q$	Nivel del Inventario	Cantidad del bien en inventario, en cualquier instante del tiempo.
$q$	Tamaño del Lote	Cantidad que debe ser producida, cada vez que se coloque una orden.
$s$	Nivel Máximo del Inventario	Número máximo de artículos que llegará a tener el inventario en el momento del reaprovisionamiento.
$D$	Demanda	Cantidad de unidades demandadas por unidad de tiempo considerado.
$t$	Ciclo del Inventario	Número de unidades de tiempo que transcurren entre dos cualesquiera pedidos consecutivos.
$l$	Retardo	Número de unidades de tiempo que transcurren entre un pedido y la llegada de los bienes al almacén.
$R$	Punto de Reorden	Valor de $Q$ que al ser alcanzado, fuerza el reaprovisionamiento.
$c_1$	Costo Unitario de Mantener	Costo que representa mantener en el inventario una unidad del producto, una unidad de tiempo.
$c_2$	Costo Unitario de Escasez	Costo que representa no tener una unidad del producto, cuando se requiere para ser vendida.
$c_3$	Costo de Ordenar	Costo que representa colocar una orden de producción del producto.
$c_p$	Costo Unitario del Producto	Costo de cada unidad del producto a ser reaprovisionada.
$C$	Costo total	Función de costos del sistema de inventarios.
$T$	Tiempo	Horizonte de planeación.

Poniendo en práctica los supuestos mencionados, las relaciones entre las variables pueden ser vistas en la siguiente figura:

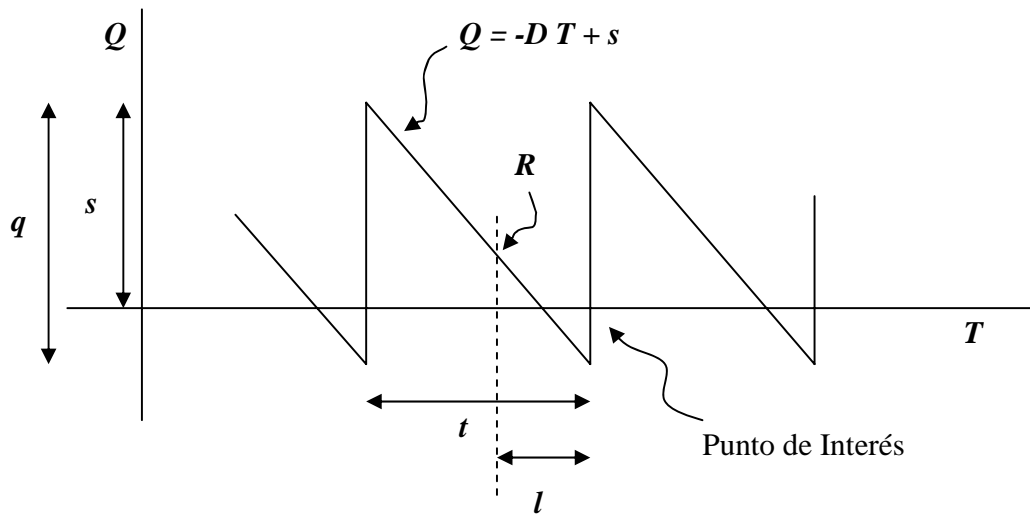


Figura 1. Modelo EOQ con demora y faltantes

Es sencillo (y se puede encontrar en la mayoría de los libros que tratan el tema) justificar que la función de costos totales, por unidad de tiempo, asociada con el problema, es:

$$C(s, q) = \frac{c_1 s^2}{2q} + \frac{c_2 (q - s)^2}{2q} + \frac{Dc_3}{q} + Dc_p$$

y el óptimo se obtiene en el punto:

$$s^* = \sqrt{\frac{2Dc_3}{c_1}} \times \sqrt{\frac{c_2}{c_2 + c_1}}$$

$$q^* = \sqrt{\frac{2Dc_3}{c_1}} \times \sqrt{\frac{c_2 + c_1}{c_2}}$$

Y no depende de  $c_p$ . Además, el ciclo óptimo es:

$$t^* = q^* / D$$

<sup>1</sup> Se utilizaron algunas de las deducciones de HILLIER S. y LIEBERMAN, G. "Introducción a la Investigación de Operaciones. Quinta Edición". McGraw-Hill. México, 1991, p. 687-697, excepto la manera de presentar el Punto de Reorden que surge de las reflexiones del autor.

y la cantidad máxima de faltantes es:

$$q^* - s^*$$

Por otra parte, encontrar el punto de reorden para el punto de interés mostrado en la figura, requiere algunas consideraciones:

1.- Si el retardo  $l$  es cero, claramente el punto de reorden ocurre cuando el nivel del inventario llega a la cantidad máxima de faltantes (ya que el reaprovisionamiento es instantáneo a una tasa infinita), es decir:

$$R = s - q \quad , \quad l = 0$$

2.- Si  $l$  es positiva pero inferior a  $t$ , el reaprovisionamiento debe ocurrir  $l$  unidades de tiempo antes del punto de interés y esto ocurre en el ciclo inmediatamente anterior. El nivel de inventario en este momento es:

$$R = -D \times (t - l) + s \quad , \quad 0 < l < t$$

3.- Si  $l$  es positiva pero superior a  $t$ , reflexionando, se concluye que el reaprovisionamiento debe ocurrir tantos ciclos antes del punto de interés, como veces quepa  $t$  en  $l$ . Sea  $k$  el número de ciclos antes, como en todos los ciclos, la ecuación que representa el nivel del inventario es equivalente, el punto de reorden puede ser calculado en general y aplicado,  $k$  ciclos antes del punto de interés, así:

$$R = -D \times (kt - l) + s \quad , \quad l \geq t$$

$k$  ciclos antes

Con todos estos valores, sólo resta calcular el costo mínimo total y el punto de reorden óptimo. Es obvio que ambas cifras se obtienen sustituyendo  $s$  y  $q$  por sus valores óptimos.

Otro aspecto de suma importancia incorporado al sistema es la **verificación del supuesto de demanda constante**<sup>2</sup>. La idea de suponer que la tasa de demanda es constante, es un tanto arriesgada *a priori*, sin embargo, en este caso, como se tienen los valores puntuales de la demanda, que promediados resultan en la estimación utilizada por el sistema, es posible calcular alguna medida de la variabilidad de la demanda que indique si el supuesto es o no razonable.

La medida seleccionada para esta verificación es el coeficiente de variación (*CV*) de la demanda. Si *CV* es un valor pequeño, el supuesto puede ser válido, si no lo es tanto, el supuesto está en duda, y si es grande, el supuesto debe ser descartado. En el sistema, en primera instancia pero totalmente sujeto a análisis futuros, se ha decidido presentar la conclusión, sobre el supuesto de la demanda, de la siguiente forma:

Aceptable,	$CV \leq 0.25$
Dudoso,	$0.25 < CV \leq 0.75$
Inaceptable,	$CV > 0.75$

Estas son las ecuaciones que se han programado en el sistema para los modelos EOQ. A continuación, el formulario que las implementa y las instrucciones para utilizarlo. Los cálculos pueden verificarse examinando el código Visual Basic que se muestra más adelante.

---

<sup>2</sup> Para un enfoque similar al utilizado aquí, puede verse WAYNE L., Winston. "Investigación de Operaciones. Aplicaciones y Algoritmos". Grupo Editorial Iberoamérica. México, 1994, p'. 885.

### IV.1.2. Formulario para los Modelos EOQ

**Modelos EOQ**

Nº de Modelo:

Producto:

Descripción:

Desde:  Horizonte:  ¿Utilizando Historia?

Demora:  Historia Desde:  Hasta:

Lapso de la Demora:  Lapso de la Demanda:

Demanda en el Período:  **Costos Unitarios**

POR:

Del Producto:  De Ordenar:  De Mantener:  De Escasez:

¿Se permite escasez?  POR:

Registro:  de 3

**Modelos EOQ**

**Resultados**

Tamaño Óptimo del Lote:

Ciclo Óptimo:

Máx. Faltante Permitido:

Máx. Nivel de Inventario:

Punto de Reorden:   Ciclo(s) Antes

Costo del Plan:

De la Demanda: **Revisión del supuesto de DEMANDA**

Desviación Estándar:  **DUDOSO**

Coefficiente de Variación:

Ejecutar el Modelo

Actualizar Punto de Reorden

Elaborar un Plan de Producción

Editar Unidades de Medición

Editar Productos

Editar Planes de Producción

Editar Conversiones

Registro:  de 3



### IV.1.3. Descripción

El [N° de Modelo] es decidido automáticamente por el sistema. El primer paso es entonces la selección del producto, esto se hace tomando de la lista de productos, uno en particular, en el cuadro de lista [Producto].

Opcionalmente, es posible introducir una cadena de texto larga que represente una descripción del plan en el control [Descripción]. A continuación, el cuadro de texto [Desde] acepta una fecha que representa el momento en que se espera dé inicio la programación que propondrá el modelo (seguramente una fecha futura). El cuadro [Horizonte] permite seleccionar de entre la lista de unidades del tiempo el lapso para el que se desean los pronósticos (un año, un bienio, un semestre, etc.), contado a partir de la fecha introducida en el cuadro [Desde].

El campo lógico [¿Utilizando Historia?] se emplea para decidir si la demanda, representada en el formulario por el control [Demanda en el Período], será tomada directamente de la base de datos (cuando está activado) o será un valor introducido por el usuario (cuando está desactivado). Si está activado se habilitan en el formulario los controles [Historia Desde], [Hasta] (que son fechas, la primera necesariamente inferior a la segunda) y el botón [Calcular Demanda Histórica] para que el usuario introduzca los valores deseados. Si se desea que el sistema estime la demanda tomando datos de la base de datos, se deberá introducir estas fechas, los demás parámetros relacionados y presionar el botón [Calcular Demanda Histórica].

El control [Lapso de la Demanda] es de importancia capital para las estimaciones, ya que la [Demanda en el Período] se calcula de forma diferente, dependiendo del lapso introducido. Por ejemplo, si la demanda se expresa en días, el sistema tomará las fechas históricas dadas, contará el número de días, sumará todas las unidades del producto pedidas entre estas dos fechas y dividirá el resultado por el número de días calculado antes, pero si la demanda se expresa en meses se determinará el número de meses que hay entre las fechas señaladas y será este el denominador de la división, y así sucesivamente.

La presión del botón [Calcular Demanda Histórica] asigna un valor calculado al control [Demanda en el Período], sin embargo, el control [Demanda en el Período] puede ser

directamente editado, lo que permite que el usuario no utilice la historia contenida en los archivos, sino que estime por su cuenta un valor para la demanda y lo introduzca. La única limitante que existe sobre esta modalidad es que debe entonces, para ser consecuente, inhabilitar la opción [¿Utilizando Historia?] antes de ejecutar el modelo.

El control [Lapso de la Demanda] le permite al usuario seleccionar los períodos que deben considerarse como unidades, entre las cuales toda transacción debe ser acumulada. Por ejemplo, en la gráfica se están considerando sólo los datos del año 1997 y la unidad básica de cómputo es la semana. Hay que recordar que el modelo EOQ asume demanda constante, por consiguiente, la labor que realiza el sistema es calcular puntos de demanda (en este caso cincuenta y dos, uno para cada semana del año), acumulando los pedidos para cada punto, y luego promedia estos valores para conseguir uno que será tomado como la demanda constante que necesita el modelo. Esta afirmación es contrastada más adelante en la sección [Revisión del Supuesto de Demanda].

Los cuadros de texto [Demora] y [Lapso de la Demora] aceptan respectivamente un número y una unidad de medida, que representan la demora, supuesta constante conocida, del proveedor (en este caso, la Planta de Medicamentos misma) en surtir al almacén lo solicitado. El sistema, a través de la tabla [Conversiones], automáticamente realiza las conversiones en las unidades de tiempo apropiadas para el cómputo, sin embargo, es responsabilidad del usuario garantizar que existen todas las conversiones de unidades necesarias. Si el modelo deseado no contempla demoras, basta con colocar cero en el campo.

Los cuadros de texto “Costos Unitarios” [Del Producto], [De Ordenar] y [De Mantener] son parámetros constantes supuestos conocidos por el usuario y que deben ser introducidos para la función de costos del modelo. El costo unitario del producto es el valor que le asigna la organización a cada producto individual susceptible de ser transado (generalmente se trata del precio al cual se compra el producto en la organización, pero en este caso, tratándose de un proceso productivo, deberá tomarse el costo de producción unitario del producto). El costo de ordenar es en este caso el costo de preparar una orden de producción del producto, normalmente incluyendo papelería, tiempo, maquinaria, etc., y no se espera que dependa del número de productos a producir. El costo de mantener es el costo de tener, durante el período considerado

para la demanda, una unidad del producto en inventario (normalmente incluye espacio físico, papeleo del almacén, personal, etc.).

El campo lógico [¿Se Permite escasez?] indica al sistema si el modelo debe permitir que se incurra en escasez del producto o no. Cuando está activado se muestra el costo unitario [De Escasez], que representa el valor asignado por la organización como costo por no tener una unidad del producto terminado cuando se le solicita en un pedido. Normalmente, este costo es el más difícil de determinar, ya que puede ir desde la simple pérdida de la venta hasta mucho más allá, por ejemplo, el enfado del cliente, quien, en consecuencia, transmite a otros su impresión, lo que restaría en el futuro una cantidad (a veces muy grande) de ventas.

Estos costos deberán ser conocidos o estimados por la organización antes de poder ejecutar el modelo.

Una vez dados los parámetros al modelo, el usuario está listo para recibir la política óptima del modelo. Esto se hace presionando el botón [Ejecutar el Modelo]. El sistema procede a calcular todos los valores resultantes. En el ejemplo:

[Tamaño Óptimo del Lote] ( $q^*$ ): deberán ordenarse cada vez 739 unidades del producto 001.

[Ciclo Óptimo] ( $t^*$ ): deberá ordenarse una corrida de producción cada 2,11 semanas.

[Máx. Faltante Permitido]: hasta 29 unidades pueden estar debiéndose. Si el número parece demasiado grande o si el usuario estima que es exagerado, puede controlar su cuantía aumentando a voluntad el costo unitario de escasez.

[Máx. Nivel de Inventario] ( $s^*$ ): El inventario programado del producto 001 nunca superará (con estos parámetros) las 710 unidades.

[Punto de Reorden] ( $R^*$ ): deberá ordenarse el reaprovisionamiento de  $q^*$  unidades del producto 001, cada vez que el inventario toque las 282 unidades, para reaprovisionar el inventario dentro de 4,5 semanas aproximadamente (dos ciclos antes).

[Costo del Plan] ( $C^*$ ): el plan mostrado cuesta en total 84.210,00 bolívares.

Si es necesario editar los productos, las unidades de medida o las conversiones, se presionan los botones habituales.

Una vez que se han calculado los valores óptimos, el usuario tiene varias opciones. Como es de suponerse, puede imprimir los resultados con los botones de lupa e impresora (ver Anexo 2, Punto 36, pág. 37 y Anexo 2, “Modelo del Tamaño Económico del Lote”). Si está conforme con la solución, puede automáticamente incorporar a la base de datos el punto de reorden  $R$  para el artículo de interés presionando el botón [Actualizar Punto de Reorden], el cual introduce el valor en la tabla [Productos Terminados]. Este punto de reorden se utiliza en los informes de existencia para mostrar al usuario con un asterisco (\*) cuando es urgente el reaprovisionamiento.

Por último, conocidos los valores óptimos, el usuario puede ordenar la elaboración automática de un plan de producción e incorporarlo a la base de datos presionando el botón [Elaborar un Plan de Producción]. Si desea editarlo, puede hacerlo presionando el botón [Editar Planes de Producción]. El sistema está programado para, partiendo de la fecha dada [Desde] y hasta que se cumpla una unidad del [Horizonte], calcular fechas puntuales sumando el [Ciclo Óptimo], colocando en cada una la cantidad [Tamaño Óptimo del Lote] a producir. Esto configura un plan que se incorpora a las tablas [Plan de Producción] y [Productos en Planes], y por consiguiente se tiene en el acto (gracias al informe de Explosión de Materiales) las demandas asociadas de insumos, requeridas para ejecutar el modelo sugerido. De esta forma, el óptimo calculado en el almacén de productos terminados se propaga automáticamente hacia “atrás” al almacén de insumos.

#### IV.1.4. Código Fuente

```
Option Compare Database
Option Explicit

Dim ctlLista As Control

Private Sub ¿Hay_Escasez __AfterUpdate()

    [Tamaño Óptimo del Lote] = 0
    [Ciclo Óptimo] = 0
    [Máximo Nivel] = 0
```

```
[Faltante Óptimo] = 0
[Costo del Plan] = 0
[Reorden] = 0

If [¿Hay Escasez ?] Then
  [Costo U de Escasez ].Visible = True
  [Faltante Óptimo].Visible = True
  [Máximo Nivel].Visible = True
  [Etiqueta67].Visible = True
  [Cuadro combinado66].Visible = True
Else
  [Costo U de Escasez ].Visible = False
  [Faltante Óptimo].Visible = False
  [Máximo Nivel].Visible = False
  [Etiqueta67].Visible = False
  [Cuadro combinado66].Visible = False
End If

End Sub

Private Sub ¿Utilizando_Historia__AfterUpdate()

  If [¿Utilizando Historia?] Then
    [Historia Desde].Enabled = True
    [Historia Hasta].Enabled = True
    [Calcula Demanda].Enabled = True
    [DS de la Demanda].Enabled = True
    [CV de la Demanda].Enabled = True
    [Conclusión].Enabled = True
    [Etiqueta87].Visible = True
  Else
    [Historia Desde].Enabled = False
    [Historia Hasta].Enabled = False
    [Calcula Demanda].Enabled = False
    [DS de la Demanda].Enabled = False
    [CV de la Demanda].Enabled = False
    [Conclusión].Enabled = False
    [Etiqueta87].Visible = False
  End If

End Sub

Public Sub CalculaEstadísticas(F1 As Date, F2 As Date, Inc As Integer)

  Dim dbs As Database, rst As Recordset, F As Date, _
    s As Single, s2 As Single, n As Integer, _
    CV As Single, strSQL As String

  F = F1
  s = 0
  s2 = 0
  n = 0
  Set dbs = CurrentDb

  Do While F <= F2
    n = n + 1
```

```

strSQL = "SELECT Producto, Sum([Demanda Puntual]) AS [Unidades
        Demandas] " & _
        "FROM [Demanda por Fechas Tabla] " & _
        "WHERE ((([Fecha Base]) Between #" & Str(F) & "# And #"
        & IIf(F + Inc - 1 > [Historia Hasta], Str([Historia
        Hasta]), Str(F + Inc - 1)) & "#)) " & _
        "GROUP BY Producto;"
Set rst = dbs.OpenRecordset(strSQL)

If rst.RecordCount <> 0 Then
    rst.MoveFirst
    s = s + rst![Unidades Demandas]
    s2 = s2 + rst![Unidades Demandas] ^ 2
End If

rst.Close
F = F + Inc
Loop

If n > 1 Then
    If s > 0 Then
        [Demanda en el Período] = s / n
        [DS de la Demanda] = Sqr((s2 - s ^ 2 / n) / (n - 1))
        [CV de la Demanda] = n * Sqr((s2 - s ^ 2 / n) / (n - 1)) / s

        If [CV de la Demanda] <= 0.25 Then
            [Conclusión] = "ACEPTABLE"
        Else
            If [CV de la Demanda] <= 0.75 Then
                [Conclusión] = "DUDOSO"
            Else
                [Conclusión] = "INACEPTABLE"
            End If
        End If
    End If

    Else
        [Demanda en el Período] = 0
        [DS de la Demanda] = -1
        [CV de la Demanda] = -1
        [Conclusión] = "DATOS INSUFICIENTES"
    End If
Else
    [Demanda en el Período] = 0
    [DS de la Demanda] = -1
    [CV de la Demanda] = -1
    [Conclusión] = "DATOS INSUFICIENTES"
End If

End Sub

Private Sub Calcula_Demanda_Click()

    Dim strSQL As String, dbs As Database, rst As Recordset

    If [Historia Desde] = Null Or [Historia Hasta] = Null Then
        MsgBox "Entre primero las fechas históricas"
    End If

```

```

Else

If [Lapso de la Demanda] = Null Then
  MsgBox "Entre primero el Lapso de la Demanda"
Else
  DoCmd.SetWarnings False
  strSQL = "DELETE * FROM [Demanda por Fechas Tabla];"
  DoCmd.RunSQL strSQL
  strSQL = "PARAMETERS Forms![Modelos EOQ]![Historia Desde]
            DateTime, Forms![Modelos EOQ]![Historia Hasta]
            DateTime;" & _
  " INSERT INTO [Demanda por Fechas Tabla] ( Producto,
            [Fecha Base], [Demanda Puntual], [Demanda en Bs] )" & _
  " SELECT [Demanda por Fechas 0].Producto, [Demanda
            por Fechas 0].[Fecha Base], [Demanda por Fechas
            0].[Demanda Puntual], [Demanda por Fechas
            0].[Demanda en Bs]" & _
  " FROM [Demanda por Fechas 0] WHERE ((([Demanda por
            Fechas 0].[Fecha Base]) Between [Forms]![Modelos
            EOQ]![Historia Desde] And" & _
  " [Forms]![Modelos EOQ]![Historia Hasta]) And
            ([Producto])=[Forms]![Modelos EOQ]![Código del
            Producto]);"
  DoCmd.RunSQL strSQL

Select Case [Lapso de la Demanda]
  Case "Días"
    CalculaEstadísticas [Historia Desde], [Historia
                        Hasta], 1

  Case "Semanas"
    If NDías([Historia Desde], [Historia Hasta]) < 7 Then
      MsgBox "Las fechas históricas deben incluir al
            menos una semana"
    Else
      CalculaEstadísticas [Historia Desde], [Historia
                        Hasta], 7
    End If

  Case "Quincenas"
    If NDías([Historia Desde], [Historia Hasta]) < 15
      Then
      MsgBox "Las fechas históricas deben incluir al
            menos una quincena"
    Else
      CalculaEstadísticas [Historia Desde], [Historia
                        Hasta], 15
    End If

  Case "Meses"
    If NDías([Historia Desde], [Historia Hasta]) < 30
      Then
      MsgBox "Las fechas históricas deben incluir al
            menos un mes"
    Else
      CalculaEstadísticas [Historia Desde], [Historia
                        Hasta], 30
    End If

```

```
Case "Bimestres"
  If NDías([Historia Desde], [Historia Hasta]) < 60
    Then
      MsgBox "Las fechas históricas deben incluir al
              menos un bimestre"
    Else
      CalculaEstadísticas [Historia Desde], [Historia
                          Hasta], 60
    End If
Case "Trimestres"
  If NDías([Historia Desde], [Historia Hasta]) < 90
    Then
      MsgBox "Las fechas históricas deben incluir al
              menos un trimestre"
    Else
      CalculaEstadísticas [Historia Desde], [Historia
                          Hasta], 90
    End If
Case "Semestres"
  If NDías([Historia Desde], [Historia Hasta]) < 180
    Then
      MsgBox "Las fechas históricas deben incluir al
              menos un semestre"
    Else
      CalculaEstadísticas [Historia Desde], [Historia
                          Hasta], 180
    End If
Case "Años"
  If NDías([Historia Desde], [Historia Hasta]) < 360
    Then
      MsgBox "Las fechas históricas deben incluir al
              menos un año"
    Else
      CalculaEstadísticas [Historia Desde], [Historia
                          Hasta], 360
    End If
Case "Bienios"
  If NDías([Historia Desde], [Historia Hasta]) < 720
    Then
      MsgBox "Las fechas históricas deben incluir al
              menos un bienio"
    Else
      CalculaEstadísticas [Historia Desde], [Historia
                          Hasta], 720
    End If
End Select

DoCmd.SetWarnings True

End If

End If

End Sub

Private Sub Código_del_Producto_AfterUpdate()
```



```

[Costo U Estimado] = DLookup("[Costo Unitario]", "Productos
                             Terminados", "[Código del Producto]='" & [Código
                             del Producto] & "'")

End Sub

Private Sub Editar_Unidades_de_Medición_Exit(Cancel As Integer)

    Set ctlLista = Forms![Modelos EOQ]!Horizonte
    ctlLista.Requery
    Set ctlLista = Forms![Modelos EOQ]![Lapso de la Demora]
    ctlLista.Requery
    Set ctlLista = Forms![Modelos EOQ]![Lapso de la Demanda]
    ctlLista.Requery

End Sub

Private Sub Editar_Productos_Exit(Cancel As Integer)

    Set ctlLista = Forms![Modelos EOQ]![Código del Producto]
    ctlLista.Requery

End Sub

Private Sub Actualizar_Reorden_Click()

    Dim Cadena As String

    Cadena = "Ciudadano... Esta acción cambiará el valor del campo " & _
            "[Punto de Reorden], en la Tabla 'Productos Terminados', " & _
            "actualizándolo a " & Str(Forms![Modelos EOQ]![Reorden]) & _
            " lo que puede afectar la operación actual del sistema."

    If MsgBox(Cadena, vbOKCancel, "Confirmación") = vbOK Then
        Cadena = "UPDATE [Productos Terminados] " & _
                "SET [Punto de Reorden] = " & Str([Reorden]) & _
                " WHERE (([Código del Producto]='" & [Código del
                Producto] & "'));"
        DoCmd.SetWarnings False
        DoCmd.RunSQL Cadena
        DoCmd.SetWarnings True
    End If

End Sub

Private Sub Correr_Click()

    Dim Pond As Single

    Pond = DLookup("[Ponderador]", "Conversiones", "[De]='" & [Lapso de
    la Demora] & "' and [A]='" & [Lapso de la Demanda] & "'")

    If [¿Hay Escasez ?] Then
        [Tamaño Óptimo del Lote] = CLng(Sqr(2 * [Demanda en el Período] *

```

```

        [Costo U de Ordenar] * ([Costo U de Escasez ] + [Costo U
de Mantener]) / ([Costo U de Mantener] * [Costo U de
Escasez ])))
[Ciclo Óptimo] = [Tamaño Óptimo del Lote] / [Demanda en el
Período]
[Máximo Nivel] = CLng(Sqr(2 * [Demanda en el Período] * [Costo U
de Ordenar] * [Costo U de Escasez] / ([Costo U de
Mantener] * ([Costo U de Escasez ] + [Costo U de
Mantener]))))
[Faltante Óptimo] = [Tamaño Óptimo del Lote] - [Máximo Nivel]

If [Demora] > 0 Then

    If [Demora] * Pond < [Ciclo Óptimo] Then
        [Ciclos Antes] = 1
    Else
        [Ciclos Antes]= Int([Demora] * Pond / [Ciclo Óptimo]) + 1
    End If

    [Reorden] = CLng(-[Demanda en el Período] * ([Ciclos Antes] *
[Ciclo Óptimo] - [Demora] * Pond)) + [Máximo Nivel]

Else
    [Reorden] = -[Faltante Óptimo]
    [Ciclos Antes] = 1
End If

[Costo del Plan]=[Demanda en el Período]*[Costo U de Ordenar] /
[Tamaño Óptimo del Lote] + [Demanda en el Período] * [Costo
U Estimado] + [Costo U de Mantener]*[Máximo Nivel]^2 / (2 *
[Tamaño Óptimo del Lote]) + [Costo U de Escasez ] *
[Faltante
Óptimo] ^ 2 / (2 * [Tamaño Óptimo del Lote])
Else
    [Tamaño Óptimo del Lote] = CLng(Sqr(2 * [Demanda en el Período] *
[Costo U de Ordenar] / [Costo U de Mantener]))
[Ciclo Óptimo] = [Tamaño Óptimo del Lote]/[Demanda en el Período]

If [Demora] > 0 Then

    If [Demora] * Pond < [Ciclo Óptimo] Then
        [Ciclos Antes] =1
    Else
        [Ciclos Antes]= Int([Demora] * Pond / [Ciclo Óptimo]) + 1
    End If

    [Reorden] = CLng(-[Demanda en el Período] * ([Ciclos Antes] *
[Ciclo Óptimo] - [Demora] * Pond) + [Tamaño Óptimo del
Lote])

Else
    [Reorden] = 0
    [Ciclos Antes] = 1
End If
[Costo del Plan] = [Demanda en el Período] * [Costo U de Ordenar]
/ [Tamaño Óptimo del Lote] + [Costo U de Mantener] * [Tamaño

```

```

        Óptimo del Lote] / 2 + [Demanda en el Período] * [Costo U
        Estimado]
    End If

End Sub

Private Sub Form_Current()

    If [¿Utilizando Historia?] Then
        [Historia Desde].Enabled = True
        [Historia Hasta].Enabled = True
        [Calcula Demanda].Enabled = True
        [DS de la Demanda].Enabled = True
        [CV de la Demanda].Enabled = True
        [Conclusión].Enabled = True
    Else
        [Historia Desde].Enabled = False
        [Historia Hasta].Enabled = False
        [Calcula Demanda].Enabled = False
        [DS de la Demanda].Enabled = False
        [CV de la Demanda].Enabled = False
        [Conclusión].Enabled = False
    End If

    If [¿Hay Escasez ?] Then
        [Costo U de Escasez ].Visible = True
        [Faltante Óptimo].Visible = True
        [Máximo Nivel].Visible = True
        [Etiqueta67].Visible = True
        [Cuadro combinado66].Visible = True
    Else
        [Costo U de Escasez ].Visible = False
        [Faltante Óptimo].Visible = False
        [Máximo Nivel].Visible = False
        [Etiqueta67].Visible = False
        [Cuadro combinado66].Visible = False
    End If

End Sub

Private Sub Editar_Productos_Click()
On Error GoTo Err_Editar_Productos_Click

    Dim stDocName As String
    Dim stLinkCriteria As String

    stDocName = "Productos Terminados"
    DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria

Exit_Editar_Productos_Click:
    Exit Sub

Err_Editar_Productos_Click:
    MsgBox Err.Description
    Resume Exit_Editar_Productos_Click

```

```

End Sub

Private Sub Plan_Click()
  Dim dbs As Database, tbl As Recordset, F As Date, _
    F1 As Date, F2 As Date, NPlan As Long, NDías As Integer

  Set dbs = CurrentDb
  Set tbl = dbs.OpenRecordset("Planes de Producción", dbOpenTable)
  With tbl
    .AddNew
    !Descripción = "Plan de Producción Generado con Modelos EOQ el" &
      Str(Date) & ". Modelo N° " & Str([N° de Modelo]) &
      ". " & [Descripción]![Fecha de Inicio] = [Desde]
  Select Case [Horizonte]
    Case "Días"
      ![Fecha de Culminación] = [Desde] + 1
    Case "Semanas"
      ![Fecha de Culminación] = [Desde] + 7
    Case "Quincenas"
      ![Fecha de Culminación] = [Desde] + 15
    Case "Meses"
      ![Fecha de Culminación] = [Desde] + 30
    Case "Bimestres"
      ![Fecha de Culminación] = [Desde] + 60
    Case "Trimestres"
      ![Fecha de Culminación] = [Desde] + 90
    Case "Semestres"
      ![Fecha de Culminación] = [Desde] + 120
    Case "Años"
      ![Fecha de Culminación] = [Desde] + 365
    Case "Bienios"
      ![Fecha de Culminación] = [Desde] + 730
  End Select
  .Update
  .MoveLast
  NPlan = ![N° del Plan]
  F1 = ![Fecha de Inicio]
  F2 = ![Fecha de Culminación]
End With
tbl.Close

Set tbl = dbs.OpenRecordset("Productos en Planes", dbOpenTable)
Select Case [Lapso de la Demanda]
  Case "Días"
    NDías = [Ciclo Óptimo]
  Case "Semanas"
    NDías = [Ciclo Óptimo] * 7
  Case "Quincenas"
    NDías = [Ciclo Óptimo] * 15
  Case "Meses"
    NDías = [Ciclo Óptimo] * 30
  Case "Bimestres"
    NDías = [Ciclo Óptimo] * 60
  Case "Trimestres"
    NDías = [Ciclo Óptimo] * 90
  Case "Semestres"

```

```
        NDías = [Ciclo Óptimo] * 180
    Case "Años"
        NDías = [Ciclo Óptimo] * 365
    Case "Bienios"
        NDías = [Ciclo Óptimo] * 730
End Select

With tbl
    F = F1 + NDías
    Do While F <= F2
        .AddNew
        ![N° del Plan] = NPlan
        ![Código del Producto] = [Código del Producto]
        ![Fecha] = F
        ![Cantidad] = [Tamaño Óptimo del Lote]
        .Update
        F = F + NDías
    Loop
End With
tbl.Close

End Sub

Private Sub Editar_Planes_Click()
On Error GoTo Err_Editar_Planes_Click

    Dim stDocName As String
    Dim stLinkCriteria As String

    stDocName = "Planes de Producción"
    DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria

Exit_Editar_Planes_Click:
    Exit Sub

Err_Editar_Planes_Click:
    MsgBox Err.Description
    Resume Exit_Editar_Planes_Click

End Sub

Private Sub Presentar_Click()
On Error GoTo Err_Presentar_Click

    Dim stDocName As String

    stDocName = "Modelos EOQ"
    DoCmd.DoMenuItem acFormBar, acRecordsMenu, 5, , acMenuVer70
    DoCmd.OpenReport stDocName, acPreview, , "[N° de Modelo] = " &
        Str([N° de Modelo])

Exit_Presentar_Click:
    Exit Sub

Err_Presentar_Click:
    MsgBox Err.Description
```

```
Resume Exit_Presentar_Click

End Sub

Private Sub Imprimir_Click()
On Error GoTo Err_Imprimir_Click

Dim stDocName As String
stDocName = "Modelos EOQ"
DoCmd.DoMenuItem acFormBar, acRecordsMenu, 5, , acMenuVer70
DoCmd.OpenReport stDocName, acNormal, "[N° de Modelo] = " &
    Str([N° de Modelo])
Exit_Imprimir_Click:
Exit Sub
Err_Imprimir_Click:
MsgBox Err.Description
Resume Exit_Imprimir_Click
End Sub

Private Sub Editar_Conversiones_Click()
On Error GoTo Err_Editar_Conversiones_Click

Dim stDocName As String
Dim stLinkCriteria As String

stDocName = "Conversiones"
DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria

Exit_Editar_Conversiones_Click:
Exit Sub
Err_Editar_Conversiones_Click:
MsgBox Err.Description
Resume Exit_Editar_Conversiones_Click

End Sub

Private Sub Editar_Unidades_de_Medición_Click()
On Error GoTo Err_Editar_Unidades_de_Medición_Click

Dim stDocName As String
Dim stLinkCriteria As String

stDocName = "Unidades de Medición"
DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria

Exit_Editar_Unidades_de_Medición_Click:
Exit Sub
Err_Editar_Unidades_de_Medición_Click:
MsgBox Err.Description
Resume Exit_Editar_Unidades_de_Medición_Click

End Sub
```

## IV.2. Modelos de Programación

Esta familia de modelos pertenece al tipo de los modelos de Programación Lineal, ampliamente estudiados utilizando el Método Simplex. Un modelo de programación (o programación matemática) es todo aquel que supone la existencia de un conjunto de variables, interactuando en una función que generalmente incluye costos, ganancias o penalizaciones como ponderadores de las variables, y está sujeta a un número finito de restricciones (normalmente dispuestas como ecuaciones o inecuaciones) en las que también intervienen las variables. El objetivo es “optimizar” (maximizar o minimizar según sea el caso) la función descrita, sujeta a las restricciones dadas, para lo cual existen diversos algoritmos.

Un modelo de programación lineal es entonces aquel modelo de programación matemática donde, además de lo dicho, se exige (por la naturaleza del problema) que tanto la función objetivo como cada una de las restricciones sean funciones lineales de las variables.

Así pues, el modelo programado en el sistema es de Programación Lineal, resuelto con el algoritmo Excel–Solver (de la versión 97), cuya función objetivo es la minimización de los costos asociados con la producción, inventario y escasez de un producto en particular (caso univariante). Este modelo hace las siguientes suposiciones:

1. Tanto la función objetivo como las restricciones son funciones lineales de las variables.
2. Se trata de un modelo de múltiples períodos o dinámico.
3. La demanda en cada período es conocida (más no constante) al inicio del mismo.
4. Los costos del producto, de manutención del inventario y de escasez, son conocidos para el primer período y en caso que varíen de un período a otro lo hacen según un porcentaje de inflación, constante conocida para el horizonte de planeación.
5. Sólo se permiten horizontes de planeación de Semestres, Años, Bienios y períodos de demanda en Meses, Bimestres y Trimestres. Entonces se forma una familia de modelos que

incluye un mínimo de 2 períodos (Horizonte de un Semestre con Períodos Trimestrales) y un máximo de 24 períodos (Horizonte de un Bienio con Períodos Mensuales).

### IV.2.1. Modelo Teórico<sup>3</sup>

Las variables que intervienen en este modelo, son:

Variable	Descripción
$k$	Número de Períodos . Cantidad de períodos en que se divide el horizonte de planeación para el cual se ejecuta el modelo. En esta versión, $k = 2, 3, 4, 6, 8, 12, 24$ , según sea la combinación de horizonte y lapso de la demanda.
$x_i$	Número de Unidades a Producir en el $i$ -ésimo Período, $i = 1, \dots, k$ .
$y_i$	Número de Unidades en Inventario al final del $i$ -ésimo Período, $i = 1, \dots, k$ .
$z_i$	Número de Unidades en Escasez en el $i$ -ésimo Período, $i = 1, \dots, k$ . Esto es, unidades cuya entrega al cliente debió ocurrir en el período, más no ocurrió (se demoró a los períodos siguientes).
$d_i$	Número de Unidades Demandadas del Producto en el $i$ -ésimo Período, $i = 1, \dots, k$ .
$y_0$	Nivel del Inventario Inicial del Producto (al comenzar el período 1).
$L$	Límite de la capacidad productiva. Número máximo de unidades del producto que pueden ser producidas en un período cualquiera.
$c_1$	Costo Unitario del Producto.
$c_2$	Costo Unitario de Mantener. Costo que representa mantener en el inventario una unidad del producto de un período al siguiente.
$c_3$	Costo Unitario de Escasez. Costo que representa tener que demorar la entrega de una unidad del producto, de un período al siguiente.
$F$	Porcentaje Anual de Inflación.
$E$	Porcentaje de Exceso (>0) o Defecto (<0) estimado para la demanda. Normalmente es 1.

Las relaciones entre las variables, son entonces:

#### La función objetivo

Sea  $C$  el costo total del sistema, luego

$$C = c_1 \sum_{i=1}^k x_i + c_2 \sum_{i=1}^k y_i + c_3 \sum_{i=1}^k z_i$$

<sup>3</sup> Op. Cit. Wayne, p. 108, 188 y 191



Ecuación básica de costos del sistema. Sin embargo, los costos  $c_1$ ,  $c_2$  y  $c_3$  no son constantes para todos los períodos, sino que se incrementan según el porcentaje de inflación anual  $F$ . Ahora bien, en el entendido de que existen tres posibilidades para seleccionar el período de la demanda, Meses, Bimestres y Trimestres, el porcentaje de inflación calculado para cada caso es respectivamente  $F/12$ ,  $F/6$  y  $F/4$ . Llámese  $F'$  al porcentaje de inflación aplicable según sea el caso; los costos asociados con el sistema, adquieren la forma:

$$c_{m,i} = \left\{ \begin{array}{ll} c_m & , i = 1 \\ c_{m,i-1} + c_{m,i-1} \times F' & , i = 2, \dots, k \end{array} \right\}$$

donde,  $m = 1, 2, 3$  se utiliza respectivamente para el Costo Unitario del Producto en el  $i$ -ésimo período, Costo Unitario de Mantener en el  $i$ -ésimo período y Costo Unitario de Escasez en el  $i$ -ésimo período.

La nueva función de costos totales del sistema, es ahora:

$$C = \sum_{i=1}^k c_{1,i} x_i + \sum_{i=1}^k c_{2,i} y_i + \sum_{i=1}^k c_{3,i} z_i$$

### Las Restricciones

Sea  $N_i$  el nivel del inventario al final de  $i$ -ésimo período. Esto es:

$$\begin{aligned} N_1 &= y_0 + x_1 - d_1 \\ N_2 &= N_1 + x_2 - d_2 \\ &\vdots \\ N_i &= N_{i-1} + x_i - d_i \end{aligned}$$

En palabras, el nivel del inventario al final del primer período, se conforma con el inventario inicial más lo producido en ese período, menos lo demandado. En general, para el final de cualquier período, el nivel del inventario es lo que quedó en el período inmediatamente anterior, más lo producido en el período actual, menos lo demandado.

Es fácil notar que la expresión anterior es recursiva y puede escribirse como:

$$N_i = y_0 + \sum_{j=1}^i x_j - \sum_{j=1}^i d_j$$

Un poco de reflexión basta entonces para encontrar la principal restricción del modelo, cual es:

$$N_i = y_i - z_i \quad , i = 1, \dots, k$$

En efecto, el nivel del inventario en cualquier período debe ser la cantidad a mantener de este período al siguiente, si ha habido existencias suficientes luego de satisfacer la demanda demorada y la demanda actual o debe ser la cantidad que se deje en espera para el período siguiente, en el caso de que no haya alcanzado lo producido ni lo acarreado del período anterior, para satisfacer la demanda. Por supuesto,  $N_i$  es positivo si se ha satisfecho toda la demanda hasta el  $i$ -ésimo período, y negativo en caso contrario. Volviendo a las variables originales del problema, la expresión anterior resulta:

$$y_i + \sum_{j=1}^i x_j - z_i = y_0 - \sum_{j=1}^i d_j \quad , i = 1, \dots, k$$

Obviamente, es necesario agregar restricciones de no negatividad a las variables, de la forma:

$$x_i, y_i, z_i \geq 0 \quad , i = 1, \dots, k$$

Además, para garantizar que en el horizonte de planeación, no quedan faltantes por surtir, es necesario agregar la restricción:

$$z_k = 0$$

Y, dado que hay un límite máximo en la capacidad productiva de cada período, hay que agregar:

$$x_i \leq L \quad , i = 1, \dots, k$$

No se incluyen (aunque sería posible) restricciones a la capacidad de almacenamiento, ya que éstas no existen con los niveles del negocia actuales en la Planta y las restricciones a los faltantes se suponen controladas por el óptimo, en el entendido de que su costo para la empresa es elevado.

Es de hacer notar que el porcentaje de exceso o defecto ( $E$ ) no está incluido en el modelo. Este es un valor que se aplica a la demanda estimada, antes de comenzar los cálculos, de forma constante para cada periodo y tal que se obtiene un nuevo valor de la demanda ( $d_i = d_i + d_i * E$ , para toda  $i = 1, \dots, k$ ) y en consecuencia, cuando se propone el modelo, se supone aplicado previamente y no es necesario contemplarlo.

En conclusión, el problema de Programación de Producción que resuelve el sistema, es el siguiente:

$$\begin{aligned} \min_{\substack{x_i, y_i, z_i \\ i=1, \dots, k}} C &= \sum_{i=1}^k c_{1,i} x_i + \sum_{i=1}^k c_{2,i} y_i + \sum_{i=1}^k c_{3,i} z_i \\ \text{sujeto a :} \\ y_i + \sum_{j=1}^i x_j - z_i &= y_0 - \sum_{j=1}^i d_j \quad , i = 1, \dots, k \\ x_i &\leq L \quad , i = 1, \dots, k \\ z_k &= 0 \\ x_i, y_i, z_i &\geq 0 \quad , i = 1, \dots, k \end{aligned}$$

### Observaciones

La estadística interviene en este modelo cuando se estima automáticamente la demanda, a partir de los datos presentes en el sistema, considerando promedios, máximos, mínimos o combinación de promedios con la desviación estándar estimada, sin embargo, una vez realizada la estimación, la demanda se considera determinada y en consecuencia el modelo es en esencia determinístico.

En la deducción de las ecuaciones se ha supuesto que los períodos de interés parten del número 1 y culminan en el  $k$ -ésimo, sin embargo, en el sistema, el período inicial depende de la fecha en que el usuario desee comenzar la planeación, y si no se trata de un período que esté al comienzo del horizonte de planeación (como sería el mes de enero o el primer bimestre o el primer trimestre), el período inicial puede ser distinto de 1 (lo cual no afecta en lo más mínimo al modelo, es sólo una opción de presentación de resultados).

### IV.2.2. Formulario para los Modelos de Programación

**Modelos de Programación de la Producción**

Producto:   N° de Modelo:

Límite de Producción:  Descripción:

Desde:  Horizonte:  Lapso de la Demanda:

¿Utilizando Historia?

Con la..

Demanda Promedio      Historia Desde:  Historia Hasta:

Demanda Máxima

Demanda Mínima      % de Exceso o Defecto:  Inventario Inicial:

Demanda Promedio +  veces la DS      % Anual de Inflación:

Registro:  de 1

**Modelos de Programación de la Producción**

**Costos Unitarios**

Del Producto:  De Mantener:  POR:

De Escasez:  POR:

**Demanda Estimada**

Período	Demanda	Costos Unitarios			Producir	Mantener	Demorar
		De Producción	De Mantener	De Escasez			
7	1350	Bs.400,00	Bs.20,00	Bs.25,00	1200	0	0
8	750	Bs.413,33	Bs.20,67	Bs.25,83	1200	450	0
9	1245	Bs.427,11	Bs.21,36	Bs.26,69	1200	405	0
10	1958	Bs.441,35	Bs.22,07	Bs.27,58	1200	0	353
11	220	Bs.456,06	Bs.22,80	Bs.28,50	573	0	0
12	158	Bs.471,26	Bs.23,56	Bs.29,45	158	0	0

Registro:  de 6

Registro:  de 1

The screenshot shows a software window titled "Modelos de Programación de la Producción". The main content area is titled "Resultados:" and displays a table of costs. To the left of the table is a button labeled "Actualizar Costos Globales". To the right is a button labeled "Elaborar un Plan de Producción". Below the table are several buttons: "Editar Productos", "Editar Unidades de Medición", "Editar Conversiones", and "Editar Planes de Producción". At the bottom left, there is a "Registro:" field with navigation arrows and the number "1 de 1".

Costos Globales Óptimos		Porcentaje
De Producción	Bs.2.353.932,61	98,84%
De Inventario	Bs.17.949,03	0,75%
De Demanda Diferida	Bs.9.737,26	0,41%
<b>Del Plan Total</b>	<b>Bs.2.381.618,90</b>	<b>100,00%</b>

### IV.2.3. Descripción

Como antes, el [N° de Modelo] es decidido automáticamente por el sistema. El primer paso es entonces la selección del producto, lo cual se hace tomando de la lista de productos uno en particular en el cuadro de lista [Producto]. El sistema, automáticamente, encuentra el producto en la base de datos, extrae el costo unitario estimado (que es un campo de la tabla [Productos Terminados]) y lo coloca en el cuadro de texto Costo Unitario [Del Producto] como valor por defecto.

También es posible introducir una cadena de texto larga que represente una descripción del plan en el control [Descripción]. El control [Límite de Producción] acepta un número que representa la cantidad máxima de unidades del producto que pueden ser producidas en cualquier período (*L*).

El cuadro de texto [Desde] acepta una fecha que representa el momento en que se espera dé inicio la programación que propondrá el modelo (seguramente una fecha futura). El cuadro [Horizonte] permite seleccionar de entre la lista de unidades del tiempo, el lapso para el que se

desean los pronósticos (un año, un bienio o un semestre). El control [Lapso de la Demanda] acepta los valores meses, bimestres o trimestres.

En este caso, la combinación de los valores para los campos [Desde], [Horizonte] y [Lapso de la Demanda] son de importancia primordial, ya que de cada combinación posible surge una estimación de la demanda para el horizonte de planeación diferente. En efecto, la fecha [Desde] fija el período inicial de interés (en el formulario, el 7° mes del año) y como se desea pronosticar para un semestre, de mes en mes, los períodos a considerar serán los meses 7, 8, 9, 10, 11 y 12 del año. Para la estimación de la demanda se agrupará entonces dentro de las fechas históricas dadas para estos meses del año y no para otros (**¡ya que es muy posible que la demanda sea estacional!**).

El campo lógico [¿Utilizando Historia?] se emplea para decidir si la demanda, representada en el formulario por la columna [Demanda], será tomada directamente de la base de datos (cuando está activado) o será un valor introducido por el usuario (cuando está desactivado). Si está activado se habilitan en el formulario el grupo de opciones [Demanda Promedio], [Demanda Máxima], [Demanda Mínima], [Demanda Promedio] +  $n$  [veces la DS], los controles [Historia Desde], [Historia Hasta] (que son fechas, la primera necesariamente inferior a la segunda), los controles [% de Exceso o Defecto], [% Anual de Inflación] y el botón [Estimar la Demanda] para que el usuario introduzca los valores deseados. Si se quiere que el sistema estime la demanda tomando datos de la base de datos, se deberán introducir estas fechas, los demás parámetros relacionados y presionar el botón [Estimar la Demanda].

Del grupo de opciones [Demanda Promedio], [Demanda Máxima], [Demanda Mínima], [Demanda Promedio] +  $n$  [veces la DS] deberá seleccionarse una sola. En base a la selección, se estimará la demanda o bien utilizando el promedio histórico de cada período (entre las fechas dadas), o bien el valor máximo o el mínimo o el promedio más un cierto número de veces la desviación estándar calculada. Estas son las distintas opciones estadísticas con las que cuenta el usuario para la estimación de la demanda, automáticamente partiendo de los datos presentes en la base de datos.

La presión del botón [Estimar la Demanda] asigna un valor calculado a cada campo de las columnas [Período] (con base en el inicio, horizonte y lapso de la demanda), [Demanda] (estimándola como se ha mencionado, incluyendo si ha sido dado el porcentaje de exceso o defecto) y los costos en cada periodo (aplicando, si ha sido dado, el porcentaje de inflación).

El control [Inventario Inicial] recibe la cantidad en inventario que se supone existe, justo antes de comenzar la proyección ( $y_0$ ).

Los últimos parámetros necesarios son entonces los costos unitarios iniciales que deben introducirse en los controles [Del Producto] (colocado por defecto cuando se seleccionó el código del producto, pero editable), [De Mantener] y [De Escasez]. Estos costos deberán ser conocidos o estimados por la organización antes de poder ejecutar el modelo.

Aun cuando las columnas [Período], [Demanda] y [-----Costos Unitarios-----], del subformulario [Demanda Estimada] pueden ser calculadas automáticamente, también son editables, lo que permite al usuario obviar la historia contenida en los archivos (o modificarla) y estimar por su cuenta, los valores. Como antes, la limitante que existe sobre esta modalidad es que debe entonces, para ser consecuente, inhabilitar la opción [¿Utilizando Historia?] antes de ejecutar el modelo.

Una vez dados los parámetros al modelo, el usuario está listo para recibir la política óptima. Esto se hace presionando el botón [Resolver el Modelo]. El sistema procede a calcular todos los valores resultantes llamando para ello al Excel-Solver, pasándole los datos y parámetros y devolviendo el control al sistema. En el ejemplo, la política óptima sugerida es:

Producir 1200 unidades del producto 001, los meses siguientes 7, 8, 9 y 10, producir 563 unidades el mes 11 y 158, el mes 12. Mantener en inventario del mes 8 al 9, 450 unidades del producto y del mes 9 al 10, 405 unidades y demorar la entrega del mes 10 al mes 11, de 353 unidades.

La presión del botón [Actualizar Costos Globales] totaliza los montos asociados con la solución óptima y determina el porcentaje de contribución de cada rubro al costo total.



De nuevo, cuando se hayan calculado los valores óptimos, el usuario tiene varias opciones. Puede imprimir los resultados con los botones de lupa e impresora (ver Anexo 2, Punto 35, pág. 28 y Anexo 3, “Modelo de Programación de la producción”), y conocidos los valores óptimos, puede ordenar la elaboración automática de un plan de producción e incorporarlo a la base de datos presionando el botón [Elaborar un Plan de Producción]. Si desea editarlo, puede hacerlo presionando el botón [Editar Planes de Producción]. El sistema está programado para, partiendo de la fecha dada [Desde] y por el número de períodos calculado, junto con el [Lapso de la Demanda] y el [Horizonte], calcular fechas puntuales colocando la producción óptima sugerida en aquella fecha que corresponda. Esto configura un plan que se incorpora a las tablas [Plan de Producción] y [Productos en Planes] y con los mismos argumentos del modelo anterior (EOQ) se tienen en el acto (gracias al informe de Explosión de Materiales) las demandas asociadas de insumos, requeridas para ejecutar el modelo sugerido. De esta forma, el óptimo calculado en el almacén de productos terminados se propaga automáticamente hacia “atrás” al almacén de insumos.

#### IV.2.4. Ejemplo de la Hoja de Cálculo

	A	B	C
1	Límite de Producción	Inventario Inicial	Nº de Registros
2	1200	150	6

	A	B	C	D	E	F
7	Nº de Modelo	Período	Demanda	Costo de Producción	Costo de Mantener	Costo de Escasez
8	2	7	1350	400,00	20,00	25,00
9	2	8	750	413,33	20,67	25,83
10	2	9	1245	427,11	21,36	26,69
11	2	10	1958	441,35	22,07	27,58
12	2	11	220	456,06	22,80	28,50
13	2	12	158	471,26	23,56	29,45

	G	H	I	J	K	L
7	A Producir	A Mantener	A Demorar	Prod*Costo	Inven*Costo	Esca*Costo
8	1200	0	0	480.000,00	0,00	0,00
9	1200	450	0	495.999,96	9.300,02	0,00
10	1200	405	0	512.533,32	8.649,02	0,00
11	1200	0	353	529.617,72	0,00	9.737,26
12	573	0	0	261.322,27	0,00	0,00
13	158	0	0	74.459,35	0,00	0,00

	M	N
3		Plan 6
4		2.381.618,90
5		Modelo General
6		2381619
7	Restricciones	18
8	0	VERDADERO
9	450	VERDADERO
10	405	VERDADERO
11	-353	VERDADERO
12	0	VERDADERO
13	0	VERDADERO
14		VERDADERO
15		VERDADERO
16		100

**FÓRMULAS**

	<b>J</b>	<b>K</b>	<b>L</b>
<b>7</b>	<b>Prod*Costo</b>	<b>Inven*Costo</b>	<b>Esca*Costo</b>
<b>8</b>	=D8*G8	=E8*H8	=F8*I8
<b>9</b>	=D9*G9	=E9*H9	=F9*I9
<b>10</b>	=D10*G10	=E10*H10	=F10*I10
<b>11</b>	=D11*G11	=E11*H11	=F11*I11
<b>12</b>	=D12*G12	=E12*H12	=F12*I12
<b>13</b>	=D13*G13	=E13*H13	=F13*I13

	<b>M</b>	<b>N</b>
<b>3</b>		<b>Plan 6</b>
<b>4</b>		=SUMA(\$J\$8:\$L\$13)
<b>5</b>		<b>Modelo General</b>
<b>6</b>		=MIN(\$N\$4)
<b>7</b>	<b>Restricciones</b>	=CONTAR(\$G\$8:\$I\$13)
<b>8</b>	=B2+G8-C8	=\$M\$8=Datos!\$H\$8-Datos!\$I\$8
<b>9</b>	=M8+G9-C9	=\$M\$9=Datos!\$H\$9-Datos!\$I\$9
<b>10</b>	=M9+G10-C10	=\$M\$10=Datos!\$H\$10-Datos!\$I\$10
<b>11</b>	=M10+G11-C11	=\$M\$11=Datos!\$H\$11-Datos!\$I\$11
<b>12</b>	=M11+G12-C12	=\$M\$12=Datos!\$H\$12-Datos!\$I\$12
<b>13</b>	=M12+G13-C13	=\$M\$13=Datos!\$H\$13-Datos!\$I\$13
<b>14</b>		=\$I\$13=0
<b>15</b>		=\$G\$8:\$G\$13<=Datos!\$A\$2
<b>16</b>		=IF(100\100\0,000001\0,05\VERDADERO\FALSO\FALSO\1\1\0,001\VERDADERO)

Este ejemplo ha tomado los parámetros del formulario para mostrar el funcionamiento de las hojas de cálculo utilizadas en la ejecución. Existen siete combinaciones distintas, considerando el número de períodos resultante de definir un lapso de la demanda y un horizonte de planeación. Éstas son:

<b>N° de Períodos</b>	<b>Horizonte</b>	<b>Demanda</b>
<b>2</b>	Semestral	Trimestral
<b>3</b>	Semestral	Bimestral
<b>4</b>	Anual	Trimestral
<b>6</b>	Semestral Anual	Mensual Bimestral

N° de Períodos	Horizonte	Demanda
8	Bienio	Trimestral
12	Anual Bienio	Mensual Bimestral
24	Bienio	Mensual

Para cada combinación hay una hoja de cálculo diferente, ya que el número de variables del modelo depende del número de períodos (y también las restricciones). La hoja mostrada opera con 6 períodos y es completamente análoga al resto, las cuales simplemente aumentan o disminuyen el número de filas y las restricciones correspondientes.

Cada hoja de cálculo es actualizada al abrir gracias a la ejecución de una consulta SQL en Excel, que toma los datos directamente de la base de datos ACCESS y los ubica en el área de la hoja.

Otro aspecto que merece ser explicado puede verse en las fórmulas de la hoja en la columna N. En especial, la celda N7 contiene el rango de definición de las variables de decisión; de las celdas N8 a la N15 están definidas las restricciones, y la celda N16 contiene las opciones del Solver (las que permiten cargar el modelo cuando se abre la hoja debido a la macro programada).

## IV.2.5. Código Fuente

### DEL FORMULARIO

```
Option Compare Database
Option Explicit
```

```
Dim ctlLista As Control
```

```
Private Declare Function FindWindow Lib "user32" Alias _
    "FindWindowA" (ByVal lpClassName As String, _
    ByVal lpWindowName As Long) As Long
```

```
Private Declare Function SendMessage Lib "user32" Alias _
    "SendMessageA" (ByVal hWnd As Long, ByVal wParam As Long, _
    ByVal lParam As Long) As Long
```

```

Private Sub ¿Utilizando_Historia__AfterUpdate()

    If [¿Utilizando Historia?] Then
        [Historia Desde].Enabled = True
        [Historia Hasta].Enabled = True
        [Valores de la Demanda].Enabled = True
        [Veces la DS].Enabled = True
        [Etiqueta67].Visible = True
        [% de Exceso o Defecto].Enabled = True
        [% Anual de Inflación].Enabled = True
        [Demanda].Enabled = True
    Else
        [Historia Desde].Enabled = False
        [Historia Hasta].Enabled = False
        [Valores de la Demanda].Enabled = False
        [Veces la DS].Enabled = False
        [Etiqueta67].Visible = False
        [% de Exceso o Defecto].Enabled = False
        [% Anual de Inflación].Enabled = False
        [Demanda].Enabled = False
    End If

End Sub

Private Sub Código_del_Producto_AfterUpdate()

    [Costo U Estimado] = DLookup("[Costo Unitario]", "Productos
    Terminados", "[Código del Producto]='" & [Código del Producto] &
    "'")

End Sub

Private Sub Actualizar_Click()

    CostoGP = DSum("[A Producir]*[Costo de Producción]", "Productos en
    Modelos P", "[N° de Modelo]=" & Str([N° de Modelo]))
    CostoGI = DSum("[A Mantener]*[Costo de Mantener]", "Productos en
    Modelos P", "[N° de Modelo]=" & Str([N° de Modelo]))
    CostoGE = DSum("[A Demorar]*[Costo de Escasez ]", "Productos en Modelos
    P", "[N° de Modelo]=" & Str([N° de Modelo]))
    [Costo del Plan] = CostoGP + CostoGI + CostoGE

    If [Costo del Plan] <> 0 Then
        PorcGP = CostoGP / [Costo del Plan]
        PorcGI = CostoGI / [Costo del Plan]
        PorcGE = CostoGE / [Costo del Plan]
    Else
        PorcGP = 0
        PorcGI = 0
        PorcGE = 0
    End If

End Sub

Private Sub Editar_Unidades_de_Medición_Exit(Cancel As Integer)

```

```

Set ctlLista = Forms![Modelos de Programación]![Horizonte]
ctlLista.Requery
Set ctlLista = Forms![Modelos de Programación]![Lapso de la Demanda]
ctlLista.Requery

```

```
End Sub
```

```
Private Sub Plan_Click()
```

```

Dim dbs As Database, tbl As Recordset, pln As Recordset, F As Date, _
    F1 As Date, F2 As Date, NPlan As Long, NDías As Integer, _
    NIter As Integer, i As Integer, plnF As Recordset

```

```

Set dbs = CurrentDb
Set tbl = dbs.OpenRecordset("Planes de Producción", dbOpenTable)
With tbl

```

```

    .AddNew
    !Descripción = "Plan de Producción Generado con Modelos de
                    Producción el " & _
                    Str(Date) & ". Modelo N° " & Str([N° de Modelo]) &
                    ". " & [Descripción]
    ![Fecha de Inicio] = [Desde]

```

```

Select Case [Horizonte]
    Case "Semestres"
        ![Fecha de Culminación] = [Desde] + 120
    Case "Años"
        ![Fecha de Culminación] = [Desde] + 365
    Case "Bienios"
        ![Fecha de Culminación] = [Desde] + 730
End Select

```

```

    .Update
    .MoveLast
    NPlan = ![N° del Plan]
    F1 = ![Fecha de Inicio]
    F2 = ![Fecha de Culminación]
End With
tbl.Close

```

```

Set tbl = dbs.OpenRecordset("Productos en Planes", dbOpenTable)
Set pln = dbs.OpenRecordset("Productos en Modelos P", dbOpenDynaset)
pln.Filter = "[N° de Modelo] =" & Str([N° de Modelo])
Set plnF = pln.OpenRecordset

```

```

Select Case [Lapso de la Demanda]
    Case "Meses"
        NDías = 30
    Case "Bimestres"
        NDías = 60
    Case "Trimestres"
        NDías = 90
End Select

```

```

F = F1
NIter = DCount("[N° de Modelo]", "Productos en Modelos P", "[N° de

```

```

        Modelo]=" & Str([N° de Modelo])
    plnF.MoveFirst

    For i = 1 To NIter

        If plnF![A Producir] <> 0 Then
            tbl.AddNew
            tbl![N° del Plan] = NPlan
            tbl![Código del Producto] = [Código del Producto]
            tbl![Fecha] = F
            tbl![Cantidad] = plnF![A Producir]
            tbl.Update
        End If

        plnF.MoveNext
        F = F + NDías

    Next i

tbl.Close
pln.Close
plnF.Close

End Sub

Private Sub Demanda_Click()

    Dim strSQL As String, dbs As Database, rst As Recordset, _
        i As Integer, Primero As Integer, Último As Integer, _
        Cad1 As String, Cad2 As String, CostoP As Single, _
        CostoM As Single, CostoE As Single, DivisorI As Single

    If [Historia Desde] = Null Or [Historia Hasta] = Null Then
        MsgBox "Entre primero las fechas históricas"
    Else

        If [Lapso de la Demanda] = Null Then
            MsgBox "Entre primero el Lapso de la Demanda"
        Else
            DoCmd.SetWarnings False
            strSQL = "DELETE * FROM [Demanda por Fechas Tabla];"
            DoCmd.RunSQL strSQL
            strSQL = "DELETE * FROM [Productos en Modelos P] WHERE [N° de
                Modelo]=[Forms![Modelos de Programación]![N° de
                Modelo];"
            DoCmd.RunSQL strSQL
            strSQL = "PARAMETERS Forms![Modelos de
                Programación]![Historia Desde] DateTime,
                Forms![Modelos de Programación]![Historia Hasta]
                DateTime;" & _
                " INSERT INTO [Demanda por Fechas Tabla] ( Producto,
                [Fecha Base], [Demanda Puntual], [Demanda en Bs] )"
                & " SELECT [Demanda por Fechas 0].Producto, [Demanda
                por Fechas 0].[Fecha Base], [Demanda por Fechas
                0].[Demanda Puntual], [Demanda por Fechas
                0].[Demanda en Bs]" & " FROM [Demanda por Fechas 0]

```

```

WHERE ((([Demanda por Fechas 0].[Fecha Base])
Between [Forms]![Modelos de Programación]![Historia
Desde] And" & " [Forms]![Modelos de
Programación]![Historia Hasta]) And
([Producto])=[Forms]![Modelos de
Programación]![Código del Producto]);"
DoCmd.RunSQL strSQL
Forms![Modelos de Programación].Repaint

Select Case [Lapso de la Demanda]
Case "Meses"
Primerero = Month([Desde])
Cad2 = "Demanda Promedio Mensual"
DivisorI = 12
Case "Bimestres"
Primerero = IIf(Month([Desde]) <= 2, 1,
IIf(Month([Desde]) <= 4, 2, IIf(Month([Desde])
<= 6, 3, IIf(Month([Desde]) <= 8, 4,
IIf(Month([Desde]) <= 10, 5, 6))))))
Cad2 = "Demanda Promedio Bimensual"
DivisorI = 6
Case "Trimestres"
Primerero = IIf(Month([Desde]) <= 3, 1,
IIf(Month([Desde]) <= 6, 2, IIf(Month([Desde])
<= 9, 3, 4)))
Cad2 = "Demanda Promedio Trimestral"
DivisorI = 4
End Select

Select Case [Valores de la Demanda]
Case 1
Cad1 = "[Promedio]"
Case 2
Cad1 = "[Máxima]"
Case 3
Cad1 = "[Mínima]"
Case 4
Cad1 = "[Promedio]+[DS]*" & Str([Veces la DS])
End Select

Select Case [Horizonte]

Case "Semestres"

Select Case [Lapso de la Demanda]
Case "Meses"
Último = Primerero + 6 - 1
Case "Bimestres"
Último = Primerero + 3 - 1
Case "Trimestres"
Último = Primerero + 2 - 1
End Select

Case "Años"

Select Case [Lapso de la Demanda]

```



```

    Case "Meses"
        Último = Primero + 12 - 1
    Case "Bimestres"
        Último = Primero + 6 - 1
    Case "Trimestres"
        Último = Primero + 4 - 1
End Select

Case "Bienios"

Select Case [Lapso de la Demanda]
Case "Meses"
    Último = Primero + 24 - 1
Case "Bimestres"
    Último = Primero + 12 - 1
Case "Trimestres"
    Último = Primero + 8 - 1
End Select

End Select

Set dbs = CurrentDb
Set rst = dbs.OpenRecordset("Productos en Modelos P")
With rst
    CostoP = [Costo U Estimado]
    CostoM = [Costo U de Mantener]
    CostoE = [Costo U de Escasez ]

    For i = Primero To Último
        .AddNew
        ![N° de Modelo] = [N° de Modelo]
        ![Período] = i
        ![Demanda] = DLookup(Cad1, Cad2, "[Período]=" &
            Str(i))
        ![Demanda] = ![Demanda] + [% de Exceso o Defecto] *
            ![Demanda]
        ![Costo de Producción] = CostoP
        ![Costo de Mantener] = CostoM
        ![Costo de Escasez ] = CostoE
        CostoP = CostoP + CostoP * [% Anual de Inflación] /
            DivisorI
        CostoM = CostoM + CostoM * [% Anual de Inflación] /
            DivisorI
        CostoE = CostoE + CostoE * [% Anual de Inflación] /
            DivisorI
        .Update
    Next i

End With

rst.Close
[Subformulario Productos en Modelos P].Requery
DoCmd.SetWarnings True

CostoGP = 0
CostoGI = 0

```

```
        CostoGE = 0
        [Costo del Plan] = 0
        PorcGP = 0
        PorcGI = 0
        PorcGE = 0

    End If

End If

End Sub

Private Sub Editar_Productos_Exit(Cancel As Integer)

    Set ctlLista = Forms![Modelos de Programación]![Código del Producto]
    ctlLista.Requery

End Sub

Private Sub Form_Current()

    If [¿Utilizando Historia?] Then
        [Historia Desde].Enabled = True
        [Historia Hasta].Enabled = True
        [Valores de la Demanda].Enabled = True
        [Veces la DS].Enabled = True
        [Etiqueta67].Visible = True
        [% de Exceso o Defecto].Enabled = True
        [% Anual de Inflación].Enabled = True
        [Demanda].Enabled = True
    Else
        [Historia Desde].Enabled = False
        [Historia Hasta].Enabled = False
        [Valores de la Demanda].Enabled = False
        [Veces la DS].Enabled = False
        [Etiqueta67].Visible = False
        [% de Exceso o Defecto].Enabled = False
        [% Anual de Inflación].Enabled = False
        [Demanda].Enabled = False
    End If

End Sub

Private Sub Horizonte_BeforeUpdate(Cancel As Integer)

    If [Horizonte] <> "Años" And [Horizonte] <> "Semestres" And
        [Horizonte] <> "Bienios" Then
        MsgBox "Por favor, seleccione Bienios, Años o Semestres
            únicamente."
        Cancel = 1
    End If

End Sub

Private Sub Lapso_de_la_Demanda_BeforeUpdate(Cancel As Integer)
```

```

If [Lapso de la Demanda] <> "Meses" And [Lapso de la Demanda] <>
  "Bimestres" And [Lapso de la Demanda] <> "Trimestres" Then
  MsgBox "Por favor, seleccione Meses, Bimestres o Trimestres
    únicamente."
  Cancel = 1
End If

```

```
End Sub
```

```
Private Sub Resolver_Click()
```

```

  Dim strSQL As String, NumReg As Integer, CadObj As String, _
    CadMac As String
  Dim oApp As Object ' Variable que contiene la referencia de Microsoft
    Excel.
  Dim ExcelNoSeEjecutaba As Boolean ' Indicador para liberación final

  DoCmd.SetWarnings False
  strSQL = "DELETE * FROM [Demanda Intercambio Excel];"
  DoCmd.RunSQL strSQL

  strSQL = "INSERT INTO [Demanda Intercambio Excel] ( [N° de Modelo],
    Período, Demanda, [Costo de Producción], [Costo de
    Mantener], [Costo de Escasez ] ) " & _
    "SELECT [Productos en Modelos P].[N° de Modelo], [Productos
    en Modelos P].Período, [Productos en Modelos P].Demanda,
    [Productos en Modelos P].[Costo de Producción], [Productos
    en Modelos P].[Costo de Mantener], [Productos en Modelos
    P].[Costo de Escasez ] " & _
    "FROM [Productos en Modelos P] " & _
    "WHERE ((([Productos en Modelos P].[N° de Modelo]) =
    [forms]![Modelos de Programación]![N° de Modelo]));"
  DoCmd.RunSQL strSQL

  strSQL = "DELETE * FROM [Demanda Intercambio Excel Modelo];"
  DoCmd.RunSQL strSQL

  NumReg = DCount("[N° de Modelo]", "Productos en Modelos P", "[N° de
    Modelo]=" & Str([N° de Modelo]))

  strSQL = "INSERT INTO [Demanda Intercambio Excel Modelo] ( [Límite de
    Producción], [Inventario Inicial], [N° de Registros] ) " & _
    "SELECT [Modelos de Programación].[Límite de Producción],
    [Modelos de Programación].[Inventario Inicial], " &
    Str(NumReg) & " AS Expr1 " & _
    "FROM [Modelos de Programación] " & _
    "WHERE ((([Modelos de Programación].[N° de Modelo]) =
    [forms]![Modelos de Programación]![N° de Modelo]));"
  DoCmd.RunSQL strSQL
  DoCmd.SetWarnings True

  ' Prueba para ver si hay una copia de Microsoft Excel ejecutándose.
  On Error Resume Next ' Inicializa la interceptación del error.
  ' La llamada a la función Getobject sin el primer argumento devuelve
    una referencia a una instancia de la aplicación . Si no se está
    ejecutando, se produce un error . Observe que se utiliza la coma

```

```
    como el primer marcador del argumento.
Set oApp = GetObject(, "Excel.Application")

If Err.Number <> 0 Then ExcelNoSeEjecutaba = True
' Borra el objeto Err si se produce un error.
Err.Clear
' Comprueba Excel. Si se está ejecutando Excel,
' lo introduce en la tabla Running Object .
DetectExcel
' Establece la variable de objeto para hacer referencia al archivo que
  desea ver.

Select Case NumReg
  Case 2
    CadObj = "c:\datos\proula 97\Hojas\Modelo02.xls"
    CadMac = "Modelo02.xls!Calcular2"
  Case 3
    CadObj = "c:\datos\proula 97\Hojas\Modelo03.xls"
    CadMac = "Modelo03.xls!Calcular3"
  Case 4
    CadObj = "c:\datos\proula 97\Hojas\Modelo04.xls"
    CadMac = "Modelo04.xls!Calcular4"
  Case 6
    CadObj = "c:\datos\proula 97\Hojas\Modelo06.xls"
    CadMac = "Modelo06.xls!Calcular6"
  Case 8
    CadObj = "c:\datos\proula 97\Hojas\Modelo08.xls"
    CadMac = "Modelo08.xls!Calcular8"
  Case 12
    CadObj = "c:\datos\proula 97\Hojas\Modelo12.xls"
    CadMac = "Modelo12.xls!Calcular12"
  Case 24
    CadObj = "c:\datos\proula 97\Hojas\Modelo24.xls"
    CadMac = "Modelo24.xls!Calcular24"
End Select

Set oApp = GetObject(CadObj)
oApp.Application.Visible = True
oApp.Parent.Windows(1).Visible = True
oApp.Application.Run CadMac
oApp.Application.DisplayAlerts = False
oApp.Application.Save
oApp.Application.Quit
oApp.Application.DisplayAlerts = True

' Si no se está ejecutando esta copia de Microsoft Excel cuando
' comenzó, ciérrela utilizando el método Quit de la propiedad
  Application.
' Observe que cuando intenta salir de Microsoft Excel, la barra de
  título de Microsoft Excel parpadea y Microsoft Excel muestra un
  mensaje preguntándole si desea guardar los archivos cargados.

If ExcelNoSeEjecutaba Then
  oApp.Application.Quit
End If
```

Set oApp = Nothing ' Libera la referencia a la aplicación y a la hoja de cálculo.

```
' Comienza nuevamente el trabajo en Access
DoCmd.SetWarnings False
strSQL = "UPDATE [Productos en Modelos P] INNER JOIN [Modelos
    Resultados Excel] " & "ON ([Productos en Modelos P].Período
    = [Modelos Resultados Excel].Período) " & "AND ([Productos
    en Modelos P].[N° de Modelo] = [Modelos Resultados
    Excel].[N° de Modelo]) " & _
"SET [Productos en Modelos P].[A Producir] = [Modelos
    Resultados Excel]![A Producir], [Productos en Modelos P].[A
    Mantener] = " & _
"[Modelos Resultados Excel]![A Mantener], [Productos en
    Modelos P].[A Demorar] = [Modelos Resultados Excel]![A
    Demorar];"
DoCmd.RunSQL strSQL
DoCmd.SetWarnings True

CostoGP = 0
CostoGI = 0
CostoGE = 0
[Costo del Plan] = 0
PorcGP = 0
PorcGI = 0
PorcGE = 0
```

End Sub

Sub DetectExcel()

```
' El procedimiento detecta un Excel en ejecución y lo registra.
Const WM_USER = 1024
Dim hWnd As Long
' Si se está ejecutando Excel esta llamada API devuelve el controlador
.

hWnd = FindWindow("XLMAIN", 0)

If hWnd = 0 Then
    ' 0 quiere decir que Excel no se está ejecutando .
    Exit Sub
Else
    ' Excel se está ejecutando por lo que se utiliza la función API
    SendMessage
    ' para introducirlo en la tabla Running Object.
    SendMessage hWnd, WM_USER + 18, 0, 0
End If
```

End Sub

```
Private Sub Editar_Productos_Click()
On Error GoTo Err_Editar_Productos_Click
```

```
Dim stDocName As String
Dim stLinkCriteria As String
```

```
    stDocName = "Productos Terminados"
    DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria

Exit_Editar_Productos_Click:
    Exit Sub

Err_Editar_Productos_Click:
    MsgBox Err.Description
    Resume Exit_Editar_Productos_Click

End Sub

Private Sub Editar_Planes_de_Producción_Click()
On Error GoTo Err_Editar_Planes_de_Producción_Click

    Dim stDocName As String
    Dim stLinkCriteria As String

    stDocName = "Planes de Producción"
    DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria

Exit_Editar_Planes_de_Producción_Click:
    Exit Sub

Err_Editar_Planes_de_Producción_Click:
    MsgBox Err.Description
    Resume Exit_Editar_Planes_de_Producción_Click

End Sub

Private Sub Presentar_Click()
On Error GoTo Err_Presentar_Click

    Dim stDocName As String

    stDocName = "Modelos de Programación"
    DoCmd.DoMenuItem acFormBar, acRecordsMenu, 5, , acMenuVer70
    DoCmd.OpenReport stDocName, acPreview, , "[N° de Modelo] = " &
        Str([N° de Modelo])

Exit_Presentar_Click:
    Exit Sub

Err_Presentar_Click:
    MsgBox Err.Description
    Resume Exit_Presentar_Click

End Sub

Private Sub Imprimir_Click()
On Error GoTo Err_Imprimir_Click

    Dim stDocName As String

    stDocName = "Modelos de Programación"
```

```
DoCmd.DoMenuItem acFormBar, acRecordsMenu, 5, , acMenuVer70
DoCmd.OpenReport stDocName, acNormal, , "[Nº de Modelo] = " &
    Str([Nº de Modelo])

Exit_Imprimir_Click:
Exit Sub

Err_Imprimir_Click:
MsgBox Err.Description
Resume Exit_Imprimir_Click

End Sub

Private Sub Editar_Unidades_de_Medicación_Click()
On Error GoTo Err_Editar_Unidades_de_Medicación_Click

Dim stDocName As String
Dim stLinkCriteria As String

stDocName = "Unidades de Medicación"
DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria

Exit_Editar_Unidades_de_Medicación_Click:
Exit Sub

Err_Editar_Unidades_de_Medicación_Click:
MsgBox Err.Description
Resume Exit_Editar_Unidades_de_Medicación_Click

End Sub

Private Sub Editar_Conversiones_Click()
On Error GoTo Err_Editar_Conversiones_Click

Dim stDocName As String
Dim stLinkCriteria As String

stDocName = "Conversiones"
DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria

Exit_Editar_Conversiones_Click:
Exit Sub

Err_Editar_Conversiones_Click:
MsgBox Err.Description
Resume Exit_Editar_Conversiones_Click

End Sub
```

**DE LAS MACROS EXCEL 97**

```
Sub Calcular2()

    Range("A1").Select
    Selection.QueryTable.Refresh BackgroundQuery:=False
    Range("A7").Select
    Selection.QueryTable.Refresh BackgroundQuery:=False

    SolverLoad LoadArea:="$N$6:$N$12"
    SolverSolve True

    Workbooks.Open FileName:="C:\Datos\PROULA 97\Hojas\Resultados.xls"
    Columns("A:E").Select
    Selection.ClearContents

    Windows("Modelo02.xls").Activate
    Range("A7:B9").Select
    Selection.Copy
    Windows("Resultados.xls").Activate
    Range("A1").Select
    Selection.PasteSpecial Paste:=xlValues, Operation:=xlNone,
        SkipBlanks:=False, Transpose:=False

    Windows("Modelo02.xls").Activate
    Range("G7:I9").Select
    Selection.Copy
    Windows("Resultados.xls").Activate
    Range("C1").Select
    Selection.PasteSpecial Paste:=xlValues, Operation:=xlNone,
        SkipBlanks:=False, Transpose:=False

    Windows("Resultados.xls").Activate
    ActiveWorkbook.Save
    ActiveWindow.Close
End Sub

Sub Calcular3()

    Range("A1").Select
    Selection.QueryTable.Refresh BackgroundQuery:=False
    Range("A7").Select
    Selection.QueryTable.Refresh BackgroundQuery:=False

    SolverLoad LoadArea:="$N$6:$N$13"
    SolverSolve True

    Workbooks.Open FileName:="C:\Datos\PROULA 97\Hojas\Resultados.xls"
    Columns("A:E").Select
    Selection.ClearContents

    Windows("Modelo03.xls").Activate
    Range("A7:B10").Select
    Selection.Copy
```



```
Windows("Resultados.xls").Activate
Range("A1").Select
Selection.PasteSpecial Paste:=xlValues, Operation:=xlNone,
    SkipBlanks:=False, Transpose:=False

Windows("Modelo03.xls").Activate
Range("G7:I10").Select
Selection.Copy
Windows("Resultados.xls").Activate
Range("C1").Select
Selection.PasteSpecial Paste:=xlValues, Operation:=xlNone,
    SkipBlanks:=False, Transpose:=False

Windows("Resultados.xls").Activate
ActiveWorkbook.Save
ActiveWindow.Close
End Sub

Sub Calcular4()

Range("A1").Select
Selection.QueryTable.Refresh BackgroundQuery:=False
Range("A7").Select
Selection.QueryTable.Refresh BackgroundQuery:=False

SolverLoad LoadArea:="$N$6:$N$14"
SolverSolve True

Workbooks.Open FileName:="C:\Datos\PROULA 97\Hojas\Resultados.xls"
Columns("A:E").Select
Selection.ClearContents

Windows("Modelo04.xls").Activate
Range("A7:B11").Select
Selection.Copy
Windows("Resultados.xls").Activate
Range("A1").Select
Selection.PasteSpecial Paste:=xlValues, Operation:=xlNone,
    SkipBlanks:=False, Transpose:=False

Windows("Modelo04.xls").Activate
Range("G7:I11").Select
Selection.Copy
Windows("Resultados.xls").Activate
Range("C1").Select
Selection.PasteSpecial Paste:=xlValues, Operation:=xlNone,
    SkipBlanks:=False, Transpose:=False

Windows("Resultados.xls").Activate
ActiveWorkbook.Save
ActiveWindow.Close
End Sub

Sub Calcular6()

Range("A1").Select
```

```
Selection.QueryTable.Refresh BackgroundQuery:=False
Range("A7").Select
Selection.QueryTable.Refresh BackgroundQuery:=False

SolverLoad LoadArea:="$N$6:$N$16"
SolverSolve True

Workbooks.Open FileName:="C:\Datos\PROULA 97\Hojas\Resultados.xls"
Columns("A:E").Select
Selection.ClearContents

Windows("Modelo06.xls").Activate
Range("A7:B13").Select
Selection.Copy
Windows("Resultados.xls").Activate
Range("A1").Select
Selection.PasteSpecial Paste:=xlValues, Operation:=xlNone,
    SkipBlanks:=False, Transpose:=False

Windows("Modelo06.xls").Activate
Range("G7:I13").Select
Selection.Copy
Windows("Resultados.xls").Activate
Range("C1").Select
Selection.PasteSpecial Paste:=xlValues, Operation:=xlNone,
    SkipBlanks:=False, Transpose:=False

Windows("Resultados.xls").Activate
ActiveWorkbook.Save
ActiveWindow.Close
End Sub

Sub Calcular8()

Range("A1").Select
Selection.QueryTable.Refresh BackgroundQuery:=False
Range("A7").Select
Selection.QueryTable.Refresh BackgroundQuery:=False

SolverLoad LoadArea:="$N$6:$N$18"
SolverSolve True

Workbooks.Open FileName:="C:\Datos\PROULA 97\Hojas\Resultados.xls"
Columns("A:E").Select
Selection.ClearContents

Windows("Modelo08.xls").Activate
Range("A7:B15").Select
Selection.Copy
Windows("Resultados.xls").Activate
Range("A1").Select
Selection.PasteSpecial Paste:=xlValues, Operation:=xlNone,
    SkipBlanks:=False, Transpose:=False

Windows("Modelo08.xls").Activate
Range("G7:I15").Select
```

```
Selection.Copy
Windows("Resultados.xls").Activate
Range("C1").Select
Selection.PasteSpecial Paste:=xlValues, Operation:=xlNone,
    SkipBlanks:=False, Transpose:=False

Windows("Resultados.xls").Activate
ActiveWorkbook.Save
ActiveWindow.Close
End Sub

Sub Calcular12()

Range("A1").Select
Selection.QueryTable.Refresh BackgroundQuery:=False
Range("A7").Select
Selection.QueryTable.Refresh BackgroundQuery:=False

SolverLoad LoadArea:="$N$6:$N$22"
SolverSolve True

Workbooks.Open FileName:="C:\Datos\PROULA 97\Hojas\Resultados.xls"
Columns("A:E").Select
Selection.ClearContents

Windows("Modelo12.xls").Activate
Range("A7:B19").Select
Selection.Copy
Windows("Resultados.xls").Activate
Range("A1").Select
Selection.PasteSpecial Paste:=xlValues, Operation:=xlNone,
    SkipBlanks:=False, Transpose:=False

Windows("Modelo12.xls").Activate
Range("G7:I19").Select
Selection.Copy
Windows("Resultados.xls").Activate
Range("C1").Select
Selection.PasteSpecial Paste:=xlValues, Operation:=xlNone,
    SkipBlanks:=False, Transpose:=False

Windows("Resultados.xls").Activate
ActiveWorkbook.Save
ActiveWindow.Close
End Sub

Sub Calcular24()

Range("A1").Select
Selection.QueryTable.Refresh BackgroundQuery:=False
Range("A7").Select
Selection.QueryTable.Refresh BackgroundQuery:=False

SolverLoad LoadArea:="$N$6:$N$34"
SolverSolve True
```

```

Workbooks.Open FileName:="C:\Datos\PROULA 97\Hojas\Resultados.xls"
Columns("A:E").Select
Selection.ClearContents

Windows("Modelo24.xls").Activate
Range("A7:B31").Select
Selection.Copy
Windows("Resultados.xls").Activate
Range("A1").Select
Selection.PasteSpecial Paste:=xlValues, Operation:=xlNone,
    SkipBlanks:=False, Transpose:=False

Windows("Modelo24.xls").Activate
Range("G7:I31").Select
Selection.Copy
Windows("Resultados.xls").Activate
Range("C1").Select
Selection.PasteSpecial Paste:=xlValues, Operation:=xlNone,
    SkipBlanks:=False, Transpose:=False

Windows("Resultados.xls").Activate
ActiveWorkbook.Save
ActiveWindow.Close
End Sub

```

**DE LAS CONSULTAS DE INTERCAMBIO  
DE DATOS EN LAS HOJAS DE CÁLCULO EXCEL 97**

Celdas A1 - C1:

```

SELECT `Demanda Intercambio Excel Modelo`. `Límite de Producción`,
    `Demanda Intercambio Excel Modelo`. `Inventario Inicial`, `Demanda
    Intercambio Excel Modelo`. `Nº de Registros`
FROM `C:\Datos\PROULA 97\SARCAI2`. `Demanda Intercambio Excel Modelo`
    `Demanda Intercambio Excel Modelo`

```

Celdas A7 - F7:

```

SELECT `Demanda Intercambio Excel`. `Nº de Modelo`, `Demanda Intercambio
Excel`. Período, `Demanda Intercambio Excel`. Demanda, `Demanda
Intercambio Excel`. `Costo de Producción`, `Demanda Intercambio
Excel`. `Costo de Mantener`, `Demanda Intercambio Excel`. `Costo de
Escasez`
FROM `C:\Datos\PROULA 97\SARCAI2`. `Demanda Intercambio Excel` `Demanda
Intercambio Excel`
ORDER BY `Demanda Intercambio Excel`. `Nº de Modelo`, `Demanda Intercambio
Excel`. Período

```

## Conclusiones y Recomendaciones

Uno de los principales objetivos propuestos y alcanzados con este proyecto es haber sentado una base sólida de información en materia de inventarios (pero que puede expandirse vertical y horizontalmente) con la cual, a partir de este momento, será posible coleccionar datos en suficiente cuantía, asegurando su calidad y confiabilidad al máximo posible para la construcción de estadísticas y modelos complejos en el futuro.

Cuando se habla de expansión vertical se intenta señalar que es posible ahondar en el mismo tema incorporando nuevas facilidades relacionadas, y cuando se habla de expansión horizontal, se señala que también es posible agregar temas nuevos integrándolos al sistema (como serían los subsistemas de control de calidad, personal, control de correspondencia, agenda, contabilidad, etc.).

Los modelos programados son en esencia sencillos, pero muy prácticos para apoyar el proceso decisonal. Quedó demostrado, al menos en la experiencia del autor, que no tiene sentido proponerse la estructuración de modelos complejos sin antes haber resuelto la disponibilidad de la información que los nutrirá. Los modelos creados, o ajustados a la situación particular, como se prefiera, resultan interesantes, ya que interactúan con la base de datos de forma natural. Pensando en alcanzar esta meta fue que se decidió consumir grandes recursos en el diseño y la programación del sistema de información operativo y no darlo por hecho.

El camino lógico que debe seguir este proyecto, es el siguiente: lo primero (luego de probarlo exhaustivamente) es garantizar el aspecto operativo del software, contrastando su funcionamiento en tiempo real con el sistema actual y afinando los detalles que surjan. Luego, deberían perfeccionarse los modelos creados, incorporando automáticamente más información relativa a los procesos productivos (costos, detalles de los procesos, etc.) que evitarían la necesidad de estimar aparte varios de los parámetros empleados en ellos. Cuando se cuente con datos suficientes, habría que analizarlos estadísticamente con miras a crear nuevos modelos más

complejos que incorporen, por ejemplo, probabilidades, series de tiempo y un enfoque multivariante del problema de inventarios.

En el camino sería interesante estudiar el impacto del sistema programado en la vida de la organización. Podrían construirse indicadores de gestión que ayudaran a mejorar la gerencia de la organización. Seguramente, ésta se verá afectada positivamente desde el punto de vista económico y desde el punto de vista de la eficiencia laboral.

Este proyecto constituye el primero en su tipo para la Planta de Medicamentos y seguirá el trabajo asociado con él, con miras a perfeccionarlo, de ser posible, hasta convertirlo en un producto generalizado que pueda ser utilizado en cualquier organización cuyo sistema productivo se ajuste al aquí descrito, es decir, Proveedores-Almacén de Insumos-Proceso Productivo-Almacén de Productos Terminados-Clientes.

La versión del proyecto presentada en este documento está un poco más allá de lo que podría llamarse fase prototipo. Es completamente funcional, pero necesita dársele algún tiempo para ser probado intensivamente y lograr la colección de datos que sirvan para afinarlo. Los reportes y consultas que se documentan, son apenas los básicos, ideados a partir de la experiencia de análisis y del sentido común, sin embargo se podrían elaborar un sinnúmero de elementos de este tipo adicionales.

Una recomendación especial, con relación al uso de los modelos programados, es la ejecución de los mismos solapadamente en el tiempo (esto es, tratando de predecir la historia) para monitorear que sus resultados están ajustándose efectivamente a la realidad.

El enfoque de bases de datos relacionales utilizado fue una decisión acertada. La base de datos es en extremo sólida desde el punto de vista conceptual y de normalización, lo que facilitó la construcción de consultas de índole diversa.

La herramienta empleada (Access 97) se comportó a la altura de las exigencias, es de fácil comprensión, poderosa y ágil para la elaboración de proyectos de bases de datos, aun cuando se trata de un producto considerado por los entendidos dentro de los de baja potencia. Posee tres características que a juicio del autor son primordiales: permite incorporar a la base de datos,

explícitamente, el modelo relacional con todas sus implicaciones (control de concurrencia, revisión de la integridad referencial, actualización y eliminación en cascada, etc.), provee al programador una serie de asistentes que le ayudan a construir la mayoría de los objetos importantes, generando en buena medida el código Visual Basic, y permite fácilmente el intercambio de datos entre aplicaciones Microsoft Windows 95, con lo que no sólo se cuenta con las herramientas que se puedan programar, sino que se pueden utilizar otras herramientas de terceros (tal como se hizo en los Modelos de Programación, con Excel).

Es justo decir por otra parte que la carga gráfica de los controles de formularios y el gran tamaño de la base de datos hace imperativo contar con un equipo de computación robusto (Pentium 200 MHz en adelante, 32 Mb en RAM, Disco Duro de 3 Gb. al menos) y que, aunque toda la programación se realizó en Windows 95, se encontró la necesidad de utilizar para su implementación, de preferencia, Windows NT (que es mucho más robusto que el primero).

En el aspecto computacional, cuatro tareas quedan pendientes: la implementación del diseño en otra plataforma más profesional (como UNIX-Sparc o IBM), la incorporación de las ideas de DATA MINING, que están en el tapete de la investigación en computación estadística y análisis de datos, la interconexión de la base de datos en línea con procesadores estadísticos avanzados y probar la funcionalidad del programa en ambiente de redes.

En vista del gran volumen de trabajo y esfuerzo que ha significado este proyecto para el autor, una recomendación lógica que cabe mencionar es la necesidad de enfrentar los problemas de sistemas de información conformando grupos interdisciplinarios de trabajo. Se necesita una división del trabajo convocando la participación de varias personas conformando un equipo. En este caso, el autor enfrentó por sí mismo el análisis, diseño y programación de la base de datos, programación de las interfaces, creación y ajuste de los modelos y su programación, lo que a todas luces es demasiado esfuerzo para una sola persona. En el futuro se procurarán auxiliares en las tareas sobre todo rutinarias.

Finalmente, es de esperar que el lector haya disfrutado la lectura de este texto tanto como el autor disfrutó escribiéndolo, y que la Planta de Medicamentos de PROULA se proponga decididamente incorporar este producto a su gestión cotidiana.

## Bibliografía

1. BATINI, CERI y NAVATHE. “Diseño Conceptual de Bases de Datos. Un enfoque de Entidades – Interrelaciones”. Addison – Wesley / Díaz de Santos. E.U.A., 1994.
2. DARWEN, H. & DATE, C. J. “Introducing The Third Manifesto”. Database Programing & Design, January, 1995.
3. DATE, C.J. “Introducción a los Sistemas de Bases de Datos. Volumen 1. V Edición”. Addison – Wesley Iberoamericana. México, 1993.
4. GOULD, EPPEN y SCHMIDT. “Investigación de Operaciones en la Ciencia Administrativa. Tercera Edición”. Prentice – Hall Hispanoamérica S.A. México, 1992.
5. HERAS, A. “La Planta de Medicamentos va al Cuartel”. Alcance / ULA. Mérida, Mayo 1996.
6. HILLIER S. y LIEBERMAN, G. “Introducción a la Investigación de Operaciones. Quinta Edición”. McGraw-Hill. México, 1991.
7. IBM Corp. “Home Page”. <http://www.ibm.com/>
8. KORTH, H. y SILBERSCHATZ, A. “Fundamentos de Bases de Datos”. McGraw – Hill. México, 1997.
9. LAUMAILLE, R. “Gestión de Stocks”. Ediciones Gestión 2000 S.A. / Serie Empresa. España, 1992.
10. LAW, A. y KELTON W. “Simulation Modeling and Analysys. Second Edition”. McGraw – Hill. U.S.A., 1991.



11. MCKINNEY, Bruce. "Programación Avanzada con Visual Basic. Versión 4". Microsoft Press / McGraw – Hill. España, 1996.
12. Microsoft Corp. "Access Home Page". <http://www.microsoft.com/msaccess/>
13. Microsoft Corp. "Manual de Ayuda en Línea del Access 97".
14. Microsoft Corp. "Visual FoxPRO Home Page". <http://www.microsoft.com/VFOXPRO/>
15. MOOD, GRAYBILL & BOES. "Introduction to the Theory of Statistics. Third Edition". McGraw – Hill. U.S.A., 1963.
16. NADDOR, Eliezer. "Inventory Systems". John Wiley & Sons. U.S.A., 1966.
17. PONSOT B., Ernesto A. "Apuntes de Clases: Introducción a las Bases de Datos Relacionales". Universidad de Los Andes, Mérida - Venezuela, en proceso.
18. PONSOT B., Ernesto A. "Computarización de Modelos de Inventario. Versión 1.0". Universidad de Los Andes, Biblioteca de la Facultad de Ingeniería. Mérida - Venezuela, 1991.
19. PROULA C.A. División Medicamentos. "Informe al Consejo de Fomento". Septiembre, 1996.
20. PROULA C.A División Medicamentos, "Presentación. (bíptico)", Lagunillas – Mérida.
21. TAHA, Hamdy A. "Investigación de Operaciones. V Edición". Alfaomega Grupo Editor. México, 1995.
22. TIZNADO S., Marco A. "El Camino Fácil a Access 7.0". McGraw – Hill, Serie Enter. Colombia, 1996.
23. WAYNE L., Winston. "Investigación de Operaciones. Aplicaciones y Algoritmos". Grupo Editorial Iberoamérica. México, 1994.

---

<b>CAPÍTULO IV</b>	<b>167</b>
<b>Modelos</b>	<b>167</b>
IV.1. Modelos EOQ	169
IV.1.1. Modelo Teórico	170
Formulario para los Modelos EOQ	174
IV.1.3. Descripción	175
IV.1.4. Código Fuente	178
IV.2. Modelos de Programación	189
IV.2.1. Modelo Teórico	190
Formulario para los Modelos de Programación	195
IV.2.3. Descripción	196
IV.2.4. Ejemplo de la Hoja de Cálculo	200
IV.2.5. Código Fuente	202
<b>Conclusiones y Recomendaciones</b>	<b>219</b>
<b>Bibliografía</b>	<b>222</b>

# **ANEXO 1**

## **Formularios Básicos de Datos y Código Fuente que los Controla**

## Índice de Formularios

Formulario	Pág.
Cargos	5
Clientes (1° Página)	5
Clientes (2° y 3° Páginas)	6
Condiciones	10
Conversiones	11
Cotizaciones	12
Departamentos	15
Destinos	15
Documentación	16
Entradas de Insumos 3ros (1° Página)	16
Entradas de Insumos 3ros (2° Página)	17
Escalas de Precios	19
Estados Civiles	19
Estados de Insumos	20
Facturas (1° Página)	20
Facturas (2° y 3° Páginas)	21
Figuras Jurídicas	24
Fórmulas (1° y 2° Páginas)	25
Fórmulas (3° y 4° Páginas)	26
Grupos de Clientes	30
Insumos	34
Llegadas de Insumos al Almacén (1° Página)	36
Llegadas de Insumos al Almacén (2° Página)	37
Magnitudes Físicas	41
Notas	41
Órdenes de Compra (1° y 2° Páginas)	42
Órdenes de Producción (1° y 2° Páginas)	45

Formulario	Pág.
Órdenes de Producción (3° y 4° Páginas)	46
Países	50
Pedidos de Clientes (1° Página)	52
Pedidos de Clientes (2° Página)	53
Pedir Año para resumen de Ventas	56
Solicitud de Fechas	61
Pedir Fechas para la Demanda	64
Pedir Fechas para Ventas	68
Pedir Fechas para Ventas por Productos	71
Pedir Número del Plan	74
Personas (1° y 2° Páginas)	76
Planes de Producción	79
Presentaciones	81
Productos Genéricos	81
Productos Terminados (1° y 2° Páginas)	82
Proveedores (1° Página)	86
Proveedores (2° Página)	87
Razones de Salidas	90
Sectores Económicos	90
Tenencia de Locales	91
Tipos de Clientes	91
Tipos de Insumos	91
Tipos de Órdenes de Pedido	92
Tipos de Proveedores	92
Tipos de Salidas	92
Unidades de Medición	93
Vehículos	94

Este anexo muestra cada uno de los formularios de datos del sistema, con excepción de los formularios dedicados a los Modelos, los cuales se explican con detalle en el Capítulo III. Dos de los formularios (**Grupos de Clientes y Países**) muestran una breve explicación de particularidades que es útil comprender a cabalidad. En algunos formularios hay botones cuyo nombre comienza con el rótulo “Editar...”; estos botones abren en modo edición el formulario al que hacen referencia. Otros botones tienen como icono una lupa o una impresora, éstos permiten imprimir el informe asociado con los datos que se encuentra editando el usuario, la lupa como presentación preliminar y la impresora, directo al dispositivo de impresión. Otros botones, como el símbolo de sumatoria, al presionarse actualizan los cálculos a que hubiere lugar en el formulario.

Los elementos que componen un formulario son objetos (como etiquetas, cuadros de texto, cuadros de lista, botones, selectores de registro, barras de desplazamiento y otros) susceptibles de recibir código programado que responda a eventos eventualmente a ocurrir en tiempo de ejecución. Un evento es cualquier acción iniciada por el usuario cuando algún control tiene enfoque dentro del formulario (eventos son, por ejemplo, un clic de ratón sobre un botón, la apertura y el cierre del formulario, la eliminación del contenido de un cuadro de texto, etc.). Así pues, cada objeto del formulario tiene entre sus propiedades, eventos a los cuales puede responder; cada evento se materializa en un procedimiento o función que se ejecutará cuando éste sea detectado. Estos procedimientos y funciones son escritos por el programador en caso que desee que la respuesta al evento dado sea distinta de la respuesta pre-programada por defecto (la cual generalmente es no hacer nada).

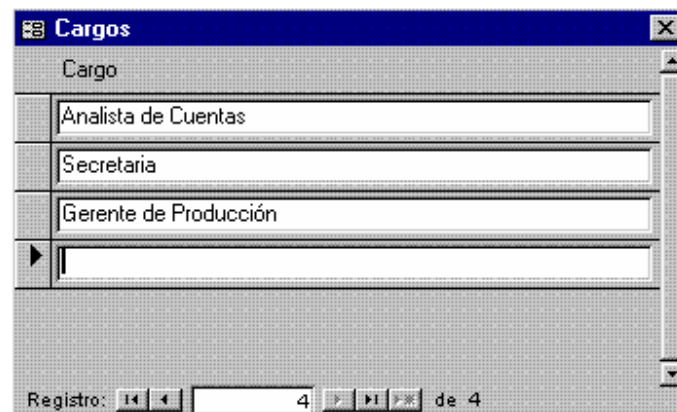
Cualquier manejador de bases de datos provee al menos dos formas de insertar o modificar la información. La primera forma se obtiene abriendo cada tabla y alterando su contenido directamente. La segunda forma se materializa cuando, a partir de las relaciones establecidas entre las tablas, se diseña una pantalla especial que contiene la representación de cada uno de los datos involucrados, dispuestos de manera que el trabajo de edición representa menor complejidad al usuario. Esta segunda forma tiene ventajas importantes sobre la primera. Una de ellas es que el espacio de pantalla puede ser modificado a conveniencia, ocultando y mostrando información diferente a cada usuario; otra ventaja es que se pueden crear distintas vistas de un

mismo conjunto de tablas, algunas con capacidades de modificación de la base de datos y otras sin esta capacidad; otro aspecto importante es la posibilidad de disponer en un solo espacio distintas representaciones de las tablas y controlar los datos de unas y otras en un mismo acto de edición. La ventaja más importante de la segunda forma es la posibilidad de incorporar, al código que controla la pantalla de datos instrucciones complejas que realizan cálculos o validaciones, que en la primera representación no es posible implantar.

En ACCESS, la segunda forma señalada se denomina Formulario. Algunos formularios se emplean en la recolección de parámetros o donde se definen los cálculos necesarios, para resolver los modelos y campos calculados. El formulario provee una interfaz apropiada para el control de la información que entra a la base de datos, esto es, hace posible la implantación de controles y verificaciones adicionales a las validaciones establecidas en el diseño de las tablas. Esto último es importante cuando se trata de mantener la integridad de la base de datos; por esta razón, es altamente recomendable introducir datos a través de formularios y no directamente en las tablas. El control que es posible programar en cada formulario se realiza utilizando código Visual Basic para Aplicaciones, incluido en lo que se denomina el módulo del formulario.

Dentro de cada uno de los formularios siguientes se ha procurado incorporar alguna información a manera de ejemplo (que no necesariamente se corresponde con la verdad) y se espera que sirva para demostrar las posibilidades del formulario. Donde sea aplicable se incorporará una sección que lista el código Visual Basic programado para dominar su actividad.

Se omiten adrede las explicaciones detalladas del funcionamiento de los formularios por considerar que esto es materia del ACCESS, es decir, los formularios se emplean en SARCAI de la misma forma como se emplean en cualquier otra aplicación desarrollada en ACCESS 97. Sólo se darán breves explicaciones en los casos de que ciertos controles tengan un significado especial, que aparece por primera vez en el texto. Las explicaciones referidas a los datos pueden encontrarse en los capítulos del texto. El lector deberá utilizar el sistema para formarse un criterio sobre su funcionamiento. Esta sección no sustituye en modo alguno la práctica.

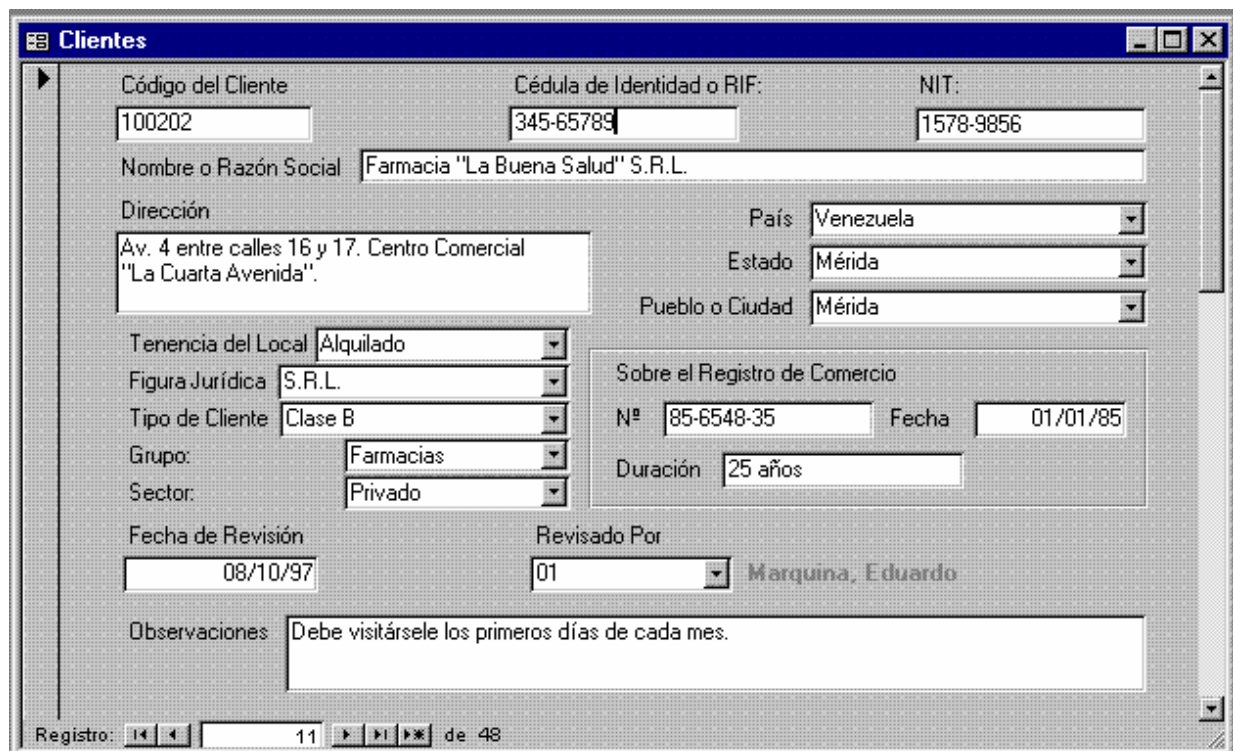


The screenshot shows a window titled "Cargos" with a list of job titles. The list contains three entries: "Analista de Cuentas", "Secretaria", and "Gerente de Producción". Below the list, there is a status bar that reads "Registro: 4 de 4".

Cargo
Analista de Cuentas
Secretaria
Gerente de Producción

Registro: 4 de 4

Figura 1. Cargos



The screenshot shows a window titled "Clientes" with a form for client information. The form includes fields for "Código del Cliente", "Cédula de Identidad o RIF", "NIT", "Nombre o Razón Social", "Dirección", "País", "Estado", "Pueblo o Ciudad", "Tenencia del Local", "Figura Jurídica", "Tipo de Cliente", "Grupo", "Sector", "Fecha de Revisión", "Revisado Por", "Observaciones", "N°", "Fecha", and "Duración". The status bar at the bottom reads "Registro: 11 de 48".

Código del Cliente	Cédula de Identidad o RIF:	NIT:
100202	345-65789	1578-9856
Nombre o Razón Social: Farmacia "La Buena Salud" S.R.L.		
Dirección: Av. 4 entre calles 16 y 17. Centro Comercial "La Cuarta Avenida".		País: Venezuela
		Estado: Mérida
		Pueblo o Ciudad: Mérida
Tenencia del Local: Alquilado	Sobre el Registro de Comercio	
Figura Jurídica: S.R.L.	N°: 85-6548-35	Fecha: 01/01/85
Tipo de Cliente: Clase B	Duración: 25 años	
Grupo: Farmacias		
Sector: Privado		
Fecha de Revisión: 08/10/97	Revisado Por: 01 Marquina, Eduardo	
Observaciones: Debe visitársele los primeros días de cada mes.		

Registro: 11 de 48

Figura 2. Clientes (1° Página)

**Cientes**

**Teléfonos**

Teléfono	Otro Código o Celular	¿Es Fax?
524789		<input type="checkbox"/>
527564		<input checked="" type="checkbox"/>

Registro: 1 de 2

**Accionistas**

Accionista: 0015  
Santiago de Pérez, María

Registro: 1 de 2

**Referencias Bancarias**

Nombre del Banco	Nº de Cuenta	Dirección y Teléfonos	Créditos
Mercantil	1065-586974-2	Av. 5. Edif. La Torre de Los Andes.	Bs.25.000.000,00
Provincial	65-894-68	Av. Urdaneta. Frente al Colegio La Salle	Bs.15.000.000,00

Registro: 1 de 2

Registro: 11 de 48

**Cientes**

**Referencias Comerciales**

Nombre del Comercio	Dirección y Teléfonos	Montos de Créditos
Proveeduría Estudiantil S.R.L.	Av. 16 de Septiembre, al lado de la Facultad de Medicina	Bs.350.000,00
*		Bs.0,00

Registro: 1 de 1

**Sucursales**

Sucursal: 17  
Farmacia Boulevard

Registro: 1 de 1

**Contactos**

Contacto: 0016  
Pérez M., Pedro A.

Registro: 1 de 1

Registro: 11 de 48

Figura 3. Clientes (2º y 3º Páginas)

**Código**



```
1 Attribute VB_Name = "Form_Clientes"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8 Dim ctlLista As Control
9
11 Private Sub Código_del_Cliente_Exit(Cancel As Integer)
12     Forms!Clientes![Sucursales de Clientes]
13 End Sub
14
15
16 Private Sub Editar_Grupos_Exit(Cancel As Integer)
17     Set ctlLista = Forms!Clientes!Grupo
18     ctlLista.Requery
19 End Sub
20
21 Sub Editar_Países__Estados_y_Ciudades_Click()
22 On Error GoTo Err_Editar_Países__Estados_y_Ciudades_Click
24     Dim stDocName As String
25     Dim stLinkCriteria As String
26
27     stDocName = "Países"
28     DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria
29
30 Exit_Editar_Países__Estados_y_Ciudades_C:
31     Exit Sub
32
33 Err_Editar_Países__Estados_y_Ciudades_Click:
34     MsgBox Err.Description
35     Resume Exit_Editar_Países__Estados_y_Ciudades_C
36
37 End Sub
38
39 Private Sub Editar_Países__Estados_y_Ciudades_Exit(Cancel As
Integer)
40
41     Set ctlLista = Forms!Clientes!Ciudad
42     ctlLista.Requery
43     Set ctlLista = Forms!Clientes!Estado
44     ctlLista.Requery
45     Set ctlLista = Forms!Clientes!País
46     ctlLista.Requery
47
48 End Sub
49
52 Private Sub Editar_Personas_Exit(Cancel As Integer)
53
54     Set ctlLista = Forms!Clientes!Responsable
55     ctlLista.Requery
56     Set ctlLista = Forms!Clientes![Contactos en Clientes]
57     ctlLista.Requery
58     Set ctlLista = Forms!Clientes![Accionistas de Clientes]
subformulario]!Accionista
59     ctlLista.Requery
```

```
60 End Sub
61
62 Private Sub Editar_Personas_Jurídicas_Exit(Cancel As Integer)
63
64     Set ctlLista = Forms!Clientes![Figura Jurídica]
65     ctlLista.Requery
66
67 End Sub
68
69 Private Sub Editar_Sectores_Exit(Cancel As Integer)
70     Set ctlLista = Forms!Clientes!Sector
71     ctlLista.Requery
72 End Sub
73
74 Sub Editar_Tenencias_de_Locales_Click()
75 On Error GoTo Err_Editar_Tenencias_de_Locales_Click
76
77     Dim stDocName As String
78     Dim stLinkCriteria As String
79     stDocName = "Tenencia de Locales"
80     DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria
81
82 Exit_Editar_Tenencias_de_Locales_Click:
83     Exit Sub
84
85
86 Err_Editar_Tenencias_de_Locales_Click:
87     MsgBox Err.Description
88     Resume Exit_Editar_Tenencias_de_Locales_Click
89
90 End Sub
91 Sub Editar_Personas_Jurídicas_Click()
92 On Error GoTo Err_Editar_Personas_Jurídicas_Click
93
94     Dim stDocName As String
95     Dim stLinkCriteria As String
96
97     stDocName = "Figuras Jurídicas"
98     DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria
99
100 Exit_Editar_Personas_Jurídicas_Click:
101     Exit Sub
102
103 Err_Editar_Personas_Jurídicas_Click:
104     MsgBox Err.Description
105     Resume Exit_Editar_Personas_Jurídicas_Click
106
107 End Sub
108
109 Private Sub Editar_Tenencias_de_Locales_Exit(Cancel As Integer)
110
111     Set ctlLista = Forms!Clientes![Tenencia del Local]
112     ctlLista.Requery
113
114 End Sub
115
116
117
118 Sub Editar_Tipos_de_Clientes_Click()
```

```
119 On Error GoTo Err_Editar_Tipos_de_Clientes_Click
120
121 Dim stDocName As String
122 Dim stLinkCriteria As String
123
124 stDocName = "Tipos de Clientes"
125 DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria
126
127 Exit_Editar_Tipos_de_Clientes_Click:
128 Exit Sub
129
130 Err_Editar_Tipos_de_Clientes_Click:
131 MsgBox Err.Description
132 Resume Exit_Editar_Tipos_de_Clientes_Click
133
134 End Sub
137
138 Private Sub Editar_Tipos_de_Clientes_Exit(Cancel As Integer)
139
140 Set ctlLista = Forms!Clientes![Tipo de Cliente]
141 ctlLista.Requery
142
143 End Sub
149
150 Sub Editar_Personas_Click()
151 On Error GoTo Err_Editar_Personas_Click
152
153 Dim stDocName As String
154 Dim stLinkCriteria As String
155
156 stDocName = "Personas"
157 DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria
158
159 Exit_Editar_Personas_Click:
160 Exit Sub
161
162 Err_Editar_Personas_Click:
163 MsgBox Err.Description
164 Resume Exit_Editar_Personas_Click
165
166 End Sub
167
168 Private Sub Sucursales_del_Cliente_Enter()
169 Set ctlLista = Forms!Clientes![Sucursales del Cliente]
170 ctlLista.Requery
171 End Sub
172
173 Private Sub Editar_Grupos_Click()
174 On Error GoTo Err_Editar_Grupos_Click
175
176
177 Dim stDocName As String
178 Dim stLinkCriteria As String
179
180 stDocName = "Grupos de Clientes"
181 DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria
182 Exit_Editar_Grupos_Click:
```

```
183 Exit Sub
184
185 Err_Editar_Grupos_Click:
186 MsgBox Err.Description
187 Resume Exit_Editar_Grupos_Click
188
189 End Sub
190 Private Sub Editar_Sectores_Click()
191 On Error GoTo Err_Editar_Sectores_Click
192
193 Dim stDocName As String
194 Dim stLinkCriteria As String
195
196 stDocName = "Sectoros Económicos"
197 DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria
198
199 Exit_Editar_Sectores_Click:
200 Exit Sub
201
202 Err_Editar_Sectores_Click:
203 MsgBox Err.Description
204 Resume Exit_Editar_Sectores_Click
205
206 End Sub
```



Figura 4. Condiciones

De	A	Ponderador: 1(De) = (Pond) * 1(A)
Años	Años	1
Años	Días	365
Bienios	Años	2
Bienios	Días	730
Bimestres	Años	0,1666667
Bimestres	Días	60
Días	Años	0,002739726
Días	Días	1
Días	Meses	0,03333334

Editar Unidades de Medición

Registro: 1 de 27

**Figura 5. Conversiones**

### Código

```

1  Attribute VB_Name = "Form_Conversiones"
2  Attribute VB_Creatable = True
3  Attribute VB_PredeclaredId = True
4  Attribute VB_Exposed = False
5  Option Compare Database
6  Option Explicit
7
8  Dim ctlLista As Control
9  Sub Editar_Unidades_de_Medición_Click()
10 On Error GoTo Err_Editar_Unidades_de_Medición_Click
11
12     Dim stDocName As String
13     Dim stLinkCriteria As String
14
15     stDocName = "Unidades de Medición"
16     DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria
17
18 Exit_Editar_Unidades_de_Medición_Click:
19     Exit Sub
20
21 Err_Editar_Unidades_de_Medición_Click:
22     MsgBox Err.Description
23     Resume Exit_Editar_Unidades_de_Medición_Click
24
25 End Sub
26
27 Private Sub Editar_Unidades_de_Medición_Exit(Cancel As Integer)
28     Set ctlLista = Forms!Conversiones!De

```

```

29     ctlLista.Requery
30     Set ctlLista = Forms!Conversiones!A
31     ctlLista.Requery
32 End Sub

```

**Cotizaciones**

Nº de la Cotización: 1      Código del Cliente: 100202      Farmacia "La Buena Salud" S.R.

Fechas: De Emisión: 29/11/97      Envío: 30/11/97      Recepción: 30/11/97

Productos Cotizados

Producto	Nombre:	Cantidad Cotizada
001	Acetaminofén 500 mg. x 20 Tabletas	10
002	Ácido Fólico 5 mg. x 20 Tabletas	15
*		

Registro: 1 de 2

Notas

Nota Nº: 1      Validez de la Cotización: 15 días.

Registro: 1 de 1

Remite: 03      Rojas, Pedro

Registro: 1 de 1

Editar Clientes  
Editar Productos  
Editar Notas  
Editar Personas

Figura 6. Cotizaciones

### Código

```

1     Attribute VB_Name = "Form_Cotizaciones"
2     Attribute VB_Creatable = True
3     Attribute VB_PredeclaredId = True
4     Attribute VB_Exposed = False
5     Option Compare Database
6     Option Explicit
7
8     Dim ctlLista As Control
9
10    Private Sub Editar_Clientes_Click()
11    On Error GoTo Err_Editar_Clientes_Click
12
13        Dim stDocName As String
14        Dim stLinkCriteria As String
15
16        stDocName = "Clientes"

```

```
17     DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria
18 Exit_Editar_Clientes_Click:
19     Exit Sub
20
21 Err_Editar_Clientes_Click:
22     MsgBox Err.Description
23     Resume Exit_Editar_Clientes_Click
24
25 End Sub
26
27 Private Sub Editar_Clientes_Exit(Cancel As Integer)
28     Set ctlLista = Forms!Cotizaciones![Código del Cliente]
29     ctlLista.Requery
30 End Sub
31
32 Private Sub Editar_Notas_Exit(Cancel As Integer)
33     Set ctlLista = Forms!Cotizaciones![Subformulario Notas en
Cotizaciones]![Nota N°]
34     ctlLista.Requery
35 End Sub
36
37 Private Sub Editar_Personas_Exit(Cancel As Integer)
38     Set ctlLista = Forms!Cotizaciones![Remitente]
39     ctlLista.Requery
40 End Sub
41
42 Private Sub Editar_Productos_Terminados_Click()
43 On Error GoTo Err_Editar_Productos_Terminados_Click
44
45     Dim stDocName As String
46     Dim stLinkCriteria As String
47
48     stDocName = "Productos Terminados"
49     DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria
50
51 Exit_Editar_Productos_Terminados_Click:
52     Exit Sub
53
54 Err_Editar_Productos_Terminados_Click:
55     MsgBox Err.Description
56     Resume Exit_Editar_Productos_Terminados_Click
57
58 End Sub
59 Private Sub Editar_Notas_Click()
60 On Error GoTo Err_Editar_Notas_Click
61
62     Dim stDocName As String
63     Dim stLinkCriteria As String
64
65     stDocName = "Notas"
66     DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria
67
68 Exit_Editar_Notas_Click:
69     Exit Sub
70
71 Err_Editar_Notas_Click:
```

```
72     MsgBox Err.Description
73     Resume Exit_Editar_Notas_Click
74
75 End Sub
76
76 Private Sub Editar_Personas_Click()
77 On Error GoTo Err_Editar_Personas_Click
78
79     Dim stDocName As String
80     Dim stLinkCriteria As String
81
82     stDocName = "Personas"
83     DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria
84
85 Exit_Editar_Personas_Click:
86     Exit Sub
87
88 Err_Editar_Personas_Click:
89     MsgBox Err.Description
90     Resume Exit_Editar_Personas_Click
91
92 End Sub
93
94 Private Sub Editar_Productos_Terminados_Exit(Cancel As Integer)
95     Set ctlLista = Forms!Cotizaciones![Productos Cotizados a Clientes
96     subformulario]![Código del Producto]
97     ctlLista.Requery
98 End Sub
99
99 Private Sub Ver_Informe_Click()
100 On Error GoTo Err_Ver_Informe_Click
101
102     Dim stDocName As String
103
104     stDocName = "Cotizaciones"
105     DoCmd.DoMenuItem acFormBar, acRecordsMenu, 5, , acMenuVer70
106     DoCmd.OpenReport stDocName, acPreview, , "[N° de la Cotización] =
Forms!Cotizaciones![N° de la Cotización]"
107
108 Exit_Ver_Informe_Click:
109     Exit Sub
110
111 Err_Ver_Informe_Click:
112     MsgBox Err.Description
113     Resume Exit_Ver_Informe_Click
114
115 End Sub
116
116 Private Sub Imprimir_Informe_Click()
117 On Error GoTo Err_Imprimir_Informe_Click
118
119     Dim stDocName As String
120
121     stDocName = "Cotizaciones"
122     DoCmd.DoMenuItem acFormBar, acRecordsMenu, 5, , acMenuVer70
123     DoCmd.OpenReport stDocName, acNormal, , "[N° de la Cotización] =
```



```
Forms!Cotizaciones![N° de la Cotización]"
124
125 Exit_Imprimir_Informe_Click:
126     Exit Sub
127
128 Err_Imprimir_Informe_Click:
129     MsgBox Err.Description
130     Resume Exit_Imprimir_Informe_Click
131
132 End Sub
```

The screenshot shows a window titled "Departamentos" with a list of departments. The list contains four entries: "Antibióticos", "Inyectables", "Líquidos", and "Sólidos". Below these is an empty field with an asterisk (\*) next to it. The status bar at the bottom indicates "Registro: 1 de 4".

Figura 7. Departamentos

The screenshot shows a window titled "Destinos" with a list of destinations. The list contains three entries: "M.S.A.S.", "Muestra Médica", and "Venta". Below these is an empty field with an asterisk (\*) next to it. The status bar at the bottom indicates "Registro: 1 de 3".

Figura 8. Destinos

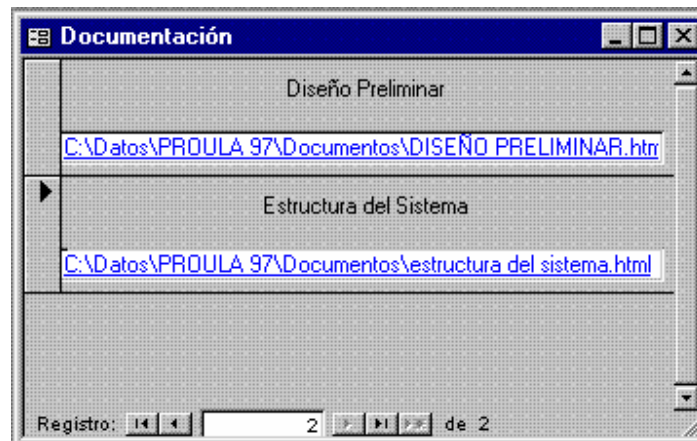


Figura 9. Documentación

Figura 10. Entradas de Insumos 3ros (1° Página)

Figura 11. Entradas de Insumos 3ros (2° Página)

### Código

```

1  Attribute VB_Name = "Form_Entradas de Insumos 3ros"
2  Attribute VB_Creatable = True
3  Attribute VB_PredeclaredId = True
4  Attribute VB_Exposed = False
5  Option Compare Database
6  Option Explicit
7
8  Dim ctlLista As Control
12
13 Private Sub Editar_Clientes_Exit(Cancel As Integer)
14     Set ctlLista = Forms![Entradas de Insumos 3ros]!Cliente
15     ctlLista.Requery
16 End Sub
17
18 Private Sub Editar_Insumos_Exit(Cancel As Integer)
19     Set ctlLista = Forms![Entradas de Insumos 3ros]!Insumo
20     ctlLista.Requery
21 End Sub
22
23 Private Sub Editar_Personas_Exit(Cancel As Integer)
24     Set ctlLista = Forms![Entradas de Insumos 3ros]![Recibido Por]
25     ctlLista.Requery
26 End Sub
27
28 Private Sub Estados_de_Insumos_Exit(Cancel As Integer)
29     Set ctlLista = Forms![Entradas de Insumos 3ros]!Estado
30     ctlLista.Requery
31 End Sub
32
33 Sub Estados_de_Insumos_Click()
34 On Error GoTo Err_Estados_de_Insumos_Click
35     Dim stDocName As String
36     Dim stLinkCriteria As String
37
38
39     stDocName = "Estados de Insumos"

```

```
40     DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria
41
42 Exit_Estados_de_Insumos_Click:
43     Exit Sub
44
45 Err_Estados_de_Insumos_Click:
46     MsgBox Err.Description
47     Resume Exit_Estados_de_Insumos_Click
48
49 End Sub
50 Sub Editar_Personas_Click()
51 On Error GoTo Err_Editar_Personas_Click
52
53     Dim stDocName As String
54     Dim stLinkCriteria As String
55
56     stDocName = "Personas"
57     DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria
58
59 Exit_Editar_Personas_Click:
60     Exit Sub
61
62 Err_Editar_Personas_Click:
63     MsgBox Err.Description
64     Resume Exit_Editar_Personas_Click
65
66 End Sub
67
68 Sub Editar_Clientes_Click()
69 On Error GoTo Err_Editar_Clientes_Click
70
71     Dim stDocName As String
72     Dim stLinkCriteria As String
73
74     stDocName = "Clientes"
75     DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria
76
77 Exit_Editar_Clientes_Click:
78     Exit Sub
79
80 Err_Editar_Clientes_Click:
81     MsgBox Err.Description
82     Resume Exit_Editar_Clientes_Click
83
84 End Sub
85 Sub Editar_Insumos_Click()
86 On Error GoTo Err_Editar_Insumos_Click
87
88     Dim stDocName As String
89     Dim stLinkCriteria As String
90
91     stDocName = "Insumos"
92     DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria
93
94 Exit_Editar_Insumos_Click:
95     Exit Sub
```

```

97 Err_Editar_Insumos_Click:
98     MsgBox Err.Description
99     Resume Exit_Editar_Insumos_Click
100
101 End Sub

```

		----- Límites -----		
Escala	Descripción	Inferior	Superior	% Descuento
▶	1 Precio Full	0	200	0,00%
	2 Precio Económico	201	400	5,00%
	3 Precio Mayorista	401	1000	15,00%
	4 Precio Hospital	1001	500000	20,00%
*	Autonómico)	0	0	0,00%

Registro: 1 de 4

**Figura 12. Escalas de Precios**

Estado Civil	
	Casado(a)
▶	Divorciado(a)
	Soltero(a)
	Viudo(a)
*	

Registro: 2 de 4

**Figura 13. Estados Civiles**

Estado	¿Es Utilizable?
En Almacén	<input checked="" type="checkbox"/>
En Cuarentena	<input type="checkbox"/>
*	<input type="checkbox"/>

Registro: 1 de 2

Figura 14. Estados de Insumos

N° de Factura: 10    N° de Pedido: 5    Fecha de Emisión: 29/11/97    Fecha de Despacho: 05/12/97

Datos del Pedido

Cliente: 100202    Tipo de Orden: Respaldada    N°: 15263

Nombre o Razón Social: Farmacia "La Buena Salud" S.R.L.    Cédula de Identidad o RIF: 345-65789

Dirección: Av. 4 entre calles 16 y 17. Centro Comercial "La Cuarta Avenida".

País: Venezuela    Estado: Mérida    Pueblo o Ciudad: Mérida

Fecha del Pedido: 11/10/97    Condiciones: Contado

Vendedor: 03 Rojas, Pedro

Observaciones: El cliente exige una máxima demora de 15 días, caso contrario retirará el pedido.

Registro: 7 de 7

Figura 15. Facturas (1° Página)

**Facturas**

Productos Facturados

Producto	Orden de Prod. N°:	Cantidad Facturada
▶ Acetaminofén 500 mg. x 20 Tabletas	001 1	200
Ácido Fólico 5 mg. x 20 Tabletas	002 2	170
* [ ]	[ ]	[ ]

Registro: 1 1 de 2

Registro: 7 de 7

**Facturas**

Notas

Nota N°	Contenido
▶ 3	Las cantidades faltantes serán despachadas próximamente
* [ ]	[ ]

Registro: 1 de 1

Elaborada Por: 01 Marquina, Eduardo

Registro: 7 de 7

Figura 16. Facturas (2° y 3° Páginas)

## Código

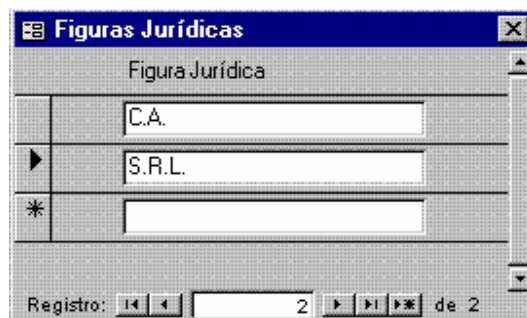
```
1 Attribute VB_Name = "Form_Facturas"
```

```
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8 Dim ctlLista As Control
9
10 Sub Editar_Notas_Click()
11 On Error GoTo Err_Editar_Notas_Click
12
13     Dim stDocName As String
14     Dim stLinkCriteria As String
15
16     stDocName = "Notas"
17     DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria
18
19 Exit_Editar_Notas_Click:
20     Exit Sub
21
22 Err_Editar_Notas_Click:
23     MsgBox Err.Description
24     Resume Exit_Editar_Notas_Click
25
26 End Sub
27
28 Private Sub Editar_Notas_Exit(Cancel As Integer)
29     Set ctlLista = Forms!Facturas![Notas en Facturas
30     subformulario]![Nota
31     ctlLista.Requery
32 End Sub
33
34 Sub Editar_Pedidos_Click()
35 On Error GoTo Err_Editar_Pedidos_Click
36
37     Dim stDocName As String
38     Dim stLinkCriteria As String
39
40     stDocName = "Pedidos de Clientes"
41     DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria
42
43 Exit_Editar_Pedidos_Click:
44     Exit Sub
45
46 Err_Editar_Pedidos_Click:
47     MsgBox Err.Description
48     Resume Exit_Editar_Pedidos_Click
49
50 End Sub
51
52 Private Sub Editar_Pedidos_Exit(Cancel As Integer)
53     Set ctlLista = Forms!Facturas![N° de Pedido]
54     ctlLista.Requery
55 End Sub
56
57 Sub Editar_Personas_Click()
```



```
58 On Error GoTo Err_Editar_Personas_Click
59
60 Dim stDocName As String
61 Dim stLinkCriteria As String
62
63 stDocName = "Personas"
64 DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria
65
66 Exit_Editar_Personas_Click:
67 Exit Sub
68
69 Err_Editar_Personas_Click:
70 MsgBox Err.Description
71 Resume Exit_Editar_Personas_Click
72
73 End Sub
74
75 Private Sub Editar_Personas_Exit(Cancel As Integer)
76 Set ctlLista = Forms!Facturas![Elaborada Por]
77 ctlLista.Requery
78 End Sub
79
80
81 Sub Comando28_Click()
82 On Error GoTo Err_Comando28_Click
83
84 Dim stDocName As String
85
86 stDocName = "Facturas"
87 DoCmd.DoMenuItem acFormBar, acRecordsMenu, 5, , acMenuVer70
88 DoCmd.OpenReport stDocName, acPreview, , "[N° de Factura] =
Forms!Facturas![N° de Factura]"
89
90 Exit_Comando28_Click:
91 Exit Sub
92
93 Err_Comando28_Click:
94 MsgBox Err.Description
95 Resume Exit_Comando28_Click
96
97 End Sub
98
99 Sub Comando29_Click()
100 On Error GoTo Err_Comando29_Click
101
102 Dim stDocName As String
103
104 stDocName = "Facturas"
105 DoCmd.DoMenuItem acFormBar, acRecordsMenu, 5, , acMenuVer70
106 DoCmd.OpenReport stDocName, acNormal, , "[N° de Factura] =
Forms!Facturas![N° de Factura]"
107
108 Exit_Comando29_Click:
109 Exit Sub
110
111 Err_Comando29_Click:
112 MsgBox Err.Description
113 Resume Exit_Comando29_Click
```

```
114 End Sub
115
116 Private Sub Form_Current()
117     Forms!Facturas.Refresh
118 End Sub
119
120 Private Sub N°_de_Pedido_AfterUpdate()
121     Forms!Facturas![Productos Facturados a Clientes
122     subformulario]![Código del Producto].Requery
123 End Sub
124
125 Private Sub N°_de_Pedido_Enter()
126     Forms!Facturas![Productos Facturados a Clientes
127     subformulario]![Código del Producto].Requery
128 End Sub
129
130 Private Sub N°_de_Pedido_Exit(Cancel As Integer)
131     Forms!Facturas![Productos Facturados a Clientes
132     subformulario]![Código del Producto].Requery
133 End Sub
```



**Figura 17. Figuras Jurídicas**

**Fórmulas**

Nº de Fórmula: 14      Registro Sanitario: MSAS - 90-15789      Departamento: Sólidos

Producto: Acetaminofén      Vehículo:      Concentración: 500      Unidades de Concentración: mg.

Validez: 3      Unidades de Validez: Años      Tamaño del Lote: 500000      Unidades de Presentación: Tabletas

Registro: 14 de 20

**Fórmulas**

Insumos (Asumiendo, donde aplique, potencia del 100%)

Código	Nombre	Cantidad	Unidades
01	Acetaminofén	312,5	Kg.
05	Alcohol Etílico	102,5	Lts.
36	P.V.P. K-30	23,25	Kg.

Registro: 1 de 10

Actualizar Subtotales

Subtotales Kg./Lts.

336,2500	Kg.
102,5000	Lts.

Registro: 14 de 20

Figura 18. Fórmulas (1º y 2º Páginas)

**Fórmulas**

Técnica de Elaboración

- 1.- Revisar los insumos necesarios
- 2.- Preparar la mezcladora.
- 3.- Ajustar la tableteadora a 500 mg.
- 4.- Expandir el principio activo.
- 5.- Añadir los estabilizantes
- 6.- Cargar la mezcladora.
- 7.- Etc...

Fecha de Emisión: 31/08/90      Elaborado por: 03 Rojas, Pedro

Registro: 14 de 20

**Fórmulas**

Revisiones de Fórmulas

Fecha de Revisión	Revisado por	Observaciones
06/06/94	02	Cambio del proceso de entabletado ya que se adquirió una tableteadora automatizada.

Registro: 1 de 1

Editar Departamentos    Editar Productos Genéricos    Editar Vehículos    Editar Unidades de Medición  
 Editar Personas    Editar Insumos    Editar Conversiones

Registro: 14 de 20

Figura 19. Fórmulas (3° y 4° Páginas)

### Código

```

1  Attribute VB_Name = "Form_Fórmulas"
2  Attribute VB_Creatable = True
3  Attribute VB_PredeclaredId = True
4  Attribute VB_Exposed = False
5  Option Compare Database
6  Option Explicit
7
8  Dim ctlLista As Control
12
13 Private Sub Comando94_Click()
14     [Subtotal Kg y Lt Insumos en Fórmulas subformulario].Requery
15 End Sub
16
17 Private Sub Editar_Departamentos_Exit(Cancel As Integer)
18

```

```
19     Set ctlLista = Forms!Fórmulas!Departamento
20     ctlLista.Requery
21 End Sub
22
24 Private Sub Editar_Insumos_Exit(Cancel As Integer)
25     Set ctlLista = Forms!Fórmulas![Insumos en Fórmulas
subformulario]![Código de Insumo]
26     ctlLista.Requery
27 End Sub
28
29 Private Sub Editar_Personas_Exit(Cancel As Integer)
30     Set ctlLista = Forms!Fórmulas![Elaborado por]
31     ctlLista.Requery
32     Set ctlLista = Forms!Fórmulas![Revisiones de Fórmulas]![Revisado
Por]
33     ctlLista.Requery
34 End Sub
35
36 Private Sub Editar_Productos_Genéricos_Exit(Cancel As Integer)
37     Set ctlLista = Forms!Fórmulas!Producto
38     ctlLista.Requery
39 End Sub
40
41 Private Sub Editar_Unidades_de_Medición_Exit(Cancel As Integer)
42     Set ctlLista = Forms!Fórmulas![Unidades de Concentración]
43     ctlLista.Requery
44     Set ctlLista = Forms!Fórmulas![Unidades de Presentación]
45     ctlLista.Requery
46     Set ctlLista = Forms!Fórmulas![Unidades de Validez]
47     ctlLista.Requery
48 End Sub
49
50 Private Sub Editar_Vehículos_Exit(Cancel As Integer)
51     Set ctlLista = Forms!Fórmulas!Vehículo
52     ctlLista.Requery
53 End Sub
54
55
56
57
58
59 Sub Editar_Departamentos_Click()
60 On Error GoTo Err_Editar_Departamentos_Click
61
62     Dim stDocName As String
63     Dim stLinkCriteria As String
64
65     stDocName = "Departamentos"
66     DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria
67
68 Exit_Editar_Departamentos_Click:
69     Exit Sub
70
71 Err_Editar_Departamentos_Click:
72     MsgBox Err.Description
73     Resume Exit_Editar_Departamentos_Click
74
75 End Sub
76
77 Sub Editar_Productos_Genéricos_Click()
```

```
77 On Error GoTo Err_Editar_Productos_Genéricos_Click
78
79 Dim stDocName As String
80 Dim stLinkCriteria As String
81
82 stDocName = "Productos Genéricos"
83 DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria
84
85 Exit_Editar_Productos_Genéricos_Click:
86 Exit Sub
87
88 Err_Editar_Productos_Genéricos_Click:
89 MsgBox Err.Description
90 Resume Exit_Editar_Productos_Genéricos_Click
92 End Sub
93 Sub Editar_Vehículos_Click()
94 On Error GoTo Err_Editar_Vehículos_Click
95
96 Dim stDocName As String
97 Dim stLinkCriteria As String
98
99 stDocName = "Vehículos"
100 DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria
101
102 Exit_Editar_Vehículos_Click:
103 Exit Sub
104
105 Err_Editar_Vehículos_Click:
106 MsgBox Err.Description
107 Resume Exit_Editar_Vehículos_Click
108
109 End Sub
110
111 Sub Editar_Unidades_de_Medición_Click()
112 On Error GoTo Err_Editar_Unidades_de_Medición_Click
113
114 Dim stDocName As String
115 Dim stLinkCriteria As String
116
117 stDocName = "Unidades de Medición"
118 DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria
119
120 Exit_Editar_Unidades_de_Medición_Click:
121 Exit Sub
122
123 Err_Editar_Unidades_de_Medición_Click:
124 MsgBox Err.Description
125 Resume Exit_Editar_Unidades_de_Medición_Click
126
127 End Sub
128 Sub Editar_Personas_Click()
129 On Error GoTo Err_Editar_Personas_Click
130
131 Dim stDocName As String
132 Dim stLinkCriteria As String
```

```
133     stDocName = "Personas"
134     DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria
135
136 Exit_Editar_Personas_Click:
137     Exit Sub
138
139 Err_Editar_Personas_Click:
140     MsgBox Err.Description
141     Resume Exit_Editar_Personas_Click
142
143 End Sub
144 Sub Editar_Insumos_Click()
145 On Error GoTo Err_Editar_Insumos_Click
146
147     Dim stDocName As String
148     Dim stLinkCriteria As String
149
150     stDocName = "Insumos"
151     DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria
152
153 Exit_Editar_Insumos_Click:
154     Exit Sub
155
156 Err_Editar_Insumos_Click:
157     MsgBox Err.Description
158     Resume Exit_Editar_Insumos_Click
159
160 End Sub
161
161 Sub Editar_Conversiones_Click()
162 On Error GoTo Err_Editar_Conversiones_Click
163
164     Dim stDocName As String
165     Dim stLinkCriteria As String
166
167     stDocName = "Conversiones"
168     DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria
169
170 Exit_Editar_Conversiones_Click:
171     Exit Sub
172
173 Err_Editar_Conversiones_Click:
174     MsgBox Err.Description
175     Resume Exit_Editar_Conversiones_Click
176
177 End Sub
178
180 Private Sub Form_AfterDelConfirm(Status As Integer)
181     Forms!Fórmulas.Refresh
182 End Sub
183
184 Sub Imprimir_Informe_Click()
185 On Error GoTo Err_Imprimir_Informe_Click
186
187     Dim stDocName As String
188
```

```
189 stDocName = "Fórmulas"
190
191 DoCmd.DoMenuItem acFormBar, acRecordsMenu, 5, , acMenuVer70
192 DoCmd.OpenReport stDocName, acNormal, , "[N° de Fórmula] = '" &
Forms![Fórmulas]![N° de Fórmula] & "'"
193
194 Exit_Imprimir_Informe_Click:
195 Exit Sub
196
197 Err_Imprimir_Informe_Click:
198 MsgBox Err.Description
199 Resume Exit_Imprimir_Informe_Click
200
201 End Sub
202 Sub Presentar_Informe_Click()
203 On Error GoTo Err_Presentar_Informe_Click
204
205 Dim stDocName As String
206
207 stDocName = "Fórmulas"
208 DoCmd.DoMenuItem acFormBar, acRecordsMenu, 5, , acMenuVer70
209 DoCmd.OpenReport stDocName, acPreview, , "[N° de Fórmula] = '" &
Forms![Fórmulas]![N° de Fórmula] & "'"
210
211 Exit_Presentar_Informe_Click:
212 Exit Sub
213
214 Err_Presentar_Informe_Click:
215 MsgBox Err.Description
216 Resume Exit_Presentar_Informe_Click
217
218 End Sub
```

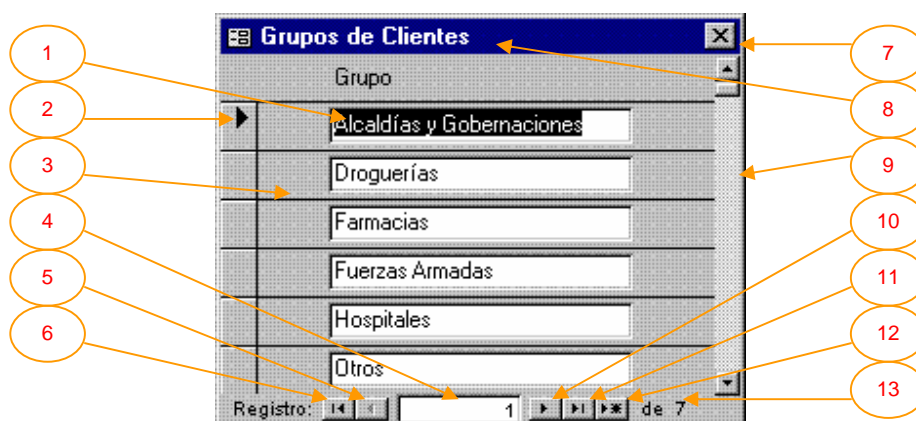


Figura 20. Grupos de Clientes



Este es un formulario continuo, del tipo cuadro de diálogo, que controla los datos provenientes de la tabla [Grupos de Clientes]. Aquí se definen los grupos genéricos de clientes que luego serán adjudicados en la tabla [Clientes] a cada cliente. Estos grupos se utilizan para la elaboración de informes formando conjuntos de registros que pertenecen al mismo grupo. Los controles del formulario, mostrados en la figura con flechas, deben entenderse como sigue:

1. Cuadro de texto. En estos controles deben introducirse los datos. En este caso, el único campo que compone la tabla [Grupos de Clientes] se representa aquí por la columna [Grupo], cada cuadro de texto mostrado, por lo tanto, es un nuevo registro de la tabla. Por defecto, los formularios se abren seleccionando el primer campo, del primer registro, de la tabla base; por esta razón, el primer cuadro de texto mostrado en la figura se encuentra en vídeo inverso. Cualquier porción de datos que se encuentre en vídeo inverso (o equivalentemente se encuentre seleccionado) puede ser automáticamente reemplazada con tan sólo escribir sobre ella o borrada presionando la tecla {Delete}. Si una porción de texto o campo, o registro, es eliminada o modificada por error, la acción puede deshacerse (en la mayoría de los casos) presionando la tecla {Esc}. Un cuadro de texto recibe enfoque cuando se da clic sobre él y en consecuencia el cursor se torna en una línea vertical titilante que indica el modo edición del contenido del cuadro de texto. En el modo de edición de un cuadro de texto funcionan normalmente las teclas {Backspace} (borrar a la izquierda), {Delete} (borrar a la derecha), etc., y adicionalmente hay algunas combinaciones de método abreviado que operan también. Una de las más útiles teclas de método abreviado es la combinación {Ctrl}+{'}. Esta combinación produce la repetición automática del último valor de campo que haya sido introducido. El ratón puede ser utilizado para seleccionar todo o parte del cuadro de texto, dando clic y arrastrando (como es habitual en Windows).
2. Selector de Registro. Este objeto muestra en cada momento el registro de la tabla actualmente seleccionado. Si se da un clic de ratón sobre él, se consigue la selección de todo el registro correspondiente. Si, posteriormente a la selección de un registro, se presiona la tecla {Delete}, éste será eliminado de la tabla (siempre que la integridad de la base de datos lo permita y previa la presentación de un mensaje de advertencia). Como es normal en ambientes gráficos de trabajo, se pueden seleccionar varios objetos a la vez; en este caso, varios registros. Esta operación se realiza dando clic en el primer registro (más exactamente

en el primer selector de registro) y posteriormente dando clic en el último de la selección pero manteniendo presionada la tecla {Shift} ({Shift}+clic). Estando seleccionado un registro o conjunto de registros puede darse un clic con el botón derecho del ratón, en el selector de registros, para acceder a un menú pequeño, donde es posible copiar o cortar (y si esta operación ya fue realizada, pegar) el contenido de los registros. El selector de registro es un botón que adquiere el tamaño de cada registro que representa, como se verá más adelante, eventualmente puede ocupar varias páginas, en los casos en que cada registro ocupe varias páginas para ser presentado.

3. Separador de Registros. Este objeto es una línea divisora que muestra al usuario el área que corresponde a cada registro.
4. Indicador del Número de Registro. Es un cuadro de texto que muestra en cada instante el número del registro actualmente seleccionado o enfocado. En la gráfica muestra el número 1, indicando que el usuario se encuentra operando con el primer registro de la tabla base.
5. Botón de Retroceso Un Registro a la Vez. Este botón pequeño puede ser presionado para retroceder el enfoque, un registro a la vez. En la figura se encuentra inactivo, puesto que no se puede retroceder en la tabla más allá del primer registro.
6. Botón de Retroceso al Primer Registro. Este botón puede ser presionado para retroceder el enfoque, desde cualquier registro, al registro número 1.
7. Botón de Cierre del Formulario. Un formulario es en definitiva una ventana abierta, que debe ser cerrada cuando ya no se le necesite. La operación de cierre de un formulario se realiza dando clic sobre este botón. El sistema realiza, donde sea pertinente, las validaciones necesarias antes de cerrar el formulario y presenta los mensajes apropiados, en consecuencia. En este caso particular, tratándose de un formulario abierto tipo cuadro de diálogo, el usuario deberá cerrarlo antes de poder continuar operando con el resto del sistema. Esto permite al sistema actualizar automáticamente todas aquellas tablas donde se utilice la información proveniente de los [Grupos de Clientes] con los últimos datos.

8. Título del Formulario. Presenta el nombre con que se ha bautizado al formulario. Generalmente (pero no siempre), este nombre coincidirá con el nombre de la tabla que controla.
9. Barra de Desplazamiento Vertical. El área del formulario es limitada y se hace necesario un control que permita desplazarse en ésta para visualizar la información. Este control es la barra de desplazamiento. Dando clic en cualquiera de los botones inferior o superior, se avanza o retrocede (respectivamente) poco a poco en la visual del formulario. Dando clic sobre alguna región interna de la barra, se avanza o retrocede aceleradamente (de página en página). Nótese sin embargo que el movimiento en la visual del formulario no es equivalente al movimiento de registros. Si el usuario desea ir a la página siguiente en el formulario y seleccionar allí un registro posterior, debe mover la visual con la barra de desplazamiento y posteriormente dar clic al registro deseado. Si no da clic sobre el registro, continúa seleccionado el anterior.
10. Botón de Avance Un Registro a la Vez. Este botón pequeño puede ser presionado para avanzar el enfoque, un registro a la vez. Si en la figura se diera clic sobre él, el selector y el indicador de registro avanzarían una posición (hasta el número 2).
11. Botón de Avance al Último Registro. Este botón puede ser presionado para ir al final de la tabla (en el ejemplo mostrado, el registro número 7).
12. Botón de Anexión de Nuevos Registros. Este botón, caracterizado por un asterisco, lleva el enfoque más allá del último registro, hacia un espacio de registro abierto para que el usuario introduzca datos nuevos. El diseño del SARCAI hace un uso extensivo de las claves primarias de tablas; esto implica, entre otras cosas, que el orden interno de los registros será siempre mantenido de acuerdo con la clave y, por lo tanto, sea cual sea el orden en que se introduzcan nuevos datos, una vez actualizada la base de datos (lo que ocurre cuando se cierra el formulario), éstos serán ordenados automáticamente. Por lo tanto, este botón es la mejor forma de iniciar la adición de nuevos registros a la tabla controlada por el formulario.

13. Sección que Muestra el Número de Registros de la Tabla. Este espacio está dedicado, en todos los formularios, a informar al usuario el número total de registros que contiene la tabla controlada.

Figura 21. Insumos

### Código

```

1  Attribute VB_Name = "Form_Insumos"
2  Attribute VB_Creatable = True
3  Attribute VB_PredeclaredId = True
4  Attribute VB_Exposed = False
5  Option Compare Database
6  Option Explicit
7
8  Dim ctlLista As Control, ExisteInicial As Double
9
10 Sub Editar_Personas_Click()
11 On Error GoTo Err_Editar_Personas_Click
12
13     Dim stDocName As String
14     Dim stLinkCriteria As String
15
16     stDocName = "Proveedores"
17     DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria
18
19 Exit_Editar_Personas_Click:
20     Exit Sub
21
22 Err_Editar_Personas_Click:

```

```
23     MsgBox Err.Description
24     Resume Exit_Editar_Personas_Click
25
26 End Sub
27
28 Sub Editar_Proveedores_Click()
29 On Error GoTo Err_Editar_Personas_Click
30     Dim stDocName As String
31     Dim stLinkCriteria As String
32
33     stDocName = "Proveedores"
34     DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria
35
36 Exit_Editar_Personas_Click:
37     Exit Sub
38
39 Err_Editar_Personas_Click:
40     MsgBox Err.Description
41     Resume Exit_Editar_Personas_Click
42
43 End Sub
44
45 Private Sub Editar_Proveedores_Exit(Cancel As Integer)
46     Set ctlLista = Forms!Insumos![Precios de Insumos]
47     ctlLista.Requery
48 End Sub
49
50 Sub Editar_Tipos_de_Insumos_Click()
51 On Error GoTo Err_Editar_Tipos_de_Insumos_Click
52
53     Dim stDocName As String
54     Dim stLinkCriteria As String
55
56     stDocName = "Tipos de Insumos"
57     DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria
58
59 Exit_Editar_Tipos_de_Insumos_Click:
60     Exit Sub
61
62 Err_Editar_Tipos_de_Insumos_Click:
63     MsgBox Err.Description
64     Resume Exit_Editar_Tipos_de_Insumos_Click
65
66 End Sub
67
68 Private Sub Editar_Tipos_de_Insumos_Exit(Cancel As Integer)
69     Set ctlLista = Forms!Insumos![Tipo de Insumo]
70     ctlLista.Requery
71 End Sub
72
73 Sub Editar_Unidades_de_Presentación_Click()
74 On Error GoTo Err_Editar_Unidades_de_Presentación_Click
75
76     Dim stDocName As String
77     Dim stLinkCriteria As String
78
79 End Sub
80
```

```

81     stDocName = "Unidades de Medición"
82     DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria
83
84     Exit_Editar_Unidades_de_Presentación_Cli:
85     Exit Sub
86
87     Err_Editar_Unidades_de_Presentación_Click:
88     MsgBox Err.Description
89     Resume Exit_Editar_Unidades_de_Presentación_Cli
90
91 End Sub
92
93 Private Sub Editar_Unidades_de_Presentación_Exit(Cancel As Integer)
94     Set ctlLista = Forms!Insumos![Unidades de Presentación]
95     ctlLista.Requery
96 End Sub

```

Figura 22. Llegadas de Insumos al Almacén (1° Página)

Figura 23. Llegadas de Insumos al Almacén (2° Página)

## Código

```

1  Attribute VB_Name = "Form_Llegadas de Insumos al Almacén"
2  Attribute VB_Creatable = True
3  Attribute VB_PredeclaredId = True
4  Attribute VB_Exposed = False
5  Option Compare Database
6  Option Explicit
7
8  Dim ctlLista As Control, LlegadaActual As Single
9
10 Private Sub Cantidad_Enter()
11     LlegadaActual = Cantidad
12 End Sub
13
14 Private Sub Cantidad_Exit(Cancel As Integer)
15     If LlegadaActual <> Cantidad Then
16         If Forms![Llegadas de Insumos al Almacén]![Subformulario
Cantidad Recibida de Insumo por Orden]![Resta] + LlegadaActual <
Cantidad Then
17             MsgBox "La Cantidad que llega NO puede exceder la Cantidad que
resta por llegar de la Orden"
18             Cantidad = LlegadaActual
19         End If
20     If Not IsNull(Forms![Llegadas de Insumos al
Almacén]![Subformulario Salidas del Almacén de Insumos]![Fecha]) _
21         And Not IsNull(Forms![Llegadas de Insumos al
Almacén]![Subformulario Salidas del Almacén de Insumos]![Tipo de
Salida])


```

```
22     And Not IsNull(Forms![Llegadas de Insumos al
Almacén]![Subformulario Salidas del Almacén de Insumos]![Razón])
Then
23     If Forms![Llegadas de Insumos al Almacén]![Subformulario
Salidas del Almacén de Insumos]![Subformulario Cantidad que ha
Salido de Insumo por Lote]![Salidas] > Cantidad Then
24         MsgBox "La Cantidad que llega NO puede ser inferior a las
Salidas. Revise las salidas antes de actualizar las Llegadas"
25         Cantidad = LlegadaActual
26     End If
27 End If
28     Forms![Llegadas de Insumos al Almacén].Refresh
29 End If
30 End Sub
31
32 Private Sub Editar_Órdenes_de_Compra_Exit(Cancel As Integer)
33     Set ctlLista = Forms![Llegadas de Insumos al Almacén]![Orden N°]
34     ctlLista.Requery
35 End Sub
36
37 Private Sub Editar_Personas_Exit(Cancel As Integer)
38     Set ctlLista = Forms![Llegadas de Insumos al Almacén]![Recibido
Por]
39     ctlLista.Requery
40     Set ctlLista = Forms![Llegadas de Insumos al
Almacén]![Subformulario Salidas del Almacén de Insumos]![Autoriza
por Producción]
41     ctlLista.Requery
42     Set ctlLista = Forms![Llegadas de Insumos al
Almacén]![Subformulario Salidas del Almacén de Insumos]![Autoriza
por Control]
43     ctlLista.Requery
44     Set ctlLista = Forms![Llegadas de Insumos al
Almacén]![Subformulario Salidas del Almacén de Insumos]![Ejecuta]
45     ctlLista.Requery
46 End Sub
47
48 Private Sub Editar_Razones_de_Salidas_Exit(Cancel As Integer)
49     Set ctlLista = Forms![Llegadas de Insumos al
Almacén]![Subformulario Salidas del Almacén de Insumos]![Razón]
50     ctlLista.Requery
51 End Sub
52
53 Private Sub Editar_Tipos_de_Salidas_Exit(Cancel As Integer)
54     Set ctlLista = Forms![Llegadas de Insumos al
Almacén]![Subformulario Salidas del Almacén de Insumos]![Tipo de
Salida]
55     ctlLista.Requery
56 End Sub
57
58 Private Sub Estados_de_Insumos_Exit(Cancel As Integer)
59     Set ctlLista = Forms![Llegadas de Insumos al Almacén]!Estado
60     ctlLista.Requery
61 End Sub
62
63 Private Sub Insumo_AfterUpdate()
```



```
64     Cantidad = 0
65     Forms![Llegadas de Insumos al Almacén].Refresh
66     Cantidad = Forms![Llegadas de Insumos al Almacén]![Subformulario
Cantidad Recibida de Insumo por Orden]![Resta]
67     Forms![Llegadas de Insumos al Almacén]![Subformulario Cantidad
Recibida de Insumo por Orden].Requery
68     Forms![Llegadas de Insumos al Almacén].Refresh
69 End Sub
70
71 Private Sub Orden_Nº_AfterUpdate()
72     Forms![Llegadas de Insumos al Almacén]!Insumo = " "
73     Set ctlLista = Forms![Llegadas de Insumos al Almacén]!Insumo
ctlLista.Requery
74 End Sub
75
76
77
78 Sub Editar_Órdenes_de_Compra_Click()
79 On Error GoTo Err_Editar_Órdenes_de_Compra_Click
80
81     Dim stDocName As String
82     Dim stLinkCriteria As String
83
84     stDocName = "Órdenes de Compra"
85     DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria
86
87 Exit_Editar_Órdenes_de_Compra_Click:
88     Exit Sub
89
90 Err_Editar_Órdenes_de_Compra_Click:
91     MsgBox Err.Description
92     Resume Exit_Editar_Órdenes_de_Compra_Click
93
94 End Sub
95
96 Sub Estados_de_Insumos_Click()
97 On Error GoTo Err_Estados_de_Insumos_Click
98
99     Dim stDocName As String
100    Dim stLinkCriteria As String
101
102    stDocName = "Estados de Insumos"
103    DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria
104
105 Exit_Estados_de_Insumos_Click:
106     Exit Sub
107
108 Err_Estados_de_Insumos_Click:
109     MsgBox Err.Description
110     Resume Exit_Estados_de_Insumos_Click
111
112 End Sub
113
114 Sub Editar_Personas_Click()
115 On Error GoTo Err_Editar_Personas_Click
116
117     Dim stDocName As String
118     Dim stLinkCriteria As String
119
120     stDocName = "Personas"
```

```
119 DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria
120
121 Exit_Editar_Personas_Click:
122 Exit Sub
123
124 Err_Editar_Personas_Click:
125 MsgBox Err.Description
126 Resume Exit_Editar_Personas_Click
127
128 End Sub
129 Private Sub Editar_Tipos_de_Salidas_Click()
130 On Error GoTo Err_Editar_Tipos_de_Salidas_Click
131
132 Dim stDocName As String
133 Dim stLinkCriteria As String
134
135 stDocName = "Tipos de Salidas"
136 DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria
137
138 Exit_Editar_Tipos_de_Salidas_Click:
139 Exit Sub
140
141 Err_Editar_Tipos_de_Salidas_Click:
142 MsgBox Err.Description
143 Resume Exit_Editar_Tipos_de_Salidas_Click
144
145 End Sub
146 Private Sub Editar_Razones_de_Salidas_Click()
147 On Error GoTo Err_Editar_Razones_de_Salidas_Click
148
149 Dim stDocName As String
150 Dim stLinkCriteria As String
151
152 stDocName = "Razones de Salidas"
153 DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria
154
155 Exit_Editar_Razones_de_Salidas_Click:
156 Exit Sub
157
158 Err_Editar_Razones_de_Salidas_Click:
159 MsgBox Err.Description
160 Resume Exit_Editar_Razones_de_Salidas_Click
161
162 End Sub
```

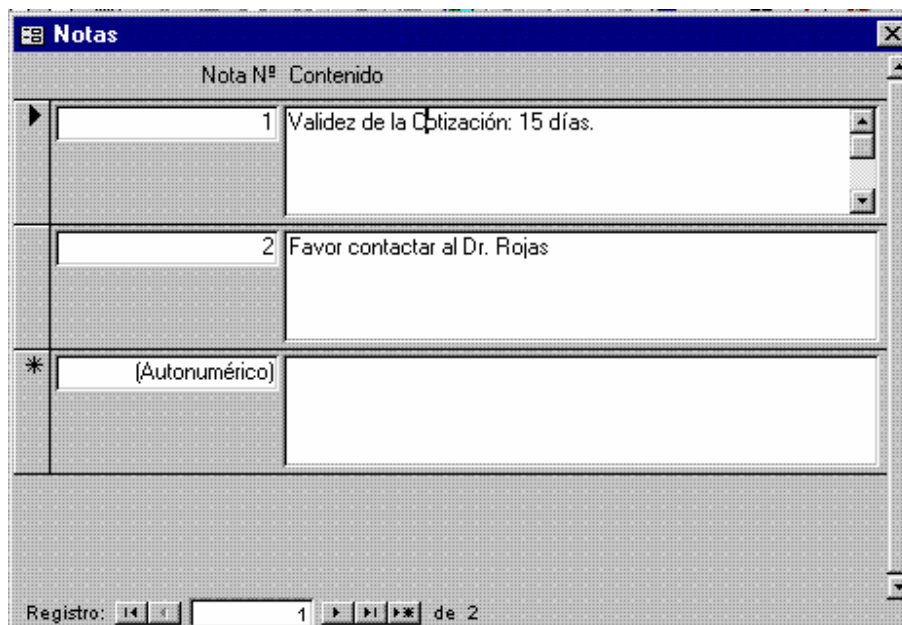


The screenshot shows a window titled "Magnitudes Físicas" with a list of physical magnitudes. The list has a header "Magnitud Física" and contains the following entries:

Magnitud Física
Peso
Tiempo
Unidad
Volumen
*

At the bottom of the window, there is a status bar that reads "Registro: 1 de 4".

Figura 24. Magnitudes Físicas



The screenshot shows a window titled "Notas" with a list of notes. The list has a header "Nota N° Contenido" and contains the following entries:

Nota N°	Contenido
1	Validez de la Certificación: 15 días.
2	Favor contactar al Dr. Rojas
*	(Autonómico)

At the bottom of the window, there is a status bar that reads "Registro: 1 de 2".

Figura 25. Notas

**Órdenes de Compra**

Nº: 1      Proveedor: 001      Fecha: 10/10/97

Datos del Proveedor

Nombre o Razón Social: Importadora de Insumos Biológicos OMEGA C.A.      Cédula de Identidad o RIF: 150-4589-6

Dirección: Zona Industrial "El Puente Sobre El Lago". Carrera 4. Local 23.

País: Venezuela      Estado: Zulia      Pueblo o Ciudad: Maracaibo

Registro: 1 de 8

**Órdenes de Compra**

Insumos Solicitados

Insumo	Cantidad	Precio Unitario	Precio Total
<b>Acetaminofén</b>			
Presentación Mínima: 1      Unid.: Kg			
01	500	Bs.5.000,00	Bs.2.500.000,00
<b>Alcohol Etílico</b>			
Presentación Mínima: 1      Unid.: Lts.			
05	20	Bs.358,82	Bs.7.176,40

Registro: 1 de 10

           Subtotal: Bs.3.688.776,00  
            Cargos %: 10,00%  
 Subtotal\*Cargos: Bs.368.877,61  
**Total: Bs.4.057.653,61**

Registro: 1 de 8

Figura 26. Órdenes de Compra (1° y 2° Páginas)

### Código

```

1 Attribute VB_Name = "Form_Órdenes de Compra"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7

```

```
8 Dim ctlLista As Control
17
18 Private Sub Editar_Insumos_Exit(Cancel As Integer)
19 Set ctlLista = Forms![Órdenes de Compra]![Insumos en
Órdenes]!Insumo
20 ctlLista.Requery
21 End Sub
22
23 Private Sub Editar_Proveedores_Exit(Cancel As Integer)
24 Set ctlLista = Forms![Órdenes de Compra]!Proveedor
25 ctlLista.Requery
26 End Sub
27
28
29
30
31
32
33
34 Sub Imprimir_Orden_Click()
35 On Error GoTo Err_Imprimir_Orden_Click
36
37 Dim stDocName As String
38
39 stDocName = "Órdenes de Compra"
40 DoCmd.DoMenuItem acFormBar, acRecordsMenu, 5, , acMenuVer70
41 DoCmd.OpenReport stDocName, acNormal, , "[N°] = " &
Str(Forms![Órdenes de Compra]![N°])
42
43 Exit_Imprimir_Orden_Click:
44 Exit Sub
45
46 Err_Imprimir_Orden_Click:
47 MsgBox Err.Description
48 Resume Exit_Imprimir_Orden_Click
49
50 End Sub
51 Sub Comando43_Click()
52 On Error GoTo Err_Comando43_Click
53
54 Dim stDocName As String
55
56 stDocName = "Órdenes de Compra"
57 DoCmd.DoMenuItem acFormBar, acRecordsMenu, 5, , acMenuVer70
58 DoCmd.OpenReport stDocName, acPreview, , "[N°] = " &
Str(Forms![Órdenes de Compra]![N°])
59
60 Exit_Comando43_Click:
61 Exit Sub
62
63 Err_Comando43_Click:
64 MsgBox Err.Description
65 Resume Exit_Comando43_Click
66
67 End Sub
68 Sub Editar_Proveedores_Click()
69 On Error GoTo Err_Editar_Proveedores_Click
70
71 Dim stDocName As String
72 Dim stLinkCriteria As String
73
74 stDocName = "Proveedores"
```

```
75     DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria
76
77 Exit_Editar_Proveedores_Click:
78     Exit Sub
79
80 Err_Editar_Proveedores_Click:
81     MsgBox Err.Description
82     Resume Exit_Editar_Proveedores_Click
83
84 End Sub
85 Sub Editar_Insumos_Click()
86 On Error GoTo Err_Editar_Insumos_Click
87
88     Dim stDocName As String
89     Dim stLinkCriteria As String
90
91     stDocName = "Insumos"
92     DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria
93
94 Exit_Editar_Insumos_Click:
95     Exit Sub
96
97 Err_Editar_Insumos_Click:
98     MsgBox Err.Description
99     Resume Exit_Editar_Insumos_Click
100
101 End Sub
102 Sub Comando49_Click()
103 On Error GoTo Err_Comando49_Click
104
105
106     DoCmd.DoMenuItem acFormBar, acRecordsMenu, 5, , acMenuVer70
107
108 Exit_Comando49_Click:
109     Exit Sub
110
111 Err_Comando49_Click:
112     MsgBox Err.Description
113     Resume Exit_Comando49_Click
114
115 End Sub
116 Sub Actualizar_Montos_Click()
117 On Error GoTo Err_Actualizar_Montos_Click
118
119
120     DoCmd.DoMenuItem acFormBar, acRecordsMenu, 5, , acMenuVer70
121
122 Exit_Actualizar_Montos_Click:
123     Exit Sub
124
125 Err_Actualizar_Montos_Click:
126     MsgBox Err.Description
127     Resume Exit_Actualizar_Montos_Click
128
129 End Sub
```

**Órdenes de Producción**

Nº de Orden:  Producto:

Datos del Producto y la Fórmula

Departamento	Fórmula N°	Destino
<input type="text" value="Sólidos"/>	<input type="text" value="14"/>	<input type="text" value="Venta"/>

Producto	Vehículo	Concentración
<input type="text" value="Acetaminofén"/>	<input type="text"/>	<input type="text" value="500 mg"/>

Presentación	Tamaño del Lote
<input type="text" value="20"/> <input type="text" value="Tabletas"/>	<input type="text" value="500000"/>

¿De Producción?    
 ¿De Acondicionamiento?

Requerido Para:    
 Fecha de Emisión:

Cantidad:     
 Unidades Terminadas:

Registro:  de 2

**Órdenes de Producción**


Insumos Requeridos

Insumo	Para Lote Estándar	Para la Orden	Unidades
Algodón Hidrófilo	<input type="text" value="4"/>	<input type="text" value="4"/>	<input type="text" value="Folios"/>
Tipo: <input type="text" value="Empaque Principal"/>			¿Se Controla Potencia? <input type="checkbox"/>

Lotes y Cantidades Solicitadas

Lote N°	Cantidad en Lote	Cantidad Sugerida	Surtida el
<input type="text" value="5"/>	<input type="text" value="10"/>	<input type="text" value="4"/>	<input type="text" value="15/10/97"/>

Registro:  de 1

Total del Insumo Utilizado en la Orden:     
 Resta:

Registro:  de 10

Subtotales Kg/Lts:

Registro:  de 2

Figura 27. Órdenes de Producción (1° y 2° Páginas)

**Órdenes de Producción**

Emitida Por: 02 Chanan, Gasem  
 Autorizada Por: 03 Rojas, Pedro  
 Recibida Por: 01 Marquina, Eduardo  
 Despachado Por: 03 Rojas, Pedro  
 Contabilizada Por: 01 Marquina, Eduardo

Observaciones:  
 Tiene prioridad debido a los atrasos en entregas de este producto



Datos de la Producción y Control de Calidad

Lote N°:	Fecha de Inicio:	Fecha de Término:	Horas Hombre:
001	15/10/97	17/10/97	120
Fecha de Vencimiento:	Cantidad de Muestras:	Horas de Análisis:	Rendimiento:
17/10/00	3	5	15
Protocolo de Análisis:	Fecha de Aprobación:		
ASR-4000	17/10/97		
Aprobado Por:	02 Chanan, Gasem		

Registro: 1 de 2

**Órdenes de Producción**

Cantidad Efectivamente Producida: 498764 Tabletas  
 Unidades Efectivamente Terminadas: 24938,2 Frascos Plásticos

Editar Productos Terminados    Editar Llegadas de Insumos    Editar Personas        

Registro: 1 de 2

Figura 28. Órdenes de Producción (3° y 4° Páginas)

### Código

```

1 Attribute VB_Name = "Form_Órdenes de Producción"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7

```



```
8 Dim ctlLista As Control
9 Private Sub Cantidad_AfterUpdate()
10 [Presentación] = Forms![Órdenes de Producción]![Productos
Terminados y Fórmulas Consulta subformulario]![Presentación]
11 [Unidades Terminadas] = [Cantidad] / Forms![Órdenes de
Producción]![Productos Terminados y Fórmulas Consulta
subformulario]![Cantidad en Presentación]
12 [Producción Efectiva] = [Cantidad]
13 Forms![Órdenes de Producción].Refresh
14 End Sub
15
16 Private Sub Editar_Compras_de_Insumos_Exit(Cancel As Integer)
17 Set ctlLista = Forms![Órdenes de Producción]![Productos Terminados
y Fórmulas Consulta subformulario1]
18 ctlLista.Requery
19 End Sub
20
21 Private Sub Editar_Personas_Exit(Cancel As Integer)
22 Set ctlLista = Forms![Órdenes de Producción]![Emitida Por]
23 ctlLista.Requery
24 Set ctlLista = Forms![Órdenes de Producción]![Despachado Por]
25 ctlLista.Requery
26 Set ctlLista = Forms![Órdenes de Producción]![Recibida Por]
27 ctlLista.Requery
28 Set ctlLista = Forms![Órdenes de Producción]![Autorizada Por]
29 ctlLista.Requery
30 Set ctlLista = Forms![Órdenes de Producción]![Contabilizada Por]
31 ctlLista.Requery
32 End Sub
33
34 Private Sub Editar_Productos_Terminados_Exit(Cancel As Integer)
35 Set ctlLista = Forms![Órdenes de Producción]!Producto
36 ctlLista.Requery
37 End Sub
38
39 Private Sub Form_Current()
40 If Not IsNull(Producto) Then
41 [Unidades de Presentación] = Forms![Órdenes de
Producción]![Productos Terminados y Fórmulas Consulta
42 [Presentación] = Forms![Órdenes de Producción]![Productos
Terminados y Fórmulas Consulta subformulario]![Presentación]
43 [Unidades Terminadas] = [Cantidad] / Forms![Órdenes de
Producción]![Productos Terminados y Fórmulas Consulta
subformulario]![Cantidad en Presentación]
44 [Presentación E] = Forms![Órdenes de Producción]![Productos
Terminados y Fórmulas Consulta subformulario]![Presentación]
45 [Unidades Terminadas E] = [Producción Efectiva] / Forms![Órdenes
de Producción]![Productos Terminados y Fórmulas Consulta
subformulario]![Cantidad en Presentación]
46 End If
47 End Sub
48
49 Private Sub Producción_Efectiva_AfterUpdate()
50 [Presentación E] = Forms![Órdenes de Producción]![Productos
Terminados y Fórmulas Consulta subformulario]![Presentación]
51 [Unidades Terminadas E] = [Producción Efectiva] / Forms![Órdenes
```

```
de Producción![[Productos Terminados y Fórmulas Consulta
subformulario![[Cantidad en Presentación]
52 End Sub
53
54 Private Sub Producto_AfterUpdate()
55     Forms![[Órdenes de Producción].Refresh
56     Cantidad = Forms![[Órdenes de Producción]![[Productos Terminados y
Fórmulas Consulta subformulario]![[Tamaño del Lote]
57     [Unidades de Presentación] = Forms![[Órdenes de
Producción]![[Productos Terminados y Fórmulas Consulta
subformulario]![[Unidades de Presentación]
58     [Presentación] = Forms![[Órdenes de Producción]![[Productos
Terminados y Fórmulas Consulta subformulario]![[Presentación]
59     [Unidades Terminadas] = [Cantidad] / Forms![[Órdenes de
Producción]![[Productos Terminados y Fórmulas Consulta
subformulario]![[Cantidad en Presentación]
60     [Producción Efectiva] = [Cantidad]
61     [Presentación E] = Forms![[Órdenes de Producción]![[Productos
Terminados y Fórmulas Consulta subformulario]![[Presentación]
62     [Unidades Terminadas E] = [Producción Efectiva] / Forms![[Órdenes
de Producción]![[Productos Terminados y Fórmulas Consulta
subformulario]![[Cantidad en Presentación]
63     Forms![[Órdenes de Producción].Refresh
64 End Sub
65
66 Sub Editar_Productos_Terminados_Click()
67 On Error GoTo Err_Editar_Productos_Terminados_Click
68
69     Dim stDocName As String
70     Dim stLinkCriteria As String
71
72     stDocName = "Productos Terminados"
73     DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria
74
75 Exit_Editar_Productos_Terminados_Click:
76     Exit Sub
77
78 Err_Editar_Productos_Terminados_Click:
79     MsgBox Err.Description
80     Resume Exit_Editar_Productos_Terminados_Click
81
82 End Sub
83 Sub Editar_Compras_de_Insumos_Click()
84 On Error GoTo Err_Editar_Compras_de_Insumos_Click
85
86     Dim stDocName As String
87     Dim stLinkCriteria As String
88
89     stDocName = "Llegadas de Insumos al Almacén"
90     DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria
91
92 Exit_Editar_Compras_de_Insumos_Click:
93     Exit Sub
94
95 Err_Editar_Compras_de_Insumos_Click:
96     MsgBox Err.Description
```

```
97     Resume Exit_Editar_Compras_de_Insumos_Click
98
99 End Sub
100 Sub Editar_Personas_Click()
101 On Error GoTo Err_Editar_Personas_Click
102
103     Dim stDocName As String
104     Dim stLinkCriteria As String
105
106     stDocName = "Personas"
107     DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria
108
109 Exit_Editar_Personas_Click:
110     Exit Sub
111
112 Err_Editar_Personas_Click:
113     MsgBox Err.Description
114     Resume Exit_Editar_Personas_Click
115
116 End Sub
117 Sub Presentar_Orden_Click()
118 On Error GoTo Err_Presentar_Orden_Click
119
120     Dim stDocName As String
121
122     stDocName = "Órdenes de Producción"
123     DoCmd.DoMenuItem acFormBar, acRecordsMenu, 5, , acMenuVer70
124     DoCmd.OpenReport stDocName, acPreview, , "[Nº de Orden] =
Forms![Órdenes de Producción]![Nº de Orden]"
125
126 Exit_Presentar_Orden_Click:
127     Exit Sub
128
129 Err_Presentar_Orden_Click:
130     MsgBox Err.Description
131     Resume Exit_Presentar_Orden_Click
132
133 End Sub
134 Sub Imprimir_Orden_Click()
135 On Error GoTo Err_Imprimir_Orden_Click
136
137     Dim stDocName As String
138
139     stDocName = "Órdenes de Producción"
140     DoCmd.DoMenuItem acFormBar, acRecordsMenu, 5, , acMenuVer70
141     DoCmd.OpenReport stDocName, acNormal, , "[Nº de Orden] =
Forms![Órdenes de Producción]![Nº de Orden]"
142
143 Exit_Imprimir_Orden_Click:
144     Exit Sub
145
146 Err_Imprimir_Orden_Click:
147     MsgBox Err.Description
148     Resume Exit_Imprimir_Orden_Click
149
150 End Sub
```

The screenshot shows a dialog box titled "Países" with a blue title bar. It contains a form with the following structure:

- País:** A text field containing "Venezuela" and a "Código Telefónico" field containing "058".
- Estados:** A sub-form containing an "Estado" field with "Mérida".
- Ciudades:** A table with columns "Ciudad" and "Código Telefónico". It contains three rows:
 

Ciudad	Código Telefónico
Mérida	074
El Vigía	075
*	
- Record Navigation:** At the bottom of each sub-form, there are "Registro:" labels and navigation buttons. The "Ciudades" table shows "Registro: 2 de 2", the "Estados" sub-form shows "Registro: 1 de 3", and the main "Países" form shows "Registro: 1 de 1".

Numbered callouts (1-6) point to specific elements: 1 points to the "País" label, 2 to the "País" text field, 3 to the "Estados" label, 4 to the "Ciudades" table, 5 to the "Registro:" label of the "Ciudades" table, and 6 to the "Registro:" label of the "Estados" sub-form.

Figura 29. Países

Este es un formulario continuo, del tipo cuadro de diálogo, que introduce por primera vez una variante importante. Controla los datos provenientes de tres tablas simultáneamente, a saber: [Países], [Estados de Países] y [Ciudades de Estados]. En realidad, el formulario se basa en la tabla [Países] e incluye un subformulario, enlazado con el primero, que controla los estados dentro de cada país, y en éste un sub-subformulario, que controla las ciudades dentro de los estados, dentro de los países. Cada vez que el usuario intenta introducir una nueva ciudad, se le exige circunscribirla a un estado y a su vez se exige que el estado esté circunscrito a un país. Sin embargo, el usuario no debe complicarse adjudicando códigos de estados y países en ciudades o de países en estados, el sistema (y el diseño de la base de datos) automáticamente asigna a las tablas convenientes, los códigos apropiados. Por ejemplo, una vez que se ha introducido el país "Venezuela", cuando se intenta introducir el estado "Mérida", el sistema automáticamente asigna al campo [País] de la tabla [Estados de Países], el valor "Venezuela" (que es el campo de enlace). Lo propio se realiza con las [Ciudades de Estados].

Vale la pena observar que un formulario puede contener varios subformularios y dentro de cada uno de ellos, sub-subformularios.<sup>1</sup>

Los elementos novedosos en este caso son:

1. Selector de Registros del Formulario. Ya se ha mencionado el significado y uso del selector de registros. En un formulario sencillo, se aplica lo dicho anteriormente, sin embargo, en este caso, el selector de registros tiene más usos. En primer término, tiene el tamaño apropiado para abarcar todos los contenidos de datos que representan un registro (prácticamente toda la página). Por otra parte, cuando aquí se selecciona un registro, en realidad se está seleccionando un registro de [Países], todos los registros asociados con él, de [Estados de Países] y todos los asociados con los estados en [Ciudades de Estados]. Por consiguiente, si se desea borrar el registro del formulario utilizando el selector de registros, no sólo se eliminará un registro de [Países], sino todos los estados asociados a él y todas las ciudades asociadas con cada estado de aquellos.
2. Selector de Registros del Subformulario. De forma recursiva, el tratamiento del sistema a los formularios se aplica a los subformularios. Así, este selector de registros se comporta como fue explicado, con la salvedad de que su ámbito de acción se restringe a los registros controlados por el subformulario (y no todos los registros). Por esta razón, cuando el usuario selecciona con este selector un registro, en realidad está seleccionando un estado y todas las ciudades definidas para él, y si luego decide eliminar el registro, elimina el estado y todas las ciudades de éste, pero no elimina el país.
3. Selector de Registros del Sub-subformulario. De forma similar a lo expuesto antes, este selector sólo toca las ciudades (y no los estados, ni los países) y por consiguiente, cualquier selección hecha con él, por ejemplo, para eliminar registros, eliminará únicamente ciudades dejando intactos los estados y los países.

---

<sup>1</sup> El Access permite un anidamiento máximo de tres niveles en subformularios

4. Sección de Desplazamiento del Sub-subformulario. Como ya se imaginará el lector, los botones de desplazamiento señalados aquí permiten desplazarse entre los registros de las ciudades dejando fijos los registros de estados y países.
5. Sección de Desplazamiento del Subformulario. En este caso, el desplazamiento se refiere a los estados, pero afecta, por supuesto, el desplazamiento de las ciudades mas no el de los países.
6. Sección de Desplazamiento del Formulario. Aquí, el desplazamiento se refiere a los países y afecta a los estados y las ciudades.

The screenshot shows a window titled "Pedidos de Clientes" with a blue title bar. The form contains the following fields and values:

- Nº de Pedido:** 5
- Código del Cliente:** 100202
- Fecha:** 11/10/97
- Vendedor:** 03 Rojas, Pedro
- Datos del Cliente:**
  - Nombre o Razón Social:** Farmacia "La Buena Salud" S.R.L.
  - Cédula de Identidad o RIF:** 345-65789
  - Dirección:** Av. 4 entre calles 16 y 17. Centro Comercial "La Cuarta Avenida".
  - País:** Venezuela
  - Estado:** Mérida
  - Pueblo o Ciudad:** Mérida
- Condiciones:** Contado
- Tipo de Orden:** Respaldada
- Nº de Orden:** 15263
- Observaciones:** El cliente exige una máxima demora de 15 días, caso contrario retirará el pedido.

At the bottom, there is a navigation bar with the text "Registro: 5 de 13" and several navigation icons.

Figura 30. Pedidos de Clientes (1º Página)

Producto	Cantidad
001 Acetaminofén 500 mg. x 20 Tabletas	250
002 Acido Fólico 5 mg. x 20 Tabletas	175
003 Adenar Emulsión 120 ml. x 1 Frascos	210
* [ ]	[ ]

Registro: 1 de 3

Editar Tipos de Órdenes    Editar Clientes    Editar Personas    Editar Productos Terminados    Editar Condiciones

Registro: 5 de 13

Figura 31. Pedidos de Clientes (2° Página)

## Código

```

1  Attribute VB_Name = "Form_Pedidos de Clientes"
2  Attribute VB_Creatable = True
3  Attribute VB_PredeclaredId = True
4  Attribute VB_Exposed = False
5  Option Compare Database
6  Option Explicit
7
8  Dim ctlLista As Control
9  Sub Editar_Clientes_Click()
10 On Error GoTo Err_Editar_Clientes_Click
11
12     Dim stDocName As String
13     Dim stLinkCriteria As String
14
15     stDocName = "Clientes"
16     DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria
17
18 Exit_Editar_Clientes_Click:
19     Exit Sub
20
21 Err_Editar_Clientes_Click:
22     MsgBox Err.Description
23     Resume Exit_Editar_Clientes_Click
24

```

```
25 End Sub
26
27 Private Sub Editar_Clientes_Exit(Cancel As Integer)
28     Set ctlLista = Forms![Pedidos de Clientes]![Código del Cliente]
29     ctlLista.Requery
30 End Sub
31
32 Private Sub Editar_Condiciones_Exit(Cancel As Integer)
33     Set ctlLista = Forms![Pedidos de Clientes]![Condición]
34     ctlLista.Requery
35 End Sub
36
37 Sub Editar_Personas_Click()
38 On Error GoTo Err_Editar_Personas_Click
39
40     Dim stDocName As String
41     Dim stLinkCriteria As String
42
43     stDocName = "Personas"
44     DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria
45
46 Exit_Editar_Personas_Click:
47     Exit Sub
48
49 Err_Editar_Personas_Click:
50     MsgBox Err.Description
51     Resume Exit_Editar_Personas_Click
52
53 End Sub
54
55 Private Sub Editar_Personas_Exit(Cancel As Integer)
56     Set ctlLista = Forms![Pedidos de Clientes]![Vendedor]
57     ctlLista.Requery
58 End Sub
59
60 Sub Editar_Productos_Terminados_Click()
61 On Error GoTo Err_Editar_Productos_Terminados_Click
62
63     Dim stDocName As String
64     Dim stLinkCriteria As String
65
66     stDocName = "Productos Terminados"
67     DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria
68
69 Exit_Editar_Productos_Terminados_Click:
70     Exit Sub
71
72 Err_Editar_Productos_Terminados_Click:
73     MsgBox Err.Description
74     Resume Exit_Editar_Productos_Terminados_Click
75
76 End Sub
77 Sub Editar_Condiciones_Click()
78 On Error GoTo Err_Editar_Condiciones_Click
79
80     Dim stDocName As String
```



```
81     Dim stLinkCriteria As String
82
83     stDocName = "Condiciones"
84     DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria
85
86 Exit_Editar_Condiciones_Click:
87     Exit Sub
88
89 Err_Editar_Condiciones_Click:
90     MsgBox Err.Description
91     Resume Exit_Editar_Condiciones_Click
92
93 End Sub
94
95 Private Sub Editar_Productos_Terminados_Exit(Cancel As Integer)
96     Set ctlLista = Forms![Pedidos de Clientes]![Productos Pedidos de
Clientes subformulario]![Código del Producto]
97     ctlLista.Requery
98 End Sub
99
100 Private Sub Editar_Tipos_de_Órdenes_Exit(Cancel As Integer)
101     Set ctlLista = Forms![Pedidos de Clientes]![Tipo de Orden]
102     ctlLista.Requery
103 End Sub
104
105 Sub Presentar_Pedidos_Click()
106 On Error GoTo Err_Presentar_Pedidos_Click
107
108     Dim stDocName As String
109
110     stDocName = "Pedidos de Clientes"
111     DoCmd.DoMenuItem acFormBar, acRecordsMenu, 5, , acMenuVer70
112     DoCmd.OpenReport stDocName, acPreview, , "[N° de Pedido] = " &
Str(Forms![Pedidos de Clientes]![N° de Pedido])
113
114 Exit_Presentar_Pedidos_Click:
115     Exit Sub
116
117 Err_Presentar_Pedidos_Click:
118     MsgBox Err.Description
119     Resume Exit_Presentar_Pedidos_Click
120
121 End Sub
122 Sub Imprimir_Pedidos_Click()
123 On Error GoTo Err_Imprimir_Pedidos_Click
124
125     Dim stDocName As String
126
127     stDocName = "Pedidos de Clientes"
128     DoCmd.DoMenuItem acFormBar, acRecordsMenu, 5, , acMenuVer70
129     DoCmd.OpenReport stDocName, acNormal, , "[N° de Pedido] = " &
Str(Forms![Pedidos de Clientes]![N° de Pedido])
130
131 Exit_Imprimir_Pedidos_Click:
132     Exit Sub
133
```

```

134 Err_Imprimir_Pedidos_Click:
135     MsgBox Err.Description
136     Resume Exit_Imprimir_Pedidos_Click
137
138 End Sub
139 Private Sub Editar_Tipos_de_Órdenes_Click()
140 On Error GoTo Err_Editar_Tipos_de_Órdenes_Click
141
142     Dim stDocName As String
143     Dim stLinkCriteria As String
144
145     stDocName = "Tipos de Órdenes de Pedido"
146     DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria
147
148 Exit_Editar_Tipos_de_Órdenes_Click:
149     Exit Sub
150
151 Err_Editar_Tipos_de_Órdenes_Click:
152     MsgBox Err.Description
153     Resume Exit_Editar_Tipos_de_Órdenes_Click
154
155 End Sub

```

Figura 32. Pedir Año para resumen de Ventas

## Código

```

1 Attribute VB_Name = "Form_Pedir Año para resumen de Ventas"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True

```

```
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8 Dim FITempo As Variant, FFTempo As Variant
9
10 Private Sub Comando4_Click()
11 On Error GoTo Err_Comando4_Click
12
13 Dim stDocName As String
14
15 Select Case MarcoInforme
16 Case 1
17     Select Case TipoInfo
18     Case 1
19         stDocName = "Ventas Anuales G"
20     Case 2
21         If Not IsNull(Departamento) Then
22             stDocName = "Ventas Anuales D"
23         Else
24             MsgBox "ERROR, debe escoger un Departamento"
25             Exit Sub
26         End If
27     Case 3
28         If Not IsNull(Producto) Then
29             stDocName = "Ventas Anuales P"
30         Else
31             MsgBox "ERROR, debe escoger un Producto"
32             Exit Sub
33         End If
34     End Select
35 Case 2
36     Select Case TipoInfo
37     Case 1
38         stDocName = "Ventas Semestrales G"
39     Case 2
40         If Not IsNull(Departamento) Then
41             stDocName = "Ventas Semestrales D"
42         Else
43             MsgBox "ERROR, debe escoger un Departamento"
44             Exit Sub
45         End If
46     Case 3
47         If Not IsNull(Producto) Then
48             stDocName = "Ventas Semestrales P"
49         Else
50             MsgBox "ERROR, debe escoger un Producto"
51             Exit Sub
52         End If
53     End Select
54 Case 3
55     Select Case TipoInfo
56     Case 1
57         stDocName = "Ventas Trimestrales G"
58     Case 2
59         If Not IsNull(Departamento) Then
```

```
60         stDocName = "Ventas Trimestrales D"
61     Else
62         MsgBox "ERROR, debe escoger un Departamento"
63     Exit Sub
64 End If
65 Case 3
66     If Not IsNull(Producto) Then
67         stDocName = "Ventas Trimestrales P"
68     Else
69         MsgBox "ERROR, debe escoger un Producto"
70     Exit Sub
71 End If
72 End Select
73 Case 4
74     Select Case TipoInfo
75     Case 1
76         stDocName = "Ventas Mensuales G"
77     Case 2
78         If Not IsNull(Departamento) Then
79             stDocName = "Ventas Mensuales D"
80         Else
81             MsgBox "ERROR, debe escoger un Departamento"
82         Exit Sub
83     End If
84     Case 3
85         If Not IsNull(Producto) Then
86             stDocName = "Ventas Mensuales P"
87         Else
88             MsgBox "ERROR, debe escoger un Producto"
89         Exit Sub
90     End If
91 End Select
92 End Select
93
94 DoCmd.OpenReport stDocName, acPreview
95
96 Exit_Comando4_Click:
97     Exit Sub
98
99 Err_Comando4_Click:
100     MsgBox Err.Description
101     Resume Exit_Comando4_Click
102
103 End Sub
104 Private Sub Comando6_Click()
105 On Error GoTo Err_Comando6_Click
106
107     Dim stDocName As String
108
109     Select Case MarcoInforme
110     Case 1
111         Select Case TipoInfo
112         Case 1
113             stDocName = "Ventas Anuales G"
114         Case 2
115             If Not IsNull(Departamento) Then
```

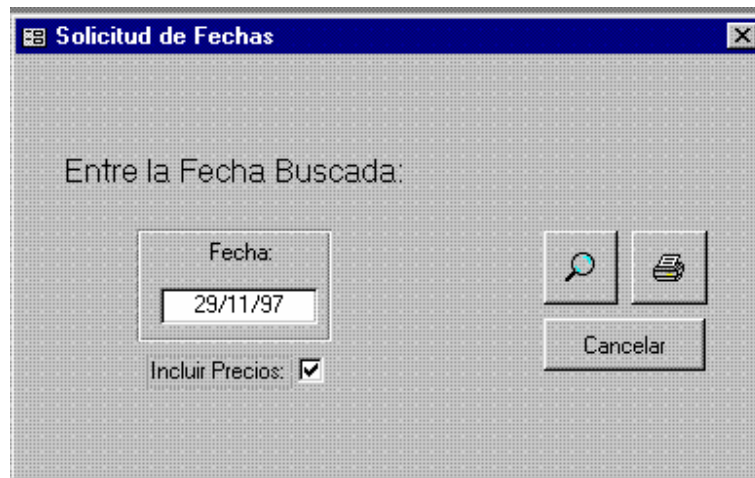
```
116         stDocName = "Ventas Anuales D"
117     Else
118         MsgBox "ERROR, debe escoger un Departamento"
119         Exit Sub
120     End If
121     Case 3
122         If Not IsNull(Producto) Then
123             stDocName = "Ventas Anuales P"
124         Else
125             MsgBox "ERROR, debe escoger un Producto"
126             Exit Sub
127         End If
128     End Select
129     Case 2
130         Select Case TipoInfo
131             Case 1
132                 stDocName = "Ventas Semestrales G"
133             Case 2
134                 If Not IsNull(Departamento) Then
135                     stDocName = "Ventas Semestrales D"
136                 Else
137                     MsgBox "ERROR, debe escoger un Departamento"
138                     Exit Sub
139                 End If
140             Case 3
141                 If Not IsNull(Producto) Then
142                     stDocName = "Ventas Semestrales P"
143                 Else
144                     MsgBox "ERROR, debe escoger un Producto"
145                     Exit Sub
146                 End If
147             End Select
148     Case 3
149         Select Case TipoInfo
150             Case 1
151                 stDocName = "Ventas Trimestrales G"
152             Case 2
153                 If Not IsNull(Departamento) Then
154                     stDocName = "Ventas Trimestrales D"
155                 Else
156                     MsgBox "ERROR, debe escoger un Departamento"
157                     Exit Sub
158                 End If
159             Case 3
160                 If Not IsNull(Producto) Then
161                     stDocName = "Ventas Trimestrales P"
162                 Else
163                     MsgBox "ERROR, debe escoger un Producto"
164                     Exit Sub
165                 End If
166             End Select
167     Case 4
168         Select Case TipoInfo
169             Case 1
170                 stDocName = "Ventas Mensuales G"
171             Case 2
```

```
172         If Not IsNull(Departamento) Then
173             stDocName = "Ventas Mensuales D"
174         Else
175             MsgBox "ERROR, debe escoger un Departamento"
176             Exit Sub
177         End If
178     Case 3
179         If Not IsNull(Producto) Then
180             stDocName = "Ventas Mensuales P"
181         Else
182             MsgBox "ERROR, debe escoger un Producto"
183             Exit Sub
184         End If
185     End Select
186 End Select
187
188 DoCmd.OpenReport stDocName, acNormal
189
190 Exit_Comando6_Click:
191     Exit Sub
192
193 Err_Comando6_Click:
194     MsgBox Err.Description
195     Resume Exit_Comando6_Click
196
197 End Sub
198 Private Sub Comando7_Click()
199 On Error GoTo Err_Comando7_Click
200
201     DoCmd.Close
202
203 Exit_Comando7_Click:
204     Exit Sub
205
206 Err_Comando7_Click:
207     MsgBox Err.Description
208     Resume Exit_Comando7_Click
209
210 End Sub
211
212 Private Sub MarcoInforme_Click()
213     Select Case MarcoInforme
214         Case 1
215             Año.Enabled = False
216         Case Else
217             Año.Enabled = True
218     End Select
219 End Sub
220
221 Private Sub TipoInfo_Click()
222     Select Case TipoInfo
223         Case 1
224             Departamento.Enabled = False
225             Producto.Enabled = False
226         Case 2
227             Producto.Enabled = False
```

```

228     Departamento.Enabled = True
229     Case 3
230     Producto.Enabled = True
231     Departamento.Enabled = False
232 End Select
233 End Sub

```



**Figura 33. Solicitud de Fechas**

### **Código**

```

1  Attribute VB_Name = "Form_Pedir Dos Fechas"
2  Attribute VB_Creatable = True
3  Attribute VB_PredeclaredId = True
4  Attribute VB_Exposed = False
5  Option Compare Database
6  Option Explicit
7
8  Private Sub Comando4_Click()
9  On Error GoTo Err_Comando4_Click
10
11     Dim stDocName As String
12
13     If NomInforme = "Explosión de Materiales" Then
14         If NInforme = 1 Then
15             stDocName = "Planes de Producción Mensuales"
16         Else
17             If NInforme = 2 Then
18                 stDocName = "Planes de Producción Globales 1"
19             Else
20                 stDocName = "Explosión de Materiales"
21             End If

```

```
22     End If
23     Else
24         stDocName = NomInforme
25     End If
26
27     DoCmd.OpenReport stDocName, acPreview
28
29 Exit_Comando4_Click:
30     Exit Sub
31
32 Err_Comando4_Click:
33     MsgBox Err.Description
34     Resume Exit_Comando4_Click
35
36 End Sub
37 Private Sub Comando6_Click()
38 On Error GoTo Err_Comando6_Click
39
40     Dim stDocName As String
41
42     If NomInforme = "Explosión de Materiales" Then
43         If NInforme = 1 Then
44             stDocName = "Planes de Producción Mensuales"
45         Else
46             If NInforme = 2 Then
47                 stDocName = "Planes de Producción Globales 1"
48             Else
49                 stDocName = "Explosión de Materiales"
50             End If
51         End If
52     Else
53         stDocName = NomInforme
54     End If
55
56     DoCmd.OpenReport stDocName, acNormal
57
58 Exit_Comando6_Click:
59     Exit Sub
60
61 Err_Comando6_Click:
62     MsgBox Err.Description
63     Resume Exit_Comando6_Click
64
65 End Sub
66 Private Sub Comando7_Click()
67 On Error GoTo Err_Comando7_Click
68
69
70     DoCmd.Close
71
72 Exit_Comando7_Click:
73     Exit Sub
74
75 Err_Comando7_Click:
76     MsgBox Err.Description
77     Resume Exit_Comando7_Click
```



```
78
79 End Sub
80
81 Private Sub FechaInicial_AfterUpdate()
82     FechaÚnica = FechaInicial - 1
83 End Sub
84
85 Private Sub FechaÚnica_Click()
86     FechaÚnica = Null
87 End Sub
88
89 Private Sub Form_Open(Cancel As Integer)
90
91     If NomInforme = "Explosión de Materiales" Then
92         Título1.Visible = True
93         FechaInicial.Visible = True
94         FechaFinal.Visible = True
95         NInforme.Visible = True
96         Exceso.Visible = True
97         Título2.Visible = False
98         MarcoFecha.Visible = False
99         FechaÚnica.Visible = False
100        IncluirPrecios.Visible = False
101    Else
102        Título1.Visible = False
103        FechaInicial.Visible = False
104        FechaFinal.Visible = False
105        NInforme.Visible = False
106        Exceso.Visible = False
107        Título2.Visible = True
108        FechaÚnica.Visible = True
109        MarcoFecha.Visible = True
110        If (NomInforme = "Existencia Real en Almacén I") Or _
111            (NomInforme = "Existencia Total Real en Almacén I") Or _
112            (NomInforme = "Existencia Total Real en Almacén T") Then
113            IncluirPrecios.Visible = True
114        Else
115            IncluirPrecios.Visible = False
116        End If
117    End If
118
119 End Sub
```

Figura 34. Pedir Fechas para la Demanda

## Código

```

1  Attribute VB_Name = "Form_Pedir Fechas para la Demanda"
2  Attribute VB_Creatable = True
3  Attribute VB_PredeclaredId = True
4  Attribute VB_Exposed = False
5  Option Compare Database
6  Option Explicit
7
8  Dim FITempo As Variant, FFTempo As Variant, strSQL As String
9
10 Private Sub Comando4_Click()
11 On Error GoTo Err_Comando4_Click
12
13     Dim stDocName As String
14
15     If Not EsGeneral Then
16         If IsNull(FechaInicial) Or IsNull(FechaFinal) Then
17             MsgBox "Si el reporte NO es General, debe introducir DOS
18             Exit Sub
19         Else
20             If FechaFinal < FechaInicial Then
21                 MsgBox "La fecha final NO puede ser inferior a la
22                 Exit Sub
23             End If
24         End If
25     End If
26
27     If Todos Then
28         DoCmd.SetWarnings False

```

```
29     strSQL = "INSERT INTO [Productos Demandados] ( Producto )" & _
30         " SELECT [Productos Terminados].[Código del Producto]" &
31         " FROM [Productos Terminados];"
32     DoCmd.RunSQL strSQL
33     DoCmd.SetWarnings True
34 End If
35
36 If MarcoSalida = 1 Then
37     stDocName = "Demanda por Fechas"
38 Else
39     DoCmd.SetWarnings False
40     strSQL = "DELETE * FROM [Demanda por Fechas Tabla];"
41     DoCmd.RunSQL strSQL
42     strSQL = "PARAMETERS Forms![Pedir Fechas para la
Demanda]!FechaInicial DateTime, Forms![Pedir Fechas para la
43         " INSERT INTO [Demanda por Fechas Tabla] ( Producto, [Fecha
Base], [Demanda Puntual], [Demanda en Bs] )" & _
44         " SELECT [Demanda por Fechas 0].Producto, [Demanda por
Fechas 0].[Fecha Base], [Demanda por Fechas 0].[Demanda Puntual],
45         [Demanda por Fechas 0].[Demanda en Bs]" & _
46         " FROM [Demanda por Fechas 0] WHERE ((([Demanda por Fechas
0].[Fecha Base]) Between [Forms]![Pedir Fechas para la
Demanda]![FechaInicial] And" & _
47         " [Forms]![Pedir Fechas para la Demanda]![FechaFinal])) OR
((( [Fecha Base]=[Forms]![Pedir Fechas para la
48         Demanda]![FechaInicial]) Or ([Fecha Base]=[Forms]![Pedir Fechas
para la Demanda]);"
49     DoCmd.RunSQL strSQL
50     DoCmd.SetWarnings True
51     If MarcoInforme = 1 Then
52         stDocName = "Demanda por Fechas GU"
53     Else
54         stDocName = "Demanda por Fechas GB"
55     End If
56 End If
57
58 DoCmd.OpenReport stDocName, acPreview
59
60 Exit_Comando4_Click:
61 Exit Sub
62
63 Err_Comando4_Click:
64 MsgBox Err.Description
65 Resume Exit_Comando4_Click
66
67 End Sub
68 Private Sub Comando6_Click()
69 On Error GoTo Err_Comando6_Click
70
71 Dim stDocName As String
72
73 If Not EsGeneral Then
74     If IsNull(FechaInicial) Or IsNull(FechaFinal) Then
75         MsgBox "Si el reporte NO es General, debe introducir DOS
76         Exit Sub
77     Else
```

```
76         If FechaFinal < FechaInicial Then
77             MsgBox "La fecha final NO puede ser inferior a la
78                 Exit Sub
79             End If
80         End If
81     End If
82
83     If Todos Then
84         DoCmd.SetWarnings False
85         strSQL="INSERT INTO [Productos Demandados] ( Producto )" & _
86             " SELECT [Productos Terminados].[Código del Producto]" & _
87             " FROM [Productos Terminados];"
88         DoCmd.RunSQL strSQL
89         DoCmd.SetWarnings True
90     End If
91
92     If MarcoSalida = 1 Then
93         stDocName = "Demanda por Fechas"
94     Else
95         DoCmd.SetWarnings False
96         strSQL = "DELETE * FROM [Demanda por Fechas Tabla];"
97         DoCmd.RunSQL strSQL
98         strSQL = "PARAMETERS Forms![Pedir Fechas para la
Demanda]!FechaInicial DateTime, Forms![Pedir Fechas para la
99             " INSERT INTO [Demanda por Fechas Tabla] ( Producto, [Fecha
Base], [Demanda Puntual], [Demanda en Bs] )" & _
100             " SELECT [Demanda por Fechas 0].Producto, [Demanda por
Fechas 0].[Fecha Base], [Demanda por Fechas 0].[Demanda Puntual],
101             [Demanda por Fechas 0].[Demanda en Bs]" & _
102             " FROM [Demanda por Fechas 0] WHERE ((([Demanda por Fechas
0].[Fecha Base]) Between [Forms]![Pedir Fechas para la
Demanda]![FechaInicial] And" & _
103             " [Forms]![Pedir Fechas para la Demanda]![FechaFinal])) OR
((( [Fecha Base]=[Forms]![Pedir Fechas para la
Demanda]![FechaInicial]) Or ([Fecha Base]=[Forms]![Pedir Fechas
para la Demanda]);"
104         DoCmd.RunSQL strSQL
105         DoCmd.SetWarnings True
106         If MarcoInforme = 1 Then
107             stDocName = "Demanda por Fechas GU"
108         Else
109             stDocName = "Demanda por Fechas GB"
110         End If
111     End If
112     DoCmd.OpenReport stDocName, acNormal
113
114 Exit_Comando6_Click:
115     Exit Sub
116
117 Err_Comando6_Click:
118     MsgBox Err.Description
119     Resume Exit_Comando6_Click
120
121 End Sub
122
123 Private Sub Comando7_Click()
124 On Error GoTo Err_Comando7_Click
125     DoCmd.Close
126
127 Exit_Comando7_Click:
128     Exit Sub
129
130 Err_Comando7_Click:
```

```
132     MsgBox Err.Description
133     Resume Exit_Comando7_Click
135 End Sub
136
137 Private Sub EsGeneral_AfterUpdate()
139     If EsGeneral Then
140         FechaInicial = Null
141         FechaFinal = Null
142         FechaInicial.Enabled = False
143         FechaFinal.Enabled = False
144     Else
145         If Not IsNull(FITempo) Then
146             FechaInicial = FITempo
147         End If
148         If Not IsNull(FFTempo) Then
149             FechaFinal = FFTempo
150         End If
151         FechaInicial.Enabled = True
152         FechaFinal.Enabled = True
153     End If
154 End Sub
155
156 Private Sub EsGeneral_BeforeUpdate(Cancel As Integer)
157     If Not IsNull(FechaInicial) Then
158         FITempo = FechaInicial
159     End If
160     If Not IsNull(FechaFinal) Then
161         FFTempo = FechaFinal
162     End If
163 End Sub
164 Private Sub Inicializar_Click()
165 On Error GoTo Err_Comando58_Click
166     DoCmd.SetWarnings False
167     strSQL = "DELETE * FROM [Productos Demandados];"
168     DoCmd.RunSQL strSQL
169     DoCmd.SetWarnings True
170     Forms![Pedir Fechas para la Demanda].Refresh
171 Exit_Comando58_Click:
172     Exit Sub
173 Err_Comando58_Click:
174     MsgBox Err.Description
175     Resume Exit_Comando58_Click
176 End Sub
177
178 Private Sub Todos_AfterUpdate()
179     If Todos = True Then
180         [Subformulario Productos Demandados].Visible = False
181         Inicializar.Visible = False
182     Else
183         [Subformulario Productos Demandados].Visible = True
184         Inicializar.Visible = True
185     End If
186 End Sub
```

Figura 35. Pedir Fechas para Ventas

## Código

```

1  Attribute VB_Name = "Form_Pedir Fechas para Ventas"
2  Attribute VB_Creatable = True
3  Attribute VB_PredeclaredId = True
4  Attribute VB_Exposed = False
5  Option Compare Database
6  Option Explicit
7
8  Dim FITempo As Variant, FFTempo As Variant
9
10 Private Sub Comando4_Click()
11 On Error GoTo Err_Comando4_Click
12
13     Dim stDocName As String
14
15     If Not EsGeneral Then
16         If IsNull(FechaInicial) Or IsNull(FechaFinal) Then
17             MsgBox "Si el reporte NO es General, debe introducir DOS"
18             Exit Sub
19         Else
20             If FechaFinal < FechaInicial Then
21                 MsgBox "La fecha final NO puede ser inferior a la"
22                 Exit Sub
23             End If
24         End If
25     End If
26
27     If NInforme = 1 Then
28         stDocName = "Clientes, Pedidos y Facturas G"
29     Else
30         If NInforme = 2 Then
31             stDocName = "Clientes, Pedidos y Facturas S"
32         Else
33             stDocName = "Clientes, Pedidos y Facturas V"

```

```
34     End If
35 End If
36
37 DoCmd.OpenReport stDocName, acPreview
38
39 Exit_Comando4_Click:
40     Exit Sub
41
42 Err_Comando4_Click:
43     MsgBox Err.Description
44     Resume Exit_Comando4_Click
45
46 End Sub
47 Private Sub Comando6_Click()
48 On Error GoTo Err_Comando6_Click
49
50     Dim stDocName As String
51
52     If Not EsGeneral Then
53         If IsNull(FechaInicial) Or IsNull(FechaFinal) Then
54             MsgBox "Si el reporte NO es General, debe introducir DOS
55             Exit Sub
56         Else
57             If FechaFinal < FechaInicial Then
58                 MsgBox "La fecha final NO puede ser inferior a la
59                 Exit Sub
60             End If
61         End If
62     End If
63
64     If NInforme = 1 Then
65         stDocName = "Clientes, Pedidos y Facturas G"
66     Else
67         If NInforme = 2 Then
68             stDocName = "Clientes, Pedidos y Facturas S"
69         Else
70             stDocName = "Clientes, Pedidos y Facturas V"
71         End If
72     End If
73
74     DoCmd.OpenReport stDocName, acNormal
75
76 Exit_Comando6_Click:
77     Exit Sub
78
79 Err_Comando6_Click:
80     MsgBox Err.Description
81     Resume Exit_Comando6_Click
82
83 End Sub
84 Private Sub Comando7_Click()
85 On Error GoTo Err_Comando7_Click
86
87
88     DoCmd.Close
89
```

```
90 Exit_Comando7_Click:
91     Exit Sub
92
93 Err_Comando7_Click:
94     MsgBox Err.Description
95     Resume Exit_Comando7_Click
96
97 End Sub
98
99 Private Sub EsGeneral_AfterUpdate()
100
101     If EsGeneral Then
102         FechaInicial = Null
103         FechaFinal = Null
104         FechaInicial.Enabled = False
105         FechaFinal.Enabled = False
106     Else
107         If Not IsNull(FITempo) Then
108             FechaInicial = FITempo
109         End If
110         If Not IsNull(FFTempo) Then
111             FechaFinal = FFTempo
112         End If
113         FechaInicial.Enabled = True
114         FechaFinal.Enabled = True
115     End If
116 End Sub
117
118 Private Sub EsGeneral_BeforeUpdate(Cancel As Integer)
119     If Not IsNull(FechaInicial) Then
120         FITempo = FechaInicial
121     End If
122     If Not IsNull(FechaFinal) Then
123         FFTempo = FechaFinal
124     End If
125 End Sub
```



**Figura 36. Pedir Fechas para Ventas por Productos**

### Código

```

1  Attribute VB_Name = "Form_Pedir Fechas para Ventas por Productos"
2  Attribute VB_Creatable = True
3  Attribute VB_PredeclaredId = True
4  Attribute VB_Exposed = False
5  Option Compare Database
6  Option Explicit
7
8  Dim FITempo As Variant, FFTempo As Variant
9
10 Private Sub Comando4_Click()
11 On Error GoTo Err_Comando4_Click
12
13     Dim stDocName As String
14
15     If Not EsGeneral Then
16         If IsNull(FechaInicial) Or IsNull(FechaFinal) Then
17             MsgBox "Si el reporte NO es General, debe introducir DOS"
18             Exit Sub
19         Else
20             If FechaFinal < FechaInicial Then
21                 MsgBox "La fecha final NO puede ser inferior a la"
22                 Exit Sub
23             End If
24         End If
25     End If
26
27     Select Case NInforme
28     Case 1
29         stDocName = "Productos Pedidos y Despachados entre Fechas"
30     Case 2
31         stDocName = "Productos Pedidos y Despachados entre Fechas D"
32     Case 3

```

```
33     PreparaGráficoDeProductosEstrella
34     stDocName = "Gráfico de Productos Estrella"
35     Case 4
36     stDocName = "Gráfico de Productos Vendidos Depto"
37 End Select
38
39 DoCmd.OpenReport stDocName, acPreview
40
41 Exit_Comando4_Click:
42     Exit Sub
43
44 Err_Comando4_Click:
45     MsgBox Err.Description
46     Resume Exit_Comando4_Click
47
48 End Sub
49 Private Sub Comando6_Click()
50 On Error GoTo Err_Comando6_Click
51
52     Dim stDocName As String
53
54     If Not EsGeneral Then
55         If IsNull(FechaInicial) Or IsNull(FechaFinal) Then
56             MsgBox "Si el reporte NO es General, debe introducir DOS
57             Exit Sub
58         Else
59             If FechaFinal < FechaInicial Then
60                 MsgBox "La fecha final NO puede ser inferior a la
61                 Exit Sub
62             End If
63         End If
64     End If
65
66     Select Case NInforme
67     Case 1
68         stDocName = "Productos Pedidos y Despachados entre Fechas"
69     Case 2
70         stDocName = "Productos Pedidos y Despachados entre Fechas D"
71     Case 3
72         PreparaGráficoDeProductosEstrella
73         stDocName = "Gráfico de Productos Estrella"
74     Case 4
75         stDocName = "Gráfico de Productos Vendidos Depto"
76     End Select
77
78     DoCmd.OpenReport stDocName, acNormal
79
80 Exit_Comando6_Click:
81     Exit Sub
82
83 Err_Comando6_Click:
84     MsgBox Err.Description
85     Resume Exit_Comando6_Click
86
87 End Sub
88 Private Sub Comando7_Click()
```

```
89 On Error GoTo Err_Comando7_Click
90
91 DoCmd.Close
92
93 Exit_Comando7_Click:
94 Exit Sub
95
96 Err_Comando7_Click:
97 MsgBox Err.Description
98 Resume Exit_Comando7_Click
99
100 End Sub
101
102 Private Sub EsGeneral_AfterUpdate()
103
104 If EsGeneral Then
105 FechaInicial = Null
106 FechaFinal = Null
107 FechaInicial.Enabled = False
108 FechaFinal.Enabled = False
109 Else
110 If Not IsNull(FITempo) Then
111 FechaInicial = FITempo
112 End If
113 If Not IsNull(FFTempo) Then
114 FechaFinal = FFTempo
115 End If
116 FechaInicial.Enabled = True
117 FechaFinal.Enabled = True
118 End If
119 End Sub
120
121 Private Sub EsGeneral_BeforeUpdate(Cancel As Integer)
122 If Not IsNull(FechaInicial) Then
123 FITempo = FechaInicial
124 End If
125 If Not IsNull(FechaFinal) Then
126 FFTempo = FechaFinal
126 End If
127 End Sub
```

**Figura 37. Pedir Número del Plan**

### Código

```

1  Attribute VB_Name = "Form_Pedir Número del Plan"
2  Attribute VB_Creatable = True
3  Attribute VB_PredeclaredId = True
4  Attribute VB_Exposed = False
5  Option Compare Database
6  Option Explicit
7
8  Private Sub Comando4_Click()
9  On Error GoTo Err_Comando4_Click
10
11  Dim stDocName As String
12
13  If NInforme = 1 Then
14  stDocName = "Planes de Producción Mensuales"
15  Else
16  If NInforme = 2 Then
17  stDocName = "Planes de Producción Globales 1"
18  Else
19  stDocName = "Explosión de Materiales"
20  End If
21  End If
22
23  DoCmd.OpenReport stDocName, acPreview
24
25  Exit_Comando4_Click:
26  Exit Sub
27
28  Err_Comando4_Click:
29  MsgBox Err.Description
30  Resume Exit_Comando4_Click
31
32  End Sub
33  Private Sub Comando6_Click()
34  On Error GoTo Err_Comando6_Click

```

```
35
36 Dim stDocName As String
37
38 If NInforme = 1 Then
39     stDocName = "Planes de Producción Mensuales"
40 Else
41     If NInforme = 2 Then
42         stDocName = "Planes de Producción Globales 1"
43     Else
44         stDocName = "Explosión de Materiales"
45     End If
46 End If
47
48 DoCmd.OpenReport stDocName, acNormal
49
50 Exit_Comando6_Click:
51     Exit Sub
52
53 Err_Comando6_Click:
54     MsgBox Err.Description
55     Resume Exit_Comando6_Click
56
57 End Sub
58 Private Sub Comando7_Click()
59 On Error GoTo Err_Comando7_Click
60
61     DoCmd.Close
62
63 Exit_Comando7_Click:
64     Exit Sub
65
66 Err_Comando7_Click:
67     MsgBox Err.Description
68     Resume Exit_Comando7_Click
69
70 End Sub
71
72 Private Sub Plan_AfterUpdate()
73     [FecIni] = DLookup("[Fecha de Inicio]", "Planes de Producción",
74     "[N° del Plan]=" & Str([Plan]))
75 End Sub
```

**Personas**

Cédula de Identidad o Clave: 01      Apellidos: Marquina      Nombres: Eduardo

Dirección de Habitación: Urb. El Encanto. Calle 2-3. Edificio "Santa María", piso 4, apto. 4-3.      Dirección de Trabajo: PROULA. Planta de Medicamentos. Lagunillas.

País: Venezuela      Estado: Mérida      Ciudad: Mérida

Fecha de Nacimiento: 10/05/66      País de Nacimiento: Venezuela      Estado Civil: Casado(a)      Nº de Hijos: 3      Sexo Femenino?

Registro: 1 de 1

**Personas**

Referencias Bancarias

Nombre del Banco	Nº de Cuenta	Dirección y Teléfonos	Montos de Créditos
Mercantil	1065-25849-1	Av. 5. Edificio La Torre de Los Andes.	Bs.3.500.000,00
			Bs.0,00

Registro: 2 de 2

Referencias Comerciales

Nombre del Comercio	Dirección y Teléfonos	Montos de Créditos
Comercial Occidente C.A.	Av. 3. Media cuadra arriba de la Plaza Glorias Patrias	Bs.785.000,00
		Bs.0,00

Registro: 2 de 2

Empleado de PROULA?

Cargo: Analista de Cuentas

Editar Cargos      Editar Estados Civiles      Editar Países, Estados y Ciudades

Registro: 1 de 1

Figura 38. Personas (1° y 2° Páginas)

**Código**

```
1 Attribute VB_Name = "Form_Personas"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8 Private Sub Editar_Cargos_Exit(Cancel As Integer)
9     Dim ctlLista As Control
10
11     Set ctlLista = Forms!Personas!Cargo
12     ctlLista.Requery
13 End Sub
14
15
16 Private Sub Editar_Estados_Civiles_Exit(Cancel As Integer)
17     Dim ctlLista As Control
18
19     Set ctlLista = Forms!Personas![Estado Civil]
20     ctlLista.Requery
21 End Sub
22
23
24 Sub Editar_Países__Estados_y_Ciudades_Click()
25 On Error GoTo Err_Editar_Países__Estados_y_Ciudades_Click
26
27     Dim stDocName As String
28     Dim stLinkCriteria As String
29
30     stDocName = "Países"
31     DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria
32
33 Exit_Editar_Países__Estados_y_Ciudades_C:
34     Exit Sub
35
36 Err_Editar_Países__Estados_y_Ciudades_Click:
37     MsgBox Err.Description
38     Resume Exit_Editar_Países__Estados_y_Ciudades_C
39
40 End Sub
41
42
43 Private Sub Editar_Países__Estados_y_Ciudades_Exit(Cancel As
Integer)
44
45     Dim ctlLista As Control
46
47     Set ctlLista = Forms!Personas!Ciudad
48     ctlLista.Requery
49     Set ctlLista = Forms!Personas!Estado
50     ctlLista.Requery
51     Set ctlLista = Forms!Personas!País
52     ctlLista.Requery
53     Set ctlLista = Forms!Personas![País de Nacimiento]
```

```
54     ctlLista.Requery
55 End Sub
56
57
58 Sub Editar_Estados_Civiles_Click()
59 On Error GoTo Err_Editar_Estados_Civiles_Click
60
61     Dim stDocName As String
62     Dim stLinkCriteria As String
63
64     stDocName = "Estados Civiles"
65     DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria
66
67 Exit_Editar_Estados_Civiles_Click:
68     Exit Sub
69
70 Err_Editar_Estados_Civiles_Click:
71     MsgBox Err.Description
72     Resume Exit_Editar_Estados_Civiles_Click
73
74 End Sub
75 Sub Editar_Cargos_Click()
76 On Error GoTo Err_Editar_Cargos_Click
77
78     Dim stDocName As String
79     Dim stLinkCriteria As String
80
81     stDocName = "Cargos"
82     DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria
83
84 Exit_Editar_Cargos_Click:
85     Exit Sub
86
87 Err_Editar_Cargos_Click:
88     MsgBox Err.Description
89     Resume Exit_Editar_Cargos_Click
90
91 End Sub
```



**Planes de Producción**

Nº del Plan: 25

Descripción: Plan de Producción Generado con Modelos de Producción el 25/10/97. Modelo Nº 2. Modificado para incluir más de un producto...

Fecha de Inicio: 01/01/98      Fecha de Culminación: 01/01/00

Plan de Producción

Fecha	Código del Producto	Cantidad
01/01/98	001	629
01/01/98	002	629
29/08/98	002	264
28/09/98	001	1200
28/10/98	002	220
27/11/98	001	158

Registro: 1 de 6

Registro: 6 de 7

Editar Productos Terminados

**Figura 39. Planes de Producción**

### Código

```

1  Attribute VB_Name = "Form_Planes de Producción"
2  Attribute VB_Creatable = True
3  Attribute VB_PredeclaredId = True
4  Attribute VB_Exposed = False
5  Option Compare Database
6  Option Explicit
7
8  Dim ctlLista As Control
9
10 Private Sub Editar_Productos_Terminados_Click()
11 On Error GoTo Err_Editar_Productos_Terminados_Click
12
13     Dim stDocName As String
14     Dim stLinkCriteria As String
15
16     stDocName = "Productos Terminados"
17     DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria
18
19 Exit_Editar_Productos_Terminados_Click:
20     Exit Sub
21
22 Err_Editar_Productos_Terminados_Click:
23     MsgBox Err.Description
24     Resume Exit_Editar_Productos_Terminados_Click

```

```
25
26 End Sub
27 Private Sub Comando14_Click()
28 On Error GoTo Err_Comando14_Click
29
30 Dim stDocName As String
31
32 stDocName = "Planes de Producción"
33 DoCmd.DoMenuItem acFormBar, acRecordsMenu, 5, , acMenuVer70
34 DoCmd.OpenReport stDocName, acPreview, , "[N° del Plan] = " &
Str(Forms![Planes de Producción]![N° del Plan])
35
36 Exit_Comando14_Click:
37 Exit Sub
38
39 Err_Comando14_Click:
40 MsgBox Err.Description
41 Resume Exit_Comando14_Click
42
43 End Sub
44 Private Sub Comando15_Click()
45 On Error GoTo Err_Comando15_Click
46
47 Dim stDocName As String
48
49 stDocName = "Planes de Producción"
50 DoCmd.DoMenuItem acFormBar, acRecordsMenu, 5, , acMenuVer70
51 DoCmd.OpenReport stDocName, acNormal, , "[N° del Plan] = " &
Str(Forms![Planes de Producción]![N° del Plan])
52
53 Exit_Comando15_Click:
54 Exit Sub
55
56 Err_Comando15_Click:
57 MsgBox Err.Description
58 Resume Exit_Comando15_Click
59
60 End Sub
61
62 Private Sub Editar_Productos_Terminados_Exit(Cancel As Integer)
63 Set ctlLista = Forms![Planes de Producción]![subformulario
Productos en Planes]![Código del Producto]
64 ctlLista.Requery
65 End Sub
```

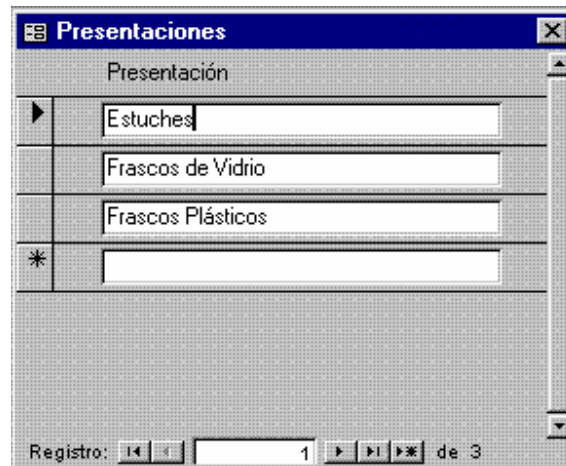


Figura 40. Presentaciones

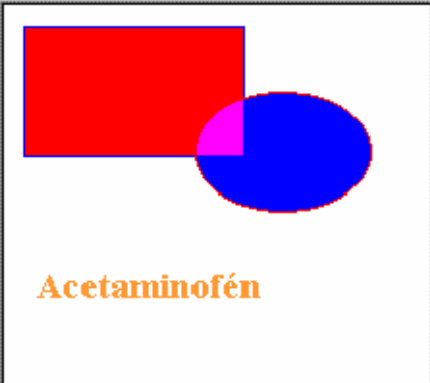


Figura 41. Productos Genéricos

**Productos Terminados**

Código del Producto: 001      Fórmula: 14

Fotografía



**Acetaminofén**

Datos de la Fórmula

Producto: Acetaminofén      Departamento: Sólidos

Vehículo:      Concentración: 500 mg

Tamaño del Lote: 500000      Presentación: Tabletas      Período de Validez: 3 Años

Destino: Venta

Punto de Reorden: 1241

Costo Unitario: Bs.215,00

Presentación y Cantidad: Frascos Plásticos, 20 Tabletas

Registro: 1 de 8

**Productos Terminados**

Escalas de Descuentos por Cantidad

Escala	Descripción	Límite Inferior	Límite Superior	% Descuento
1	Precio Full	0	200	0,00%
2	Precio Económico	201	400	5,00%
3	Precio Mayorista	401	1000	15,00%
4	Precio Hospital	1001	500000	20,00%

Registro: 1 de 4

Precios

Fecha	Precio Unitario de Venta
01/11/97	Bs.235,50
09/10/97	Bs.217,35

Registro: 1 de 2

Registro: 1 de 8

Editar Fórmulas      Editar Presentaciones      Editar Destinos

Editar Escalas de Precios

Figura 42. Productos Terminados (1° y 2° Páginas)

## Código

- 1 Attribute VB\_Name = "Form\_Productos Terminados"
- 2 Attribute VB\_Creatable = True
- 3 Attribute VB\_PredeclaredId = True

```
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8 Dim ctlLista As Control
9
10 Private Sub Editar_Destinos_Exit(Cancel As Integer)
11     Set ctlLista = Forms![Productos Terminados]!Destino
12     ctlLista.Requery
13 End Sub
14
15
16 Private Sub Editar_Escalas_de_Precios_Exit(Cancel As Integer)
17     Set ctlLista = Forms![Productos Terminados]![Productos Terminados
18 y Escalas de Precios subformulario]!Escala
19     ctlLista.Requery
20 End Sub
21
22 Sub Editar_Fórmulas_Click()
23 On Error GoTo Err_Editar_Fórmulas_Click
24
25     Dim stDocName As String
26     Dim stLinkCriteria As String
27
28     stDocName = "Fórmulas"
29     DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria
30 Exit_Editar_Fórmulas_Click:
31     Exit Sub
32
33 Err_Editar_Fórmulas_Click:
34     MsgBox Err.Description
35     Resume Exit_Editar_Fórmulas_Click
36
37 End Sub
38
39 Private Sub Editar_Fórmulas_Exit(Cancel As Integer)
40     Set ctlLista = Forms![Productos Terminados]![N° de Fórmula]
41     ctlLista.Requery
42 End Sub
43
44
45 Sub Editar_Presentaciones_Click()
46 On Error GoTo Err_Editar_Presentaciones_Click
47
48     Dim stDocName As String
49     Dim stLinkCriteria As String
50
51     stDocName = "Presentaciones"
52     DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria
53
54 Exit_Editar_Presentaciones_Click:
55     Exit Sub
56
57 Err_Editar_Presentaciones_Click:
58     MsgBox Err.Description
```

```
59     Resume Exit_Editar_Presentaciones_Click
60
61 End Sub
62 Sub Editar_Destinos_Click()
63 On Error GoTo Err_Editar_Destinos_Click
64
65     Dim stDocName As String
66     Dim stLinkCriteria As String
67
68     stDocName = "Destinos"
69     DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria
70
71 Exit_Editar_Destinos_Click:
72     Exit Sub
73
74 Err_Editar_Destinos_Click:
75     MsgBox Err.Description
76     Resume Exit_Editar_Destinos_Click
77
78 End Sub
79
80 Private Sub Editar_Presentaciones_Exit(Cancel As Integer)
81     Set ctlLista = Forms![Productos Terminados]![Presentación]
82     ctlLista.Requery
83 End Sub
84
85
86 Sub Editar_Escalas_de_Precios_Click()
87 On Error GoTo Err_Editar_Escalas_de_Precios_Click
88
89     Dim stDocName As String
90     Dim stLinkCriteria As String
91
92     stDocName = "Escalas de Precios"
93     DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria
94
95 Exit_Editar_Escalas_de_Precios_Click:
96     Exit Sub
97
98 Err_Editar_Escalas_de_Precios_Click:
99     MsgBox Err.Description
100     Resume Exit_Editar_Escalas_de_Precios_Click
101
102 End Sub
103 Sub Imprimir_Producto_Click()
104 On Error GoTo Err_Imprimir_Producto_Click
105
106     Dim stDocName As String
107
108     stDocName = "Productos Terminados"
109     DoCmd.DoMenuItem acFormBar, acRecordsMenu, 5, , acMenuVer70
110     DoCmd.OpenReport stDocName, acNormal, , "[Código del Producto] =
    '" & Forms![Productos Terminados]![Código del Producto] & "'"
111
112 Exit_Imprimir_Producto_Click:
113     Exit Sub
```

```
114
115 Err_Imprimir_Producto_Click:
116     MsgBox Err.Description
117     Resume Exit_Imprimir_Producto_Click
118
119 End Sub
120 Sub Presentar_Producto_Click()
121 On Error GoTo Err_Presentar_Producto_Click
122
123     Dim stDocName As String
124
125     stDocName = "Productos Terminados"
126     DoCmd.DoMenuItem acFormBar, acRecordsMenu, 5, , acMenuVer70
127     DoCmd.OpenReport stDocName, acPreview, , "[Código del Producto] =
128     '" & Forms![Productos Terminados]![Código del Producto] & "'"
129 Exit_Presentar_Producto_Click:
130     Exit Sub
131
132 Err_Presentar_Producto_Click:
133     MsgBox Err.Description
134     Resume Exit_Presentar_Producto_Click
135
136 End Sub
```

**Proveedores**

Código del Proveedor: 001      Cédula de Identidad o RIF: 150-4589-6

Nombre o Razón Social: Importadora de Insumos Biológicos OMEGA C.A.

Dirección: Zona Industrial "El Puente Sobre El Lago", Carrera 4. Local 23.

País: Venezuela

Estado: Zulia

Pueblo o Ciudad: Maracaibo

Tenencia del Local: Propio

Figura Jurídica: C.A.

Tipo de Proveedor: Clase A

Fecha de Revisión: 10/10/97      Responsable: 01

Marquina, Eduardo

Sobre el Registro de Comercio

Nº: 75-6849-256      Fecha: 06/04/75

Duración: 50 Años

Observaciones: Deben colocársele los pedidos, al menos 15 días antes de necesitarlos

Registro: 1 de 9

Figura 43. Proveedores (1° Página)



**Proveedores**

**Teléfonos**

Teléfono	Otro Código o Celular	¿Fax?
965879		<input type="checkbox"/>
965478		<input checked="" type="checkbox"/>
▶ 9758496	014	<input type="checkbox"/>
* [ ]		<input type="checkbox"/>

Registro: 3 de 3

**Contactos**

Contacto: 020  
Dávila J., Mariano

Registro: 1 de 1

**Sucursales**

Sucursal: [ ]

Registro: 1 de 1

Registro: 1 de 9

Editar Tipos de Proveedores    Editar Personas    Editar Países, Estados y Ciudades  
 Editar Personas Jurídicas    Editar Tenencias de Locales

Figura 44. Proveedores (2° Página)

**Código**

```

1  Attribute VB_Name = "Form_Proveedores"
2  Attribute VB_Creatable = True
3  Attribute VB_PredeclaredId = True
4  Attribute VB_Exposed = False
5  Option Compare Database
6  Option Explicit
7
8  Dim ctlLista As Control
9
10 Private Sub Código_del_Proveedor_Exit(Cancel As Integer)
11     Forms![Proveedores]![Sucursales de Proveedores
12     subformulario]!Sucursal.Requery
13 End Sub
14
15 Sub Editar_Países__Estados_y_Ciudades_Click()
16 On Error GoTo Err_Editar_Países__Estados_y_Ciudades_Click
17
18     Dim stDocName As String
19     Dim stLinkCriteria As String
20
21     stDocName = "Países"
22     DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria
23

```

```
24 Exit_Editar_Países__Estados_y_Ciudades_C:
25     Exit Sub
26
27 Err_Editar_Países__Estados_y_Ciudades_Click:
28     MsgBox Err.Description
29     Resume Exit_Editar_Países__Estados_y_Ciudades_C
30
31 End Sub
32
33
34 Private Sub Editar_Países__Estados_y_Ciudades_Exit(Cancel As
Integer)
35     Set ctlLista = Forms!Proveedores!Ciudad
36     ctlLista.Requery
37     Set ctlLista = Forms!Proveedores!Estado
38     ctlLista.Requery
39     Set ctlLista = Forms!Proveedores!País
40     ctlLista.Requery
41 End Sub
42
43
44 Sub Editar_Personas_Click()
45 On Error GoTo Err_Editar_Personas_Click
46
47     Dim stDocName As String
48     Dim stLinkCriteria As String
49
50     stDocName = "Personas"
51     DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria
52
53 Exit_Editar_Personas_Click:
54     Exit Sub
55
56 Err_Editar_Personas_Click:
57     MsgBox Err.Description
58     Resume Exit_Editar_Personas_Click
59
60 End Sub
61
62
63 Private Sub Editar_Personas_Exit(Cancel As Integer)
64     Set ctlLista = Forms!Proveedores!Responsable
65     ctlLista.Requery
66     Set ctlLista = Forms!Proveedores![Contactos en
Proveedores]!Contacto
67     ctlLista.Requery
68 End Sub
69
70 Sub Editar_Personas_Jurídicas_Click()
71 On Error GoTo Err_Editar_Personas_Jurídicas_Click
72
73     Dim stDocName As String
74     Dim stLinkCriteria As String
75
76     stDocName = "Figuras Jurídicas"
77     DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria
```

```
78
79 Exit_Editar_Personas_Jurídicas_Click:
80     Exit Sub
81
82 Err_Editar_Personas_Jurídicas_Click:
83     MsgBox Err.Description
84     Resume Exit_Editar_Personas_Jurídicas_Click
85
86 End Sub
87
88
89 Private Sub Editar_Personas_Jurídicas_Exit(Cancel As Integer)
90     Set ctlLista = Forms!Proveedores![Figura Jurídica]
91     ctlLista.Requery
92 End Sub
93
94 Sub Editar_Tenencias_de_Locales_Click()
95 On Error GoTo Err_Editar_Tenencias_de_Locales_Click
96
97     Dim stDocName As String
98     Dim stLinkCriteria As String
99
100     stDocName = "Tenencia de Locales"
101     DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria
102
103 Exit_Editar_Tenencias_de_Locales_Click:
104     Exit Sub
105
106 Err_Editar_Tenencias_de_Locales_Click:
107     MsgBox Err.Description
108     Resume Exit_Editar_Tenencias_de_Locales_Click
109
110 End Sub
111
112 Private Sub Editar_Tenencias_de_Locales_Exit(Cancel As Integer)
113     Set ctlLista = Forms!Proveedores![Tenencia del Local]
114     ctlLista.Requery
115 End Sub
116
117 Sub Editar_Tipos_de_Clientes_Click()
118 On Error GoTo Err_Editar_Tipos_de_Clientes_Click
119
120     Dim stDocName As String
121     Dim stLinkCriteria As String
122
123     stDocName = "Tipos de Proveedores"
124     DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria
125
126 Exit_Editar_Tipos_de_Clientes_Click:
127     Exit Sub
128
129 Err_Editar_Tipos_de_Clientes_Click:
130     MsgBox Err.Description
131     Resume Exit_Editar_Tipos_de_Clientes_Click
132
133 End Sub
```

```
134
135 Private Sub Editar_Tipos_de_Clientes_Exit(Cancel As Integer)
136     Set ctlLista = Forms!Proveedores![Tipo de Proveedor]
137     ctlLista.Requery
138 End Sub
149
150 Private Sub Sucursales_de_Proveedores_Enter()
151     Set ctlLista = Forms!Proveedores![Sucursales de
152     Proveedores]!Sucursal
153 End Sub
```

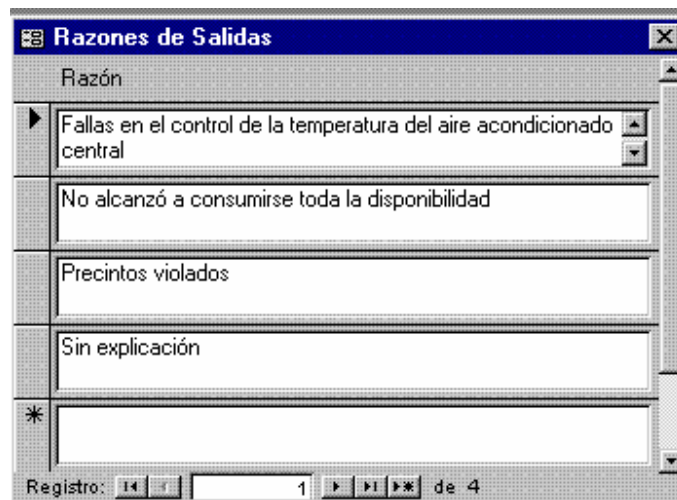


Figura 45. Razones de Salidas



Figura 46. Sectorés Económicos

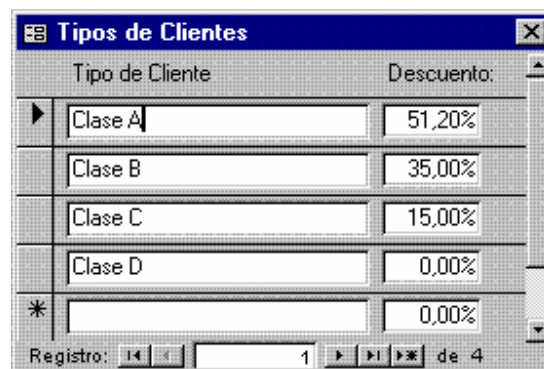


The screenshot shows a window titled "Tenencia de Locales". It contains a table with the following data:

Tenencia	
	Alquilado
▶	Propio
*	

At the bottom, there is a record navigation bar: "Registro: 2 de 2".

Figura 47. Tenencia de Locales

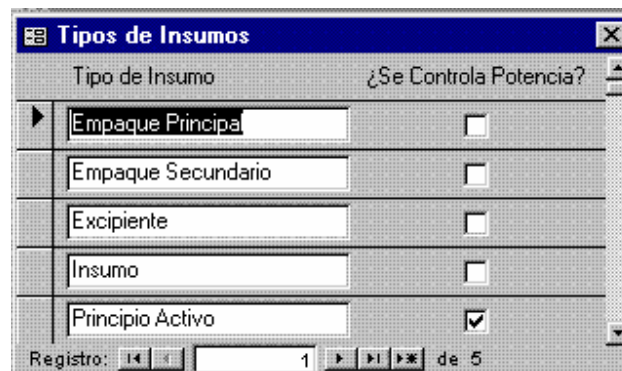


The screenshot shows a window titled "Tipos de Clientes". It contains a table with the following data:

Tipo de Cliente	Descuento:
▶ Clase A	51,20%
Clase B	35,00%
Clase C	15,00%
Clase D	0,00%
*	0,00%

At the bottom, there is a record navigation bar: "Registro: 1 de 4".

Figura 48. Tipos de Clientes



The screenshot shows a window titled "Tipos de Insumos". It contains a table with the following data:

Tipo de Insumo	¿Se Controla Potencia?
▶ Empaque Principa	<input type="checkbox"/>
Empaque Secundario	<input type="checkbox"/>
Excipiente	<input type="checkbox"/>
Insumo	<input type="checkbox"/>
Principio Activo	<input checked="" type="checkbox"/>

At the bottom, there is a record navigation bar: "Registro: 1 de 5".

Figura 49. Tipos de Insumos



Figura 50. Tipos de Órdenes de Pedido

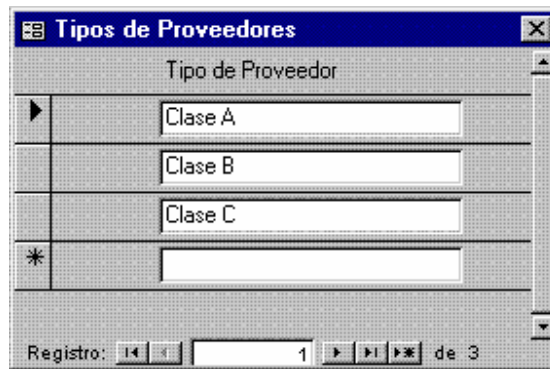


Figura 51. Tipos de Proveedores



Figura 52. Tipos de Salidas



**Figura 53. Unidades de Medición**

### Código

```

1  Attribute VB_Name = "Form_Unidades de Medición"
2  Attribute VB_Creatable = True
3  Attribute VB_PredeclaredId = True
4  Attribute VB_Exposed = False
5  Option Compare Database
6  Option Explicit
7
8  Dim ctLista As Control
9  Sub Editar_Magnitudes_Físicas_Click()
10 On Error GoTo Err_Editar_Magnitudes_Físicas_Click
11
12     Dim stDocName As String
13     Dim stLinkCriteria As String
14
15     stDocName = "Magnitudes Físicas"
16     DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria
17
18 Exit_Editar_Magnitudes_Físicas_Click:
19     Exit Sub
20
21 Err_Editar_Magnitudes_Físicas_Click:
22     MsgBox Err.Description
23     Resume Exit_Editar_Magnitudes_Físicas_Click
24
25 End Sub
26
27 Private Sub Editar_Magnitudes_Físicas_Exit(Cancel As Integer)
28     Set ctLista = Forms![Unidades de Medición]![Magnitud Física]
29     ctLista.Requery
30 End Sub

```

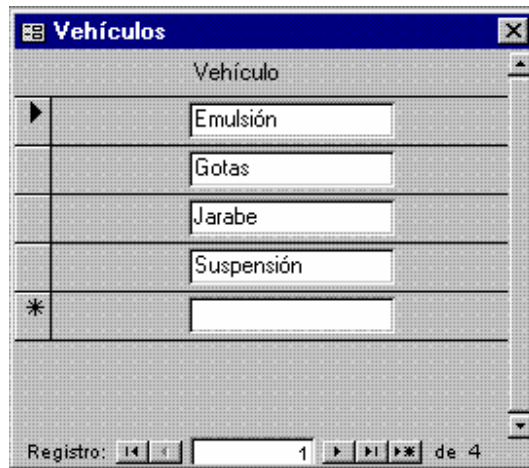


Figura 54. Vehículos



## **ANEXO 2**

### **Informes del SARCAI**

### **y Detalles de Programación**

## Índice de Informes

Informe	Pág.
Accionistas de Clientes subinforme	4
Ciudades de Estados subinforme	4
Clientes	4
Clientes subinforme de enlace	4
Clientes, Pedidos y Facturas G	4
Clientes, Pedidos y Facturas S	6
Clientes, Pedidos y Facturas V	8
Contactos en Clientes subinforme	9
Contactos en Proveedores subinforme	9
Conversión de Insumos subinforme	10
Cotizaciones	10
Demanda por Fechas	11
Demanda por Fechas GB	12
Demanda por Fechas GU	13
Entradas de Insumos 3ros	14
Estado de Pedidos	14
Existencia Real en Almacén I	15
Existencia Real en Almacén T	16
Existencia Total Real en Almacén I	17
Existencia Total Real en Almacén T	18
Explosión de Materiales	20
Facturas	21
Fórmulas	22
Fórmulas subinforme	22
Fórmulas y Productos Terminados subinforme	22
Gráfico de Productos Estrella	22
Gráfico de Productos Vendidos Depto	23
Gráfico Temporal de Productos Vendidos	24
Insumos	25
Insumos en Órdenes de Producción subinforme	25
Insumos en Órdenes subinforme	26
Lista de Precios Vigente en Fecha Buscada	26
Lista Sencilla de Productos Terminados subinforme	27
Llegadas de Insumos al Almacén	27
Modelos de Programación	28
Modelos EOQ	37
Notas en Facturas subinforme	46
Órdenes de Compra	46
Órdenes de Producción	47
Países subinforme	48
Pedidos de Clientes	48
Pedidos de Clientes subinforme	48

Informe	Pág.
Personas subinforme	48
Planes de Producción	80
Planes de Producción Globales 1	49
Planes de Producción Mensuales	49
Precios de Productos subinforme	54
Productos Facturados a Clientes subinforme	54
Productos Pedidos de Clientes subinforme	55
Productos Pedidos y Despachados entre Fechas	56
Productos Pedidos y Despachados entre Fechas D	57
Productos Terminados	59
Informe: Productos Terminados subinforme	59
Productos Terminados y Escalas de Precios subinforme	60
Proveedores	60
Proveedores subinforme de enlace	60
Referencias Bancarias de Clientes subinforme	60
Referencias Comerciales de Clientes subinforme	60
Resumen de Órdenes de Producción	60
Revisiones de Fórmulas subinforme	61
Subinforme Lista Sencilla de Productos Terminados	61
Subinforme Notas en Cotizaciones	61
Subinforme Productos Cotizados a Clientes	62
Subinforme Productos en Modelos P	62
Subinforme Tipos de Clientes	62
Subtotal Kg y Lt Insumos Ord de Prod subinforme	63
Sucursales de Clientes subinforme	63
Sucursales de Proveedores subinforme	63
Teléfonos de Clientes subinforme	63
Teléfonos de Proveedores subinforme	63
Vendedores y sus Clientes	63
Ventas Anuales D	64
Ventas Anuales G	65
Ventas Anuales P	66
Ventas Mensuales D	68
Ventas Mensuales G	69
Ventas Mensuales P	70
Ventas Semestrales D	72
Ventas Semestrales G	73
Ventas Semestrales P	74
Ventas Trimestrales D	76
Ventas Trimestrales G	77
Ventas Trimestrales P	78

Este anexo está dedicado mostrar los detalles de programación de los informes que presenta el sistema. Un informe es el resultado de relacionar tablas para componer información útil en la toma de decisiones. Como informes se incluyen listados preparados a partir de los datos y también gráficos que resumen datos. En el Anexo 3 se muestran ejemplos, impresos en papel, de los informes que pueden ser obtenidos. Los datos que éstos contienen son ficticios y se emplean como ejemplos solamente.

Un informe contiene diversos elementos. Los datos son presentados en forma muy similar a como fueron introducidos (es decir, empleando en su mayoría cuadros de texto). Existen al menos tres formas de extraer estos datos de la base de datos, por ejemplo, tal como se encuentran en cada tabla, utilizando una instrucción SQL y/o utilizando una función Visual Basic. Sólo en los dos últimos casos se documentará la operación, puesto que el primer caso puede entenderse a partir de la revisión del diseño de la base de datos. Por otra parte, los informes pueden contener subinformes (informes dentro de informes), que se explican aparte ya que un mismo subinforme puede eventualmente utilizarse en más de un informe. En conclusión, de cada informe se detallan únicamente aquellos controles de datos importantes.

## 1. Accionistas de Clientes subinforme

### Origen del registro:

```
SELECT DISTINCTROW [Accionistas de Clientes Consulta].Cliente,  
    [Accionistas de Clientes Consulta].Accionista, [Accionistas de  
    Clientes Consulta].[Apellidos], [Accionistas de Clientes  
    Consulta].[Nombres]  
FROM [Accionistas de Clientes Consulta];
```

## 2. Ciudades de Estados subinforme

### Origen del registro:

```
SELECT DISTINCTROW [Ciudades de Estados].[País], [Ciudades de  
    Estados].[Estado], [Ciudades de Estados].[Ciudad], [Ciudades de  
    Estados].[Código Telefónico]  
FROM [Ciudades de Estados];
```

## 3. Clientes

Origen del registro: Clientes Consulta.

## 4. Clientes subinforme de enlace

### Origen del registro:

```
SELECT DISTINCTROW [Clientes].[Código del Cliente], [Clientes].[Nombre o  
    Razón Social]  
FROM [Clientes];
```

## 5. Clientes, Pedidos y Facturas G

Origen del registro: Clientes, Pedidos y Facturas G.

Campo: Desde.

=[Forms]![Pedir Fechas para Ventas]![FechaInicial]

Marco de objeto independiente: Gráfico1

Clase OLE: Gráfico de Microsoft Graph 97

Origen de la fila:

```
SELECT [Grupo],Sum([Total Pedido]) AS [SumaDeTotal Pedido]
FROM [Clientes, Pedidos y Facturas G]
GROUP BY [Grupo];
```

### Marco de objeto independiente: Gráfico2

#### Clase OLE: Gráfico de Microsoft Graph 97

#### Origen de la fila:

```
SELECT [Clientes, Pedidos y Facturas G].Grupo, Sum([Clientes, Pedidos y
Facturas G].[Total Surtido]) AS [SumaDeTotal Surtido]
FROM [Clientes, Pedidos y Facturas G]
GROUP BY [Clientes, Pedidos y Facturas G].Grupo;
```

#### Campo: Hasta

```
=[Forms]![Pedir Fechas para Ventas]![FechaFinal]
```

#### Campo: Texto19

```
=[Total Pedido]/[Total General Pedido]
```

#### Campo: Texto22

```
=[Total Surtido]/[Total General Surtido]
```

#### Campo: Texto25

```
=1-[Total Surtido]/[Total Pedido]
```

#### Campo: Texto27

```
=1-[Total General Surtido]/[Total General Pedido]
```

#### Campo: Total General Pedido

```
=Sum([Total Pedido])
```

#### Campo: Total General Surtido

```
=Sum([Total Surtido])
```

#### Código

```
1 Attribute VB_Name = "Report_Clientes, Pedidos y Facturas G"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8 Private Sub Report_Open(Cancel As Integer)
9     If Forms![Pedir Fechas para Ventas]!EsGeneral Then
10         General.Visible = True
```

```

11     Desde.Visible = False
12     Hasta.Visible = False
13     CuadroDesdeHasta.Visible = False
14     Else
15     General.Visible = False
16     Desde.Visible = True
17     Hasta.Visible = True
18     CuadroDesdeHasta.Visible = True
19     End If
20
21     Gráfico1.Visible = False
22     Gráfico2.Visible = False
23     If Forms![Pedir Fechas para Ventas]!ConGráficos Then
24     Gráfico1.Visible = True
25     Gráfico2.Visible = True
26     End If
27     End Sub

```

## 6. Clientes, Pedidos y Facturas S

Origen del registro: Clientes, Pedidos y Facturas S.

Campo: Desde

```
=[Forms]![Pedir Fechas para Ventas]![FechaInicial]
```

Marco de objeto independiente: Gráfico1

Clase OLE: Gráfico de Microsoft Graph 97

Origen de la fila:

```

SELECT [Clientes, Pedidos y Facturas S].Sector, Sum([Clientes, Pedidos y
Facturas S].[Total Pedido]) AS [SumaDeTotal Pedido]
FROM [Clientes, Pedidos y Facturas S]
GROUP BY [Clientes, Pedidos y Facturas S].Sector;

```

Marco de objeto independiente: Gráfico2

Clase OLE: Gráfico de Microsoft Graph 97

Origen de la fila:

```

SELECT [Clientes, Pedidos y Facturas S].Sector, Sum([Clientes, Pedidos y
Facturas S].[Total Surtido]) AS [SumaDeTotal Surtido]
FROM [Clientes, Pedidos y Facturas S]
GROUP BY [Clientes, Pedidos y Facturas S].Sector;

```

Campo: Hasta

```
=[Forms]![Pedir Fechas para Ventas]![FechaFinal]
```

Campo: Texto19

=[Total Pedido]/[Total General Pedido]

Campo: Texto22

=[Total Surtido]/[Total General Surtido]

Campo: Texto25

=1-[Total Surtido]/[Total Pedido]

Campo: Texto27

=1-[Total General Surtido]/[Total General Pedido]

Campo: Total General Pedido

=Sum([Total Pedido])

Campo: Total General Surtido

=Sum([Total Surtido])

Código

```
1 Attribute VB_Name = "Report_Clientes, Pedidos y Facturas S"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8 Private Sub Report_Open(Cancel As Integer)
9     If Forms![Pedir Fechas para Ventas]!EsGeneral Then
10         General.Visible = True
11         Desde.Visible = False
12         Hasta.Visible = False
13         CuadroDesdeHasta.Visible = False
14     Else
15         General.Visible = False
16         Desde.Visible = True
17         Hasta.Visible = True
18         CuadroDesdeHasta.Visible = True
19     End If
20     Gráfico1.Visible = False
21     Gráfico2.Visible = False
22     If Forms![Pedir Fechas para Ventas]!ConGráficos Then
23         Gráfico1.Visible = True
24         Gráfico2.Visible = True
25     End If
26 End Sub
```

## 7. Clientes, Pedidos y Facturas V

Origen del registro: Clientes, Pedidos y Facturas V.

Campo: Desde

=[Forms]![Pedir Fechas para Ventas]![FechaInicial]

Marco de objeto independiente: Gráfico1

Clase OLE: Gráfico de Microsoft Graph 97

Origen de la fila:

```
SELECT [Clientes, Pedidos y Facturas V].Agente, Sum([Clientes, Pedidos y
Facturas V].[Total Pedido]) AS [SumaDeTotal Pedido]
FROM [Clientes, Pedidos y Facturas V]
GROUP BY [Clientes, Pedidos y Facturas V].Agente;
```

Marco de objeto independiente: Gráfico2

Clase OLE: Gráfico de Microsoft Graph 97

Origen de la fila:

```
SELECT [Clientes, Pedidos y Facturas V].Agente, Sum([Clientes, Pedidos y
Facturas V].[Total Surtido]) AS [SumaDeTotal Surtido]
FROM [Clientes, Pedidos y Facturas V]
GROUP BY [Clientes, Pedidos y Facturas V].Agente;
```

Campo: Hasta

=[Forms]![Pedir Fechas para Ventas]![FechaFinal]

Campo: Texto19

=[Total Pedido]/[Total General Pedido]

Campo: Texto22

=[Total Surtido]/[Total General Surtido]

Campo: Texto25

=1-[Total Surtido]/[Total Pedido]

Campo: Texto27

=1-[Total General Surtido]/[Total General Pedido]

Campo: Total General Pedido



=Sum([Total Pedido])

### Campo: Total General Surtido

=Sum([Total Surtido])

### Código

```

1  Attribute VB_Name = "Report_Clientes, Pedidos y Facturas V"
2  Attribute VB_Creatable = True
3  Attribute VB_PredeclaredId = True
4  Attribute VB_Exposed = False
5  Option Compare Database
6  Option Explicit
7
8  Private Sub Report_Open(Cancel As Integer)
9      If Forms![Pedir Fechas para Ventas]!EsGeneral Then
10         General.Visible = True
11         Desde.Visible = False
12         Hasta.Visible = False
13         CuadroDesdeHasta.Visible = False
14     Else
15         General.Visible = False
16         Desde.Visible = True
17         Hasta.Visible = True
18         CuadroDesdeHasta.Visible = True
19     End If
20     Gráfico1.Visible = False
21     Gráfico2.Visible = False
22     If Forms![Pedir Fechas para Ventas]!ConGráficos Then
23         Gráfico1.Visible = True
24         Gráfico2.Visible = True
25     End If
26 End Sub

```

## **8. Contactos en Clientes subinforme**

### Origen del registro:

```

SELECT DISTINCTROW [Contactos en Clientes Consulta].[Cliente], [Contactos
en Clientes Consulta].[Cédula de Identidad o Clave], [Contactos en
Clientes Consulta].[Apellidos], [Contactos en Clientes
Consulta].[Nombres]
FROM [Contactos en Clientes Consulta];

```

## **9. Contactos en Proveedores subinforme**

### Origen del registro:

```

SELECT DISTINCTROW [Contactos en Proveedores Consulta].[Código del
Proveedor], [Contactos en Proveedores Consulta].[Cédula de Identidad
o Clave], [Contactos en Proveedores Consulta].[Apellidos],[Contactos
en Proveedores Consulta].[Nombres]
FROM [Contactos en Proveedores Consulta];

```

## 10. Conversión de Insumos subinforme

### Origen del registro:

```
SELECT DISTINCTROW [Conversión de Insumos].[N° de Fórmula], [Conversión
de Insumos].[Código de Insumo], [Conversión de Insumos].[Nombre],
[Conversión de Insumos].[Cantidad], [Conversión de Insumos].[A]
FROM [Conversión de Insumos];
```

### Nivel de grupo 0: A.

### Campo: Subtotal

```
=Sum([Cantidad])
```

### Campo: Texto29

```
=[Cantidad]*1000/[Reports]![Fórmulas]![Tamaño del Lote]
```

### Campo: Texto30

```
=IIf([A]="Kg.", "g.", IIf([A]="Lts.", "ml.", ""))
```

### Campo: Texto31

```
=[Subtotal]*1000/[Reports]![Fórmulas]![Tamaño del Lote]
```

### Campo: Texto32

```
=IIf([A]="Kg.", "g.", IIf([A]="Lts.", "ml.", ""))
```

## 11. Cotizaciones

### Origen del registro: Cotizaciones

### Filtro:

```
([N° de la Cotización] = Forms!Cotizaciones![N° de la Cotización])
```

### Código

```
1 Attribute VB_Name = "Report_Cotizaciones"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8 Private Sub EncabezadoDePágina_Format(Cancel As Integer, FormatCount
As
9 [Forms]![Pedir Dos Fechas]![FechaÚnica] = [Fecha de Emisión]
10 End Sub
11
```

```

12 Private Sub Report_Close()
13   DoCmd.Close acForm, "Pedir Dos Fechas"
14 End Sub
15
16 Private Sub Report_Open(Cancel As Integer)
17   DoCmd.OpenForm "Pedir Dos Fechas", , , , , acHidden
18 End Sub

```

## 12. Demanda por Fechas

Origen del registro: Demanda por Fechas

Nivel de grupo 0: Nombre

Nivel de grupo 1: Fecha Base

Campo: Texto16

=Sum([Demanda Puntual])

Campo: Texto18

=Sum([Demanda en Bs])

Campo: Texto21

=Sum([Demanda Puntual])

Campo: Texto23

=Sum([Demanda en Bs])

Campo: Texto25

=[Forms]![Pedir Fechas para la Demanda]![FechaInicial]

Campo: Texto27

=[Forms]![Pedir Fechas para la Demanda]![FechaFinal]

Código

```

1 Attribute VB_Name = "Report_Demanda por Fechas"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8 Private Sub Report_Open(Cancel As Integer)
9   If Forms![Pedir Fechas para la Demanda]!EsGeneral Then
10    General.Visible = True

```

```

11     Desde.Visible = False
12     Hasta.Visible = False
13     CuadroDesdeHasta.Visible = False
14     Else
15     General.Visible = False
16     Desde.Visible = True
17     Hasta.Visible = True
18     CuadroDesdeHasta.Visible = True
19     End If
20     End Sub

```

### 13. Demanda por Fechas GB

Marco de objeto independiente: OLEIndependiente0

Clase OLE: Gráfico de Microsoft Graph 97

Origen de la fila:

```

TRANSFORM Sum([Demanda en Bs]) AS [SumaDeDemanda en Bs]
SELECT (Format([Fecha Base],"DDDDD"))
FROM [Demanda por Fechas Tabla]
GROUP BY (Int([Fecha Base]),(Format([Fecha Base],"DDDDD")))
PIVOT [Producto];

```

Campo: Texto25

```
=[Forms]![Pedir Fechas para la Demanda]![FechaInicial]
```

Campo: Texto27

```
=[Forms]![Pedir Fechas para la Demanda]![FechaFinal]
```

Código

```

1  Attribute VB_Name = "Report_Demanda por Fechas GB"
2  Attribute VB_Creatable = True
3  Attribute VB_PredeclaredId = True
4  Attribute VB_Exposed = False
5  Option Compare Database
6  Option Explicit
7
8  Private Sub Report_Open(Cancel As Integer)
9      If Forms![Pedir Fechas para la Demanda]!EsGeneral Then
10         General.Visible = True
11         Desde.Visible = False
12         Hasta.Visible = False
13         CuadroDesdeHasta.Visible = False
14     Else
15         General.Visible = False

```

```

16     Desde.Visible = True
17     Hasta.Visible = True
18     CuadroDesdeHasta.Visible = True
19     End If
20     End Sub

```

## 14. Demanda por Fechas GU

Marco de objeto independiente: OLEIndependiente0

Clase OLE: Gráfico de Microsoft Graph 97

Origen de la fila:

```

TRANSFORM Sum([Demanda por Fechas Tabla].[Demanda Puntual]) AS
    [SumaDeDemanda Puntual]
SELECT (Format([Fecha Base],"ddddd")) AS Expr1
FROM [Demanda por Fechas Tabla]
GROUP BY (Int([Fecha Base]), (Format([Fecha Base],"ddddd")))
PIVOT [Demanda por Fechas Tabla].Producto;

```

Campo: Texto25

```
=[Forms]![Pedir Fechas para la Demanda]![FechaInicial]
```

Campo: Texto27

```
=[Forms]![Pedir Fechas para la Demanda]![FechaFinal]
```

Código

```

1  Attribute VB_Name = "Report_Demanda por Fechas GU"
2  Attribute VB_Creatable = True
3  Attribute VB_PredeclaredId = True
4  Attribute VB_Exposed = False
5  Option Compare Database
6  Option Explicit
7
8  Private Sub Report_Open(Cancel As Integer)
9      If Forms![Pedir Fechas para la Demanda]!EsGeneral Then
10         General.Visible = True
11         Desde.Visible = False
12         Hasta.Visible = False
13         CuadroDesdeHasta.Visible = False
14     Else
15         General.Visible = False
16         Desde.Visible = True
17         Hasta.Visible = True
18         CuadroDesdeHasta.Visible = True
19     End If
20 End Sub

```

## 15. Entradas de Insumos 3ros

Origen del registro:

```
SELECT DISTINCTROW [Entradas de Insumos 3ros].[Lote N°], [Entradas de
Insumos 3ros].[Fecha], [Entradas de Insumos 3ros].[Cliente],
[Entradas de Insumos 3ros].[Insumo], [Insumos].[Código de Insumo],
[Insumos].[Nombre], [Insumos].[Tipo de Insumo], [Insumos].[Unidades
de Presentación], [Insumos].[Condiciones de Entrega], [Entradas de
Insumos 3ros].[Cantidad], [Entradas de Insumos 3ros].[Estado],
[Entradas de Insumos 3ros].[Observaciones], [Entradas de Insumos
3ros].[Recibido Por]
FROM ([Insumos] INNER JOIN [Entradas de Insumos 3ros] ON
[Insumos].[Código de Insumo] =[Entradas de Insumos 3ros].[Insumo]);
```

Nivel de grupo 0: Cliente

Nivel de grupo 1: Tipo de Insumo

Nivel de grupo 2: Código de Insumo

Campo: Texto40

=Sum([Cantidad])

Campo: Texto43

=Sum([Cantidad])

Campo: Texto45

=Sum([Cantidad])

Campo: Texto47

=Sum([Cantidad])

## 16. Estado de Pedidos

Origen del registro: Estado de Pedidos

Nivel de grupo 0: N° de Pedido

Nivel de grupo 1: Nombre o Razón Social

Nivel de grupo 2: Nombre

Nivel de grupo 3: N° de Factura

Campo: Texto44

= [Cantidad Facturada] \* [Precio con Descuento]

Campo: Texto50

= Sum([Cantidad Facturada])

Campo: Texto52

= Sum([Cantidad Facturada] \* [Precio con Descuento])

Campo: Texto55

= [Cantidad Pedida] - Sum([Cantidad Facturada])

Campo: Texto57

= [Precio con Descuento] \* ([Cantidad Pedida] - Sum([Cantidad Facturada]))

Campo: Texto65

= [Monto Global del Pedido] - Sum([Cantidad Facturada] \* [Precio con Descuento])

## 17. Existencia Real en Almacén I

Origen del registro: Existencia Real en Almacén I 2

Nivel de grupo 0: Tipo de Insumo

Nivel de grupo 1: Nombre

Nivel de grupo 2: N° Lote

Campo: Precio1

= [Precio Unitario] \* [Existencia Real]

Campo: Precio2

= Avg([Precio Unitario])

Campo: Precio3

= Max([Precio Unitario])

Campo: Precio4

= Avg([Precio Unitario]) \* Sum([Existencia Real])

Campo: Precio5

=Max([Precio Unitario])\*Sum([Existencia Real])

**Campo: Texto33**

=Sum([Cantidad Recibida])

**Campo: Texto35**

=Sum([Cantidad Despachada Total])

**Campo: Texto37**

=Sum([Existencia Real])

**Campo: Texto42**

=[Forms]![Pedir Dos Fechas]![FechaÚnica]

**Código**

```

1  Attribute VB_Name = "Report_Existencia Real en Almacén I"
2  Attribute VB_Creatable = True
3  Attribute VB_PredeclaredId = True
4  Attribute VB_Exposed = False
5  Option Compare Database
6  Option Explicit
7
8  Private Sub Report_Open(Cancel As Integer)
9      If Forms![Pedir Dos Fechas]!IncluirPrecios Then
10         Precio0.Visible = True
11         Precio1.Visible = True
12         Precio2.Visible = True
13         Precio3.Visible = True
14         Precio4.Visible = True
15         Precio5.Visible = True
16         PrecioE1.Visible = True
17         PrecioE2.Visible = True
18         PrecioE3.Visible = True
19         PrecioE4.Visible = True
20     Else
21         Precio0.Visible = False
22         Precio1.Visible = False
23         Precio2.Visible = False
24         Precio3.Visible = False
25         Precio4.Visible = False
26         Precio5.Visible = False
27         PrecioE1.Visible = False
28         PrecioE2.Visible = False
29         PrecioE3.Visible = False
30         PrecioE4.Visible = False
31     End If
32 End Sub

```

**18. Existencia Real en Almacén T**

Origen del registro: Existencia Real en Almacén T



Nivel de grupo 0: Departamento

Nivel de grupo 1: Nombre

Campo: Existencia del Producto

=Sum([Existencia Real])

Campo: Fecha del Filtro

=[Forms]![Pedir Dos Fechas]![FechaÚnica]

### **19. Existencia Total Real en Almacén I**

Origen del registro: Existencia Total Real en Almacén I

Nivel de grupo 0: Tipo de Insumo

Nivel de grupo 1: Nombre

Campo: Precio1

=[Existencia Total Real]\*[Precio Unitario]

Campo: Precio2

=[Existencia Total Proyectada]\*[Precio Unitario]

Campo: Precio3

=Sum([Existencia Total Real]\*[Precio Unitario])

Campo: Precio4

=Sum([Existencia Total Proyectada]\*[Precio Unitario])

Campo: Precio5

=Sum([Existencia Total Real]\*[Precio Unitario])

Campo: Precio6

=Sum([Existencia Total Proyectada]\*[Precio Unitario])

Campo: Texto32

=[Forms]![Pedir Dos Fechas]![FechaÚnica]

Código

1 Attribute VB\_Name = "Report\_Existencia Total Real en Almacén I"

```

2  Attribute VB_Creatable = True
3  Attribute VB_PredeclaredId = True
4  Attribute VB_Exposed = False
5  Option Compare Database
6  Option Explicit
7
8  Private Sub Report_Open(Cancel As Integer)
9      If Forms![Pedir Dos Fechas]!IncluirPrecios Then
10         Precio0.Visible = True
11         Precio1.Visible = True
12         Precio2.Visible = True
13         Precio3.Visible = True
14         Precio4.Visible = True
15         Precio5.Visible = True
16         Precio6.Visible = True
17         PrecioE0.Visible = True
18         PrecioE1.Visible = True
19         PrecioE2.Visible = True
20         PrecioE3.Visible = True
21         PrecioE4.Visible = True
22     Else
23         Precio0.Visible = False
24         Precio1.Visible = False
25         Precio2.Visible = False
26         Precio3.Visible = False
27         Precio4.Visible = False
28         Precio5.Visible = False
29         Precio6.Visible = False
30         PrecioE0.Visible = False
31         PrecioE1.Visible = False
32         PrecioE2.Visible = False
33         PrecioE3.Visible = False
34         PrecioE4.Visible = False
35     End If
36 End Sub
37

```

## 20. Existencia Total Real en Almacén T

### Origen del registro:

```

SELECT DISTINCTROW [Existencia Total Real en Almacén T].[Código del
Producto], [Lista Sencilla de Productos Terminados].[Punto de
Reorden], [Existencia Total Real en Almacén T].[Total Producido],
[Existencia Total Real en Almacén T].[Total Pedido], [Existencia
Total Real en Almacén T].[Total Facturado], [Existencia Total Real
en Almacén T].[Total Despachado], [Existencia Total Real en Almacén
T].[Existencia Real Total], [Existencia Total Real en Almacén
T].[Existencia Proyectada Total], [Lista Sencilla de Productos
Terminados].Nombre, [Lista Sencilla de Productos
Terminados].Departamento, [Existencia Total Real en Almacén
T].[Precio Unitario de Venta]
FROM [Lista Sencilla de Productos Terminados] INNER JOIN [Existencia
Total Real en Almacén T] ON [Lista Sencilla de Productos
Terminados].[Código del Producto] = [Existencia Total Real en

```

Almacén T].[Código del Producto];

Nivel de grupo 0: Departamento

Nivel de grupo 1: Nombre

Campo: Precio1

= [Existencia Real Total] \* [Precio Unitario de Venta]

Campo: Precio2

= [Existencia Proyectada Total] \* [Precio Unitario de Venta]

Campo: Precio3

= Sum([Existencia Real Total] \* [Precio Unitario de Venta])

Campo: Precio4

= Sum([Existencia Proyectada Total] \* [Precio Unitario de Venta])

Campo: Precio5

= Sum([Existencia Real Total] \* [Precio Unitario de Venta])

Campo: Precio6

= Sum([Existencia Proyectada Total] \* [Precio Unitario de Venta])

Campo: Texto32

= [Forms]![Pedir Dos Fechas]![FechaÚnica]

Campo: Texto51

= IIf([Existencia Real Total] <= [Punto de Reorden], "\*", "")

Código

```

1  Attribute VB_Name = "Report_Existencia Total Real en Almacén T"
2  Attribute VB_Creatable = True
3  Attribute VB_PredeclaredId = True
4  Attribute VB_Exposed = False
5  Option Compare Database
6  Option Explicit
7
8  Private Sub Report_Open(Cancel As Integer)
9      If Forms![Pedir Dos Fechas]!IncluirPrecios Then
10         Precio0.Visible = True
11         Precio1.Visible = True
12         Precio2.Visible = True
13         Precio3.Visible = True
14         Precio4.Visible = True
15         Precio5.Visible = True

```

```
16     Precio6.Visible = True
17     PrecioE0.Visible = True
18     PrecioE1.Visible = True
19     PrecioE2.Visible = True
20     PrecioE3.Visible = True
21     PrecioE4.Visible = True
22     Else
23     Precio0.Visible = False
24     Precio1.Visible = False
25     Precio2.Visible = False
26     Precio3.Visible = False
27     Precio4.Visible = False
28     Precio5.Visible = False
29     Precio6.Visible = False
30     PrecioE0.Visible = False
31     PrecioE1.Visible = False
32     PrecioE2.Visible = False
33     PrecioE3.Visible = False
34     PrecioE4.Visible = False
35     End If
36     End Sub
```

## 21. Explosión de Materiales

Origen del registro: Explosión de Materiales 2

Nivel de grupo 0: Tipo de Insumo

Nivel de grupo 1: Nombre

Campo: Otra Fecha

=[forms]![Pedir Dos Fechas]![FechaÚnica]-1

Campo: Texto41

=Sum([Total Requerido en Bolívares])

Campo: Texto43

=Sum([Existencias a la Fecha en Bolívares])

Campo: Texto45

=Sum([Necesidades Adicionales en Bolívares])

Campo: Texto54

=Sum([Total Requerido en Bolívares])

Campo: Texto56

=Sum([Existencias a la Fecha en Bolívares])

Campo: Texto57

```
=Sum([Necesidades Adicionales en Bolívares])
```

Campo: Texto69

```
=[Forms]![Pedir Número del Plan]![Exceso]
```

Código

```

1  Attribute VB_Name = "Report_Explosión de Materiales"
2  Attribute VB_Creatable = True
3  Attribute VB_PredeclaredId = True
4  Attribute VB_Exposed = False
5  Option Compare Database
6  Option Explicit
7
8  Private Sub Report_Close()
9      DoCmd.Close acForm, "Pedir Dos Fechas"
10     End Sub
11
12     Private Sub Report_Open(Cancel As Integer)
13         DoCmd.OpenForm "Pedir Dos Fechas", acFormDS, , , , acHidden
14         Forms![Pedir Dos Fechas]!FechaÚnica.Visible = True
15         Forms![Pedir Dos Fechas]!FechaÚnica = Forms![Pedir Número del
16         Plan]!FecIni - 1
17     End Sub

```

**22. Facturas**Origen del registro: Facturas y Pedidos consultas

Filtro: ([N° de Factura] = Forms!Facturas![N° de Factura])

Código

```

1  Attribute VB_Name = "Report_Facturas"
2  Attribute VB_Creatable = True
3  Attribute VB_PredeclaredId = True
4  Attribute VB_Exposed = False
5  Option Compare Database
6  Option Explicit
7
8  Private Sub EncabezadoDePágina_Format(Cancel As Integer, FormatCount
9  As Integer)
10     Forms![Pedir Dos Fechas]!FechaÚnica = [Fecha]
11     End Sub
12
13     Private Sub Report_Close()
14         DoCmd.Close acForm, "Pedir Dos Fechas"
15     End Sub

```

```

16
17 Private Sub Report_Open(Cancel As Integer)
18     DoCmd.OpenForm "Pedir Dos Fechas", , , , , acHidden
19 End Sub

```

### 23. Fórmulas

Origen del registro: Fórmulas Consulta

### 24. Fórmulas subinforme

Origen del registro:

```

SELECT DISTINCTROW [Fórmulas].[N° de Fórmula], [Fórmulas].[Producto],
    [Fórmulas].[Vehículo], [Fórmulas].[Unidades de Presentación],
    [Fórmulas].[Concentración], [Fórmulas].[Unidades de Concentración],
    [Fórmulas].[Departamento], [Fórmulas].[Tamaño del Lote]
FROM [Fórmulas];

```

### 25. Fórmulas y Productos Terminados subinforme

Origen del registro:

```

SELECT DISTINCTROW [Fórmulas y Productos Terminados].[N° de Fórmula],
    [Fórmulas y Productos Terminados].[Presentación], [Fórmulas y
    Productos Terminados].[Cantidad en Presentación], [Fórmulas y
    Productos Terminados].[Destino]
FROM [Fórmulas y Productos Terminados];

```

### 26. Gráfico de Productos Estrella

Marco de objeto independiente: OLEIndependiente0

Clase OLE: Gráfico de Microsoft Graph 97

Origen de la fila: Gráfico de Productos Estrella 2

Campo: Texto25

```
=[Forms]![Pedir Fechas para Ventas por Productos]![FechaInicial]
```

Campo: Texto27

```
=[Forms]![Pedir Fechas para Ventas por Productos]![FechaFinal]
```

Campo: Texto4

```
=[Forms]![Pedir Fechas para Ventas por Productos]![NProd]
```

Código

```

1  Attribute VB_Name = "Report_Gráfico de Productos Estrella"
2  Attribute VB_Creatable = True
3  Attribute VB_PredeclaredId = True
4  Attribute VB_Exposed = False
5  Option Compare Database
6  Option Explicit
7
8  Private Sub Report_Open(Cancel As Integer)
9
10     If Forms![Pedir Fechas para Ventas por Productos]!EsGeneral Then
11         General.Visible = True
12         Desde.Visible = False
13         Hasta.Visible = False
14         CuadroDesdeHasta.Visible = False
15     Else
16         General.Visible = False
17         Desde.Visible = True
18         Hasta.Visible = True
19         CuadroDesdeHasta.Visible = True
20     End If
21
22 End Sub

```

**27. Gráfico de Productos Vendidos Depto**

Marco de objeto independiente: OLEIndependiente0

Clase OLE: Gráfico de Microsoft Graph 97

Origen de la fila:

```

SELECT [Productos Pedidos y Despachados entre Fechas].Departamento,
       Sum([Productos Pedidos y Despachados entre Fechas].[Total Ventas])
AS [Total Ventas]
FROM [Productos Pedidos y Despachados entre Fechas]
GROUP BY [Productos Pedidos y Despachados entre Fechas].Departamento;

```

Campo: Texto25

```
=[Forms]![Pedir Fechas para Ventas por Productos]![FechaInicial]
```

Campo: Texto27

```
=[Forms]![Pedir Fechas para Ventas por Productos]![FechaFinal]
```

Código

```

1  Attribute VB_Name = "Report_Gráfico de Productos Vendidos Depto"
2  Attribute VB_Creatable = True
3  Attribute VB_PredeclaredId = True
4  Attribute VB_Exposed = False

```

```

5   Option Compare Database
6   Option Explicit
7
8   Private Sub Report_Open(Cancel As Integer)
9
10  If Forms![Pedir Fechas para Ventas por Productos]!EsGeneral Then
11    General.Visible = True
12    Desde.Visible = False
13    Hasta.Visible = False
14    CuadroDesdeHasta.Visible = False
15  Else
16    General.Visible = False
17    Desde.Visible = True
18    Hasta.Visible = True
19    CuadroDesdeHasta.Visible = True
20  End If
21
22  End Sub
23

```

## 28. Gráfico Temporal de Productos Vendidos

Marco de objeto independiente: OLEIndependiente0

Clase OLE: Gráfico de Microsoft Graph 97

Origen de la fila:

```

SELECT [Productos Pedidos y Despachados entre Fechas].Departamento,
       Sum([Productos Pedidos y Despachados entre Fechas].[Total Ventas])
AS [Total Ventas]
FROM [Productos Pedidos y Despachados entre Fechas]
GROUP BY [Productos Pedidos y Despachados entre Fechas].Departamento;

```

Marco de objeto independiente: OLEIndependiente7

Clase OLE: Gráfico de Microsoft Graph 97

Origen de la fila:

```

SELECT [Productos Pedidos y Despachados entre Fechas].Departamento,
       Sum([Productos Pedidos y Despachados entre Fechas].[Total
       Despachado]) AS [Total Despachado]
FROM [Productos Pedidos y Despachados entre Fechas]
GROUP BY [Productos Pedidos y Despachados entre Fechas].Departamento;

```

Campo: Texto25

```

=[Forms]![Pedir Fechas para Ventas por Productos]![FechaInicial]

```



**Campo: Texto27**

```
=[Forms]![Pedir Fechas para Ventas por Productos]![FechaFinal]
```

**Código**

```

1  Attribute VB_Name = "Report_Gráfico Temporal de Productos Vendidos"
2  Attribute VB_Creatable = True
3  Attribute VB_PredeclaredId = True
4  Attribute VB_Exposed = False
5  Option Compare Database
6  Option Explicit
7
8  Private Sub Report_Open(Cancel As Integer)
9
10     If Forms![Pedir Fechas para Ventas por Productos]!EsGeneral Then
11         General.Visible = True
12         Desde.Visible = False
13         Hasta.Visible = False
14         CuadroDesdeHasta.Visible = False
15     Else
16         General.Visible = False
17         Desde.Visible = True
18         Hasta.Visible = True
19         CuadroDesdeHasta.Visible = True
20     End If
21
22 End Sub
23

```

**29. Insumos****Origen del registro: Insumos****Nivel de grupo 0: Tipo de Insumo****Campo: Texto29**

```
=Count([Código de Insumo])
```

**Campo: Texto33**

```
=Count([Código de Insumo])
```

**30. Insumos en Órdenes de Producción subinforme****Origen del registro:**

```

SELECT DISTINCTROW [Insumos en Órdenes de Producción].[N° de Orden],
    [Insumos en Órdenes de Producción].[Código de Insumo], [Insumos en
    Órdenes de Producción].[Lote N°], [Insumos en Órdenes de

```

```

Producción].[Cantidad a Utilizar], Insumos.Nombre, Insumos.[Tipo de
Insumo], Insumos.[Unidades de Presentación], [Insumos en Órdenes de
Producción].[Fecha del Surtido]
FROM Insumos INNER JOIN ([Llegadas de Insumos al Almacén] INNER JOIN
[Insumos en Órdenes de Producción] ON [Llegadas de Insumos al
Almacén].[Lote N°] = [Insumos en Órdenes de Producción].[Lote N°])
ON Insumos.[Código de Insumo] = [Llegadas de Insumos al
Almacén].Insumo;

```

Nivel de grupo 0: Tipo de Insumo

Nivel de grupo 1: Nombre

Campo: Texto23

```
=Sum([Cantidad a Utilizar])
```

### **31. Insumos en Órdenes subinforme**

Origen del registro:

```

SELECT DISTINCTROW [Insumos en Órdenes].[Orden N°], [Insumos en
Órdenes].[Insumo], [Insumos].[Nombre], [Insumos].[Presentación
Mínima], [Insumos].[Unidades de Presentación],
[Insumos].[Condiciones de Entrega], [Insumos en Órdenes].[Precio
Unitario], [Insumos en Órdenes].[Cantidad]
FROM ([Insumos] INNER JOIN [Insumos en Órdenes] ON [Insumos].[Código de
Insumo] =[Insumos en Órdenes].[Insumo]);

```

Nivel de grupo 0: Nombre

Campo: Texto21

```
=[Precio Unitario]*[Cantidad]
```

### **32. Lista de Precios Vigente en Fecha Buscada**

Origen del registro: Lista de Precios Vigente en Fecha Buscada

Nivel de grupo 0: Departamento

Nivel de grupo 1: Nombre

Nivel de grupo 2: Límite Inferior

Campo: Texto34

=[Forms]![Pedir Dos Fechas]![FechaÚnica]

### 33. Lista Sencilla de Productos Terminados subinforme

#### Origen del registro:

```
SELECT DISTINCTROW [Lista Sencilla de Productos Terminados].[Código del
    Producto], [Lista Sencilla de Productos Terminados].[Nombre]
FROM [Lista Sencilla de Productos Terminados];
```

### 34. Llegadas de Insumos al Almacén

#### Origen del registro:

```
SELECT DISTINCTROW [Llegadas de Insumos al Almacén].[Lote N°], [Llegadas
    de Insumos al Almacén].Fecha, [Llegadas de Insumos al
    Almacén].[Orden N°], [Llegadas de Insumos al Almacén].Insumo,
    Insumos.[Código de Insumo], Insumos.Nombre, Insumos.[Tipo de
    Insumo], [Tipos de Insumos].[¿Se Controla Potencia?],
    Insumos.[Unidades de Presentación], Insumos.[Condiciones de
    Entrega], [Llegadas de Insumos al Almacén].Cantidad, [Llegadas de
    Insumos al Almacén].Estado, [Llegadas de Insumos al
    Almacén].Potencia, [Llegadas de Insumos al Almacén].Observaciones,
    [Llegadas de Insumos al Almacén].[Recibido Por]
FROM [Tipos de Insumos] INNER JOIN (Insumos INNER JOIN [Llegadas de
    Insumos al Almacén] ON Insumos.[Código de Insumo] = [Llegadas de
    Insumos al Almacén].Insumo) ON [Tipos de Insumos].[Tipo de Insumo] =
    Insumos.[Tipo de Insumo];
```

#### Nivel de grupo 0: Tipo de Insumo

#### Nivel de grupo 1: Código de Insumo

#### Campo: Subtotal Insumos

=Sum([Cantidad])

#### Campo: Subtotal Tipos

=Sum([Cantidad])

#### Campo: Texto44

=Sum([Cantidad])

#### Código

```
1 Attribute VB_Name = "Report_Llegadas de Insumos al Almacén"
2 Attribute VB_Creatable = True
```

```

3   Attribute VB_PredeclaredId = True
4   Attribute VB_Exposed = False
5   Option Compare Database
6   Option Explicit
7
8   Private Sub EncabezadoDelGrupo0_Format(Cancel As Integer,
    FormatCount As Integer)
9       If [¿Se Controla Potencia?] Then
10          [Potencia].Visible = True
11          [PotenciaE].Visible = True
12      Else
13          [Potencia].Visible = False
14          [PotenciaE].Visible = False
15      End If
16  End Sub

```

### 35. Modelos de Programación

Origen del registro: Modelos de Programación

Campo: TipoDemanda

```

=IIf([forms]![Modelos de Programación]![Valores de la
    Demanda]=1,"Utilizando la Demanda Promedio",IIf([forms]![Modelos de
    Programación]![Valores de la Demanda]=2,"Utilizando la Demanda
    Máxima",IIf([forms]![Modelos de Programación]![Valores de la
    Demanda]=3,"Utilizando la Demanda Mínima","Utilizando la Demanda
    Promedio + " & Str([forms]![Modelos de Programación]![Veces la DS])
    & " veces la DS"))

```

Código

```

1   Attribute VB_Name = "Report_Modelos de Programación"
2   Attribute VB_Creatable = True
3   Attribute VB_PredeclaredId = True
4   Attribute VB_Exposed = False
5   Option Compare Database
6   Option Explicit
7
8   Dim ctlLista As Control
9
10  Private Declare Function FindWindow Lib "user32" Alias _
    "FindWindowA" (ByVal lpClassName As String, _
11      ByVal lpWindowName As Long) As Long
12
13
14  Private Declare Function SendMessage Lib "user32" Alias _
    "SendMessageA" (ByVal hWnd As Long, ByVal wParam As Long, _
15      ByVal lParam As Long) As Long
16
17
18
19  Private Sub ¿Utilizando_Historia__AfterUpdate()
20      If [¿Utilizando Historia?] Then
21          [Historia Desde].Enabled = True
22          [Historia Hasta].Enabled = True

```

```

23     [Valores de la Demanda].Enabled = True
24     [Veces la DS].Enabled = True
25     [Etiqueta67].Visible = True
26     [% de Exceso o Defecto].Enabled = True
27     [% Anual de Inflación].Enabled = True
28     [Demanda].Enabled = True
29     Else
30     [Historia Desde].Enabled = False
31     [Historia Hasta].Enabled = False
32     [Valores de la Demanda].Enabled = False
33     [Veces la DS].Enabled = False
34     [Etiqueta67].Visible = False
35     [% de Exceso o Defecto].Enabled = False
36     [% Anual de Inflación].Enabled = False
37     [Demanda].Enabled = False
38     End If
39 End Sub
40
41 Private Sub Código_del_Producto_AfterUpdate()
42     [Costo U Estimado] = DLookup("[Costo Unitario]", "Productos
Terminados", "[Código del Producto]='" & [Código del Producto] &
"'")
43 End Sub
44
45 Private Sub Comando89_Click()
46     CostoGP = DSum("[A Producir]*[Costo de Producción]", "Productos en
Modelos P", "[N° de Modelo]='" & Str([N° de Modelo]))
47     CostoGI = DSum("[A Mantener]*[Costo de Mantener]", "Productos en
Modelos P", "[N° de Modelo]='" & Str([N° de Modelo]))
48     CostoGE = DSum("[A Demorar]*[Costo de Escasez ]", "Productos en
Modelos P", "[N° de Modelo]='" & Str([N° de Modelo]))
49     [Costo del Plan] = CostoGP + CostoGI + CostoGE
50     If [Costo del Plan] <> 0 Then
51         PorcGP = CostoGP / [Costo del Plan]
52         PorcGI = CostoGI / [Costo del Plan]
53         PorcGE = CostoGE / [Costo del Plan]
54     Else
55         PorcGP = 0
56         PorcGI = 0
57         PorcGE = 0
58     End If
59 End Sub
60
61 Private Sub Comando90_Click()
62     Dim dbs As Database, tbl As Recordset, pln As Recordset, F As Date,
63     F1 As Date, F2 As Date, NPlan As Long, NDías As Integer, _
64     NIter As Integer, i As Integer, plnF As Recordset
65
66     Set dbs = CurrentDb
67     Set tbl = dbs.OpenRecordset("Planes de Producción", dbOpenTable)
68     With tbl
69         .AddNew
70         !Descripción = "Plan de Producción Generado con Modelos de
Producción el " & _
71         Str(Date) & ". Modelo N° " & Str([N° de Modelo]) & ". " &

```

```
[Descripción]
72  ![Fecha de Inicio] = [Desde]
73  Select Case [Horizonte]
74    Case "Semestres"
75      ![Fecha de Culminación] = [Desde] + 120
76    Case "Años"
77      ![Fecha de Culminación] = [Desde] + 365
78    Case "Bienios"
79      ![Fecha de Culminación] = [Desde] + 730
80  End Select
81  .Update
82  .MoveLast
83  NPlan = ![N° del Plan]
84  F1 = ![Fecha de Inicio]
85  F2 = ![Fecha de Culminación]
86  End With
87  tbl.Close
88
89  Set tbl = dbs.OpenRecordset("Productos en Planes", dbOpenTable)
90  Set pln = dbs.OpenRecordset("Productos en Modelos P",
dbOpenDynaset)
91  pln.Filter = "[N° de Modelo] =" & Str([N° de Modelo])
92  Set plnF = pln.OpenRecordset
93
94  Select Case [Lapso de la Demanda]
95    Case "Meses"
96      NDías = 30
97    Case "Bimestres"
98      NDías = 60
99    Case "Trimestres"
100     NDías = 90
101  End Select
102
103  F = F1
104  NIter = DCount("[N° de Modelo]", "Productos en Modelos P", "[N° de
Modelo]=" & Str([N° de Modelo]))
105  plnF.MoveFirst
106  For i = 1 To NIter
107    If plnF![A Producir] <> 0 Then
108      tbl.AddNew
109      tbl![N° del Plan] = NPlan
110      tbl![Código del Producto] = [Código del Producto]
111      tbl![Fecha] = F
112      tbl![Cantidad] = plnF![A Producir]
113      tbl.Update
114    End If
115    plnF.MoveNext
116    F = F + NDías
117  Next i
118  tbl.Close
119  pln.Close
120  plnF.Close
121  End Sub
122
123  Private Sub Demanda_Click()
124
```

```

125 Dim strSQL As String, dbs As Database, rst As Recordset, _
126 i As Integer, Primero As Integer, Último As Integer, _
127 Cad1 As String, Cad2 As String, CostoP As Single, _
128 CostoM As Single, CostoE As Single
129
130 If [Historia Desde] = Null Or [Historia Hasta] = Null Then
131 MsgBox "Entre primero las fechas históricas"
132 Else
133 If [Lapso de la Demanda] = Null Then
134 MsgBox "Entre primero el Lapso de la Demanda"
135 Else
136 DoCmd.SetWarnings False
137 strSQL = "DELETE * FROM [Demanda por Fechas Tabla];"
138 DoCmd.RunSQL strSQL
139 strSQL = "DELETE * FROM [Productos en Modelos P] WHERE [N° de
Modelo]=[Forms]![Modelos de Programación]![N° de Modelo];"
140 DoCmd.RunSQL strSQL
141 strSQL = "PARAMETERS Forms![Modelos de
Programación]![Historia Desde] DateTime, Forms![Modelos de
Programación]![Historia Hasta] DateTime;" & _
142 " INSERT INTO [Demanda por Fechas Tabla] ( Producto, [Fecha
Base], [Demanda Puntual], [Demanda en Bs])" & _
143 " SELECT [Demanda por Fechas 0].Producto, [Demanda por Fechas
0].[Fecha Base], [Demanda por Fechas 0].[Demanda Puntual],[Demanda
por Fechas 0].[Demanda en Bs]" & _
144 " FROM [Demanda por Fechas 0] WHERE ((([Demanda por Fechas
0].[Fecha Base]) Between [Forms]![Modelos de Programación]![Historia
Desde] And" & _
145 " [Forms]![Modelos de Programación]![Historia
Hasta]) And ([Producto]=[Forms]![Modelos de Programación]![Código
del Producto]);"
146 DoCmd.RunSQL strSQL
147 Forms![Modelos de Programación].Repaint
148 Select Case [Lapso de la Demanda]
149 Case "Meses"
150 Primero = Month([Desde])
151 Cad2 = "Demanda Promedio Mensual"
152 Case "Bimestres"
153 Primero = IIf(Month([Desde]) <= 2, 1, IIf(Month([Desde]) <= 4,
2, IIf(Month([Desde]) <= 6, 3, IIf(Month([Desde]) <= 8, 4,
IIf(Month([Desde]) <= 10, 5, 6))))))
154 Cad2 = "Demanda Promedio Bimensual"
155 Case "Trimestres"
156 Primero = IIf(Month([Desde]) <= 3, 1, IIf(Month([Desde]) <= 6,
2, IIf(Month([Desde]) <= 9, 3, 4)))
157 Cad2 = "Demanda Promedio Trimestral"
158 End Select
159 Select Case [Valores de la Demanda]
160 Case 1
161 Cad1 = "[Promedio]"
162 Case 2
163 Cad1 = "[Máxima]"
164 Case 3
165 Cad1 = "[Mínima]"
166 Case 4
167 Cad1 = "[Promedio]+[DS]*" & Str([Veces la DS])

```

```
168 End Select
169 Select Case [Horizonte]
170 Case "Semestres"
171 Select Case [Lapso de la Demanda]
172 Case "Meses"
173 Último = Primero + 6 - 1
174 Case "Bimestres"
175 Último = Primero + 3 - 1
176 Case "Trimestres"
177 Último = Primero + 2 - 1
178 End Select
179 Case "Años"
180 Select Case [Lapso de la Demanda]
181 Case "Meses"
182 Último = Primero + 12 - 1
183 Case "Bimestres"
184 Último = Primero + 6 - 1
185 Case "Trimestres"
186 Último = Primero + 4 - 1
187 End Select
188 Case "Bienios"
189 Select Case [Lapso de la Demanda]
190 Case "Meses"
191 Último = Primero + 24 - 1
192 Case "Bimestres"
193 Último = Primero + 12 - 1
194 Case "Trimestres"
195 Último = Primero + 8 - 1
196 End Select
197 End Select
198 Set dbs = CurrentDb
199 Set rst = dbs.OpenRecordset("Productos en Modelos P")
200 With rst
201 CostoP = [Costo U Estimado]
202 CostoM = [Costo U de Mantener]
203 CostoE = [Costo U de Escasez ]
204 For i = Primero To Último
205 .AddNew
206 ![N° de Modelo] = [N° de Modelo]
207 ![Período] = i
208 ![Demanda] = DLookup(Cad1, Cad2, "[Período]=" &
209 ![Demanda] = ![Demanda] + [% de Exceso o Defecto] * ![Demanda]
210 ![Costo de Producción] = CostoP
211 ![Costo de Mantener] = CostoM
212 ![Costo de Escasez ] = CostoE
213 CostoP = CostoP + CostoP * [% Anual de Inflación] / 12
214 CostoM = CostoM + CostoM * [% Anual de Inflación] / 12
215 CostoE = CostoE + CostoE * [% Anual de Inflación] / 12
216 .Update
217 Next i
218 End With
219 rst.Close
220 [Subformulario Productos en Modelos P].Requery
221 DoCmd.SetWarnings True
222 End If
223 End If
```



```
224 End Sub
225
226 Private Sub Editar_Productos_Exit(Cancel As Integer)
227     Set ctlLista = Forms![Modelos de Programación]![Código del
228     Producto]
229     ctlLista.Requery
230 End Sub
231
232 Private Sub Form_Current()
233     If [¿Utilizando Historia?] Then
234         [Historia Desde].Enabled = True
235         [Historia Hasta].Enabled = True
236         [Valores de la Demanda].Enabled = True
237         [Veces la DS].Enabled = True
238         [Etiqueta67].Visible = True
239         [% de Exceso o Defecto].Enabled = True
240         [% Anual de Inflación].Enabled = True
241         [Demanda].Enabled = True
242     Else
243         [Historia Desde].Enabled = False
244         [Historia Hasta].Enabled = False
245         [Valores de la Demanda].Enabled = False
246         [Veces la DS].Enabled = False
247         [Etiqueta67].Visible = False
248         [% de Exceso o Defecto].Enabled = False
249         [% Anual de Inflación].Enabled = False
250         [Demanda].Enabled = False
251     End If
252 End Sub
253
254 Private Sub Horizonte_BeforeUpdate(Cancel As Integer)
255     If [Horizonte] <> "Años" And [Horizonte] <> "Semestres" And
256     [Horizonte] <> "Bienios" Then
257         MsgBox "Por favor, seleccione Bienios, Años o Semestres
258         Cancel = 1
259     End If
260 End Sub
261
262 Private Sub Lapso_de_la_Demanda_BeforeUpdate(Cancel As Integer)
263     If [Lapso de la Demanda] <> "Meses" And [Lapso de la Demanda] <>
264     "Bimestres" And [Lapso de la Demanda] <> "Trimestres" Then
265         MsgBox "Por favor, seleccione Meses, Bimestres o Trimestres
266         Cancel = 1
267     End If
268 End Sub
269
270 Private Sub Comando54_Click()
271     Dim strSQL As String, NumReg As Integer, CadObj As String, _
272     CadMac As String
273     Dim oApp As Object ' Variable que contiene la referencia de
274     Microsoft
275     Dim ExcelNoSeEjecutaba As Boolean ' Indicador para liberación final
276
277     DoCmd.SetWarnings False
278     strSQL = "DELETE * FROM [Demanda Intercambio Excel];"
279     DoCmd.RunSQL strSQL
```

```

276  strSQL = "INSERT INTO [Demanda Intercambio Excel] ( [N° de Modelo],
Período, Demanda, [Costo de Producción], [Costo de Mantener], [Costo
de Escasez ] ) " & _
277  "SELECT [Productos en Modelos P].[N° de Modelo], [Productos en
Modelos P].Período, [Productos en Modelos P].Demanda, [Productos en
Modelos P].[Costo de Producción], [Productos en Modelos P].[Costo
de Mantener], [Productos en Modelos P].[Costo de Escasez ] " & _
278  "FROM [Productos en Modelos P] " & _
279  "WHERE ((([Productos en Modelos P].[N° de
Modelo])=[forms]![Modelos de Programación]![N° de Modelo]));"
280  DoCmd.RunSQL strSQL
281  strSQL = "DELETE * FROM [Demanda Intercambio Excel Modelo];"
282  DoCmd.RunSQL strSQL
283  NumReg = DCount("[N° de Modelo]", "Productos en Modelos P", "[N° de
Modelo]=" & Str([N° de Modelo]))
284  strSQL = "INSERT INTO [Demanda Intercambio Excel Modelo] ( [Límite
de Producción], [Inventario Inicial], [N° de Registros] ) " & _
285  "SELECT [Modelos de Programación].[Límite de Producción],
[Modelos de Programación].[Inventario Inicial], " & Str(NumReg) & "
AS Expr1 " & _
286  "FROM [Modelos de Programación] " & _
287  "WHERE ((([Modelos de Programación].[N° de
Modelo])=[forms]![Modelos de Programación]![N° de Modelo]));"
288  DoCmd.RunSQL strSQL
289  DoCmd.SetWarnings True
290
291  ' Prueba para ver si hay una copia de Microsoft Excel ejecutándose.
292  On Error Resume Next ' Inicializa la interceptación del error.
293  ' La llamada a la función Getobject sin el primer argumento
devuelve
294  ' referencia a una instancia de la aplicación . Si no se está
295  ' se produce un error . Observe que se utiliza la coma como el
primer marcador del
296  ' argumento.
297  Set oApp = GetObject(, "Excel.Application")
298  If Err.Number <> 0 Then ExcelNoSeEjecutaba = True
299  ' Borra el objeto Err si se produce un error.
300  Err.Clear
301  ' Comprueba Excel. Si se está ejecutando Excel,
302  ' lo introduce en la tabla Running Object .
303  DetectExcel
304  ' Establece la variable de objeto para hacer referencia al archivo
que desea ver.
305  Select Case NumReg
306  Case 2
307  CadObj = "c:\datos\proula 97\Hojas\Modelo02.xls"
308  CadMac = "Modelo02.xls!Calcular2"
309  Case 3
310  CadObj = "c:\datos\proula 97\Hojas\Modelo03.xls"
311  CadMac = "Modelo03.xls!Calcular3"
312  Case 4
313  CadObj = "c:\datos\proula 97\Hojas\Modelo04.xls"
314  CadMac = "Modelo04.xls!Calcular4"
315  Case 6
316  CadObj = "c:\datos\proula 97\Hojas\Modelo06.xls"
317  CadMac = "Modelo06.xls!Calcular6"

```

```
318 Case 8
319 CadObj = "c:\datos\proula 97\Hojas\Modelo08.xls"
320 CadMac = "Modelo08.xls!Calcular8"
321 Case 12
322 CadObj = "c:\datos\proula 97\Hojas\Modelo12.xls"
323 CadMac = "Modelo12.xls!Calcular12"
324 Case 24
325 CadObj = "c:\datos\proula 97\Hojas\Modelo24.xls"
326 CadMac = "Modelo24.xls!Calcular24"
327 End Select
328 Set oApp = GetObject(CadObj)
329 oApp.Application.Visible = True
330 oApp.Parent.Windows(1).Visible = True
331 oApp.Application.Run CadMac
332 oApp.Application.DisplayAlerts = False
333 oApp.Application.Save
334 oApp.Application.Quit
335 oApp.Application.DisplayAlerts = True
336
337 ' Si no se está ejecutando esta copia de Microsoft Excel cuando
338 ' comenzó, ciérrela utilizando el método Quit de la propiedad
339 ' Observe que cuando intenta salir de Microsoft Excel, la barra de
340 ' de Microsoft Excel parpadea y Microsoft Excel muestra un mensaje
341 ' preguntándole si desea guardar los archivos cargados.
342 If ExcelNoSeEjecutaba Then
343 oApp.Application.Quit
344 End If
345 Set oApp = Nothing ' Libera la referencia a la aplicación y a la
346 hoja de cálculo.
347
348 'Comienza nuevamente el trabajo en Access
349 DoCmd.SetWarnings False
350 strSQL = "UPDATE [Productos en Modelos P] INNER JOIN [Modelos
351 Resultados Excel] " & _
352 "ON ([Productos en Modelos P].Período = [Modelos Resultados
353 Excel].Período) " & _
354 "AND ([Productos en Modelos P].[N° de Modelo] = [Modelos
355 Resultados Excel].[N° de Modelo]) " & _
356 "SET [Productos en Modelos P].[A Producir] = [Modelos Resultados
357 Excel]![A Producir], [Productos en Modelos P].[A Mantener] = "
358 "[Modelos Resultados Excel]![A Mantener], [Productos en Modelos
359 P].[A Demorar] = [Modelos Resultados Excel]![A Demorar];"
360 DoCmd.RunSQL strSQL
361 DoCmd.SetWarnings True
362
363 End Sub
364
365 Sub DetectExcel()
366 ' El procedimiento detecta un Excel en ejecución y lo registra.
367 Const WM_USER = 1024
368 Dim hWnd As Long
369 ' Si se está ejecutando Excel esta llamada API devuelve el
370 hWnd = FindWindow("XLMAIN", 0)
371 If hWnd = 0 Then
372 ' 0 quiere decir que Excel no se está ejecutando .
373 Exit Sub
```

```
368 Else
369 ' Excel se está ejecutando por lo que se utiliza la función API
SendMessage
370 ' para introducirlo en la tabla Running Object.
371 SendMessage hWnd, WM_USER + 18, 0, 0
372 End If
373 End Sub
374 Private Sub Editar_Productos_Click()
375 On Error GoTo Err_Editar_Productos_Click
376
377 Dim stDocName As String
378 Dim stLinkCriteria As String
379
380 stDocName = "Productos Terminados"
381 DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria
382
383 Exit_Editar_Productos_Click:
384 Exit Sub
385
386 Err_Editar_Productos_Click:
387 MsgBox Err.Description
388 Resume Exit_Editar_Productos_Click
389
390 End Sub
391 Private Sub Editar_Planes_de_Producción_Click()
392 On Error GoTo Err_Editar_Planes_de_Producción_Click
393
394 Dim stDocName As String
395 Dim stLinkCriteria As String
396
397 stDocName = "Planes de Producción"
398 DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria
399
400 Exit_Editar_Planes_de_Producción_Click:
401 Exit Sub
402
403 Err_Editar_Planes_de_Producción_Click:
404 MsgBox Err.Description
405 Resume Exit_Editar_Planes_de_Producción_Click
406
407 End Sub
408
409 Private Sub EncabezadoDePágina_Format(Cancel As Integer, FormatCount
As
410 CostoGP = DSum("[A Producir]*[Costo de Producción]", "Productos en
Modelos P", "[N° de Modelo]=" & Str([N° de Modelo]))
411 CostoGI = DSum("[A Mantener]*[Costo de Mantener]", "Productos en
Modelos P", "[N° de Modelo]=" & Str([N° de Modelo]))
412 CostoGE = DSum("[A Demorar]*[Costo de Escasez ]", "Productos en
Modelos P", "[N° de Modelo]=" & Str([N° de Modelo]))
413 If [Costo del Plan] <> 0 Then
414 PorcGP = CostoGP / [Costo del Plan]
415 PorcGI = CostoGI / [Costo del Plan]
416 PorcGE = CostoGE / [Costo del Plan]
417 Else
418 PorcGP = 0
```

```

419 PorcGI = 0
420 PorcGE = 0
421 End If
422 End Sub

```

### 36. Modelos EOQ

Origen del registro: Modelos EOQ

Nivel de grupo 0: N° de Modelo

Campo: Conclusión

```

=SiInm([CV de la Demanda]=0;"DATOS INSUFICIENTES";SiInm([CV de la
Demanda]<=0,25;"ACEPTABLE";SiInm([CV de la
Demanda]<=0,75;"DUDOSO";"INACEPTABLE")))

```

Código

```

1 Attribute VB_Name = "Report_Modelos EOQ"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8 Dim ctlLista As Control
9
10 Private Sub ¿Hay_Escasez __AfterUpdate()
11
12 [Tamaño Óptimo del Lote] = 0
13 [Ciclo Óptimo] = 0
14 [Máximo Nivel] = 0
15 [Faltante Óptimo] = 0
16 [Costo del Plan] = 0
17 [Reorden] = 0
18
19 If [¿Hay Escasez ?] Then
20 [Costo U de Escasez ].Visible = True
21 [Faltante Óptimo].Visible = True
22 [Máximo Nivel].Visible = True
23 [Etiqueta67].Visible = True
24 [Cuadro combinado66].Visible = True
25 Else
26 [Costo U de Escasez ].Visible = False
27 [Faltante Óptimo].Visible = False
28 [Máximo Nivel].Visible = False
29 [Etiqueta67].Visible = False
30 [Cuadro combinado66].Visible = False
31 End If
32 End Sub
33
34 Private Sub ¿Utilizando_Historia__AfterUpdate()
35 If [¿Utilizando Historia?] Then

```

```

36     [Historia Desde].Enabled = True
37     [Historia Hasta].Enabled = True
38     [Calcula Demanda].Enabled = True
39     [Media de la Demanda].Enabled = True
40     [DS de la Demanda].Enabled = True
41     [CV de la Demanda].Enabled = True
42     [Conclusión].Enabled = True
43     Else
44     [Historia Desde].Enabled = False
45     [Historia Hasta].Enabled = False
46     [Calcula Demanda].Enabled = False
47     [Media de la Demanda].Enabled = False
48     [DS de la Demanda].Enabled = False
49     [CV de la Demanda].Enabled = False
50     [Conclusión].Enabled = False
51     End If
52 End Sub
53 Sub CalculaEstadísticas(F1 As Date, F2 As Date, Inc As Integer)
54
55     Dim dbs As Database, rst As Recordset, F As Date, _
56         s As Single, s2 As Single, n As Integer, _
57         CV As Single, strSQL As String
58
59     F = F1
60     s = 0
61     s2 = 0
62     n = 0
63     Set dbs = CurrentDb
64     Do While F <= F2
65         n = n + 1
66         strSQL = "SELECT Producto, Sum([Demanda Puntual]) AS [Unidades
67 Demandas] " & _
68             "FROM [Demanda por Fechas Tabla] " & _
69             "WHERE ((([Fecha Base]) Between #" & Str(F) & "# And #" & IIf(F
+ 7 - 1 > [Historia Hasta], Str([Historia Hasta]), Str(F + Inc - 1))
& "#)) " & _
70             "GROUP BY Producto;"
71     Set rst = dbs.OpenRecordset(strSQL)
72     If rst.RecordCount <> 0 Then
73         rst.MoveFirst
74         s = s + rst![Unidades Demandas]
75         s2 = s2 + rst![Unidades Demandas] ^ 2
76     End If
77     rst.Close
78     F = F + Inc
79     Loop
80     [Media de la Demanda] = s / n
81     [DS de la Demanda] = Sqr((s2 - s ^ 2 / n) / (n - 1))
82     [CV de la Demanda] = n * Sqr((s2 - s ^ 2 / n) / (n - 1)) / s
83 End Sub
84 Private Sub Calcula_Demanda_Click()
85
86     Dim strSQL As String, dbs As Database, rst As Recordset, _
87         NumDías As Integer
88
89     If [Historia Desde] = Null Or [Historia Hasta] = Null Then

```

```

89     MsgBox "Entre primero las fechas históricas"
90     Else
91         NumDías = NDías([Historia Desde], [Historia Hasta])
92         If [Lapso de la Demanda] = Null Then
93             MsgBox "Entre primero el Lapso de la Demanda"
94         Else
95             DoCmd.SetWarnings False
96             strSQL = "DELETE * FROM [Demanda por Fechas Tabla];"
97             DoCmd.RunSQL strSQL
98             strSQL = "PARAMETERS Forms![Modelos EOQ]![Historia Desde]
DateTime, Forms![Modelos EOQ]![Historia Hasta] DateTime;" & _
99             " INSERT INTO [Demanda por Fechas Tabla] ( Producto, [Fecha
Base], [Demanda Puntual], [Demanda en Bs] )" & _
100            " SELECT [Demanda por Fechas 0].Producto, [Demanda por Fechas
0].[Fecha Base], [Demanda por Fechas 0].[Demanda Puntual], [Demanda
por Fechas 0].[Demanda en Bs]" & _
101            " FROM [Demanda por Fechas 0] WHERE ((([Demanda por Fechas
0].[Fecha Base]) Between [Forms]![Modelos EOQ]![Historia Desde] And"
& _
102            " [Forms]![Modelos EOQ]![Historia Hasta]) And
([Producto]=[Forms]![Modelos EOQ]![Código del Producto]));"
103             DoCmd.RunSQL strSQL
104             strSQL = "SELECT Producto, Sum([Demanda Puntual]) AS [Unidades
Demandas] FROM [Demanda por Fechas Tabla] GROUP BY Producto;"
105             Set dbs = CurrentDb
106             Set rst = dbs.OpenRecordset(strSQL)
107             rst.MoveFirst
108             Select Case [Lapso de la Demanda]
109                 Case "Días"
110                     [Demanda en el Período] = rst![Unidades Demandas] / NumDías
111                     rst.Close
112                     CalculaEstadísticas [Historia Desde], [[Historia Hasta], 7
113                 Case "Semanas"
114                     If NumDías < 7 Then
115                         MsgBox "Las fechas históricas deben incluir al menos una
semana"
116                     Else
117                         [Demanda en el Período] = rst![Unidades Demandas] /
NSemanas([Historia Desde], [Historia Hasta])
118                         rst.Close
119                         CalculaEstadísticas [Historia Desde], [Historia Hasta], 7
120                     End If
121                 Case "Quincenas"
122                     If NumDías < 14 Then
123                         MsgBox "Las fechas históricas deben incluir al menos una
quincena"
124                     Else
125                         [Demanda en el Período] = rst![Unidades Demandas] /
NQuincenas([Historia Desde], [Historia Hasta])
126                         CalculaEstadísticas [Historia Desde], [Historia Hasta], 14
127                     End If
128                 Case "Meses"
129                     If NumDías < 28 Then
130                         MsgBox "Las fechas históricas deben incluir al menos un mes"
131                     Else
132                         [Demanda en el Período] = rst![Unidades Demandas] /

```

```

    NMeses([Historia Desde], [Historia Hasta])
133     CalculaEstadísticas [Historia Desde], [Historia Hasta], 28
134     End If
135     Case "Bimestres"
136     If NumDías < 60 Then
137     MsgBox "Las fechas históricas deben incluir al menos un
bimestre"
138     Else
139     [Demanda en el Período] = rst![Unidades Demandas] /
NBimestres([Historia Desde], [Historia Hasta])
140     CalculaEstadísticas [Historia Desde], [Historia Hasta], 60
141     End If
142     Case "Trimestres"
143     If NumDías < 90 Then
144     MsgBox "Las fechas históricas deben incluir al menos un
trimestre"
145     Else
146     [Demanda en el Período] = rst![Unidades Demandas] /
NDías([Historia Desde], [Historia Hasta])
147     CalculaEstadísticas [Historia Desde], [Historia Hasta], 90
148     End If
149     Case "Semestres"
150     If NumDías < 180 Then
151     MsgBox "Las fechas históricas deben incluir al menos un
semestre"
152     Else
153     [Demanda en el Período] = rst![Unidades Demandas] /
NSemestres([Historia Desde], [Historia Hasta])
154     CalculaEstadísticas [Historia Desde], [Historia Hasta], 180
155     End If
156     Case "Años"
157     If NumDías < 360 Then
158     MsgBox "Las fechas históricas deben incluir al menos un año"
159     Else
160     [Demanda en el Período] = rst![Unidades Demandas] /
NAños([Historia Desde], [Historia Hasta])
161     CalculaEstadísticas [Historia Desde], [Historia Hasta], 360
162     End If
163     Case "Bienios"
164     If NumDías < 720 Then
165     MsgBox "Las fechas históricas deben incluir al menos un
bienio"
166     Else
167     [Demanda en el Período] = rst![Unidades Demandas] /
Nbiaños([Historia Desde], [Historia Hasta])
168     CalculaEstadísticas [Historia Desde], [Historia Hasta], 720
169     End If
170     End Select
171     DoCmd.SetWarnings True
172     End If
173     End If
174 End Sub
175
176 Private Sub Código_del_Producto_AfterUpdate()
177     [Costo U Estimado] = DLookup("[Costo Unitario]", "Productos
Terminados", "[Código del Producto]='" & [Código del Producto] &
```



```

    """)
178 End Sub
179
180 Private Sub Comando70_Exit(Cancel As Integer)
181   Set ctlLista = Forms![Modelos EOQ]!Horizonte
182   ctlLista.Requery
183   Set ctlLista = Forms![Modelos EOQ]![Lapso de la Demora]
184   ctlLista.Requery
185   Set ctlLista = Forms![Modelos EOQ]![Lapso de la Demanda]
186   ctlLista.Requery
187 End Sub
188
189 Private Sub Comando72_Exit(Cancel As Integer)
190   Set ctlLista = Forms![Modelos EOQ]![Código del Producto]
191   ctlLista.Requery
192 End Sub
193
194 Private Sub Comando73_Click()
195   Dim Cadena As String
196
197   Cadena = "Ciudadano... Esta acción cambiará el valor del campo " & _
198     "[Punto de Reorden], en la Tabla 'Productos Terminados', " & _
199     "actualizándolo a " & Str(Forms![Modelos EOQ]![Reorden]) & _
200     " lo que puede afectar la operación actual del sistema."
201   If MsgBox(Cadena, vbOKCancel, "Confirmación") = vbOK Then
202     Cadena = "UPDATE [Productos Terminados] " & _
203       "SET [Punto de Reorden] = " & Str([Reorden]) & _
204       " WHERE (([Código del Producto]='" & [Código del Producto] &
205     "'));"
206     DoCmd.SetWarnings False
207     DoCmd.RunSQL Cadena
208     DoCmd.SetWarnings True
209   End If
210 End Sub
211
212 Private Sub Correr_Click()
213   Dim Pond As Single
214
215   Pond = DLookup("[Ponderador]", "Conversiones", "[De]='" & [Lapso de
216     la Demora] & "' and [A]='" & [Lapso de la Demanda] & "'")
217   If [¿Hay Escasez ?] Then
218     [Tamaño Óptimo del Lote] = CInt(Sqr(2 * [Demanda en el Período] *
219     [Costo U de Ordenar] * _
220     ([Costo U de Escasez ] + [Costo U de Mantener]) / ([Costo U
221     de Mantener] * _
222     [Costo U de Escasez ])))
223     [Ciclo Óptimo] = [Tamaño Óptimo del Lote] / [Demanda en el
224     [Máximo Nivel] = CInt(Sqr(2 * [Demanda en el Período] * [Costo U
225     de Ordenar] * _
226     [Costo U de Escasez ] / ([Costo U de Mantener] *
227     ([Costo U de Escasez ] + [Costo U de Mantener])))
228     [Faltante Óptimo] = [Tamaño Óptimo del Lote] - [Máximo Nivel]
229     [Reorden] = CInt(-[Faltante Óptimo] + [Demanda en el Período] *
230     [Demora] * Pond)
231     [Costo del Plan] = [Demanda en el Período] * [Costo U de Ordenar]
232     / [Tamaño Óptimo del Lote] + _

```

```
226     [Demanda en el Período] * [Costo U Estimado] + _
227     [Costo U de Mantener] * [Máximo Nivel] ^ 2 / (2 * [Tamaño
Óptimo del Lote]) + _
228     [Costo U de Escasez ] * [Faltante Óptimo] ^ 2 / (2 * [Tamaño
Óptimo del Lote])
229     Else
230     [Tamaño Óptimo del Lote] = CInt(Sqr(2 * [Demanda en el Período] *
[Costo U de Ordenar] / [Costo U de Mantener]))
231     [Ciclo Óptimo] = [Tamaño Óptimo del Lote] / [Demanda en el
232     [Reorden] = CInt([Demanda en el Período] * [Demora] * Pond)
233     [Costo del Plan] = [Demanda en el Período] * [Costo U de Ordenar]
/ [Tamaño Óptimo del Lote] + _
234     [Costo U de Mantener] * [Tamaño Óptimo del Lote] / 2 + _
235     [Demanda en el Período] * [Costo U Estimado]
236     End If
237 End Sub
238
239 Private Sub Form_Current()
240
241     If [¿Utilizando Historia?] Then
242     [Historia Desde].Enabled = True
243     [Historia Hasta].Enabled = True
244     [Calcula Demanda].Enabled = True
245     [Media de la Demanda].Enabled = True
246     [DS de la Demanda].Enabled = True
247     [CV de la Demanda].Enabled = True
248     [Conclusión].Enabled = True
249     Else
250     [Historia Desde].Enabled = False
251     [Historia Hasta].Enabled = False
252     [Calcula Demanda].Enabled = False
253     [Media de la Demanda].Enabled = False
254     [DS de la Demanda].Enabled = False
255     [CV de la Demanda].Enabled = False
256     [Conclusión].Enabled = False
257     End If
258
259     [Tamaño Óptimo del Lote] = 0
260     [Ciclo Óptimo] = 0
261     [Máximo Nivel] = 0
262     [Faltante Óptimo] = 0
263     [Reorden] = 0
264     [Costo del Plan] = 0
265
266     If [¿Hay Escasez ?] Then
267     [Costo U de Escasez ].Visible = True
268     [Faltante Óptimo].Visible = True
269     [Máximo Nivel].Visible = True
270     [Etiqueta67].Visible = True
271     [Cuadro combinado66].Visible = True
272     Else
273     [Costo U de Escasez ].Visible = False
274     [Faltante Óptimo].Visible = False
275     [Máximo Nivel].Visible = False
276     [Etiqueta67].Visible = False
277     [Cuadro combinado66].Visible = False
```

```
278 End If
279 End Sub
280 Private Sub Comando70_Click()
281 On Error GoTo Err_Comando70_Click
282
283 Dim stDocName As String
284 Dim stLinkCriteria As String
285
286 stDocName = "Unidades de Medición"
287 DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria
288
289 Exit_Comando70_Click:
290 Exit Sub
291
292 Err_Comando70_Click:
293 MsgBox Err.Description
294 Resume Exit_Comando70_Click
295
296 End Sub
297 Private Sub Comando71_Click()
298 On Error GoTo Err_Comando71_Click
299
300 Dim stDocName As String
301 Dim stLinkCriteria As String
302
303 stDocName = "Conversiones"
304 DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria
305
306 Exit_Comando71_Click:
307 Exit Sub
308
309 Err_Comando71_Click:
310 MsgBox Err.Description
311 Resume Exit_Comando71_Click
312
313 End Sub
314 Private Sub Comando72_Click()
315 On Error GoTo Err_Comando72_Click
316
317 Dim stDocName As String
318 Dim stLinkCriteria As String
319
320 stDocName = "Productos Terminados"
321 DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria
322
323 Exit_Comando72_Click:
324 Exit Sub
325
326 Err_Comando72_Click:
327 MsgBox Err.Description
328 Resume Exit_Comando72_Click
329
330 End Sub
331
332 Private Sub Plan_Click()
333 Dim dbs As Database, tbl As Recordset, F As Date, _
```

```
334     F1 As Date, F2 As Date, NPlan As Long, NDías As Integer
335
336     Set dbs = CurrentDb
337     Set tbl = dbs.OpenRecordset("Planes de Producción", dbOpenTable)
338     With tbl
339         .AddNew
340         !Descripción = "Plan de Producción Generado con Modelos EOQ el " &
Str(Date) & ". " & [Descripción]
341         ![Fecha de Inicio] = [Desde]
342         Select Case [Horizonte]
343             Case "Días"
344                 ![Fecha de Culminación] = [Desde] + 1
345             Case "Semanas"
346                 ![Fecha de Culminación] = [Desde] + 7
347             Case "Quincenas"
348                 ![Fecha de Culminación] = [Desde] + 15
349             Case "Meses"
350                 ![Fecha de Culminación] = [Desde] + 30
351             Case "Bimestres"
352                 ![Fecha de Culminación] = [Desde] + 60
353             Case "Trimestres"
354                 ![Fecha de Culminación] = [Desde] + 90
355             Case "Semestres"
356                 ![Fecha de Culminación] = [Desde] + 120
357             Case "Años"
358                 ![Fecha de Culminación] = [Desde] + 365
359             Case "Bienios"
360                 ![Fecha de Culminación] = [Desde] + 730
361         End Select
362         .Update
363         .MoveLast
364         NPlan = ![N° del Plan]
365         F1 = ![Fecha de Inicio]
366         F2 = ![Fecha de Culminación]
367     End With
368     tbl.Close
369
370     Set tbl = dbs.OpenRecordset("Productos en Planes", dbOpenTable)
371     Select Case [Lapso de la Demanda]
372         Case "Días"
373             NDías = [Ciclo Óptimo]
374         Case "Semanas"
375             NDías = [Ciclo Óptimo] * 7
376         Case "Quincenas"
377             NDías = [Ciclo Óptimo] * 15
378         Case "Meses"
379             NDías = [Ciclo Óptimo] * 30
380         Case "Bimestres"
381             NDías = [Ciclo Óptimo] * 60
382         Case "Trimestres"
383             NDías = [Ciclo Óptimo] * 90
384         Case "Semestres"
385             NDías = [Ciclo Óptimo] * 180
386         Case "Años"
387             NDías = [Ciclo Óptimo] * 365
388         Case "Bienios"
```

```
389     NDías = [Ciclo Óptimo] * 730
390 End Select
391
392 With tbl
393     F = F1 + NDías
394     Do While F <= F2
395         .AddNew
396         ![N° del Plan] = NPlan
397         ![Código del Producto] = [Código del Producto]
398         ![Fecha] = F
399         ![Cantidad] = [Tamaño Óptimo del Lote]
400         .Update
401         F = F + NDías
402     Loop
403 End With
404 tbl.Close
405
406 End Sub
407 Private Sub Comando78_Click()
408 On Error GoTo Err_Comando78_Click
409
410     Dim stDocName As String
411     Dim stLinkCriteria As String
412
413     stDocName = "Planes de Producción"
414     DoCmd.OpenForm stDocName, , , , , acDialog, stLinkCriteria
415
416 Exit_Comando78_Click:
417     Exit Sub
418
419 Err_Comando78_Click:
420     MsgBox Err.Description
421     Resume Exit_Comando78_Click
422
423 End Sub
424
425 Private Sub EncabezadoDelGrupo0_Format(Cancel As Integer,
FormatCount As Integer)
426     If [¿Utilizando Historia?] Then
427         [Historia Desde].Visible = True
428         [Historia Hasta].Visible = True
429         [Media de la Demanda].Visible = True
430         [DS de la Demanda].Visible = True
431         [CV de la Demanda].Visible = True
432         [Conclusión].Visible = True
433     Else
434         [Historia Desde].Visible = False
435         [Historia Hasta].Visible = False
436         [Media de la Demanda].Visible = False
437         [DS de la Demanda].Visible = False
438         [CV de la Demanda].Visible = False
439         [Conclusión].Visible = False
440     End If
441     If [¿Hay Escasez ?] Then
442         [Costo U de Escasez ].Visible = True
443         [Faltante Óptimo].Visible = True
```

```

444     [Máximo Nivel].Visible = True
445     [Etiqueta67].Visible = True
446     [Cuadro combinado66].Visible = True
447     Else
448     [Costo U de Escasez ].Visible = False
449     [Faltante Óptimo].Visible = False
450     [Máximo Nivel].Visible = False
451     [Etiqueta67].Visible = False
452     [Cuadro combinado66].Visible = False
453     End If
454     End Sub

```

### 37. Notas en Facturas subinforme

#### Origen del registro:

```

SELECT DISTINCTROW [Notas en Facturas].[N° de Factura], [Notas en
Facturas].[Nota N°], [Notas].[Contenido]
FROM ([Notas] INNER JOIN [Notas en Facturas] ON [Notas].[Nota N°] =[Notas
en Facturas].[Nota N°]);

```

### 38. Órdenes de Compra

#### Origen del registro:

```

SELECT DISTINCTROW [Órdenes de Compra].[N°], [Órdenes de
Compra].Proveedor, Proveedores.[Nombre o Razón Social],
Proveedores.[CI o RIF], Proveedores.Dirección, Proveedores.País,
Proveedores.Estado, Proveedores.Ciudad, [Órdenes de Compra].Fecha,
[Órdenes de Compra].[Cargos %]
FROM Proveedores INNER JOIN [Órdenes de Compra] ON Proveedores.[Código
del Proveedor] = [Órdenes de Compra].Proveedor;

```

#### Nivel de grupo 0: N°

#### Campo: Subtotal

```

=IIf(Not IsNull([N°]),DSum("([Precio Unitario]*[Cantidad])", "Insumos en
Órdenes", "[Orden N°]=" & [N°]),0)

```

#### Campo: Subtotal+Cargos

```

=[Subtotal]*[Cargos %]

```

#### Campo: Texto43

```

=[Subtotal]+([Subtotal]*[Cargos %])

```

### 39. Órdenes de Producción

#### Origen del registro: Órdenes de Producción

Campo: Texto36

```
=[Productos Terminados subinforme].[Report]![Unidades de Presentación]
```

Campo: Texto67

```
=[Productos Terminados subinforme].[Report]![Unidades de Presentación]
```

Campo: Texto68

```
=[Cantidad]/[Productos Terminados subinforme].[Report]![Cantidad en
Presentación]
```

Campo: Texto70

```
=[Productos Terminados subinforme].[Report]![Presentación]
```

Campo: Texto72

```
=Int([Producción Efectiva]/[Productos Terminados
subinforme].[Report]![Cantidad en Presentación])
```

Campo: Texto74

```
=[Productos Terminados subinforme].[Report]![Presentación]
```

**40. Países subinforme**Origen del registro:

```
SELECT DISTINCTROW [Países].[País], [Países].[Código Telefónico]
FROM [Países];
```

**41. Pedidos de Clientes**Origen del registro: Pedidos de ClientesCódigo

```
1 Attribute VB_Name = "Report_Pedidos de Clientes"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8 Private Sub EncabezadoDePágina_Format(Cancel As Integer, FormatCount
As Integer)
9 [Forms]![Pedir Dos Fechas]![FechaÚnica] = [Fecha]
10
11 End Sub
```

```

12
13 Private Sub Report_Close()
14 DoCmd.Close acForm, "Pedir Dos Fechas"
15 End Sub
16
17 Private Sub Report_Open(Cancel As Integer)
18 DoCmd.OpenForm "Pedir Dos Fechas", , , , , acHidden
19 End Sub

```

## 42. Pedidos de Clientes subinforme

### Origen del registro:

```

SELECT DISTINCTROW [Pedidos de Clientes].[N° de Pedido], [Pedidos de
    Clientes].[Código del Cliente], [Pedidos de Clientes].Fecha,
    [Pedidos de Clientes].Vendedor, [Pedidos de Clientes].Condición,
    [Pedidos de Clientes].[Orden de Compra]
FROM [Pedidos de Clientes];

```

## 43. Personas subinforme

### Origen del registro:

```

SELECT DISTINCTROW [Personas].[Cédula de Identidad o Clave],
    [Personas].[Apellidos], [Personas].[Nombres]
FROM [Personas];

```

## 44. Planes de Producción Globales 1

### Origen del registro: Planes de Producción Globales 1

Nivel de grupo 0: Departamento

Nivel de grupo 1: Nombre

Campo: Porcentaje Producción

=[Total Producción]/[Total General Producción]

Campo: Porcentaje Unidades

=[Total Unidades]/[Total General Unidades]

Campo: Texto49

=[Total Parcial Unidades]/[Total General Unidades]

Campo: Texto51



=[Total Parcial Producción]/[Total General Producción]

**Campo: Total General Producción**

=Sum([Precio Estimado]\*[Total Unidades])

**Campo: Total General Unidades**

=Sum([Total Unidades])

**Campo: Total Parcial Producción**

=Sum([Precio Estimado]\*[Total Unidades])

**Campo: Total Parcial Unidades**

=Sum([Total Unidades])

## **45. Planes de Producción Mensuales**

**Origen del registro: Planes de Producción Mensuales RC**

**Nivel de grupo 0: Departamento**

**Campo: Texto109**

=DLookup("[Descripción]","Planes de Producción","[N° del Plan]=" & [N° del Plan])

**Campo: Texto111**

=DLookup("[Fecha de Inicio]","Planes de Producción","[N° del Plan]=" & [N° del Plan])

**Campo: Texto113**

=DLookup("[Fecha de Culminación]","Planes de Producción","[N° del Plan]=" & [N° del Plan])

**Código**

```

1  Attribute VB_Name = "Report_Planes de Producción Mensuales"
2  Attribute VB_Creatable = True
3  Attribute VB_PredeclaredId = True
4  Attribute VB_Exposed = False
5  Option Compare Database
6  Option Explicit
7
8  ' Constante para el número máximo de columnas que crearía la
9  ' consulta VentasPorEmpleado más 1 para una columna de totales.
10 Const NCamposVar = 12
11 Const NCamposFijos = 5
12
13 ' Variables para los objetos Database y Recordset.
```

```
14 Dim bdInforme As Database
15 Dim rstInforme As Recordset
16
17 ' Variables para el número de columnas y filas y para los totales
del informe.
18 Dim entCuentaColumnas As Integer
19 Dim lngRgColumnTotal(1 To NCamposVar) As Long
20 Dim GrpColumnTotal(1 To NCamposVar) As Long
21
22 Private Sub Detalle_Format(Cancelar As Integer, ContarFormato As
Integer)
23 ' Colocar valores en los cuadros de texto y ocultar los cuadros de
texto no utilizados.
24
25 Dim entX As Integer
26 ' Comprobar que no ha llegado al final del conjunto de registros.
27 If Not rstInforme.EOF Then
28 ' Si ContarFormato es 1, colocar los valores del conjunto de
registros en los cuadros de edición
29 ' de la sección Detalle.
30 If Me.FormatCount = 1 Then
31 For entX = NCamposFijos To entCuentaColumnas - 1
32 ' Convertir los valores Null a 0.
33 Me("Col" + Format(entX - NCamposFijos + 1)) =
xTabCnulos(rstInforme(entX))
34 Next entX
35
36 ' Ocultar los cuadros de edición no utilizados de la sección
Detalle.
37 For entX = entCuentaColumnas To NCamposVar + NCamposFijos - 1
38 Me("Col" + Format(entX - NCamposFijos + 1)).Visible = False
39 Next entX
40
41 ' Ir al siguiente registro del conjunto de registros.
42 rstInforme.MoveNext
43 End If
44 End If
45
46 End Sub
47
48 Private Sub IniciarVarGrp()
49
50 Dim entX As Integer
51
52 ' Iniciar la matriz que almacena los totales de las columnas.
53 For entX = 1 To NCamposVar
54 GrpColumnTotal(entX) = 0
55 Next entX
56
57 End Sub
58
59 Private Sub Detalle_Print(Cancelar As Integer, ContarImpresión As
60
61 Dim entX As Integer
62
63 ' Si ContarImpresión es 1 agregar a los totales de columnas.
```

```
64   If Me.PrintCount = 1 Then
65
66     For entX = 1 To entCuentaColumnas - NCamposFijos
67       ' Comenzando en la columna variable 1 (primer cuadro de texto con
        valor de referencias cruzadas),
68       ' Agregar valor de referencias cruzadas al total de la columna
        actual.
69       lngRgColumnTotal(entX) = lngRgColumnTotal(entX) + Me("Col" +
        Format(entX))
70       GrpColumnTotal(entX) = GrpColumnTotal(entX) + Me("Col" +
        Format(entX))
71     Next entX
72
73   End If
74 End Sub
75 Private Sub Detalle_Retreat()
76
77   ' Volver siempre al registro anterior al retirarse de la sección
78   rstInforme.MovePrevious
79
80 End Sub
81 Private Sub IniciarVariables()
82
83   Dim entX As Integer
84
85   ' Iniciar la matriz que almacena los totales de las columnas.
86   For entX = 1 To NCamposVar
87     lngRgColumnTotal(entX) = 0
88   Next entX
89
90 End Sub
91
92 Private Sub EncabezadoDelGrupo1_Format(Cancelar As Integer,
        ContarFormato As Integer)
93
94   Dim entX As Integer
95
96   ' Inicializar los totales del grupo
97   IniciarVarGrp
98
99   ' Colocar los encabezados de columnas en los cuadros de edición del
        encabezado de página.
100  For entX = 1 To entCuentaColumnas - NCamposFijos
101    Me("Enc" + Format(entX)) = rstInforme(NCamposFijos + entX -
102    Next entX
103
104  ' Ocultar los cuadros de edición no utilizados en el encabezado de
        página.
105  For entX = entCuentaColumnas To NCamposVar + NCamposFijos - 1
106    Me("Enc" + Format(entX - NCamposFijos + 1)).Visible = False
107  Next entX
108 End Sub
109
110 Private Sub Report_Close()
111
112   On Error Resume Next
```

```
113
114 ' Cerrar conjunto de registros.
115 rstInforme.Close
116
117 End Sub
118
119 Private Sub Report_NoData(Cancelar As Integer)
120 MsgBox "Ningún registro coincide con el criterio que introdujo.",
vbExclamation, "No se encontró ningún registro"
121 rstInforme.Close
122 Cancelar = True
123 End Sub
124
125 Private Sub Report_Open(Cancelar As Integer)
126
127 ' Crear conjunto de registros base para el informe utilizando el
criterio introducido
128 ' en el formulario CuadroDeDiálogoVentasPorEmpleado.
129
130 Dim entX As Integer
131 Dim qdf As QueryDef
132 Dim formulario As Form
133
134 ' No abrir el informe si no está cargado el formulario Pedir Dos
Fechas.
135 If Not (EstáCargado("Pedir Número del Plan")) Then
136 Cancelar = True
137 MsgBox "Para obtener una vista previa de este informe o para
imprimirlo, debe abrir " _
138 & "Pedir Número del Plan en Vista Formulario.", vbExclamation, _
139 "Debe abrir el cuadro de diálogo"
140 Exit Sub
141 End If
142
143 ' Establecer la variable de la base de datos a la base de datos
144 Set bdInforme = CurrentDb
145 Set formulario = Forms![Pedir Número del Plan]
146 ' Abrir el objeto QueryDef.
147 Set qdf = bdInforme.QueryDefs("Planes de Producción Mensuales RC")
148 ' Establecer los parámetros para la consulta basándose en los
valores
149 ' introducidos en el formulario CuadroDeDiálogoVentasPorEmpleado.
150 qdf.Parameters("Formularios![Pedir Número del Plan]![Plan]") =
formulario![Plan]
151
152 ' Abrir el objeto Recordset.
153 Set rstInforme = qdf.OpenRecordset
154
155 ' Establecer una variable para guardar el número de columnas de la
consulta de referencias cruzadas.
156 entCuentaColumnas = rstInforme.Fields.Count
157
158 End Sub
159 Private Sub PieDelGrupo2_Print(Cancelar As Integer, ContarImpresión
As Integer)
160
```

```
161 Dim entX As Integer
162
163 ' Colocar los totales de columnas en los cuadros de edición del pie
del informe.
164 ' Comenzar en la columna 2 (primer cuadro de edición con valores de
referencias cruzadas).
165 For entX = 1 To entCuentaColumnas - NCamposFijos
166 Me("Tot" + Format(entX)) = GrpColumnTotal(entX)
167 Next entX
168
169 ' Ocultar los cuadros de edición no utilizados del pie del informe.
170 For entX = entCuentaColumnas To NCamposVar + NCamposFijos - 1
171 Me("Tot" + Format(entX - NCamposFijos + 1)).Visible = False
172 Next entX
173 End Sub
174
175 Private Sub PieDelInforme_Print(Cancelar As Integer, ContarImpresión
As Integer)
176
177 Dim entX As Integer
178
179 ' Colocar los totales de columnas en los cuadros de edición del pie
del informe.
180 ' Comenzar en la columna 2 (primer cuadro de edición con valores de
referencias cruzadas).
181 For entX = 1 To entCuentaColumnas - NCamposFijos
182 Me("TotG" + Format(entX)) = lngRgColumnTotal(entX)
183 Next entX
184
185 ' Ocultar los cuadros de edición no utilizados del pie del informe.
186 For entX = entCuentaColumnas To NCamposVar + NCamposFijos - 1
187 Me("TotG" + Format(entX - NCamposFijos + 1)).Visible = False
188 Next entX
189 End Sub
190
191 Private Sub EncabezadoDelInforme_Format(Cancelar As Integer,
ContarFormato As Integer)
192
193 ' Ir al primer registro del conjunto de registros al iniciar el
194 ' o al reiniciar el informe. (Un informe se reinicia cuando
195 ' se imprime un informe desde la ventana Vista preliminar o al
volver
196 ' a una página anterior durante la vista previa.)
197 rstInforme.MoveFirst
198
199 ' Iniciar las variables.
200 IniciarVariables
201
202 End Sub
203
204 Private Function xtabCnulos(varX As Variant)
205
206 ' Probar si un valor es nulo.
207 If IsNull(varX) Then
208 ' Si varX es nulo, establecer varX a 0.
209 xtabCnulos = 0
```

```

210 Else
211     ' De lo contrario, devolver varX.
212     xtabCnulos = varX
213 End If
214
215 End Function

```

#### 46. Precios de Productos subinforme

##### Origen del registro:

```

SELECT DISTINCTROW [Precios de Productos].[Código del Producto], [Precios
de Productos].[Fecha], [Precios de Productos].[Precio Unitario de
Venta]
FROM [Precios de Productos];

```

#### 47. Productos Facturados a Clientes subinforme

##### Origen del registro:

```

SELECT DISTINCTROW [Productos Pedidos y Facturados].[N° de Factura],
[Lista de Precios Vigente en Fecha Buscada].Nombre, [Productos
Pedidos y Facturados].[Código del Producto], [Productos Pedidos y
Facturados].[Cantidad Pedida], [Productos Pedidos y
Facturados].[Cantidad Facturada], [Lista de Precios Vigente en Fecha
Buscada].Departamento, [Lista de Precios Vigente en Fecha
Buscada].[Precio Unitario de Venta], [Lista de Precios Vigente en
Fecha Buscada].[% Descuento], [Lista de Precios Vigente en Fecha
Buscada].[Precio con Descuento], [Lista de Precios Vigente en Fecha
Buscada]![Precio con Descuento]*[Productos Pedidos y
Facturados]![Cantidad Facturada] AS [Monto Total]
FROM [Productos Pedidos y Facturados] INNER JOIN [Lista de Precios
Vigente en Fecha Buscada] ON [Productos Pedidos y
Facturados].[Código del Producto] = [Lista de Precios Vigente en
Fecha Buscada].[Código del Producto]
WHERE ((([Lista de Precios Vigente en Fecha Buscada].[Límite
Inferior])<=[Cantidad Pedida]) AND ((([Lista de Precios Vigente en
Fecha Buscada].[Límite Superior])>=[Cantidad Pedida]));

```

##### Nivel de grupo 0: Departamento

##### Campo: Texto17

```
=Sum([Monto Total])
```

##### Campo: Texto20

```
=Sum([Monto Total])
```

## 48. Productos Pedidos de Clientes subinforme

### Origen del registro:

```
SELECT DISTINCTROW [Productos Pedidos de Clientes].[N° de Pedido],
  [Productos Pedidos de Clientes].[Código del Producto], [Lista de
  Precios Vigente en Fecha Buscada].Nombre, [Productos Pedidos de
  Clientes].[Cantidad Pedida], [Lista de Precios Vigente en Fecha
  Buscada].[Precio Unitario de Venta], [Lista de Precios Vigente en
  Fecha Buscada].[% Descuento], [Lista de Precios Vigente en Fecha
  Buscada].[Precio con Descuento], [Cantidad Pedida]*[Precio con
  Descuento] AS [Monto a Cancelar]
FROM [Productos Pedidos de Clientes] INNER JOIN [Lista de Precios Vigente
  en Fecha Buscada] ON [Productos Pedidos de Clientes].[Código del
  Producto] = [Lista de Precios Vigente en Fecha Buscada].[Código del
  Producto]
WHERE ((([Lista de Precios Vigente en Fecha Buscada].[Límite
  Inferior])<=[Cantidad Pedida]) AND (([Lista de Precios Vigente en
  Fecha Buscada].[Límite Superior])>=[Cantidad Pedida]));
```

### Nivel de grupo 0: N° de Pedido

#### Campo: Monto Total Solicitado

```
=Sum([Monto a Cancelar])
```

## 49. Productos Pedidos y Despachados entre Fechas

### Origen del registro: Productos Pedidos y Despachados entre Fechas

### Nivel de grupo 0: Total Ventas

#### Campo: TBDespachado

```
=Sum([Total Despachado])
```

#### Campo: TBInsatisfecho

```
=Sum([Total Insatisfecho])
```

#### Campo: TBVendidas

```
=Sum([Total Ventas])
```

#### Campo: Texto25

```
=[Forms]![Pedir Fechas para Ventas por Productos]![FechaInicial]
```

#### Campo: Texto27

```
=[Forms]![Pedir Fechas para Ventas por Productos]![FechaFinal]
```

Campo: Texto36

=[Total Ventas]/[TBVendidas]

Campo: Texto40

=Sum([Unidades Despachadas])

Campo: Texto45

=[Total Despachado]/[TBDespachado]

Campo: Texto46

=Sum([Unidades Insatisfechas])

Campo: Texto50

=[Total Insatisfecho]/[TBInsatisfecho]

Campo: TUVendidas

=Sum([Unidades Vendidas])

Código

```

1  Attribute VB_Name = "Report_Productos Pedidos y Despachados entre
   Fechas"
2  Attribute VB_Creatable = True
3  Attribute VB_PredeclaredId = True
4  Attribute VB_Exposed = False
5  Option Compare Database
6  Option Explicit
7
8  Private Sub Report_Open(Cancel As Integer)
9      If Forms![Pedir Fechas para Ventas por Productos]!EsGeneral Then
10         General.Visible = True
11         Desde.Visible = False
12         Hasta.Visible = False
13         CuadroDesdeHasta.Visible = False
14     Else
15         General.Visible = False
16         Desde.Visible = True
17         Hasta.Visible = True
18         CuadroDesdeHasta.Visible = True
19     End If
20
21 End Sub
22

```

**50. Productos Pedidos y Despachados entre Fechas D**

Origen del registro: Productos Pedidos y Despachados entre Fechas



Nivel de grupo 0: DepartamentoNivel de grupo 1: Total VentasCampo: TBDespachado

=Sum([Total Despachado])

Campo: TBInsatisfecho

=Sum([Total Insatisfecho])

Campo: TBVendidas

=Sum([Total Ventas])

Campo: TDDespacho

=Sum([Total Despachado])

Campo: TDInsatisfecho

=Sum([Total Insatisfecho])

Campo: TDVentas

=Sum([Total Ventas])

Campo: Texto25

=[Forms]![Pedir Fechas para Ventas por Productos]![FechaInicial]

Campo: Texto27

=[Forms]![Pedir Fechas para Ventas por Productos]![FechaFinal]

Campo: Texto36

=[Total Ventas]/[TBVendidas]

Campo: Texto40

=Sum([Unidades Despachadas])

Campo: Texto45

=[Total Despachado]/[TBDespachado]

Campo: Texto46

=Sum([Unidades Insatisfechas])

Campo: Texto50

=[Total Insatisfecho]/[TBInsatisfecho]

Campo: Texto52

=Sum([Unidades Vendidas])

Campo: Texto57

=Sum([Unidades Despachadas])

Campo: Texto60

=Sum([Unidades Insatisfechas])

Campo: Texto63

=[TDVentas]/[TBVendidas]

Campo: Texto65

=[TDDespacho]/[TBDespachado]

Campo: Texto66

=[TDInsatisfecho]/[TBInsatisfecho]

Campo: TUVendidas

=Sum([Unidades Vendidas])

Código

```
1  Attribute VB_Name = "Report_Productos Pedidos y Despachados entre
   Fechas
2  Attribute VB_Creatable = True
3  Attribute VB_PredeclaredId = True
4  Attribute VB_Exposed = False
5  Option Compare Database
6  Option Explicit
7
8  Private Sub Report_Open(Cancel As Integer)
9      If Forms![Pedir Fechas para Ventas por Productos]!EsGeneral Then
10         General.Visible = True
11         Desde.Visible = False
12         Hasta.Visible = False
13         CuadroDesdeHasta.Visible = False
14     Else
15         General.Visible = False
16         Desde.Visible = True
17         Hasta.Visible = True
18         CuadroDesdeHasta.Visible = True
19     End If
20
21 End Sub
```

## 51. Productos Terminados

Origen del registro: Productos Terminados

Campo: Texto21

= [Fórmulas subinforme].[Report]![Departamento]

Campo: Texto23

= [Fórmulas subinforme].[Report]![Unidades de Presentación]

## 52. Informe: Productos Terminados subinforme

Origen del registro:

```
SELECT DISTINCTROW [Productos Terminados].[Código del Producto],
  [Productos Terminados].[N° de Fórmula], [Productos
Terminados].Presentación, [Productos Terminados].[Cantidad en
Presentación], [Productos Terminados].Destino, Fórmulas.Producto,
Fórmulas.Vehículo, Fórmulas.[Unidades de Presentación],
Fórmulas.Concentración, Fórmulas.[Unidades de Concentración],
Fórmulas.Departamento, Fórmulas.[Tamaño del Lote]
FROM Fórmulas INNER JOIN [Productos Terminados] ON Fórmulas.[N° de
Fórmula] = [Productos Terminados].[N° de Fórmula];
```

## 53. Productos Terminados y Escalas de Precios subinforme

Origen del registro:

```
SELECT DISTINCTROW [Productos Terminados y Escalas de Precios].[Código
del Producto], [Productos Terminados y Escalas de Precios].[Escala],
[Escalas de Precios].[Escala], [Escalas de Precios].[Descripción],
[Escalas de Precios].[Límite Inferior], [Escalas de Precios].[Límite
Superior], [Escalas de Precios].[% Descuento]
FROM ([Escalas de Precios] INNER JOIN [Productos Terminados y Escalas de
Precios] ON [Escalas de Precios].[Escala] =[Productos Terminados y
Escalas de Precios].[Escala]);
```

## 54. Proveedores

Origen del registro: Proveedores Consulta

## 55. Proveedores subinforme de enlace

Origen del registro:

```
SELECT DISTINCTROW [Proveedores].[Código del Proveedor],
    [Proveedores].[Nombre o Razón Social]
FROM [Proveedores];
```

## 56. Referencias Bancarias de Clientes subinforme

### Origen del registro:

```
SELECT DISTINCTROW [Referencias Bancarias de Clientes].[Cliente],
    [Referencias Bancarias de Clientes].[Nombre del Banco], [Referencias
    Bancarias de Clientes].[N° de Cuenta], [Referencias Bancarias de
    Clientes].[Dirección y Teléfonos], [Referencias Bancarias de
    Clientes].[Montos de Créditos]
FROM [Referencias Bancarias de Clientes];
```

## 57. Referencias Comerciales de Clientes subinforme

### Origen del registro:

```
SELECT DISTINCTROW [Referencias Comerciales de Clientes].[Cliente],
    [Referencias Comerciales de Clientes].[Nombre del Comercio],
    [Referencias Comerciales de Clientes].[Dirección y Teléfonos],
    [Referencias Comerciales de Clientes].[Montos de Créditos]
FROM [Referencias Comerciales de Clientes];
```

## 58. Resumen de Órdenes de Producción

Origen del registro: Resumen de Órdenes de Producción

Nivel de grupo 0: Departamento

Nivel de grupo 1: Nombre

Nivel de grupo 2: N° de Orden

Campo: Texto36

=Sum([Cantidad Ordenada])

Campo: Texto39

=Sum([Cantidad Ordenada])

Campo: Texto42

=Sum([Cantidad Ordenada])

Campo: Texto48

=Sum([Cantidad Producida])

Campo: Texto50

Campo: Texto52

=Sum([Cantidad Producida])

=Sum([Cantidad Producida])

## 59. Revisiones de Fórmulas subinforme

Origen del registro:

```
SELECT DISTINCTROW [Revisiones de Fórmulas].[N° de Fórmula], [Revisiones
de Fórmulas].[Fecha de Revisión], [Revisiones de Fórmulas].[Revisado
Por], [Revisiones de Fórmulas].[Observaciones],
[Personas].[Apellidos], [Personas].[Nombres]
FROM ([Personas] INNER JOIN [Revisiones de Fórmulas] ON
[Personas].[Cédula de Identidad o Clave] =[Revisiones de
Fórmulas].[Revisado Por]);
```

## 60. Subinforme Lista Sencilla de Productos Terminados

Origen del registro:

```
SELECT DISTINCTROW [Lista Sencilla de Productos Terminados].[Código del
Producto], [Lista Sencilla de Productos Terminados].[Nombre]
FROM [Lista Sencilla de Productos Terminados];
```

## 61. Subinforme Notas en Cotizaciones

Origen del registro:

```
SELECT DISTINCTROW [Notas en Cotizaciones].[N° de la Cotización], [Notas
en Cotizaciones].[Nota N°], [Notas en Cotizaciones].[Contenido]
FROM [Notas en Cotizaciones] INNER JOIN [Notas en Cotizaciones] ON
[Notas en Cotizaciones].[Nota N°] = [Notas en Cotizaciones].[Nota N°];
```

## 62. Subinforme Productos Cotizados a Clientes

Origen del registro:

```
SELECT DISTINCTROW [Productos Cotizados a Clientes].[N° de la
Cotización], [Productos Cotizados a Clientes].[Código del Producto],
[Listas de Precios Vigentes en Fecha Buscada].Nombre, [Productos
Cotizados a Clientes].[Cantidad Cotizada], [Listas de Precios Vigentes
en Fecha Buscada].[Precio Unitario de Venta], [Listas de Precios
Vigentes en Fecha Buscada].[% Descuento], [Listas de Precios Vigentes
en Fecha Buscada].[Precio con Descuento]
FROM [Productos Cotizados a Clientes] INNER JOIN [Listas de Precios
Vigentes en Fecha Buscada] ON [Productos Cotizados a
```

```

Clientes].[Código del Producto] = [Lista de Precios Vigente en Fecha
Buscada].[Código del Producto]
WHERE ((([Productos Cotizados a Clientes].[Cantidad Cotizada]) Between
[Límite Inferior] And [Límite Superior]));

```

Campo: Texto13

```
= [Cantidad Cotizada] * [Precio con Descuento]
```

Campo: Texto16

```
= Sum([Cantidad Cotizada] * [Precio con Descuento])
```

### 63. Subinforme Productos en Modelos P

Origen del registro:

```

SELECT DISTINCTROW [Productos en Modelos P].[N° de Modelo], [Productos en
Modelos P].[Período], [Productos en Modelos P].[Demanda], [Productos
en Modelos P].[Costo de Producción], [Productos en Modelos P].[Costo
de Mantener], [Productos en Modelos P].[Costo de Escasez ],
[Productos en Modelos P].[A Producir], [Productos en Modelos P].[A
Mantener], [Productos en Modelos P].[A Demorar]
FROM [Productos en Modelos P];

```

### 64. Subinforme Tipos de Clientes

Origen del registro:

```

SELECT DISTINCTROW [Tipos de Clientes].[Tipo de Cliente], [Tipos de
Clientes].[Descuento]
FROM [Tipos de Clientes];

```

### 65. Subtotal Kg y Lt Insumos Ord de Prod subinforme

Origen del registro:

```

SELECT DISTINCTROW [Subtotal Kg y Lt Insumos en Ordenes de Prod].[N° de
Orden], [Subtotal Kg y Lt Insumos en Ordenes de Prod].Conversión,
[Subtotal Kg y Lt Insumos en Ordenes de Prod].A
FROM [Subtotal Kg y Lt Insumos en Ordenes de Prod];

```

### 66. Sucursales de Clientes subinforme

Origen del registro:

```

SELECT DISTINCTROW [Sucursales de Clientes].Cliente, [Sucursales de
Clientes].Sucursal
FROM [Sucursales de Clientes];

```

## 67. Sucursales de Proveedores subinforme

### Origen del registro:

```
SELECT DISTINCTROW [Sucursales de Proveedores].[Proveedor], [Sucursales de Proveedores].[Sucursal]
FROM [Sucursales de Proveedores];
```

## 68. Teléfonos de Clientes subinforme

### Origen del registro:

```
SELECT DISTINCTROW [Teléfonos de Clientes].[Cliente], [Teléfonos de Clientes].[Teléfono], [Teléfonos de Clientes].[Otro Código o Celular], [Teléfonos de Clientes].[¿Es Fax?]
FROM [Teléfonos de Clientes];
```

## 69. Teléfonos de Proveedores subinforme

### Origen del registro:

```
SELECT DISTINCTROW [Teléfonos de Proveedores].[Proveedor], [Teléfonos de Proveedores].[Teléfono], [Teléfonos de Proveedores].[Otro Código o Celular], [Teléfonos de Proveedores].[¿Es Fax?]
FROM [Teléfonos de Proveedores];
```

## 70. Vendedores y sus Clientes

### Origen del registro:

```
SELECT DISTINCTROW [Pedidos de Clientes].Vendedor, [Pedidos de Clientes].[Código del Cliente], Trim([Apellidos]) & ", " & Trim([Nombres]) AS Nombre, [Lista Sencilla de Clientes].[Nombre o Razón Social], [Lista Sencilla de Clientes].[Tipo de Cliente], [Lista Sencilla de Clientes].Grupo, [Lista Sencilla de Clientes].Sector, Personas.[Cédula de Identidad o Clave]
FROM Personas INNER JOIN ([Lista Sencilla de Clientes] INNER JOIN [Pedidos de Clientes] ON [Lista Sencilla de Clientes].[Código del Cliente] = [Pedidos de Clientes].[Código del Cliente]) ON Personas.[Cédula de Identidad o Clave] = [Pedidos de Clientes].Vendedor;
```

Nivel de grupo 0: Nombre

Nivel de grupo 1: Sector

Nivel de grupo 2: Grupo

Nivel de grupo 3: Tipo de ClienteNivel de grupo 4: Nombre o Razón SocialCampo: Texto35

```
=Count([Código del Cliente])
```

**71. Ventas Anuales D**Origen del registro: Ventas Anuales DNivel de grupo 0: AñoMarco de objeto independiente: Gráfico1

## Clase OLE: Gráfico de Microsoft Graph 97

## Origen de la fila:

```
SELECT [Ventas Anuales D].Año, Sum([Ventas Anuales D].[Monto del
      Producto]) AS [SumaDeMonto del Producto]
FROM [Ventas Anuales D]
GROUP BY [Ventas Anuales D].Año;
```

Campo: SumaT

```
=Sum([Monto del Producto])
```

Campo: TítuloP

```
="DEPARTAMENTO: " & [Forms]![Pedir Año para resumen de
      Ventas]![Departamento]
```

Código

```
1  Attribute VB_Name = "Report_Ventas Anuales D"
2  Attribute VB_Creatable = True
3  Attribute VB_PredeclaredId = True
4  Attribute VB_Exposed = False
5  Option Compare Database
6  Option Explicit
7
8  Private Sub EncabezadoDelInforme_Format(Cancel As Integer,
      FormatCount As Integer)
9      Select Case Forms![Pedir Año para resumen de Ventas]!MarcoSalida
10         Case 1
11             EncabezadoDePágina.Visible = True
12             Detalle.Visible = True
13             LíneaT.Visible = True
```



```

14     SumaT.Visible = True
15     EtiquetaT.Visible = True
16     Gráfico1.Visible = False
17     Case 2
18     EncabezadoDePágina.Visible = False
19     Detalle.Visible = False
20     LíneaT.Visible = False
21     SumaT.Visible = False
22     EtiquetaT.Visible = False
23     Gráfico1.Visible = True
24     Case 3
25     EncabezadoDePágina.Visible = True
26     Detalle.Visible = True
27     LíneaT.Visible = True
28     SumaT.Visible = True
29     EtiquetaT.Visible = True
30     Gráfico1.Visible = True
31     End Select
32     End Sub

```

## 72. Ventas Anuales G

Origen del registro: Ventas Anuales G

Nivel de grupo 0: Año

Marco de objeto independiente: Gráfico1

Clase OLE: Gráfico de Microsoft Graph 97

Origen de la fila:

```

SELECT [Año],Sum([Monto del Producto]) AS [SumaDeMonto del Producto]
FROM [Ventas Anuales G]
GROUP BY [Año];

```

Campo: SumaT

```

=Sum([Monto del Producto])

```

Código

```

1     Attribute VB_Name = "Report_Ventas Anuales G"
2     Attribute VB_Creatable = True
3     Attribute VB_PredeclaredId = True
4     Attribute VB_Exposed = False
5     Option Compare Database
6     Option Explicit
7
8     Private Sub EncabezadoDelInforme_Format(Cancel As Integer,
        FormatCount As Integer)

```

```

9      Select Case Forms![Pedir Año para resumen de Ventas]!MarcoSalida
10     Case 1
11         EncabezadoDePágina.Visible = True
12         Detalle.Visible = True
13         LíneaT.Visible = True
14         SumaT.Visible = True
15         EtiquetaT.Visible = True
16         Gráfico1.Visible = False
17     Case 2
18         EncabezadoDePágina.Visible = False
19         Detalle.Visible = False
20         LíneaT.Visible = False
21         SumaT.Visible = False
22         EtiquetaT.Visible = False
23         Gráfico1.Visible = True
24     Case 3
25         EncabezadoDePágina.Visible = True
26         Detalle.Visible = True
27         LíneaT.Visible = True
28         SumaT.Visible = True
29         EtiquetaT.Visible = True
30         Gráfico1.Visible = True
31     End Select
32 End Sub

```

### **73. Ventas Anuales P**

Origen del registro: Ventas Anuales P

Nivel de grupo 0: Año

Marco de objeto independiente: Gráfico1

Clase OLE: Gráfico de Microsoft Graph 97

Origen de la fila:

```

SELECT [Ventas Anuales P].Año, Sum([Ventas Anuales P].[Monto del
    Producto]) AS [SumaDeMonto del Producto]
FROM [Ventas Anuales P]
GROUP BY [Ventas Anuales P].Año;

```

Marco de objeto independiente: OLEIndependiente22

Clase OLE: Gráfico de Microsoft Graph 97

Origen de la fila:

```

SELECT [Ventas Anuales P].Año, Sum([Ventas Anuales P].[Cantidad Pedida])
    AS [SumaDeCantidad Pedida]

```

```
FROM [Ventas Anuales P]
GROUP BY [Ventas Anuales P].Año;
```

**Campo: SumaT**

```
=Sum([Monto del Producto])
```

**Campo: Texto20**

```
=Sum([Cantidad Pedida])
```

**Campo: Texto23**

```
=DLookup("[Nombre]", "[Lista Sencilla de Productos Terminados]", "[Código
del Producto]='" & [Forms]![Pedir Año para resumen de
Ventas]![Producto] & "'")
```

**Campo: TítuloP**

```
="PRODUCTO: " & [Forms]![Pedir Año para resumen de Ventas]![Producto]
```

**Código**

```
1 Attribute VB_Name = "Report_Ventas Anuales P"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8 Private Sub EncabezadoDelInforme_Format(Cancel As Integer,
FormatCount As Integer)
9 Select Case Forms![Pedir Año para resumen de Ventas]!MarcoSalida
10 Case 1
11 EncabezadoDePágina.Visible = True
12 Detalle.Visible = True
13 LíneaT.Visible = True
14 SumaT.Visible = True
15 EtiquetaT.Visible = True
16 Gráficol.Visible = False
17 Case 2
18 EncabezadoDePágina.Visible = False
19 Detalle.Visible = False
20 LíneaT.Visible = False
21 SumaT.Visible = False
22 EtiquetaT.Visible = False
23 Gráficol.Visible = True
24 Case 3
25 EncabezadoDePágina.Visible = True
26 Detalle.Visible = True
27 LíneaT.Visible = True
28 SumaT.Visible = True
29 EtiquetaT.Visible = True
30 Gráficol.Visible = True
31 End Select
32 End Sub
```

## 74. Ventas Mensuales D

Origen del registro: Ventas Mensuales D

Nivel de grupo 0: Mes

Marco de objeto independiente: Gráfico1

Clase OLE: Gráfico de Microsoft Graph 97

Origen de la fila:

```
SELECT [Ventas Mensuales D].Mes, Sum([Ventas Mensuales D].[Monto del
      Producto]) AS [SumaDeMonto del Producto]
FROM [Ventas Mensuales D]
GROUP BY [Ventas Mensuales D].Mes;
```

Campo: SumaT

```
=Sum([Monto del Producto])
```

Campo: Texto20

```
=[Forms]![Pedir Año para resumen de Ventas]![Año]
```

Campo: TítuloP

```
="DEPARTAMENTO: " & [Forms]![Pedir Año para resumen de
      Ventas]![Departamento]
```

Código

```
1  Attribute VB_Name = "Report_Ventas Mensuales D"
2  Attribute VB_Creatable = True
3  Attribute VB_PredeclaredId = True
4  Attribute VB_Exposed = False
5  Option Compare Database
6  Option Explicit
7
8  Private Sub EncabezadoDelInforme_Format(Cancel As Integer,
      FormatCount As Integer)
9      Select Case Forms![Pedir Año para resumen de Ventas]!MarcoSalida
10         Case 1
11             EncabezadoDePágina.Visible = True
12             Detalle.Visible = True
13             LíneaT.Visible = True
14             SumaT.Visible = True
15             EtiquetaT.Visible = True
16             Gráfico1.Visible = False
17         Case 2
18             EncabezadoDePágina.Visible = False
19             Detalle.Visible = False
```

```

20     LíneaT.Visible = False
21     SumaT.Visible = False
22     EtiquetaT.Visible = False
23     Gráfico1.Visible = True
24     Case 3
25     EncabezadoDePágina.Visible = True
26     Detalle.Visible = True
27     LíneaT.Visible = True
28     SumaT.Visible = True
29     EtiquetaT.Visible = True
30     Gráfico1.Visible = True
31     End Select
32     End Sub

```

## 75. Ventas Mensuales G

Origen del registro: Ventas Mensuales G

Nivel de grupo 0: Mes

Marco de objeto independiente: Gráfico1

Clase OLE: Gráfico de Microsoft Graph 97

Origen de la fila:

```

SELECT [Ventas Mensuales G].Mes, Sum([Ventas Mensuales G].[Monto del
      Producto]) AS [SumaDeMonto del Producto]
FROM [Ventas Mensuales G]
GROUP BY [Ventas Mensuales G].Mes;

```

Campo: SumaT

```
=Sum([Monto del Producto])
```

Campo: Texto20

```
=[Forms]![Pedir Año para resumen de Ventas]![Año]
```

Código

```

1     Attribute VB_Name = "Report_Ventas Mensuales G"
2     Attribute VB_Creatable = True
3     Attribute VB_PredeclaredId = True
4     Attribute VB_Exposed = False
5     Option Compare Database
6     Option Explicit
7
8     Private Sub EncabezadoDelInforme_Format(Cancel As Integer,
      FormatCount As Integer)
9         Select Case Forms![Pedir Año para resumen de Ventas]!MarcoSalida
10            Case 1

```

```
11     EncabezadoDePágina.Visible = True
12     Detalle.Visible = True
13     LíneaT.Visible = True
14     SumaT.Visible = True
15     EtiquetaT.Visible = True
16     Gráfico1.Visible = False
17     Case 2
18     EncabezadoDePágina.Visible = False
19     Detalle.Visible = False
20     LíneaT.Visible = False
21     SumaT.Visible = False
22     EtiquetaT.Visible = False
23     Gráfico1.Visible = True
24     Case 3
25     EncabezadoDePágina.Visible = True
26     Detalle.Visible = True
27     LíneaT.Visible = True
28     SumaT.Visible = True
29     EtiquetaT.Visible = True
30     Gráfico1.Visible = True
31     End Select
32     End Sub
```

## **76. Ventas Mensuales P**

Origen del registro: Ventas Mensuales P

Nivel de grupo 0: Mes

Marco de objeto independiente: Gráfico1

Clase OLE: Gráfico de Microsoft Graph 97

Origen de la fila:

```
SELECT [Ventas Mensuales P].Mes, Sum([Ventas Mensuales P].[Monto del
      Producto]) AS [SumaDeMonto del Producto]
FROM [Ventas Mensuales P]
GROUP BY [Ventas Mensuales P].Mes;
```

Marco de objeto independiente: OLEIndependiente22

Clase OLE: Gráfico de Microsoft Graph 97

Origen de la fila:

```
SELECT [Ventas Mensuales P].Mes, Sum([Ventas Mensuales P].[Cantidad
      Pedida]) AS [SumaDeCantidad Pedida]
FROM [Ventas Mensuales P]
```

GROUP BY [Ventas Mensuales P].Mes;

**Campo: SumaT**

=Sum([Monto del Producto])

**Campo: Texto20**

=Sum([Cantidad Pedida])

**Campo: Texto23**

=DLookup("[Nombre]","[Lista Sencilla de Productos Terminados]","[Código del Producto]='" & [Forms]![Pedir Año para resumen de Ventas]![Producto] & "'")

**Campo: Texto24**

=[Forms]![Pedir Año para resumen de Ventas]![Año]

**Campo: TítuloP**

="PRODUCTO: " & [Forms]![Pedir Año para resumen de Ventas]![Producto]

**Código**

```

1  Attribute VB_Name = "Report_Ventas Mensuales P"
2  Attribute VB_Creatable = True
3  Attribute VB_PredeclaredId = True
4  Attribute VB_Exposed = False
5  Option Compare Database
6  Option Explicit
7
8  Private Sub EncabezadoDelInforme_Format(Cancel As Integer,
9  FormatCount As Integer)
10     Select Case Forms![Pedir Año para resumen de Ventas]!MarcoSalida
11     Case 1
12         EncabezadoDePágina.Visible = True
13         Detalle.Visible = True
14         LíneaT.Visible = True
15         SumaT.Visible = True
16         EtiquetaT.Visible = True
17         Gráficol.Visible = False
18     Case 2
19         EncabezadoDePágina.Visible = False
20         Detalle.Visible = False
21         LíneaT.Visible = False
22         SumaT.Visible = False
23         EtiquetaT.Visible = False
24         Gráficol.Visible = True
25     Case 3
26         EncabezadoDePágina.Visible = True
27         Detalle.Visible = True
28         LíneaT.Visible = True
29         SumaT.Visible = True
30         EtiquetaT.Visible = True
31         Gráficol.Visible = True

```

```

31 End Select
32 End Sub

```

## 77. Ventas Semestrales D

Origen del registro: Ventas Semestrales D

Nivel de grupo 0: Semestre

Marco de objeto independiente: Gráfico1

Clase OLE: Gráfico de Microsoft Graph 97

Origen de la fila:

```

SELECT [Ventas Semestrales D].Semestre, Sum([Ventas Semestrales D].[Monto
del Producto]) AS [SumaDeMonto del Producto]
FROM [Ventas Semestrales D]
GROUP BY [Ventas Semestrales D].Semestre;

```

Campo: SumaT

```
=Sum([Monto del Producto])
```

Campo: Texto20

```
=[Forms]![Pedir Año para resumen de Ventas]![Año]
```

Campo: TítuloP

```
= "DEPARTAMENTO: " & [Forms]![Pedir Año para resumen de
Ventas]![Departamento]
```

Código

```

1 Attribute VB_Name = "Report_Ventas Semestrales D"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8 Private Sub EncabezadoDelInforme_Format(Cancel As Integer,
FormatCount As Integer)
9 Select Case Forms![Pedir Año para resumen de Ventas]!MarcoSalida
10 Case 1
11 EncabezadoDePágina.Visible = True
12 Detalle.Visible = True
13 LíneaT.Visible = True
14 SumaT.Visible = True
15 EtiquetaT.Visible = True
16 Gráfico1.Visible = False

```



```

17 Case 2
18 EncabezadoDePágina.Visible = False
19 Detalle.Visible = False
20 LíneaT.Visible = False
21 SumaT.Visible = False
22 EtiquetaT.Visible = False
23 Gráfico1.Visible = True
24 Case 3
25 EncabezadoDePágina.Visible = True
26 Detalle.Visible = True
27 LíneaT.Visible = True
28 SumaT.Visible = True
29 EtiquetaT.Visible = True
30 Gráfico1.Visible = True
31 End Select
32 End Sub

```

## 78. Ventas Semestrales G

Origen del registro: Ventas Semestrales G

Nivel de grupo 0: Semestre

Marco de objeto independiente: Gráfico1

Clase OLE: Gráfico de Microsoft Graph 97

Origen de la fila:

```

SELECT [Ventas Semestrales G].Semestre, Sum([Ventas Semestrales G].[Monto
del Producto]) AS [SumaDeMonto del Producto]
FROM [Ventas Semestrales G]
GROUP BY [Ventas Semestrales G].Semestre;

```

Campo: SumaT

```
=Sum([Monto del Producto])
```

Campo: Texto20

```
=[Forms]![Pedir Año para resumen de Ventas]![Año]
```

Código

```

1 Attribute VB_Name = "Report_Ventas Semestrales G"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7

```

```

8 Private Sub EncabezadoDelInforme_Format(Cancel As Integer,
FormatCount As Integer)
9 Select Case Forms![Pedir Año para resumen de Ventas]!MarcoSalida
10 Case 1
11 EncabezadoDePágina.Visible = True
12 Detalle.Visible = True
13 LíneaT.Visible = True
14 SumaT.Visible = True
15 EtiquetaT.Visible = True
16 Gráfico1.Visible = False
17 Case 2
18 EncabezadoDePágina.Visible = False
19 Detalle.Visible = False
20 LíneaT.Visible = False
21 SumaT.Visible = False
22 EtiquetaT.Visible = False
23 Gráfico1.Visible = True
24 Case 3
25 EncabezadoDePágina.Visible = True
26 Detalle.Visible = True
27 LíneaT.Visible = True
28 SumaT.Visible = True
29 EtiquetaT.Visible = True
30 Gráfico1.Visible = True
31 End Select
32 End Sub

```

## 79. Ventas Semestrales P

Origen del registro: Ventas Semestrales P

Nivel de grupo 0: Semestre

Marco de objeto independiente: Gráfico1

Clase OLE: Gráfico de Microsoft Graph 97

Origen de la fila:

```

SELECT [Ventas Semestrales P].Semestre, Sum([Ventas Semestrales P].[Monto
del Producto]) AS [SumaDeMonto del Producto]
FROM [Ventas Semestrales P]
GROUP BY [Ventas Semestrales P].Semestre;

```

Marco de objeto independiente: OLEIndependiente22

Clase OLE: Gráfico de Microsoft Graph 97

Origen de la fila:

```
SELECT [Ventas Semestrales P].Semestre, Sum([Ventas Semestrales
      P].[Cantidad Pedida]) AS [SumaDeCantidad Pedida]
FROM [Ventas Semestrales P]
GROUP BY [Ventas Semestrales P].Semestre;
```

Campo: SumaT

```
=Sum([Monto del Producto])
```

Campo: Texto20

```
=Sum([Cantidad Pedida])
```

Campo: Texto23

```
=DLookup("[Nombre]", "[Lista Sencilla de Productos Terminados]", "[Código
del Producto]='" & [Forms]![Pedir Año para resumen de
Ventas]![Producto] & "'")
```

Campo: Texto24

```
=[Forms]![Pedir Año para resumen de Ventas]![Año]
```

Campo: TítuloP

```
="PRODUCTO: " & [Forms]![Pedir Año para resumen de Ventas]![Producto]
```

Código

```
1  Attribute VB_Name = "Report_Ventas Semestrales P"
2  Attribute VB_Creatable = True
3  Attribute VB_PredeclaredId = True
4  Attribute VB_Exposed = False
5  Option Compare Database
6  Option Explicit
7
8  Private Sub EncabezadoDelInforme_Format(Cancel As Integer,
FormatCount As Integer)
9      Select Case Forms![Pedir Año para resumen de Ventas]!MarcoSalida
10         Case 1
11             EncabezadoDePágina.Visible = True
12             Detalle.Visible = True
13             LíneaT.Visible = True
14             SumaT.Visible = True
15             EtiquetaT.Visible = True
16             Gráficol.Visible = False
17         Case 2
18             EncabezadoDePágina.Visible = False
19             Detalle.Visible = False
20             LíneaT.Visible = False
21             SumaT.Visible = False
22             EtiquetaT.Visible = False
23             Gráficol.Visible = True
24         Case 3
25             EncabezadoDePágina.Visible = True
26             Detalle.Visible = True
27             LíneaT.Visible = True
```

```

28     SumaT.Visible = True
29     EtiquetaT.Visible = True
30     Gráfico1.Visible = True
31 End Select
32 End Sub

```

## 80. Ventas Trimestrales D

Origen del registro: Ventas Trimestrales D

Nivel de grupo 0: Trimestre

Marco de objeto independiente: Gráfico1

Clase OLE: Gráfico de Microsoft Graph 97

Origen de la fila:

```

SELECT [Ventas Trimestrales D].Trimestre, Sum([Ventas Trimestrales
      D].[Monto del Producto]) AS [SumaDeMonto del Producto]
FROM [Ventas Trimestrales D]
GROUP BY [Ventas Trimestrales D].Trimestre;

```

Campo: SumaT

```
=Sum([Monto del Producto])
```

Campo: Texto20

```
=[Forms]![Pedir Año para resumen de Ventas]![Año]
```

Campo: TítuloP

```
= "DEPARTAMENTO: " & [Forms]![Pedir Año para resumen de
      Ventas]![Departamento]
```

Código

```

1  Attribute VB_Name = "Report_Ventas Trimestrales D"
2  Attribute VB_Creatable = True
3  Attribute VB_PredeclaredId = True
4  Attribute VB_Exposed = False
5  Option Compare Database
6  Option Explicit
7
8  Private Sub EncabezadoDelInforme_Format(Cancel As Integer,
      FormatCount As Integer)
9      Select Case Forms![Pedir Año para resumen de Ventas]!MarcoSalida
10         Case 1
11             EncabezadoDePágina.Visible = True
12             Detalle.Visible = True

```

```

13     LíneaT.Visible = True
14     SumaT.Visible = True
15     EtiquetaT.Visible = True
16     Gráfico1.Visible = False
17     Case 2
18         EncabezadoDePágina.Visible = False
19         Detalle.Visible = False
20         LíneaT.Visible = False
21         SumaT.Visible = False
22         EtiquetaT.Visible = False
23         Gráfico1.Visible = True
24     Case 3
25         EncabezadoDePágina.Visible = True
26         Detalle.Visible = True
27         LíneaT.Visible = True
28         SumaT.Visible = True
29         EtiquetaT.Visible = True
30         Gráfico1.Visible = True
31     End Select
32 End Sub

```

## 81. Ventas Trimestrales G

Origen del registro: Ventas Trimestrales G

Nivel de grupo 0: Trimestre

Marco de objeto independiente: Gráfico1

Clase OLE: Gráfico de Microsoft Graph 97

Origen de la fila:

```

SELECT [Ventas Trimestrales G].Trimestre, Sum([Ventas Trimestrales
      G].[Monto del Producto]) AS [SumaDeMonto del Producto]
FROM [Ventas Trimestrales G]
GROUP BY [Ventas Trimestrales G].Trimestre;

```

Campo: SumaT

=Sum([Monto del Producto])

Campo: Texto20

=[Forms]![Pedir Año para resumen de Ventas]![Año]

Código

```

1  Attribute VB_Name = "Report_Ventas Trimestrales G"
2  Attribute VB_Creatable = True
3  Attribute VB_PredeclaredId = True
4  Attribute VB_Exposed = False

```

```

5 Option Compare Database
6 Option Explicit
7
8 Private Sub EncabezadoDelInforme_Format(Cancel As Integer,
FormatCount As Integer)
9 Select Case Forms![Pedir Año para resumen de Ventas]!MarcoSalida
10 Case 1
11 EncabezadoDePágina.Visible = True
12 Detalle.Visible = True
13 LíneaT.Visible = True
14 SumaT.Visible = True
15 EtiquetaT.Visible = True
16 Gráfico1.Visible = False
17 Case 2
18 EncabezadoDePágina.Visible = False
19 Detalle.Visible = False
20 LíneaT.Visible = False
21 SumaT.Visible = False
22 EtiquetaT.Visible = False
23 Gráfico1.Visible = True
24 Case 3
25 EncabezadoDePágina.Visible = True
26 Detalle.Visible = True
27 LíneaT.Visible = True
28 SumaT.Visible = True
29 EtiquetaT.Visible = True
30 Gráfico1.Visible = True
31 End Select
32 End Sub

```

## 82. Ventas Trimestrales P

Origen del registro: Ventas Trimestrales P

Nivel de grupo 0: Trimestre

Marco de objeto independiente: Gráfico1

Clase OLE: Gráfico de Microsoft Graph 97

Origen de la fila:

```

SELECT [Ventas Trimestrales P].Trimestre, Sum([Ventas Trimestrales
P].[Monto del Producto]) AS [SumaDeMonto del Producto]
FROM [Ventas Trimestrales P]
GROUP BY [Ventas Trimestrales P].Trimestre;

```

Marco de objeto independiente: OLEIndependiente22

Clase OLE: Gráfico de Microsoft Graph 97

**Origen de la fila:**

```
SELECT [Ventas Trimestrales P].Trimestre, Sum([Ventas Trimestrales
P].[Cantidad Pedida]) AS [SumaDeCantidad Pedida]
FROM [Ventas Trimestrales P]
GROUP BY [Ventas Trimestrales P].Trimestre;
```

**Campo: SumaT**

```
=Sum([Monto del Producto])
```

**Campo: Texto20**

```
=Sum([Cantidad Pedida])
```

**Campo: Texto23**

```
=DLookup("[Nombre]","[Lista Sencilla de Productos Terminados]","[Código
del Producto]='" & [Forms]![Pedir Año para resumen de
Ventas]![Producto] & "'")
```

**Campo: Texto24**

```
=[Forms]![Pedir Año para resumen de Ventas]![Año]
```

**Campo: TítuloP**

```
="PRODUCTO: " & [Forms]![Pedir Año para resumen de Ventas]![Producto]
```

**Código**

```
1 Attribute VB_Name = "Report_Ventas Trimestrales P"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8 Private Sub EncabezadoDelInforme_Format(Cancel As Integer,
FormatCount As Integer)
9 Select Case Forms![Pedir Año para resumen de Ventas]!MarcoSalida
10 Case 1
11 EncabezadoDePágina.Visible = True
12 Detalle.Visible = True
13 LíneaT.Visible = True
14 SumaT.Visible = True
15 EtiquetaT.Visible = True
16 Gráfico1.Visible = False
17 Case 2
18 EncabezadoDePágina.Visible = False
19 Detalle.Visible = False
20 LíneaT.Visible = False
21 SumaT.Visible = False
22 EtiquetaT.Visible = False
23 Gráfico1.Visible = True
24 Case 3
```

```
25     EncabezadoDePágina.Visible = True
26     Detalle.Visible = True
27     LíneaT.Visible = True
28     SumaT.Visible = True
29     EtiquetaT.Visible = True
30     Gráfico1.Visible = True
31     End Select
32     End Sub
```

### 83. Planes de Producción

#### Origen del registro:

```
SELECT DISTINCTROW [Planes de Producción].[N° del Plan], [Productos en
Planes].[Código del Producto], [Productos en Planes].Fecha,
[Productos en Planes].Cantidad, [Planes de Producción].Descripción,
[Planes de Producción].[Fecha de Inicio], [Planes de
Producción].[Fecha de Culminación], [Lista Sencilla de Productos
Terminados].Nombre
FROM [Planes de Producción] INNER JOIN ([Lista Sencilla de Productos
Terminados] INNER JOIN [Productos en Planes] ON [Lista Sencilla de
Productos Terminados].[Código del Producto] = [Productos en
Planes].[Código del Producto]) ON [Planes de Producción].[N° del
Plan] = [Productos en Planes].[N° del Plan];
```

#### Nivel de grupo 0: Planes de Producción.N° del Plan

#### Nivel de grupo 1: Nombre

#### Nivel de grupo 2: Fecha

#### Campo: Texto33

```
=Sum([Cantidad])
```



## **ANEXO 3**

### **Ejemplos de Informes**

#### **SARCAI v. 2.0**

## **ANEXO 4**

### **Macros y Código fuente del Módulo**

#### **General del SARCAI v. 2.0**

## **MACROS**

### **Abrir Existencia Real en Almacén I**

```
InicializarInforme("Existencia Real en Almacén I")  
OpenForm Pedir Dos Fechas
```

### **Abrir Existencia Real en Almacén T**

```
InicializarInforme("Existencia Real en Almacén T")  
OpenForm Pedir Dos Fechas
```

### **Abrir Existencia Total Real en Almacén I**

```
InicializarInforme("Existencia Total Real en Almacén I")  
OpenForm Pedir Dos Fechas
```

### **Abrir Existencia Total Real en Almacén T**

```
InicializarInforme("Existencia Total Real en Almacén T")  
OpenForm Pedir Dos Fechas
```

### **Abrir Explosión de Materiales**

```
InicializarInforme("Explosión de Materiales")  
OpenForm Pedir Número del Plan
```

### **Abrir Lista de Precios Vigente en Fecha Buscada**

```
InicializarInforme(" Lista de Precios Vigente en Fecha Buscada")  
OpenForm Pedir Dos Fechas
```

### **Llamar a Procedimiento**

```
llamar ()
```

## MÓDULO BIBLIOTECA

```
1 Attribute VB_Name = "Biblioteca"
2 Option Compare Database
3 Option Explicit
4
5 Public NomInforme As String
6
7 Public Function TransformaFecha(F As Date) As Date
8     TransformaFecha = CDate(Str(Day(F)) & "/" & Str(Month(F)) & "/" &
9     Str(Year(F)))
10 End Function
11
12 Public Sub PreparaGráficoDeProductosEstrella()
13     Dim dbs As Database, rst As Recordset, strSQL As String, _
14     NReg As Long, qdf As QueryDef
15
16     Set dbs = CurrentDb
17     Set qdf = dbs.CreateQueryDef("")
18     strSQL = "PARAMETERS [Forms]![Pedir Fechas para Ventas por
19     Productos]![FechaInicial] DateTime, [Forms]![Pedir Fechas para
20     Ventas por Productos]![FechaFinal] DateTime;" & _
21     "SELECT [Productos Pedidos y Despachados entre Fechas].Nombre AS
22     Producto, Sum([Productos Pedidos y Despachados entre Fechas].[Total
23     Ventas]) AS [Total Ventas] " & _
24     "FROM [Productos Pedidos y Despachados entre Fechas] " & _
25     "GROUP BY [Productos Pedidos y Despachados entre Fechas].Nombre
26     " & _
27     "ORDER BY Sum([Productos Pedidos y Despachados entre
28     Fechas].[Total Ventas]) DESC;"
29     qdf.SQL = strSQL
30     qdf.Parameters("[Forms]![Pedir Fechas para Ventas por
31     Productos]![FechaInicial]") = [Forms]![Pedir Fechas para Ventas por
32     Productos]![FechaInicial]
33     qdf.Parameters("[Forms]![Pedir Fechas para Ventas por
34     Productos]![FechaFinal]") = [Forms]![Pedir Fechas para Ventas por
35     Productos]![FechaFinal]
36     Set rst = qdf.OpenRecordset(dbOpenSnapshot)
37
38     rst.MoveLast
39     NReg = rst.RecordCount
40     rst.Close
41     Set dbs = Nothing
42
43     DoCmd.SetWarnings False
44
45     'Inicialización de la tabla con los NProd primeros productos
46     strSQL = "DELETE [Gráfico de Productos Estrella T1].*" & _
47     "FROM [Gráfico de Productos Estrella T1];"
48     DoCmd.RunSQL strSQL
49
50     'Inicialización de la tabla con los NReg - NProd últimos productos
51     strSQL = "DELETE [Gráfico de Productos Estrella T2].*" & _
52     "FROM [Gráfico de Productos Estrella T2];"
53     DoCmd.RunSQL strSQL
```

```
44
45 'Inserción de los primeros NProd productos en T1
46 strSQL = "INSERT INTO [Gráfico de Productos Estrella T1] (
47     Producto, [Total Ventas] ) " & _
48     "SELECT TOP " & Str(Forms![Pedir Fechas para Ventas por
49     Productos]!NProd) & " [Gráfico de Productos Estrella].Producto,
50     [Gráfico de Productos Estrella].[Total Ventas] " & _
51     "FROM [Gráfico de Productos Estrella] " & _
52     "ORDER BY [Gráfico de Productos Estrella].[Total Ventas]
53     DoCmd.RunSQL strSQL
54
55 If Forms![Pedir Fechas para Ventas por Productos]!NProd < NReg Then
56 'Inserción de los últimos NReg - NProd productos en T2
57 strSQL = "INSERT INTO [Gráfico de Productos Estrella T2] (
58     Producto, [Total Ventas] ) " & _
59     "SELECT TOP " & Str(NReg - Forms![Pedir Fechas para Ventas por
60     Productos]!NProd) & " [Gráfico de Productos Estrella].Producto,
61     [Gráfico de Productos Estrella].[Total Ventas] " & _
62     "FROM [Gráfico de Productos Estrella] " & _
63     "ORDER BY [Gráfico de Productos Estrella].[Total Ventas]
64     DoCmd.RunSQL strSQL
65 End If
66
67 DoCmd.SetWarnings True
68
69 End Sub
70
71 Public Function NDías(F1 As Date, F2 As Date) As Integer
72     NDías = F2 - F1 + 1
73 End Function
74
75 Public Function NSemanas(F1 As Date, F2 As Date) As Integer
76     NSemanas = NDías(F1, F2) / 7
77 End Function
78
79 Public Function NQuincenas(F1 As Date, F2 As Date) As Integer
80     NQuincenas = NDías(F1, F2) / 15
81 End Function
82
83 Public Function NMeses(F1 As Date, F2 As Date) As Integer
84     NMeses = NDías(F1, F2) / 30
85 End Function
86
87 Public Function NBimestres(F1 As Date, F2 As Date) As Integer
88     NBimestres = NDías(F1, F2) / 60
89 End Function
90
91 Public Function NTrimestres(F1 As Date, F2 As Date) As Integer
92     NTrimestres = NDías(F1, F2) / 90
93 End Function
94
95 Public Function NSemestres(F1 As Date, F2 As Date) As Integer
96     NSemestres = NDías(F1, F2) / 180
97 End Function
98
99 Public Function NAños(F1 As Date, F2 As Date) As Integer
```

```
94   NAños = NDías(F1, F2) / 365
95 End Function
96
97 Public Function Nbiaños(F1 As Date, F2 As Date) As Integer
98   Nbiaños = NDías(F1, F2) / 730
99 End Function
100
101 Public Function InicializarInforme(Nom As String)
102
103   NomInforme = Nom
104
105 End Function
106
107 Public Sub Exportar_Tabla(NomTab As String)
108   DoCmd.TransferSpreadsheet acExport, 5, _
109     NomTab, "c:\datos\excel\" & NomTab, True
110 End Sub
111
112 Sub Compacta()
113   DBEngine.CompactDatabase "C:\Datos\PROULA Tesis\SARCAI versión
114     1.0.mdb", "C:\Datos\PROULA Tesis\Prueba.mdb"
115 End Sub
116
116 Public Sub Cerrar()
117   DoCmd.DoMenuItem acFormBar, acFile, 3, , acMenuVer70
118   DoCmd.DoMenuItem acFormBar, acFile, 4, , acMenuVer70
119 End Sub
120
121 Sub CompactaInventario()
122   DBEngine.CompactDatabase "C:\Datos\Access\Inventario.mdb",
123     "C:\Datos\PROULA Tesis\Prueba.mdb"
124 End Sub
125
125 Sub VerTablas()
126   Dim dbs As Database, tabla As TableDef, i As Integer
127
128   Set dbs = CurrentDb
129
130   With dbs
131     'Debug.Print .TableDefs.Count & _
132     ' "TableDefs en " & .Name
133     i = 0
134     For Each tabla In .TableDefs
135       If Left(tabla.Name, 4) <> "MSys" Then
136         i = i + 1
137         Debug.Print "No.: " & Str(i) & " " & tabla.Name
138       End If
139     Next tabla
140   End With
141
142 End Sub
143
144 Sub InicializarBD()
145   Dim SQLstr As String, dbs As Database
146   Dim tabla As TableDef
147
```

```
148 Set dbs = CurrentDb
149 DoCmd.SetWarnings False
150 With dbs
151   For Each tabla In .TableDefs
152     If (Left(tabla.Name, 4) <> "MSys") And (tabla.Name <> "Elementos
del Panel de control") Then
153       SQLstr = "DELETE DISTINCTROW * FROM [" & tabla.Name
154       DoCmd.RunSQL SQLstr
155     End If
156   Next tabla
157 End With
158 DoCmd.SetWarnings True
159 End Sub
160
161 Function llamar()
162   'InicializarBD
163   'EstablecerParámetro
164   'VerTablas
165 End Function
166
167 Public Sub Llena_PruebaOLE(Nobs As Long)
168   Dim Cont As Long
169   DoCmd.OpenForm "PruebaOLE"
170   For Cont = 1 To Nobs
171     DoCmd.GoToRecord acForm, "PruebaOLE", acGoTo, Cont
172     Forms!PruebaOLE.[Fecha] = Now() + Cont
173     Forms!PruebaOLE.[Demanda] = Int(50001 * Rnd)
174   Next Cont
175 End Sub
176
177 Public Sub PoneClaves(Hasta As Long)
178   Dim Cont As Long
179
180   For Cont = 1 To Hasta
181     DoCmd.GoToRecord acForm, "Insumos Tempo", acGoTo, Cont
182     Forms![Insumos Tempo]![CODIGO] = Str(Cont)
183   Next Cont
184 End Sub
185
186 Public Function PonPuntoyCadena(num As Double) As String
187   Dim PosComa As Integer, LonCad As Integer, CantiCad As String
188   CantiCad = Str(num)
189   LonCad = Len(CantiCad)
190   PosComa = InStr(1, CantiCad, ",")
191   MsgBox PosComa
192   PonPuntoyCadena = Left(CantiCad, PosComa - 1) & "." &
Right(CantiCad, LonCad - PosComa)
193 End Function
194
195 Function EstáCargado(cadNombreFormulario As String) As Boolean
196
197   ' Determina si un formulario está cargado.
198
199   Const conDiseñoFormulario = 0
200   Dim entX As Integer
201
```

```
202  EstáCargado = False
203  For entX = 0 To Forms.Count - 1
204    If Forms(entX).FormName = cadNombreFormulario Then
205      If Forms(entX).CurrentView <> conDiseñoFormulario Then
206        EstáCargado = True
207        Exit Function ' Salir de la función al encontrar el
208        End If
209      End If
210    Next
211
212  End Function
213
214  Sub EstablecerParámetro(prm As Variant)
215
216    Dim dbs As Database, qdf As QueryDef, rst As Recordset
217
218    Set dbs = CurrentDb
219    Set qdf = dbs.QueryDefs![Existencia Total Real an Almacén I]
220    qdf.Parameters![Fecha Buscada] = prm
221    Set rst = qdf.OpenRecordset
222    ' SalidaAbrirRecordset rst
223
224  End Sub
225
226  Sub SalidaAbrirRecordset(rstOutput As Recordset)
227
228    Dim i As Integer
229    With rstOutput
230      For i = 0 To rstOutput.Fields.Count - 1
231        Debug.Print .Fields(i).Name,
232      Next i
233      Debug.Print
234      Do While Not .EOF
235        For i = 0 To rstOutput.Fields.Count - 1
236          Debug.Print .Fields(i),
237        Next i
238        Debug.Print
239        .MoveNext
240      Loop
241    End With
242
243  End Sub
```



## Estadísticas Generales

Base de datos: C:\Datos\PROULA 97\SARCAI2.mdb

AccessVersion:	07.53
AllowBuiltInToolbars:	Verdadero
AllowShortcutMenus:	Verdadero
AllowToolbarChanges:	Verdadero
Build:	3512
Mostrar límite de	1000
Mostrar valores en No	1
Orden de combinación:	General
Show Values in	1
Show Values in	0
StartUpForm:	Panel de control
StartUpShowStatusBar:	Verdadero
Transactions:	Verdadero

AllowBreakIntoCode:	Verdadero
AllowFullMenus:	Verdadero
AllowSpecialKeys:	Verdadero
AppTitle:	SARCAI versión 2.0
Def. actualizable:	Verdadero
Mostrar valores en	1
Mostrar valores en	0
Registros afectados:	0
Show Values in	1
Show Values Limit:	1000
StartUpShowDBWindo	Falso
Tiempo de espera de la	60
Versión :	3.0

Permisos de usuario: admin

Permisos de grupo: Admins, Users