



FACULTAD DE INGENIERÍA UNAM
DIVISIÓN DE EDUCACIÓN CONTINUA

"Tres décadas de orgullosa excelencia" 1971 - 2001

CURSOS INSTITUCIONALES

ACCESS

AVANZADO

Del 01 al 14 de Diciembre de 2004

APUNTES GENERALES

CI-209

Instructor: Ing. Otelio Galicia Cedillo
DELEGACIÓN MIGUEL HIDALGO
DICIEMBRE 2004



Access XP

Facilitador Oteló Galicia Cedillo

Contenido

CONTENIDO	2
OBJETIVO.....	4
INTRODUCCIÓN.....	5
CONSULTAS EN ACCESS XP.....	6
Uso de Consultas para Buscar y Filtrar Datos.....	6
Entender las Consultas.....	6
Cómo seleccionar consultas.....	7
Consultas de referencias cruzadas	9
Consultas parametrizadas.....	10
Consultas de acción.....	12
Consultas SQL.....	14
Creación y Modificación de Consultas.....	15
Como insertar, eliminar y reorganizar los campos.....	16
Definición de una columna calculada	17
Definición de criterios.....	17
Definición de las propiedades de consulta	18
Creación y Aplicación de Filtros	19
DISEÑO Y USO DE FORMULARIOS	21
Uso de Formularios para Ver e Introducir Datos.....	21
Creación de un Formulario Nuevo.....	23
Creación de un formulario básico con un clic.....	24
Uso del Asistente para formularios.....	24
Ver o Introducir Datos con un Formulario.....	26
Cómo Cambiar el Diseño de un Formulario	26
Como agregar y modificar controles en la vista diseño	27
Como colocar controles en un formulario.....	29
Ajustar otras propiedades y opciones de formulario.....	30

PRESENTACIÓN DE DATOS CON INFORMES	31
Creación de un Informe a Partir de Datos de Access XP	31
Como trabajar con la vista diseño	31
Cómo crear un informe básico	33
Como convertir los datos de Access XP a etiquetas	34
Creación de gráficos	35
Modificación de un Informe	37
Cómo agrupar y ordenar registros en un informe.	37
Como agregar y modificar encabezados y pies de página.....	38
Como trabajar con cálculos en un informe	39
Cómo Exportar un Informe a otro Formato de Datos.....	39
Cómo guardar un informe en un formato de imagen instantánea	39
Guardar un informe como página Web.....	40
Exportar un informe a Microsoft Word	40
Analizar un informe con Excel	40
ANEXO B	41
Expresiones en Access XP	41
Operadores.....	42
FUNCIONES.....	49

Objetivo

Proporcionar al participante los conocimientos y habilidades para realizar Formularios e Informes Avanzados e introducirlos en el ambiente de la programación en Access XP utilizando eventos de objetos y realizando cálculos con los diferentes objetos de la base de datos.

Introducción

El presente curso es complemento de Access XP 1 en él veremos la forma de obtener la información que deseamos de una manera rápida y sistemática. Empezaremos con dos sesiones sobre operadores y funciones utilizadas en lenguajes de programación, con esto obtendremos los referentes teóricos necesarios para comprender la forma que Access XP realiza las consultas y la manera de utilizarlos adecuadamente. Posteriormente el participante se iniciara en la creación de consultas, realizara diversas consultas que le permitirán realizar diversas operaciones en el programa de Access XP. En el curso realizaremos consultas de selección, consultas de actualización, consultas para creación de tablas y por ultimo trataremos las consultas parametrizadas y su forma de optimizar la búsqueda... Posteriormente veremos la forma de crear formularios para darle una mejor función y presentación a nuestros objetos, trataremos los autoformularios, formularios con asistentes y formularios en vista diseño, la forma de programar los principales objetos de los formularios. Por ultimo veremos la forma de crear informes para obtener la información de forma escrita de una manera automática. Esperamos que el curso sea tan interesante como lo hemos diseñado, ya que se ha tomado en cuenta los aspectos andragógicos necesarios para lograr la significatividad que los participantes esperan. El curso esta lleno de ejercicios dirigidos a su ambiente de trabajo, en todo momento trataremos que el curso sea agradable y ameno creando confianza para aclarar cualquier duda que surja.

Consultas en Access XP

Uso de Consultas para Buscar y Filtrar Datos

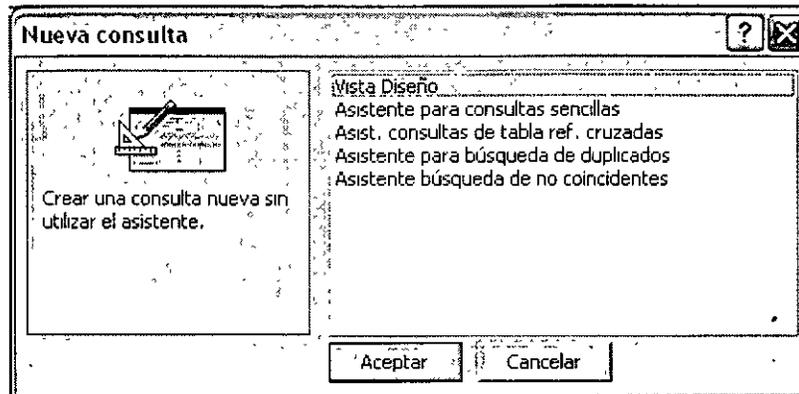
Entender las Consultas

Las consultas son objetos de bases de datos que permiten extraer datos desde una Base de datos para utilizarlos dentro de otro modo: como de datos utilizados en un informe impreso, por ejemplo, o para producir una lista de elementos que se quiere utilizar en un control de búsqueda en un formulario de entrada. Una consulta se puede basar en una única tabla o en muchas Tablas relacionadas. Además de campos trazados directamente desde tablas, una consulta también puede contener campos calculados que transforman datos, añadiendo impuestos comerciales a una cantidad de factura, por ejemplo, o llevando a cabo análisis estadísticos (totales, medias y el resto) en Grupos de registros trazados desde tablas múltiples.

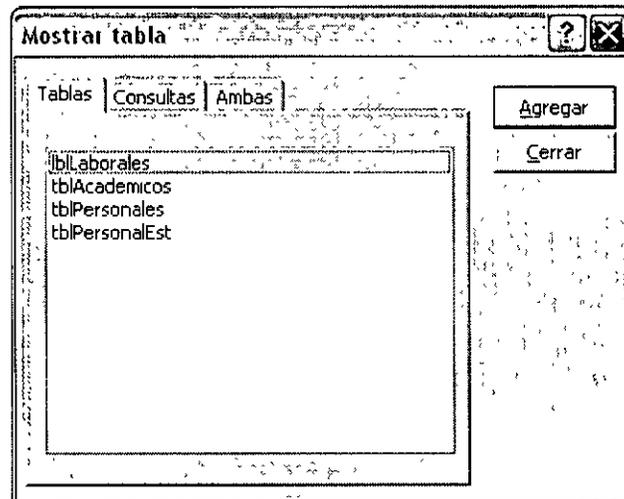
Access XP permite Crear varios tipos de consultas. Lo más común es una consulta de selección, que extrae información de una o más tablas. También puedes crear consultas de referencias cruzadas, que agrupan y resumen información en un formato de filas y de columnas, como una tabla dinámica de **Excel**. Algunas de las cosas más potentes (y potencialmente peligrosas) que puedes hacer con **Access XP** involucran consultas de acción, que realmente cambian los datos en una tabla subyacente basada en los criterios que define en la consulta.

Como las consultas, los filtros permiten trabajar con un subconjunto de registros en una Base de datos. Los filtros ofrecen un modo rápido de limitar temporalmente la visualización de registros en las vistas *Hoja de datos* o *Formulario*. Puedes crear un filtro introduciendo datos en un formulario o haciendo una selección en la vista *Hoja de datos*.

Para crear una consulta nueva, elige Insertar, Consulta. Aparece el cuadro de diálogo Nueva consulta que permite empezar partiendo desde cero en la vista Diseño, o utilizar un Asistente para crear uno de varios tipos específicos de consultas.



Elige en el cuadro de dialogo las tablas o consultas que formarán tu nueva consulta.



Cómo seleccionar consultas

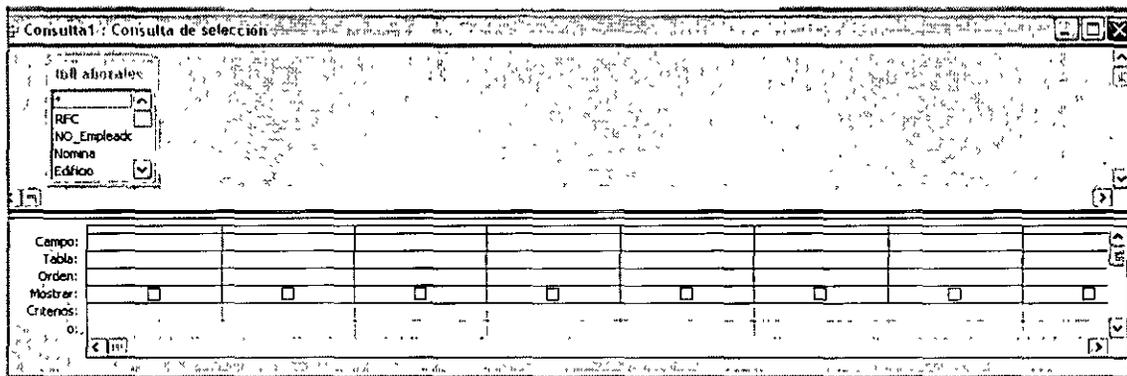
Cuando utilices la vista Diseño para crear una consulta nueva a partir de cero, Access XP crea una consulta de selección predeterminada. Como el Nombre lo indica, el propósito de una consulta de selección es recoger datos de una o más tablas y presentarlos en un formato que guarda como parte de la propia consulta. Seleccionar las consultas puede incluir cualquiera de los elementos siguientes:

- ❖ Los campos trazados desde una o más tablas o consultas. Puedes basar una consulta en otra consulta, una técnica común cuando deseas crear una vista de resumen de datos contenidos de múltiples tablas. Access XP utilizará relaciones definidas para relacionar registros desde diferentes orígenes y encontrar conexiones

relevantes entre los datos. También puedes definir las relaciones nuevas entre tablas y/o consultas y definir como parte de la consulta.

- ❖ Campos calculados, que muestran el resultado de expresiones utilizando campos desde una o más tablas de origen.
- ❖ Totales, que llevan a cabo operaciones, como una suma o un promedio, en los campos desde una tabla de origen.
- ❖ Criterios de selección, que definen el conjunto específico de registros que la consulta devuelve. Por ejemplo, en una tabla *Facturas* puede definir criterios para el campo *FechaFactura* para devolver las *Facturas* emitidas en los últimos 30 días.
- ❖ Instrucciones de ordenación, que disponen los resultados de la consulta de orden numérico, alfabético o cronológico para uno o más columnas.
- ❖ Campos ocultos, que están incluidos con el propósito de definir criterios u ordenar instrucciones, pero que no se muestran en realidad en los resultados de la consulta.

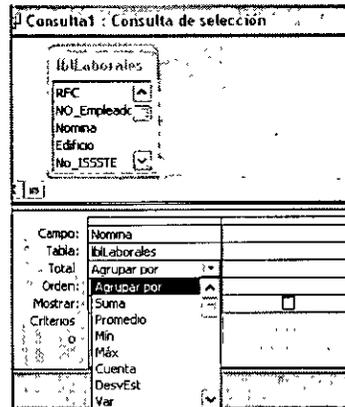
Cuando tengas una consulta, guarda las instrucciones para recuperar y ver registros desde una Base de datos, no desde los propios registros. Como resultado, ejecutar una consulta guardada siempre muestra el conjunto de datos actuales.



En la vista *Diseño*, una consulta incluye dos paneles: el panel superior contiene listas de campos para cada tabla y consulta utilizadas como origen de datos; este panel también muestra relaciones entre los orígenes de datos. El panel inferior contiene una cuadrícula con una columna para cada campo que forma parte de la consulta. Cuando diseñas una consulta, puedes arrastrar cualquier referencia de campo directamente desde las listas en el panel superior, o puede hacer una selección en las listas desplegadas que aparecen cuando activas una columna dada en la cuadrícula. Puedes introducir manualmente las columnas calculadas o con la ayuda del *Generador de expresiones*.

En las filas debajo de cada nombre de campo, puedes ver las especificaciones que explícitamente determinan el contenido de la consulta:

- La fila *Tabla* muestra el origen de cada campo. Esta fila está visible por defecto.
- La fila *Total* permite especificar las operaciones que se deben realizar en este campo: suma, promedio, etc. Esta fila está normalmente oculta. La selección predeterminada es *Agrupar por*, que muestra todos los valores en el campo seleccionado sin llevar a cabo un cálculo.

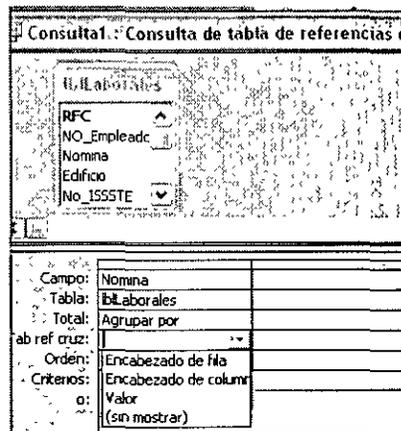


- La fila *Orden* especifica si una columna particular se utilizará para ordenar y, si así es, si la ordenación está en orden ascendente o descendente. Si especifica un tipo de ordenación en las columnas múltiples. **Access XP** ordena por cada columna, en orden de izquierda a derecha.
- La fila *Mostrar* contiene una marca de verificación para cada campo que aparecerá como parte de los resultados de la consulta. Desactiva esta casilla de verificación cuando quieras utilizar un campo para ordenar o filtrar, pero no quieres que aparezca en la vista *Hoja de datos*.
- La fila *Criterios* contiene una expresión de criterios para determinar qué registros se incluirán en la consulta.

Consultas de referencias cruzadas

Otro tipo de consulta, conocido como de referencias cruzadas, transforman los datos orientados hacia registros en una vista de resumen que se parece a la hoja de trabajo de **Excel**. Por ejemplo, supongamos que quieres mostrar la devolución media de números iguales de una Base de datos rellena con datos históricos sobre el mercado de acciones. Utiliza la consulta de referencias cruzadas para resumir los datos en una orientación fila-columna, donde los encabezados de fila representan grupos de acciones por nombre de industria y los encabezados de columna representan los tres mayores intercambios de acciones. Ten en cuenta (por ejemplo) que esta consulta puede utilizar sólo un grupo de columnas procedentes de una Base de datos que contiene 17 campos y el resultado filtra una Base de datos de casi 2.200 registros en un informe de resumen con sólo 200 registros dispuestos por categorías.

La cuadrícula en el panel inferior de la ventana incluye una fila *Tab ref cruz* que no aparece en las consultas de selección, donde el contenido de los campos seleccionados se identifican como *Encabezado de columna* y *Valor* de la consulta de referencias cruzadas. Para añadir la fila *Tab ref cruz* a la cuadrícula de diseño de una consulta de selección, elige *Consulta*, *Consulta de tabla de referencias cruzadas*.



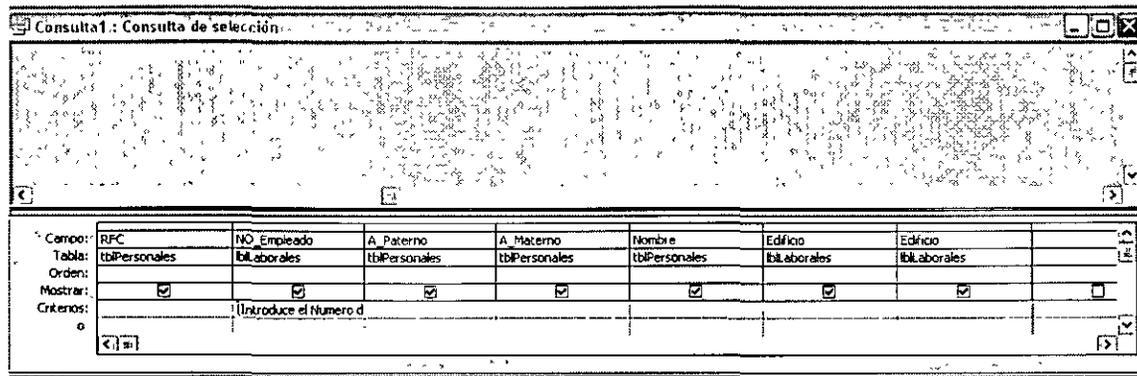
La creación de una consulta de referencias cruzadas desde cero en la ventana *Diseño* es un proceso bastante claro, pero es mucho más fácil cuando utilizas el *Asistente para consultas de referencias cruzadas*.

Consultas parametrizadas

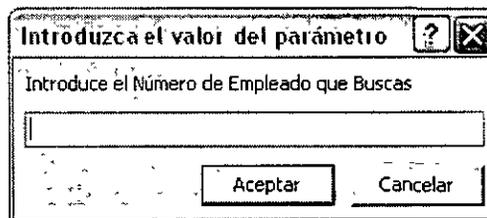
Normalmente, guardar una consulta incluye todos los criterios que hayas definido para esa consulta. Si deseas ver todos los pedidos por producto para todos los clientes, es fácil guardar una consulta que extraiga esos resultados desde el contenido actual de la base de datos cada vez que la ejecutas. ¿Pero qué debes hacer cuando quieras especificar ligeramente los diferentes criterios cada vez que ejecutes la consulta? Por ejemplo, ¿qué pasa si deseas introducir un número de cliente específico o un precio máximo cuando ejecutas una consulta? Para esa tarea, necesitas una consulta parametrizada.

Cada vez que abras una consulta parametrizada, **Access XP** muestra un cuadro de diálogo que te pide que introduzcas un fragmento de datos a utilizar en los criterios de selección para la consulta. Define la solicitud de entrada como parte de la definición de la consulta.

Para crear una consulta parametrizada, abre la consulta en vista *Diseño* y haz clic en el cuadro *Criterios* para el campo en el que deseas añadir los criterios de selección. La expresión debería incluir el texto que deseas ver como solicitud de entrada, escrito entre paréntesis cuadrados donde normalmente introducirás un valor constante.



Cuando abres una consulta parametrizada, **Access XP** muestra un cuadro de entrada que contiene una solicitud para introducir un parámetro de selección en un proceso interactivo. El valor que el usuario introduce en la solicitud se convierte en el parámetro numérico en una expresión mayor o igual que, que, a su vez, se convierte en un criterio de selección para la consulta. Por ejemplo, si introduces 10 en el cuadro de diálogo *Introducir valor del parámetro*, **Access XP** muestra una lista de empleados con el criterio que introdujiste.



La consulta parametrizada más simple incluye un único valor; pero también puedes utilizar comodines o multiplicar parámetros como parte de una consulta.

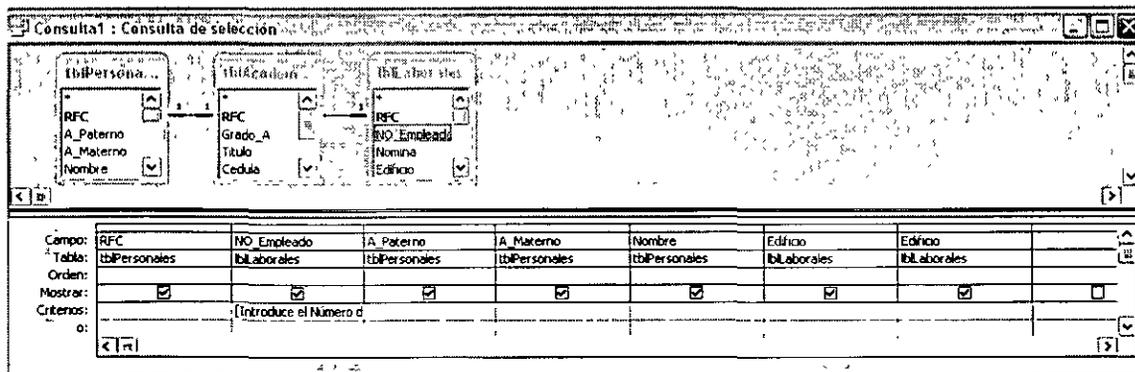
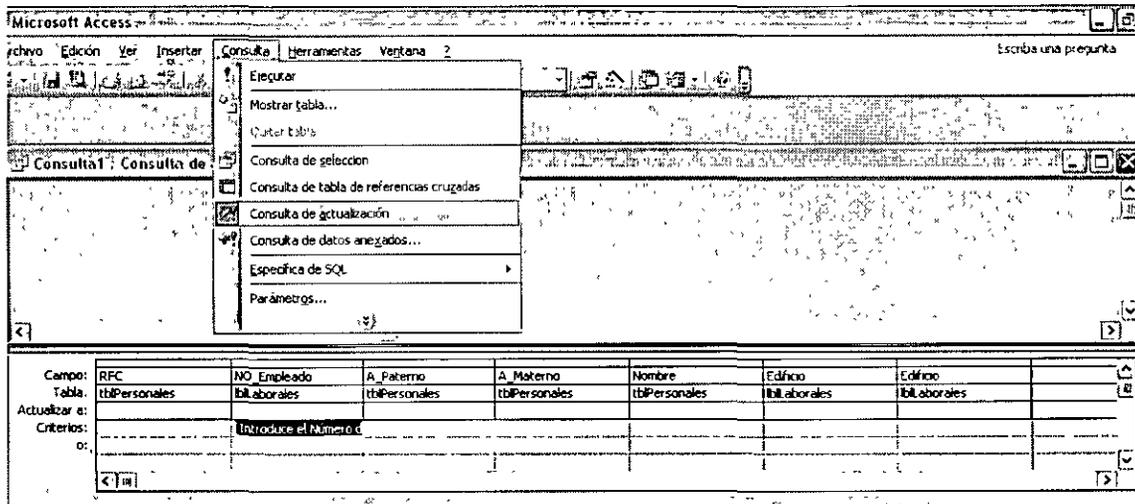
- ❖ Para crear un cuadro de diálogo de entrada que solicite al usuario que introduzcas el primero de una cadena de búsqueda, utiliza un asterisco con el parámetro. Por ejemplo, introduciendo *Como [Introducir el comienzo del nombre del producto] & "*"* buscará todos los registros en los que el campo especificado empieza con el valor que el usuario introduce.
- ❖ Para buscar una cadena en cualquier sitio en un campo dado, utiliza dos asteriscos *Como "*" & [Introduzca cualquier texto que aparezca en el Nombre del producto] & "*"* servirá.
- ❖ Para definir un rango de principio y de final de número o fechas, utiliza dos parámetros en una sola expresión, como: *Entre [Introducir fecha inicial] Y [Introducir fecha final].* Cuando ejecutes una consulta con parámetros múltiples, **Access XP** muestra un cuadro de diálogo de entrada para cada uno.

Cabe aclarar que los parámetros de búsqueda deben de concordar con el tipo de datos según el campo donde se introduzcan estos criterios, esto quiere decir que no puedo introducir un parámetro de búsqueda de fechas si el campo contiene texto o números.

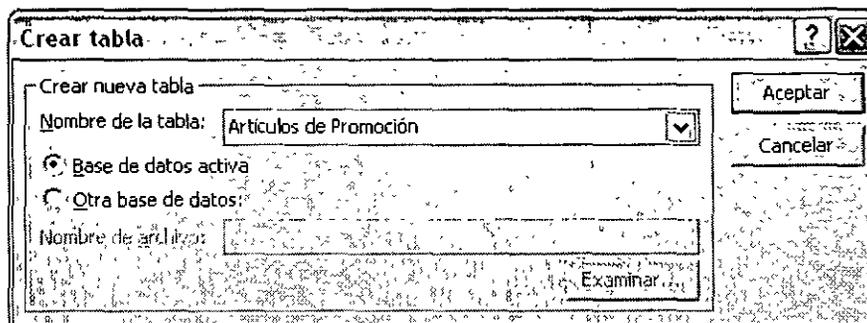
Consultas de acción

Una consulta de acción cambia potencialmente los datos en una tabla existente o crea una tabla nueva. Access XP permite crear cuatro tipos de consultas de acción:

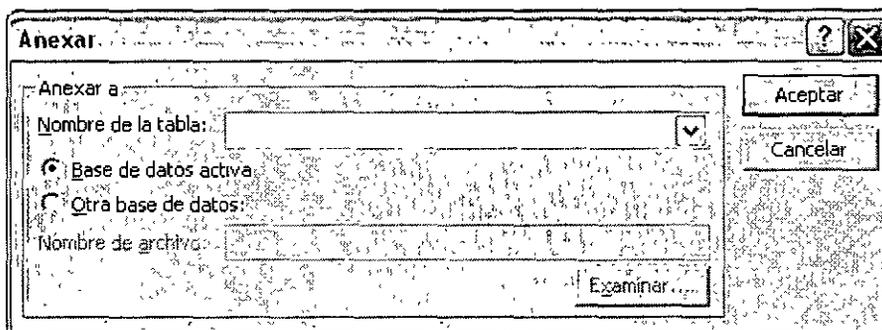
- ❖ Una *Consulta de actualización* reemplaza los datos en registro existentes. En el diseño de una consulta de actualización, escribe criterios de selección para identificar los registros de destino y proporciona una expresión que genere el reemplazo de datos. Utiliza una consulta de actualización para cambiar un grupo de registros enseguida: cuando cambia un código de área, por ejemplo, o para hacer un aumento de precio general.



- ❖ Una consulta *Crear tabla* crea un objeto de tabla nuevo a partir del resultado de la propia consulta. Por ejemplo, puede que crees una consulta que produce una lista de todas las compañías a las que no han realizado envíos en el pasado año y copia esos registros a una tabla *Compañías de envío*. Puedes elegir una tabla desde la base de datos actual o desde otro archivo de base de datos. Si introduces el nombre de una tabla que no existe actualmente, **Access XP** la crea.



- ❖ Una *Consulta de datos anexados* añade registros nuevos a una tabla existente desde una consulta origen. Este tipo de consulta se utiliza más comúnmente cuando importa datos desde una fuente externa. Las consultas de datos anexados permiten tomar algo o toda la información importada y desplazarla a un conjunto existente de tablas en el formato correcto. Cuando elijas *Consulta*, *Consulta de datos anexados*, **Access XP** solicita que introduzcas un nombre de tabla utilizando un cuadro de diálogo que es idéntico al de la consulta *Crear tabla*.



- ❖ Una Consulta de eliminación quita los registros que coinciden con los criterios de selección especificados para una tabla existente. Puedes utilizar una consulta de eliminación para purgar registros anticuados: Access añade una fila *Eliminar* a la cuadrícula de diseño de consulta.

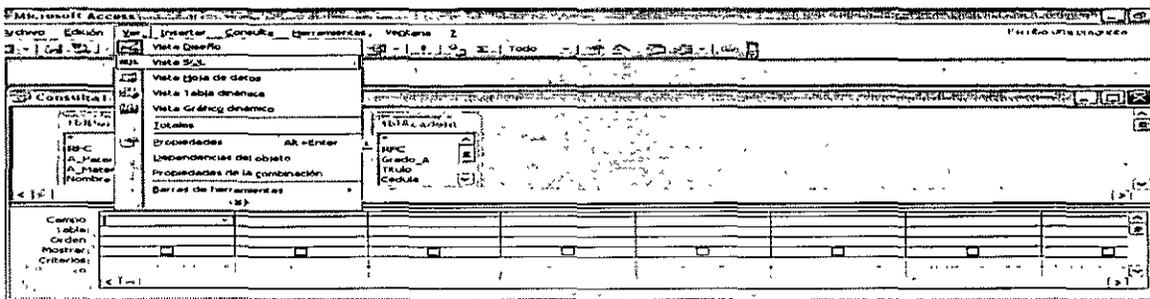
A continuación tienes dos mecanismos infalibles que pueden impedir un desastre en los datos:

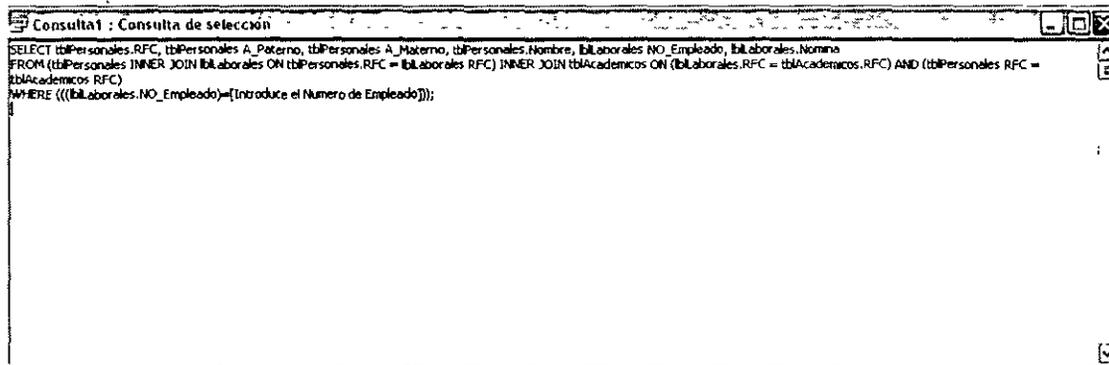
- ❖ Antes de que empieces a diseñar una consulta de acción, crea una copia de la tabla que tienes intención de modificar. En la ventana de la base de datos, selecciona el icono de la tabla, pulsa Ctrl+C para copiarla al Portapapeles y, a continuación, pulsa inmediatamente Ctrl+V. En el cuadro de diálogo *Pegar tabla*, introduce un nombre como *Copia de Nombredetabla*, elige la opción *Estructura y datos* y haz clic en *Aceptar* para crear la copia. Si la consulta de acción no funciona adecuadamente, puedes volver al estado original eliminando la tabla modificada y renombrando la copia de seguridad con el mismo nombre que el original.
- ❖ Antes de ejecutar una consulta de acción, prueba sus efectos diseñando una consulta de selección que utilice los mismos criterios de selección. Cuando abras esta consulta en la vista *Hoja de datos*, inspecciona los resultados cuidadosamente, porque estos registros serán, en última instancia, los mismos que cambiarán como resultado de la consulta de acción. Después de investigar los resultados, vuelve a la vista *Diseño* y elige *Consulta*, seguido de la elección de menú para el tipo de consulta que desees crear.

Consultas SQL

El Lenguaje de Consultas Estructurado (**SQL**) es un lenguaje de base de datos estándar de la industria que está disponible para utilizarlo en varios entornos **Software**. En general, puedes crear consultas de **Access XP** de forma interactiva comprobando las opciones a través de un cuadro de diálogo sin tener que tratar con **SQL**. Sin embargo, cada consulta de **Access XP** existe entre bastidores como una sentencia **SQL** y, en algunos casos, puedes encontrar útil trabajar directamente con el código **SQL**.

Para ver la sentencia **SQL** que subyace a una consulta que hayas creado en **Access XP**, abre la consulta en la vista *Hoja de datos* o *Diseño* y a continuación elige *Ver, Vista SQL*. **Access XP** abre la ventana de vista **SQL**, mostrando la sentencia que se combina con la consulta que ha creado en la vista *Diseño*. Por ejemplo, el equivalente de **SQL** de la consulta *Crear tabla*.





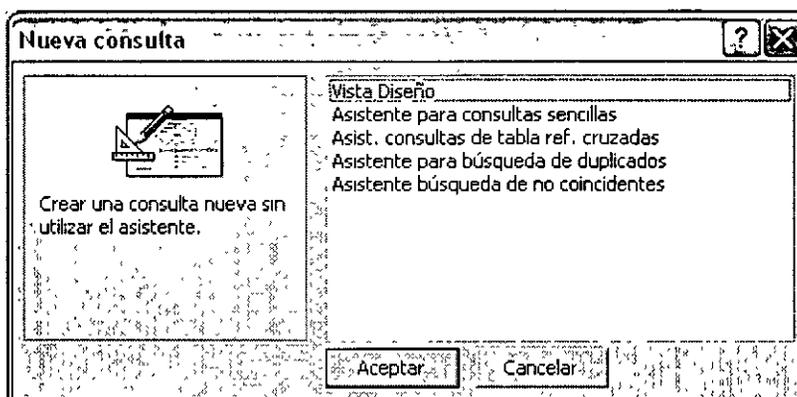
¿Por qué debería importarte el código **SQL**? En circunstancias normales, no lo necesitas. Sin embargo, el código **SQL** puede ser excepcionalmente útil en una circunstancia específica: cuando desees utilizar una consulta compleja como origen de datos para otro objeto, como un formulario o informe. Abre la consulta subyacente en la vista **SQL**, copia el contenido entero al Portapapeles y pégalo en el cuadro de propiedad *Origen del registro* para el formulario o informe.

Creación y Modificación de Consultas

Como otros asistentes de **Office**, los asistentes de consulta de **Access XP** son eficaces guiándolo con soltura por los pasos de un complejo proceso, proporcionando explicaciones detalladas de las elecciones que necesitas hacer y permitiéndote ver representaciones gráficas de los resultados.

Aunque la vista *Diseño* es un entorno más versátil en el que crear consultas, es, a menudo, más fácil empezar a utilizar un asistente para crear una consulta básica. Después de terminar con el asistente, puedes Abrir la consulta en la vista *Diseño* para modificar el resultado.

Para ver una lista de asistentes de consulta disponibles, pulsa **F11** para abrir una ventana de base de datos y, a continuación, elige *Insertar, Consulta*. El cuadro de diálogo nueva consulta muestra cuatro asistentes para consultas. Haz doble clic en cualquiera de estas opciones para iniciar un asistente.



En sus esfuerzos para simplificar tareas, los asistentes de consulta, algunas veces restringen sus elecciones. En el asistente para consultas de referencias cruzadas, por ejemplo, debe basar la consulta nueva en una sola tabla existente o consulta. Si deseas utilizar campos de más de una tabla, debes crear primero una consulta que contenga todos los campos de destino. Al contrario, si creas una consulta de referencia cruzadas en vista *Diseño*, puedes añadir campos desde dos o más tablas relacionadas.

El primer paso en el *Diseño* de una consulta a partir de cero en vista *Diseño* es seleccionar las tablas o consultas en las que se basará la nueva consulta. Puedes añadir cualquier combinación de tablas y consultas existentes al panel superior de la ventana de *Diseño* de consulta. Elige *Ver*, *Mostrar tabla* o haz clic en el botón *Mostrar tabla* en la barra de Herramientas de **Access XP** para abrir este cuadro. El cuadro de diálogo *Mostrar tabla* proporciona lista de todos los objetos disponibles para la creación de una consulta nueva.

Como insertar, eliminar y reorganizar los campos

Para añadir un campo nuevo a una consulta, utiliza cualquiera de las técnicas siguientes:

- ❖ Arrastra el nombre de campo directamente desde una lista en el panel superior y colócalo sobre la cuadrícula. Cuando sueltes el botón del ratón, aparece el campo nuevo en la cuadrícula, a la izquierda del campo en el que lo haz soltado.
- ❖ Haz doble clic en cualquier nombre de campo en una lista para añadir ese nombre de campo a la siguiente columna disponible.
- ❖ Haz clic en el botón de la flecha hacia abajo en la celda superior de cualquier columna vacía desde la lista desplegable de todos los campos disponibles. Ten en cuenta que los campos y las Tablas se identifican utilizando la notación *NombreTabla.NombreCampo*, aunque cuando seleccionas el campo sólo aparece su nombre en el cuadro.

En algunos casos, puede que quieras añadir todos los campos desde una tabla seleccionada a la cuadrícula de consulta. Si no necesitas trabajar con las propiedades de campos individuales de la cuadrícula, puedes arrastrar el asterisco desde la parte superior de una lista de campos a una celda en la cuadrícula. Cuando lo haces, **Access XP** representa la tabla en la notación *NombreTabla.**. En anotación significa que todos los campos de tabla se mostrarán en la salida de la consulta. Si necesitas trabajar con los ajustes individuales de campo en la cuadrícula de diseño, haz doble clic en la barra de título de la lista de la tabla de destino para seleccionar todos los campos y, a continuación, arrastra la selección a la cuadrícula de diseño.

Definición de una columna calculada

Para definir una columna calculada, introduce una expresión en la fila *Campo*. Puedes introducir una expresión directamente, como por ejemplo *[PrecioUnidad]*[UnidadesVendidas]*; Ten en cuenta que debes utilizar paréntesis en los *Nombres de campo*. O puedes hacer clic en el botón *Generar* para utilizar el *Generador de expresiones* para crear un campo calculado.

Si introduces una expresión sola, **Access XP** añade un nombre predeterminado para el campo calculado, utilizando los valores genéricos *Expr1*, *Expr2*, etc. Para especificar un nombre más descriptivo, haz doble clic en esta etiqueta genérica y escribe un nombre para sustituirla.

Definición de criterios

Cualquier expresión que introduzcas en la fila *Criterios* indica a **Access XP** que muestre sólo los registros que satisfacen los criterios. En estas expresiones pueden ser extremadamente simples: *>10*, por ejemplo, indica a **Access XP** que muestre todos los registros en los que el valor del campo seleccionado es mayor que 10. Puedes combinar varios criterios de selección en una sola columna o en columnas múltiples.

- ❖ Las expresiones en las columnas múltiples en una sola fila se tratan como criterios Y. Para seleccionar una parte de los resultados de la consulta, un registro debe satisfacer todos los criterios de una fila dada.
- ❖ Las expresiones en las diferentes filas se tratan como criterios O. Para seleccionarlas, un registro debe satisfacer los criterios en cualquier fila.

Cuando crees una entrada en la fila *Criterios*, puedes utilizar cualquier expresión que se evalúe como Verdadera o Falsa. Los bloques de construcción más comunes para los campos Numéricos y de fecha son los operadores de comparación: *<* (menor que), *>* (mayor que) *<=* (menor o igual que), *>=* (mayor o igual que), *<>* (no igual) e *=* (igual). Además, con **Access XP** puedes hacer una comparación adicional utilizando la palabra clave *Entre*, que expresa un rango numérico para compararlo con el valor de un campo.

También puedes utilizar los operadores lógicos Y, O y No. Si las dos expresiones están conectadas por Y, la operación es sólo verdadera si ambas expresiones son verdaderas. Por el contrario, la operación O es verdadera si cualquiera de las expresiones es verdadera. La operación No da como resultado lo opuesto de la expresión que modifica, Verdadero si la expresión es falsa, o Falso si la expresión es verdadera.

Por último, para los campos de texto, utiliza el operador *Como*, con o sin comodines. Si introduces una cadena de texto en el cuadro *Criterios* para un campo dado, **Access XP** añade automáticamente el operador *Como* y encierra la cadena entre comillas.

Definición de las propiedades de consulta

Además de todos los ajustes de campos individuales que están disponibles en la cuadrícula de diseño para una consulta, también puedes configurar muchos de los ajustes que se apliquen a toda la consulta. Abre una consulta y cambia a la vista *Diseño*. Si es necesario, haz clic en el botón *Propiedades* para ver el cuadro de diálogo *Propiedades* y, a continuación, haz clic en cualquier sitio en el área de fondo de la ventana de diseño de la consulta.

La ventana *Propiedades de la consulta* contiene una lista de los ajustes que se aplican al tipo específico de consulta que estas creando. Aunque algunos de estos ajustes son para usos especializados, los siguientes son valiosos en situaciones comunes donde puede que utilices una consulta:

- ❖ **Valores superiores.** Devuelve un número específico o porcentaje de registros. Este ajuste se utiliza más a menudo junto con un ajuste de ordenación: para ver los 10 productos más caros, por ejemplo, haz clic en la columna *PrecioUnidad* y ajusta *Los 10 más...* y la propiedad *Orden a Descendente*.
- ❖ **Valores únicos.** Devuelve un resultado de consulta en el que no existen registros duplicados. Utiliza este ajuste si deseas extraer un conjunto único de valores de una base de datos, como nombres de proveedores. **Access XP** elimina los duplicados desde el conjunto resultante en base a los registros visibles en el resultado de la consulta.
- ❖ **Registros únicos.** Devuelve un resultado de consulta después de eliminar los registros duplicados en el origen de datos. Dependiendo de los campos que elige mostrar, puedes ver los valores duplicados en los resultados de consulta.
- ❖ **Encabezados de columna.** Esta propiedad, utilizada sólo en las consultas de referencias cruzadas, permite limitar las columnas que se van a ver. Separa las entradas por puntos y comillas. En un origen de datos que contiene un campo *OficinaRegional*, por ejemplo, puedes especificar *Este;Oeste;Sur* en esta propiedad. **Access XP** ignora los demás valores cuando se realizan en la consulta de referencias cruzadas y muestra estas tres columnas en el orden especificado.

- ❖ **Mostrar todos los campos.** Especifica que deseas que la consulta devuelva todos los campos de todas las tablas incluidas en la consulta, independientemente de si el Nombre de campo está en la cuadrícula de Diseño o la casilla Mostrar activada. Cuando ajustes esta propiedad, debes añadir campos a la cuadrícula sólo para ajustar las propiedades *Criterios* y *Orden*.
- ❖ **Vincular campos secundarios y Vincular campos principales.** Se utilizan para ajustar la relación entre un formulario principal y un subformulario u otro objeto incrustado. Normalmente, **Access XP** ajusta esta propiedad automáticamente en base a las relaciones definidas entre las tablas.

Creación y Aplicación de Filtros

Cuando utilizas criterios en una consulta, **Access XP** muestra un subconjunto de registros en el origen de datos subyacente. Para modificar los criterios de selección, tienes que Abrir la consulta en vista *Diseño* e introducir una o más expresiones en la fila *Criterios* de la cuadrícula de diseño. A continuación tienes la opción de guardar estos criterios como parte del diseño permanente de la consulta.

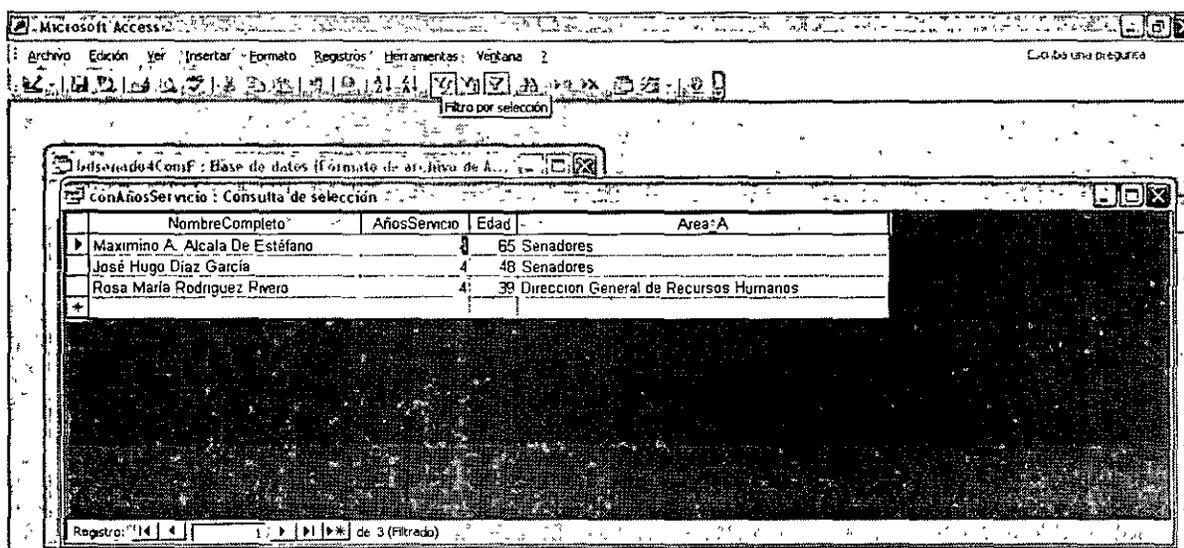
Un filtro es un modo más rápido y conveniente de centrar la atención temporalmente en los registros específicos en una consulta o tabla. Cuando abras una consulta o tabla en la vista *Hoja de datos*, puedes desarrollar y aplicar filtros rápidamente, sin cambiar a la vista *Diseño*, y volver a la vista sin filtrar siempre que desees ver el conjunto entero de registros otra vez.

El modo más fácil de crear un filtro es basarlo en el contenido de un registro existente. Cuando una consulta o tabla se abre en la vista *Hoja de datos*, haz clic en el botón *Filtro por selección*. La acción de filtro exacto depende de la selección que hagas.

- ❖ Si seleccionas todo el contenido de un campo o coloca el punto de inserción en un campo sin hacer ninguna selección, el filtro encuentra todos los registros en los que el contenido de ese campo coincide con el contenido exacto de la celda seleccionada. Esta técnica es especialmente eficaz cuando un campo contiene una descripción de categoría o un nombre que se repite en registros a través del origen de datos.
- ❖ Si seleccionas una porción de la celda que incluye el primer carácter de la misma, el filtro encuentra todos los registros por los que empieza el campo con la selección. Si deseas ver sólo esos productos que empiecen con la letra A, por ejemplo, encuentra cualquier producto que empiece con esa letra, selecciónala y haz clic en el botón *Filtro por selección*.
- ❖ Por último, si seleccionas una porción de contenidos de celdas que no incluyen el primer carácter, el filtro muestra todos los registros en la consulta que contiene la cadena de caracteres o números en cualquier sitio en el campo de destino. Esta técnica es útil para

buscar registros basados en los contenidos de un campo que contiene texto variable en lugar de entradas consistentes.

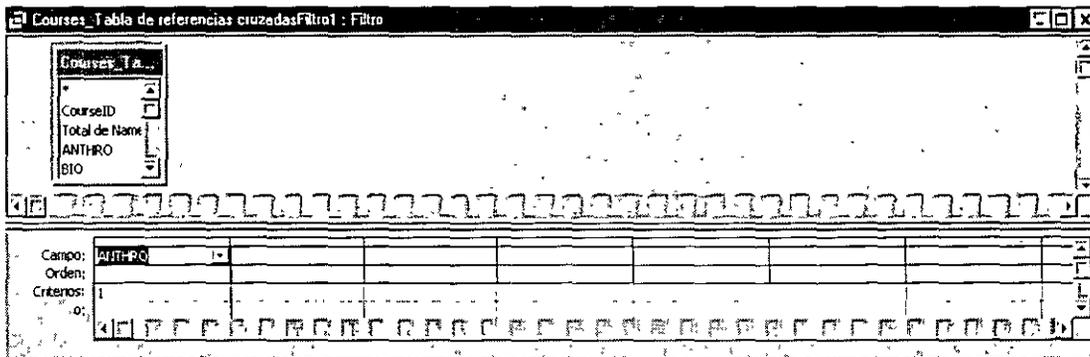
Si ningún registro que coincide con el filtro está visible, haz clic en el botón *Filtro por formulario*. Esta opción elimina el contenido actual de la consulta o tabla y muestra una cuadrícula simple que contiene cada una de las columnas en la consulta o tabla, con una celda en blanco debajo de cada una. Cuando hagas clic en cualquiera de estas celdas en blanco, puedes introducir una expresión o seleccionar de entre una lista desplegable de elementos únicos contenidos en cada campo de la consulta.



Después de introducir los criterios en la ventana *Filtro por formulario*, haz clic en el botón *Alternar filtro* para aplicar el filtro y consulta los resultados. Si necesitas refinar el filtro, haz clic en el botón *Filtro por formulario* otra vez y añade o elimina los criterios.

Independientemente de cómo crees un filtro, siempre puedes restaurar la vista de todos los registros, haciendo clic en el botón *Alternar filtro* otra vez. Ten en cuenta que la sugerencia para este botón dirá *Aplicar filtro* o *Quitar filtro*, dependiendo del estado actual.

Para crear filtros complejos o editar un filtro existente, elige *Registros*, *Filtro*, *Filtro u orden avanzado*. La ventana de filtro resultante contiene una lista de campos y una cuadrícula de diseño, idéntica a la vista *Diseño* para una consulta. Para refinar un filtro, añade uno o más campos a la cuadrícula y escribe expresiones de criterios para seleccionar un subconjunto de registros; luego haz clic en el botón *Aplicar filtro* para ver el resultado del filtro.



Diseño y Uso de Formularios

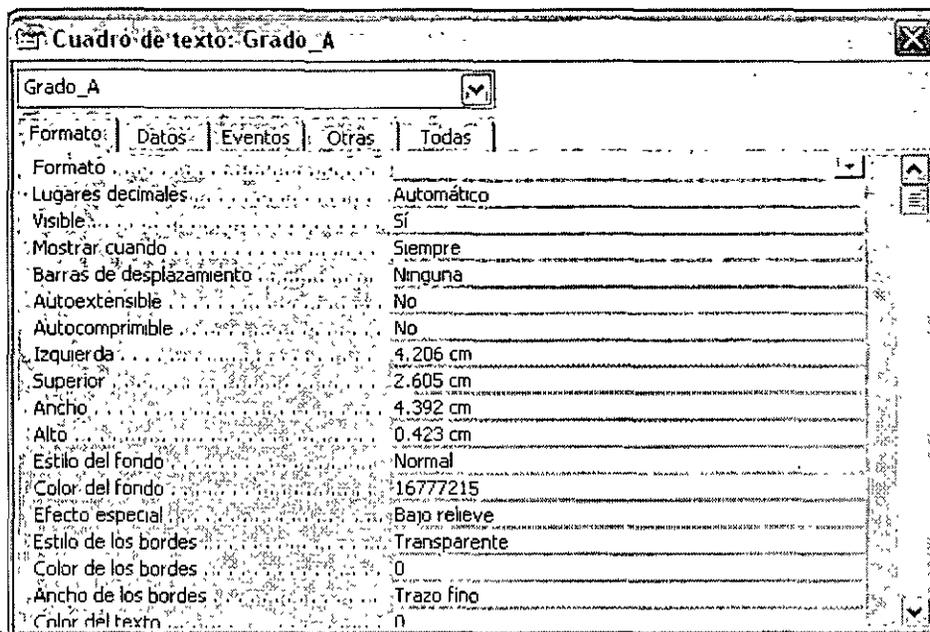
Uso de Formularios para Ver e Introducir Datos

En **Access XP**, un formulario es un objeto de base de datos formateando, generalmente utilizado para visualizar registros uno a uno en una ventana de pantalla. Los formularios se utilizan más normalmente para crear ventanas de entrada de datos para rellenar; en este caso, utiliza la tabla o consulta como el origen de datos para el formulario.

Un formulario bien diseñado facilita y proporciona precisión a la hora de introducir datos. Si deseas que otras personas introduzcan datos en una base de datos, por ejemplo, no es realista esperar que trabajen directamente con cada tabla en la vista *Hoja de datos*; en lugar de esto, puedes crear un formulario que muestre sólo los campos con los que deseas que trabajen, trazados desde varias tablas relacionadas. Disponiendo cuidadosamente las áreas para entrada de datos y proporcionando texto explicativo, puede guiar al usuario a través del proceso de entrada.

Access XP proporciona un diseño de formulario y entorno de desarrollo que permite controlar completamente el aspecto y funcionalidad de un formulario. Los bloques de construcción de un formulario son objetos llamados controles, que incluyen cuadros de texto, etiquetas, botones de opción, listas, botones de comando y otros elementos familiares de la interfaz de **Windows**:

- ❖ Algunos controles están directamente unidos a un campo en una consulta de tabla; en el idioma peculiar de **Access XP**, éstos se llaman controles vinculados. Cuando introduces datos en un control que está destinado a un campo particular, **Access XP** lo añade a ese campo; cuando visualizas datos utilizando un formulario, **Access XP** controla la propiedad Origen del control para que cada control vea qué datos debería mostrar.



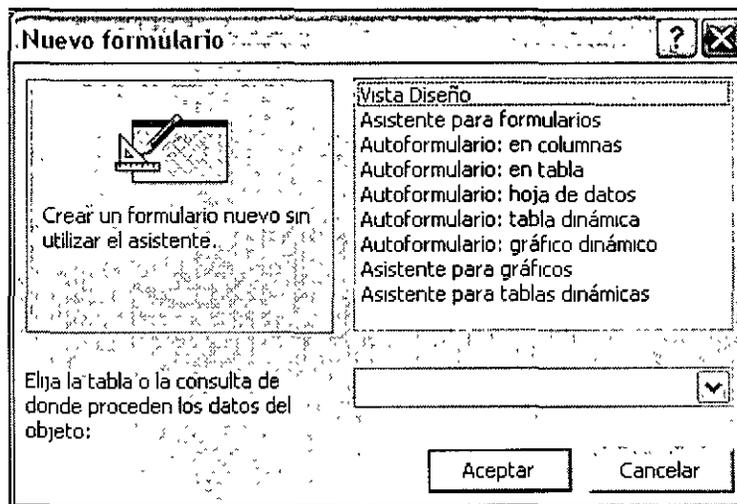
- ❖ Algunos controles del formulario no están vinculados, es decir, no están unidos a ningún origen de datos. Por ejemplo, una línea, cuadro o etiqueta de texto autónomo es un control que no está vinculado. Cuando introduces una expresión en el cuadro de propiedad *Origen de control*, **Access XP** crea un control calculado. La expresión $=[\text{Cantidad por Unidad}] * 2$, por ejemplo, multiplica el contenido del campo *Cantidad por Unidad* por 2 y muestra el resultado en el formulario.
- ❖ Cuando abres un formulario en la vista *Diseño*, puedes cambiar la fuente, el tamaño de fuente, el color, los bordes y otras Propiedades de formato de cualquier control.
- ❖ Un formulario puede incluir hasta cinco secciones. Los propios datos normalmente aparecen en la sección *Detalle*; además, cada formulario puede tener hasta dos encabezados y dos pies de página, con una combinación de encabezado y pie de página para el propio formulario y otra para las páginas individuales. Utiliza el menú *Ver* para ocultar o mostrar los encabezados y los pies de página.
- ❖ En la vista *Diseño*, **Access XP** te permite colocar controles en una rejilla para procesar la alineación. También puedes agrupar y alinear controles.
- ❖ Cada control individual tiene docenas de propiedades que puedes ajustar. Por ejemplo, puedes definir el orden en el que se mueve de un control al siguiente ajustando la propiedad *ÍndiceDeTabulación* para cada control. También puedes ajustar las propiedades para todo el formulario: ajustando las propiedades, *Permitir ediciones*, *Permitir eliminación* y *Permitir agregar* a *No*, por ejemplo. Puedes asegurar que un formulario se utilice sólo para ver datos.

Los formularios simples generalmente muestran el contenido de registros de uno en uno, pero también puedes diseñar un formulario que incluye un subformulario, que muestra información desde una tabla relacionada o consulta. Utilizando este tipo de formulario, puedes desplazarte por grupos de registros, o buscar información utilizando filtros y otras herramientas de búsqueda. Ten en cuenta que los dos conjuntos de botones de navegación que aparecen en la parte inferior del formulario. Utiliza los botones de registro *Siguiente* y *Anterior* para que el formulario principal navegue por la tabla *Cientes*; utiliza los botones de navegación dentro del subformulario para pasar a la lista de productos para cada cliente.

Creación de un Formulario Nuevo

Como otros objetos de base de datos, **Access XP** proporciona varias formas de crear un formulario. Cuando hagas clic en el icono *Formularios* en la ventana base de datos y a continuación hagas clic en el botón *Nuevo*: **Access XP** abre el cuadro de diálogo *Nuevo Formulario*.

Las dos primeras opciones duplican los dos métodos abreviados en la parte superior de la ventana *Formularios*: una permite crear un formulario partiendo de cero, abriendo un formulario en blanco a la vista *Diseño*; el segundo método abreviado inicia el *Asistente para formularios*, que crea automáticamente un formulario después de seleccionar una tabla o consulta, seleccionar qué campos van en el formulario y elegir algunas opciones de formato. Este cuadro de diálogo también incluye tres opciones *AutoFormulario* que permite crear un formulario con un solo clic.



Creación de un formulario básico con un clic

Para los formularios simples, utiliza la característica *AutoFormulario* para convertir cualquier tabla o consulta en un formulario genérico que puedas utilizar. Para crear un *AutoFormulario*, selecciona cualquier tabla o icono de consulta en la ventana base de datos y a continuación elige *Insertar, AutoFormulario*. *AutoFormulario* es también la primera selección en la lista desplegable *Nuevo objeto* en la barra de herramientas de **Access XP**; si el icono de *AutoFormulario* está visible, haz clic en él para crear un *AutoFormulario* instantáneamente.

Los *AutoFormularios* son útiles, pero deberías tener en cuenta dos limitaciones significativas:

- ❖ Todos los campos de la tabla o consulta aparecerán en el *AutoFormulario*. Para una tabla pequeña que incluye sólo un grupo de campos, esto puede ser ideal, pero puede que desees ser más selectivo cuando diseñes un formulario para una tabla o consulta con más de 10 campos.
- ❖ El autoformulario coloca los campos en una columna, utilizando el orden de campos predeterminado en la tabla. Cambiar el orden de los campos puede ser un problema.

Si eliges una tabla que incluya una hoja de datos secundarias y a continuación creas un autoformulario, **Access XP** añade un subformulario. Puedes utilizar este formulario nuevo para ver, editar, o agregar registros nuevos en ambas tablas relacionadas.

Cuando cierras un objeto Autoformulario por primera vez, **Access XP** muestra un cuadro de diálogo que permite guardar el formulario. Haz clic en Sí e introduce un nombre para guardar el formulario, o haz clic en No para desechar el autoformulario sin guardarlo.

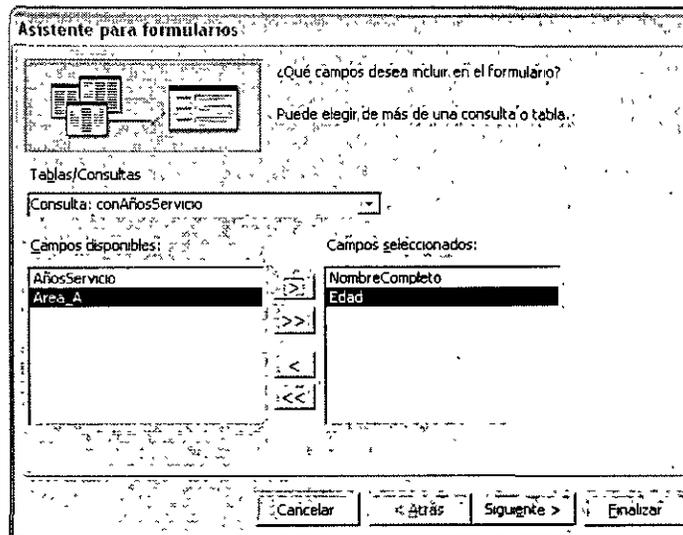
Uso del Asistente para formularios

Como se ha indicado anteriormente, *AutoFormulario* es rápido pero limitado. Para controlar mejor el diseño de un formulario, utiliza en su lugar el *Asistente para Formularios*. Necesitas unos cuantos pasos más que la opción *Autoformulario*, pero el enfoque paso a paso del asistente te permite elegir qué campos deberían aparecer en el formulario y a continuación personalizar el aspecto básico del mismo.

Para iniciar el *Asistente para Formularios*, haz clic en el botón *Formularios* situado en la parte izquierda de la ventana base de datos y a continuación haz doble clic en el método abreviado *Crear Formulario utilizando el Asistente*. En total, el asistente engloba cuatro pasos:

1. Elegir la tabla o consulta en la que basar el formulario, utilizando la lista desplegable *Tablas/Consultas*. En la lista de los campos disponibles, selecciona cada campo que quieras agregar al formulario y haz clic en el botón >; Haz clic en el botón >> para mover todos los campos al formulario. Conforme elaboras la lista de campos, las elecciones aparecerán en la lista *Campos seleccionados*. Haz clic en *Siguiente*.

2. Selecciona uno de los cuatro formatos diferentes para disponer los campos en el formulario. Conforme selecciones cada opción, ten en cuenta la representación gráfica del resultado. Haz clic en *Siguiente*.



3. Selecciona un estilo para el formulario; las opciones disponibles controlan la fuente, el tamaño, el color y el sombreado de las etiquetas y los controles de datos, así como el fondo del formulario en sí.
4. En la ventana final del *Asistente para Formularios*, proporciona un nombre nuevo para el formulario que estas a punto de crear. (El Nombre predeterminado es el mismo que el de la tabla subyacente o el de la consulta.) Elige si deseas abrir el formulario inmediatamente para ver o introducir datos, o si deseas abrirlo en la ventana *Diseño* para modificarlo. Haz clic en *Finalizar*.

Ver o Introducir Datos con un Formulario

Cuando abras un formulario, puedes ver, editar o introducir datos. Para navegar por un formulario utilizando el ratón, haz clic en los botones de la barra de navegación situada en la parte inferior del formulario. Para Abrir un formulario en blanco de modo que puedas Crear un registro Nuevo, haz clic en el botón Nuevo registro situado a la derecha.

Datos Laborales			
RFCI	AA5M390218-QY1	Fecha_In	1/07/2000
NO_Empleado	1803	H_Entrada	10:00
Nomina	Confianza	H_Salida	18:00
Edificio	Torre Caballero	Sueldo	20000
No_ISSSTE		Telefono_Of	53-45-32-13
Puesto	Asesor	Extencion	3213
Nivel			
Area_A	Senadores		

Registro: 14 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20

Para desplazarte por los registros con el teclado, utiliza estas técnicas:

- ❖ Pulsa Ctrl+AvPág o Ctrl+RePág para desplazarse al registro siguiente o al anterior, permaneciendo en el mismo campo.
- ❖ Utiliza las teclas Inicio y Fin para ir al primer o último campo en un formulario.
- ❖ Pulsa Ctrl+Inicio para moverte al primer registro; pulsa Ctrl+Fin para saltar al último campo en el último registro del origen de datos.
- ❖ Pulsa F5 para moverte al punto de inserción en el cuadro selector de registro, donde puedes introducir un número de registro específico.

Cuando un formulario esté abierto, hay tres botones *Filtro* disponibles en la barra de herramientas de **Access XP**. Igual que con las consultas, puedes hacer clic en el botón *Filtro por selección* o *Filtro por formulario* e introducir valores que restringen la selección de datos que aparecen en el formulario. Haz clic en el botón *Alternar filtro* para cambiar entre los datos filtrados y una vista de todos los registros en el origen de datos.

Cómo Cambiar el Diseño de un Formulario

Cuando abras un formulario en la vista *Diseño*, puedes manipular las propiedades de cualquier control para producir los efectos exactos y conseguir el aspecto que deseas. Puedes mover un control a una ubicación nueva dentro de un formulario, cambiar su tamaño y color; y cambiar las propiedades de datos y eventos que determinan en última instancia como funciona un control en el contexto del formulario que está diseñado. También puedes

añadir controles nuevos y cambiar el tipo del control existente desde el cuadro de texto predeterminado a un amplio conjunto de alternativas.

Para Abrir cualquier formulario en la vista *Diseño*, haz clic en el botón *Formularios* situado en la parte izquierda de la ventana base de datos, selecciona el objeto de formulario con el que deseas trabajar y haz clic en el botón *Diseño*. Si el formulario ya está abierto, haz clic en el botón *Diseño* para cambiar la vista *Diseño*.

Cuando trabajas con un formulario en la vista *Diseño*, tres elementos de interfaz son esenciales:

- ❖ Utiliza el *Cuadro de Herramientas* para añadir controles nuevos o cambiar controles existentes. Puedes personalizar los botones del *Cuadro de Herramientas* igual que cualquier otra barra de herramientas de Office; para activarlo o desactivarlo mientras estas trabajando con un formulario, haz clic en el botón *Cuadro de Herramientas* o elige *Ver, Cuadro de Herramientas*.
- ❖ La *Lista de campos* muestra una lista de todos los campos en la consulta o tabla en la que se basa un formulario. Si arrastras un nombre de campo desde la *Lista de campos* a un formulario, **Access XP** crea automáticamente un control de cuadro de texto destinado a ese campo. Si haces clic otra vez en el botón *Cuadro de Herramientas* primero y a continuación arrastras un campo al formulario, **Access XP** inicia un asistente que crea el tipo de control que haz seleccionado. Para mostrar u ocultar esta lista mientras está abierto un formulario en la vista *Diseño*, haz clic en el botón *Lista de campos* o elige *Ver, Lista de campos*.
- ❖ Para ajustar el aspecto o comportamiento de un control, sección o el propio formulario, abre el cuadro *Propiedades*. Puedes tener abierto el cuadro de diálogo *Propiedades* mientras trabajas; conforme seleccionas diferentes objetos, las *Propiedades* visualizadas en este cuadro de diálogo cambian para reflejar las elecciones disponibles.

Como agregar y modificar controles en la vista diseño

Cuando utilices el Asistente para formularios para crear un formulario, **Access XP** crea un control para cada campo que selecciona en el asistente. Sin embargo, es posible que quieras añadir un control nuevo o modificar uno existente para hacer que un formulario funcione como quieras.

Para añadir un control de cuadro de texto nuevo para un campo existente, arrastra el nombre de campo desde el cuadro *Lista de campos* y colócalo en el formulario. Sin embargo en algunos casos, un control de cuadro de texto no es adecuado. Por ejemplo, puede que quieras hacer que los usuarios escojan de una lista desplegable en lugar de escribir texto para rellenar un campo. O, en el caso de un campo *Sí/No*, puede que quieras añadir una casilla de verificación al formulario.

Para añadir otro tipo de control a un formulario, sigue estos pasos:

1. Asegúrate de que el *Cuadro de herramientas* está visible y que haz seleccionado el botón *Asistente para controles*. A continuación haz clic en el botón para el tipo de control que deseas añadir.
2. El procedimiento para colocar el control en el formulario es ligeramente diferente, dependiendo de si el control está vinculado o no.
 - ❖ Si deseas colocar un control no vinculado en el formulario, haz clic en la cuadrícula del formulario en el punto donde quieres que aparezca la esquina superior izquierda del control y arrastra el ratón hacia abajo y hacia la derecha para definir el tamaño del control.
 - ❖ Para crear un control vinculado, arrastra un nombre de campo de la Lista de campos y colócalo en el formulario.
3. Independientemente de si el control está vinculado o no, el asistente para controles se abre. Sigue las instrucciones del asistente para definir el origen de datos y comportamiento del control.

El *Cuadro de Herramientas* contiene botones para los controles comunes que puede que quieras agregar a un formulario. Colocando el puntero del ratón sobre un botón en el *Cuadro de Herramientas*, puedes ver una sugerencia que muestra el nombre del propio control. La lista siguiente describe algunos de los controles más comunes y útiles.

- ❖ **Casilla de verificación.** Introduce datos en un campo *Sí/No* o un grupo de opción. En un formulario que utilizas para rellenar los nombres de los clientes nuevos, por ejemplo, la tabla subyacente puede incluir un campo llamado *ListaCorrespondencia*. Si el valor de este campo es *Sí*, significa que el cliente desea recibir correos de promoción y puedes utilizar este campo para producir etiquetas de correo o cartas modelo. Utilizar una casilla de verificación para este tipo de datos es mucho más fácil que escribir *Sí* o *No*.
- ❖ **Cuadro combinado.** Combina las funciones de un texto y un cuadro de lista. El usuario puede introducir un valor en el cuadro de texto o hacer clic en la flecha situada a la derecha del cuadro y seleccionar un elemento disponible en la lista desplegable. Un control de cuadro combinado es una buena elección para representar un campo clave en una relación de uno a varios, especialmente cuando la lista es corta.
- ❖ **Cuadro de lista.** Permite elegir entre una lista de opciones desplazable. Al contrario que un cuadro combinado, este tipo de control no utiliza una flecha hacia abajo ni permite introducir valores nuevos.

- ❖ **Etiqueta.** Se utiliza para añadir un texto descriptivo a un control o a un formulario. Las etiquetas nunca están vinculadas y no cambian conforme se mueve de un registro a otro. **Access XP** añade automáticamente etiquetas a los campos nuevos que coloca en un formulario; también puedes utilizar etiquetas para títulos e instrucciones.
- ❖ **Botón de opción, Grupo de opciones.** Utiliza estos controles en un formulario cuando quiera que el usuario seleccione exactamente un elemento de entre un grupo. Por ejemplo, en un formulario de factura puede incluir un grupo de opción que contiene los siguientes elementos: *Bebidas, Comidas, Hoteles, Libros*. Cuando añadas el grupo de opción al formulario, el asistente permite especificar el número y formato de cada botón de opción.

Como colocar controles en un formulario

Cuando utilices el Asistente para formularios para crear un formulario, el tipo predeterminado para todos los controles es un cuadro de texto con una etiqueta adjunta a la izquierda. Sin embargo, en algunos casos, puede que quieras que las etiquetas aparezcan sobre el cuadro de texto y quieras cambiar la posición, alineación, tamaño o agrupar los controles de **Access XP**: encontrará fácil colocar los controles precisamente donde los desees. Aún así, requiere algo de práctica.

Access XP muestra los controladores, pequeños rectángulos negros alrededor de la parte externa de un control seleccionado. Estos controladores son una indicación visual de que ha seleccionado el control. Los ocho cuadrados pequeños negros son controladores de tamaño: encontrarás uno en cada esquina y uno en el centro de cada lateral del control seleccionado. Cuando coloques el puntero del ratón sobre un controlador de tamaño, adquiere la forma de una flecha doble, momento en el que puedes arrastra el controlador en cualquier dirección para cambiar el tamaño y forma del propio control.

El gran cuadrado negro situado en la esquina superior izquierda de un control seleccionado permite mover el objeto a una posición nueva dentro del formulario. Cuando arrastras este controlador, el puntero de ratón adquiere la forma de una mano con un dedo señalando hacia arriba; conforme mueves el control seleccionado, muestra sólo su contorno, facilitando la colocación en la cuadrícula del formulario. Cuando sueltes el botón del ratón, el propio control se mueve a la ubicación que se ha seleccionado.

La mayor parte de las veces, cada control que añade un formulario consiste en realidad en dos controles: el control vinculado o el no vinculado (cuadro de texto o cuadro combinado, por ejemplo) y una etiqueta que coincide. Si conoces las técnicas, es fácil colocar estos controles correctamente:

- ❖ Utiliza los cuadros grandes en la esquina superior izquierda de cualquier control o la etiqueta para mover cualquiera de ellos independientemente. Esta técnica es eficaz si deseas mover la etiqueta de la izquierda de un cuadro de texto de modo que se asiente sobre el control.

- ❖ Para mover tanto el control como su etiqueta al mismo tiempo, señala a cualquier borde del control o la etiqueta, hasta que el puntero del ratón adquiere la forma de una mano abierta. Arrastra para colocar la combinación de etiqueta y control en su ubicación nueva.

Para borrar cualquier control del formulario, selecciónalo y pulsa la tecla Supr. Si seleccionas un control, **Access XP** también selecciona la etiqueta; si haces clic en la propia etiqueta, **Access XP** no selecciona el control que coincide.

Por último, cuando trabajes con un formulario en vista *Diseño*, puedes agrupar, distribuir y alinear controles y otros objetos, igual que lo puede hacer en la capa de dibujo de **Office**.

En todos los casos, empieza por seleccionar todos los controles con los que desees trabajar simultáneamente. Para ello, mantén pulsada la tecla Mayús y haz clic en cada selección en sucesión. O utiliza el ratón para trazar un rectángulo alrededor de un grupo de objetos y selecciónalos al mismo tiempo. Supón que haz añadido rápidamente seis campos nuevos a un formulario y ahora desees ordenar la colección:

- ❖ Para distribuir los controles en dos grupos de tres, cada uno de ellos distribuido igualmente, selecciona el primer grupo de controles y elige *Formato, Espacio vertical, Igualar*. A continuación, repite el proceso para el segundo grupo de tres campos.
- ❖ Para alinear cada grupo de campos, selecciona los controles y elige *Formato, Alinear, Izquierda*.
- ❖ Con cada grupo correctamente alineado y espaciado, recoge los tres campos y elige *Formato, Agrupar*. Esta opción bloquea la posición actual de todos los elementos de modo que puedes moverlos como una unidad. Repite lo mismo para el segundo grupo de tres campos.

Ajustar otras propiedades y opciones de formulario

Para formularios de propósito especial, un área potente a explorar es el cuadro de diálogo *Propiedades* del propio formulario. Esta lista contiene ajustes que afectan características importantes del aspecto y comportamiento del formulario. Por ejemplo, en la ficha *Datos* puedes ajustar tres propiedades que determinan si el formulario está disponible sólo como de lectura o si se puede utilizar para editar, borrar y agregar registros.

Cuando creas un formulario desde una sola tabla, las propiedades *Permitir ediciones*, *Permitir eliminación* y *Permitir agregar* se ajustan a *Sí*. Como resultado, el usuario puede añadir, editar y borrar registros utilizando nada más que este formulario. Dependiendo de quién utilice el formulario, puede que desees restringir su capacidad. Cambia el ajuste de propiedad a *No* en uno o más de estas propiedades importantes si desees restringir la capacidad del usuario para modificar los datos.

Presentación de Datos con Informes

Creación de un Informe a Partir de Datos de Access XP

Un informe de **Access XP** organiza datos en un formato adaptado de forma ideal para la impresión. Aunque un formulario es un modo útil de ver datos en pantalla, los informes son, a menudo, el producto final más importante de la base de datos. Cualquiera que sea la audiencia a la que va dirigido el informe (fabricantes de organizaciones, asistentes a la conferencia o la persona en la cabina al otro lado de la entrada), la presentación de datos debería ser clara, concisa y profesional. **Access XP** proporciona un diseño de informe sofisticado que puede ayudarte a agrupar, resumir y disponer los datos de forma eficaz.

Debes tomar varias decisiones a la hora de diseñar un informe de **Access XP**. ¿Deberías contener una gran cantidad de datos detallados, incluyendo muchos campos individuales y registros? ¿Cómo deberían agruparse y clasificarse los registros individuales? ¿Necesita un resumen de los datos en el formulario, totales, promedios o gráficos? ¿Debería basarse el informe en una sola tabla, o en alguna combinación de tablas o consultas relacionadas? ¿Qué formato de presentación será mejor para resaltar el significado de los datos?

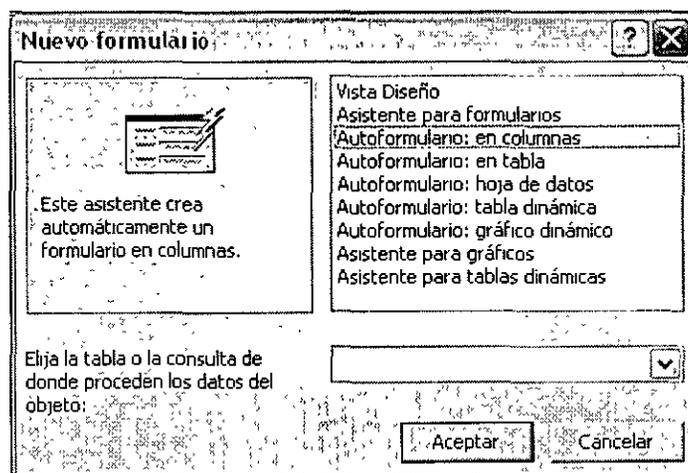
Como trabajar con la vista diseño

Las herramientas que utilices para diseñar los informes de **Access XP** tienen un parecido increíble con las que utilizas con los formularios. En ambos casos, debes especificar un origen de datos (una o más tablas o consultas, o una sentencia escrita en SQL) y los controles de posición en una cuadrícula de diseño. Aunque puedes crear un informe partiendo de cero en la vista *Diseño*, el asistente para informes es casi siempre el mejor punto de partida. Después de utilizar el asistente para crear la estructura básica de un informe, puedes a continuación abrir el informe en la vista *Diseño* para hacer cambios detallados del contenido y aspecto.

Los informes de **Access XP** están organizados en dos secciones horizontales configuradas en un orden específico. (Si haz trabajado con otros programas de bases de datos, puede que estés familiarizado con el concepto de crear un informe mediante bandas: las secciones de **Access XP** son las mismas que las bandas en otros programas de base de datos) Entender cómo funciona cada sección es un paso crucial en el diseño de un informe eficaz.

- ❖ **El Encabezado o pie de página del informe** aparece al principio y al final del informe. Una encabezado de informe, a menudo, incluye el título del informe y el control calculado que contiene la expresión =Fecha() para mostrar los datos en los que se ha impreso la copia del informe. Los pies de página del informe suelen contener muchos totales o promedios para los datos expuestos.

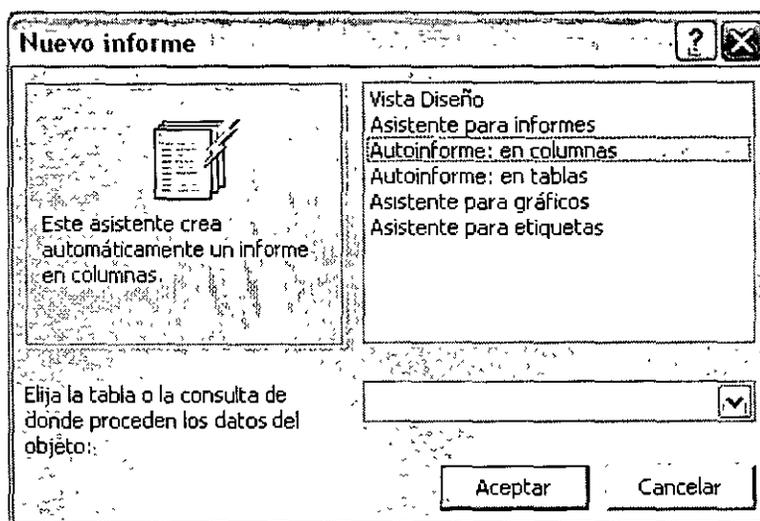
- ❖ El **Encabezado y pie de página** aparece en la parte superior e inferior de cada página, incluso si la sección *Detalle* es una continuación de los datos de la página anterior. Los encabezados de página se utilizan normalmente para los encabezados de columnas, para que los lectores puedan seguir una larga lista, y los pies de página son útiles para las fechas y los números de página.
- ❖ Las secciones **Encabezado del grupo y Encabezado de pie** aparecen automáticamente cuando define las opciones de ordenación y agrupamiento para un informe. Colocando los campos en cualquiera de estas secciones, puedes ver resúmenes de los datos dentro de cada grupo.
- ❖ La **sección Detalle** incluye campos de cada registro del origen de datos. Cada campo en la sección *Detalle* aparece una vez para cada registro del origen de datos, convirtiéndolo en el lugar adecuado para especificar cómo desea que aparezca una lista.



- ❖ Como en los formularios de **Access XP**, los controles pueden tomar una de tres formas: ligados a un campo en una tabla o consulta; calculados, utilizando una expresión compuesta de cualquier combinación de funciones de **Access XP** y referencias de campo; o no ligados, como es el caso de las líneas, cuadros y etiquetas de texto. Utiliza los cuadros de diálogo *Lista de campos* y *Cuadro de herramientas* para añadir nuevos controles ligados a un formulario. Los controles tienen sus propios ajustes de propiedad, como lo hacen las secciones individuales del informe: cambiando los valores de estas propiedades puedes modificar el aspecto y contenido del propio informe.

Cómo crear un informe básico

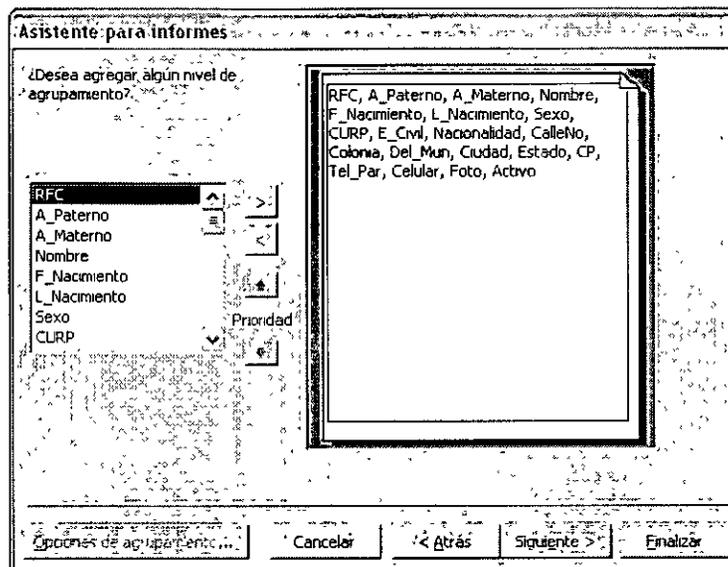
Access XP proporciona una herramienta única, *Autoinforme*, para generar el informe más simple posible con un solo clic. Si pulsas *F11* para cambiar a la ventana de base de datos y a continuación elige *Insertar, Informe, Access XP* muestra el cuadro de diálogo *Nuevo informe*. Elige cualquiera de las opciones *Autoinforme*, especifica una tabla o consulta como origen de datos y haz clic en *Aceptar* para producir un informe instantáneo.



La salida de las opciones de *Autoinforme* consiste en todos los campos y todos los registros de una sola tabla o consulta sin consultas de agrupamiento o resumen. Aunque esta opción es de una rapidez innegable, muy rara vez produce resultados satisfactorios. Es más eficaz cuando haz definido una consulta con un número pequeño de campos (no más de cuatro) y deseas ver todos los registros de esa consulta en un formato de lista básico.

Una opción mucho mejor para iniciarse con un informe es el *Asistente para informes*. Abre el cuadro de diálogo *Nuevo informe*, selecciona una tabla o consulta y haz clic en *Aceptar*. A medida que recorras los pasos del *Asistente para informes*, puedes añadir tablas o consultas al informe además de aquél con el que haz empezado. Entre esas fuentes de registro, puedes seleccionar campos individuales y colocarlos en el orden que deseas verlos en el informe final.

Agrupar opciones disponibles en el *Asistente para informes* es ligeramente diferente, dependiendo de los campos que hayas seleccionado. Si todos los campos son de una sola columna o consulta, puedes elegir uno o más campos para *Agrupar*; si haz seleccionado campos de dos o más tablas relacionadas, por otro lado, verás un cuadro de diálogo que permite designar qué tablas utilizar para el agrupamiento.



El botón *Opciones* de resumen permite elegir hasta cuatro cálculos estadísticos para cada campo numérico en la sección *Detalle del informe*. La opción *Suma* calcula el total de todos los valores *numéricos*; *Promedio* muestra la media de los valores; y *Mínimo* y *Máximo* buscan los valores más pequeños y más grandes.

Los pasos finales del asistente permiten elegir opciones de diseño y ordenación. El último caso, puede ver una vista previa de las opciones elegidas antes de tomar la decisión. También puedes elegir entre las orientaciones *Horizontal* y *Vertical*, una opción útil, la primera, para los informes que contienen muchas columnas, por último, el asistente permite introducir un título nuevo para el informe. Haz clic en *Finalizar* para guardar el objeto de informe.

En este punto, puedes ver antes de *Imprimir* el resultado o cambiar inmediatamente a la vista *Diseño* para personalizarlo todavía más. A no ser que el informe sea extremadamente simple, probablemente desearás elegir la segunda opción.

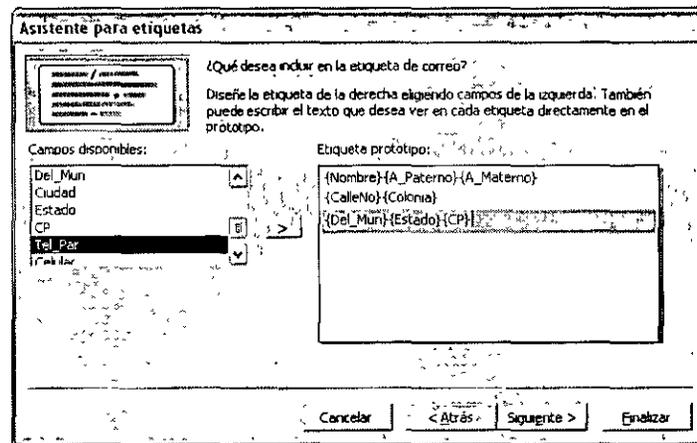
Como convertir los datos de Access XP a etiquetas

Un uso común de las tablas de base de datos es guardar y administrar las listas, nombres, direcciones de clientes y usuarios. Si creas una lista en **Access XP**, puedes *Imprimirla* en un formato estándar de etiqueta con la ayuda del asistente para etiquetas. El asistente para etiquetas realiza el trabajo sucio de la creación de columnas, ajustando el tamaño y posición de los cuadros de texto, e insertando campos de texto en el diseño de informe de modo que se puedan *Imprimir* en el formato adecuado.

Para iniciar el asistente para etiquetas, pulsa *F11* para cambiar a la ventana base de datos y a continuación elige *Insertar, Informe*. Elige el *Asistente para etiquetas* del cuadro de diálogo *Nuevo informe* y sigue las solicitudes para definir un origen de datos, selecciona los campos a utilizar para el informe y elegir un formato de etiqueta para nombre de fabricantes y código de producto.

Sigue las solicitudes del *Asistente para seleccionar* una tabla o consulta, elige el número de fabricante o producto de la hoja de etiqueta sobre la que piensas *Imprimir* las direcciones y especifica una fuente y color predeterminados para el texto de la etiqueta. Después de trabajar con estos preliminares, llegas al paso donde puedes especificar los campos que deseas que aparezcan en cada etiqueta. Ten en cuenta que también puedes añadir texto y signos de puntuación: puedes separar los campos de ciudad y destinatario con una coma, por ejemplo.

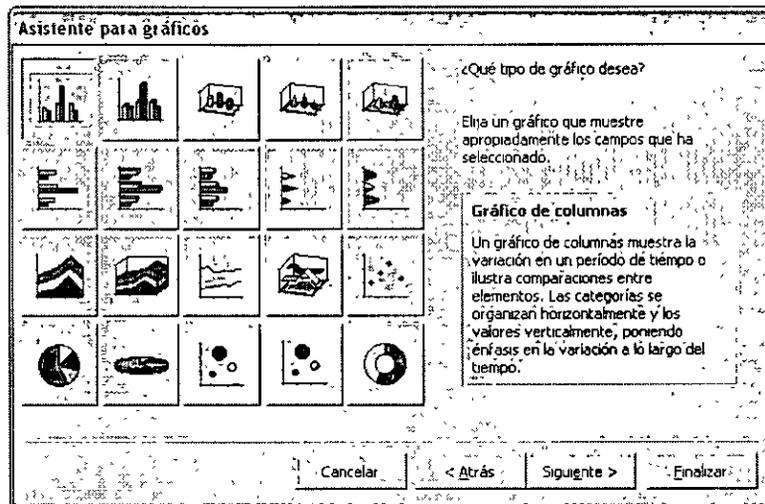
Los pasos finales del *Asistente para etiquetas* permiten elegir un tipo de ordenación para las etiquetas impresas y proporcionar un nombre para el objeto de informe que el asistente genera. Cuando hagas clic en el botón *Finalizar*, **Access XP** crea el informe. Examina las etiquetas para asegurarte que el formato es correcto. A continuación, haz clic en el botón *Imprimir* de la barra de herramientas de **Access XP** para *Imprimir* las etiquetas.



Creación de gráficos

Una de las opciones del cuadro de diálogo *Nuevo informe* es que permite crear un gráfico basado en los datos de **Access XP**. El *Asistente para gráficos* utiliza **Microsoft Graph 2000**, para generar cualquiera de los 20 tipos de organigramas y gráficos a partir de los datos numéricos seleccionados en una base de datos y para vincular el objeto resultante a los campos en un origen de datos.

El *Asistente para gráficos* funciona bastante mejor de lo que se dice: en una sucesión de cuadros de diálogo, especifica la tabla o consulta a utilizar como origen de datos, selecciona los campos a utilizar en el gráfico y elige un tipo de gráfico. Después de abrirse difícilmente el camino entre estos pasos previos. Los detalles exactos varían, dependiendo del tipo de gráfico que hayas elegido, pero el principio es el mismo: arrastra los campos elegidos de la lista situada a la derecha, colócalos en el diseño del gráfico y a continuación haz clic en el icono *Vista previa del gráfico*, situado en la esquina superior izquierda, para ver el aspecto del gráfico.



En la ventana final del *Asistente para gráficos*, puedes introducir un título, cuando hagas clic en *Finalizar*, **Access XP** genera un objeto de informe nuevo y muestra una vista previa del gráfico.

Sin embargo, por el simple hecho de que puedes crear un gráfico en **Access XP**, no significa que debas hacerlo. Modificar un gráfico suele ser una tarea difícil; en la mayoría de los casos, es más fácil borrar un gráfico existente y crear uno nuevo, que intentar cambiar los vínculos entre los datos y el objeto gráfico. **Microsoft Graph 2000**, el motor que subyace tras el *Asistente para gráficos*, es muy poco potente, comparado con **Excel**. Siempre que sea posible, te recomiendo que exportes los datos a **Excel** y utiliza sus capacidades gráficas. Todavía mejor, utiliza **Microsoft Query** dentro de **Excel** para abrir un vínculo en directo a una base de datos de **Access XP** y crea el gráfico basado en esos datos.

Modificación de un Informe

Como suele ser el caso a partir de la salida del *Asistente para informes*, el informe resultante necesita un poco de orden. Para los principiantes, puede que quieran reemplazar algunos de los encabezados de columna con texto que es más legible y descriptivo. También puedes borrar con seguridad la línea de texto que empieza con resumen para el final de cada grupo; realmente no añade ninguna información útil a este informe. Por último, desearás ajustar los tamaños y Propiedades de algunos de los campos numéricos para asegurarte que están completamente visibles y consistentemente formateados.

Cómo agrupar y ordenar registros en un informe

En las presentaciones complejas de datos **Access XP**, los grupos son la esencia de un diseño de informe. Un grupo define cómo se organizan los registros en la producción de un informe y cómo se puede resumir la información con cálculos estadísticos, como, por ejemplo, totales y promedios. En la vista *Diseño* de un informe, los grupos están representados por una jerarquía de secciones *Encabezado*, *Detalle* y *Pie de página*.

Al utilizar el *Asistente para informes*, puedes elegir los campos que quieras utilizar para crear un grupo, y puedes especificar como deseas que se resuman los grupos por cálculos específicos. Sin embargo, para ajustar las opciones de agrupamiento y ordenación, debes pasar a la vista *Diseño* y hacer clic en el botón *Ordenar y Agrupar*. Esta opción presenta un cuadro de diálogo que permite controlar, virtualmente, todas las opciones de agrupamiento.

Por ejemplo, en un informe que pienses utilizar como un catálogo de producto, puede que quieras Agrupar registros por *IdPedido*, en orden alfabético, con los nombres de producto enumerados bajo cada nombre de pedido, también en orden alfabético. En este ejemplo, incluirás el nombre del proveedor y los campos *UnidadesEnAlmacen* en la sección *Detalle*; de modo opcional, puedes añadir un campo calculado en la sección *Encabezado del grupo*, utilizando la expresión `=SUMA([PrecioProveedor]*[UnidadesEnAlmacén])`.

Para indicar a **Access XP** que deseas ordenar según *IdPedido* y luego por *IdProducto*, abre el cuadro de diálogo *Agrupar* y utiliza las flechas desplegables para seleccionar el orden *Ascendente*; si deseas utilizar el orden *Descendente*, haz el ajuste aquí.

Eligiendo estos dos campos, indica a **Access XP** que deseas ordenar los registros en este orden, pero necesitas dar un paso extra para agrupar registros: en el cuadro de diálogo *Ordenar y Agrupar*, selecciona la fila que contiene el campo *IdPedido* y cambia la propiedad *Encabezado del grupo* a *Sí*. Si deseas añadir un pie de página para esta sección, cambia también la propiedad *Pie del grupo* a *Sí*.

Como agregar y modificar encabezados y pies de página

El *Asistente para informes* hace generalmente un buen trabajo de disposición de campos en la cuadrícula del informe, pero normalmente hace un trabajo pésimo a la hora de organizar encabezados y pies de página. En particular, normalmente desearás hacer los ajustes siguientes:

- ❖ **Encabezado del informe.** Generalmente esto no contiene más de un título. Puede que quieras añadir un control calculado con la expresión *=Fecha()* para mostrar la fecha en la que se preparó el informe. Para ocultar la sección *Encabezado del informe*, cambia su propiedad visible a No.
- ❖ **Encabezado de página.** Para informes en tablas, con muchas filas de datos dispuestos en formato de fila y columna, considera la modificación de los encabezados de columna como parte del encabezado de página. El informe resultante es mucho más fácil de leer.
- ❖ **Encabezado del grupo.** Si haz agrupado datos utilizando un campo que contiene información de fecha, utiliza la opción *Agrupar en*, para administrarla a intervalos: mes, trimestre o año, por ejemplo. Combinando este encabezado con otras agrupaciones, puedes ver una lista de todas las ventas mensuales por cliente, incluso si la fecha aparece sólo por día. Además, **Access XP** normalmente hace un trabajo pésimo de alineación de encabezados. Puede que tengas que cambiar el texto o formato en una etiqueta, por motivos de legibilidad.
- ❖ **Detalle.** Observa los datos antes de *Imprimirlos* cuidadosamente; son muchas las posibilidades de que los campos sean demasiado estrechos o demasiado anchos para los datos que contiene.
- ❖ **Pie de página de grupo.** Si deseas empezar una página nueva para cada agrupamiento, ajusta la propiedad *Forzar nueva página* a *Después de la sección*. Si esta sección no está visible, ajusta en su lugar esta propiedad para la sección *Detalle*.
- ❖ **Pie de página.** Normalmente, el *Asistente para informes* añade un sello de fecha, hora y un contador de página aquí. Para mostrar sólo el sello de fecha, cambia la expresión *=Hora()* a *=Fecha()*; para ocultar esta sección completamente, cambia su propiedad *Visible* a No.
- ❖ **Encabezado del informe.** Cuando añadas un *Encabezado del informe*, **Access XP** añade automáticamente esta sección, pero por defecto está ajustada a 0: en otras palabras, no está visible, si deseas resumir los detalles en el informe o añadir responsabilidades o notas de pie, hazlo aquí: ajusta la propiedad *Forzar nueva página* a *Antes de la sección* y a continuación añade los campos de resumen que quieras ver en el informe.

Como trabajar con cálculos en un informe

El uso más común de las secciones de pie de página en un informe es resumir datos. Cuando utilizas el *Asistente para informes* y eliges *Opciones de resumen*, **Access XP** añade aquí fórmulas de resumen automáticamente. Los dos problemas más comunes con las fórmulas de resumen son un signo igual que falta y un formato incorrecto:

- ❖ Si introduces una expresión directamente en un control calculado, asegúrate de agregarle un signo igual como prefijo. En caso contrario, **Access XP** no será capaz de interpretar correctamente la expresión y mostrará un mensaje de error o un cuadro de diálogo de parámetros.
- ❖ En el caso de datos formateados, un problema común es cuando aparecen valores de moneda como valores decimales no formateados. Para corregir este problema, haz clic en el control calculado y observa el cuadro de diálogo *Propiedades*; a continuación *Formato* para corregir los valores.

Cómo Exportar un Informe a otro Formato de Datos

Un modo de hacer uso de un informe que haz creado es *Imprimirlo* y distribuir copias a cualquiera que necesite la información que contiene el informe. O puedes *Exportar* un objeto de informe a otros entornos **Software**, incluyendo **Microsoft Word** o **Microsoft Excel**, donde puedes procesar el informe de formas que no es conveniente o incluso posible hacerlo en **Access XP**. Si deseas que otras personas puedan ver la salida exactamente como la haz preparado, pero no está seguro de que tengan **Access XP**, intenta guardar el informe en el **Formato Snapshot** o como una página **Web**.

Cómo guardar un informe en un formato de imagen instantánea

La aplicación **Snapshot Viewer** debutó en el primer paquete de servicio para **Office 97**, pero la mayor parte de los usuarios de **Access XP** todavía no saben que existe. Los archivos de imagen instantánea captura la presentación de datos de un informe de **Access XP** en un archivo notablemente compacto, de modo que cualquiera puede ver o **Imprimir** el informe. Sin embargo, no permiten modificarlo. La gran ventaja de los archivos **Snapshot** es que no necesitan que su audiencia tenga una copia de **Access XP** instalada para ver un informe.

Para guardar un informe en formato **Snapshot**, abre el informe y elige *Archivo, Exportar*. Introduce un nombre de archivo, elige una ubicación de almacenamiento y en el cuadro *Guardar como tipo*, elige **Formato Snapshot**. **Access XP** guarda el *archivo*, que puedes enviar a continuación a la **Web**, guardar un servidor compartido o enviarlo por correo electrónico. Los posibles receptores necesitan descargar el **Access XP Snapshot Viewer** del sitio **Web** de **Office Update** de **Microsoft** (<http://officeupdate.microsoft.com>); cuando lo hagan, podrán abrir el archivo sin ningún problema.

Guardar un informe como página Web

Si el **Snapshot Viewer** supone demasiado trabajo, considera guardar un informe de **Access XP** en el *formato HTML*, de modo que cualquiera pueda verlo directamente en un explorador **Web**. Abre el informe y elige *Archivo, Exportar*, pero esta vez elige documentos **HTML** en el cuadro *Guardar como tipo*. Ten en cuenta que **Access XP** guarda cada página de informe como un documento **HTML** independientemente, con un gran conjunto de controles de navegación rudimentarios en la parte inferior de cada página que permite al lector ir a la primera, a la última o a la página siguiente.

Exportar un informe a Microsoft Word

Cuando abras un informe, **Access XP** muestra un conjunto de botones **Vínculos de Office** en la barra de herramientas y en el menú *Herramientas*. Si estás tentado de probar el botón *Publicarlo con MS Word*, no te hagas muchas ilusiones. Haciendo clic en este botón hace que **Access XP** cree un archivo **RTF** (formato de texto enriquecido) y lo abra inmediatamente en **Microsoft Word**. Sin embargo, en general, el formato no será fiel al original y probablemente tendrás que modificarlo bastante para que el resultado sea funcional.

Esta opción es más útil cuando deseas incorporar algunos o todos los informes a otro documento. Utiliza el Portapapeles para cortar y pegar las secciones del informe conforme las necesite.

Analizar un informe con Excel

Puedes seguir con la misma secuencia de pasos para *Exportar* un informe **Access XP** a **Excel** y los resultados pueden ser sorprendentemente eficaces. Selecciona el nombre del informe en la ventana de base de datos y elige una versión apropiada de **Excel** de la lista desplegable *Guardar como tipo*. **Access XP** crea un archivo **XLS** nuevo y lo abre directamente en **Excel**. Cualquier agrupamiento en el informe general se traduce en esquematización de la hoja de trabajo **Excel**. Después de que los datos estén disponibles en una hoja de trabajo, puede analizar o crear un gráfico con relativa facilidad.

Anexo B

Expresiones en Access XP

Una expresión es una instrucción de intenciones. Si deseas que se produzca una acción después de satisfacer una condición específica, la expresión deberá especificar esa condición. Para seleccionar los registros de una consulta que contengan, por ejemplo, valores **CódPostal** de **90000** o superior, tendrás que usar la expresión:

CÓDPOSTAL >= 90000

También puedes usar expresiones en cálculos aritméticos. Si necesitas un campo **CantidadExtendida** en una consulta, por ejemplo, tendrás que usar:

CantidadExtendida: Cantidad * PrecioUnidad

Como expresión para crear valores calculados en las celdas de datos de la columna **CantidadExtendida**.

Para ser calificado como expresión, una instrucción deberá tener por lo menos un operador y un literal, identificador o función. La lista siguiente describe estos elementos:

Los **Operadores** incluyen los conocidos símbolos aritméticos + [nd], * (multiplicación) y / (división), aparte de muchos símbolos de abreviaturas. En algunos operadores son específicos de **Access XP** o **SQL**, como los operadores **Between**, **In**, **Is** y **Like**.

Los **Literales** están formados por valores que se escriben, como 12345 o **ABCDE**. Los literales se utilizan casi siempre para crear valores predeterminados y, en combinación con los identificadores de campo, para comparar valores en campos de tabla y columnas de consulta.

Los **identificadores** son los nombres de los objetos de Access (como los campos de las tablas) que devuelven valores numéricos o de texto distintos. En término devolver, cuando se emplea con expresiones, significa que el valor presente del identificador se sustituye por su nombre en la expresión. Por ejemplo, el identificador de nombre de campo **NombreCompañía** de una expresión devuelve el valor (nombre de una empresa) del campo **NombreCompañía** del registro seleccionado en ese momento. **Access XP** posee cinco constantes predefinidas provistas de nombre que también sirven como identificadores: **True**, **False**, **Yes**, **No** y **Null**. Las constantes y variables provistas de nombre que se crean en Access **VBA** también son identificadores.

Las **funciones** devuelven un valor en lugar del nombre de función de la expresión, como las funciones **Date** y **Format**. A diferencia de los identificadores, casi todas las funciones requieren que proporcione paréntesis a un identificador o valor como argumento.

Cuando los literales, identificadores o funciones se utilizan con operadores, estas combinaciones se llaman **operandos**.

Operadores

Access XP y **VBA** proporcionan seis categorías de operadores que se pueden usar para crear expresiones.

- Los operadores aritméticos ejecutan la suma, resta, multiplicación y división.
- Los operadores **de asignación** y **comprobación** establecen valores y los comparan.
- Los operadores **lógicos** tratan con los valores, que sólo se pueden ser verdaderos o falsos.
- Los operadores de **concatenación** combinan cadenas de caracteres.
- Los operadores de **identificación** crean nombres no ambiguos para objetos de base de datos, de forma que es posible asignar el mismo nombre de campo, por ejemplo, en varias tablas y consultas.

Otros operadores, como **Like**, **Is** y **Between**, simplifican la creación de expresiones para seleccionar registros con consultas.

Los operadores de las cuatro primeras categorías están disponibles en casi todos los lenguajes de programación. Los operadores de identificación son específicos de **Access XP**, mientras que los demás operadores de la última categoría sólo se ofrecen en sistemas de administración de base de datos relacionales (URDBMS) que crean consultas basadas en el Lenguaje Normalizado de Consultas (SQL) o en un lenguaje de consultan patentado.

Operadores aritméticos

Los operadores aritméticos sólo funcionan con valores numéricos y deben tener dos operandos numéricos, con las expresiones siguientes:

Cuando el signo ([**-**]) cambie el signo (niegue el valor) de un operando. En este caso, el signo menos se denomina el menos **unario**.

Cuando el signo (**=**) asigne un valor al objeto de **Access XP** o a un identificador de variable de **Access VBA**.

La siguiente tabla enumera los operadores aritméticos que se pueden usar en las expresiones de **Access XP**.

Operador	Descripción	Ejemplo
+	Agrega dos operandos	Subtotal + Impuesto
[-]	Resta dos operandos	Fecha - 30
[-] (unario)	Cambia el signo de un operando	-12345
*	Multiplica dos operandos	Unidades * PrecioUnidad
/	Divide un operando por otro	Cantidad / 12.55
\	Divide un operando de entero por otro	Unidades \ 2
Mod	Devuelve lo que queda de una división por un entero	Unidades Mod 12
^	Eleva a un operando a una potencia (exponente)	Valor ^ Exponente

Operador	Descripción
\	El símbolo de división de entero es el equivalente de <<entrar en>>, como se usa en las clases de matemática elemental de la escuela: tres entra en 13 cuatro veces, y sobra uno. Cuando se usa la división de entero, los operadores que tienen fracciones decimales se redondean a enteros, pero cualquiera de las funciones decimales del resultado queda truncado.
Mod	Una abreviatura de módulo , este operador devuelve el valor resto de la división de entero. Por tanto, 13 Mod 4, por ejemplo, devuelve 1.
^	El operador de exponenciación eleva el primer operando a la potencia del segundo. Por ejemplo 2 ^ 4, o dos a la cuarta potencia, devuelve 14 (2*2*2*2)

Estos tres operadores se usan muy pocas veces en aplicaciones de negocios, pero a menudo se utilizan en el código de programación de **Access VBA**.

Operadores de asignación y comparación

En la tabla anterior omití el signo igual, que va asociado a las expresiones aritméticas puesto que en **Access XP** se emplea de dos formas (ninguna de la cual recae bajo la categoría aritmética) El uso más normal del signo igual es un operador de asignación: = asigna el valor del operando simple a un objeto de **Access XP** o una variable o constante. Cuando se usa la expresión = <<Q>> para asignar un valor predeterminado a un campo, el signo igual actúa como un valor de asignación. Si no = será un operador de comparación que determinará si uno de los dos operandos es igual al otro.

Los operadores de **comparación** comparan los valores de dos operandos y devuelven los valores lógicos (**True** o **False**) dependiendo de la relación que hay entre los dos operandos y el operador. Una excepción se da cuando uno de los operandos posee el valor **Null**. En este caso, toda comparación devuelve un valor de **Null**. Dado que **Null** representa un valor desconocido, no se puede comparar un valor desconocido y llegar a una conclusión **True** o **False** válida.

La siguiente tabla enumera los operadores de comparación disponibles en **Access XP**.

Operador	Descripción	Ejemplo	Resultado
<	Menor que	123 < 1000	True
<=	Menor o igual que	15 <= 15	True
=	Igual	2 = 4	False
>=	Mayor o igual que	1234 >= 456	True
>	Mayor que	123 > 123	False
<>	Distinto	123 <> 456	True

Los usos principales de los operadores de comparación consisten en crear reglas de validación, con el fin de establecer criterios para seleccionar registros en las consultas, para determinar las acciones llevadas a cabo por las macros, para crear combinaciones por medio de la cláusula **SQL WHERE** y para controlar el flujo de programas en **Access VBA**.

Operadores lógicos

Los **operadores lógicos** (también llamados **operadores booleanos**) se emplean para cambiar los resultados de dos no más expresiones de comparación en un solo resultado. Los operadores lógicos solamente pueden combinar expresiones que devuelvan los valores lógicos **True**, **False** o **Null**. Con la excepción de **Not**, que es equivalente lógico del menos unario, los operadores lógicos siempre requieren dos operandos.

La siguiente tabla enumera los operadores lógicos de **Access XP**.

Operador	Descripción	Ejemplo 1, Ejemplo 2	Resultado 1, Resultado 2
And	Y lógico	True And TRue, True And False	True, False
Or	O inclusivo	True Or False, False Or False	True, False
Not	No lógico	Not True, Not False	True, False
Xor	O exclusivo	True Xor False, True Xor True	True, False

Los operadores lógicos **And**, **Or** y **Not** se usan mucho en las expresiones de **Access XP** y en las instrucciones **SQL**; en las instrucciones **SQL** estos operadores van en mayúscula, como **AND**, **OR**, y **NOT**. **Xor** se usa muy poco en **Access XP**. **eqv.** (equivalente) e **imp.** (implicación) se ven muy poco, incluso en código de programación.

Operadores de concatenación

Los **operadores de concatenación** cambian dos valores de texto en una sola cadena de caracteres, si, por ejemplo se concatena ABC a DEF, el resultado será ABCDEF. El signo & es el operador de concatenación preferido de **Access XP**.

Operadores identificadores

Los operadores identificadores! (el signo de exclamación) y . (El operador **Punto** de **Access XP**) son separadores y realizan las siguientes operaciones:

Combinar los nombres de las clases de objetos y nombres de objetos con el fin de seleccionar un objeto específico o propiedad de un objeto. Por ejemplo, la siguiente expresión identifica el formulario de Acciones de Personal:

Forms!AccionesdePersonal

Esta identificación es necesaria porque es posible que también tengas una tabla llamada AccionesdePersonal.

Distingue los nombres de objetos de los nombres de propiedades. Considera la siguiente expresión:

TextBox1.FontSize = 8

Identifica campos específicos de las tablas, como en la siguiente expresión que especifica el campo Nombre de Compañía de la tabla Clientes:

Clientes!NombreCompañía

El carácter ! Se utiliza para separar referencias de objetos; la sintaxis general es NombreClase!NombreObjeto. El carácter . Separa los objetos y sus propiedades o métodos, como en ClaseObjeto!Objeto.Propiedad o ClaseObjeto!NombreObjeto.Método.

Otros operadores

Los operadores restantes de **Access XP** están relacionados con los operadores de comparación. Estos operadores devuelven **True** o **False**, dependiendo de si el valor de un campo cumple con la especificación elegida del operador. Un valor **True** hace que se incluya un registro en una consulta, mientras que un valor **False** rechaza el registro. Cuando se usan estos operadores en reglas de validación, las entradas se aceptarán o rechazarán en base al valor lógico devuelto por la expresión.

La siguiente tabla enumera los cuatro operadores restantes que se usan en las consultas y reglas de validación de **Access XP**.

Operador	Descripción	Ejemplo
Is	Se usa con Null para determinar si un valor es Null o Not Null	Is Null, Is Not Null
Like	Determina si un valor de cadena empieza por uno o más caracteres (para que Like funcione correctamente ha que añadir un comodín *, o uno o más signos?)	Like <<Jon*AA>> Like <<FILE????>>
In	Determina si un valor de cadena es un integrante de una lista de valores	In(<<CA>>, <<OR>>, <<WA>>)
Between	Determina si un valor numérico o de fecha está dentro de un rango especificado de valores	Between 1 And 5

Los caracteres comodín * y ? se usan con el operador Like de la misma forma que se utilizan en DOS. El carácter * (asterisco o estrella) ocupa el lugar de cualquier cantidad de caracteres. El signo ? Ocupa el lugar de un solo carácter. Por ejemplo, Like <<Jon*>> devuelve True en valores como **Jonás** o **Jonathan**. Like <<*on*>> devuelve True en cualquier valor que contenga On. Like <<FILE????>> devuelve True para **Filename** pero no para **File000** o **FileNumber**. Los caracteres comodín pueden preceder a los caracteres que deseas que coincidan, como en Like <<*son>> o Like <<????Name>>.

A excepción de IS, los operadores de esta otra categoría son equivalentes a las palabras reservadas SQL LIKE, IN y BETWEEN. Access XP incluye estos operadores para promocionar la compatibilidad de SQL. Puede crear cada uno de estos operadores mediante la combinación de otros operadores o funciones de Access XP. Like <<Jon*>> equivalente de Access XP InStr (Left (NombreCampo, 3), <<Jon>>) en VBA. In (<<CA>>, <<OR>>, <<WA>>) es similar a InStr (<<CAORWA>>, NombreCampo) a excepción de que no se producen coincidencias para los ambiguos AO y RW. Between 1 And 5 es el equivalente de >= 1 And <= 5.

Literales

Access XP proporciona tres tipos de literales, que se pueden combinar con operadores para crear expresiones. La lista siguiente describe estos tipos de literales:

Los literales **numéricos** se escriben como una serie de cifras, entre las que se incluyen el signo aritmético y la coma decimal, cuando se aplique. No es necesario anteponer números positivos con el signo más, ya que **Access XP** asume que los valores son positivos a menos que el signo menos esté presente. Los literales numéricos pueden incluir E o e y el signo de exponente para indicar un exponente en notación científica, por ejemplo, -1.23E-02.

Literales de **Texto** o de cadena pueden incluir cualquier carácter imprimible, más los caracteres imprimibles que devuelve la función chr. La función chr devuelve los caracteres especificados por un valor numérico de la tabla de caracteres ANSI (similar a la tabla de caracteres ACSII) que emplea Windows. Por ejemplo, Chr (9) devuelve el carácter Tab. Las expresiones de **Access XP** requieren que se coloquen los literales de cadena entre comillas dobles (<<>>). Las combinaciones de caracteres imprimibles y no imprimibles se concatenen con el signo &. Por ejemplo la expresión siguiente separa dos cadenas con un par de nueva línea.

<<Primera línea>> & Chr(13) & Chr(10) & <<Segunda línea>>

Chr (13) es el retorno de carro (CR), y **chr(10)** es el carácter de avance de línea (LF); juntos forman el par de nueva línea.

Cuando se introducen literales de cadena en las celdas de las tablas y cuadrículas de diseño de la consulta, **Access XP** agregará las comillas automáticamente. En otros sitios tendrás que introducir las comillas.

Los literales **Fecha/Hora** se encierran entre signos de número (#), como en las expresiones #1-Ene-80# o #10:20:30#. **Access XP** agrega estos signos si el programa detecta que está escribiendo en una cuadrícula de diseño una fecha u hora en uno de los formatos estándar Fecha/Hora de **Access XP**.

Identificadores

Un **identificador** es por regla general el nombre de un objeto; las bases de datos, las tablas, los campos, las consultas y los informes son objetos de **Access XP**. Cada objeto tiene un nombre que identifica únicamente ese objeto. A veces, para identificar un subobjeto, un nombre de identificador está formado por un **nombre de familia** (clase de objeto) separado de un **nombre dado** (nombre de objeto) por un signo de exclamación o un punto (un operador identificador). El nombre de familia del identificador viene primero, seguido por el separador y el nombre dado. **SQL** emplea un punto como separador de objetos. Un ejemplo de identificador es una instrucción **SQL** como se ve:

Cientes.Dirección

En este ejemplo, el identificador del objeto de campo Dirección está dentro del objeto de la tabla Cientes. **Cientes** es el nombre de familia del objeto (la tabla), mientras que **Dirección** es el nombre dado del subobjeto (el campo). Sin embargo, en **Access XP** se usa el símbolo ! para separar los nombres de tabla y nombres de campo (el punto separa los objetos y sus propiedades). Si un identificador contiene un espacio u otro signo de puntuación, rodea el identificador con comillas, como en este ejemplo:

[Detalles Pedido]!Cantidad

No es posible incluir puntos o signos de exclamación dentro de los nombres o identificadores; [Unidad!Precio], por ejemplo, no se permite.

En estas consultas sencillas que sólo utilizan una tabla, es posible omitir el prefijo **NombreTabla**. Los identificadores se utilizan para devolver los valores de los campos de los objetos formulario e informe.

Funciones

Las funciones devuelven valores a sus nombres. Las funciones pueden ocupar el lugar de los identificadores de las expresiones. Una de las funciones más comunes que se usan en las expresiones de **Access XP** es **Now** como propiedad Valor Predeterminado de un campo Fecha/Hora de una tabla. Por ejemplo, 15/3/99 9:00 aparecerá en el campo cuando se cambie a la Vista hoja de datos (las 9:00 A.M. del 15 de marzo de 1999).

Access XP y **VBA** definen cerca de 150 funciones individuales. La lista siguiente agrupa las funciones por finalidad:

Las funciones **Fecha/Hora** manipulan valores de fecha y hora de los campos o valores Fecha/Hora que se introducen como literales. Es posible extraer partes de las fechas y partes de las horas con funciones de fecha y hora.

Las funciones de **manipulación de texto** se utilizan para trabajar con cadenas de caracteres.

Las funciones de **conversión de tipo de datos** permiten especificar el tipo de datos de los valores de los campos numéricos, en vez de depender de **Access XP** para escoger el tipo de datos más apropiado.

Las funciones **matemáticas** y **trigonométricas** se ejecutan en operaciones de valores numéricos que están más allá de la capacidad de los operadores aritméticos estándar de **Access XP**. Las funciones trigonométricas sencillas se pueden utilizar, por ejemplo, para calcular la longitud de los lados de un triángulo.

Las funciones **financieras** se parecen a las funciones que proporcionan **Lotus 1-2-3** y **Microsoft Excel**. Calculan la depreciación, los valores de las pensiones y los tipos de interés. Para determinar el valor presente de un premio de lotería pagado en 25 cuotas anuales iguales, por ejemplo, podrás usar la función PV.

Las funciones de **propósito general** no se adaptan a ninguna de las clasificaciones anteriores; estas funciones se emplean para crear consultas, formularios e informes de **Access XP**.

Otras funciones son aquellas que ejecutan el intercambio dinámico de datos (DDE) con otras aplicaciones de **Windows**, funciones agregadas **SQL** y funciones que se usan principalmente en la programación de **Access VBA**.

En las consultas de **Access XP** se suelen usar solamente los tres primeros grupos de funciones.

Uso de la ventana Inmediato

Cuando se escribe código de programación **VBA** en un módulo, la ventana Inmediato estará disponible para ayudarte a depurar el código. También puedes usar la ventana Inmediato del módulo para demostrar el uso y sintaxis de las funciones.

Para experimentar con algunas de las funciones, abre la base de datos Neptuno y ejecuta estos pasos:

Haz clic en el acceso directo Módulos de la ventana Base de datos.

Haz doble clic en el módulo Funciones de utilidad para abrirlo en el Editor **VBA**. Si no haz cambiado las funciones de acoplamiento del Editor **VBA**, la ventana Inmediato aparecerá en la parte de debajo de la ventana.

Escribe `?Now` en la opción inferior de la ventana Inmediato y pulsa Intro. La fecha y la hora del reloj de la computadora aparecerán en la siguiente línea. El signo `?` Es la abreviatura de la instrucción **VBA Print** y deberá ser añadido a la función `Now` para que aparezca el valor de la función. Si haz olvidado anteponer a la entrada de la función un signo `?` O `Print`, aparecerá un mensaje de error, indicando que **Access XP** esperaba que escribiera una instrucción o un signo igual. Haz clic en Aceptar y antepón `?` Al nombre de la función en la ventana Inmediato. Pulsa Fin para devolver el caret al final de la línea y pulsa Intro para recuperar la prueba.

Para volver a colocar la ventana Inmediato de forma más sencilla, haz clic en la barra de título y arrastra la ventana a un área central para mostrarla donde permanezca desacoplada.

A medida que vas escribiendo las funciones de la ventana Inmediato, **Access XP** mostrará una sugerencia de autocompletar, que mostrará el nombre de la función y la lista completa de argumentos. Deberás poner un espacio o paréntesis de apertura detrás del nombre de la función con el fin de que se abra la ventana desplegable Autocompletar. Una lista de argumentos es la lista de información que se especificas para que la función opere. Puedes activar o desactivar esta función eligiendo Herramientas, Opciones y marcando o quitando la marca a la opción Información rápida automática de la página Editor.

Si haz hecho clic en el vínculo Ejemplo activado en cualquiera de las funciones de la ventana Ayuda, la ventana mostrará un ejemplo de la función utilizada en código **Access VBA**. Estos ejemplos muestran la sintaxis de las funciones y sus argumentos apropiados. Sin embargo, los ejemplos generalmente no son aplicables al uso de las funciones de una consulta o regla de validación de **Access XP**.

El tipo de datos Variant de Access XP y VBA

Variant es un tipo de datos especial único en los dialectos de Microsoft Visual Basic. El tipo de datos Variant permite conectar valores que normalmente poseen tipos de datos distintos, como un entero y una celda de caracteres. La posibilidad de concatenar distintos tipos de datos se denomina Desactivar la comprobación del tipo de datos. Los valores Variant están relacionados con el tipo de datos `AS ANY`, que **Access VBA** utiliza para desactivar la comprobación del tipo de datos a la hora de declarar funciones externas contenidas en las bibliotecas de vínculos dinámicos (DLL) de Windows.

El tipo de datos Variant permite conectar valores de campo de tablas y consultas provistos de tipos de datos distintos sin usar las funciones de conversión de tipo de datos de **VBA**, como **STR** (**STR** convierte los valores numéricos al tipo de datos String). El tipo de datos Variant también simplifica expresiones que combinan valores de campo para crear índices concatenados. El tipo de datos Variant también permiten usar el símbolo **&** **PATRA** concatenar los valores de distintos tipos de datos. **SQL** te obliga a usar el signo **&** para hacer una concatenación semejante.

La siguiente tabla enumera los 16 subtipos comunes del tipo de datos Variant de **Access VBA**, junto con los nombres de las constantes intrínsecas de **Visual Basic**, **vbConstant**, correspondientes al valor del subtipo Variant.

Funciones para Fecha y Hora

Access XP ofrece una serie de funciones relativas a fechas y horas. Si alguna vez haz usado **Visual Basic**, probablemente reconozcas la mayoría de las funciones aplicables a los tipos de datos Fecha y Hora. **VBA** posee varias funciones Fecha/Hora, como **DateAdd/DateDiff**, que sirven para simplificar el cálculo de valores de fecha. Las funciones **MonthName** y **WekdayName** constituyen una novedad en **VBA 6.0**. Los ejemplos siguientes emplean el año 1998, ya que en ese año es el último en el que los datos están disponibles en las tablas de Neptuno.mdb.

Función	Descripción	Ejemplo	Devuelve
Date	Devuelve la fecha y hora activa del sistema como subtipo 7 de fecha Variant ó una fecha estándar String de subtipo 8.	Date	15/7/98 15-07-98
DateAdd	Devuelve una fecha de subtipo 7 con un número especificado de días (<<d>>), semanas (<<ww>>) o años (<<a>>) agregados a la fecha.	DateAdd (<<d>>), 31, #15/7/98#)	15/8/98
DateDiff Date	Devuelve un Entero que representa la diferencia entre dos fechas utilizando la especificación d/w/m/a.	DateDiff (<<d>>, Date, #4/15/98#)	-91 (asumiendo que =15/7/98)
DatePart	Devuelve la parte especificada de una fecha como un día, mes, año, día de la semana (<<w>>), etc., como Entero	DatePart (<<w>>, #15/7/98)	7 (sábado)
DateSerial	Devuelve un subtipo 7 Variant de argumentos año, mes y día	DateSerial (98,7,15)	15/7/98
DateValue	Devuelve un subtipo 7 Variant que corresponde a un argumento fecha en un formato de caracteres	DateValue (<<15-Jul-98>>)	15/7/98

Day	Devuelve un Entero entre 1 y 31 (inclusive) que representa un día del mes de un valor Fecha/Hora	Day(Date)	15(asumiendo que la fecha es día 15 de mes)
Hour	Devuelve un Entero entre 0 y 23 (inclusive) que representa la hora de un valor Fecha/Hora	Hour(#2:30 PM#)	14
Minute	Devuelve un Entero entre 0 y 59 (inclusive) que representa el minuto de un valor Fecha/Hora	Minute(#2:30 PM#)	30
Month	Devuelve un Entero entre 1 y 12 (inclusive) que representa el mes de un valor Fecha/Hora	Month(#15-Jul-98#)	7
MonthName	Devuelve el nombre completo o abreviado de un mes a partir del número de mes (1 a 12). Si se omite el segundo argumento, la función devolverá el nombre completo.	MonthName (10, False) MonthName (10, True)	Octubre, Oct.
Now	Devuelve la fecha y hora del sistema de reloj de la computadora como Variant de subtipo 7.	Now	15/7/98 11:57:28 AM
Second	Devuelve un Integer entre 0 y 59 (inclusive) que representa el segundo de un valor Fecha/Hora	Second(Now)	28
Time,	Devuelve la porción Hora de un valor Fecha/Hora del reloj del sistema	Time	11:57:28 AM
TimeSerial	Devuelve el valor de hora serial expresada en horas, minutos y segundos en forma de enteros,	TimeSerial 11:57:28 AM	11:57:28 AM
TimeValue	Devuelve el valor de hora serial (introducida como valor String) como subtipo 7 Variant.	TimeValue (<<11:57>>)	11:57
Weekday	Devuelve el día de la semana (Domingo=1) correspondiente a la fecha como un Integer.	Weekday (#15/7/98#)	7
WeekdayName	Devuelve el nombre completo o abreviado del día a partir del número del día (de 0 a 7) estableciendo el segundo argumento a True se abrevia el nombre. Un tercer argumento opcional permite especificar el primer día de la semana.	WeekdayName (4, False) WeekdayName (4, True)	Mie, Miércoles

Year Devuelve el año de un valor Yaer(#15/7/98#) 1994
Fecha/Hora como Entero.

Funciones de manipulación de texto

La siguiente tabla enumera las funciones que tratan del tipo de datos Texto, correspondientes al tipo de datos **String** de VBA. Casi todas estas funciones están moldadas en base a funciones de cadena **BASIC**.

Función	Descripción	Ejemplo	Devuelve
ASC	Devuelve valores numéricos ANSI de caracteres como Integer	ASC (<<C>>)	67
Chr	Devuelve un carácter correspondiente al valor ANSI numérico como String	Chr(67), Chr(10)	C (avance de línea)
Format	Da formato a una expresión de acuerdo con cadenas apropiadas de formato	Format(Date,<<dd- mmm-aa>>)	15-Jul-98
InStr	Devuelve la posición de una cadena que está dentro de otra	InStr (<<ABC>>,<<C>>)	3
LCase	Devuelve la versión en minúscula de una cadena	LCase(<<ABC>>)	abcd
Left	Devuelve los caracteres que hay más hacia la izquierda de una cadena (<<ABCDEF>>,3)	Left	ABC
Len	Devuelve el número de caracteres de una cadena como Long	Len(<<ABCDE>>)	5
LTrim	Elimina los espacios iniciales de una cadena	LTrim(<<ABC>>)	ABC
Mid	Devuelve una porción de una cadena	Mid(<<ABCDE>>,2,3)	BCD
Right	Devuelve los caracteres que hay más hacia la derecha en una cadena	Rigth(<<ABCDEF>>,3)	DEF
RTrim	Elimina los espacios finales de una cadena	RTrim(<<ABC>>)	ABC
Space	Devuelve una cadena compuesta por un número especificado de espacios	Space(5)	
Str	Convierte el valor numérico de cualquier tipo de datos en una cadena	Str(123.45)	123.45
StrComp	Compara dos cadenas para su equivalencia y devuelve el resultado de entero de la comparación	StrComp (<<ABC>>,>>abc>>)	0

String	Devuelve una cadena formada por caracteres repetidos específicos	String(5,<<A>>)	AAAAA
Trim	Elimina los espacios iniciales y finales de una cadena	Trim(<<ABC>>)	ABC
UCase	Devuelve la versión en mayúscula de una cadena	UCase(<<abc>>)	ABC
Val	Devuelve el valor numérico de una cadena de un tipo de datos apropiado para el formato del argumento	Val(<<123.45>>)	123.45

Bibliografía

Microsoft Access 2002

Autor Jennings

Editorial Que Prentice Hall

Microsoft Office 2002

Autor Joe Kraynak

Editorial Prentice Hall Pearson

Traducción de The complete Guide to Microsoft Access 2002

Microsoft Office XP

Halvorson Young

Mc-Graw Hill

Access xp

Halvorson Young

Mc Graw Hill

ÍNDICE

A

Ajustar otras propiedades y opciones de formulario	30
Analizar un informe con Excel	40
Anexo B	41

B

Bibliografía	55
Botón de opción, Grupo de opciones	29

C

Casilla de verificación	28
Como agregar y modificar encabezados y pies de página	38
Como agregar y modificar controles en la vista diseño	27
Cómo agrupara y ordenar registros en un informe	37
Cómo Cambiar el Diseño de un Formulario	26
Como colocar controles en un formulario	29
Como convertir los datos de Access XP a etiquetas	34
Cómo crear un informe básico	33
Cómo Exportar un Informe a otro Formato de Datos	39
Cómo guardar un informe en un formato de imagen instantánea	39
Como insertar, eliminar y reorganizar los campos	16
Cómo seleccionar consultas	7
Como trabajar con cálculos en un informe	39
Como trabajar con la vista diseño	31
Consultas de acción	12
Consultas de referencias cruzadas	9
Consultas en Access XP	6
Consultas parametrizadas	10
Consultas SQL	14
Creación de gráficos	35
Creación de un formulario básico con un clic	24
Creación de un Formulario Nuevo	23
Creación de un Informe a Partir de Datos de Access XP	31
Creación y Aplicación de Filtros	19
Creación y Modificación de Consultas	15
Cuadro combinado	28
Cuadro de lista	28

D

Definición de criterios	17
Definición de las propiedades de consulta	18
Definición de una columna calculada	17
Diseño y Uso de Formularios	21

E

El Encabezado o pie de página del informe	31
El Encabezado y pie de página	32
Encabezados de columna	18
Entender las Consultas	6
Etiqueta	29
Exportar un informe a Microsoft Word	40
Expresiones en Access XP	41

F	
Funciones	49
G	
Guardar un informe como página Web	40
L	
La sección Detalle	32
Las secciones Encabezado del grupo y Encabezado de pie	32
M	
Modificación de un Informe	37
Mostrar todos los campos	19
P	
Presentación de Datos con Informes	31
R	
Registros únicos	18
U	
Uso de Consultas para Buscar y Filtrar Datos	6
Uso de Formularios para Ver e Introducir Datos	21
Uso del Asistente para formularios	24
V	
Valores superiores	18
Valores únicos	18
Ver o Introducir Datos con un Formulario	26
Vincular campos secundarios y Vincular campos principales	19



FACULTAD DE INGENIERÍA UNAM
DIVISIÓN DE EDUCACIÓN CONTINUA

CURSOS INSTITUCIONALES

ACCESS AVANZADO

Del 01 al 14 de Diciembre de 2004

NUEVAS CARACTERÍSTICAS

CI-209

Instructor: Ing. Otelo Galicia Cedillo
DELEGACIÓN MIGUEL HIDALGO
DICIEMBRE 2004

Nueva Característica

Sugerencia: Buscar asistentes adicionales

Para buscar asistentes adicionales de Access XP y descargarlos, has clic en el comando Plantillas en Microsoft.Com en el panel de tareas Nuevo archivo para conectarte al sitio Plantillas de Microsoft Office en el Web.

Sugerencia: Crear una base de datos en Windows

Puedes crear una nueva base de datos sin ejecutar primero Access XP seleccionando el comando Nuevo documento de Office en el menú Inicio en Microsoft Windows. Para crear una base de datos en blanco, has doble clic en el elemento Base de datos en blanco en la ficha General del cuadro de diálogo Nuevo documento de Office. Para crear una base de datos mediante un asistente, has clic en la ficha Base de datos.

A fondo: Cómo evitar la pérdida de datos

El hecho de guardar los datos automáticamente en Vista Hoja de datos o Vista Formulario puede contribuir a evitar la pérdida de la información que haya agregado. No obstante, ten en cuenta que debido a esta característica no puedes invertir los cambios que hayas realizado simplemente abandonando la base de datos sin guardar, como sí sucede en un documento de Word o en un libro de trabajo de Excel. Por ejemplo, si eliminas accidentalmente un grupo considerable de registros o campos, no puedes deshacer ese cambio simplemente cerrando la base de datos sin guardarla, ya que el cambio ya se habrá escrito en el archivo de disco de la base de datos. De la misma forma, tampoco puedes deshacer la eliminación de uno o más registros o campos seleccionando el comando deshacer. Aunque Access XP normalmente te advierte antes de eliminar los datos, este tipo de eliminaciones son permanentes.

Además, Access XP tiene comandos capaces de eliminar rápidamente grandes cantidades de datos. La mejor manera de evitar la pérdida de datos en Access XP es realizar copias de seguridad frecuentes del archivo de base de datos y crear siempre una copia de seguridad de la base de datos antes de efectuar cambios importantes en los datos o en el diseño. De esta forma, podrás volver al estado anterior de la base de datos si fuera necesario.

Nueva Característica

Sugerencia: seleccionar el mejor formato de base de datos

Cuando creas una base de datos en blanco o una basándote en un asistente, Access 2002 almacena la base de datos en un archivo de base de datos de Access (un archivo .mdb) en el mismo formato que utilizabas en Access 2000. Este formato es excelente para las bases de datos más pequeñas que ofrece la ventaja de permitirte compartir fácilmente el archivo de base de datos con los usuarios de Access 2000. Sin embargo, si creas una base de datos relativamente grande, puedes obtener un rendimiento mayor convirtiendo la base de datos a nuevo formato de Access 2002. Para ello, selecciona Herramientas, Utilidades de la base de datos, Convertir base de datos, A formato de archivo de Access 2002.

Access XP creará un archivo de base de datos diferente con el nuevo formato, dejando intacto el archivo de base de datos original. Aunque no podrás compartir inmediatamente una base de datos con formato de Access 2002 con usuarios de versiones anteriores de Access XP, si es posible convertirla de nuevo a un formato anterior seleccionando Herramientas, Utilidades de la base de datos, Convertir base de datos, A formato de archivo de Access 2000, o bien Herramientas, Utilidades de la base de datos, Convertir base de datos, A formato de archivo de Access 97.

Nueva Característica

Sugerencia: Exportar datos de Access XML u otros formatos

Puedes exportar una tabla u otro objeto de base de datos a un archivo extremo seleccionando el objeto en la ventana base de datos y eligiendo Archivo, Exportar. Puedes exportar un objeto de base de datos a una base de datos de Access, Lotus 1-2-3, Paradox o dBASE, a un libro de trabajo Excel, a un documento XML o una página Web de HRML o Web de ASP (página Active Server, Active Server Page), a un texto o a un archivo RTF (Formato de texto enriquecido, Rich Text Format), o a un origen de datos ODBC.

También puedes exportar una tabla, consulta, un formulario o un informe a otra aplicación de Microsoft submenú Vínculos de Office en el menú Herramientas.

El comando Combinar con MS Word sólo se encuentra disponible para una tabla o consulta. Este comando ejecuta una operación de combinación de correo en Microsoft Word 2002, utilizando la tabla o consulta como una lista de destinatarios.

Sugerencia: Utilizar el menú de método abreviado

A medida que te familiarices con Access XP, ten presente que a veces puede mostrar un menú de método abreviado de comandos útiles haciendo clic con el botón secundario en un campo, barra de título, columna o encabeza de una fila u otro objeto. El menú de método abreviado proporcionará los comandos adecuados para trabajar con el objeto determinado en el que has hecho clic. Por ejemplo, puedes insertar un campo nuevo en una tabla haciendo clic con el botón secundario en la fila situada por debajo del punto de inserción deseado y seleccionado Insertar filas del menú de método abreviado.

Sugerencia: Hacer que Access XP cree el campo

Cuando agregues un campo, puedes ahorrar bastante tiempo mediante el Generador de campos, que permite seleccionar e insertar un campo predefinido adecuado para un fin concreto (Como, por ejemplo, un número de teléfono en una tabla de contactos o un número de serie en una tabla de productos) La ventaja principal de utilizar el Generador de campos es que el campo que insertes tendrá todas las propiedades configuradas en los valores adecuados. (El Generador de campos ofrece la misma lista de campos predefinidos que el Asistente para tablas)

Para utilizar el Generador de campos, has clic en la fila de la lista de campos situada por debajo de la posición donde desees insertar el nuevo campo y has clic en el botón Generar de la barra de herramientas.

Precaución

Si un campo determinado ya incluye datos, ten en cuenta que la eliminación de dicho campo suprimirá los datos en cada registro de la tabla. Ahora, Access XP permite deshacer varias acciones en Vista Diseño (Para deshacer una acción, selecciona deshacer en el menú Edición, has clic en el botón deshacer de la barra de herramientas o presiona CTRL+Z) Sin embargo, una vez guarda el diseño de tabla (algo que debes hacer antes de cambiar las vistas) no podrás deshacer la eliminación y los datos se eliminarán permanentemente.

Precaución

Si un campo ya almacena datos, cambiar su tipo de datos puede provocar la pérdida de información. Por ejemplo, si cambias el tipo de datos de un campo de Texto a Número, Access XP eliminará permanentemente el contenido de cualquier campo que no incluya un valor numérico válido, y puedes incluso modificar los valores de los números según la propiedad Tamaño de campo (por ejemplo, si configura Tamaño de campo a entero largo, Access XP convertirá 29.95 a 30)

Solución de problemas: Los elementos del campo Búsqueda se muestran aleatoriamente.

Has utilizado el Asistente para búsquedas para crear un campo de búsqueda. En Vista Hoja de datos, la lista desplegable del campo muestra los campos correctos de la tabla relacionada, pero los campos aparecen enumerados de forma aleatoria.

El Asistente para búsquedas no permite aplicar un orden a los elementos que aparecen en la lista desplegable de un campo de búsqueda. Para que Access XP ordene estos elementos, has clic en el campo de búsqueda en la lista de campos de la ventana de Vista Diseño, abre la ficha Búsqueda en la parte inferior de la ventana, has clic en el cuadro propiedades origen de la fila y has clic en el botón de puntos suspensivos (...) que aparece a la derecha del cuadro. A continuación, en la ventana Instrucción SQL: Generador de consultas, selecciona Ascendente o Descendente en la fila Orden, bajo el campo que deseas utilizar para realizar la ordenación. Cierra la ventana Instrucción SQL: Generador de consultas haciendo clic en la X en la esquina superior derecha de la ventana. Si deseas guardar los cambios realizados, responde sí en el cuadro de diálogo que te pregunta "¿Desea guardar los cambios realizados a la instrucción SQL y actualizar la propiedad?"

Solución de problemas: Necesidad de una relación de varios a varios

Necesitas conectar dos tablas con una relación de varios, pero Access XP no proporciona este tipo de relación.

En Access XP, puedes configurar de manera indirecta una relación de varios a varios creando una tabla intermedia. Considera, por ejemplo, la base de datos de muestra Libro3. Si alguno de los libros tiene más de un autor, y deseas poder establecer un vínculo con todos los autores de libros y vincular un autor con todos los libros de un autor, es necesario establecer una relación de varios a varios. Es posible establecer este tipo de relación, permitiendo así que la base de datos de inventario de libros admita más de un autor para cada libro, como se indica a continuación:

1. Agrega un único campo de identificación a la tabla Libros, de forma que cada registro de la tabla intermedia pueda hacer referencia a un registro de libro específico (y convierta este campo en la clave principal mientras trabajas con él) Asimismo, elimina el campo IdAutor en la tabla Libros ya que no es necesario.
2. Crea una tabla inmediata (quizá llamada Autores de Libros) en la que cada registro incluya el identificador único de un registro de libro además del identificador único de un registro de un autor, vinculado así un libro a uno de sus autores y un autor a uno de sus libros.
3. Define una relación entre las tablas Libros y Autores de libros y una relación entre las tablas Autores de libros y Autores. Estas dos relaciones serán de uno a varios, pero el resultado general será la conexión de Autores y Libros en una relación indirecta de varios a varios.

Sugerencia: Escalar un campo

Si consideras que la escritura de datos en una celda en Vista Hoja de datos es algo limitado, o si la fuente es algo pequeña como para ver con claridad presiona MAYÚS+F2 para abrir el cuadro de diálogo Zoom, que permite escribir los datos del campo en un cuadro de lista espacioso y en el que se puede desplazar, y seleccionar cualquier fuente. (La fuente que seleccione sólo se utilizará en el cuadro de diálogo Zoom; no afectará al texto en Vista Hoja de datos) La escala es muy útil para escribir una gran cantidad de texto en un campo con el tipo de datos Memorando (que puede albergar hasta 64,000 caracteres) El cuadro de diálogo Zoom te permitirá escribir más datos de los permitidos para el campo; sin embargo, cuando hagas clic en el botón Aceptar del cuadro de diálogo Zoom, Access te advierte que estás a punto de truncar los datos. Asimismo, el cuadro de diálogo Zoom no se encuentra disponible para los campos con el tipo de datos Fecha/Hora.

Precaución

Cuando hayas eliminado un registro y hayas hecho clic en el botón Sí para confirmar la acción, no podrás restaurar el registro. No es posible deshacer la eliminación de un registro mediante el comando Deshacer. Si deseas restaurar un registro eliminado tendrás que volver a escribirlo desde cero

Sugerencia: Buscar o reemplazar texto rápidamente

Puedes buscar texto rápidamente en Vista Hoja de datos seleccionando Edición, Buscar, haciendo clic en el botón Buscar de la barra de Herramientas o presionando CTRL+B, A continuación, escribe el criterio de la búsqueda en la ficha Buscar del cuadro de diálogo Buscar y reemplazar. Para reemplazar texto, selecciona Edición, Reemplazar, o presiona CTRL+L para abrir la ficha Reemplazar del cuadro de diálogo Buscar y reemplazar.

Sugerencia: Permitir que Access ajuste automáticamente el ancho de la columna

Puedes hacer que Access XP determine el ancho más conveniente de la columna haciendo doble clic en el borde derecho del encabezado de la columna de Access XP ajustará el ancho de la columna al tamaño mínimo que permita mostrar toda la información que se incluye en esa columna (incluyendo el título de la columna)

A Fondo:

Ten en cuenta que cuando eliminas una columna estás eliminando permanentemente un campo de la tabla, junto con todos sus datos. No puedes ejecutar un comando Deshacer para invertir esta situación. Si eliminas una columna y, a continuación, te das cuenta de que debería haber dejado la información donde estaba, tendrás que volver a crear el campo y volver a escribir la información de campo para cada registro en la base de datos. En resumen, ten cuidado con la eliminación de columnas y ten siempre una copia de seguridad en su base de datos antes de realizar cualquier cambio importante.

Sugerencia: Ocultar Columnas temporalmente

También puedes ocultar temporalmente una o más columnas seleccionándolas y eligiendo Ocultar columnas en el menú Formato. Para seleccionar una única columna, has clic en el encabezado. Para incluir columnas adicionales, mantén presionado el botón del ratón y arrastra a la derecha o a la izquierda (si liberas el botón después de hacer clic, desplazará la columna inicial en lugar de ampliar la selección) Más adelante, podrás hacer que aparezca de nuevo eligiendo la opción Mostrar columnas del menú Formato y seleccionando las columnas que desees volver a ver.

Sugerencia: Incluya todos los campos necesarios

Asegúrate de incluir todos los campos necesarios (para mostrar información en la consulta, para seleccionar los datos que se muestran o para ordenar los resultados de la consulta)

A Fondo:

Si has seleccionado los campos que pertenecen a los dos objetos de la base de datos (tablas o consultas) que no tienen una relación formal definida en la ventana Relaciones, cuando has clic en el botón Siguiente, Access XP mostrará un cuadro de mensaje y te exigirá que definas explícitamente la relación en la Ventana Relaciones y, a continuación, reinicia el asistente y vuelve a seleccionar los campos que desees Por algún motivo, Access XP no crea simplemente las relaciones definidas a la ventan Diseño de la consulta. Para evitar este problema, asegúrate de definir de manera explícita las relaciones entre todas las tablas o consultas que desees utilizar en la consulta.

Sugerencia: Cómo obtener ayuda para crear una expresión

Para crear una expresión compleja, puedes utilizar el cuadro de diálogo Generador de expresiones, que ofrece botones para insertar operadores (aritméticos, de comparación, booleanos y de agrupamiento) Asimismo, incluye las listas que permiten escribir rápidamente cualquier campo en la base de datos, así como cualquier función disponible, constante, operador o expresión común. Para utilizar el Generador de expresiones, haz clic con el botón secundario en el cuadro donde desees escribir la expresión y selecciona Generar en el menú de método abreviado. Puedes hacer clic en el botón Ayuda en el cuadro de diálogo Generador de expresiones para obtener más información sobre sus características.

Sugerencia: Cómo ejecutar o modificar rápidamente una consulta

Puedes cambiar rápidamente una consulta entre Vista Diseño, en el que puedes modificar el diseño de la consulta, y Vista Hoja de datos, en la que puedes ver el resultado de la consulta, simplemente haciendo clic en el botón Ver de la barra de herramientas.

Sugerencia: Cómo ver el porcentaje superior o inferior

La lista desplegable Valores superiores también permite seleccionar o escribir un porcentaje en lugar de un número real absoluto de registros a mostrar. Por ejemplo, en una consulta que ofrezca un listado de estudiantes y sus notas, podrías seleccionar el tipo de ordenación Descendente en el campo de resultados y escribir 15% en el cuadro de lista Valores superiores para obtener una lista con el 15 por 100 de los estudiantes más aventajados de la clase.

Sugerencia de problemas: Access XP no ejecutará una consulta con un campo calculado

Has creado una consulta con un campo calculado, pero cuando intentas cambiar a Vista Hoja de datos para ver los resultados de la consulta, Access XP muestra un misterioso cuadro de diálogo llamado Valor del parámetro en lugar de ejecutar la consulta.

Si escribes un nombre que no es nombre de uno de los campos de las tablas que se utilizan en la consulta, Access XP supondrá que el nombre es un parámetro que se debe escribir cada vez que se ejecute la consulta. Por ejemplo, en la consulta de ejemplo que se describe en la sección <<Adición de campos calculados>>, si escribes por error **Coste** en lugar de **PrecioCompra** en la expresión en el campo calculado, Access trataría Coste como un parámetro y solicitará el valor cuando ejecute la consulta, ya que la tabla Libros que se ha utilizado en esta consulta no incluye el campo llamado Coste.

Ten en cuenta este comportamiento cuando desees que Access XP solicite el parámetro cada vez que ejecute la consulta. Por ejemplo, en la base de datos Libro3 de ejemplo, si deseas crear una consulta que muestre los libros que cuestan menos de una determinada cantidad, puedes escribir la expresión <[Coste]> en la fila criterios del campo PrecioCompra (Los corchetes son necesarios en este contexto para indicar que Coste es un nombre en lugar de un valor de texto literal) Como la tabla Libros no incluye un campo llamado Coste, Access XP considerará a Coste como un parámetro y solicitará su valor cada vez que ejecute la consulta.

Sugerencia: como convertir una consulta en una consulta de tabla de referencia cruzada

También puedes convertir una consulta de tablas de referencias cruzadas. Para ello, ábrela en Vista Diseño y selecciona Consulta de tabla de referencias cruzadas en el menú Consulta. Este comando agregará una fila Total y Tab ref cruz. A la cuadrícula de diseño de la consulta y convertirá la consulta en una consulta de tabla de referencias cruzadas. En la fila Tab ref cruz., selecciona Encabezado de fila para un campo con el fin de designar los encabezados de la columna, y selecciona Valor para un campo para designar el campo que desees utilizar para mostrar los valores en las intersecciones de la fila y la columna. El método más sencillo para aprender a configurar la consulta de tabla de referencias cruzadas en Vista Diseño es estudiar el diseño de una consulta de tabla de referencias cruzadas generada por el Asistente para consultas de referencias cruzadas.

Precaución:

Antes de ejecutar cualquiera de las consultas que se mencionan en las siguientes secciones, realiza una copia de seguridad de la base de datos. Estas consultas pueden modificar permanentemente la base de datos, quizá eliminando gran cantidad de datos. También puedes realizar la copia de seguridad de tablas específicas que se modificarán realizando copias de las tablas mediante la ventana Base de datos.

Sugerencia: Insertar un gráfico en cualquier formulario

Independientemente del método que sigas para crear un formulario, puedes utilizar el Asistente para insertar un gráfico en él abriendo el formulario en Vista Diseño, seleccionando Gráfico en el menú Insertar y arrastrando para determinar el área rectangular donde deseas mostrar el gráfico. El Asistente para gráficos se ejecutará y te guiará a través del proceso de diseño de un gráfico.

Sugerencia: Como controlar el orden de los campos

El orden en que aparecen los campos en la lista Campos seleccionados es el orden en que el asistente dispondrá los campos en el formulario. Aunque puedes volver a organizar los campos que ya se han agregado a la lista Campos seleccionados, puedes controlar la posición donde se agregará el siguiente campo. Para ello, primero selecciona el campo en la lista Campos seleccionados que se encuentra encima de la posición donde deseas agregar el nuevo campo. A continuación, agrega el nuevo campo, te situará justo por debajo del campo seleccionado. Si no te gusta la posición de un campo determinado en la lista Campos seleccionados, selecciónalo, has clic en el botón < para eliminarlo y, a continuación, agrégalo de nuevo a la lista en la posición deseada.

Solución de problemas: El objeto OLE incrustado no aparece

Has agregado un control de objeto independiente para mostrar un objeto OLE en el formulario. Sin embargo, lo único que aparece es un icono de programa que creó el icono.

Para mostrar el contenido de un objeto incrustado, en lugar de un simple icono, has clic en el control para seleccionarlo y elige Edición, Objeto X, Convertir (donde X es el tipo de objeto como, por ejemplo, Ecuación) A continuación, en el cuadro de diálogo Convertir, desactiva la opción Mostrar como icono.

Sugerencia: Utilice controles independientes para cambiar el formato del texto

A veces, quizá deseas que la etiqueta tenga dos formatos de texto diferentes, por ejemplo, una etiqueta que diga <<Dirección completa>> No es posible aplicar varios formatos de texto a un control, aunque sí puedes obtener el mismo resultado creando dos controles de etiquetas alineados para que parezca como si fueran uno. Como cada parte de la etiqueta es un control diferente, puedes asignar diferentes formatos a cada una.

Sugerencia: Cómo crear un control calculado

Puedes definir las propiedades de un control de cuadro de texto para que muestre el resultado de un cálculo realizado en uno o más de los campos del origen del registro, en lugar de mostrar simplemente o de configurar el contenido de un único campo. Para ello, muestra la hoja de propiedades del cuadro de texto, has clic en la ficha Datos y escribe la expresión para el cálculo que desea realizar en el cuadro de propiedades Origen del control. (Para un control dependiente, la propiedad Origen del control se suele establecer al nombre del campo al que está vinculado el control) En el formulario de ejemplo Libros que se muestra, si quisieras mostrar un control de cuadro de texto que se mostrará el precio total del libro, incluyendo los impuestos podrías escribir la siguiente expresión (suponiendo que el porcentaje de impuestos por la venta sea del 7 por 100):

$=[\text{PrecioCompra}] * 1,07$

Para dar formato correctamente al resultado calculado, podrías seleccionar el formato Moneda en el cuadro de propiedades Formato en la ficha Formato de la hoja de propiedades.

Para obtener ayuda sobre cómo escribir una expresión compleja, has clic en el botón de puntos suspensivos (...) que aparece a la derecha del cuadro de propiedad Origen del control del control cuando esté activo. Se iniciará el Generador de expresiones.

Un control calculado en un formulario es lo mismo que un campo calculado en una consulta.

Nueva Característica:

En Access 2002, puede modificar el funcionamiento del corrector ortográfico seleccionando Opciones en el menú Herramientas y abriendo la ficha Ortografía en el cuadro de diálogo Opciones.

Sugerencia: Cómo Buscar o reemplazar texto rápidamente

Puedes buscar rápidamente el texto especificado en cualquiera de los registros seleccionando Buscar en el menú Edición, haciendo clic en el botón Buscar de la barra de herramientas o presionando CTRL+B y, a continuación, escribiendo el criterio de búsqueda en la ficha Buscar y reemplazar. Para reemplazar texto, selecciona Reemplazar en el menú Edición, o presiona CTRL+L para abrir la ficha Reemplazar en el cuadro de diálogo Buscar y reemplazar.

Sugerencia: Cómo ordenar o filtrar los registros mostrados de un formulario

Puedes ordenar o filtrar los registros que ves en un formulario de la misma forma que ordenas o filtras los registros cuando muestra una tabla, consulta o formulario en Vista Hoja de datos.

Solución de problemas: Ha desaparecido todos los registros

Mientras muestras un formulario en Vista Formulario, selecciona el comando Entrada de datos del menú Registros para ver lo que hace. Ahora, aparentemente faltan todos los registros y no es posible recuperarlos.

El comando Entrada de datos se inicia de un modo especial que permite escribir nuevos registros pero oculta los registros existentes en el origen de datos. Lo verdaderamente confuso del comando es que puedes desactivarlo seleccionando la misma opción del menú que lo activa. Aunque el comando Entrada de datos no aplica en realidad un filtro, para desactivar el modo de entrada de datos selecciona Registro, Quitar filtro u orden. Después de seleccionar Quitar filtro u orden, verás todos los registros de nuevo.

Nueva Característica

También puedes crear una página de acceso a datos desde cero en Vista Diseño seleccionando el comando Página de acceso a datos en blanco del área Nuevo del panel tareas Nuevo archivo.

Nueva Característica

Sugerencia: Cómo convertir otro objeto de base de datos en una página de acceso de datos

Puedes crear una página de acceso a datos guardando una copia de una tabla, consulta, formulario o informe como una página de acceso a datos. Para ello, selecciona el objeto de base de datos en la ventana Base de datos y selecciona Guardar como en el menú Archivo. A continuación, en el cuadro de diálogo Guardar como, selecciona Página de acceso a datos en la lista desplegable Como, y escribe un nombre para la página de acceso a datos en el cuadro de texto Guardar.

Sugerencia: Como insertar un gráfico en cualquier informe

Independientemente de cómo crees un informe, puedes utilizar el Asistente para gráficos para insertar un gráfico en él para abrir un informe en Vista Diseño, seleccionando Gráfico en el menú Insertar y arrastrando para marcar el área rectangular en el informe donde desees mostrar el gráfico. El Asistente para gráficos comenzará a ejecutarse y te guiará a través del proceso de diseño de un gráfico.

Sugerencia: Utilizar Word para imprimir etiquetas

Microsoft Word permite imprimir etiquetas, cartas de formulario, sobres y otros tipos de documentos de combinación de correspondencia directamente utilizando una base de datos de Access XP como una lista de destinatarios (es decir, como el origen de datos) Para imprimir etiquetas, Word ofrece más características, mayor flexibilidad y, quizá, una interfaz más conocida que el Asistente para etiquetas de Access XP. Si ya has abierto la base de datos de Access que deseas utilizar para imprimir documentos combinados, una forma rápida de ejecutar el Asistente para combinar correspondencia de Word es seleccionar la tabla que deseas utilizar como la lista de destinatarios en la ventana Base de datos y, a continuación, seleccionar Herramientas, Vinculos con Office, Combinar con MS Word.

Sugerencia: Cómo utilizar niveles de agrupamiento para resumir la información

Una razón importante para incluir niveles de agrupamiento es que permiten resumir la información numérica en cada grupo. Por ejemplo, si incluyes los niveles de agrupamiento TipoPortada y NombreEditor en el informe de ejemplo podrás mostrar el coste total de los libros con un tipo de portada determinada, así como el coste total de todos los libros por un editor particular con un tipo de portada particular. En lugar de la suma, también podrías mostrar el coste medio, mínimo o máximo del libro.

Sugerencia: Cómo obtener ayuda acerca de las propiedades

En algunas de las propiedades de grupo, puedes obtener información detallada colocando el punto de inserción en el cuadro de propiedades en el cuadro de diálogo Ordenar y agrupar, y presionando F1.

Solución de problemas: El informe incluye páginas en blanco

Cuando imprimes el informe, se incluyen páginas en blanco no deseadas.

Si el informe es más grande que el tamaño del papel especificado en la ficha Página del cuadro de diálogo Configurar página la impresora quizá produzca la aparición de una página en blanco no deseada después de cada página impresa. Intenta reducir el ancho de los márgenes en la ficha Márgenes del cuadro de diálogo Configurar página o cambia a la orientación Horizontal en la ficha Página del mismo cuadro de diálogo. No obstante, si no fuera ninguna de estas soluciones, cambia el diseño de informe en Vista Diseño para reducir el ancho.

Si obtienes una página en blanco no deseada al final del informe y no quieres que aparezca información en el pie de página del informe, asegúrate de establecer a cero el alto del pie de página. Para ello, abre el informe en Vista Diseño, has clic con el botón secundario en la banda de la sección Pie del informe, selecciona Propiedades en el menú de método abreviado y, en la ficha Formato de la hoja de propiedades Pie del informe, establece la propiedad Alto a 0 cm.



FACULTAD DE INGENIERÍA UNAM
DIVISIÓN DE EDUCACIÓN CONTINUA

CURSOS INSTITUCIONALES

ACCESS AVANZADO

Del 01 al 14 de Diciembre de 2004

CONSULTAS

CI-209

Instructor: Ing. Otelo Galicia Cedillo
DELEGACIÓN MIGUEL HIDALGO
DICIEMBRE 2004

Consultas

En Access XP

¿Que es una **Consulta**?

Las **consultas** se utilizan comúnmente para ordenar los datos en los Informes y las Páginas de Acceso a Datos principalmente.

Otelo Galicia Cedillo 2

Uso de las Consultas

Las **consultas** también nos sirven para actualizar, modificar, agregar y eliminar datos en las tablas.

Otelo Galicia Cedillo 3

Tipos de Consultas

Las consultas se clasifican en:

1. **Consultas de Selección**
2. **Consultas de Acción**
3. **Consultas de Parámetros**
4. **Consultas de Tabla Cruzada**
5. **Consultas SQL**

Otelo Galicia Cedillo

4

Consultas de Selección

- Estas consultas recuperan los datos que cumplen determinadas condiciones, agrupan registros para la visualización de datos de resumen y visualizan los cálculos realizados sobre campos de datos

Otelo Galicia Cedillo

5

Consultas de Acción

- Estas consultas modifican de algún modo los datos existentes. Se utilizan Consultas de Acción para *Eliminar, Actualizar, Añadir Datos o Crear una Tabla.*

Otelo Galicia Cedillo

6

Consultas de Parámetros

- Estas consultas solicitan los criterios para la consulta antes de ser ejecutada.

Otelo Galicia Cedillo

7

Consultas de Tabla de Referencia Cruzada

- Estas consultas resumen los datos y a continuación agrupan los valores resumidos en categorías.

Otelo Galicia Cedillo

8

Consultas SQL

- Estas consultas requieren sentencias SQL específicas, que deben ser introducidas en la vista *SQL*. Entre estas consultas se encuentran las consultas de *Unión*, *Paso a Través* y *Definición de Datos*.

Otelo Galicia Cedillo

9

S Q L
(Structured Query Language)

- SQL o Lenguaje Estructurado de Consulta, es un lenguaje extenso y variado que dispone de muchas versiones. Access XP utiliza el motor *SQL Jet* para realizar las consultas.

Otelo Galicia Ceólio 10

S Q L
(Structured Query Language)

- Este tipo de consultas estan fuera de nuestro curso, ya que el Lenguaje SQL es una especialidad de las Bases de Datos. Pero en el curso veremos la relación entre los elementos de la *Vista Diseño* y las sentencias SQL como introducción a este lenguaje.

Otelo Galicia Ceólio 11

Empezamos

Con las

Expresiones



FACULTAD DE INGENIERÍA UNAM
DIVISIÓN DE EDUCACIÓN CONTINUA

CURSOS INSTITUCIONALES

ACCESS AVANZADO

Del 01 al 14 de Diciembre de 2004

EXPRESIONES

CI-209

Instructor: Ing. Otelo Galicia Cedillo
DELEGACIÓN MIGUEL HIDALGO
DICIEMBRE 2004

Expresiones

De Access XP

¿Que es una Expresión?

Una expresión, es una *Instrucción* de intenciones, que produce una *Acción* después de satisfacer una *Condición Específica*

Componentes de una Expresión

Una Expresión esta formada por

1. Operadores
2. Literales
3. Identificadores o Funciones

Operadores

Son símbolos *reservados* de los lenguajes de programación, como Visual Basic, C++ o cualquier otro tipo de programa, que identifican una *Operación* del tipo formal (Matemática o Lógica)

Tipos de Operadores

- Aritméticos
- De Asignación o Comprobación
- Lógicos
- De Concatenación
- De Identificación
- Otros Operadores

Operadores Aritméticos

- Ejecutan operaciones como
- + La Suma
 - [-] La Resta (unano)
 - * La Multiplicación
 - / La División
 - \ La División de Enteros
 - Mod El Módulo
 - ^ La Potencia

Operadores Lógicos

And Y lógico
Or O inclusivo
Not No lógico
Xor O exclusivo

Operadores Lógicos

Se emplean para combinar los resultados de dos o más expresiones de comparación en un solo resultado

<i>Operación</i>	<i>Devuelve</i>
3 < 5 Y 6 > 5	True (Verdadero)
3 < 5 Y 6 < 5	False (Falso)

Operadores de Concatenación

& Concatenación

Combinan dos valores de texto formando una sola cadena de caracteres

<i>Operación</i>	<i>Resultado</i>
BASE & DATOS	BASEDATOS
2B5 & "3QW"	2B5"3QW"

Operadores de Identificación

Se utiliza el signo de admiración [!] y el punto [.]

Son separadores y realizan las siguientes funciones:

1. Combinar los nombres de las clase de objetos
2. Distinguen los nombres de objetos de los nombres de las propiedades
3. Identifican campos específicos de las tablas

Operadores de Identificación

Ejemplos

- D_personal!RFC
- TextBox.FonSize = 10

Otros Operadores

Estos operadores están relacionados con los operadores de comparación y devuelven un valor True o False

Otros Operadores

- Is** Se usa con Null para determinar si un valor es Null (desconocido)
- Like** Determina si un valor de cadena empieza por uno o más caracteres.
- In** Determina si un valor de cadena es integrante de una lista
- Between** Determina si un valor numérico o de fecha esta dentro de un rango

Otros Operadores

Ejemplos

- Is Null**, (es nulo)
- Is Not Null** (No es Nulo)
- Like Pe*** (Busca cadenas con Pe)
- In (F, M)** (Esta F o M)
- Between 1 and 5** (Entre 1 y 5)

Literales

Las literales son los elementos que se combinan con los *Operadores* para formar las *Expresiones*

Tipos de Literales

- Numéricos
- De Texto
- Fecha/Hora

Literales Numéricos

Son una serie de cifras, entre las que se incluyen el signo aritmético [-] y la coma decimal [,], pueden incluir los símbolos "E" o "e" para indicar una exponenciación

1, 2, 3, 56 o 4,000 -1.674E-02

Literales de Texto

Son cualquier carácter imprimible, incluyendo los caracteres que devuelven la función *chr* como son [A-z], [0-9], [~], [@]

Literales Fecha/Hora

- Como su nombre lo indica, se utilizan para hacer referencia a datos del tipo Fecha u Hora.
- Para escribirlos de deben de encerrar entre almohadillas [#]
- #09/07/2003# #10:30:20#

Identificadores

- Es un *nombre de objeto* como las bases de datos, las tablas, los campos, las consultas o los informes (Objetos de Access).
- En ocasiones pose un nombre de familia (clase de objeto) y el nombre dado (nombre de objeto) separados por el signo de [!] o un punto [.] .

D_personal.RFC

Conclusión

- Las *Expresiones* las forman los *Operadores*, las *Literales* y los *Identificadores*.
- Nos sirven para realizar *Operaciones* del tipo *Númérico* (suma, producto etc.) o *Literal* (concatenación) así como operaciones de *Comparación* (mayor que o menor que) y *lógicas* (falso o verdadero)



FACULTAD DE INGENIERÍA UNAM
DIVISIÓN DE EDUCACIÓN CONTINUA

CURSOS INSTITUCIONALES

ACCESS AVANZADO

Del 01 al 14 de Diciembre de 2004

ANEXOS

CI-209

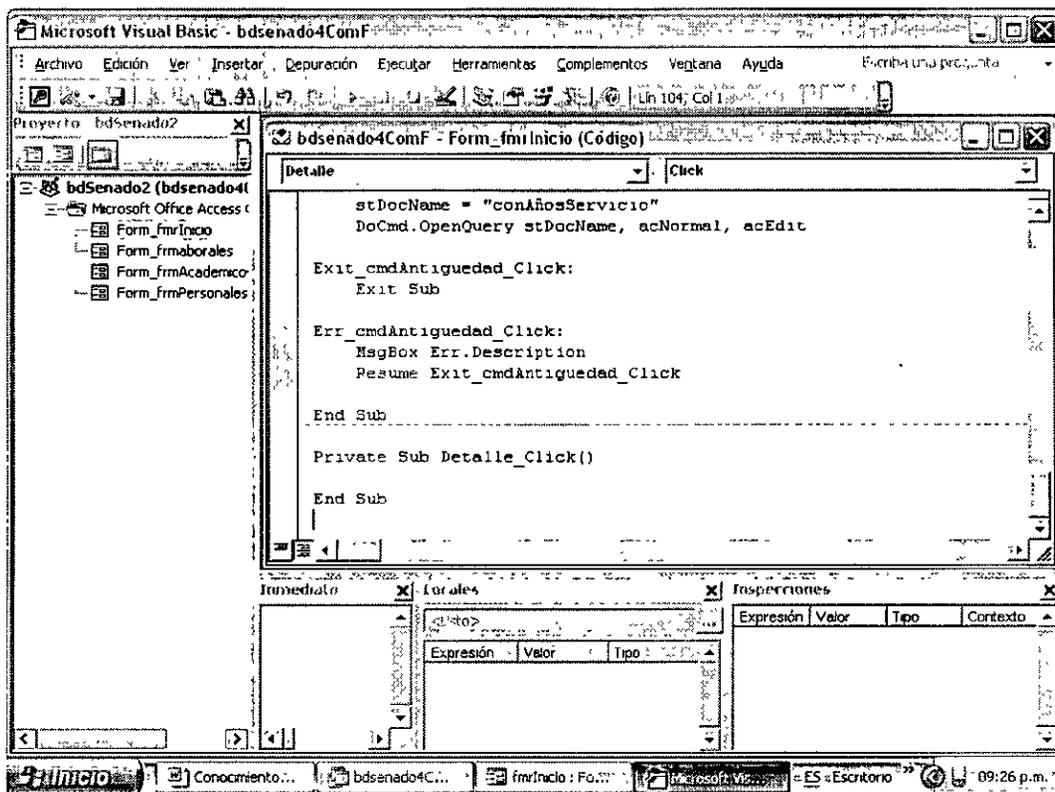
Instructor: Ing. Otelio Galicia Cedillo
DELEGACIÓN MIGUEL HIDALGO
DICIEMBRE 2004

Anexo C

Conocimientos básicos sobre VBA

El editor de Visual Basic

La ventana del editor de *Visual Basic* ya la conoces: es el entorno de desarrollo VBA en que se crea el código VBA.



En la ventana del editor de *Visual Basic* se pueden ver de manera estándar cuatro subventanas:

Elemento	Descripción
Ventana de código	En esta ventana se crea el programa
Ventana Inmediato	En esta ventana se pueden introducir comandos y ejecutarlos directamente, por ejemplo para probar el programa.
Explorador de proyectos	En esta ventana se muestra una lista con todas las partes del objeto actual de la programación, lo que significa tanto módulos como el código para formularios e informes.
Propiedades	En esta ventana se muestran las propiedades del objeto para el que se está escribiendo el código.

A través del menú *Ver* se pueden activar cuando sea necesario:

Elemento	Descripción
Ventana Locales	En esta ventana se puede ver el contenido de las variables.
Ventana Inspecciones	En esta ventana se pueden inspeccionar diversas variables.

Las seis ventanas mencionadas se pueden modificar tanto en orden como en tamaño, y también se pueden cerrar y activar desde el menú *Ver*. Tal como habrás observado, la primera ventana con la que tendrás que trabajar sobre todo es la ventana de código, que es donde se escribe el código del programa.

Vista de la ventana de código

En la parte inferior izquierda de la ventana de código verás dos pequeños botones que llevan por nombre *Ver procedimiento* y *Ver módulo completo* y que sirvan para cambiar, respectivamente, a estas dos vistas.

En la vista de procedimiento, la ventana de código sólo muestra un procedimiento cada vez. Con la combinación de teclas <Ctrl> + <Flecha hacia arriba> puede cambiar el procedimiento anterior, y <Ctrl> + <Flecha hacia abajo>, al procedimiento siguiente.

Programar en Access

En principio, un programa no es más que un conjunto de comandos ordenados de tal manera que permiten resolver determinadas tareas al ejecutarlos.

Al programar con VBA, la tarea que ha de resolver el programa se divide en varias partes o subtareas. Para resolver una de estas tareas parciales, los comandos necesarios se agrupan para formar un procedimiento. Durante el desarrollo del programa, cada procedimiento va

resolviendo la tarea que le corresponde. Y la tarea global queda resuelta finalmente por la ejecución conjunta de todos los procedimientos.

Qué se puede programar

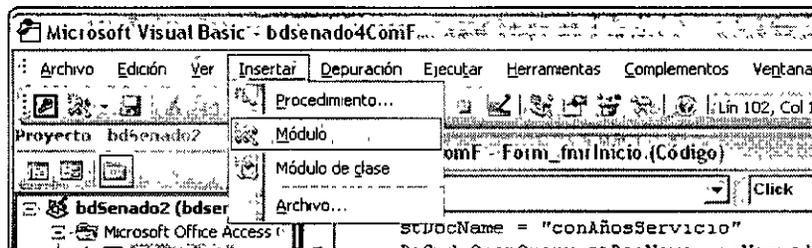
Al igual que las macros, el código VBA también puede vincular directamente al resultado de un formulario o de un informe, o utilizarlo de manera independiente para activar formularios e informes.

Code behind Forms

Es el código asignado directamente a un formulario (conocido también como Code behind Forms). Y es posible que te hayas dado cuenta de que ese código VBA no se puede activar directamente en la ventana de la base de datos de Access. Esto diferencia al código claramente de las macros, que sí se pueden iniciar directamente en la ventana de la base de datos de Access, aún cuando estén vinculadas a un formulario.

Módulos independientes

Para crear un módulo nuevo que también sea independiente:



- Activa simplemente la opción de menú *Insertar/Módulo*, después de lo cual se abrirá el editor VBA.
- En el editor VBA, activa la opción de menú *Insertar/Procedimiento* y asigne al nuevo procedimiento el nombre *ToscanaTest*.
- Después de confirmar el nombre pulsando el botón *Aceptar*, escribe el procedimiento: *Msgbox ("Escriba un mensaje")*
- Guarda el módulo con el nombre de *Módulo1*.

Si ahora pulsas la entrada *Módulo1* en la ventana de la base de datos, en lugar de ejecutarse el código del programa, se mostrará el procedimiento en el editor VBA. Para que se ejecute el procedimiento:

- Escribe *NombreTes* en la ventana *Inmediato* y pulsa a continuación la tecla <Entrar>.

En la pantalla se mostrará entonces el cuadro de mensaje (messagebox) que definió en el procedimiento *NombreTes*. Otro método más sencillo para probar un procedimiento consiste en posicionar el puntero del ratón en algún lugar dentro del procedimiento y pulsar a continuación la tecla de función <F5>, o pulsar directamente el botón *Ejecutar Sub/UserForm*

de la barra de herramientas



Cómo se programa

Un procedimiento siempre empieza con la línea *Sub name ()* y termina con *End Sub*. *Sub*., *End*, *Private*, etc. Son expresiones especiales de *Visual Basic*, conocidas también como palabras clave. Estas palabras se escriben automáticamente con mayúscula y se muestran en pantalla de color azul.

Escriba siempre con minúscula las palabras clave

Cuando salgas de una línea, todas las palabras clave de esa línea se deberían escribir automáticamente con las letras mayúsculas y minúsculas que corresponda. En caso de no suceder así, eso significa que te has equivocado al escribir alguna de las palabras clave.

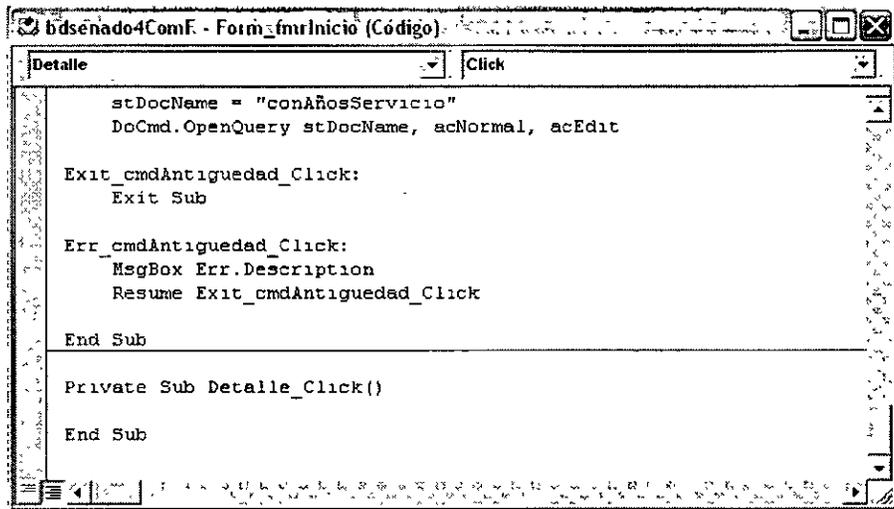
En los procedimientos que se crean para un elemento de control, el nombre se asigna automáticamente. Es un nombre formado por el nombre del elemento de control y el nombre del evento que se ha de producir para que se active el procedimiento. Por ejemplo: *Private sub cmdCerrar_Click ()* para el procedimiento de evento correspondiente a pulsar una vez con el ratón el botón *cmdCerrar*.

En los procedimientos que no están vinculados directamente a un elemento de control, el nombre se puede elegir, en principio, libremente. En caso de elegir un nombre formado por varias palabras, es conveniente escribir en mayúscula cada una de las palabras

Las líneas con las instrucciones en sí, se encuentran entre la primera y la última línea de procedimiento, que señalan el principio y el final del mismo. Un programa tanto puede estar formado por una sola línea como por cien.

Sangrar las líneas de instrucciones

Para escribir el programa con mayor claridad posible, lo normal es sangrar al mismo nivel las líneas que van juntas dentro del programa. Se sangra, por ejemplo, todo lo que queda dentro de las líneas *Sub name ()* y *End Sub*. También se han sangrado las líneas con instrucciones que quedan dentro de una estructura *If ...End If*. A veces se dice que las estructuras como *Sub...End Sub*, *If...End If*, así como los bucles encierran (forman un paréntesis) el código de programa que queda dentro de ellas



Líneas de comentario

```
Private Sub cmdPersonales_Click()
On Error GoTo Err_cmdPersonales_Click
```

```
Dim stDocName As String ' Declara la variable de tipo cadena
Dim stLinkCriteria As String
```

```
stDocName = "frmPersonales"
DoCmd.OpenForm stDocName, , , stLinkCriteria
```

```
Exit_cmdPersonales_Click:
Exit Sub
```

Para que lo que haz programado hoy lo puedas entender también al cabo del tiempo, es conveniente insertar líneas de comentario entre las líneas de instrucciones. Algunos programadores opinan incluso que un buen programa contiene tantas líneas de comentarios como de instrucciones.

Cuando quieras insertar una línea de comentario, debes empezar la línea con una comilla alta ('). Al salir de esa línea, se representará en verde. Los comentarios breves también se pueden escribir al final de una línea de instrucción, ya que todo lo que viene después de una comilla alta se considera comentario.

Líneas continuadas

Si una línea es tan larga que no se puede mostrar entera en la ventana de código, la instrucción correspondiente se puede dividir en dos o más líneas. Lo importante en ese caso es terminar cada línea (que continúe en la siguiente) con un espacio en blanco y un guión de subrayado “_”.

Para una mejor comprensión de las variables

Las variables son como si se tratara de un práctico portapapeles. En general, las variables se utilizan en los programas para guardar en ellas valores, cadenas de caracteres y otros tipos de datos para poder acceder a ellos más adelante en el programa.

Dentro del programa, cada variable tiene asignado un nombre y es a través de ese nombre es como se accede después al contenido de la variable. Una variable puede contener números, fechas, horas, textos e imágenes.

Las variables tienen un tiempo de duración limitado. Por regla general, conservan su valor, como máximo, hasta que finaliza el procedimiento al que pertenecen.

Declarar variables

Las variables que se utilizan en un programa se deben declarar al principio del mismo y definir a la vez el tipo de dato que utilizan. Al declarar las variables se está reservando, por así decir, el espacio en memoria que necesita la variable. Si bien esto no es absolutamente necesario, ayuda a prevenir errores.

Las variables se pueden declarar básicamente de dos maneras distintas; explícita e implícitamente. En la declaración explícita la variable se declara al principio del programa con la expresión *Dim nombre de la variable*. La declaración implícita, en cambio, se produce en el momento en que aparece el nombre de la variable por primera vez en el programa, sin necesidad de hacer nada más.

Esto último, que puede parecer como una facilidad en el trabajo, no siempre lo es. El problema que presenta es que cada variante de la sintaxis de una variable se considera automáticamente como una variable nueva. Si te equivocas una vez al escribir la variable, automáticamente tendrás dos variables declaradas con nombres casi idénticos.

Por eso es recomendable activar el cuadro de control *Requerir declaración de variables* de la ficha *Editor* (opción del menú *Herramientas/opciones*) Eso obligará a declarar las variables explícitamente en los nuevos proyectos. Podrás reconocer esta opción porque el código de cada nuevo proyecto empezará con la línea *Option Explicit*.

Si en el programa se utilizan variables sin haberlas declarado previamente, el programa no reconocerá el nombre de las variables y emitirá inmediatamente un mensaje de error. Esto ayuda a corregir fácilmente los posibles errores cometidos al escribir el nombre de las variables.

Asignar nombre a las variables correctamente

Los nombres asignados a las variables deben ser cortos, significativos, y fáciles de recordar —en la medida de lo posible. Además, el nombre de una variable debe empezar siempre por una letra, su longitud no puede ser de más de 256 caracteres, y no puede contener el carácter “_” ni ningún carácter especial. Si el nombre de una variable está formado por varias palabras, cada palabra debe ir escrita en mayúscula, de acuerdo con la convención establecida. Eso facilita considerablemente la lectura del nombre, por ejemplo: `txt_Sueldo`.

Nota

No utilices ninguna de las palabras clave de Visual Basic

Hay una norma que debes observar estrictamente; no utilizar como nombre de sus variables ninguna palabra clave ni ningún nombre de objeto ni de propiedad.

Tipos de datos para las variables

Además de reservar el espacio en memoria con la declaración, también es conveniente definir desde el principio qué tipo de datos es el que utiliza la variable: si es un texto, un número, un valor indicando una fecha o un valor booleano. Si en un programa queremos guardar un texto en una variable, por ejemplo “Marcombo”, podemos declarar esta variable como siguiente:

```
Dim Editorial As String
```

Al añadir `As String` se está definiendo el tipo de datos “string” para la variable. Después ya puedes asignar el texto a la variable:

```
Editorial = “Anaya”
```

Escribiéndolo entre comillas.

La tabla siguiente recoge los tipos de datos más importantes en las variables de *Visual Basic*.

Tipo de datos	Tipo de valores	Rango de valores
Byte	Números enteros	0...255
Integer	Números enteros	-32.768...32.767
Long	Números enteros	-2.147.483.648...2.147.483.647
Single	Números decimales	Números con un total de 8 cifras
Double	Números decimales	Números con un total de 16 cifras
Currency	Números decimales para moneda	15 cifras enteras y 4 decimales
Boolean	Valores de verdadero o falso	TRUE o FALSE
Date	Valores de fecha y hora	1-1-100 hasta 31-12-9999
String	Textos	Puede contener hasta 2 000 millones de caracteres
Variant	Puede contener cualquier tipo de valores	

Los tipos de datos indicados en la tabla no poseen todos la misma importancia. Así, para los números enteros, es decir, para los números sin cifras decimales, se emplea normalmente el tipo de datos *Integer*. Sólo cuando se tiene la absoluta certeza de que los números son siempre menores que 255, se utiliza el tipo de datos *Byte*, ya que ocupa mucho menos espacio en memoria. Y sólo cuando los números son mayores que 32.000 se utiliza el tipo de datos *Long*. Y cuando se utilizan números con cifras decimales, estos números se declaran normalmente como *Double*.

Si no indicas explícitamente el tipo de datos que deseas, la variable se guarda siempre como *Variant*. Las variables del tipo de datos *Variant* pueden contener cualquier tipo de valores: un texto, un número, una fecha, etc. La desventaja de este tipo de datos es que ocupa mucho más espacio en memoria que una variable declarada, por ejemplo, como fecha o como número. Por otra parte, definir el tipo de datos adecuado también sirve para evitar errores. Por ejemplo, si a una variable que se declaró como numérica se le asigna un texto, el programa emite un mensaje de error.

Las instrucciones *Dim* ocupan menos espacio en memoria

Si es necesario, puedes agrupar varias instrucciones *Dim* en una sola línea, separando las instrucciones mediante comas. Por ejemplo:

```
Dim strNombre As String, strCurso As String, dblImporte As Double
```

Al agrupar varias expresiones hay que tener en cuenta que todas las variables queden perfectamente declaradas, aunque eso suponga escribir dos palabras más. Con la instrucción:

```
Dim strNombre, StrCurso As String, dblImporte As Double
```

Estaríamos declarando *StrNombre* como *Variant*

Por otra parte, es conveniente ajustarse a las normas establecidas para los nombres de las variantes e indicar siempre el tipo de datos escribiendo al principio la abreviatura (combinación de letras) correspondiente. En la tabla siguiente se indican las abreviaturas correspondientes a cada tipo de datos.

Tipo de datos	Abreviatura	Ejemplo
Byte	byt	Dim byStatus As Byte
Integer	int	Dim intContador As Integer
Long	lng	Dim lngHabitantes As Long
Single	sng	Dim sngCambio As Single
Double	dbl	Dim dblImporte As Double
Currency	cur	Dim curImportePta As Currency
Boolean	f	Dim fMiembro As Boolean

Date	dat	Dim datFechaNacimiento As Date
String	str	Dim strNombre As String
Variant	var	Dim varDiversos As Variant

El tipo de dato Boolean

Los valores de la verdad (*False* y *True*) se gestionan con ayuda de variables del tipo *Boolean*. En los fragmentos de procedimiento que veremos a continuación se ha asignado el resultado de un cuadro de control a una variable del tipo *Boolean*.

En el programa se declara la variable *fPtaEuro* y se le asigna después el contenido del cuadro de control *chkPtaEuro*. El valor 1, que es el que devuelve el cuadro de control cuando se activa, se convierte automáticamente en el valor *true*. El valor 0, se convierte en *False*. La consulta, por lo tanto, es : *If fptaEuro = True Then*.

```
Private Sub cmdConvertir_Click ()
    ...
    Dim fPtaEuro As Boolean

    ...
    If fPtaEuro = True Then

        Else    ...
    End If    ...
    ...
End Sub
```

Los valores de la verdad en las consultas If

Si utilizas un valor de verdad en una consulta *If*, puede acortar la expresión escribiendo *If txt_Sueldo Then*. Esta condición se cumple cuando el valor de *fptaEuro* es *True*.

Supongamos que quieres asignarle al valor de verdad *fCaso* el valor *True* cuando se cumpla una condición determinada, por ejemplo *intSelección = 4*. En este caso podrías escribir:

```
If intSelección = 4 Then fCaso = True
```

O, más corto

```
fCaso = (intSelección ==4)
```

En esta última expresión, lo primero que se evalúa es el paréntesis. Si *intSelección* = 4, a *fCaso* se le asigna el valor *True*; si el resultado de la evaluación del paréntesis es que la igualdad no se cumple, a *fCaso* se le asigna entonces el valor *False*.

El tipo de dato Date

Los valores de fecha pueden cubrir el intervalo de tiempo que va desde el 1 de Enero del año 100 hasta el 31 de Diciembre del año 9999. las horas pueden ir desde las 00:00:00 hasta las 23:59:59.

Si quieres asignarle una fecha determinada a la variable *datFecha-Entrada*, del tipo *Date*, la fecha deberá ir escrita entre caracteres "#". La fecha no se puede introducir de la manera que estamos acostumbrados escribirla. Al introducir un valor de fecha en un programa se ha de utilizar uno de los siguientes formatos:

```
datFechaEntrada = # 1 2 99#  
datFechaEntrada = # 1 February 99#  
datFechaEntrada = # 1 Feb 99#  
datFechaEntrada = #1, 2, 99#
```

Al salir de la línea, la forma elegida se convierte automáticamente a la notación americana:

```
datFechaEntrada = #2/ 1/ 99#
```

Fijate en que el orden del mes y del día están intercambiados. En la notación americana, primero se escribe el mes y después el día.

La muestra de una fecha

Se produce automáticamente en la forma breve seleccionada en las configuraciones del país. Si deseas utilizar otro formato para la fecha, utiliza la función *Format* tal como lo hiciste anteriormente para dar formato a la fecha que se mostraba en el formulario. Con esta función se han de utilizar las abreviaturas americanas:

yyyy	para el año
q	para el trimestre
m	para el mes
d	para el día
y	para el día del año
w	para el día de la semana
ww	para la semana

para una fecha larga del tipo *Miércoles*, 08 de Agosto del 2004, por ejemplo, utiliza "dddd, dd.mmm.yyyy".

Si deseas utilizar la hora, también debe escribirla entre caracteres "#". Para indicar la hora sí puedes utilizar la notación habitual: `datHora = #7:00#` o `datHora = #19:00#`. La hora así indicada se convierte automáticamente en `datHora = #7:00:00 AM#` y `datHora = # 7:00:00 PM#` respectivamente. La emisión corresponderá al formato de hora que se haya seleccionado en la configuración del país.

Muestra de las horas

La forma utilizada de manera estándar para representar las horas es: "hh:mm:ss". Para dar un formato distinto a la hora hay que utilizar h para las horas, n para los minutos y s para los segundos. Por lo tanto, si quieres mostrar la hora sin indicación de segundos, utiliza simplemente "hh:nn" (utiliza la n para el formato de los minutos, pues así evitarás confusiones con el formato del mes)

El tipo de dato String

Las variables que han de contener texto se deberían declarar con el tipo de datos *String*. Para transmitir un texto a una variable, el texto se escribe entre comillas.

String vacío

Un string vacío –una cadena de caracteres que no contiene ningún carácter- se representa simplemente mediante "".

Constantes

En principio, una constante es un caso especial de variable. Las constantes como:

```
Const conPi = 3.14159265
Const conEuro = 166.386
Const conNúmero As Integer = 20
```

No cambian a lo largo del tiempo de ejecución de un programa. Las constantes llevan como distintivo el prefijo "con".

¿Por qué se utilizan las constantes? Imagínate que se utiliza el número 11 en 20 lugares distintos de su programa, para indicar con él el número de países que adoptarán inicialmente el Euro como moneda. Pasado el tiempo, se añaden países nuevos a esta cifra. En ese caso, en todos los lugares en que aparece el 11 deberían cambiar el 11 por un 16, por ejemplo, si fueran 16 los países que hubieran introducido el euro. Esta substitución, naturalmente, sólo se ha de realizar en las posiciones correctas, y no en los lugares en que el 11 exprese algo distinto de los países del Euro.

El esfuerzo que supondría realizar los 20 cambios se puede simplificar considerablemente utilizando constantes. Define la constante:

```
Const conPaísesEuro = 11
```

Y utilízala en el programa en lugar del número 11. así, cuando tengas que ampliar a 16 el número de países del Euro, sólo tendrá que hacer una modificación: la de la definición de la constante.

Arrays

Los valores de tipo similar suelen asignarse en vectores –también llamados arrays. Para acceder a los valores de un array se puede utilizar el mismo nombre para todos, pero indicando la posición de cada uno –en forma de índice- dentro del array.

Al declarar un array se indica también el tamaño (número de elementos) del mismo. Un array con tres componentes se declararía con una línea

```
Dim adblIVA (29 As Double
```

Dentro de los arrays se empieza a numerar por 0. por lo que *adblIVA* puede constar de hasta tres elementos, a saber: *adblIVA (0)*, *adblIVA (1)* y *adblIVA (2)*

Para señalar el nombre de la variable que se trata de un array, se antepone al nombre la letra “a”, inicial de la palabra inglesa *array*.

Los diferentes valores de un array se asignan con ayuda del índice. Supongamos que trabajas con tres tipos distintos de IVA: 0%, 7% y 16%. En ese caso deberás escribir:

```
adblIVA (0) = 0  
adblIVA (1) = 0.07  
adblIVA (2) = 0.16
```

Ejemplo: calcular el IVA con arrays

En un formulario *frmIVA*, por ejemplo, podríamos haber incluido el grupo de opción *fralIVA*, que puede tomar valores entre 0 y 2, según el cuadro de opción que se pulse. Tenemos además los cuadros de texto *txtNeto* para introducir los valores netos, y *txtBruto* para la emisión del valor bruto calculado. Ambos campos tienen asignado el formato *Moneda*. Pulsando una vez el botón *cmdCalcular* se indica el cálculo del IVA seleccionado. Para ello hemos vinculado el siguiente procedimiento al evento *Al hacer clic*:

```
Private Sub cmdCalcular_Click()  
  
    Dim dblBruto AS Double, dblNeto As Double  
    Dim intIndex As Integer  
    Dim adbIIVA (2) As Double  
  
    'Asignar valores de IVA  
    adbIIVA (0) = 0  
    adbIIVA (1) = 0.07  
    adbIIVA (2) = 0.16  
  
    'Para garantizar que no se transmite Null  
    If IsNull (txtNeto) Then  
        txtNeto = 0  
    End If  
  
    intIndex = fraIVA  
    dblNeto = txtNeto  
    dblBruto = dblNeto * adbIIVA (intIndex)  
    txtBruto = dblBruto  
  
End Sub
```

La función Array

Para llenar los arrays se puede utilizar la función *Array*. La ventaja de esta función es que sus valores se asignan de forma de lista, por lo que no hay que escribir tanto. Los valores de un *Array* siempre son del tipo *Variant*.

Un *Array* se ha de declarar siempre, pero no es necesario dimensionarlo. Por lo tanto, sería suficiente con escribir:

```
Dim avarIVA As Variant
```

Para darle valores al array, escribe:

```
avarIVA = Array(0, 0.07, 0.16)
```

Los diferentes valores de la lista se separan entre sí mediante comas.

Validez de las variables

¿Qué tiempo de validez tiene una variable?

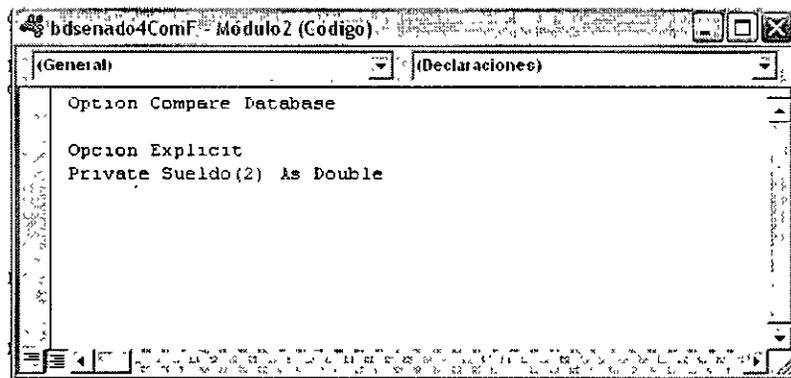
Validez a nivel de procedimiento

Si una variable se declara dentro de un procedimiento, solo tiene valor dentro de él: se dice entonces que tiene validez a nivel de procedimiento. Este tipo de variables se conocen también como variables locales.

Fuera del procedimiento en que ha sido declarada, la variable es desconocida, por lo que para poder utilizarla en otro procedimiento se tendrá que declarar también en él. De todos modos, nunca se puede acceder al valor de una variable desde otro procedimiento.

Validez a nivel de módulo privado

Si necesita una variable en varios procedimientos y quiere acceder a ella desde todos ellos, la variable se tendrá que declarar a nivel de módulo. Entonces tendrá validez en todos los procedimientos de un formulario, por ejemplo. Este tipo de variables se declaran directamente a continuación de la línea *Option Explicit*, antes del procedimiento en sí y con el añadido *Private* en lugar de *Dim*. (También es posible utilizar *Dim* en la declaración de una variable a nivel de módulo privado, pero *Private* deja mucho más claro el nivel de validez de la variable, por lo que es preferible utilizar *Private* en lugar de *Dim*.)



Para indicar la validez a nivel de módulo se ha de escribir una "M" delante de la abreviatura correspondiente al tipo de variable.

Validez a nivel de módulo público

Para que una variable no tenga validez sólo en un módulo sino en una aplicación completa, se ha de declarar como a nivel de módulo público. De ese modo se puede acceder a ella tanto desde los formularios e informes como de los demás módulos. Las variables públicas se declaran en el área de declaración de un módulo con la palabra *Public* y, para identificarlas, se añade la letra "G" (de global). Estas variables se conocen también como variables globales.

Variables con el mismo nombre

¿Qué sucede si se utiliza una variable que ha sido declarada a nivel de procedimiento –variable local- en dos procedimientos distintos pero con el mismo nombre? En este caso no hay ningún problema, pues las variables locales solo tienen validez dentro del procedimiento en que han sido declaradas. Cuando se activa un segundo procedimiento y se utiliza una variable con el mismo nombre que la anterior, el procedimiento no tiene en cuenta para nada a la otra variable. El valor de esa variable es igual a *Null* –a menos que en el segundo procedimiento se le haya asignado explícitamente un valor.

¿Qué sucede si se declara una variable, con el mismo nombre, por un lado a nivel de módulo y , por otro, localmente a nivel de procedimiento? (Si utilizas la “M” indicativa de las variables de módulo, en realidad es imposible que pase). En un caso así, dentro del procedimiento en que se ha declarado la variable local sólo tiene validez ésta. Eso significa que el valor de la variable local es igual a *Null*, independientemente del valor de la variable de módulo, mientras en el procedimiento no se le asigne un valor diferente.

Operadores

Si quieres programar utilizando formular, deberás aprender a manejar los operadores que pueden utilizarse en cada una de ellas. Algunos de los operadores incluidos en el siguiente resumen los hemos visto trabajar con Access XP en el curso.

Por ejemplo: +, / y *.

¿Qué operadores hay?

A grosso modo, los operadores se pueden dividir en operadores aritméticos (o matemáticos), lógicos, relacionales y de encadenamiento o concatenación.

Operadores lógicos

En las condiciones, además de los operadores comparativos también se pueden utilizar operadores lógicos tal como puedes ver en la tabla siguiente:

Operador	Ejemplo	Significado
And	intNúmero > 10 And intNúmero < 90	intNúmero ha de ser mayor que 10 menor que 90, es decir, ser un número entre 10 y 90.
Or	intNúmero < 10 Or intNúmero > 20	intNúmero debe ser menor que 10 o mayor que 20
Nor	Nor (intNúmero < 10)	intNúmero no debe ser menor que 10, es decir, debe ser mayor o igual que 10

Dos condiciones unidas mediante *Or* se cumplen en conjunto cuando se cumple una de ellas. Por el contrario, dos condiciones unidas con *And* sólo se cumplen en conjunto cuando se cumplen ambas a la vez. Otra posibilidad es la de utilizar varios operadores lógicos dentro de una misma condición.

por ejemplo:

Not (intNúmero <10) And Not (intNúmero >90)

Puesto que no siempre es fácil saber cómo funciona cada operador, en la tabla siguiente ofrecemos un resumen de la combinación y resultados de los diferentes operadores. La palabra *True* significa en la tabla que la condición en cuestión es verdadera, es decir, que se cumple. Si el resultado es igual a *True*, eso significa que la combinación de las dos condiciones también se da como verdadera.

Condición 1	Operador	Condición 2	Resultado
True	And	True	True
True	And	False	False
False	And	True	False
False	And	False	False
True	Or	True	True
True	Or	False	True
False	Or	True	True
False	Or	False	False
	Not	True	False
	Not	False	True

Operadores de encantamiento o concatenación

Hay dos operadores de encantamiento: "+" y " & ". Con el operador "+" se encadenan expresiones generales y valores entre sí, como *intNúmero + intCrecimiento*. Este operador es el de adición.

El operador " & ", por el contrario, encadena series de caracteres, como por ejemplo:

```
strNombreCompleto = strNombre & strApellido
                    o
strLocalidad = "85586" & "Badalona"
```

Orden de evaluación de los operadores

Los operadores no ocupan todos el mismo valor jerárquico. Por ejemplo, en la expresión *dblSumaParcial1 + dblSumaParcial2 / dblCambio*, primero se ha de realizar la división y, después, la suma.

De manera general, primero se evalúan los operadores aritméticos, por ejemplo lo operadores de encadenamiento entre cadenas de caracteres, después los operadores relacionales y, finalmente, los lógicos.

Si una expresión contiene varios operadores de la misma jerarquía, éstos se evaluarán por orden de izquierda a derecha.

La tabla siguiente indica el orden jerárquico que ocupan los operadores aritméticos entre sí:

^	Exponencial
-	Negación
* o /	Multiplicación y división
\	División de números enteros
Mod	Valor residual
+ o -	Adición y substracción

Los operadores relacionales tienen todos el mismo valor jerárquico. El orden jerárquico de los operadores lógicos es el siguiente:

Not
And
Or

¿Te gustaría asignar un orden distinto en la evaluación?

En este caso, todo lo que tienes que hacer es escribir paréntesis. Si en el ejemplo anterior deseas que se realicen primero las sumas parciales y, después, la división, sólo tienes que escribir (dbfSumaParcial1 + dbfSumaParcial2) / dbfCambio.

Consultas y ramificaciones condicionadas

Con ayuda de consultas se puede conseguir que sólo se ejecuten determinadas partes del programa. Para ello debes escribir condiciones que regulen qué partes se han de ejecutar y cuáles no al recorrer una vez el programa. Las dos estructuras más importantes en este caso son las consultas *If* y las instrucciones *Select Case*.

Consultas IF

Con ayuda de una consulta *If* puedes conseguir que una determinada parte del programa sólo se ejecute cuando se cumpla una condición concreta. La forma más sencilla de una consulta *If* es:

If Condición Then Instrucción

Una instrucción *If* de una sola línea está formada, como la anterior, por las palabras clave *If* y *Then*. Entre medias se indica una condición. En las condiciones se puede utilizar los operadores explicados anteriormente. Si la condición se cumple (es verdadera), se ejecuta la instrucción indicada detrás de la palabra *Then*, de lo contrario, no se ejecuta.

Una condición así la podrías utilizar, por ejemplo, en:

```
If intEjemplaresVendidos > 10000 Then dblDescuento = 0.03
```

Las consultas añadidas, de varias líneas, permiten mayor flexibilidad:

```
If Condición Then
    [Instrucción]
[Elseif Condición 1 Then
    [De lo contrarioSiCondiciones]]
[Elseif Condición 2 Then
    [De lo contrarioSiCondiciones]]
[Elseif Condición 3 Then
    [De lo contrarioSiCondiciones]]
...
[Else
    [OtrasCondiciones]]
End If
```

En esta notación se escriben entre corchetes las partes que no son imprescindibles, es decir, las que son opcionales. Por eso una instrucción como la que hemos visto también podría reducirse a:

```
If Condición Then
    Instrucciones
Ende If
```

O incluso a:

```
If Condición Then
    Instrucciones
Else
    OtrasInstrucciones
End If
```

Ya que las partes *Else* y *Elseif* son ambas opcionales. En un bloque *If* se pueden incluir todas las partes *Elseif* que se desee. Sólo hay que observar que después de una línea *Else* ya no puede haber ninguna instrucción *Elseif*.

Al ejecutarse el programa, los bloques *If* se procesan de arriba hacia abajo. En primer lugar se comprueba la condición que viene detrás de *If*. Si esta condición se cumple, esto es, si su valor es *True*, se ejecutan las instrucciones que la siguen. Una vez ejecutadas, el programa se sigue procesando con la instrucción *End If*. Si no es ese el caso, se comprueban todas las

partes *Elseif* –de haberlas. Al final, si ninguna de las condiciones devolvió el valor *True*, se ejecuta la instrucción *Else*.

Sangrías

Cuando se trabaja con instrucciones *If* es muy útil sangrar las líneas, como puedes apreciar en la forma general. De ese modo es mucho más fácil descubrir, por ejemplo, dónde falta un *End If*.

Con las instrucciones *If* también es posible anidar varias consultas *If* una dentro de otra.

Las Instrucciones Select Case

Las ramificaciones se pueden introducir con ayuda de las estructuras *If* también con una instrucción *Select Case*. A menudo, cuando las estructuras *If* tienen varios bloques *Elseif*, resulta mucho más clara una estructura con *Select Case*.

En general, una instrucción *Select Case* se puede expresar a través de:

```
Select Case Variable
  [Case Expresión 1
    [Instrucciones 1]
  [Case Expresión 2
    [Instrucciones 2]
  ...
  [Case Expresión n
    [Instrucciones n]]]]
[Case Else
  [Otras Instrucciones]
End Select
```

Una instrucción de este tipo viene siempre escrita entre las palabras clave *Select Case* y *End Select*. En la primera línea se indica una variable cuyo valor se comprueba en las líneas *Case* posteriores. Si la expresión que viene después de *Case* es verdadera para la variable, se procesan las instrucciones siguientes. Si la expresión no es verdadera, se comprueba la siguiente expresión *Case*. Si ninguna de las condiciones que vienen a continuación de *Case* es verdadera, se ejecutan las otras instrucciones (otras) que hay después de la entrada *Case Else*.