

# Didáctica de Nuevas Tecnologías en la E.S.O.

Bases de Datos y Matemáticas  
Vol. I



**Didáctica de Nuevas Tecnologías en la E.S.O.-  
Bases de Datos y Matemáticas  
(Vol I)**

Francisco Luis Flores Gil

© 2008. Francisco Luis Flores Gil  
Portada diseño y difusión de la obra: Íttakus



Licencia Creative Commons

Edición cortesía de [www.publicatuslibros.com](http://www.publicatuslibros.com). Debe reconocer los créditos de la obra de la manera especificada por el autor o el licenciador (pero no de una manera que sugiera que tiene su apoyo o apoyan el uso que hace de su obra). No puede utilizar esta obra para fines comerciales. Si altera o transforma esta obra, o genera una obra derivada, sólo puede distribuir la obra generada bajo una licencia idéntica a ésta. Al reutilizar o distribuir la obra, tiene que dejar bien claro los términos de la licencia de esta obra. Alguna de estas condiciones puede no aplicarse si se obtiene el permiso del titular de los derechos de autor. Nada en esta licencia menoscaba o restringe los derechos morales del autor.

Publicatuslibros.com es una iniciativa de:



Íttakus, sociedad para la información, S.L.  
C/ Sierra Mágina, 10.  
23009 Jaén-España  
[www.ittakus.com](http://www.ittakus.com)



# Índice

1	Introducción .....	6
2	¿Qué es SQL? .....	7
3	Consultas de datos con SQL y modificación de datos.....	8
3.1	Consultas con SQL .....	8
3.1.1	Cláusula <i>SELECT</i> y <i>FROM</i> .....	8
3.1.2	Cláusula <i>WHERE</i> .....	8
3.1.3	Operadores de la cláusula <i>WHERE</i> .....	9
3.1.4	Operadores <i>BETWEEN</i> .....	9
3.1.5	Operador <i>LIKE</i> y los caracteres comodín .....	9
3.1.6	Operador <i>IN</i> .....	9
3.1.7	Cláusula <i>ORDER BY</i> .....	10
3.1.8	Combinar tablas relacionadas en una consulta.....	10
3.1.9	Utilizar combinaciones externas para devolver más datos .....	11
3.1.10	Cláusula <i>UNION</i> y <i>MINUS</i> .....	11
3.1.11	Cláusula <i>AS</i> .....	12
3.1.12	Consultas que agrupan y resumen datos.....	12
3.1.13	Subconsultas .....	13
3.2	Modificación de datos .....	13
3.2.1	Actualización.....	13
3.2.2	Eliminación.....	14
3.2.3	Inserción .....	14
4	Inicio Access.....	15
5	Crear un base de datos.....	16
6	Crear una tabla .....	17
7	Introducir datos en la tabla.....	20
8	Modificar la estructura de una tabla .....	21
9	Crear un formulario y utilizarlo para modificar los datos existentes en la base de datos ...	22
9.1	Crear un formulario .....	22
9.2	Modificar e introducir datos de una tabla .....	23
9.2.1	Modificar los datos de una tabla utilizando un formulario.....	23
9.2.2	Introducir datos en una tabla usando un formulario .....	24
10	Modificar un formulario.....	25
11	Crear un consulta.....	27
12	Filtros en un formulario .....	32
13	Imprimir mediante informes.....	33
14	Creación de la base de datos II.....	35
15	Relaciones .....	38
16	Crear una consulta.....	41
16.1	Consulta 1 .....	41
16.2	Consulta 2.....	43
16.3	Consulta 3.....	44

17	Crear formulario .....	46
18	Crear informes .....	47
19	El autor .....	48

# 1. Introducción

En este libro se darán las nociones básicas en el manejo de SQL y de Access, destacando la relación entre las matemáticas y las bases de datos.

Internamente una base de datos esta fundada en nociones matemáticas. De aquí que sea recomendable que el departamento de matemáticas colabore en su creación.

SQL es un método estándar para manipular bases de datos. Los sistemas de bases de datos relacionales, como MS Access, lo implementan de diferentes maneras. Normalmente se utiliza para crear consultas que extraen datos de las bases de datos, aunque muchas instrucciones SQL realizan otras funciones sobre ellas, como crear tablas o relacionar campos.

Por todo ello es muy adecuado para aquellos profesores y alumnos de la E.S.O. que quieran aprender el funcionamiento de una base de datos.

En este libro explicaremos los puntos fundamentales de Access mediante prácticas que irán siendo explicadas paso a paso para su fácil comprensión.

## 2. ¿Qué es SQL?

Las siglas SQL (Standard Query Language) significan Lenguaje Normalizado de Consulta.

SQL es un método estándar para manipular bases de datos. Los sistemas de bases de datos relacionales, como MS Access, la implementan de diferentes maneras. Normalmente se utiliza para crear consultas que extraen datos de las bases de datos, pero la gran cantidad de instrucciones disponibles en SQL permiten realizar una gran cantidad de funciones de gran complejidad.

Las consultas SQL permiten recuperar datos de una tabla de base de datos, establecer correspondencias entre los datos relacionados de varias tablas, y manipular la estructura de las bases de datos.

Su aprendizaje no solo sirve para esta aplicación sino, también, para todas las existentes en el mercado que soporten este lenguaje ya que es un lenguaje estándar por haberse visto consolidado por el Instituto Americano de Normas (ANSI) y por la Organización de Estándares Internacional (ISO).

## 3. Consultas de datos con SQL y modificación de datos

### 3.1 Consultas con SQL

#### 3.1.1 Cláusula SELECT y FROM

En el núcleo de cualquier consulta de recuperación de datos se encuentra la cláusula SELECT. Esta cláusula significa “devolver todos los campos encontrados en la fuente especificada”. La cláusula no está completa sin la cláusula FROM.

La cláusula FROM indica la fuente de datos que utilizará la consulta para recuperar los registros; esta fuente puede ser una tabla u otra consulta almacenada.

Ejemplo:

```
SELECT * FROM CLIENTES
```

#### 3.1.2 Cláusula WHERE

Una cláusula WHERE indica al motor de base de datos que límites de registros recupera, en función de los criterios indicados. Un criterio es una expresión que evalúa una condición como verdadera o falsa.

Ejemplo:

```
SELECT * FROM CLIENTES WHERE NOMBRE = "Pepe"
```

Para crear cláusulas WHERE más sofisticadas se usan además dos o más criterios con los operadores lógicos AND y OR.

Ejemplo:

```
SELECT * FROM CLIENTES WHERE NOMBRE = "Pepe" AND (APELLIDO1 = "SANCHEZ" OR APELLIDO1 = "SANCHO")
```

### 3.1.3 Operadores de la cláusula WHERE

Son los definidos a continuación.

<	Menor que
<=	Menor igual a
>	Mayor que
>=	Mayor igual a
=	Igual a
<>	No igual a
BETWEEN	Dentro de un intervalo de valores
LIKE	Coincide con un modelo
IN	Contenido en una tabla de valores

### 3.1.4 Operadores BETWEEN

El operador BETWEEN devuelve todos los valores de registros que estén entre los límites especificados.

Ejemplo:

```
SELECT * FROM PRODUCTOS WHERE UNIDADESENEXISTENCIA  
BETWEEN 100 AND 4000.
```

### 3.1.5 Operador LIKE y los caracteres comodín

El operador LIKE hace coincidir los registros con un modelo especificado. Con frecuencia este modelo es un carácter comodín, como los signos \* o ¿.

El carácter \* indica la coincidencia parcial

Ejemplo:

```
SELECT * FROM CLIENTES WHERE APELLIDO1 LIKE "G*".
```

```
SELECT * FROM CLIENTES WHERE APELLIDO1  
LIKE "[A-C]*".(Devuelve todos los registros cuyos primer apellido empiecen por  
la A y la C).
```

### 3.1.6 Operador IN

El operador IN se utiliza para recuperar registros que coinciden con una lista de valores.

Ejemplo:

```
SELECT * FROM CLIENTES WHERE APELLIDO1 IN ('BRITO', 'GUTIERREZ').
```

### 3.1.7 Cláusula ORDER BY

La cláusula ORDER BY indica al motor de base de datos que ordene los registros recuperados. Puede ordenarse ascendentemente o descendientemente.

Ejemplo:

```
SELECT NOMBRE, APELLIDO1 FROM CLIENTES WHERE APELLIDO1 LIKE "G*" ORDER BY NOMBRE, APELLIDO1.
```

```
SELECT NOMBRE, APELLIDO1 FROM CLIENTES WHERE APELLIDO1 LIKE "G*" ORDER BY NOMBRE ASC.
```

### 3.1.8 Combinar tablas relacionadas en una consulta

Las combinaciones se utilizan para recuperar información relacionada de más de una tabla.

Para crear una combinación de una consulta, se debe asignar las claves principal y externa de las tablas implicadas en la combinación.

Consideremos las siguientes tablas:

- Tabla CLIENTES con los siguientes campos  
ID, NOMBRE, APELLIDO1, APELLIDO2.
- Tabla PEDIDOS con los siguientes campos  
ID, CLIENTEID, CANTIDAD, F\_PEDIDO.

Puede ser que nos interese información de los pedidos de un cliente determinado.

Consideremos el siguiente ejemplo:

```
SELECT NOMBRE, APELLIDO1, CANTIDAD  
FROM CLIENTES, PEDIDOS  
WHERE CLIENTES.ID = PEDIDOS.CLIENTEID
```

Este código SQL devuelve información acerca de los clientes que tenga información en la tabla PEDIDOS.

También se puede utilizar la cláusula INNER JOIN. La sintaxis es diferente pero los datos que devuelve son los mismos.

```
SELECT NOMBRE, APELLIDO1, CANTIDAD  
FROM CLIENTES  
INNER JOIN PEDIDOS ON CLIENTE.ID = PEDIDOS.CLIENTEID;
```

La sentencia INNER JOIN no la admiten todas las bases de datos ya que es de más reciente creación.

La creación de las sintaxis INNER JOIN proporciona una ventaja importante: cuando se utiliza INNER JOIN para combinar tablas en una consulta, el conjunto de datos se puede actualizar. Por eso motivo, las consultas creadas con herramientas visuales

(especialmente MS Access) utilizaran la cláusula INNER JOIN para combinar dos tablas de una consulta.

### 3.1.9 Utilizar combinaciones externas para devolver más datos

Una combinación devuelve registros de dos tablas, en las que un valor de la clave principal de una tabla coincide con un valor de la clave externa de la tabla relacionada. Pero supongamos que desea devolver todos los registros de una parte de la combinación, existan o no registros relacionados. En este caso, se deberá utilizar una combinación externa.

Por ejemplo una consulta que muestre los clientes y pedidos, incluidos que no tienen pedidos.

```
SELECT NOMBRE, APELLIDO1, APELLIDO, CANTIDAD
FROM CLIENTES LEFT JOIN
PEDIDOS ON CLIENTES. ID = PEDIDOS. CLIENTEID
WHERE NOMBRE = 'LUIS'
```

Análogamente se podría utilizar RIGHT JOIN para mostrar todos los pedidos, hubiera o no clientes asociados.

Es muy habitual crear consultas en función de una combinación que muestra cero en lugar de nulos, especialmente cuando las consultas se utilizan como base de un informe. Para ello se reemplaza la referencia por un campo de cálculo.

En este caso se utiliza la función IIF. IIF utiliza tres argumentos: una expresión, el valor que devolverá si la expresión es verdadera y el valor que se devolverá si la expresión es falsa.

En el ejemplo anterior:

```
SELECT NOMBRE, APELLIDO1, APELLIDO, IIF(IsNull(CANTIDAD),0,
CANTIDAD) FROM CLIENTES LEFT JOIN BPEDIDOS ON CLIENTES. ID =
PEDIDOS. CLIENTEID WHERE NOMBRE = 'LUIS'.
```

### 3.1.10 Cláusula UNION y MINUS

La cláusula MINUS devuelve la concatenación de dos grupos de registros.

Por ejemplo:

```
SELECT NOMBRE, APELLIDO1
FROM CLIENTES
WHERE NOMBRE = 'LUIS'
UNION
SELECT NOMBRE, APELLIDO1
FROM CLIENTES
WHERE NOMBRE = 'PEPE'
```

La cláusula MINUS devuelve la 'resta' de los dos grupos de registros.

Por ejemplo:

```
SELECT NOMBRE, APELLIDO1
FROM CLIENTES
MINUS
SELECT NOMBRE, APELLIDO1
FROM CLIENTES
WHERE NOMBRE = 'PEPE'
```

### 3.1.11 Cláusula AS

Se pueden asignar un alias o cambiar el nombre de un campo en una consulta.

Existen dos motivos para ello:

- La tabla subyacente posee nombres de campos difíciles y se desea facilitar el manejo de estos nombres de campo en el conjunto de resultados.
- La consulta creada produce algún tipo de columna de cálculo o suma que requiere un nombre.

Por ejemplo:

```
Select cli.nombre, cli.apellido1, ped.cantidad
From CLIENTES cli, pedidos ped
Where cli.id = ped.clienteID;
```

### 3.1.12 Consultas que agrupan y resumen datos

Con frecuencia, se necesitan crear consultas para responder a cuestiones como: "¿Cuántos pedidos entraron ayer?". En este caso, no importa quién haya realizado el pedido, sólo se desea saber cuantos pedidos entraron. Para ello, se utilizan las consultas de grupo y funciones de agregación.

Por ejemplo:

```
SELECT F_PEDIDO, COUNT (CLIENTESID) AS TOTAL
FROM PEDIDOS
GROUP BY F_PEDIDO
HAVING F_PÉDIDO = #6/30/2005#
```

La consulta HAVING es equivalente a la cláusula WHERE para consultas agrupadas.

La siguiente tabla muestra algunas de las funciones de agregación más comunes.

AVG	Media de todos los valores de la columna
COUNT	Recuento del número de registros devueltos
FIRST	Primer valor del campo
LAST	Ultimo valor del campo
MAX	Valor máximo de una columna
MIN	Valor mínimo de una columna
STDEV	Desviación estándar
SUM	Total de los valores de un campo
VAR	Varianza

### 3.1.13 Subconsultas

Una subconsulta es una consulta cuyo resultado se utiliza como criterio de otra consulta. Las subconsultas toman el lugar de las expresiones WHERE normales. Debido a que el resultado producido por la subconsulta toma el lugar de una expresión, la subconsulta sólo puede devolver un único valor (a diferencia de una consulta convencional, que devuelve múltiples valores en forma de filas y columnas).

Ejemplo:

```
SELECT *  
FROM PEDIDOS  
WHERE CANTIDAD > (SELECT AVG (CANTIDAD)  
FROM PEDIDOS)
```

## 3.2 Modificación de datos

Una consulta de acción es una consulta que tiene la capacidad de modificar los registros. Las consultas de acción no devuelven conjuntos de resultados; en su lugar, realizan cambios permanentes en la base de datos.

### 3.2.1 Actualización

Una consulta de actualización tiene la capacidad de modificar un grupo de registros al mismo tiempo. Una consulta de actualización está formada por tres partes:

- La cláusula UPDATE, que especifica la tabla a actualizar.
- La cláusula SET, que especifica qué datos se cambiarán.
- El criterio WHERE, opcional, que limita el número de registros afectados por la consulta de actualización.

Ejemplo:

```
UPDATE CLIENTE
SET NOMBRE = 'JOSE'
WHERE NOMBRE = 'PEPE'
```

### 3.2.2 Eliminación

Una consulta de eliminación tiene la capacidad de eliminar un grupo de registro al mismo tiempo.

Ejemplo:

```
DELETE
FROM CLIENTES
WHERE NOMBRE = 'PEPE';
```

### 3.2.3 Inserción

La consulta de inserción se utiliza con dos fines:

- Agregar un único registro a una tabla.
- Copiar uno o más registros de una tabla a otra

Ejemplo:

```
INSERT INTO TLCLIENTES (ID, NOMBRE, PELLIDO1)
VALUES (121,'LUIS','MARQUEZ');

INSERT INTO TLCLIENTESHISTORICO
SELECT * FROM TLCLIENTES
```

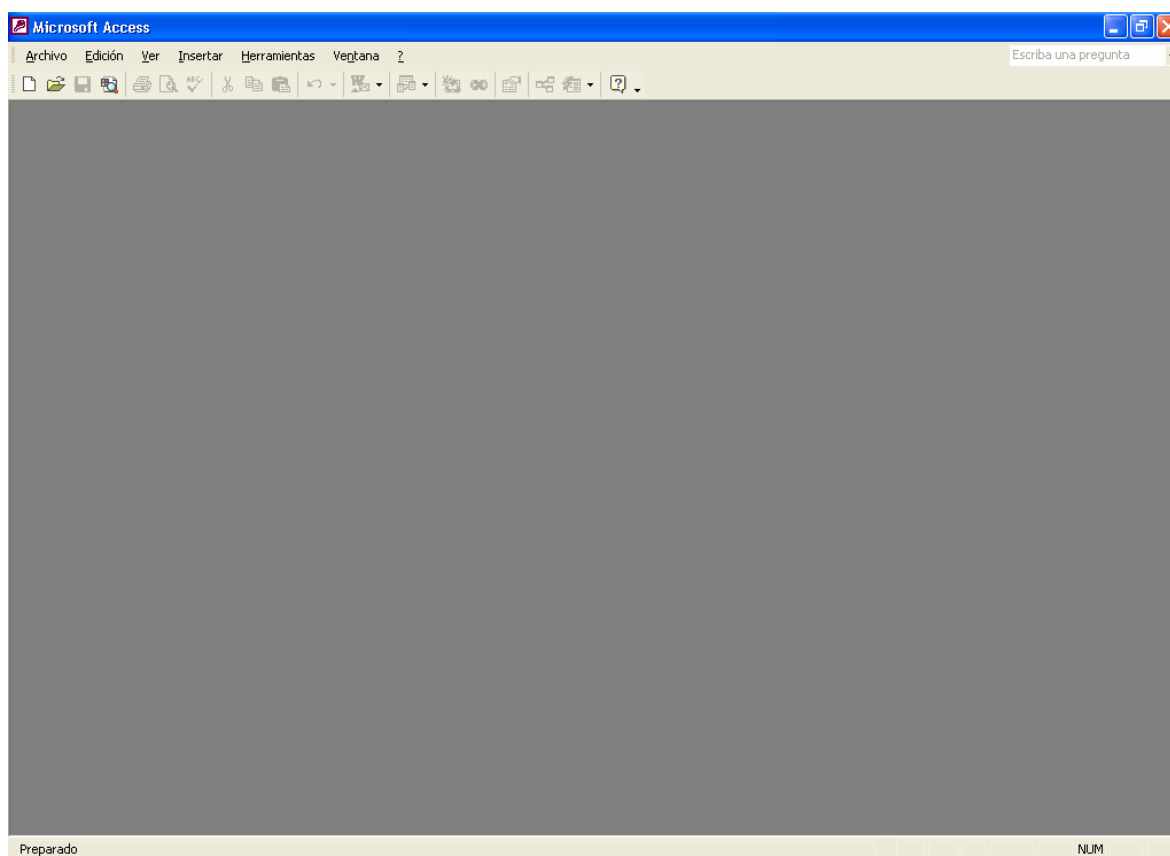
## 4. Inicio Access

A continuación vamos a dar las nociones básicas de Access.

Para acceder directamente al programa Microsoft Access se puede utilizar la opción Microsoft Access que aparece en el menú Programas correspondientes al menú Inicio, para lo cual basta con pulsar sobre dicha opción.

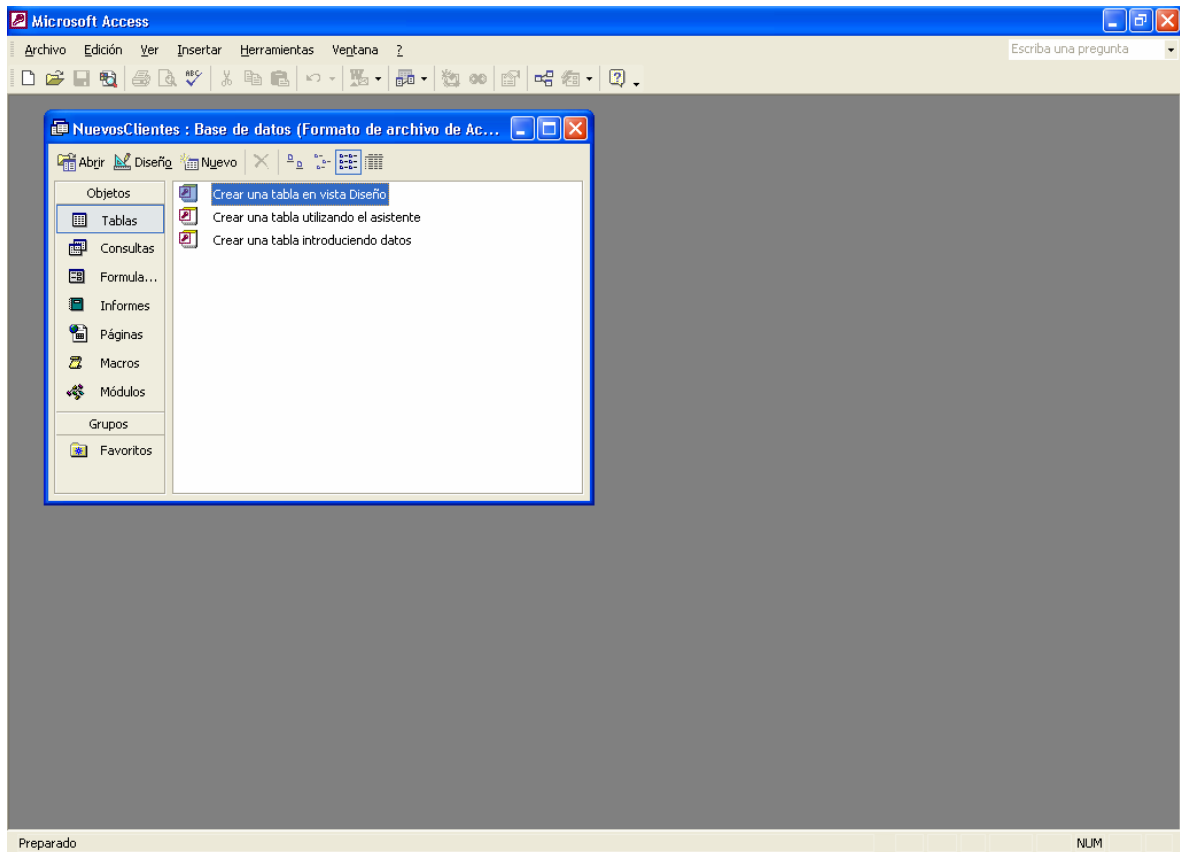
También es posible acceder mediante la barra de acceso directo de Office.

La pantalla de inicio es la siguiente:



## 5. Crear un base de datos

En la ventana de Microsoft Access seleccione la opción Base de datos en blanco, le asignamos el nombre NuevosClientes.

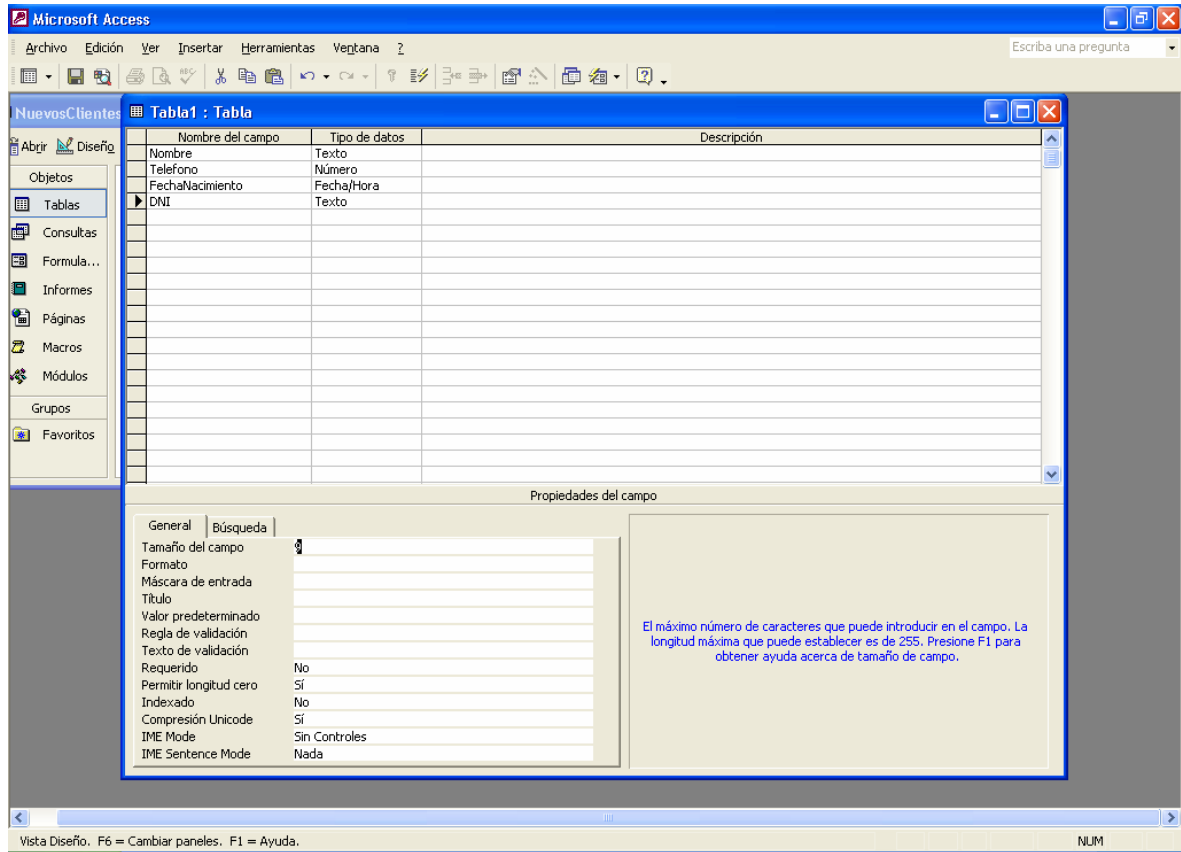


Hay otras formas de crear una nueva base de datos, por ejemplo podemos utilizar la barra de herramientas para crear una nueva base de datos.



- Objeto OLE: para datos OLE como video clips, sonidos, etc.

Continuando con el ejercicio, vamos a incorporar los campos Nombre tipo Texto, Telefono tipo Numero, FechaNacimiento tipo Fecha/Hora y DNI tipo Texto.



Para guardar esta tabla, pulse en la barra de herramientas sobre el icono que representa un disquete. En el cuadro de diálogo Guardar como, teclee Clientes. A continuación aparecerá el mensaje No hay clave principal definida; pulse el botón No.

Cierre esta ventana y observe que la base de datos NuevosClientes ya tiene una tabla denominada Clientes.

Las propiedades de los campos más importantes son:

- Tamaño del campo: el número máximo de caracteres que puede introducir en el campo.
- Formato: diseño de la presentación del campo. Se puede seleccionar un formato predeterminado o bien introducir un formato personalizado.
- Lugares decimales: el número de dígitos que aparecen a la derecha del separador decimal.
- Máscara de entrada: modelo para todos los datos introducidos en este campo.
- Título: permite definir una etiqueta de campo predeterminada para un formulario o informe.
- Valor predeterminado: introduce en el campo un valor cuando se agregan nuevos registros.
- Regla de validación: permite escribir la condición que deben satisfacer los datos introducidos para ser aceptados.

- **Requerido:** permite especificar si es obligatorio que exista un valor en el campo.
- **Permitir longitud cero:** permite especificar si una cadena de longitud cero (“ ”) es una entrada válida para el campo.
- **Indexado:** define un campo como índice o campo clave.

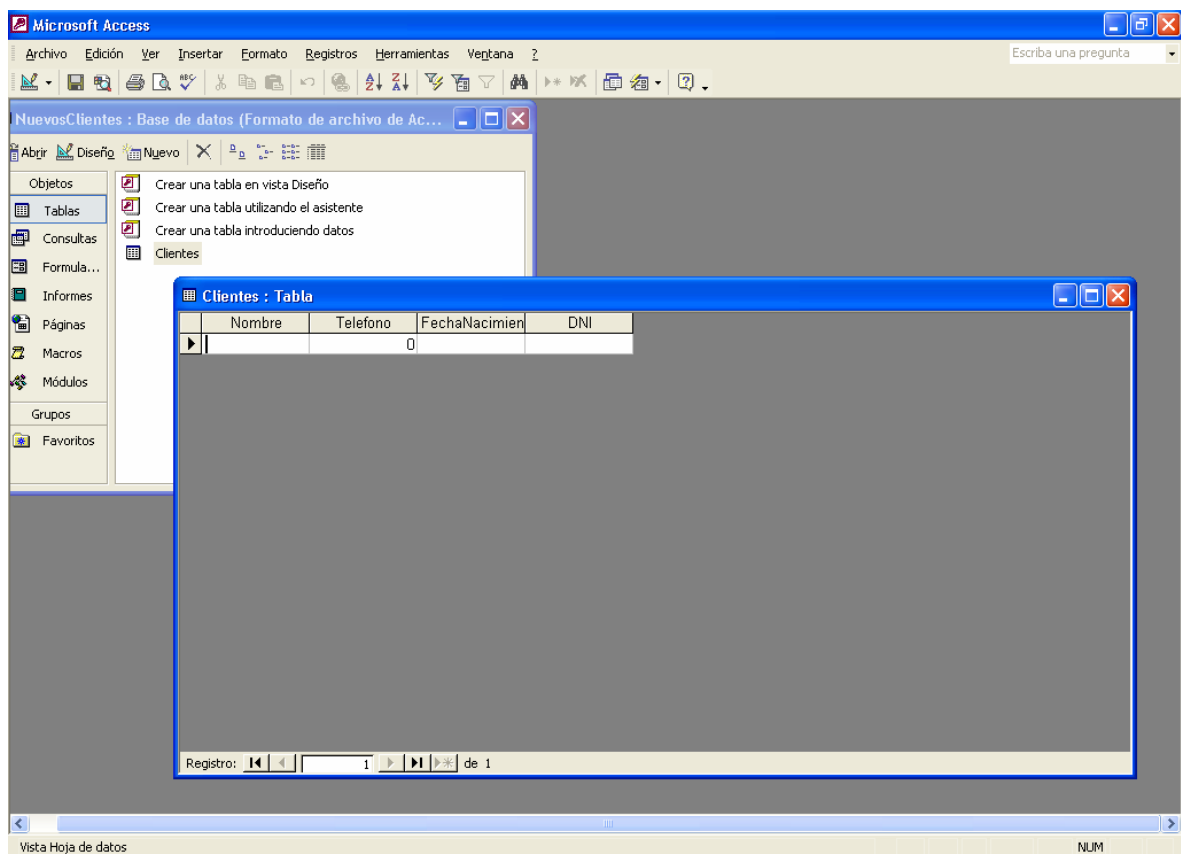
General	Búsqueda
Tamaño del campo	50
Formato	
Máscara de entrada	
Título	
Valor predeterminado	
Regla de validación	
Texto de validación	
Requerido	No
Permitir longitud cero	Sí
Indexado	No
Compresión Unicode	Sí
IME Mode	Sin Controles
IME Sentence Mode	Nada

## 7. Introducir datos en la tabla

Existen diversos modos de introducir los datos en las tablas, los dos principales son la hoja de datos de la tabla y los formularios. Vamos a utilizar el primero de ellos para meter datos en la tabla Clientes que hemos creado.

Seleccionamos la tabla Clientes y pulsamos sobre Abrir que se encuentra en la parte superior de la ventana NuevosClientes: Base de datos.

La nueva pantalla que se nos presenta tiene el siguiente aspecto:

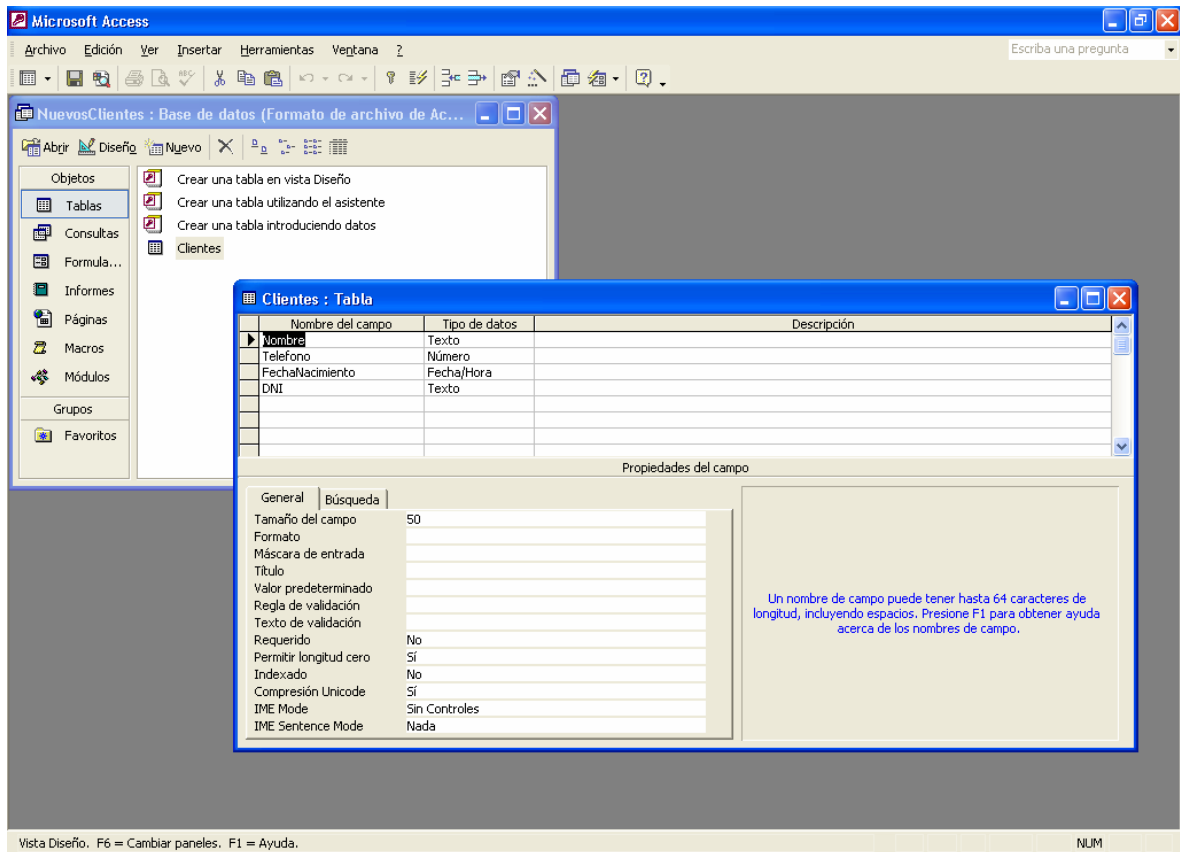


A continuación vamos rellenando los datos de las celdas que se nos presentan. Para pasar de una celda a otra podremos usar el ratón o bien pulsar la tecla Tab o Return.

Tras introducir los datos que deseemos pulse en la barra de herramientas sobre el icono que representa un disquete y cierre la ventana actual Clientes: Tabla.

## 8. Modificar la estructura de una tabla

Vamos a modificar la tabla Clientes, para ello seleccione dicha tabla de nuestra base de datos NuevosClientes y pulse sobre el botón Diseño que se encuentra en la parte superior.



En primer lugar introduzca una descripción en los 4 campos ya existentes en la tabla.

Después desplace el cursor hasta el primer campo que se encuentre vacío. Introduzca un nuevo campo Provincia tipo Texto. A dicho campo le vamos a dar los siguientes valores en algunas de sus propiedades:

- Tamaño del campo : 25
- Valor predeterminado : Sevilla

Por último modifiquemos en el campo Nombre la propiedad Requerido dándole el valor Si.

Guardemos los cambios pulsando sobre el icono disquete.

Por último salgamos de Access.

## 9. Crear un formulario y utilizarlo para modificar los datos existentes en la base de datos

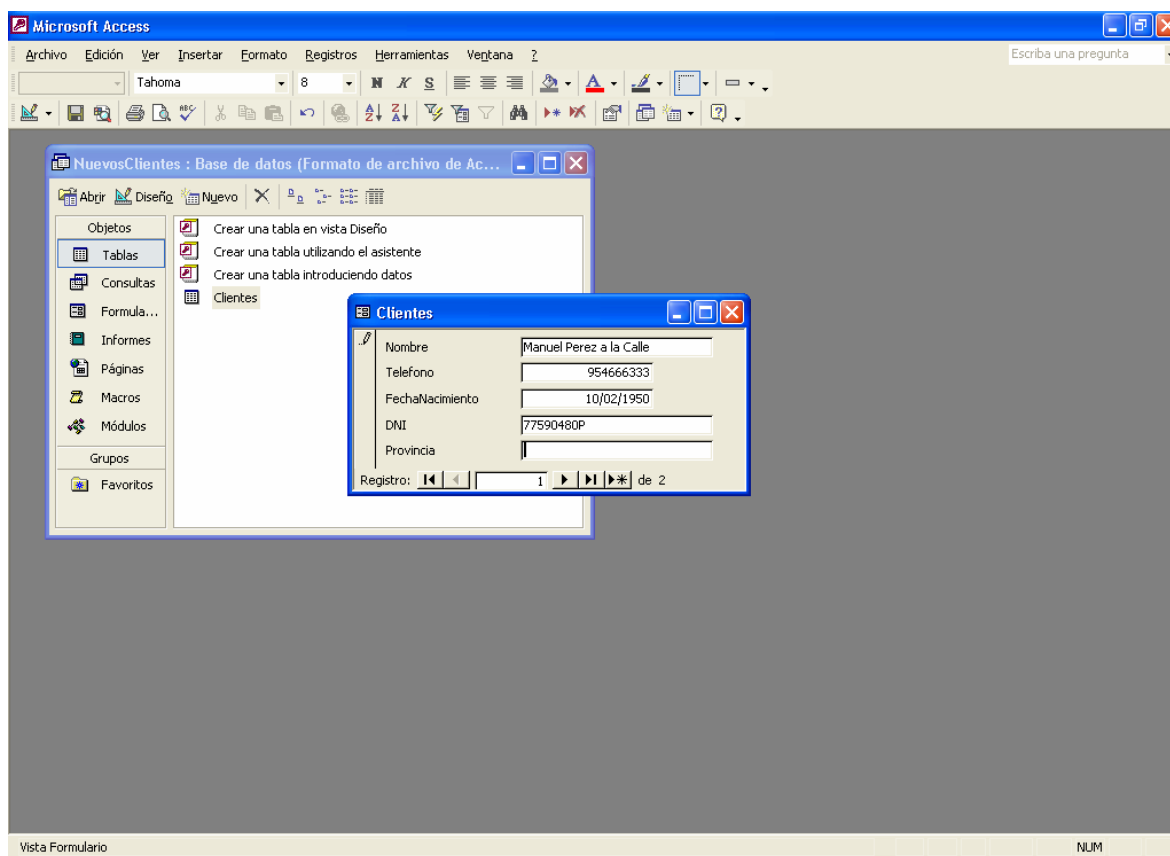
### 9.1 Crear un formulario

Vamos a crear un formulario para la tabla Clientes existente en la base de datos NuevosClientes.

Abra la base de datos NuevosClientes, para ello pulse sobre el icono con forma de carpeta abierta existente en la barra de herramientas, busque la base de datos, selecciónela y pulse Abrir.

Seleccione la tabla Clientes de la que vamos a crear el formulario.

En principio el penúltimo botón de la barra de herramientas, denominado Nuevo Objeto, que tiene una flecha en su parte derecha. Pulsemos sobre dicha flecha y seleccionemos la opción Autoformulario. La pantalla que se nos ofrece es:



Para guardar dicho formulario utilice el botón Guardar que se encuentra en la barra de herramientas, teclee el nombre Datos de Clientes y pulse sobre el botón Aceptar.

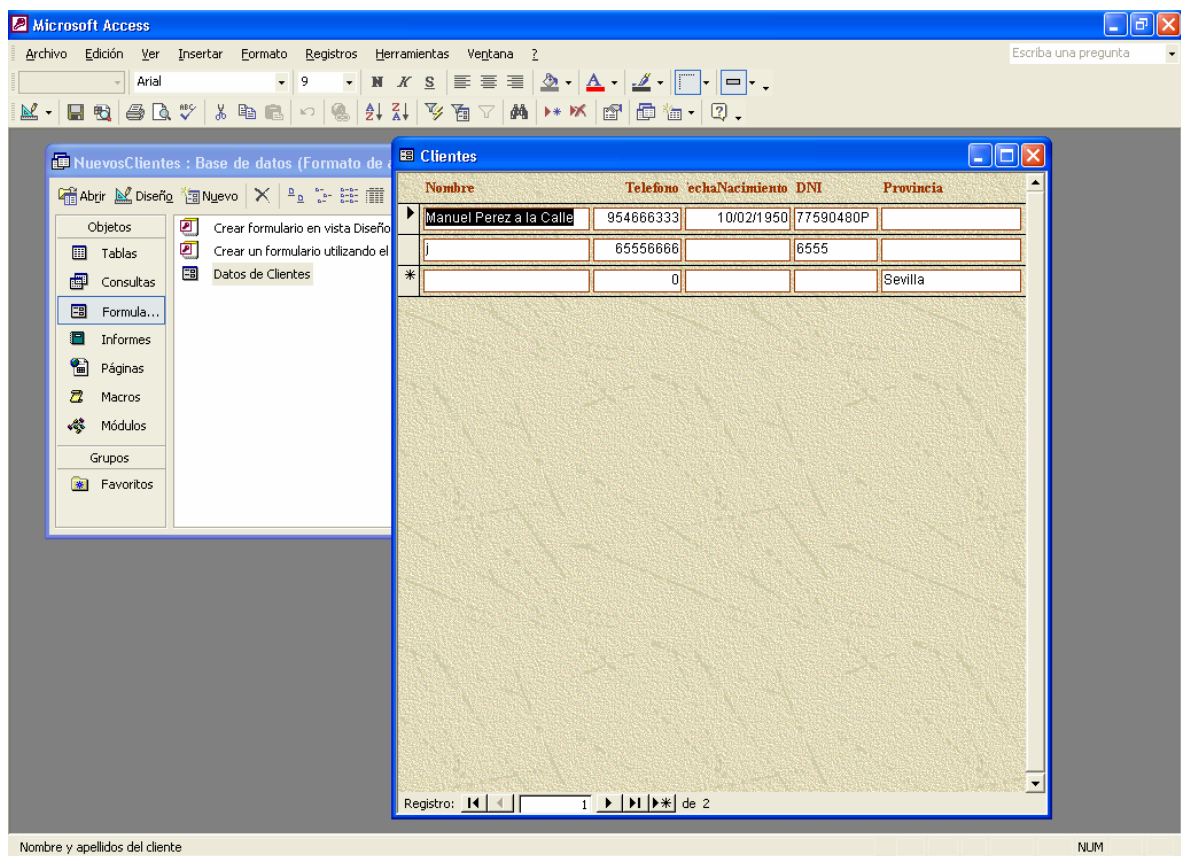
En la ventana correspondiente al formulario que hemos creado, aparece en su parte inferior un conjunto de botones que permiten pasar de un registro a otro. El cuadro de texto situado en el medio de estos botones muestra el registro que se está observando y permite, introduciendo el número de registro de interés, mostrarlo. Esto es si introducimos el número 4 y pulsamos Enter automáticamente se nos mostrará el registro situado en la cuarta posición en nuestra tabla Clientes.

Cerramos el formulario Clientes y compruebe que en la ficha Formularios de la base de datos NuevosClientes aparece el formulario que se acaba de crear.

Este formulario presenta los datos de cada registro en una columna.

Vamos a crear un formulario que nos los presente en filas.

Para ello nos situamos sobre la ficha Formulario. Pulse sobre el botón Nuevo. En el cuadro de diálogo que se nos presenta haremos varias cosas por un lado seleccione la tabla o consulta de donde provienen los datos, en este caso la tabla Clientes, luego pulsemos dos veces sobre la opción Autoformulario : en tabla. El formulario que hemos creado presenta el siguiente aspecto:



Cierre la ventana correspondiente al nuevo formulario y no lo guarde. Cierre todas las ventanas que tenga abiertas.

## 9.2 Modificar e introducir datos de una tabla

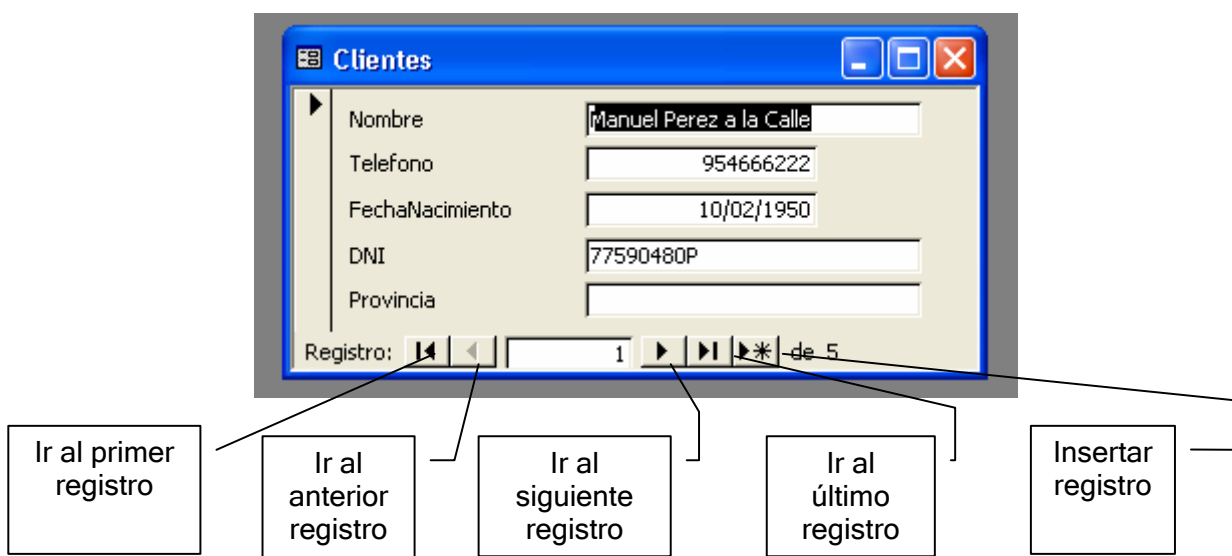
### 9.2.1 Modificar los datos de una tabla utilizando un formulario

Para modificar los datos de la tabla Clientes puede utilizar el formulario Datos de Clientes. Para ello abra la base de datos NuevosClientes. Seleccione el formulario Datos de Clientes y pulse el botón Abrir. Sitúese en el registro que quiere modificar y finalmente sustituya el valor del campo en que se encuentre por el que quiere ponerle.

Puede desplazarse por los siguientes registros de la tabla Clientes utilizando los botones para ello disponibles volviendo finalmente al registro que hemos modificado para comprobar que los cambios que en el habíamos realizado se han guardado. Cierre el formulario Datos de Clientes y compruebe en la tabla Clientes que el dato también ha sido modificado.

## 9.2.2 Introducir datos en una tabla usando un formulario

El formulario Datos de Clientes puede utilizarse para introducir nuevos datos en la tabla Clientes. Para probarlo abra el formulario y pulse sobre el botón existente que tiene un asterisco (el último por la derecha), en este momento se puede introducir el nuevo registro.

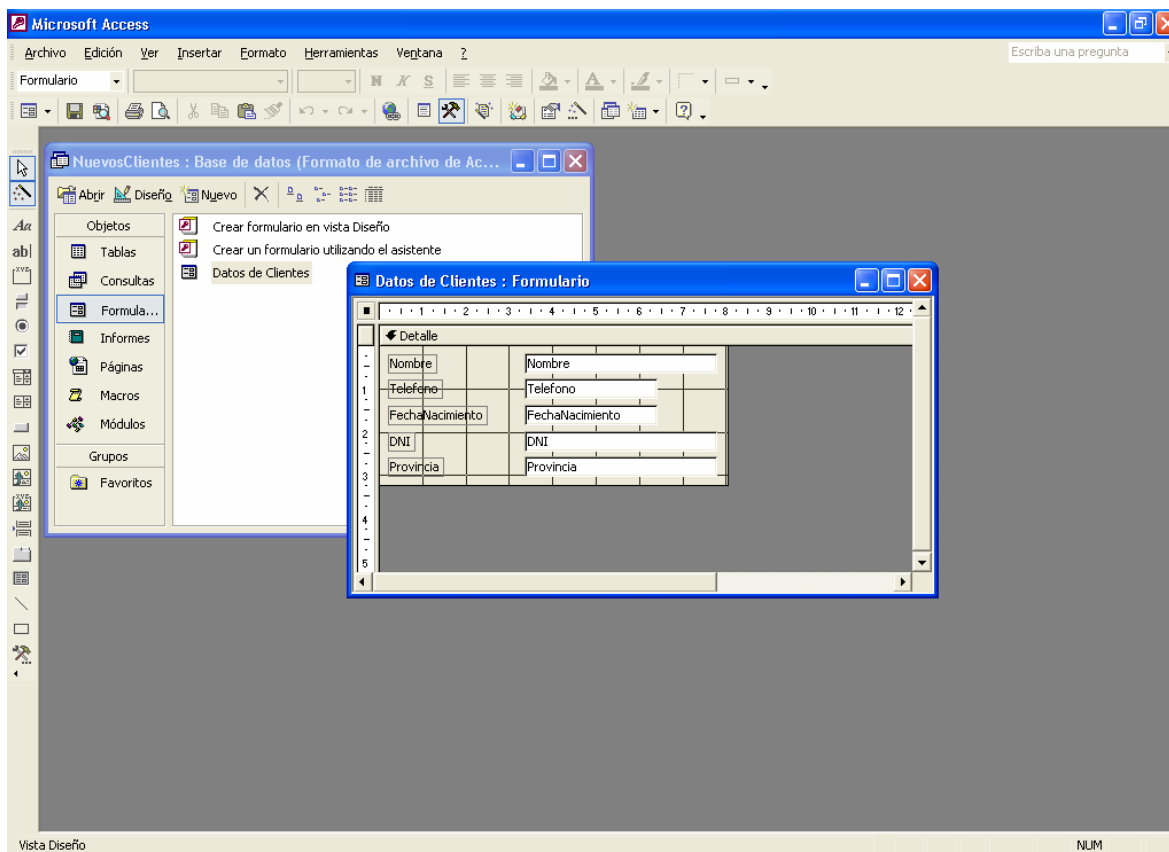


Puede desplazarse por los siguientes registros de la tabla Clientes utilizando los botones para ello disponibles volviendo finalmente al registro que hemos creado que será el último de los ahora existentes.

Cierre el formulario Datos de Clientes y compruebe en la tabla Clientes que el dato también ha sido creado. Cierre todas las ventanas.

## 10. Modificar un formulario

Es posible modificar el diseño de un formulario anteriormente creado. Abra la base de datos NuevosClientes si la tiene cerrada. Pulse sobre el botón Formularios. Seleccione el formulario Datos de Clientes. Pulse sobre el botón diseño.



Para modificar el diseño de un formulario, se han de modificar sus controles, que no son más que los cuadros de texto que representan a cada uno de los campos, así como las etiquetas de los mismos.

Para seleccionar un control debe pulsar en el mismo y aparecerán a su alrededor unos recuadros negros, denominados selectores. Al situar el puntero en el extremo de un control seleccionado, éste adopta la forma de una mano negra. Pulse y arrastre a la posición deseada el control. Si en lugar de la mano, aparece un dedo, únicamente trasladará bien el campo, bien la etiqueta del mismo, según este situado en uno o en otro.

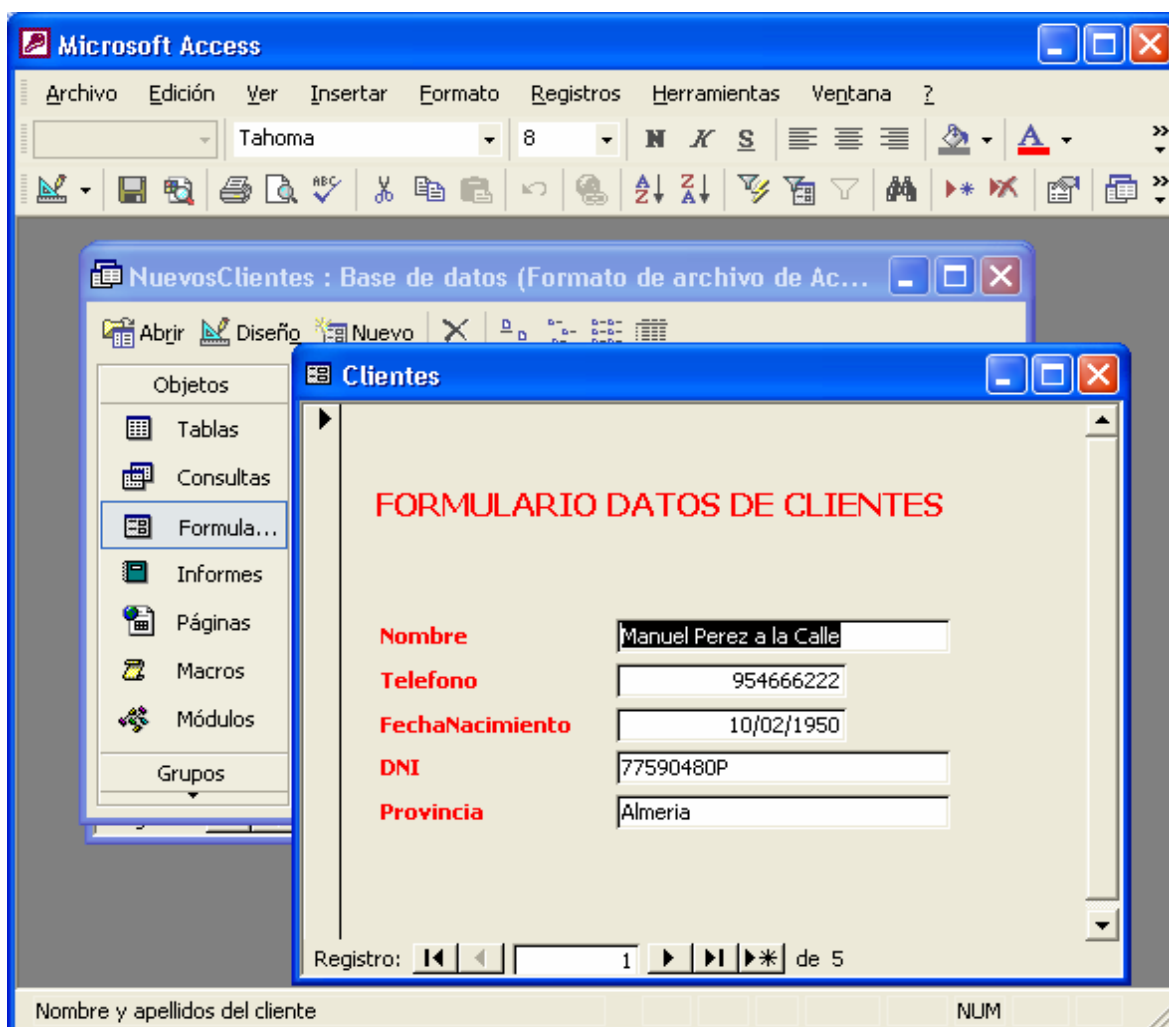
También puede alterar el tamaño del mismo utilizando los selectores que se encuentran a los lados, en la parte central y en las esquinas. Si se quiere realizar la operación sobre varios controles, puede seleccionarlos, pulsando sobre uno de ellos, y a continuación pulsando la tecla Shift y sin soltarla pulsar sobre cada uno de los controles que desea seleccionar. También puede utilizar las órdenes Copiar, Cortar y Pegar del menú Edición con los controles seleccionados. Para eliminar un control, selecciónelo y a continuación pulse la tecla Supr.

Otra sencilla posibilidad que ofrece el diseño de formularios en Access es incorporar una etiqueta, modote título explicativo del formulario. Pulse sobre el botón *Aa* (etiqueta) de la caja de herramientas que aparece en la parte izquierda de la pantalla, y

pulse en la posición del formulario en la que desee dicha etiqueta. A continuación teclee FORMULARIO DATOS DE CLIENTES.

De entre los controles existentes en nuestro formulario basta seleccionar el control que desea modificar y posteriormente elegir en el cuadro de herramientas principal de Access la propiedad que queremos modificar: fuente, negrita, alinear, color de texto,... etc.

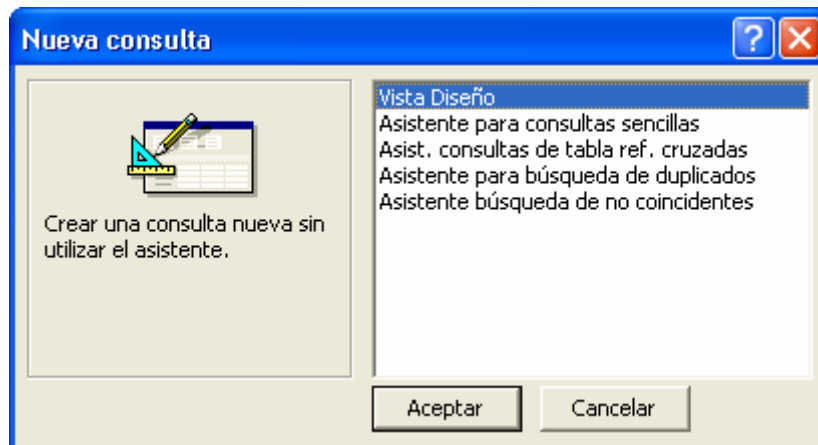
Por último también es posible agregar un nuevo control (campo) al formulario, usando el botón Lista de Campos, aunque en nuestro caso ya tenemos todos los campos existentes en la tabla Clientes en el formulario Datos de Clientes. Obtenga un diseño del formulario similar a:



Por último guardemos el nuevo diseño del formulario con la opción del botón Guardar.

## 11. Crear un consulta

Vamos a crear una consulta en Access que nos devuelva los clientes de la tabla Clientes cuya Provincia sea Sevilla. Para ello abra la base de datos NuevosAlumnos caso de tenerla cerrada, pulse sobre la ficha Consultas de la base de datos. Pulse sobre el botón Nuevo. Nos saldrá el siguiente cuadro de diálogo:



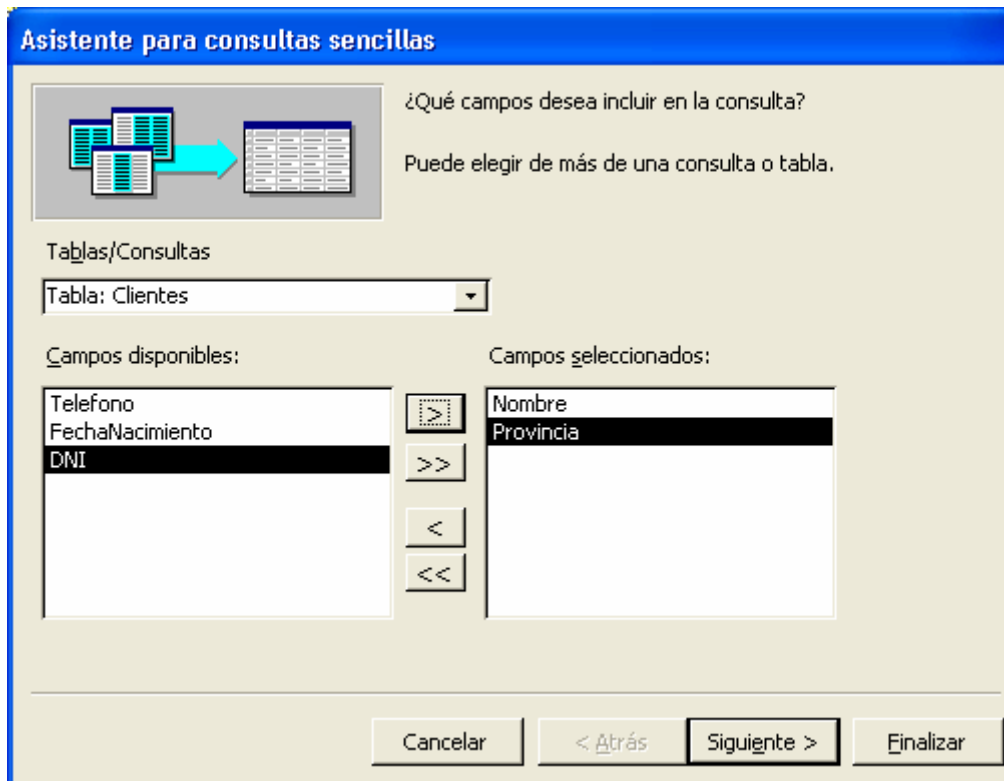
Seleccione la opción Asistente para consultas sencillas.

Sirve para pasar a Campos seleccionados el campo que se encuentre con fondo negro en Campos disponibles.

Sirve para incorporar todos los campos.

Se utilizan respectivamente para deseleccionar un campo o la totalidad de ellos de la lista de seleccionados.

En la lista de campos disponibles que se nos presenta seleccionemos aquellos que queremos mostrar en nuestra consulta. En nuestro caso serán: Nombre y Provincia.



Pulse sobre el botón siguiente y asígnele el nombre “Consulta Clientes de Sevilla” a la consulta que acaba de crear. Para acabar pulse el botón Finalizar.

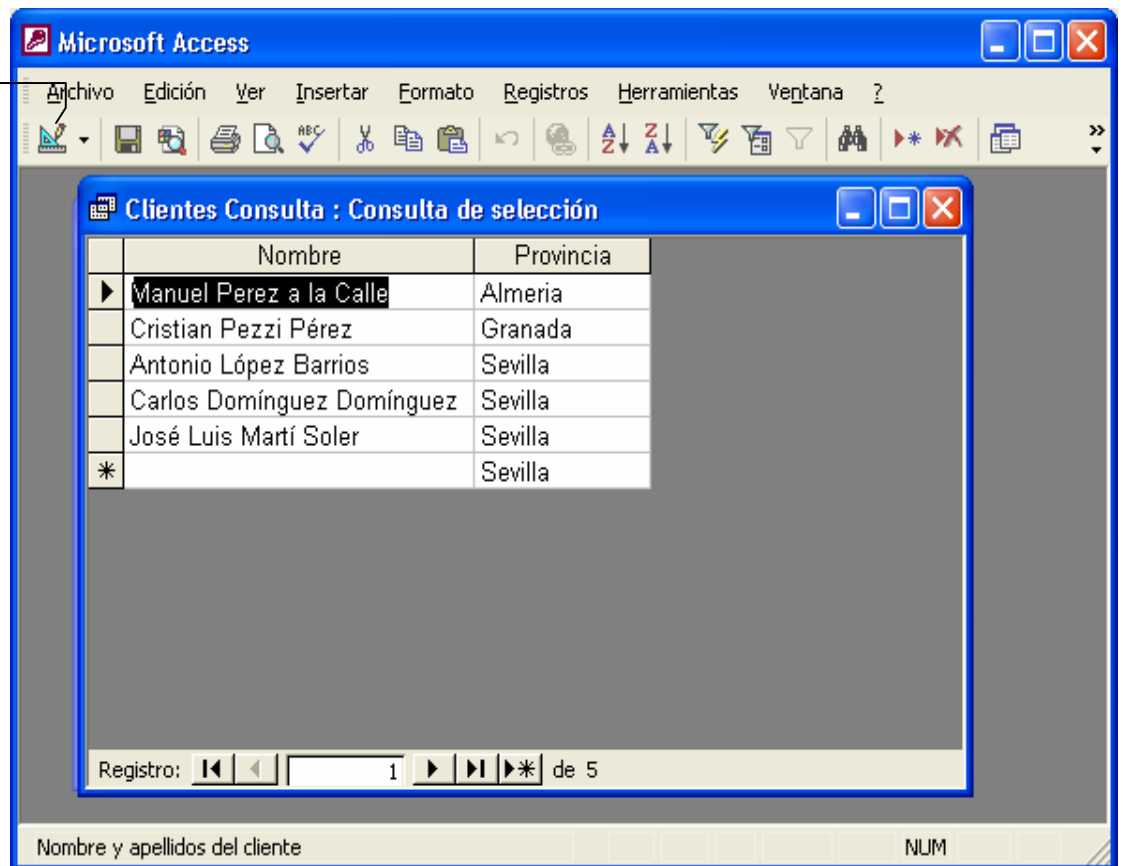
La consulta se ejecuta y nos devuelve los campos Nombre y Provincia de todos los registros existentes en la tabla Clientes.

Pulsando en este botón podemos elegir que la ventana de la consulta sea:

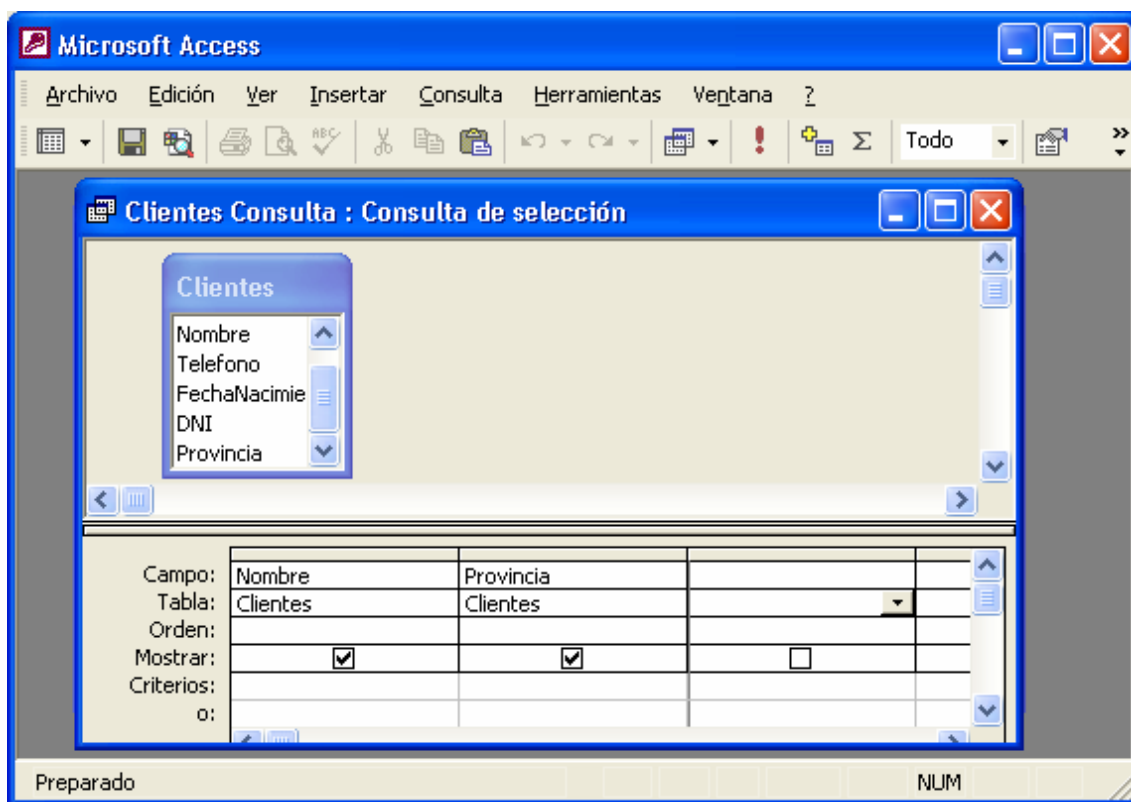
- Vista Hoja de Datos, como la que se acaba de obtener.

- Vista Diseño.

- Vista SQL (lenguaje de consulta estructurado).



Pulse sobre la opción Vista Diseño. La ventana de diseño de la consulta tiene dos elementos principales: Lista de campos, en la que se presentan las tablas que se han añadido a la consulta (en nuestro caso sólo Clientes) y la cuadrícula QBE, donde figuran los campos seleccionados para la consulta:



En el caso de querer añadir más campos a la consulta basta con pulsar dos veces sobre el nombre del campo correspondiente en la lista de campos. Si quiere borrar un campo, sitúe el puntero sobre dicho campo en la cuadrícula, y pulse en la opción Eliminar columna del menú Edición.

El orden en que aparecen los campo en la cuadrícula QBE será el mismo que en el que se nos presente la consulta al ejecutarla. Para modificar la posición de los campos sitúe el puntero justo encima del nombre del campo y pulse, se seleccionará toda la columna. Ahora puede arrastrar dicha columna al lugar deseado.

En el caso de que queramos incluir una condición sobre un campo pero no deseemos que dicho campo se visualice en la consulta, podemos desactivar la casilla de verificación Mostrar de dicho campo.

En nuestro caso agreguemos el campo DNI y pongamos el campo Provincia como no visible en la consulta.

Pero recuerde que el objetivo de la consulta era obtener sólo los clientes cuya provincia fuese Sevilla. Para conseguirlo debe incluir dicha condición a la consulta. Esto se hace incluyendo la expresión deseada en la fila de criterios de la cuadrícula.

Para añadir condiciones a la consulta, se deben atender las siguientes particularidades:

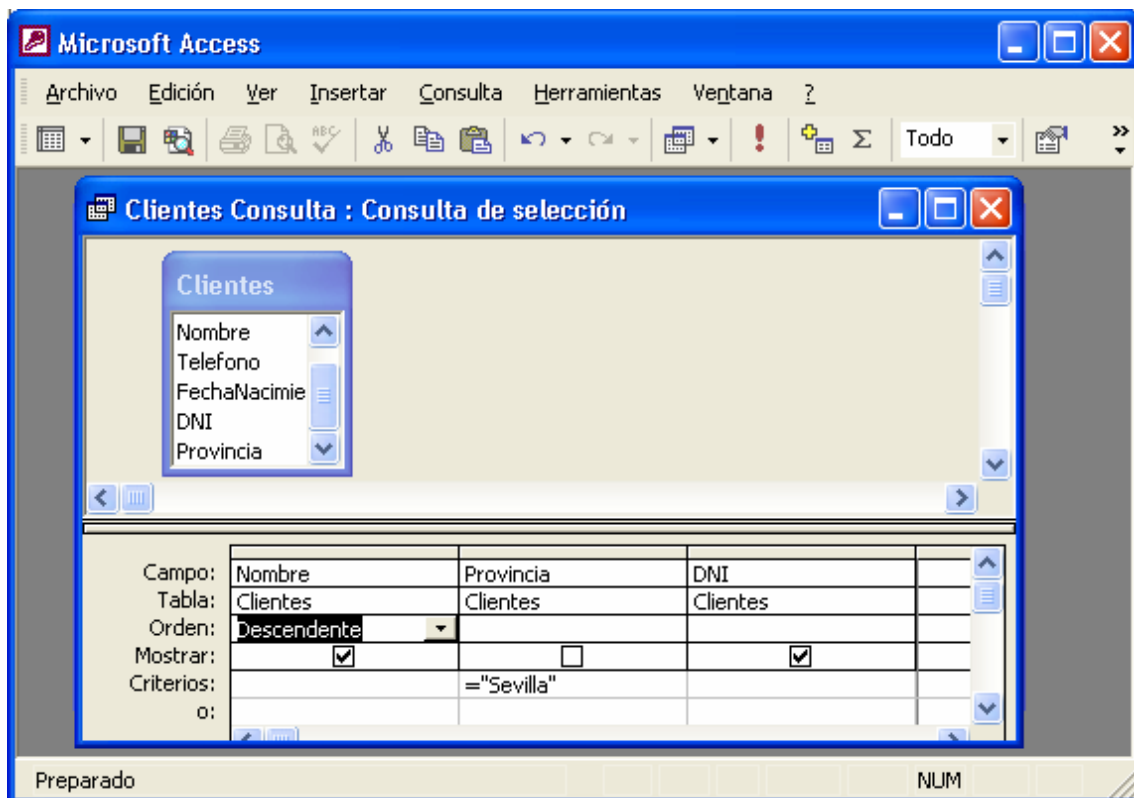
- El texto debe ir entre comillas dobles
- Las fechas deben ir entre el carácter #

- Los nombres de los objetos de Access deben ir entre corchetes.
- Los campos lógicos usan = "Si" = "No"
- Los operadores lógicos que pueden emplearse son: = , > , < , >= , <= y <> entre otros.

En nuestro caso deberíamos agregar el criterio: = "Sevilla".

También se puede modificar el orden en el que aparecen los resultados de la consulta, ya que al pulsar sobre la fila Orden de un campo, aparecen las opciones Ascendente, Descendente y sin ordenar. Ordene el campo Nombre en forma descendente.

Debe obtener el siguiente resultado:



Para que se ejecute la consulta pulse el botón Ejecutar situado casi en el centro de la barra de herramientas, el que tiene un signo de admiración en rojo.

## 12. Filtros en un formulario

Mezclando los conocimientos de formularios y consultas que tenemos podemos realizar un formulario con filtro, es decir, un formulario que nos muestre sólo los registros que verifiquen ciertas condiciones.

Vamos a modificar como práctica el formulario que habíamos creado de nombre Datos de Cliente de forma que muestre sólo los clientes cuya provincia sea Sevilla.

En primer lugar vamos a la ficha Formulario de la base de datos NuevosClientes. Abra el formulario Datos de Clientes. Al abrirlo se nos habrá aparecido una nueva opción en el menú principal, Registros, seleccione la opción Filtro, y a continuación pulse sobre la opción Filtro orden avanzado.

Botón Aplicar filtro

The screenshot shows the Microsoft Access interface. The main window is titled 'NuevosClientes : Base de datos (Formato de archivo de Ac...)' and contains a form titled 'FORMULARIO DATOS DE CLIENTES'. The form has the following fields:

- Nombre: Antonio López Barrios
- Telefono: 88
- FechaNacimiento: (empty)
- DNI: (empty)
- Provincia: Sevilla

At the bottom of the form, it says 'Registro: 1 de 3 (Filtrado)'. Below the form, a small window shows the filter criteria:

Campo:	Orden:	Criterios:
Provincia		= "Sevilla"

The status bar at the bottom of the window shows 'Nombre y apellidos del cliente', 'FILT', and 'NUM'.

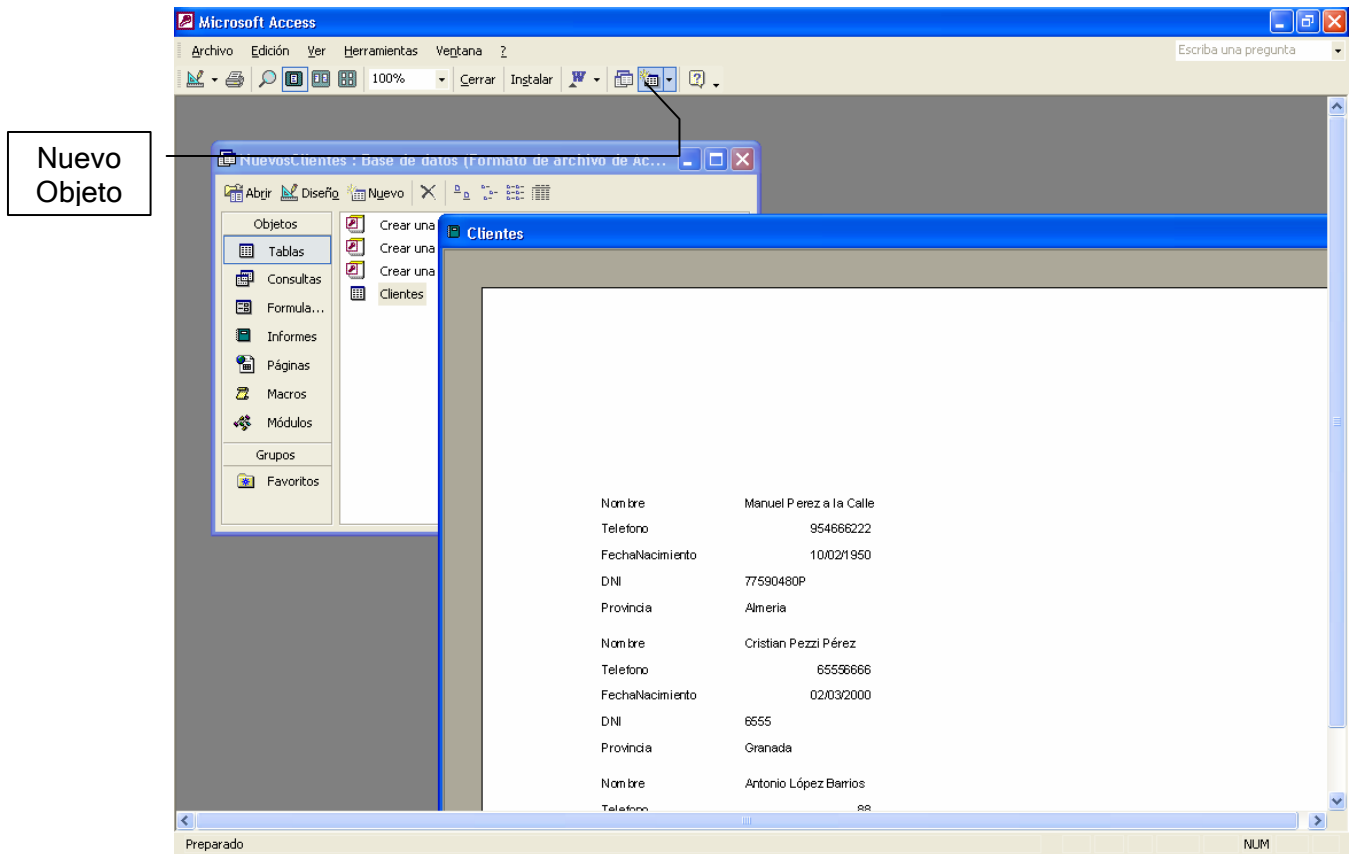
Defina en esta ventana el filtro que deseamos y pulse el botón Aplicar filtro.

Observe que los registros presentados en el formulario son sólo aquellos clientes que tienen como Provincia Sevilla.

Para desactivar el filtro, pulse de nuevo el botón Aplicar filtro (que ahora se llama Quitar filtro).

### 13. Imprimir mediante informes

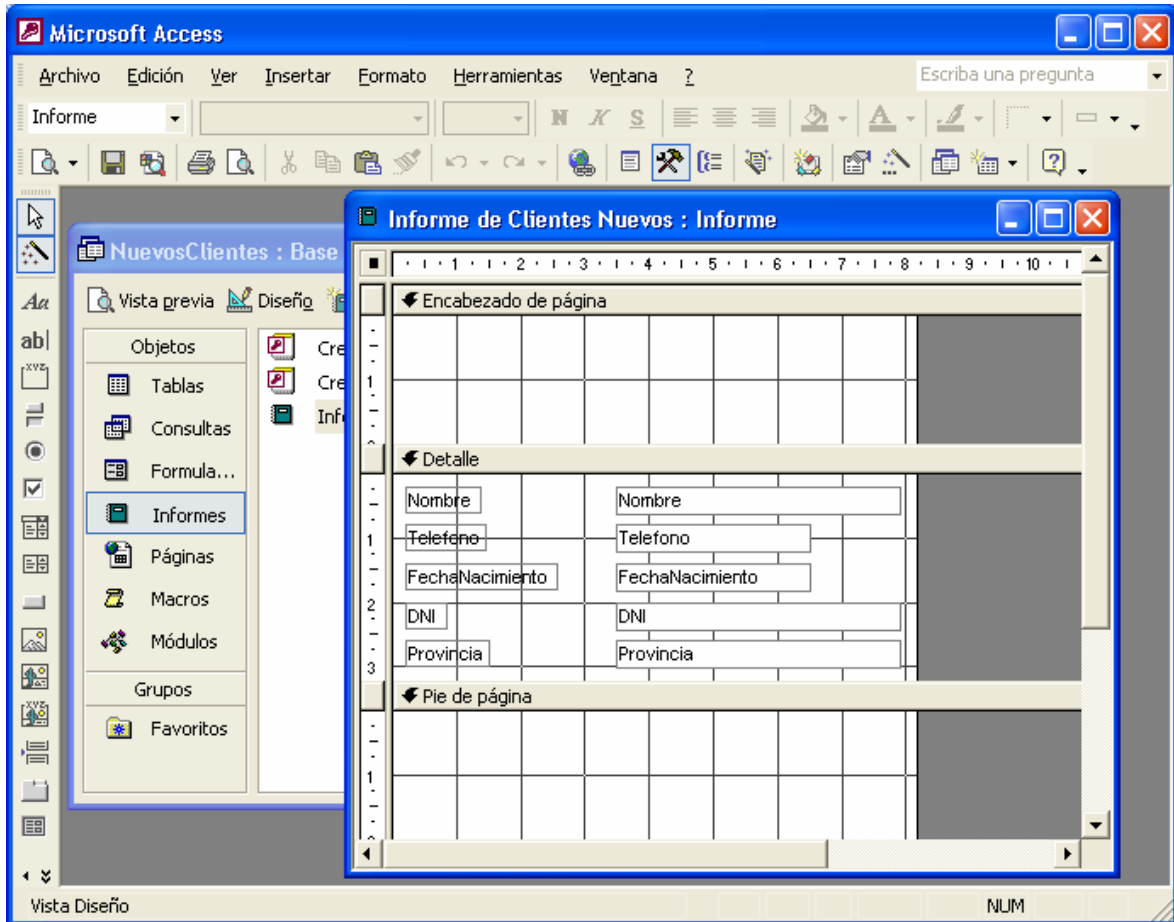
Vamos a crear un Autoinforme de la tabla Clientes. Para ello abramos la base de datos NuevosClientes. Seleccione la tabla denominada Clientes. Pulse en el botón Nuevo Objeto situado en la parte derecha de la barra de herramientas y elija la opción Autoinforme.



Guardemos este informe llamándolo “Informe de Clientes Nuevos”.

Este informe que hemos generado lo vamos a modificar tal y como hicimos con el formulario que realizamos anteriormente.

Para ello pulse en la ficha Informes de la base de datos NuevosClientes, seleccione el informe: Informe de Clientes Nuevos, y pulse sobre el botón diseño. Nos aparece la siguiente ventana:



Las modificaciones las podemos realizar igual que como hicimos en el diseño del formulario.

Observe que la ventana aparece dividida horizontalmente en distintas secciones:

- Encabezado de página: contiene la información que sólo debe aparecer al principio de cada página.
- Detalle: muestra la información que va a salir por cada registro.
- Pie de página: presenta la información que debe aparecer al final de cada página.

Mediante el menú Ver podemos mostrar dos secciones más:

- Encabezado del informe: muestra la información que debe aparecer sólo al principio del informe.
- Pie del informe: contiene la información que debe aparecer al final del informe.

## 14. Creación de la base de datos II

A partir de ahora ampliaremos la materia vista, para ello profundizaremos en los conocimientos sobre bases de datos.

Las bases de datos relacionales están formadas por una o más tablas que están relacionadas entre si y contienen la información ordenada de forma organizada.

Un concepto importante asociado a estos sistemas de gestión es el de clave de una tabla. Una clave es una referencia que se utiliza para identificar los registros de forma única. Así por ejemplo una clave puede ser el número de matrícula para un conjunto de coches o el DNI para un conjunto de personas. En definitiva la clave podrá ser cualquier propiedad que identifique de manera inequívoca un registro.

El campo que sea clave no podrá tener dos registros con el mismo valor y nunca podrá dejarse vacío.

Al crear cualquier tabla deberemos definir cual va a ser su clave.

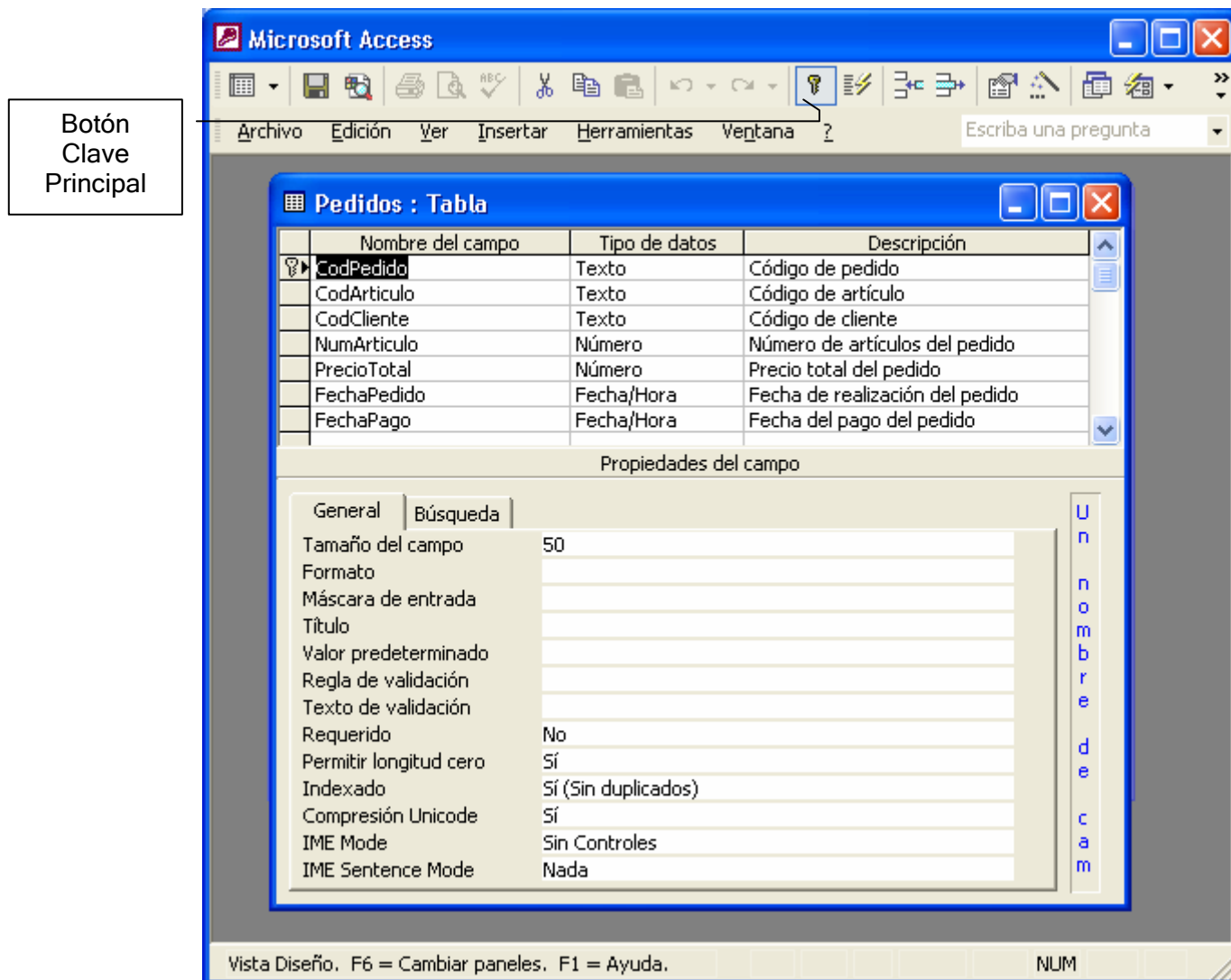
Como práctica vamos a crear una base de datos que nos sirva para gestionar los artículos, clientes y ventas de una determinada empresa.

Creemos una nueva base de datos llamada Gestión de Pedidos, en la que vamos a crear tres tablas con los siguientes nombres y campos:

Articulos		Clientes		Pedidos	
Nombre	Tipo	Nombre	Tipo	Nombre	Tipo
CodArticulo(*)	Texto	CodCliente(*)	Texto	CodPedido(*)	
Descripción	Texto	Nombre	Texto	CodArticulo	
Comentario	Memo	DNI	Texto	CodCliente	
		Telefono	Número	NumArticulos	
		Comentario	Memo	PrecioTotal	
				FechaPedido	
				FechaPago	

Los campos primeros de cada tabla (tienen un asterisco a su derecha) serán los campos clave de cada tabla.

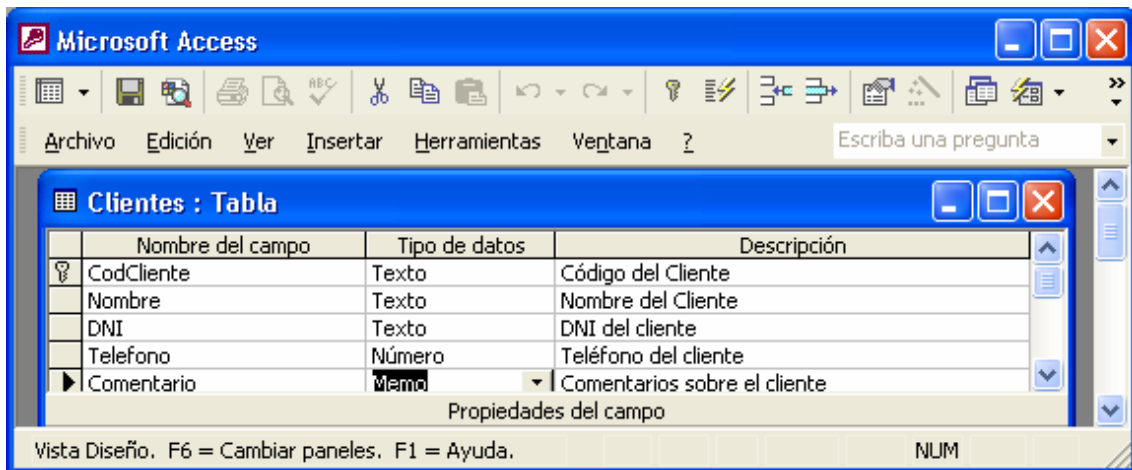
Para asignar un campo como clave deberemos seleccionarlo tras definir la tabla y pulsar sobre el botón clave principal (llave amarilla) situado en el centro de la barra de herramientas.



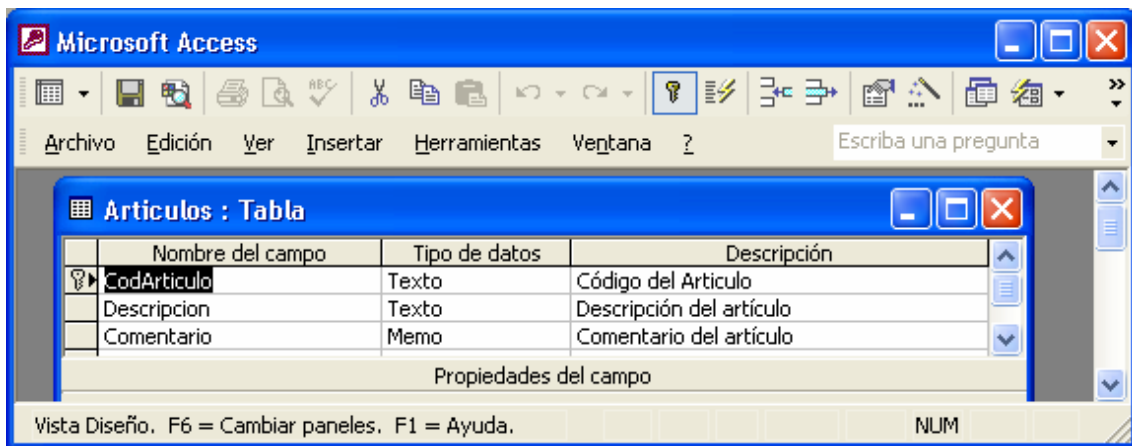
Para que dicho campo no tenga la característica de campo clave, basta con pulsar nuevamente sobre dicho botón clave principal de la barra de herramientas.

Puede comprobar que Access no le permite introducir dos registros en la tabla Pedidos con el mismo CodPedido ni tampoco introducir un nuevo registro en la misma tabla sin tener relleno el campo CodPedido. Esto es debido a que el campo CodPedido ha sido definido como campo clave de la tabla.

El diseño de la tabla Clientes sería:



Y el de la tabla Artículos:



Introduzca datos coherentes en las tablas que acaba de crear.

## 15. Relaciones

Una relación entre dos tablas lo podríamos explicar como si fuese la definición de la unión entre ambas de forma que se pueda acceder a los datos de dichas tablas de manera eficiente.

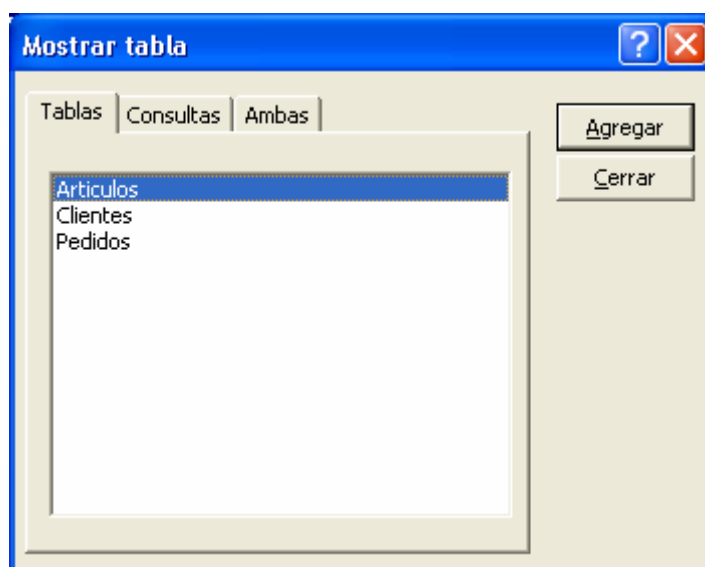
Los diferentes tipos de relación son:

- Relación uno a varios: En este tipo de relación un registro de una tabla puede tener más de un registro que coincida con la otra tabla pero la otra tabla sólo puede tener un registro coincidente con la primera.
- Relación varios a varios: En este tipo de relación un registro de una tabla puede tener más de un registro coincidente en la otra tabla. Solo podemos crear este tipo de relación si tenemos una tercera tabla (llamada la tabla de unión).
- Relación uno a uno: Cada registro de una tabla solo puede tener una correspondencia en la segunda y viceversa. Este tipo de relación no es muy habitual.

Podemos definir las relaciones entre nuestras tablas desde el menú Herramientas en la opción Relaciones o desde el botón Relaciones de la barra de Herramientas, pero cuidado debemos mantener cerradas las tablas.

Al hacerlo se nos ofrece el botón Mostrar Tablas, desde el que podremos seleccionar las tablas sobre las que vamos a definir las relaciones.

Seleccione la tabla que quiere agregar y pulse Agregar. Pulsamos Cerrar cuando termine de agregar las tres tablas.

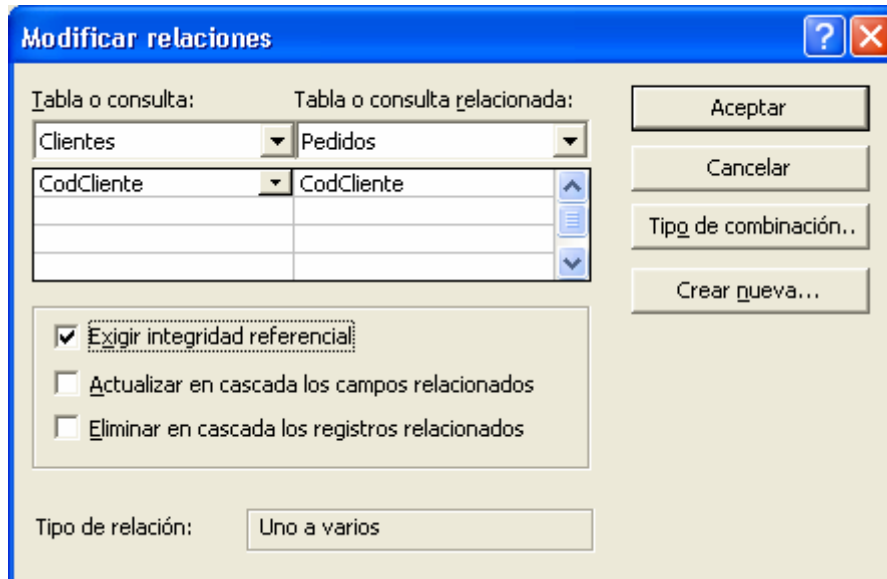


(Hemos de tener en cuenta que en Access se combinan automáticamente tablas que tengan un campo con el mismo nombre en las dos tablas aunque no exista una relación definida entre ellas)

A continuación en la ventana de relaciones seleccione el campo que quiere relacionar y lo arrastra y suelta en el campo de la otra tabla.

En nuestro caso seleccione el campo CodArticulo de la tabla Articulos y sin dejar de pulsar el botón lo arrastra y suelta sobre el campo CodArticulo de la tabla Pedidos.

Access no mostrará el siguiente diálogo:



Aquí puede seleccionar el tipo de relación (pulsando "Tipo de combinación" y si lo desea exigir la integridad referencial. Si seleccionamos esta opción cuando creamos una relación garantizamos que las relaciones entre los registros de tablas relacionadas son válidas y que no se eliminan ni modifican accidentalmente datos relacionados.

En nuestro caso marcamos la opción Exigir integridad referencial.

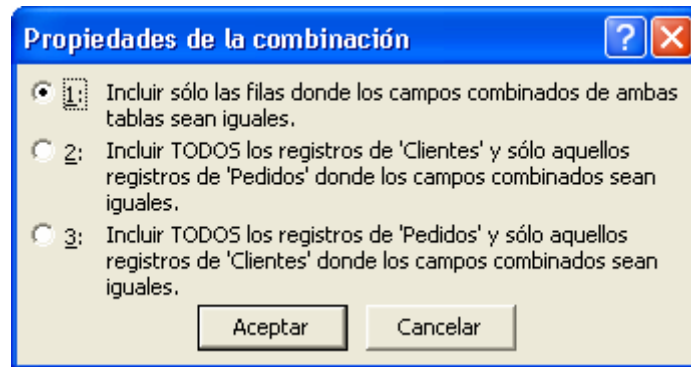
Luego seleccione el campo CodCliente de la tabla Clientes y lo arrastra sobre el campo CodCliente de la tabla Pedidos, marcando a su vez la opción Exigir integridad referencial.

Aparece una línea que refleja la relación entre ambas tablas.

Otras opciones interesantes que se nos ofrecen son:

- Actualizar en cascada los campos relacionados: hace que cuando se modifique un dato en una de las dos tablas, también sea modificado en la tabla relacionada.
- Eliminar en cascada los campos relacionados: hace que cuando se elimine un dato la tabla principal, éste también sea eliminado en la tabla relacionada.

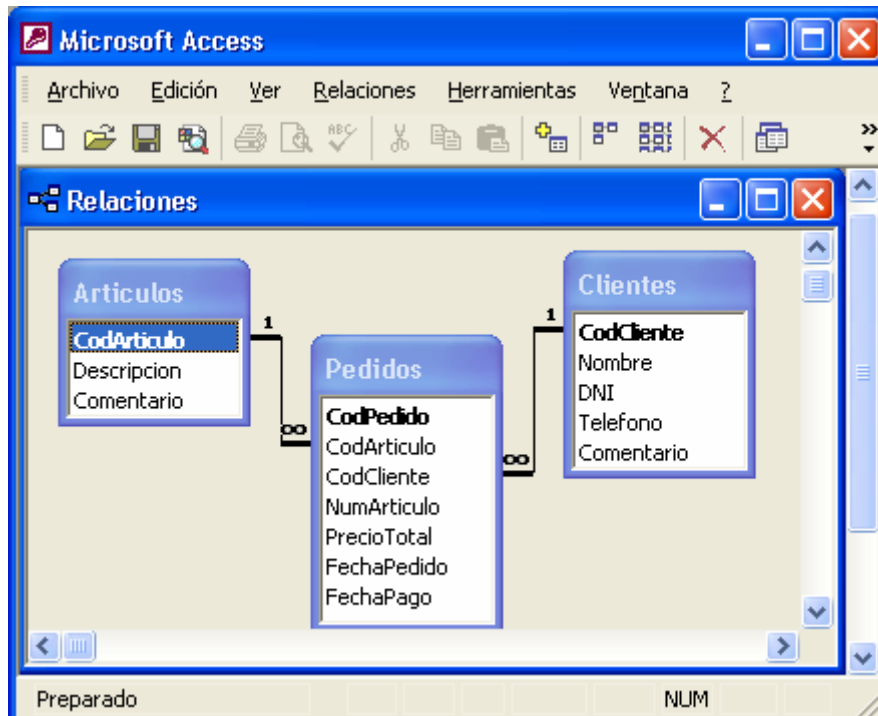
Al mismo tiempo tenemos otra opción algo más avanzada, el botón Tipo de combinación, al pulsarlo nos muestra el cuadro de diálogo:



Las tres opciones que se nos presentan significan:

- Caso 1: es la que sale por defecto e indica que en caso de realizar una consulta sólo se obtienen las filas de la tabla que tienen al menos una fila de la otra tabla que cumpla la condición.
- Caso 2: hace que en caso de realizar una consulta se obtienen todas las filas de la tabla Clientes aunque no tengan una fila de la tabla Pedidos que cumpla la condición.
- Caso 3: hace que en caso de realizar una consulta se obtienen todas las filas de la tabla Pedidos aunque no tengan una fila de la tabla Clientes que cumpla la condición.

En el ejercicio deje las dos relaciones con la opción 1.



Caso de querer modificar las relaciones deberá pulsar con el botón derecho sobre ella y seleccionar la opción Modificar relación. Para eliminar una relación: seleccionamos la línea que representa esta relación en la ventana Relaciones y pulsamos Supr.

## 16. Crear una consulta

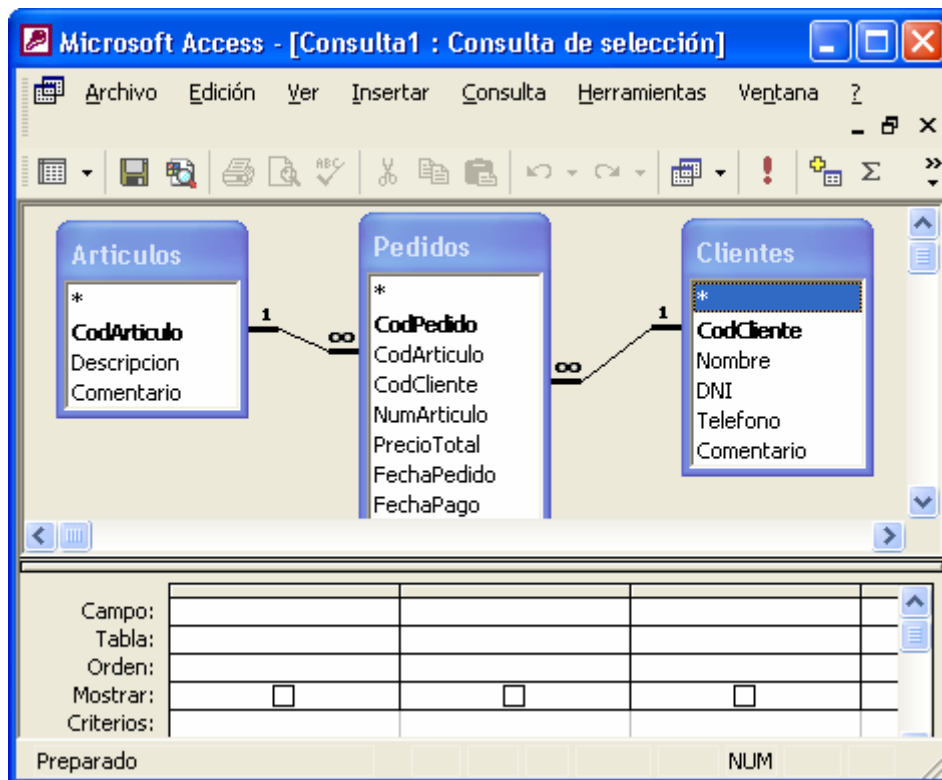
Vamos a crear varias consultas a partir de los datos introducidos en las tablas Clientes, Artículos y Pedidos.

### 16.1 Consulta 1

Mostrar los siguientes datos de los pedidos existentes:

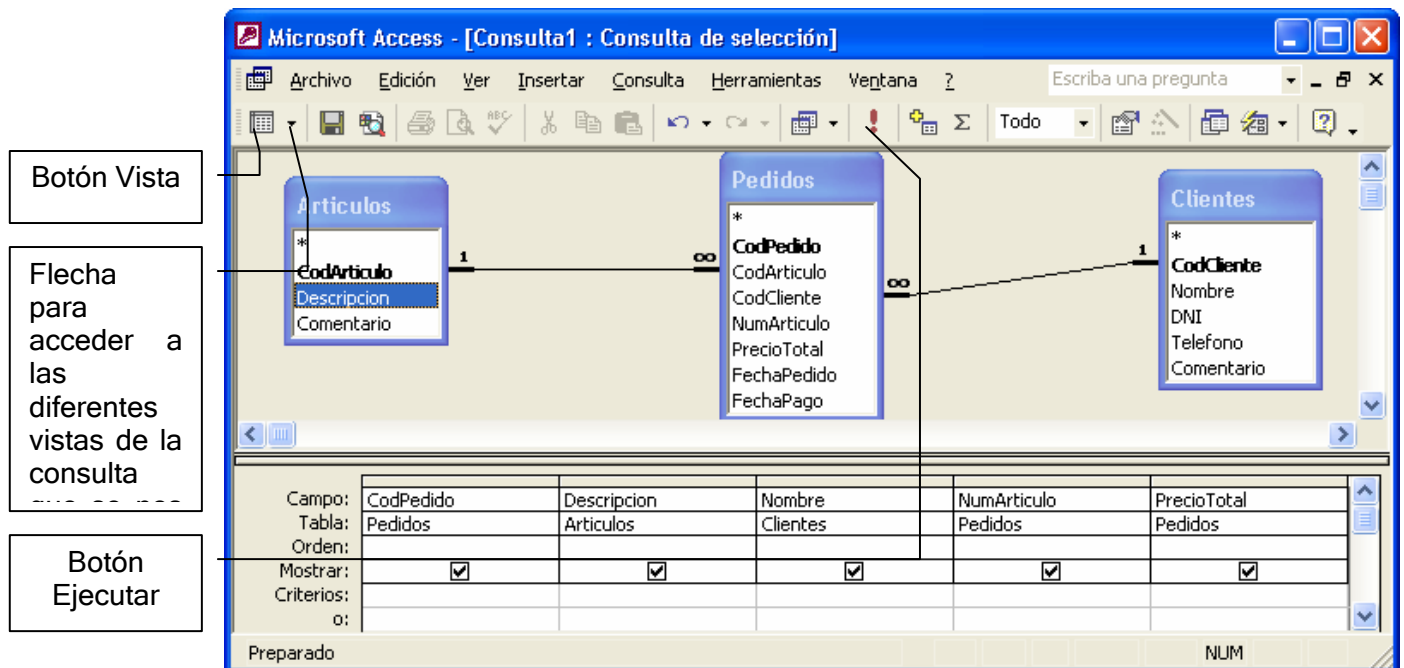
- Código del pedido
- Descripción del artículo que se vendió en el pedido
- Nombre del cliente que efectuó el pedido
- Número total de artículos solicitados en el pedido
- Precio total del pedido

Abra la base de datos Gestión de Pedidos, seleccione la ficha Consultas, elija la opción Crear una consulta en vista diseño pulsando dos veces sobre ella. A continuación agregue las tres tablas. Si se ha denominado de la misma forma a los campos claves de Clientes y Artículos en estas, que en la tabla Pedidos, Access de forma automática reconocerá que existe una relación entre dichas tablas, por lo que unirá dichos campos mediante una línea.



Agregue a la consulta los campos solicitados pulsando dos veces sobre ellos.

Debemos tener la siguiente pantalla:



En el caso de que quiera conocer los comandos usados en la select que acaba de realizar, pulse sobre el la flecha existente a la derecha del botón Vista y seleccione la opción Vista SQL o bien pulse con el botón derecho sobre la parte de la pantalla que nos enseña las tres tablas y las relaciones existentes entre ella y de entre las opciones que se le presenta elija Vista SQL.

Tanto si hace una cosa como otra obtiene una pequeña pantalla en la que se le muestra el código de la select, en nuestro caso:

```
SELECT     Pedidos.CodPedido,      Articulos.Descripcion,      Clientes.Nombre,
Pedidos.NumArticulo, Pedidos.PrecioTotal
FROM Clientes INNER JOIN (Articulos INNER JOIN Pedidos ON Articulos.CodArticulo = Pedidos.CodArticulo) ON Clientes.CodCliente = Pedidos.CodCliente
GROUP BY  Pedidos.CodPedido, Articulos.Descripcion, Clientes.Nombre, Pedidos.NumArticulo, Pedidos.PrecioTotal;
```

Puede modificar la consulta en esta pantalla.

Para volver al modo diseño pulse nuevamente sobre la flecha existente a la derecha del botón Vista y elija la opción Vista Diseño.

Finalmente ejecute la consulta pulsando el botón Ejecutar.

Guarde la consulta con el nombre Consulta1.

## 16.2 Consulta 2

Mostrar los siguientes de cada cliente:

- Nombre
- DNI
- Total de las compras realizadas

Abra la base de datos Gestión de Pedidos, seleccione la ficha Consultas, elija la opción Crear una consulta en vista diseño pulsando dos veces sobre ella. A continuación agregue las tablas Clientes y Pedidos. Incorpore los campos Nombre y DNI de la tabla Clientes y el campo PrecioTotal de la tabla Pedidos.

Pulse sobre el botón Totales (son el símbolo Sumatorio) de la barra de herramientas y elija la opción Agrupar en la nueva fila Total que ha salido para los campos Nombre y DNI y especifique Suma para el campo PrecioTotal.

Botón Totales

Campo:	Nombre	DNI	PrecioTotal
Tabla:	Clientes	Clientes	Pedidos
Total:	Agrupar por	Agrupar por	Suma
Orden:			
Mostrar:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criterios:			
o:			

Preparado NUM

Ejecute la consulta.

Guarde la consulta con el nombre Consulta2 y pulse sobre el botón Totales para volver al modo normal.

Si cambiamos a la vista SQL mediante el botón Vista obtenemos:

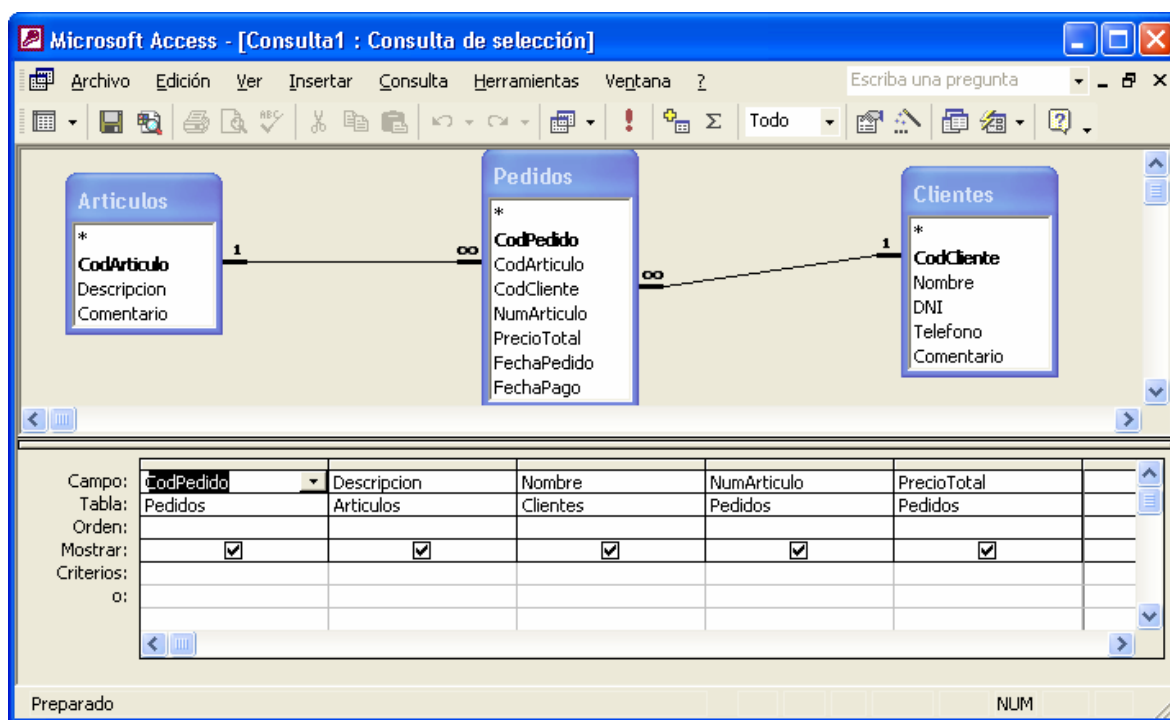
```
SELECT Clientes.Nombre, Clientes.DNI, Sum(Pedidos.PrecioTotal) AS
SumaDePrecioTotal
FROM Clientes INNER JOIN Pedidos ON Clientes.CodCliente =
Pedidos.CodCliente
GROUP BY Clientes.Nombre, Clientes.DNI;
```

### 16.3 Consulta 3

Podemos modificar una consulta de las que ya tenemos hechas.

Para ello vaya a la ficha Consultas y seleccione la consulta Consulta1, pulse el botón Diseño y asegúrese que se encuentra en la vista Diseño. Recuerde que puede acceder a dicha vista al pulsar en la flecha de la derecha del botón Vista y seleccionando la opción Vista Diseño.

La consulta que hemos recuperado tenía el siguiente aspecto:



Para crear la nueva consulta sin modificar la actual pulse sobre la opción Archivo en el menú principal y elija Guardar como y póngale el nombre Consulta3.

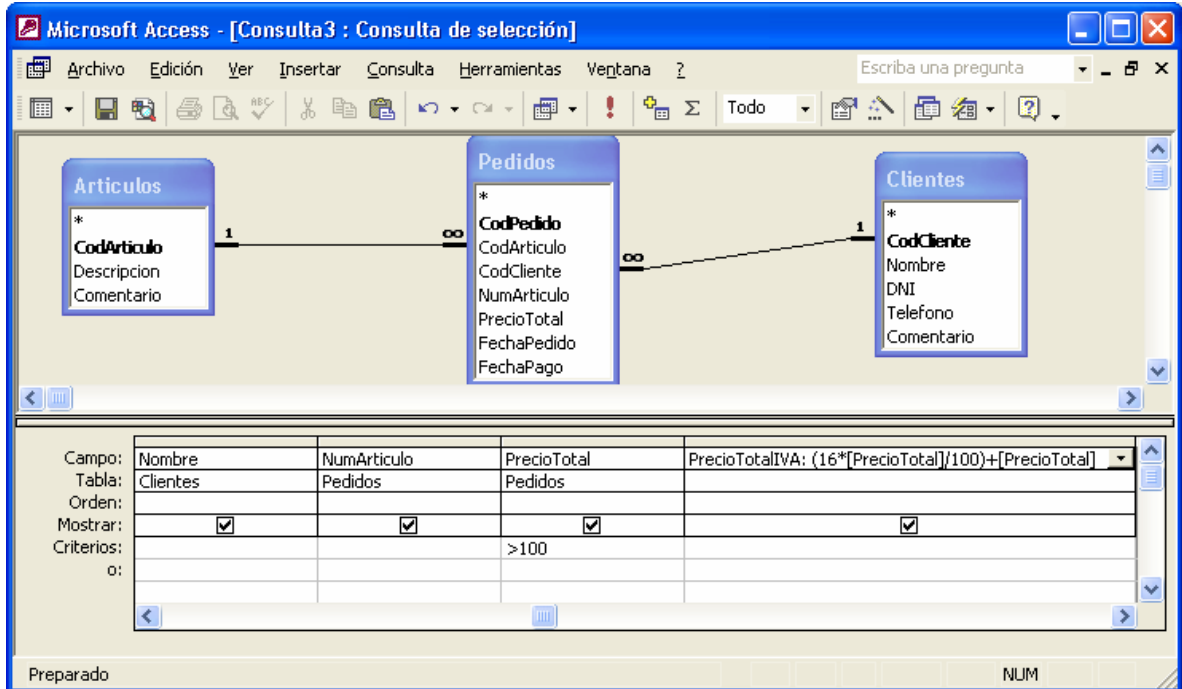
La Consulta3 ahora mismo devuelve los datos de los pedidos efectuados por todos los clientes tal y como hacía la consulta1. Vamos a restringirla haciendo que sólo nos saque aquellos pedidos cuyo Precio superase los 100 euros.

Para ello en la columna PrecioTotal de la vista diseño agregue el Criterio: > 100.

Vamos a realizar otro cambio en la consulta, pues queremos que nos muestre otro campo que será el Precio con el IVA (16%) de cada pedido.

Para ello sitúe el puntero sobre la siguiente casilla vacía después del último campo de la consulta (en nuestro caso Precio Total), pulse, teclee PrecioTotalIVA: (16\*[PrecioTotal]/100)+ [PrecioTotal] y pulse la tecla Return.

La consulta debe presentar el siguiente aspecto:

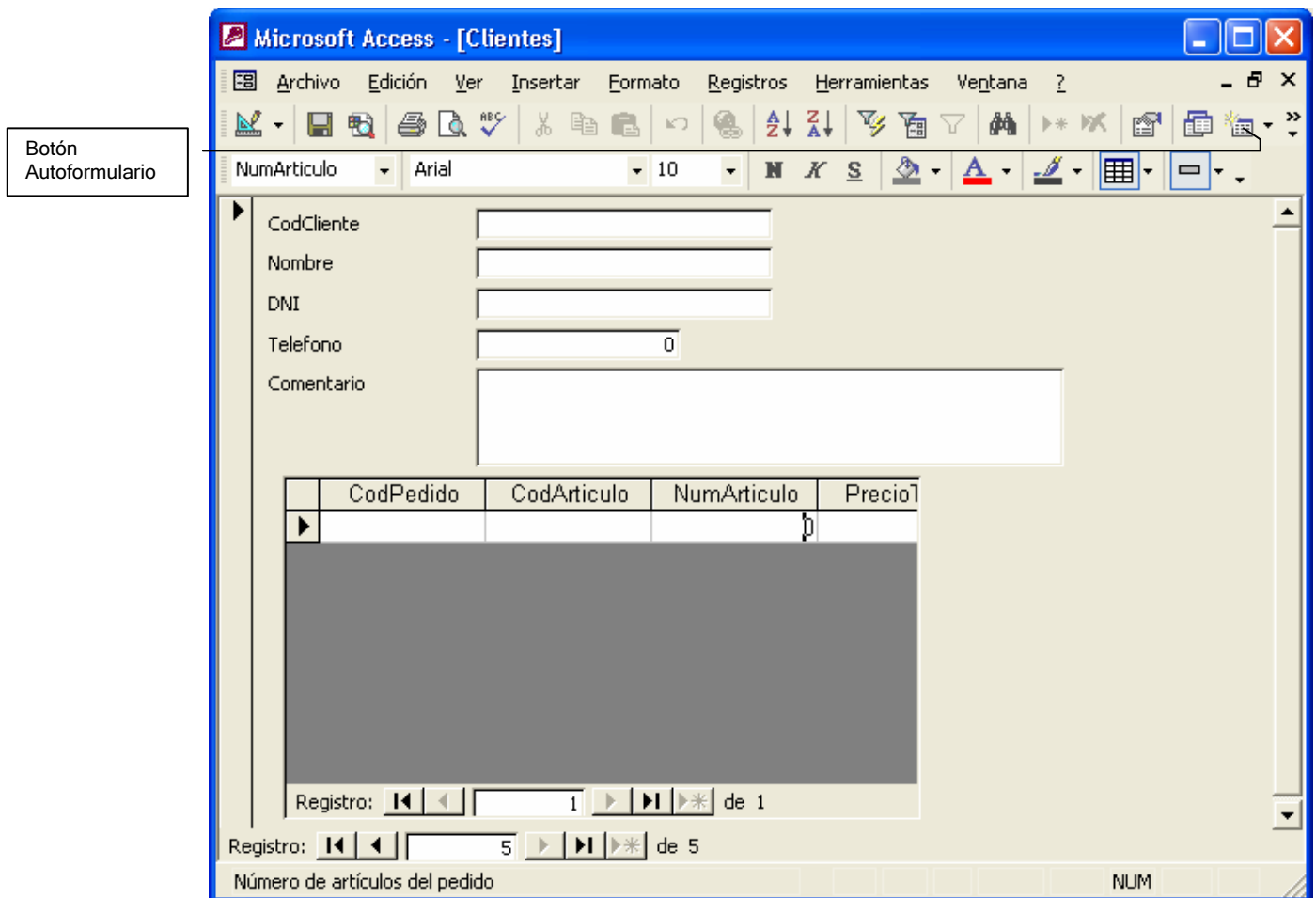


Si cambiamos a la vista SQL mediante el botón Vista obtenemos:

```
SELECT Pedidos.CodPedido, Articulos.Descripcion, Clientes.Nombre,
Pedidos.NumArticulo, Pedidos.PrecioTotal, (16*[PrecioTotal]/100)+[PrecioTotal]
AS PrecioTotalIVA
FROM Clientes INNER JOIN (Articulos INNER JOIN Pedidos ON
Articulos.CodArticulo = Pedidos.CodArticulo) ON Clientes.CodCliente =
Pedidos.CodCliente
WHERE (((Pedidos.PrecioTotal)>100));
```

## 17. Crear formulario

Vamos a crear un Autoformulario de la tabla Clientes, para ello seleccione dicha tabla y pulse en la flecha que se encuentra a la derecha del botón Nuevo objeto y dentro de elija Autoformulario. Obtendrá:



Observe que existen dos botoneras de desplazamiento y creación, uno para los clientes y otro para los pedidos. Esto se ha generado así porque la tabla Clientes está relacionada con la tabla Pedidos mediante una relación cero a muchos. Esto es un cliente puede tener muchos pedidos o no tener ninguno.

Si quiere modificar el aspecto del formulario pulse sobre el botón Vista de la barra de herramientas y elija la opción Vista diseño.

Para finalizar guarde el formulario con el nombre: Formulario Clientes - Pedidos.

## 18. Crear informes

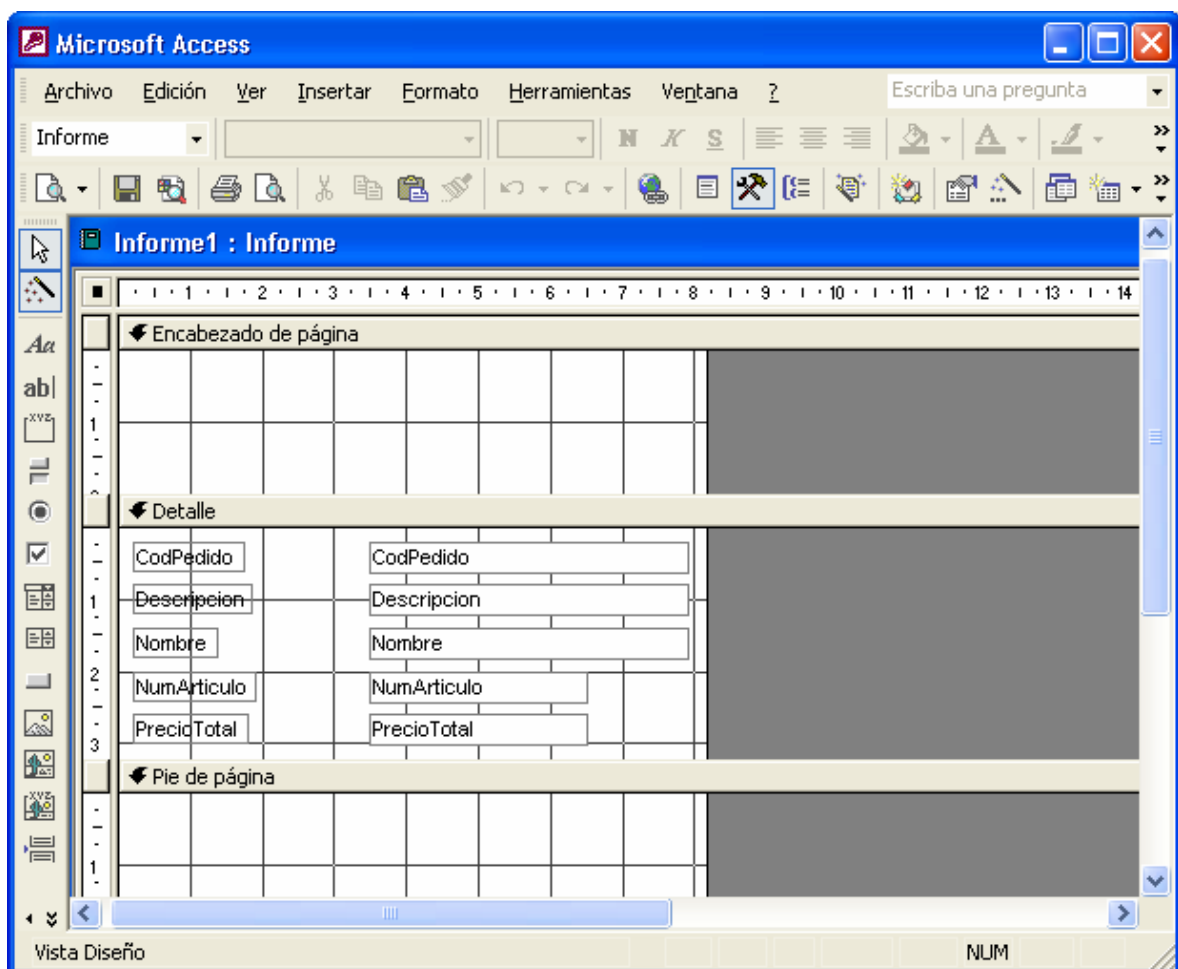
Vamos a crear un informe a partir de los datos obtenidos en la consulta: Consulta1 que recordemos mostraba los siguientes datos de los pedidos existentes:

- Código del pedido
- Descripción del artículo que se vendió en el pedido
- Nombre del cliente que efectuó el pedido
- Número total de artículos solicitados en el pedido
- Precio total del pedido

Para crear el informe, seleccione la consulta Consulta1 y pulse sobre la flecha que muestra el botón Nuevo objeto de la barra de herramientas. Pulse sobre la opción Autoinforme.

Le aparecerá un informe con los datos que mostraba la consulta.

En el caso de que quiera alterar el diseño del Autoinforme que acaba de realizar, pulse en el botón Vista y seleccione la opción Vista diseño.



## 19. El autor

Licenciado en Matemáticas. Universidad de Sevilla. (2001)

### Experiencia docente:

Profesor en la Universidad de Sevilla en el Master de Tecnologías de Análisis para la Sociedad de la información.

Profesor de Matemáticas y Ciencias de la Naturaleza de Educación Secundaria Obligatoria en el IES Mariana de Pineda (Dos Hermanas/Sevilla).

Amplia experiencia impartiendo clases en academias y a particulares de matemáticas a distintos niveles educativos, principalmente a niveles de secundaria obligatoria, bachillerato y universidad.

Elaboración de programaciones y unidades didácticas. Conocimiento de la estructura, objetivos y contenidos del sistema educativo.

Impartición de cursos sobre aprendizaje de distintas aplicaciones informáticas a sus usuarios finales en la Junta de Andalucía.

### Otra Experiencia Profesional:

Septiembre 2001 / Junio 2005: Tareas de análisis y programación de aplicaciones en entorno Oracle 9i para la Junta de Andalucía.

Junio 2005 / Septiembre 2007: Tareas de análisis y programación de aplicaciones en entorno Oracle 9i y Cobol para el Servicio Andaluz de Salud.

Marzo 2001 / Febrero 2004: Tareas de consultaría para la aplicación S.R.P. de la Consejería de Justicia de la Junta de Andalucía.

Formación:

Octubre 2004 - Marzo 2005: Curso de Adaptación Pedagógica (C.A.P.) en el Instituto de Ciencias de la Educación de la Universidad Complutense de Madrid.

Máster en Tecnologías de Análisis para la Sociedad de la Información - Universidad de Sevilla (Edición 2003- 2004)

Título de Experto Universitario en Tecnologías de Análisis para la Sociedad de la información (Edición 2001 - 2002).

Curso Superior de Capacitación en las Nuevas Tecnologías de la Información y la Comunicación. (Periodo: 01/10/2001-23/01/2002, Duración: 400 horas) Organizado por la Consejería de Empleo y Desarrollo Tecnológico de la Junta de Andalucía).

Correo electrónico: [f\\_flores\\_gil@hotmail.com](mailto:f_flores_gil@hotmail.com)

