

## TEMA XI.APLICACIÓN DE BIBLIOTECA.

### Reglas semánticas:

Cada ejemplar de un mismo libro tendrá un registro diferente.  
Un lector puede disponer de varios libros.  
La signatura será única.

### TABLAS

#### LIBROS

CAMPO	TIPO DE DATOS	TAMAÑO	FORMATO	MÁSCARA
<u>Registro</u>	Autonumérico			
Signatura	Texto	6		99-99-99
Título	Texto	35	>	
Autor	Texto	25	>	
Páginas	Numérico	Entero		
Precio	Numérico	Simple	Euro	
Editorial	Texto	15	>	
Año	Numérico	Entero		

El campo signatura se utiliza para indicar la ubicación del libro. Los dos primeros dígitos identifican el armario, los dos siguientes la balda de ese armario, y los últimos la situación del libro dentro de esa balda.

El campo Registro es una clave principal de esta tabla. Al ser de tipo autonumérico, se incrementará automáticamente cada vez que se inserte un registro nuevo.

#### LECTORES

CAMPO	TIPO DE DATOS	TAMAÑO	FORMATO	MÁSCARA
<u>Dni</u>	Texto	9		99999999L
Nombre	Texto	35	>	
Fecha nac	Fecha		Fecha corta	
Dirección	Texto	20		
Teléfono	Texto	9	>	
Curso	Texto	10	>	
Grupo	Texto	1	>	L

La clave principal en esta tabla sería el campo Dni.

#### PRESTAMOS

CAMPO	TIPO DE DATOS	TAMAÑO	FORMATO	MÁSCARA
<u>Registro</u>	Entero			
Dni	Texto	9		99999999L
Fecha	Fecha		Fecha corta	

La tabla préstamos nos permite llevar un control de los libros que tiene cada lector y la fecha del préstamo. Cuando el lector devuelve el libro, se borra el registro correspondiente en esta tabla. Este proceso supone que perderíamos información de los libros que han sido prestados y devueltos. Por ello vamos a crear una tabla HISTÓRICO para controlar los libros que han sido prestados y devueltos.

En este caso consideramos clave principal el campo Registro puesto que un libro lo puede tener un lector y un lector puede tener varios libros.

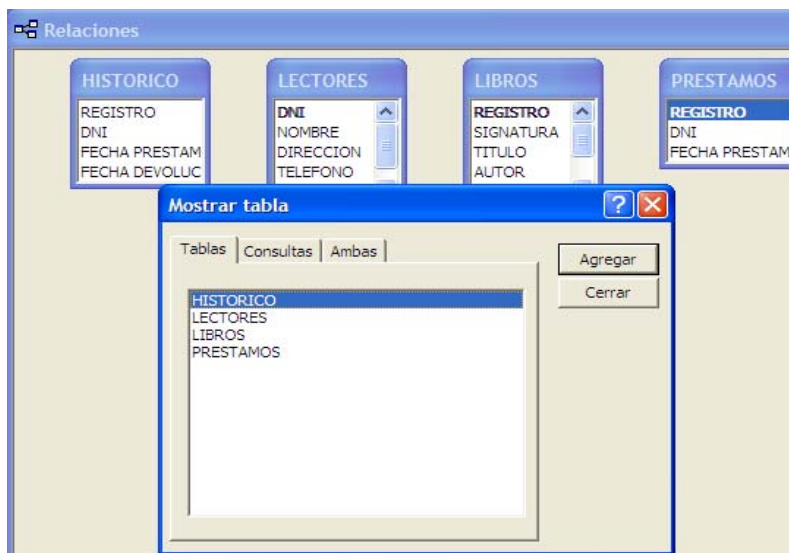
## HISTORICO

CAMPO	TIPO DE DATOS	TAMAÑO	FORMATO	MÁSCARA
Registro	Entero			
Dni	Texto	9		99999999L
Fecha Prest	Fecha		Fecha corta	
Fecha Devol	Fecha		Fecha corta	

## RELACIONES ENTRE LAS TABLAS.

Necesitamos indicarle a nuestro SGBD la forma de combinar los datos que figuran en las diferentes tablas de nuestra aplicación. Para ello establecemos relaciones entre campos comunes de las tablas, de forma que las tablas que están relacionadas se comporten como una sola tabla, evitando así, redundancias e incongruencias en los datos almacenados.

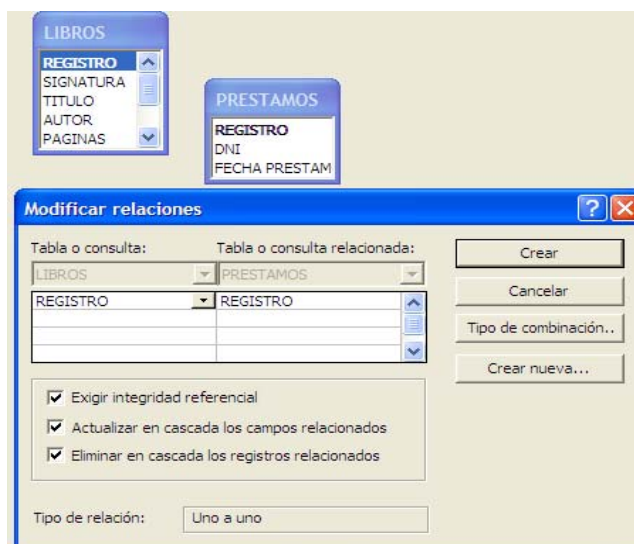
Seleccione el icono de relaciones y agregamos las cuatro tablas a la ventana de relaciones.



Una vez agregadas las cuatro tablas, tenemos que establecer las siguientes relaciones

- Un libro puede ser prestado a un lector.

Relación Uno a Uno entre Libros.Registro y Prestamos.Registro.



Arrastre el campo Libros.Registro sobre Prestamos.Registro.

En cada una de las relaciones debemos exigir integridad referencial de los datos, actualización y eliminación en cascada.

- Un libro, una vez devuelto, se puede prestar de nuevo.

Relación Uno a Varios Libros.Registro y Historico.Registro.

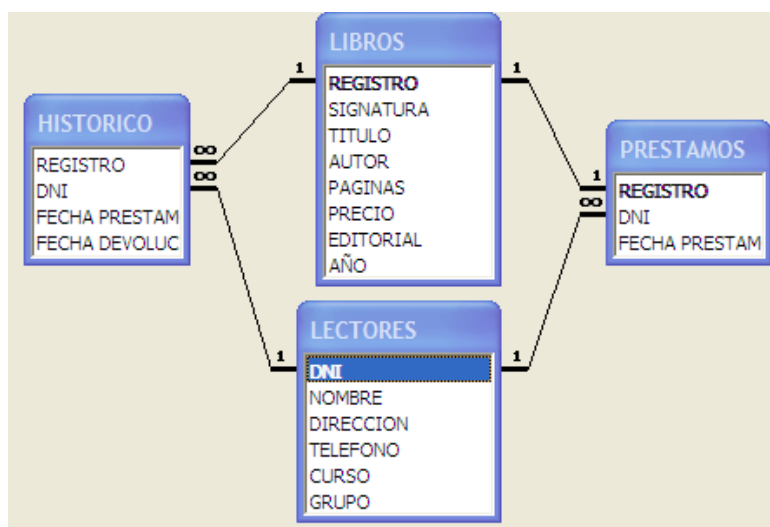
Arrastre el campo Libros.Registro sobre Historico.Registro.

- Un lector puede tener varios libros prestados y devolverlos.

Relación Uno a Varios Lectores.Dni y Prestamos.Dni.

Relación Uno a Varios Lectores.Dni y Historico.Dni.

Arrastre el campo Dni de Lectores sobre Dni de Prestamos y Historico.



Relaciones que debemos de obtener

Existen relaciones Varios a Varios que no se pueden representar directamente. Para definir las necesitamos una tercera tabla de unión entre ambas. Por ejemplo:

Pedidos(id.pedido, cliente, id.producto)  
 Productos(id.producto, descripción)

La relación en este caso es de Varios a Varios por el campo id.producto. Para evitarla se crea una tercera tabla cuya clave consta de dos campos que son las claves externas de ambas externas.

Pedidos(id.pedido, cliente)  
 Detalle(id.pedido, id.producto)  
 Productos(id.producto, descripción)

Cuando una tabla está implicada en una relación y actúa como tabla principal, su ventana hoja de datos presentará una nueva característica, para permitir abrir con un solo clic hojas de datos secundarias en las que se visualizan los registros relacionados de la tabla secundaria.

LIBROS : Tabla								
	REGIST	SIGNATU	TITULO	AUTOR	PAGINAS	PRECIO	EDITORIAL	AÑO
+	1	01-01-01	DON QUIJOTE	CERVANTES	345	9,00 €	ANAYA	1985
▶	2	01-01-02	LAZARILLO	ANONIMO	123	6,00 €	EVEREST	2000
		DNI	FECHA PREST/	FECHA DEVOL				
	▶	22222222J	01/01/2004	31/07/2004				
*								
+	3	01-01-03	EL LABERINTO DE LA	EDUARDO DE	180	7,00 €	PLANETA	1988
*		Américo)			0	0,00 €		0

## CONSULTAS.

Las consultas nos van a permitir trabajar con un determinado conjunto de datos de una o varias tablas. Estos datos pueden utilizarse en un formulario o un informe.

Las podemos clasificar en dos tipos:

1. Consultas de **selección**
2. Consultas de **acción**

Las consultas se pueden efectuar utilizando la cuadrícula **QBE** (Query By Example) o directamente en **SQL** (Structured Query Language. Ver apéndice).

### CONSULTAS DE SELECCIÓN.

Se definen :

1. Agregar las tablas necesarias.
2. Indicar en la cuadrícula QBE los campos necesarios.
3. En caso necesario, indicar criterios de selección.
4. En caso necesario, agrupar datos.

Ejemplo de consulta de todos los libros ordenados por autor:

Consulta1 : Consulta de selección				
<div style="border: 1px solid blue; padding: 5px; width: fit-content;"> <p>LIBROS</p> <p>REGISTRO ▲</p> <p>SIGNATURA ▾</p> <p>TITULO ▾</p> <p>AUTOR ▾</p> <p>PAGINAS ▾</p> </div>				
Campo:	REGISTRO	SIGNATURA	TITULO	AUTOR
Tabla:	LIBROS	LIBROS	LIBROS	LIBROS
Orden:				Ascendente
Mostrar:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criterios:				
o:				

En una consulta pueden intervenir varias tablas e indicar determinados criterios de selección. Por ejemplo muestra los libros prestados antes del 31/03/04 cuyo precio sea superior a 6 euros.



### CRITERIOS DE SELECCIÓN.

Además de los operadores **lógicos Y, O, Negado** y los operadores **relacionales >, <, >=, <=, <>**, se pueden utilizar los siguiente operadores:

#### ENTRE, EN, COMO.

<b>ENTRE</b>	Especifica un rango de valores	<b>Entre</b> 1000 <b>Y</b> 2000 <b>Entre</b> #3/10/99# <b>Y</b> #20/12/99#
<b>EN</b>	Especifica una lista de valores	<b>En</b> ("PLANETA";"ESPASA")
<b>COMO</b>	Muestras en campos de texto utilizando comodines *, ?	<b>Como</b> "B*" <b>Como</b> "1B?"

#### DIA, MES, AÑO, FECHA(), DIASEMANA

<b>Dia</b> (fecha)	Valor de 1 a 31	Dia([fecha])=15
<b>Mes</b> (fecha)	Valor de 1 a 12	Mes([fecha])=1
<b>Año</b> (fecha)	Valor de 100 a 9999	Año([fecha])<1999
<b>DiaSemana</b> (fecha)	Valor 1 (domingo) a 7 (sábado)	DiaSemana([fecha])=5
<b>Fecha</b> ()	Fecha del sistema	=Fecha() - 31

### ORDENAR LOS DATOS.

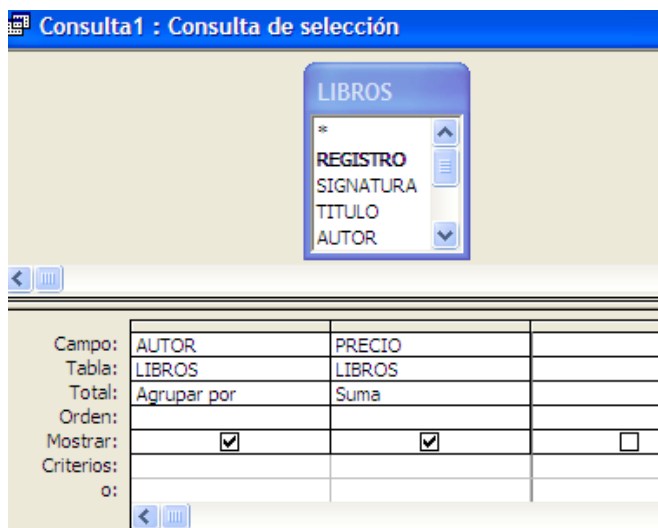
Los resultados de la consulta se pueden mostrar en orden, ascendente o descendente, por uno o varios campos. Basta indicarlo en la fila **orden** de la cuadrícula QBE.

**AGRUPAR DATOS.**

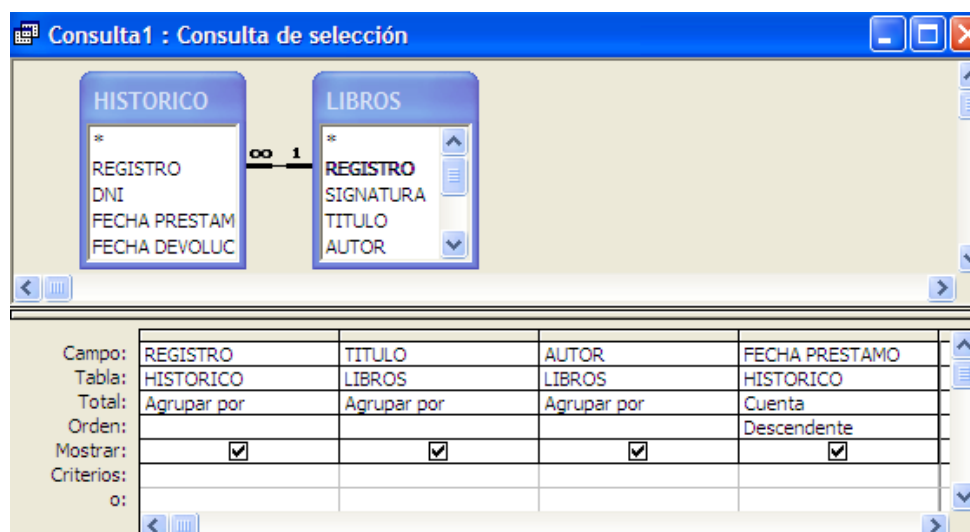
Para mostrar los totales en la cuadrícula QBE, se puede elegir el icono totales  $\Sigma$  o en el menú **VER totales**. Al seleccionar la opción totales, significa que la consulta mostrará **una fila por cada grupo de registros idénticos**. Para calcular totales de grupos, se puede reemplazar “Agrupar por” por las funciones :

<b>SUMA</b>	Suma los valores del campo para el grupo
<b>PROMEDIO</b>	Calcula valor medio del campo para el grupo
<b>MAX</b>	Muestra el máximo valor del grupo
<b>MIN</b>	Muestra el mínimo
<b>CUENTA</b>	Cuenta los registros en los que el campo del grupo es no nulo
<b>PRIMERO</b>	Obtiene el primer valor del grupo
<b>ULTIMO</b>	Obtiene el último valor del grupo
<b>DESVEST</b>	Desviación estándar
<b>VAR</b>	Varianza

**Ejemplo calculo del total importe de los libros agrupados por autor.**



**Ejemplo de consulta para obtener recuento de los libros que han sido prestados, por orden descendente de número de prestamos.**



## PARÁMETROS EN CONSULTAS.

Se pueden introducir parámetros (**texto entre corchetes [ ]**) dentro de los criterios de las consultas, para que al ejecutar la consulta, nos pregunte en un cuadro de diálogo, por el valor que deseamos para ese criterio de selección.

Por ejemplo podemos diseñar una consulta para mostrar los libros de un autor cuyo nombre lo pregunte al ejecutar la consulta. Para ello indicaremos en criterios de la columna AUTOR el parámetro **[Dame nombre de AUTOR]**.



## EJERCICIOS

1. Mostrar libros de una editorial.
2. Mostrar libros de dos editoriales ordenados cronológicamente.
3. Consultar libros prestados entre dos fechas.
4. Consultar libros cuyo nombre empieza por la letra "C".
5. Consultar el valor medio de todos los libros. El máximo y el mínimo y cuántos libros tengo.
6. Consultar todos los préstamos que se realizaron en miércoles, indicando la fecha, el libro, y lector.
7. Mostrar los préstamos de los últimos 15 días desde la fecha de hoy.

## CONSULTAS DE ACCIÓN.

### Consulta de creación de tablas.

Podemos crear una nueva tabla a partir del resultado obtenido en una consulta.

### Consulta de eliminación de registros.

Permite eliminar de una o varias tablas, un grupo de registros seleccionados en la consulta.

### Consulta para agregar registros.

Podemos añadir registros que resulten de una consulta a otra tabla.

### Consulta para actualizar registros.

Podemos modificar el valor de ciertos registros de una o varias tablas.



## FORMULARIOS.

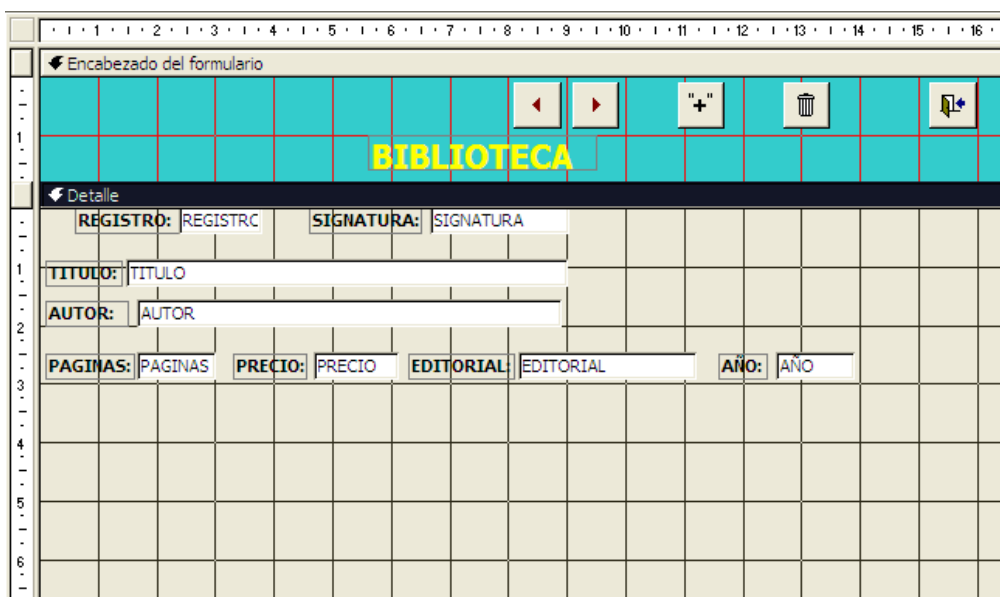
Los formularios constituyen el principal intermediario entre los usuarios y la aplicación y nos permiten :

- Presentar y editar los datos de una tabla o consulta.
- Presentar mensajes.
- Añadir controles para automatizar tareas.

Para nuestra aplicación ejemplo, necesitaremos al menos dos formularios. Todos ellos incluyen controles, en nuestro caso, botones de comando comunes para ir al registro anterior, registro siguiente, agregar registro, eliminar registro y cerrar formulario:

### FORMULARIO LIBROS.

Usando la tablas libros, diseñamos el siguiente formulario, donde en el encabezado incluye botones de comando, creados desde el cuadro de herramientas  y activado el asistente . (Registro anterior, siguiente, agregar registro, eliminar registro y cerrar formulario)



Encabezado del formulario

BIBLIOTECA

Detalle

REGISTRO: REGISTRC SIGNATURA: SIGNATURA

TITULO: TITULO

AUTOR: AUTOR

PAGINAS: PAGINAS PRECIO: PRECIO EDITORIAL: EDITORIAL AÑO: AÑO

Análogamente se diseñaría el formulario para los lectores de nuestra biblioteca..  
FORMULARIO LECTORES.

En ambos formularios se pueden realizar todas las operaciones de actualización, salvo buscar un libro o un lector. Para resolver la búsqueda podemos utilizar un botón de comando para buscar registros o bien un cuadro combinado que muestre los libros o lectores ordenados alfabéticamente y nos permita seleccionarlos.

**Asistente para cuadros combinados**

Este asistente crea un cuadro combinado, que muestra la lista con los valores que puede elegir. ¿Cómo desea que el cuadro combinado obtenga sus valores?

- Deseo que el cuadro combinado busque los valores en una tabla o consulta.
- Escribiré los valores que desee.
- Buscar un registro en el formulario según el valor que he seleccionado en el cuadro combinado.

Cancelar < Atrás Siguiente > Finalizar

**Asistente para cuadros combinados**

¿Qué ancho desea que tengan las columnas en el cuadro combinado?

Para ajustar el ancho de una columna, arrastre su borde derecho hasta el ancho que desee o haga doble clic en el borde derecho del título de columna para obtener un ajuste perfecto.

Quitar la columna clave (se recomienda)

NOMBRE
MARTIN PEREZ PEPE
GARCIA RUIZ ANTONIO

Cancelar < Atrás Siguiente > Finalizar

**Asistente para cuadros combinados**

¿Qué campos contienen los valores que desea incluir en el cuadro combinado? Los campos seleccionados se convertirán en columnas en el cuadro combinado.

Campos disponibles: DIRECCION, TELEFONO, CURSO, GRUPO

Campos seleccionados: DNI, NOMBRE

Cancelar < Atrás Siguiente > Finalizar

**Asistente para cuadros combinados**

¿Qué etiqueta desea para el cuadro combinado?

**NOMBRE**

Esto es todo lo que el asistente necesita para crear el cuadro combinado.

Ayuda para personalizar el cuadro combinado.

Cancelar < Atrás Siguiente > Finalizar

## SUBFORMULARIOS.

Hasta ahora podemos gestionar los libros y lectores de nuestra biblioteca, pero no podemos prestar, devolver o editar los libros prestados. Sería deseable por lo tanto mostrar en el formulario libros, quien posee ese libro, y en el formulario lectores, mostrar todos los libros de ese lector.

Creamos la CONSULTA PRESTADOS:

El diagrama muestra una consulta denominada 'PRESTADOS' que se basa en la tabla 'PRESTAMOS'. La consulta incluye los campos 'REGISTRO', 'DNI', 'FECHA PRESTAMO' y 'NOMBRE'. El campo 'NOMBRE' proviene de la tabla 'LECTORES'. La relación entre 'LECTORES' y 'PRESTAMOS' es uno a muchos (1 a ∞).

Campo:	REGISTRO	DNI	FECHA PRESTAMO	NOMBRE
Tabla:	PRESTAMOS	PRESTAMOS	PRESTAMOS	LECTORES
Orden:				
Mostrar:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criterios:				
o:				

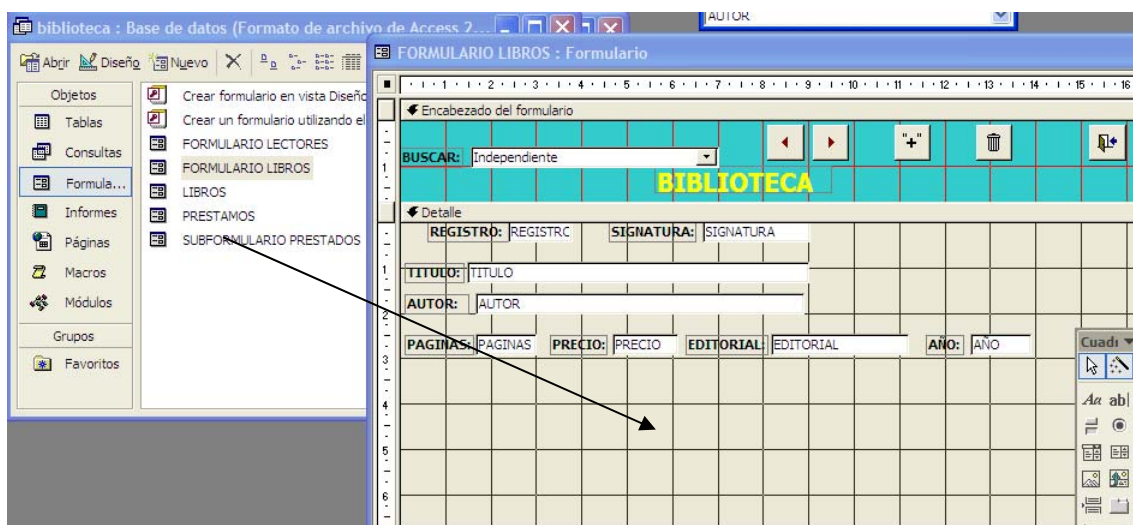
A partir de esta consulta creamos el SUBFORMULARIO PRESTADOS

El subformulario está dividido en tres secciones:

- Encabezado del formulario:** Contiene los campos 'D.N.I.', 'NOMBRE' y 'FECHA'.
- Detalle:** Contiene los campos 'REGISTRO', 'DNI', 'NOMBRE' y 'FECHA PRESTA'.
- Pie del formulario:** No contiene campos visibles.

Este formulario muestra todos los datos de los libros que están prestados.

Access nos permite insertar un formulario dentro de otro, de forma que quedan vinculados por el campo clave del formulario principal y el campo común del subformulario. De esta forma si insertamos este formulario dentro del FORMULARIO LIBROS, al seleccionar un libro se mostrará quién lo posee. Para crear la jerarquía de formularios, abra el formulario principal en diseño y arrastre desde el panel de control el formulario secundario.



El Formulario/Subformulario quedaría:

**FORMULARIO LIBROS : Formulario**

BUSCAR:

**BIBLIOTECA**

REGISTRO:  SIGNATURA:

TITULO:

AUTOR:

PAGINAS:  PRECIO:  EDITORIAL:  AÑO:

SUBFORMULARIO PRESTADOS:

	D.N.I.	NOMBRE	FECHA
▶	<input type="text" value="2 11111111L"/>	<input type="text" value="MARTIN PEREZ PEPE"/>	<input type="text" value="01/01/2004"/>

Análogamente creamos la CONSULTA LIBROS PRESTADOS:

**CONSULTA LIBROS PRESTADOS : Consulta de selección**

LIBROS: SIGNATURA, TITULO, AUTOR, PAGINAS, PRECIO

PRESTAMOS: \*, REGISTRO, DNI, FECHA PRESTAM

Campo:	<input type="text" value="DNI"/>	REGISTRO	SIGNATURA	TITULO	FECHA PRESTAMO
Tabla:	PRESTAMOS	PRESTAMOS	LIBROS	LIBROS	PRESTAMOS
Orden:					
Mostrar:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criterios:					
o:					

El SUBFORMULARIO LIBROS PRESTADOS tipo continuo y basado en esta consulta.

**SUBFORMULARIO LIBROS PRESTADOS : Formulario**

Encabezado del formulario

	REGISTRO	SIGNAT	TITULO		FECHA
--	----------	--------	--------	--	-------

Detalle

DNI	REGISTRO	SIGNATURA	TITULO		FECHA PRES
-----	----------	-----------	--------	--	------------

Pie del formulario

Insértalo en el formulario lectores.

REGISTRO	SIGNAT	TITULO	FECHA
11111111L	2	01-01-02 LAZARILLO	01/01/2004
*			

Ahora visualizamos por cada lector los libros que posee.

### ¿Cómo prestar un libro?

Bastaría con un cuadro combinado en el subformulario prestados que seleccione el lector de la tabla lectores y almacenase el campo dni en el dni de este formulario. Actualizar el campo fecha.

### ¿Cómo nos devuelve un libro el lector?.

Se busca el libro en el formulario libros y se elimina el lector del subformulario prestados. Al eliminar este registro, perdemos la información del histórico, es decir, no llevamos un control de los libros devueltos. Para ello debemos de añadir este registro a la tabla HISTORICO y eliminarlo.

Diseñamos la siguiente consulta de acción de datos anexados:

Campo	Tabla	Orden	Anexar a	Criterios	o
REGISTRO	Consulta PRESTADOS		REGISTRO	[Forms]![FORMULARIO LIBROS]![REGISTRO]	
DNI	Consulta PREST#		DNI		
FECHA PRESTAMO	Consulta PRESTADC		FECHA PRESTAMO		
Expr1: Ahora()			FECHA DEVOLUCIO		

Esta consulta añade a la tabla HISTORICO el libro de la consulta prestados cuyo registro coincide con el registro del libro actual, anotando como fecha devolución la fecha del sistema.

La consulta APUNTE HISTORICO se debe de ejecutar cuando se elimine el registro del SUBFORMULARIO PRESTADOS. Para ello seleccionamos el **Evento Al eliminar** de este subformulario y añadimos un procedimiento de evento que ejecute esta consulta:

```
Private Sub Form_Delete(Cancel As Integer)
    DoCmd.OpenQuery "CONSULTA APUNTE HISTORICO"
```

End Sub

Los formularios para nuestra aplicación quedarían :

## INFORMES.

Además de poder imprimir los datos de una tabla, el resultado de una consulta o los datos visibles de un formulario, Microsoft Access permite crear objetos informes en los que podemos diseñar un modelo de impresión a partir de los datos de las tabla o de consultas existentes.

Los informes se pueden generar usando el asistente o bien en vista diseño. Por ejemplo para mostrar un listado de todos los libros de la biblioteca:

**Asistente para informes**

¿Qué campos desea incluir en el informe?  
Puede elegir de más de una consulta o tabla.

Tablas/Consultas  
Tabla: LIBROS

Campos disponibles: REGISTRO, SIGNATURA, TITULO, AUTOR, PAGINAS, PRECIO, EDITORIAL, AÑO

Campos seleccionados: REGISTRO, SIGNATURA, TITULO, AUTOR, PAGINAS, PRECIO, EDITORIAL, AÑO

Cancelar < Atrás Siguiente > Finalizar

---

**Asistente para informes**

¿Desea agregar algún nivel de agrupamiento?

REGISTRO, SIGNATURA, TITULO, AUTOR, PAGINAS, PRECIO, EDITORIAL, AÑO

REGISTRO, SIGNATURA, TITULO, AUTOR, PAGINAS, PRECIO, EDITORIAL, AÑO

Prioridad

Opciones de agrupamiento... Cancelar < Atrás Siguiente > Finalizar

---

**Asistente para informes**

¿Qué tipo de ordenación desea utilizar para los registros?

Puede ordenar los registros hasta por cuatro campos, en orden ascendente o descendente.

1 TITULO Ascendente

2 Ascendente

3 Ascendente

4 Ascendente

Cancelar < Atrás Siguiente > Finalizar

---

**Asistente para informes**

¿Qué distribución desea aplicar al informe?

Distribución: En columnas, Tabular, Justificado

Orientación: Vertical, Horizontal

Ajustar el ancho del campo de forma que quepan todos los campos en una página.

Cancelar < Atrás Siguiente > Finalizar

---

**Asistente para informes**

¿Qué estilo desea aplicar?

Casual, Compacto, Corporativa, Formal, Gris claro, Negrita

Título, Etiqueta, Control de Detalle

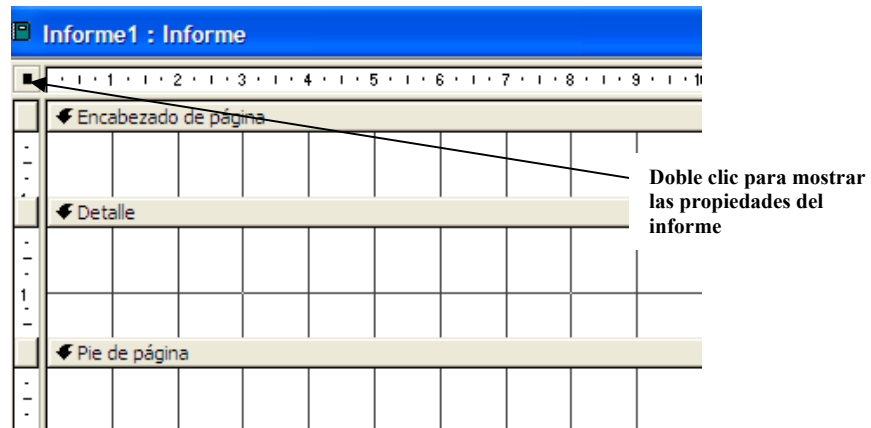
Cancelar < Atrás Siguiente > Finalizar

---

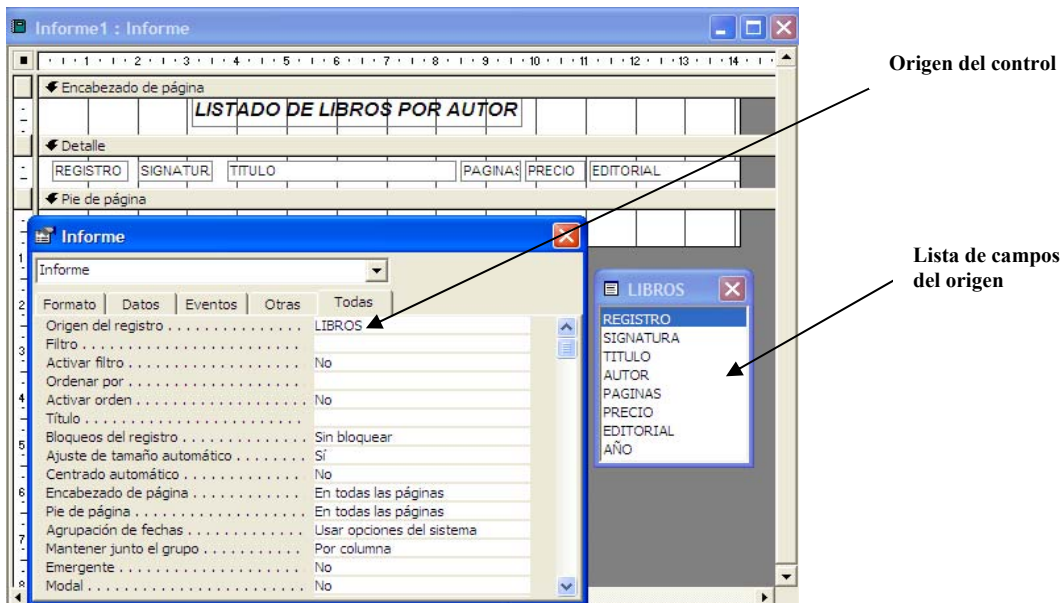
**LIBROS**


TITULO	REGISTRO	SIGNATURA	PAGINAS	PRECIO	EDITORIAL	AÑO
DON QUIJOTE	1	01-01-01	CERVANTES	345	5.00 €	ANAYA 1955
EL LABERINTO DE LA	3	01-01-03	EDUARDO DE	180	7.00 €	PLANETA 1988
LADRILLO	2	01-01-02	ANONIMO	123	6.00 €	EVEREST 2000
RINCÓN DE YCORT	4	01-01-04	CERVANTES	100	5.00 €	ANAYA 1980

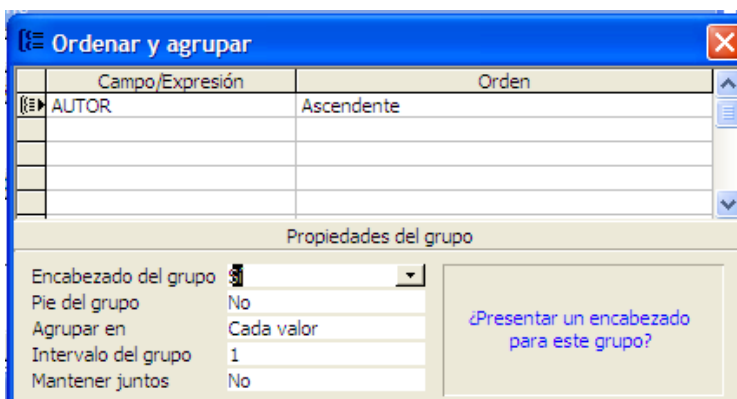
**Creando un informe en vista diseño:**



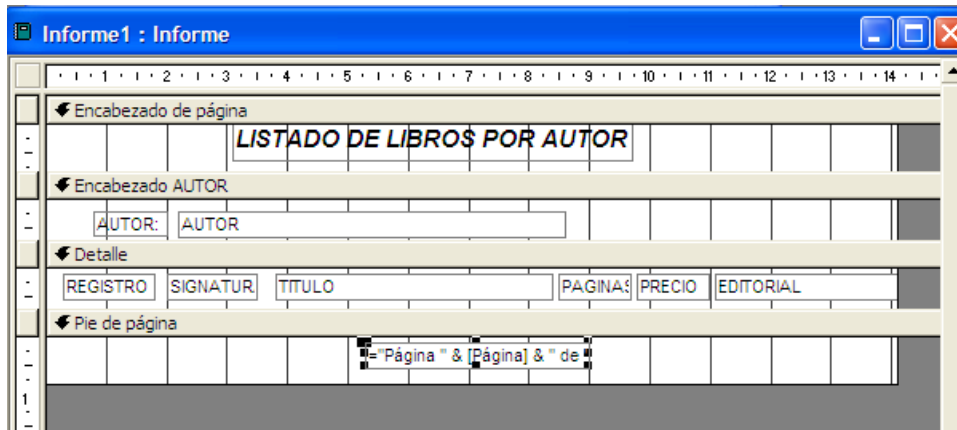
En las propiedades del informe hay que seleccionar el **Origen de los datos (LIBROS)**. Aparece una ventana con los campos disponibles, que se arrastran a la sección del informe adecuada.



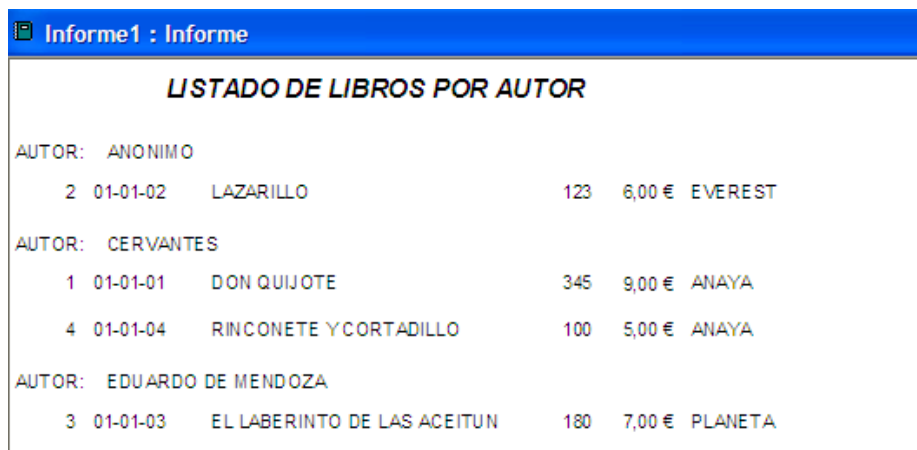
Para crear un nivel de agrupamiento (ruptura de control) abrimos el icono agrupar ordenar . Selecciona el campo de ruptura (Autor).



Aparece una nueva sección en el diseño del informe, Encabezado AUTOR, a la que arrastramos el campo autor .

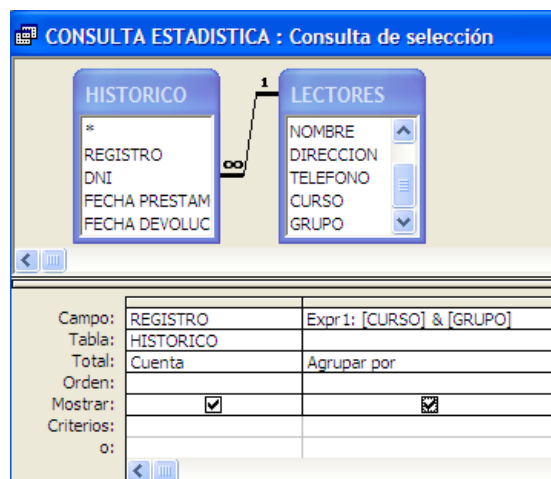


Al mostrar la vista previa del informe nos quedaría :

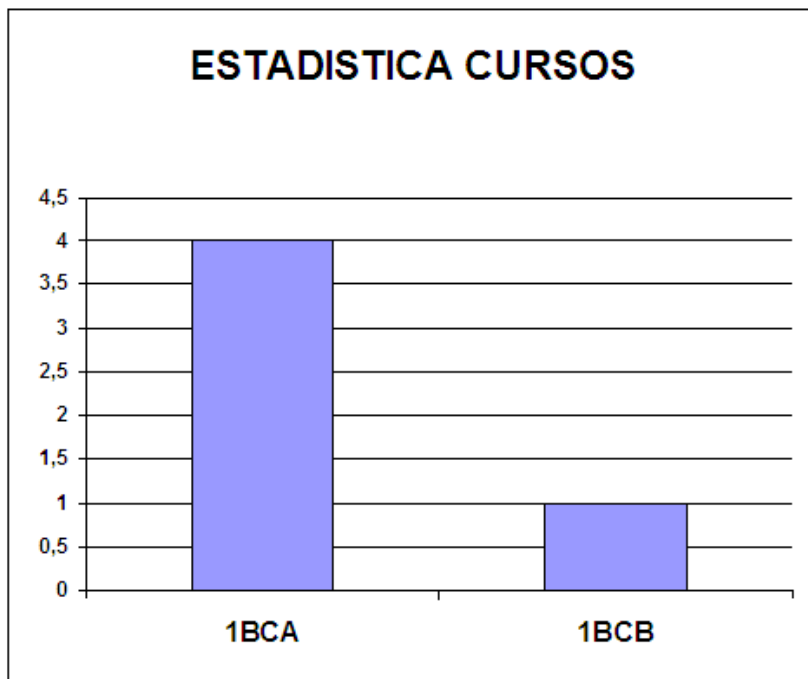


### INFORME BASADO EN UNA CONSULTA.

Los informes se pueden generar a partir de consultas de selección creadas para obtener los datos que nos convengan de las diferentes tablas de la aplicación. Por ejemplo disponemos de una consulta que cuenta el número de lectores por cursos y grupos:



A partir de esta consulta podemos generar un informe en que se inserte un gráfico de barras indicando una estadística de los lectores de la biblioteca:



## MACROS

Una macro es un conjunto de una o más acciones que cada una realiza una operación determinada, tal como abrir un formulario o imprimir un informe. Las macros pueden ayudar a automatizar las tareas comunes. Por ejemplo, puede ejecutar una macro que imprima un informe cuando el usuario haga clic en un botón de comando.

Como ejercicio crearemos una macro denominada AUTOEXEC (Se ejecuta automáticamente al abrir un fichero de access) que abrirá un formulario denominado PRINCIPAL en el que incluiremos cuatro botones de comando:

1. Botón que abra el formulario LIBROS.
2. Botón para abrir el formulario LECTORES.
3. Botón para imprimir el informe anterior
4. Botón para cerrar la aplicación de access.