

EJERCICIOS

Los ejercicios se realizarán todos sobre una misma base de datos, que tiene tres tablas:

- Tabla de Empleados.
- Tabla de Departamentos.
- Tabla de Centros de trabajo.

La estructura de estas tablas es la siguiente:

A. Tabla de Empleados. TEMPLE

NUMEMP	Integer	Número de empleado
NUMDE	Integer	Número de departamento
EXTEL	Smallinteger	Extensión telefónica del empleado
FECNA	Date	Fecha nacimiento
FECIN	Date	Fecha ingreso
SALAR	Decimal	Salario (en miles de pesetas)
COMIS	Decimal	Comisión: Es nula para todos los empleados Excepto: Vendedores, Director de vendedores y Director comercial.
NUMHI	Smallinteger	Número de hijos
NOMEM	Varchar	Nombre del empleado

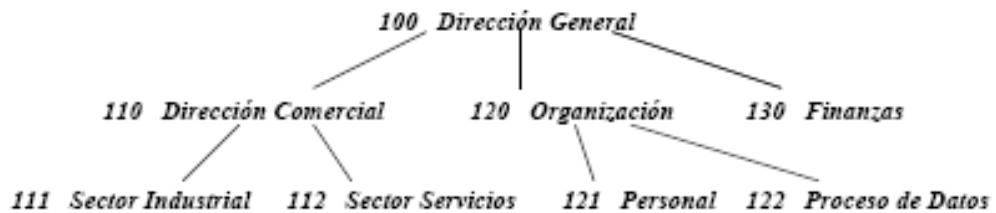
B. Tabla de Departamento. TDEPTO

NUMDE	Integer	Número de departamento
NUMCE	Integer	Número de Centro de trabajo donde está el Departamento. Todo departamento está en un Centro de trabajo. En un Centro de trabajo pueden haber varios departamentos.
DIREC	Integer	Número de empleado que es director del Departamento. Todo departamento tiene un único director. El director puede ser en propiedad o en funciones. Un empleado puede ser director en propiedad de un departamento como máximo, pero puede ser director en funciones de varios.
TIDIR	Char	Tipo director: P o F
PRESU	Decimal	Presupuesto del departamento (en millones)
DEPDE	Integer	Departamento del que depende. Todo departamento depende de otro, excepto el 1º que no depende de ninguno. Un departamento puede tener varios dependiendo de él. El director de un departamento es el Jefe de todos los empleados de su departamento, y de los departamentos dependientes del suyo. Admite el valor nulo.
NOMDE	Varchar	Nombre del departamento

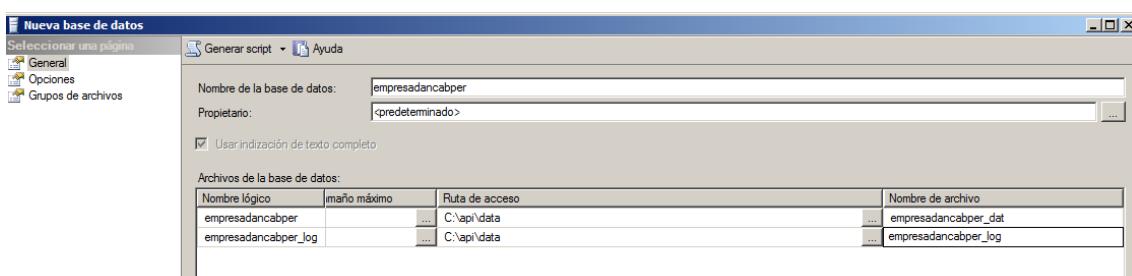
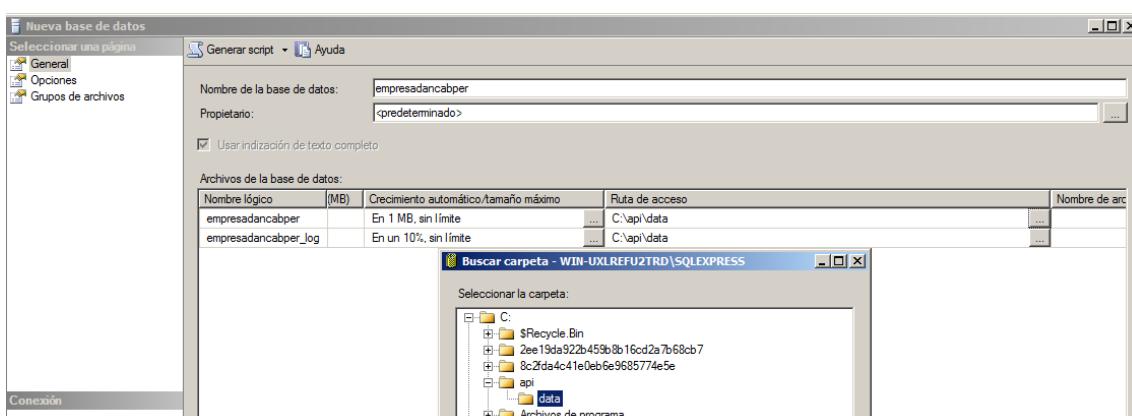
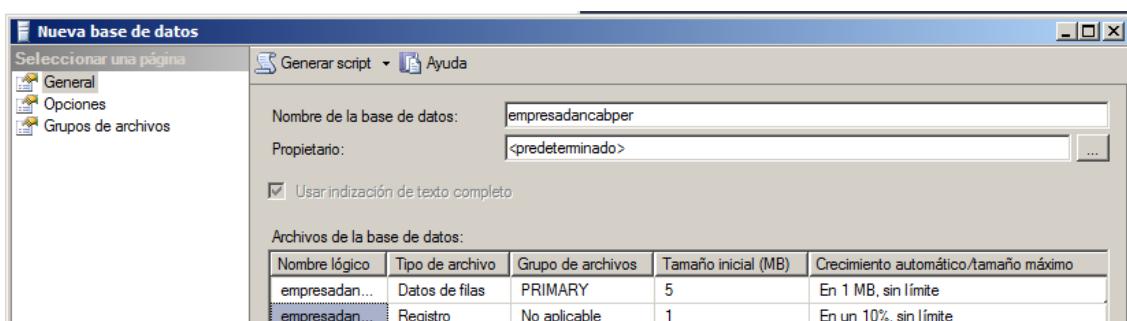
C. Tabla de Centros de Trabajo. TCENTR

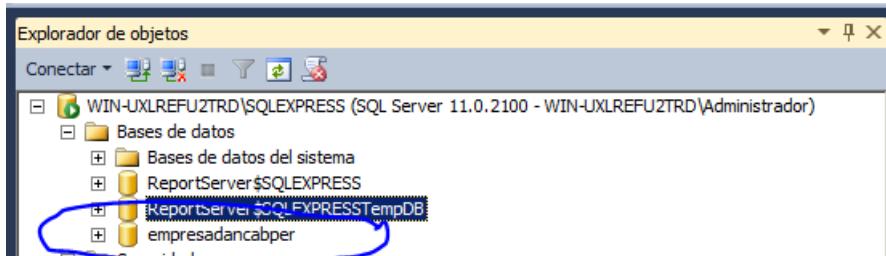
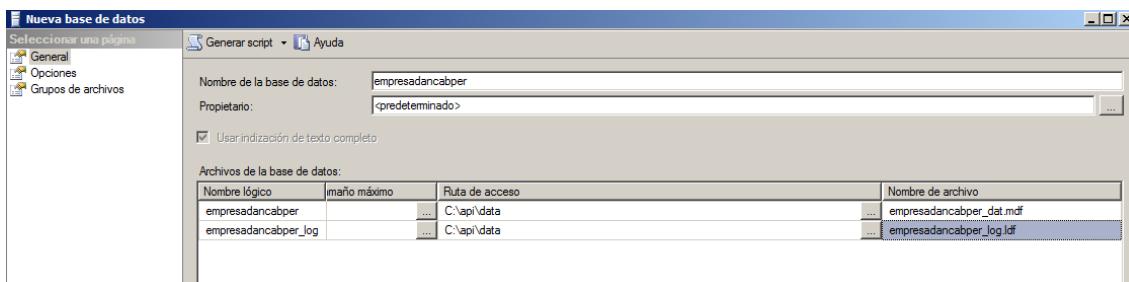
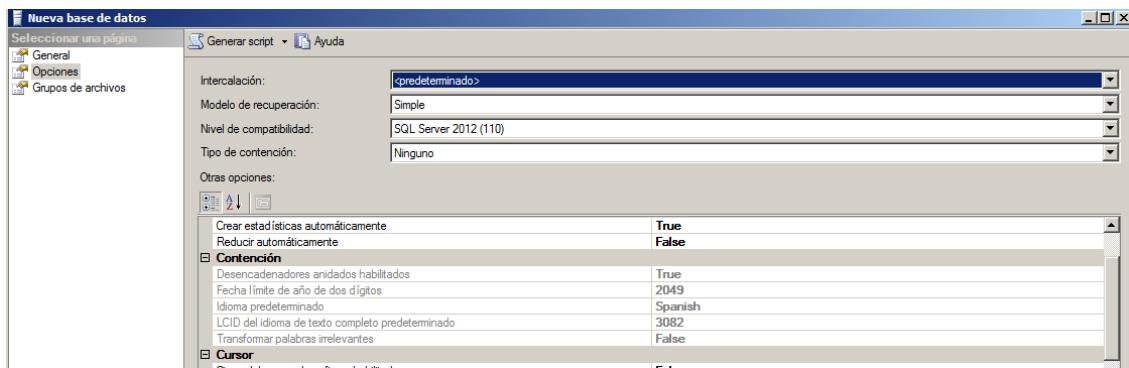
NUMCE	Integer	Número de centro de trabajo
NOMCE	Varchar	Nombre del centro de trabajo
SEÑAS	Varchar	Dirección

Estructura departamentos:



1. Crear la BD siguiente llamándola **empresa** más las iniciales del alumno (juan carrasco ramos → empresasajucara).
 - a) Ponerle los parámetros necesarios para configurar el fichero de datos y transacciones según se explicó en clase.





- b) Además crear las tablas correspondientes con las restricciones que creas necesarias. Este ejercicio hay que realizarlo en SQL Server con instrucciones escritas como una consulta en el management studio. Realizar capturas de pantalla de la instrucción y su resultado después de ejecutarla.

SQLQuery1.sql - WIN-UXLREFU2RD\Administrador (56)*

```

USE empresadancabper --este comando se usa para indicar la db que queremos usar

CREATE TABLE TEMPLE (
    NUMEMP integer, --Número de empleado
    NUMDE integer, --Número de departamento
    EXTEL smallint, --Extensión telefónica del empleado
    FECNA Date, --Fecha de nacimiento
    FECIN Date, --Fecha de ingreso
    SALAR Decimal, --Salario(en miles de pesetas)
    COMIS Decimal, --Comisión, es nula para todos los empleados
        --Excepto para Vendedores, director de vendedores, y director comercial.
    NUMHI smallint, --Número de hijos
    NOMEM Varchar, --Nombre del empleado
)

```

Explorador de objetos

Conectar ▾

SQLQuery2.sql - WIN-...\\Administrador (54)*

```

USE empresadancabper
CREATE TABLE TDEPTO(
NUMDE integer, --Número de departamento
NUMCE integer, /*
Número de centro de trabajo donde está el departamento.
Todo departamento está en un centro de trabajo. En un centro
de trabajo puede haber varios departamentos.
*/
DIREC integer/*
Número de empleado que es director del departamento. Todo
departamento tiene un único director. El director puede ser
en propiedad o en funciones.Un empleado puede ser director en propiedad
de un departamento como máximo, pero puede ser director en funciones de varios.
*/
TIDIR char, --Tipo de director: P o F
PRESU decimal, --Presupuesto del departamento (en millones)
DEPDE integer/*
Departamento del que depende. Todo departamento depende de otro,
excepto el 1º que no depende de ninguno. Un departamento puede tener varios
dependiendo de él. El director de un departamento es el jefe de todos los
empleados de su departamento, y de los departamentos dependientes del suyo.
Admite el valor nulo.
*/
NOMDE varchar, --Nombre del departamento
);

```

Explorador de objetos

Conectar ▾

SQLQuery3.sql - WIN-...\\Administrador (51)*

```

USE empresadancabper
CREATE TABLE TCENTR (
NUMCE integer, --Número del centro de trabajo
NOMCE varchar, --Nombre del centro de trabajo
SEÑAS varchar, --Dirección
);

```

c) Insertar los registros en las tablas.

- EN TODOS ESTOS EJERCICIOS MOSTRAR LA INSTRUCCIÓN SI ES ESCRITA EN UNA CONSULTA Y EL RESULTADO DE LA EJECUCIÓN.

Explorador de objetos

Conectar ▾

SQLQuery4.sql - WIN-...\\Administrador (52)*

```

USE empresadancabper
CREATE TABLE HIJOS(
NUMHI smallint, --Número de hijos
NOMEM Varchar, --Nombre del empleado
);

```

Mensajes

Comandos completados correctamente.

100 %

Consulta ejecutada correctamente.

The screenshot shows the SQL Server Management Studio interface. On the left, the Object Explorer displays a database structure with several objects under the root node. In the center, there are two results panes. The top pane shows a T-SQL script for creating a department table:

```

departamento tiene un único director en propiedad o en funciones. Un empleado de un departamento como máximo, pero
*/
TDIR char, --Tipo de director: P o
PRESU decimal, --Presupuesto del de
DEPDE integer;/*
Departamento del que depende. Todo
excepto el 1º que no depende de ninguno
dependiendo de él. El director de los
empleados de su departamento, y de
Admite el valor nulo.
*/
NOMDE varchar, --Nombre del departamento
);

```

The bottom pane shows the execution results with a green checkmark and the message "Consulta ejecutada correctamente." (Query executed successfully).

- SI ES EN ENTORNO GRÁFICO MOSTRAR CAPTURA DE PANTALLA DE LA INSTRUCCIÓN Y EL RESULTADO DE LA EJECUCIÓN.
- EN TODOS LOS EJERCICIOS HAY QUE CONTROLAR EL ORDEN DE CREACIÓN DE LA TABLAS
 - Esto es muy importante si queremos que no se produzcan errores

- Hay que tener en cuenta que numde de la tabla temple está relacionado con el campo numde de la tabla tdepto

6

The screenshot shows the SSMS interface with the following details:

- Object Explorer:** Shows the database structure under "WIN-UXLREFU2RD\SQLEXPRESS". The "Bases de datos" node is expanded, showing "empresadancabper" which contains "Tablas" (Tables). Inside "Tablas", "dbo.TCENTR", "dbo.TDEPTO", and "dbo.TEMPLE" are listed.
- SQL Query Window:** The title bar says "SQLQuery6.sql - WIN-...\\Administrador (51)*". The query entered is:


```
use empresadancabper
ALTER TABLE TDEPTO
ADD PRIMARY KEY (NUMDE)
```
- Mensajes Window:** Shows the message "Comandos completados correctamente." (Commands completed successfully).
- Status Bar:** Shows "100 %".
- Message Bar:** Shows a green checkmark icon and the message "Consulta ejecutada correctamente." (Query executed successfully).

Explorador de objetos

Conectar ▾

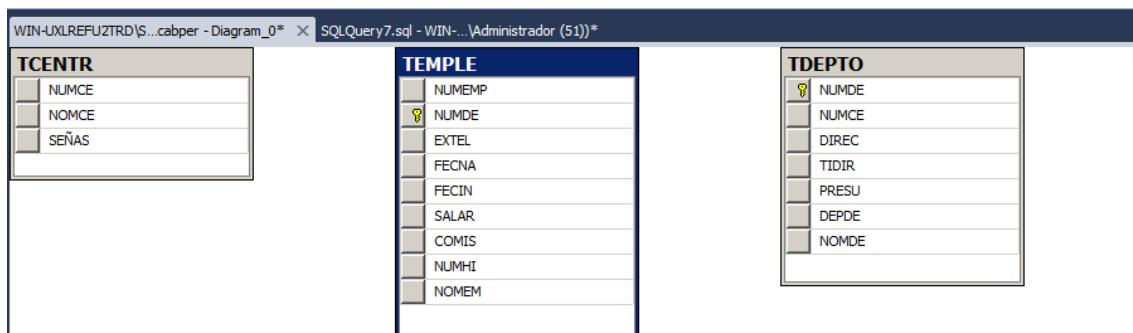
use empresadancabper

ALTER TABLE TEMPLE
ADD PRIMARY KEY (NUMDE)

100 % Mensajes

Comandos completados correctamente.

100 % Consulta ejecutada correctamente.



Explorador de objetos

Conectar ▾ Ejecutar Depurar

SQLQuery8.sql - WIN-UXLREFU2RD\Administrador (51)*

```
use empresadancabper
ALTER TABLE TEMPLE
ADD CONSTRAINT FK_TEMPLE_NUMDE
FOREIGN KEY (NUMDE) REFERENCES TDEPTO (NUMDE)
```

100 % Mensajes Comandos completados correctamente.

100 % Consulta ejecutada correctamente.

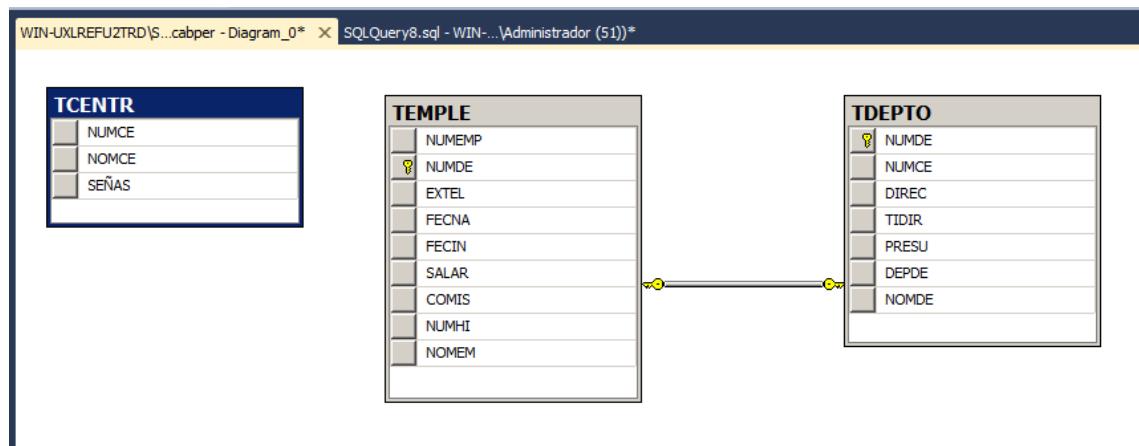


Tabla TEMPLE (Empleados) (los nulos se representan con -).

9

	1	2	3	4	5	6	7	8	9
columna 1:									
1: NUMEM	110	121	350	10.11.29	15.02.50	310	-	3	PONS, CESAR
2: NUMDE	120	112	840	09.06.35	01.10.68	350	110	1	LASA, MARIO
3: EXTEL	130	112	810	09.09.45	01.02.69	290	110	2	TEROL, LUCIANO
4: FECNA (dd.mm.aa)	150	121	340	10.08.30	15.01.48	440	-	0	PEREZ, JULIO
5: FECIN (dd.mm.aa)	160	111	740	09.07.39	11.11.68	310	110	2	AGUIRRE, AUREO
6: SALAR	180	110	508	18.10.34	18.03.56	480	50	2	PEREZ, MARCOS
7: COMIS	190	121	350	12.05.32	11.02.62	300	-	4	VEIGA, JULIANA
8: NUMHI	210	100	200	28.09.40	22.01.59	380	-	2	GALVEZ, PILAR
9: NOMEM	240	111	760	26.02.42	24.02.66	280	100	3	SANZ, LAVINIA
	250	100	250	27.10.46	01.03.67	450	-	0	ALBA, ADRIANA
	260	100	220	03.12.43	12.07.68	720	-	6	LOPEZ, ANTONIO
	270	112	800	21.05.45	10.09.66	380	80	3	GARCIA, OCTAVIO
	280	130	410	11.01.48	08.10.71	290	-	5	FLOR, DOROTEA
	285	122	620	25.10.49	15.02.68	380	-	0	POLO, OTILIA
	290	120	910	30.11.47	14.02.68	270	-	3	GIL, GLORIA
	310	130	480	21.11.46	15.01.71	420	-	0	GARCIA, AUGUSTO
	320	122	620	25.12.57	05.02.78	405	-	2	SANZ, CORNELIO
	330	112	850	19.08.48	01.03.72	280	90	0	DIEZ, AMELIA
	350	122	610	13.04.49	10.09.84	450	-	1	CAMPS, AURELIO
	360	111	750	29.10.58	10.10.68	250	100	2	LARA, DORINDA
	370	121	360	22.06.67	20.01.87	190	-	1	RUIZ, FABIOLA
	380	112	880	30.03.68	01.01.88	180	-	0	MARTIN, MICAELA
	390	110	500	19.02.66	08.10.86	215	-	1	MORAN, CARMEN
	400	111	780	18.08.69	01.11.87	185	-	0	LARA, LUCRECIA
	410	122	660	14.07.68	13.10.88	175	-	0	MUNOZ, AZUCENA
	420	130	450	22.10.66	19.11.88	400	-	0	FIERRO, CLAUDIA
	430	122	650	26.20.67	19.11.88	210	-	1	MORA, VALERIANA
	440	111	760	26.09.66	28.02.86	210	100	0	DURAN, LIVIA
	450	112	880	21.10.66	28.02.86	210	100	0	PEREZ, SABINA
	480	111	760	04.04.65	28.02.86	210	100	1	PINO, DIANA
	490	112	880	06.06.64	01.01.88	180	100	0	TORRES, HORACIO
	500	111	750	08.10.65	01.01.87	200	100	0	VAZQUEZ, HONORIA
	510	110	550	04.05.66	01.11.86	200	-	1	CAMPOS, ROMULO
	550	111	780	10.01.70	21.01.88	100	120	0	SANTOS, SANCHO

Tabla TDEPTO (Departamentos) (los valores Nulos se representan con un guion, -)

NUMDE	NUMCE	DIREC	TIDIR	PRESU	DEPDE	NOMDE
100	10	260	P	12	-	DIRECCION GENERAL
110	20	180	P	15	100	DIRECC. COMERCIAL
111	20	180	F	11	110	SECTOR INDUSTRIAL
112	20	270	P	9	110	SECTOR SERVICIOS
120	10	150	F	3	100	ORGANIZACION
121	10	150	P	2	120	PERSONAL
122	10	350	P	6	120	PROCESO DE DATOS
130	10	310	P	2	100	FINANZAS

Tabla TCENTR (Centros)

NUMCE	NOMCE	SEÑAS
10	SEDE CENTRAL	C. ALCALA, 820, MADRID
20	RELACION CON CLIENTES	C. ATOCHA, 405, MADRID

2. Crear la BD y tablas siguientes poniéndole por nombre **Instituto** más las iniciales del alumno. Asimismo crear las tablas de la misma con las constraints que se especifican. Realizar el ejercicio desde el entorno gráfico de SQL Server y capturar pantallas del proceso.

1

The screenshot shows the Microsoft SQL Server Management Studio interface. In the center-right pane, there is a query editor window titled "SQLQuery1.sql - DESKTOP-1GBQTDM\SQLEXPRESS.master (DESKTOP-1GBQTDM\Dani (54))* - Microsoft SQL Server Management Studio". The query in the editor is:

```
CREATE DATABASE InstitutoDanCabPer
go
```

To the left of the editor is the "Explorador de objetos" (Object Explorer) pane. It shows the structure of the "DESKTOP-1GBQTDM\SQLEXPRESS (SQL Server 11.0.2218 - DESKTOP-1GBQTDM\Dani)" database. Under the "Bases de datos" node, a new database named "InstitutoDanCabPer" is visible, along with other system databases like "master", "model", "msdb", and "tempdb". Other nodes in the tree include "Seguridad", "Objetos de servidor", "Replicación", and "Administración".

DDL Práctica BASES DE DATOS SQL (Definición)

Se desea tener una base de datos con la siguiente información acerca de los alumnos de una academia de idiomas donde se imparten varios cursos:

- Información acerca de los **alumnos**, que constará de su nombre y apellidos, un código único para cada alumno, su dirección, fecha de nacimiento y sexo ('H' o 'M').
- Estos alumnos estarán matriculados en un sólo **curso** cada uno. Cada curso tendrá un nombre, un código único que lo identifica, el número máximo de alumnos recomendado, un profesor, la fecha de inicio y de finalización del curso y el número de horas del curso.
- Los **profesores** tendrán una ficha en la academia donde se especifican sus datos personales y los datos del contrato. Es decir, nombre, DNI, dirección, titulación, cuota por hora.

Se pide:

- a) Generar las siguientes tablas para guardar esta información

TABLA ALUMNOS

Column Name
NOMBRE
APELLIDO1
APELLIDO2
DNI
DIRECCION

TABLA CURSOS

Column Name
NOMBRE_CURSO
COD_CURSO
DNI_PROFESOR
MAXIMO_ALUMNOS
FECHA_INICIO

TABLA PROFESORES

Column Name
NOMBRE
APELLIDO1
APELLIDO2
DNI
DIRECCION

SEXO	FECHA_NACIMIENTO	FECHA_FIN	NUM_HORAS	TITULO	1
				GANA	
CURSO					

Explorador de objetos

```

Conectar ▾       
DESKTOP-1GBQTDM\SQLEXPRESS (SQL)
  + Bases de datos
    + Bases de datos del sistema
    + dbreserva
    + InstitutoDanCabPer
      + Diagramas de base de datos
      + Tablas
        + Tablas del sistema
        + FileTables
          + dbo.ALUMNOS
        + Vistas
        + Sinónimos
        + Programación
        + Service Broker
        + Almacenamiento
        + Seguridad

```

SQLQuery1.sql - D...GBQTDM\Dani (54)*

```

CREATE DATABASE InstitutoDanCabPer
go

use InstitutoDanCabPer

CREATE TABLE ALUMNOS (
  idalumnos int not null primary key,
  nombre varchar(20),
  apellido1 varchar(30),
  apellido2 varchar(30),
  dni text,
  direccion varchar(80),
  sexo varchar(6),
  fecha_nacimiento date,
  curso varchar(80)
)
go

```

Explorador de objetos

```

Conectar ▾       
DESKTOP-1GBQTDM\SQLEXPRESS (SQL)
  + Bases de datos
    + Bases de datos del sistema
    + dbreserva
    + InstitutoDanCabPer
      + Diagramas de base de datos
      + Tablas
        + Tablas del sistema
        + FileTables
          + dbo.ALUMNOS
          + dbo.CURSOS

```

SQLQuery1.sql - D...GBQTDM\Dani (54)*

```

CREATE TABLE CURSOS (
  idcurso int not null primary key,
  nombre_curso varchar (80),
  cod_curso int,
  dni_profesor text,
  maximo_alumnos int,
  fecha_inicio date,
  fecha_fin date,
  num_horas int
)
go

```

Explorador de objetos

```

Conectar ▾       
DESKTOP-1GBQTDM\SQLEXPRESS (SQL)
  + Bases de datos
    + Bases de datos del sistema
    + dbreserva
    + InstitutoDanCabPer
      + Diagramas de base de datos
      + Tablas
        + Tablas del sistema
        + FileTables
        + dbo.ALUMNOS
        + dbo.CURSOS
        + dbo.PROFESORES

```

SQLQuery1.sql - D...GBQTDM\Dani (55)

```

CREATE TABLE PROFESORES (
  idprofesores int not null primary key,
  nombre varchar(20),
  apellido1 varchar(30),
  apellido2 varchar(30),
  dni text,
  direccion varchar(80),
  titulo varchar(30),
  gana money
)
go

```

- No es posible dar de alta un alumno si no se matricula en un curso.
-- No es posible dar de alta un alumno si no se matricula en un curso

```
ALTER TABLE ALUMNOS ALTER COLUMN curso
varchar(80) not null
go
```
- La información del número de horas del curso es imprescindible para almacenarlo.
-- La información del número de horas del curso es imprescindible para almacenarlo

```
ALTER TABLE CURSOS ALTER COLUMN num_horas
int not null
go
```
- El campo GANA de la tabla PROFESORES no puede estar en ningún caso vacío.
-- El campo GANA de la tabla PROFESORES no puede estar en ningún caso vacío

```
ALTER TABLE PROFESORES ALTER COLUMN gana
money not null
go
```
- Dos cursos no pueden llamarse igual. Lo mismo le pasa a los profesores.

```
ALTER TABLE PROFESORES
ADD CONSTRAINT UQ_NOMBRE
unique (nombre)
go

ALTER TABLE CURSOS
ADD CONSTRAINT UQ_CURSO
unique (nombre_curso)
go
```

- Podemos identificar las tuplas de las tablas CURSOS mediante el atributo CODIGO y PROFESORES y ALUMNOS usando el DNI.

```
ALTER TABLE CURSOS
DROP COLUMN idcurso
go

ALTER TABLE CURSOS ALTER COLUMN cod_curso
int not null
go

ALTER TABLE CURSOS
ADD CONSTRAINT PK_COD_CURSO
PRIMARY KEY (cod_curso)
go
```

```
ALTER TABLE PROFESORES
DROP COLUMN idprofesores
go

ALTER TABLE ALUMNOS
DROP COLUMN idalumnos
go

ALTER TABLE PROFESORES ALTER COLUMN dni
varchar(9) not null
go

ALTER TABLE ALUMNOS ALTER COLUMN dni
varchar(9) not null
go

ALTER TABLE PROFESORES
ADD CONSTRAINT PK_DNI_PROFESORES
PRIMARY KEY (dni)
go

ALTER TABLE ALUMNOS
ADD CONSTRAINT PK_DNI_ALUMNOS
PRIMARY KEY (dni)
go
```

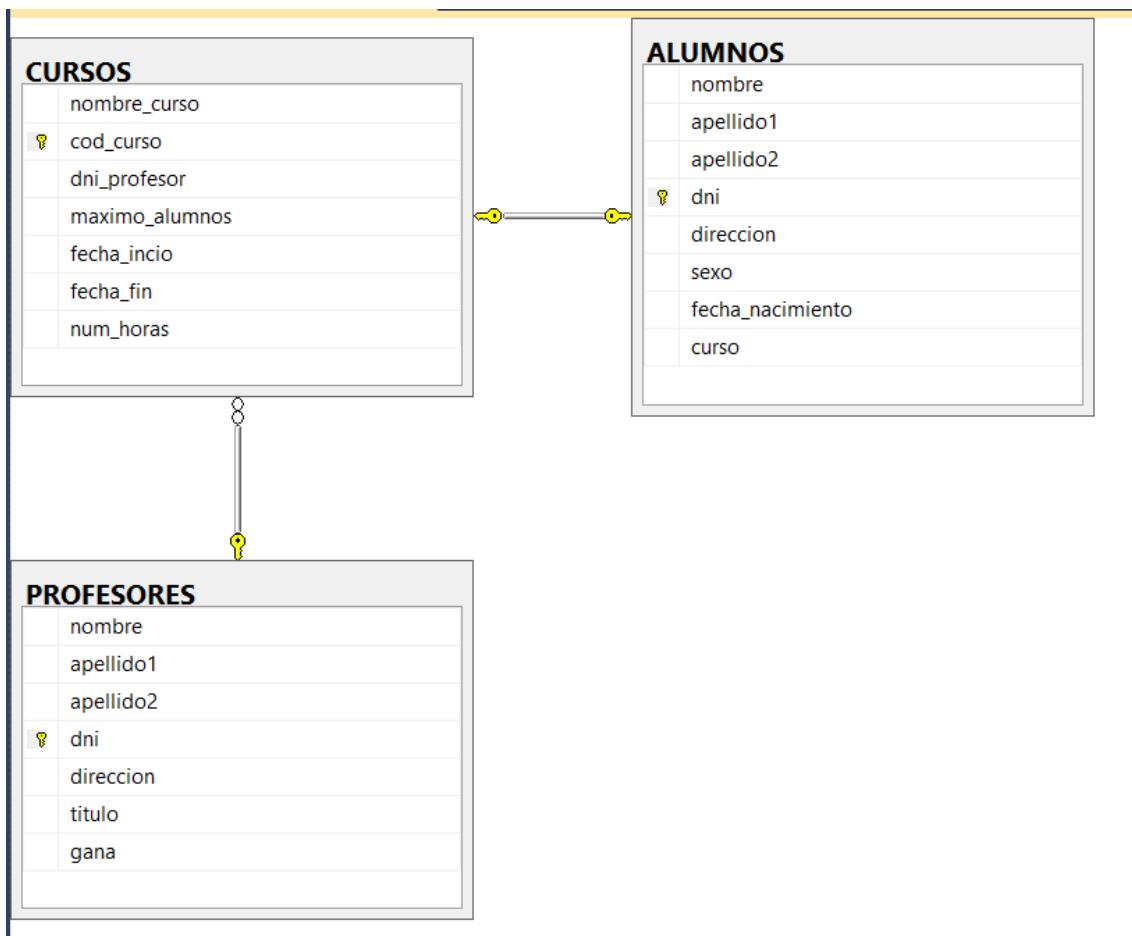
- Los valores para el atributo sexo son sólo M y H (en mayúsculas).

```
-- Los valores para el atributo sexo son sólo M y H (en mayúsculas)
ALTER TABLE ALUMNOS
ADD CONSTRAINT CHK_SEXO
CHECK (sexo in ('H', 'M'))
go
```

- Se ha de mantener la regla de integridad de referencia.

```
-- Se ha de mantener la regla de integridad de referencia.
ALTER TABLE CURSOS
ADD CONSTRAINT FK_CURSOS_PROFESORES_DNI
FOREIGN KEY (dni_profesor) REFERENCES PROFESORES(dni)
go

ALTER TABLE CURSOS
ADD CONSTRAINT FK_CURSOS_ALUMNOS_NOMCURSO
FOREIGN KEY (nombre_curso) REFERENCES ALUMNOS(curso)
go
```



b) Insertar las siguientes tuplas:

Tabla PROFESORES

NOMBRE	APELLIDO1	APELLIDO2	DNI	DIRECCION	TITULO	GANA
Juan	Arch	López	32432455	Puerta Negra, 4	Ing. Informática	7500
Maria	Oliva	Rubio	43215643	Juan Alfonso 32	Lda. Fil. Inglesa	5400

```
insert into PROFESORES(nombre,apellido1,apellido2,dni,direccion,titulo,gana)
values ('Juan','Arch','Lopez','32432455','Puerta Negra,4','Ing. Informatica','7500'),
       ('Maria','Oliva','Rubio','43215643','Juan Alfonso 32','Lda. Fil. Inglesa','5400')
go
```

Tabla CURSOS

NOMBRE_CURSO	COD_CURSO	DNI_PROF	MAX_ALU	FECHA_INI	FECHA_FIN	NUM_HORAS
Inglés Básico	1	43215643	15	01-11-00	22-12-00	120
Administración Linux	2	32432455	18	01-09-00	22-12-00	80

```
insert into CURSOS(nombre_curso,cod_curso,dni_profesor,maximo_alumnos,fecha_inicio,fecha_fin,num_horas)
values ('Inglés Basico','1','43215643','15','01-11-00','22-12-00','120'),
       ('Administración Linux','2','32432455','18','01-09-00','22-12-00','80')
go
```

Tabla ALUMNOS

NOMBRE	APELLIDO1	APELLIDO2	DNI	DIRECCION	SEXO	FECHA_N	CURSO
Lucas	Manilva	López	123523	Alhamar 3	H	01-11-1979	1
Antonia	López	Alcantara	2567567	Maniquí 21	M	23-11-1976	2
Manuel	Alcantara	Pedrós	3123689	Julian 2	H	01-12-1984	3
José	Pérez	Caballar	4896765	Jarcha 5	H	3-02-1977	4

En esta tupla tenemos que modificar el sexo ya que solo permite valores H o M, en curso tendremos que modificar también su valor ya que está ligado al campo nom_curso de la tabla CURSOS y este nos pide un nombre de uso y no un número.

```
insert into ALUMNOS(nombre,apellido1,apellido2,dni,direccion,sexo,fecha_nacimiento,curso)
values ('Lucas','Manilva','López','123523','Alhamar 3','H','01-11-1979','1'),
       ('Antonia','López','Alcantara','2567567','Maniquí 21','M','23-11-1976','2'),
       ('Manuel','Alcantara','Pedrós','3123689','Julian 2','H','01-12-1984','3'),
       ('José','Pérez','Caballar','4896765','Jarcha 5','H','3-02-1977','4')
go
```

c) Insertar la siguiente tupla en ALUMNOS:

NOMBRE	APELLIDO1	APELLIDO2	DNI	DIRECC	SEXO	FECHA_NAC	CURSO
Sergio	Navas	Retal	123523		P		

No podemos incluir esta tupla ya que existen algunas restricciones como en el sexo solo se puede ingresar H o M como valor y no se puede dar de alta un alumno sin matricularlo en un curso, también nos encontramos con que el dni que es la primary key esta repetido en una tupla anterior. Modificaremos sus datos e ingresaremos la tupla.

```
insert into ALUMNOS(nombre,apellido1,apellido2,dni,direccion,sexo,fecha_nacimiento,curso)
values ('Sergio','Navas','Retal','42213469','Amistad 12','H','01-11-1989','Filosofia')
go
```

d) Añadir el campo edad de tipo numérico a la tabla PROFESORES.

```

ALTER TABLE PROFESORES ADD edad int
go

```

1

e) Añadir las siguientes restricciones:

- La edad de los profesores está entre 18 y 65 años.

```

ALTER TABLE PROFESORES
ADD CONSTRAINT CHK_EDADMAXIN
CHECK (edad>=18 and edad<=65)
go

```

- No se puede añadir un curso si su número de alumnos máximo es menor que 10.

- El número de horas de los cursos debe ser mayor que 100.

Al modificar este parámetro nos da error, tenemos que modificar el número de horas que asignamos anteriormente (80) por 100 o mayor que 100

```

-- El número de horas de los cursos debe ser mayor que 100
ALTER TABLE CURSOS
ADD CONSTRAINT CHK_HORAS_CURSOS
CHECK (num_horas>=100)
go

```

f) Eliminar la restricción que controla los valores permitidos para el atributo sexo.

```

go

ALTER TABLE ALUMNOS
DROP CHK_SEXO
go

```

g) Se dice que cada alumno ha de estar matriculado en un solo curso. ¿Esto quiere decir que el atributo CURSO de la tabla ALUMNOS ha de ser UNIQUE? Pruebe a introducir la restricción y ver si confirma esta hipótesis.

h) Eliminar la restricción de tipo NOT NULL del atributo GANA.

1

```
ALTER TABLE PROFESORES ALTER COLUMN gana  
money null  
go
```

i) Insertar restricción no nula en el campo FECHA_INICO de CURSOS.

```
ALTER TABLE CURSOS ALTER COLUMN fecha_inicio  
date not null  
go
```

j) Cambiar la clave primaria de Profesor al nombre y apellidos.

The screenshot shows the Object Explorer on the left with the following tree structure:

- + dbo.CURSOS
- + dbo.PROFESORES
 - Columns
 - nombre (PK, varchar(20), Not Null)
 - apellido1 (PK, varchar(30), Not Null)
 - apellido2 (PK, varchar(30), Not Null)
 - dni (varchar(9), Not Null)
 - direccion (varchar(80), NULL)
 - titulo (varchar(30), NULL)
 - gana (money, NULL)
 - edad (int, NULL)
 - Claves
 - PK_NOMBRE_APELLIDOS
 - Restricciones
 - Desencadenadores
 - Índices
 - Estadísticas
 - Vistas
 - Sinónimos
 - Programación

On the right, the script pane displays the following T-SQL code:

```
ALTER TABLE PROFESORES  
DROP PK_DNI_PROFESORES  
go  
  
ALTER TABLE PROFESORES ALTER COLUMN nombre  
varchar(20) not null  
go  
  
ALTER TABLE PROFESORES ALTER COLUMN apellido1  
varchar(30) not null  
go  
  
ALTER TABLE PROFESORES ALTER COLUMN apellido2  
varchar(30) not null  
go  
  
ALTER TABLE PROFESORES  
ADD CONSTRAINT PK_NOMBRE_APELLIDOS  
PRIMARY KEY (nombre,apellido1,apellido2)  
go
```

k) Insertar las siguientes tupla en alumnos:

NOMBRE	APELLIDO1	APELLIDO2	DNI	DIRECCION	TITULO	GANAS
Juan	Arch	López	32432455	Puerta Negra, 4	Ing. Informática	NULL

Esta tupla no se puede insertar en alumnos ya que corresponde a la tabla profesores, a parte ya tendríamos la tupla repetida y lo único que habría que hacer sería dejar el campo gana en blanco, pero no podemos porque hemos puesto una restricción para que no se pueda dejar en blanco.

1

NOMBRE	APELLIDO1	APELLIDO2	DNI	DIRECC	SEXO	FECHA_NAC	CURSO
María	Jaén	Sevilla	789678	Martos 5	M	10-MAR-1977	1

```
insert into ALUMNOS(nombre,apellido1,apellido2,dni,direccion,sexo,fecha_nacimiento,curso)
values ('María','Jaén','Sevilla','789678','Martos 5','M','10-MAR-1977','1')
go
```

- I) La fecha de nacimiento de Antonia López está equivocada. La verdadera es 23 de diciembre de 1976.

```
Update ALUMNOS Set fecha_nacimiento='23-12-1976' Where fecha_nacimiento='23-11-1976'
go
```

- m) Cambiar a Antonia López al curso de código 5.

```
Update ALUMNOS Set curso='5' Where nombre='Antonia'
go
```

	nombre	apellido1	apellido2	dni	direccion	sexo	fecha_naci...	curso
▶	Lucas	Manilva	López	123523	Alhamar 3	H	1979-11-01	1
	Antonia	López	Alcantara	2567567	Maniquí 21	M	1976-12-23	5
	Manuel	Alcantara	Pedrós	3123689	Julian 2	H	1984-12-01	3
	Sergio	Navas	Retal	42213469	Amistad 12	H	1989-11-01	4
	José	Pérez	Caballar	4896765	Jarcha 5	H	1977-02-03	2
	María	Jaén	Sevilla	789678	Martos 5	M	1977-03-10	1
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

- n) Eliminar la profesora Laura Jiménez

No existe la profesora Laura Jiménez

- o) Crear una tabla de uso temporal llamada NOMBRE DE ALUMNOS que tenga un solo atributo (NOMBRE_COMPLETO) de tipo cadena de caracteres y con el contenido

de la tabla alumnos en esos campos. Observe que no hay restricción de clave primaria para esta tabla.

The screenshot shows the Object Explorer on the left with the database 'InstitutoDanCabPer' selected. Under the 'Tables' node, the table 'dbo.ALUMNOS' is highlighted. On the right, the 'Script' tab displays the T-SQL code for creating the table and inserting data into it. Below the code, the 'Results' tab shows the data inserted into the table.

```
CREATE TABLE nombre_de_alumnos (
    nombre_completo varchar (60) not null
)
go
insert into nombre_de_alumnos(nombre_completo)
values ('Lucas Manilva López'),
       ('Antonia López Alcantara'),
       ('Manuel Alcantara Pedrós'),
       ('José Pérez Caballar'),
       ('Sergio Navas Retal'),
       ('María Jaén Sevilla')
go
```

	nombre_completo
▶	Manilva López
	Antonia López Alcantara
	Manuel Alcantara Pedrós
	José Pérez Caballar
	Sergio Navas Retal
	María Jaén Sevilla
*	NULL

p) Borrar las tablas

The screenshot shows the Object Explorer on the left with the database 'InstitutoDanCabPer' selected. Under the 'Tables' node, all tables are listed: 'ALUMNOS', 'CURSOS', 'PROFESORES', and 'nombre_de_alumnos'. On the right, the 'Script' tab displays the T-SQL code for dropping these tables. The 'Results' tab shows the execution of the drop statements.

```
\ Sergio Navas Retal ',
('María Jaén Sevilla'))
go

DROP TABLE ALUMNOS
go
DROP TABLE CURSOS
go
DROP TABLE PROFESORES
go
DROP TABLE nombre_de_alumnos
go
```

3. Se dispone de la tabla **PERSONAS** con los datos de aquellos empleados que trabajan en una serie de hospitales. La Base de Datos la llamaremos Hospital + nombre alumno.

The screenshot shows the phpMyAdmin interface. In the top navigation bar, the server is set to "127.0.0.1". Below it, there are tabs for "Bases de datos", "SQL", "Estado actual", and "Cuenta". The "Bases de datos" tab is selected. On the left, a tree view lists databases: "Nueva", "hospitaldancabper" (highlighted with a blue box), "information_schema", "mysql", "performance_schema", and "phpmyadmin". In the main panel, a SQL query window titled "Ejecute la o las consultas SQL en el servidor '127.0.0.1':" contains the command: "CREATE DATABASE HOSPITALDanCabPer".

TABLA PERSONAS

The screenshot shows a table viewer for the "personas" table. The columns are: cod_hospital, dni, apellidos, funcion, salario, and localidad. The data is as follows:

	cod_hospital	dni	apellidos	funcion	salario	localidad
1	12345678	García Hernández, Eladio	CONSERJE	1200	LORCA	
1	87654321	Fuentes Bermejo, Carlos	DIRECTOR	2000	MURCIA	
2	55544433	González Marín, Alicia	CONSERJE	1200	MURCIA	
1	66655544	Castillo Montes, Pedro	MEDICO	1700	MURCIA	
2	22233322	Tristán García, Ana	MEDICO	1900	MURCIA	
3	55544411	Ruiz Hernández, Caridad	MEDICO	1900	LORCA	
3	99988333	Serrano Díaz, Alejandro	DIRECTOR	2400	CARTAGENA	
4	3322211	Mesa del Castillo, Juan	MEDICO	2200	LORCA	
2	22233333	Martínez Molina, Andrés	MEDICO	1600	CARTAGENA	
4	55544412	Jiménez Jiménez, Dolores	CONSERJE	1200	MURCIA	
4	22233311	Martínez Molina, Gloria	MEDICO	1600	MURCIA	

Ejecuta el siguiente script para cargar los datos (cambia el nombre de la base de datos y la tabla). Realizar esta práctica con el entorno gráfico de mysql (phpmyadmin).

```
create table personas(
    cod_hospital integer,
    dni           integer primary key,
    nombre        varchar(30),
    apellidos    varchar(50),
    funcion       varchar(30),
    salario       integer,
    loc           varchar(20));

insert into personas values(1,12345678,'García Hernández',
'Eladio','CONSERJE',1200,'LORCA');
insert into personas values(1,87654321,'Fuentes Bermejo',
'Carlos','DIRECTOR',2000,'MURCIA');
insert into personas values(2,55544433,'González Marín',
'Alicia','CONSERJE',1200,'MURCIA');
```

```

insert into personas values(1,66655544,'Castillo Montes',
'Pedro','MEDICO',1700,'MURCIA');
insert into personas values(2,22233322,'Tristán García',
'Ana','MEDICO',1900,'MURCIA');
insert into personas values(3,55544411,'Ruiz Hernández',
'Caridad','MEDICO',1900,'LORCA');
insert into personas values(3,99988333,'Serrano Díaz',
'Alejandro','DIRECTOR',2400,'CARTAGENA');
insert into personas values(4,33222111,'Mesa del Castillo',
'Juan','MEDICO',2200,'LORCA');
insert into personas values(2,22233333,'Martínez Molina',
'Andrés','MEDICO',1600,'CARTAGENA');
insert into personas values(4,55544412,'Jiménez Jiménez',
'Dolores','CONSERJE',1200,'MURCIA');
insert into personas values(4,22233311,'Martínez
Molina','Gloria','MEDICO',1600,'MURCIA');

```

2

The screenshot shows the MySQL Workbench interface. On the left, the database tree displays the schema 'hospitaldancabper' containing the 'personas' table. The central pane shows the SQL editor with the creation script and the inserted data. The results pane shows the query 'SELECT * FROM `personas`' with 11 rows returned.

```

create table personas(
cod_hospital integer,
dni integer primary key,
nombre varchar(30),
apellidos varchar(50),
funcion varchar(30),
salario integer,
loc varchar(20));
insert into personas values(1,12345678,'García Hernández', 'Eladio','CONSERJE',1200,'LORCA');
insert into personas values(1,87654321,'Fuentes Bermejo', 'Carlos','DIRECTOR',2000,'MURCIA');
insert into personas values(2,55544433,'González Marín', 'Alicia','CONSERJE',1200,'MURCIA');
insert into personas values(1,66655544,'Castillo Montes', 'Pedro','MEDICO',1700,'MURCIA');
insert into personas values(2,22233322,'Tristán García', 'Ana','MEDICO',1900,'MURCIA');

```

	cod_hospital	dni	nombre	apellidos	funcion	salario	loc
<input type="checkbox"/>	1	12345678	García Hernández	Eladio	CONSERJE	1200	LORCA
<input type="checkbox"/>	4	22233311	Martínez Molina	Gloria	MEDICO	1600	MURCIA
<input type="checkbox"/>	2	22233322	Tristán García	Ana	MEDICO	1900	MURCIA
<input type="checkbox"/>	2	22233333	Martínez Molina	Andrés	MEDICO	1600	CARTAGENA
<input type="checkbox"/>	4	33222111	Mesa del Castillo	Juan	MEDICO	2200	LORCA
<input type="checkbox"/>	3	55544411	Ruiz Hernández	Caridad	MEDICO	1900	LORCA
<input type="checkbox"/>	4	55544412	Jiménez Jiménez	Dolores	CONSERJE	1200	MURCIA
<input type="checkbox"/>	2	55544433	González Marín	Alicia	CONSERJE	1200	MURCIA
<input type="checkbox"/>	1	66655544	Castillo Montes	Pedro	MEDICO	1700	MURCIA
<input type="checkbox"/>	2	87654321	Fuentes Bermejo	Carlos	DIRECTOR	2000	MURCIA
<input type="checkbox"/>	3	99988333	Serrano Díaz	Alejandro	DIRECTOR	2400	CARTAGENA

a) Mostrar todos los datos de todas las personas.

Ejecutar la(s) consulta(s) SQL en la tabla hospitaldancabper.personas:

```
1 | SELECT * FROM `personas`|
```

	cod_hospital	dni	nombre	apellidos	funcion	salario	loc
<input type="checkbox"/>	1	12345678	García Hernández	Eladio	CONSERJE	1200	LORCA
<input type="checkbox"/>	4	22233311	Martínez Molina	Gloria	MEDICO	1600	MURCIA
<input type="checkbox"/>	2	22233322	Tristán García	Ana	MEDICO	1900	MURCIA
<input type="checkbox"/>	2	22233333	Martínez Molina	Andrés	MEDICO	1600	CARTAGENA
<input type="checkbox"/>	4	33222111	Mesa del Castillo	Juan	MEDICO	2200	LORCA
<input type="checkbox"/>	3	55544411	Ruiz Hernández	Caridad	MEDICO	1900	LORCA
<input type="checkbox"/>	4	55544412	Jiménez Jiménez	Dolores	CONSERJE	1200	MURCIA
<input type="checkbox"/>	2	55544433	González Marín	Alicia	CONSERJE	1200	MURCIA
<input type="checkbox"/>	1	66655544	Castillo Montes	Pedro	MEDICO	1700	MURCIA
<input type="checkbox"/>	1	87654321	Fuentes Bermejo	Carlos	DIRECTOR	2000	MURCIA
<input type="checkbox"/>	3	99988333	Serrano Díaz	Alejandro	DIRECTOR	2400	CARTAGENA

+ Opciones Seleccionar todo Para los elementos que están marcados: Editar Copiar Borrar Exportar

b) Obtén el DNI, apellidos y función de todas las personas.

Ejecutar la(s) consulta(s) SQL en la tabla hospitaldancabper.pe

```
1 | SELECT dni,apellidos,funcion| FROM `personas`|
```

	dni	apellidos	funcion
<input type="checkbox"/>	12345678	Eladio	CONSERJE
<input type="checkbox"/>	22233311	Gloria	MEDICO
<input type="checkbox"/>	22233322	Ana	MEDICO
<input type="checkbox"/>	22233333	Andrés	MEDICO
<input type="checkbox"/>	33222111	Juan	MEDICO
<input type="checkbox"/>	55544411	Caridad	MEDICO
<input type="checkbox"/>	55544412	Dolores	CONSERJE
<input type="checkbox"/>	55544433	Alicia	CONSERJE
<input type="checkbox"/>	66655544	Pedro	MEDICO
<input type="checkbox"/>	87654321	Carlos	DIRECTOR
<input type="checkbox"/>	99988333	Alejandro	DIRECTOR

Ordenar según la clave: Ninguna

+ Opciones Seleccionar todo Para los elementos que están marcados: Editar Copiar Borrar

c) Mostrar los apellidos de las personas que vivan en LORCA.

The screenshot shows the MySQL Workbench interface. On the left, the database tree displays 'Nueva', 'hospitaldancabper', 'information_schema', 'mysql', 'performance_schema', 'phpmyadmin', and 'test'. The main window contains a SQL editor with the following query:

```
1 | SELECT apellidos FROM `personas` WHERE loc ='LORCA'
```

The screenshot shows the results of the query execution. The status bar at the top indicates: "Mostrando filas 0 - 2 (total de 3, La consulta tardó 0.0005 segundos.)". The results table shows three rows of data:

apellidos
Eladio
Juan
Caridad

d) Mostrar los apellidos de las personas que vivan en MURCIA o LORCA

The screenshot shows the MySQL Workbench interface. On the left, the database tree displays 'Nueva', 'hospitaldancabper', 'information_schema', 'mysql', 'performance_schema', 'phpmyadmin', and 'test'. The main window contains a SQL editor with the following query:

```
1 | SELECT apellidos FROM `personas` WHERE loc = 'LORCA' OR loc = 'MURCIA'
```

The results table shows ten rows of data:

apellidos
Eladio
Gloria
Ana
Juan
Caridad
Dolores
Alicia
Pedro
Carlos

- e) Seleccionar los datos de aquellas personas que vivan en MURCIA y tengan un salario superior a los 1500 euros.

2

Ejecutar la(s) consulta(s) SQL en la tabla hospitaldancabper.personas:

```
1 SELECT * FROM `personas` WHERE loc = 'MURCIA' AND salario >=1500;
```

	cod_hospital	dni	nombre	apellidos	funcion	salario	loc
<input type="checkbox"/>	4	22233311	Martínez Molina	Gloria	MEDICO	1600	MURCIA
<input type="checkbox"/>	2	22233322	Tristán García	Ana	MEDICO	1900	MURCIA
<input type="checkbox"/>	1	66655544	Castillo Montes	Pedro	MEDICO	1700	MURCIA
<input type="checkbox"/>	1	87654321	Fuentes Bermejo	Carlos	DIRECTOR	2000	MURCIA

- f) Mostrar los datos de las personas que vivan en MURCIA, tengan un salario superior a los 1500 euros y sean DIRECTORES.

Ejecutar la(s) consulta(s) SQL en la tabla hospitaldancabper.personas:

```
1 SELECT * FROM `personas` WHERE loc = 'MURCIA' AND salario >=1500 AND funcion = 'DIRECTOR';
```

	cod_hospital	dni	nombre	apellidos	funcion	salario	loc
<input type="checkbox"/>	1	87654321	Fuentes Bermejo	Carlos	DIRECTOR	2000	MURCIA

- g) Mostrar los datos de las personas cuya función sea MÉDICO ordenados por apellidos descendenteamente.

Ejecutar la(s) consulta(s) SQL en la tabla hospitaldancabper.personas:

```
1 SELECT * FROM `personas` WHERE funcion = 'MEDICO' ORDER BY apellidos DESC;
```

The screenshot shows the MySQL Workbench interface with the 'coa_hospital' table selected. The table has columns: cod_hospital, dni, nombre, apellidos, funcion, salario, and loc. The data includes entries for MEDICO roles in various locations like MURCIA, LORCA, and CARTAGENA.

		cod_hospital	dni	nombre	apellidos	funcion	salario	loc
<input type="checkbox"/>		1	66655544	Castillo Montes	Pedro	MEDICO	1700	MURCIA
<input type="checkbox"/>		4	33222111	Mesa del Castillo	Juan	MEDICO	2200	LORCA
<input type="checkbox"/>		4	22233311	Martínez Molina	Gloria	MEDICO	1600	MURCIA
<input type="checkbox"/>		3	55544411	Ruiz Hernández	Caridad	MEDICO	1900	LORCA
<input type="checkbox"/>		2	22233333	Martínez Molina	Andrés	MEDICO	1600	CARTAGENA
<input type="checkbox"/>		2	22233322	Tristán García	Ana	MEDICO	1900	MURCIA
<input type="checkbox"/>								

h) Mostrar los datos de todas las localidades que hay en la tabla personas sin repeticiones (debes emplear la cláusula DISTINCT)

The screenshot shows the MySQL Workbench interface with the SQL editor containing the query: 'SELECT DISTINCT loc FROM `personas`'. The results pane displays the unique values for the 'loc' column: LORCA, MURCIA, and CARTAGENA.

The screenshot shows the MySQL Workbench interface with the SQL editor containing the query: 'SELECT DISTINCT loc FROM `personas`'. The results pane displays the unique values for the 'loc' column: LORCA, MURCIA, and CARTAGENA. Below the results, there are filtering options: 'Mostrar todo', 'Número de filas: 25', 'Filtrar filas:', and 'Ordenar según la clave: Ninguna'.

i) Mostrar los datos de las personas que tengan un salario superior a 1500 euros y sean médicos. Ordenar la salida por salario descendenteamente.

The screenshot shows the MySQL Workbench interface with the SQL editor containing the query: 'SELECT * FROM `personas` WHERE salario >=1500 AND funcion = 'MEDICO' ORDER BY salario DESC'. The results pane displays the data for MEDICO roles with salaries above 1500 euros, ordered by salary in descending order. The results include entries for LORCA, MURCIA, and CARTAGENA.

j) Seleccionar aquellas personas cuyo apellido comience por M.

The screenshot shows the MySQL Workbench interface. On the left is a tree view of databases: Nueva, hospitaldancabper, information_schema, mysql, performance_schema, phpmyadmin, and test. The main area contains a SQL editor with the following code:

```

1 SELECT * FROM `personas` WHERE nombre LIKE 'M%'
2 /* Al haber un fallo en la creación de la tabla (se pusieron nombres en los apellidos y viceversa) he creado la consulta
3 * con el where nombre
4 */

```

Below the SQL editor is a results grid for the 'personas' table. The columns are: cod_hospital, dni, nombre, apellidos, funcion, salario, and loc. The data shows three rows:

cod_hospital	dni	nombre	apellidos	funcion	salario	loc
4	22233311	Martínez Molina	Gloria	MEDICO	1600	MURCIA
2	22233333	Martínez Molina	Andrés	MEDICO	1600	CARTAGENA
4	33222111	Mesa del Castillo	Juan	MEDICO	2200	LORCA

k) Mostrar los datos de las personas que tengan una M en el apellido y cuya función sea CONSERJE

The screenshot shows the MySQL Workbench interface. On the left is a tree view of databases: Nueva, hospitaldancabper, information_schema, mysql, performance_schema, phpmyadmin, and test. The main area contains a SQL editor with the following code:

```

SELECT * FROM `personas` WHERE nombre LIKE '%M%' AND funcion = 'CONSERJE';

```

Below the SQL editor is a results grid for the 'personas' table. The columns are: cod_hospital, dni, nombre, apellidos, funcion, salario, and loc. The data shows two rows:

cod_hospital	dni	nombre	apellidos	funcion	salario	loc
4	55544412	Jiménez Jiménez	Dolores	CONSERJE	1200	MURCIA
2	55544433	González Marín	Alicia	CONSERJE	1200	MURCIA

l) Mostrar aquellas personas que tengan un salario entre 1500 y 2000 euros.

The screenshot shows the MySQL Workbench interface. On the left is a tree view of databases: Nueva, hospitaldancabper, information_schema, mysql, performance_schema, phpmyadmin, and test. The main area contains a SQL editor with the following code:

```

SELECT * FROM `personas` WHERE salario BETWEEN 1500 AND 2000

```

Below the SQL editor is a results grid for the 'personas' table. The columns are: cod_hospital, dni, nombre, apellidos, funcion, salario, and loc. The data shows six rows:

cod_hospital	dni	nombre	apellidos	funcion	salario	loc
4	22233311	Martínez Molina	Gloria	MEDICO	1600	MURCIA
2	22233322	Tristán García	Ana	MEDICO	1900	MURCIA
2	22233333	Martínez Molina	Andrés	MEDICO	1600	CARTAGENA
3	55544411	Ruiz Hernández	Caridad	MEDICO	1900	LORCA
1	66655544	Castillo Montes	Pedro	MEDICO	1700	MURCIA
1	87654321	Fuentes Bermejo	Carlos	DIRECTOR	2000	MURCIA

m) Seleccionar los datos de aquellas personas cuya función sea MÉDICO o DIRECTOR (utilizar el operador IN)

2

SELECT * FROM `personas` WHERE funcion IN('MEDICO','DIRECTOR')								
<input type="checkbox"/> Mostrar todo Número de filas: <select>25</select> Filtrar filas: <input type="text"/> Buscar en esta tabla								
Ordenar según la clave: <input type="text"/> Ninguna								
+ Opciones								
← T →	cod_hospital	dni	nombre	apellidos	funcion	salario	loc	
<input type="checkbox"/>	4	22233311	Martínez Molina	Gloria	MEDICO	1600	MURCIA	
<input type="checkbox"/>	2	22233322	Tristán García	Ana	MEDICO	1900	MURCIA	
<input type="checkbox"/>	2	22233333	Martínez Molina	Andrés	MEDICO	1600	CARTAGENA	
<input type="checkbox"/>	4	33222111	Mesa del Castillo	Juan	MEDICO	2200	LORCA	
<input type="checkbox"/>	3	55544411	Ruiz Hernández	Caridad	MEDICO	1900	LORCA	
<input type="checkbox"/>	1	66655544	Castillo Montes	Pedro	MEDICO	1700	MURCIA	
<input type="checkbox"/>	1	87654321	Fuentes Bermejo	Carlos	DIRECTOR	2000	MURCIA	
<input type="checkbox"/>	3	99988333	Serrano Díaz	Alejandro	DIRECTOR	2400	CARTAGENA	

n) Obtener los datos de aquellas personas cuya función no sea CONSERJE (utilizar el operador NOT IN) y tengan un salario superior a los 1500 euros, ordenados por apellido descendenteamente.

SELECT * FROM `personas` WHERE funcion NOT IN ('CONSERJE') AND salario >1500 ORDER BY nombre DESC								
<input type="checkbox"/> Mostrar todo Número de filas: <select>25</select> Filtrar filas: <input type="text"/> Buscar en esta tabla								
Ordenar según la clave: <input type="text"/> Ninguna								
+ Opciones								
← T →	cod_hospital	dni	nombre	apellidos	funcion	salario	loc	
<input type="checkbox"/>	2	22233322	Tristán García	Ana	MEDICO	1900	MURCIA	
<input type="checkbox"/>	3	99988333	Serrano Díaz	Alejandro	DIRECTOR	2400	CARTAGENA	
<input type="checkbox"/>	3	55544411	Ruiz Hernández	Caridad	MEDICO	1900	LORCA	
<input type="checkbox"/>	4	33222111	Mesa del Castillo	Juan	MEDICO	2200	LORCA	
<input type="checkbox"/>	4	22233311	Martínez Molina	Gloria	MEDICO	1600	MURCIA	
<input type="checkbox"/>	2	22233333	Martínez Molina	Andrés	MEDICO	1600	CARTAGENA	
<input type="checkbox"/>	1	87654321	Fuentes Bermejo	Carlos	DIRECTOR	2000	MURCIA	
<input type="checkbox"/>	1	66655544	Castillo Montes	Pedro	MEDICO	1700	MURCIA	

- o) Mostrar los datos de las personas que sean de MURCIA o CARTAGENA y que pertenezcan al hospital número 1.

2

```
SELECT * FROM `personas` WHERE cod_hospital = 1 AND loc = 'MURCIA' OR 'CARTAGENA'
```

	cod_hospital	dni	nombre	apellidos	funcion	salario	loc
<input type="checkbox"/>	1	66655544	Castillo Montes	Pedro	MEDICO	1700	MURCIA
<input type="checkbox"/>	1	87654321	Fuentes Bermejo	Carlos	DIRECTOR	2000	MURCIA

- p) Obtén los apellidos en mayúsculas de las personas que trabajen en el hospital número 1.

```
SELECT UPPER (nombre) FROM `personas` WHERE cod_hospital = 1
```

	cod_hospital	nombre	apellidos	funcion	salario	loc
<input type="checkbox"/>	1	GARCÍA HERNÁNDEZ				
<input type="checkbox"/>	1	CASTILLO MONTES				
<input type="checkbox"/>	1	FUENTES BERMEJO				

- q) Con una consulta devuelve los apellidos de todas las personas. Al lado debe aparecer la longitud de cada apellido.

```
SELECT nombre,length(nombre) FROM `personas`
```

nombre	length(nombre)
García Hernández	16
Martínez Molina	15
Tristán García	14
Martínez Molina	15
Mesa del Castillo	17
Ruiz Hernández	14
Jiménez Jiménez	15
González Marín	14
Castillo Montes	15
Fuentes Bermejo	15
Serrano Díaz	12

r) Obtener los apellidos y localidad en minúscula de todas aquellas personas que no trabajen en el hospital número 1.

The screenshot shows the phpMyAdmin interface. On the left, the database structure is visible with the 'personas' table selected. In the main area, a query is entered:

```
SELECT LOWER (nombre), LOWER (loc) FROM `personas` WHERE cod_hospital >= 2
```

Below the query, there are options to 'Mostrar todo' (Show all), set the number of rows to 25, and a search bar. A dropdown menu for sorting by key is set to 'Ninguna'. The results table shows the following data:

LOWER (nombre)	LOWER (loc)
martínez molina	murcia
tristán garcía	murcia
martínez molina	cartagena
mesa del castillo	lorca
ruiz hernández	lorca
jiménez jiménez	murcia
gonzález marín	murcia
serrano diáz	cartagena

s) Obtener los datos de las personas que trabajen en los hospitales 1 ó 2 y tengan un salario superior a 1500 euros.

The screenshot shows the phpMyAdmin interface. The 'personas' table is selected. A query is entered:

```
SELECT * FROM `personas` WHERE salario > 1500 AND cod_hospital = 2 OR cod_hospital = 1
```

The results table displays the following data:

cod_hospital	dni	nombre	apellidos	funcion	salario	loc
1	12345678	García Hernández	Eladio	CONSERJE	1200	LORCA
2	22233322	Tristán García	Ana	MEDICO	1900	MURCIA
2	22233333	Martínez Molina	Andrés	MEDICO	1600	CARTAGENA
1	66655544	Castillo Montes	Pedro	MEDICO	1700	MURCIA
1	87654321	Fuentes Bermejo	Carlos	DIRECTOR	2000	MURCIA

t) Visualizar los datos de aquellas personas que no trabajen en el hospital número 2 y que sean de MURCIA.

The screenshot shows the phpMyAdmin interface. The 'personas' table is selected. A query is entered:

```
SELECT * FROM `personas` WHERE loc = 'MURCIA' AND cod_hospital NOT IN (2)
```

The results table displays the following data:

cod_hospital	dni	nombre	apellidos	funcion	salario	loc
4	22233311	Martínez Molina	Gloria	MEDICO	1600	MURCIA
4	55544412	Jiménez Jiménez	Dolores	CONSERJE	1200	MURCIA
1	66655544	Castillo Montes	Pedro	MEDICO	1700	MURCIA
1	87654321	Fuentes Bermejo	Carlos	DIRECTOR	2000	MURCIA

4. Crear la base de datos **EmpresaPrueba** + nombre alumno. Se disponen de la siguientes tablas con los siguientes datos.

3

TABLA EMPLE

	emp_no	apellido	oficio	dir	fecha_alt	salario	comision	dept_no
►	7369	SÁNCHEZ	EMPLEADO	7902	17/12/1980	104000		20
	7499	ARROYO	VENDEDOR	7698	20/02/1980	208000	39000	30
	7521	SALA	VENDEDOR	7698	22/02/1981	162500	162500	30
	7566	JIMÉNEZ	DIRECTOR	7839	02/04/1981	386750		20
	7654	MARTÍN	VENDEDOR	7698	29/09/1981	162500	182000	30
	7698	NEGRO	DIRECTOR	7839	01/05/1981	370500		30
	7788	GIL	ANALISTA	7566	09/11/1981	390000		20
	7839	REY	PRESIDENTE		17/11/1981	650000		10
	7844	TOVAR	VENDEDOR	7698	08/09/1981	195000	0	30
	7876	ALONSO	EMPLEADO	7788	23/09/1981	143000		20
	7900	JIMENO	EMPLEADO	7698	03/12/1981	1235000		30
	7902	FERNÁNDEZ	ANALISTA	7566	03/12/1981	390000		20
	7934	MUÑOZ	EMPLEADO	7782	23/01/1982	169000		10

TABLA DEPART.

	dept_no	dnombre	loc
►	10	CONTABILIDAD	SEVILLA
+	20	INVESTIGACIÓN	MADRID
+	30	VENTAS	BARCELONA
+	40	PRODUCCIÓN	BILBAO

Ejecuta el siguiente script para cargar los datos (cambia el nombre de la base de datos y las tablas). Realizar este ejercicio desde la línea de comandos de mysql.

```

CREATE TABLE emple(
    emp_no INTEGER PRIMARY KEY,
    apellido VARCHAR(50) NOT NULL,
    oficio VARCHAR(30),
    dir INTEGER,
    fecha_alt DATE,
    salario INTEGER,
);

```

```

drop database empresaprueba;
//Poner la instrucción para que borre la base de datos y tablas sólo
//si existen, así no da error
create database empresaprueba;

create table emple(emp_no      INTEGER PRIMARY KEY,
                  apellido VARCHAR(50) NOT NULL,
                  oficio VARCHAR(30),

```

```

dir INTEGER,
fecha_alt DATE,
salario INTEGER,
comision INTEGER,
dept_no INTEGER);

create table depart(
dept_no INTEGER,
dnombre VARCHAR(30),
loc VARCHAR(30));

INSERT INTO emple VALUES (7369,'SÁNCHEZ','EMPLEADO',7902,'1990/12/17',
1040,NULL,20);
INSERT INTO emple VALUES (7499,'ARROYO','VENDEDOR',7698,'1990/02/20',
1500,390,30);
INSERT INTO emple VALUES (7521,'SALA','VENDEDOR',7698,'1991/02/22',
1625,650,30);
INSERT INTO emple VALUES (7566,'JIMÉNEZ','DIRECTOR',7839,'1991/04/02',
2900,NULL,20);
INSERT INTO emple VALUES (7654,'MARTÍN','VENDEDOR',7698,'1991/09/29',
1600,1020,30);
INSERT INTO emple VALUES (7698,'NEGRO','DIRECTOR',7839,'1991/05/01',
3005,NULL,30);
INSERT INTO emple VALUES (7782,'CEREZO','DIRECTOR',7839,'1991/06/09',
2885,NULL,10);
INSERT INTO emple VALUES (7788,'GIL','ANALISTA',7566,'1991/11/09',
3000,NULL,20);
INSERT INTO emple VALUES (7839,'REY','PRESIDENTE',NULL,'1991/11/17',
4100,NULL,10);
INSERT INTO emple VALUES (7844,'TOVAR','VENDEDOR',7698,'1991/09/08',
1350,0,30);
INSERT INTO emple VALUES (7876,'ALONSO','EMPLEADO',7788,'1991/09/23',
1430,NULL,20);
INSERT INTO emple VALUES (7900,'JIMENO','EMPLEADO',7698,'1991/12/03',
1335,NULL,30);
INSERT INTO emple VALUES
(7902,'FERNÁNDEZ','ANALISTA',7566,'1991/12/03',
3000,NULL,20);
INSERT INTO emple VALUES (7934,'MUÑOZ','EMPLEADO',7782,'1992/01/23',
1690,NULL,10);

INSERT INTO depart VALUES (10,'CONTABILIDAD','SEVILLA');
INSERT INTO depart VALUES (20,'INVESTIGACIÓN','MADRID');
INSERT INTO depart VALUES (30,'VENTAS','BARCELONA');
INSERT INTO depart VALUES (40,'PRODUCCIÓN','BILBAO');

```

- El campo salario hay que multiplicarlo por 100 porque así es como está almacenado en la tabla.

Realizar las siguientes consultas:

3

- a) Mostrar el apellido, oficio y número de departamento de cada empleado.

The screenshot shows the MySQL Workbench interface. On the left, the database schema is displayed with several databases listed: Nueva, empresapruebadancabper, hospitaldancabper, personas, information_schema, mysql, performance_schema, phpmyadmin, and test. The 'personas' table under the 'Nueva' database is selected. On the right, the results of a SQL query are shown. The query is:

```
Select apellido, oficio, dept_no from emple
```

The results are presented in a table with columns: apellido, oficio, and dept_no. The data is as follows:

apellido	oficio	dept_no
SÁNCHEZ	EMPLEADO	20
ARROYO	VENDEDOR	30
SALA	VENDEDOR	30
JIMÉNEZ	DIRECTOR	20
MARTÍN	VENDEDOR	30
NEGRO	DIRECTOR	30
CEREZO	DIRECTOR	10
GIL	ANALISTA	20
REY	PRESIDENTE	10
TOVAR	VENDEDOR	30
ALONSO	EMPLEADO	20
JIMENO	EMPLEADO	30
FERNÁNDEZ	ANALISTA	20
MUÑOZ	EMPLEADO	10

- b) Mostrar el número, nombre y localización de cada departamento.

The screenshot shows the MySQL Workbench interface. On the left, the database schema is displayed with several databases listed: `empresapruebadancabper`.`depart` and `hospitaldancabper`.`personas`. The 'personas' table under the 'Nueva' database is selected. On the right, the results of a SQL query are shown. The query is:

```
Select dept_no, dnombre, loc from depart
```

The results are presented in a table with columns: dept_no, dnombre, and loc. The data is as follows:

dept_no	dnombre	loc
10	CONTABILIDAD	SEVILLA
20	INVESTIGACIÓN	MADRID
30	VENTAS	BARCELONA
40	PRODUCCIÓN	BILBAO

c) Mostrar todos los datos de todos los empleados.

3

The screenshot shows the phpMyAdmin interface with the database 'empresapruebadancabper' selected. A query 'Select * from emple' is run, displaying 25 rows of employee data. The columns are emp_no, apellido, oficio, dir, fecha_alt, salario, comision, and dept_no. The data includes various employees like SÁNCHEZ, ARROYO, SALA, JIMÉNEZ, MARTÍN, NEGRO, CEREZO, GIL, REY, TOVAR, ALONSO, JIMENO, and MUÑOZ, with salaries ranging from 1040 to 4100.

	emp_no	apellido	oficio	dir	fecha_alt	salario	comision	dept_no
<input type="checkbox"/>	7369	SÁNCHEZ	EMPLEADO	7902	1990-12-17	1040	NULL	20
<input type="checkbox"/>	7499	ARROYO	VENDEDOR	7698	1990-02-20	1500	390	30
<input type="checkbox"/>	7521	SALA	VENDEDOR	7698	1991-02-22	1625	650	30
<input type="checkbox"/>	7566	JIMÉNEZ	DIRECTOR	7839	1991-04-02	2900	NULL	20
<input type="checkbox"/>	7654	MARTÍN	VENDEDOR	7698	1991-09-29	1600	1020	30
<input type="checkbox"/>	7698	NEGRO	DIRECTOR	7839	1991-05-01	3005	NULL	30
<input type="checkbox"/>	7782	CEREZO	DIRECTOR	7839	1991-06-09	2885	NULL	10
<input type="checkbox"/>	7788	GIL	ANALISTA	7566	1991-11-09	3000	NULL	20
<input type="checkbox"/>	7839	REY	PRESIDENTE	NULL	1991-11-17	4100	NULL	10
<input type="checkbox"/>	7844	TOVAR	VENDEDOR	7698	1991-09-08	1350	0	30
<input type="checkbox"/>	7876	ALONSO	EMPLEADO	7788	1991-09-23	1430	NULL	20
<input type="checkbox"/>	7900	JIMENO	EMPLEADO	7698	1991-12-03	1335	NULL	30
<input type="checkbox"/>	7902	FERNÁNDEZ	ANALISTA	7566	1991-12-03	3000	NULL	20
<input type="checkbox"/>	7934	MUÑOZ	EMPLEADO	7782	1992-01-23	1690	NULL	10

d) Datos de los empleados ordenados por apellidos.

The screenshot shows the phpMyAdmin interface with the database 'empresapruebadancabper' selected. A query 'Select * from emple order by apellido' is run, displaying 25 rows of employee data. The data is sorted by apellido, with entries starting from MUÑOZ and ending with SÁNCHEZ. The columns are emp_no, apellido, oficio, dir, fecha_alt, salario, comision, and dept_no.

	emp_no	apellido	oficio	dir	fecha_alt	salario	comision	dept_no
<input type="checkbox"/>	7876	ALONSO	EMPLEADO	7788	1991-09-23	1430	NULL	20
<input type="checkbox"/>	7499	ARROYO	VENDEDOR	7698	1990-02-20	1500	390	30
<input type="checkbox"/>	7782	CEREZO	DIRECTOR	7839	1991-06-09	2885	NULL	10
<input type="checkbox"/>	7902	FERNÁNDEZ	ANALISTA	7566	1991-12-03	3000	NULL	20
<input type="checkbox"/>	7788	GIL	ANALISTA	7566	1991-11-09	3000	NULL	20
<input type="checkbox"/>	7566	JIMÉNEZ	DIRECTOR	7839	1991-04-02	2900	NULL	20
<input type="checkbox"/>	7900	JIMENO	EMPLEADO	7698	1991-12-03	1335	NULL	30
<input type="checkbox"/>	7654	MARTÍN	VENDEDOR	7698	1991-09-29	1600	1020	30
<input type="checkbox"/>	7934	MUÑOZ	EMPLEADO	7782	1992-01-23	1690	NULL	10
<input type="checkbox"/>	7698	NEGRO	DIRECTOR	7839	1991-05-01	3005	NULL	30
<input type="checkbox"/>	7839	REY	PRESIDENTE	NULL	1991-11-17	4100	NULL	10
<input type="checkbox"/>	7521	SALA	VENDEDOR	7698	1991-02-22	1625	650	30
<input type="checkbox"/>	7369	SÁNCHEZ	EMPLEADO	7902	1990-12-17	1040	NULL	20
<input type="checkbox"/>	7844	TOVAR	VENDEDOR	7698	1991-09-08	1350	0	30

- e) Datos de los empleados ordenados por número de departamento descendente.

3

The screenshot shows the MySQL Workbench interface. On the left is a tree view of databases: 'empresapruебadancabper', 'hospitaldancabper', 'Nueva', 'personas', 'information_schema', 'mysql', 'performance_schema', 'phpmyadmin', and 'test'. The main area displays a table of employee data. The SQL query at the top is:

```
Select * from emple order by dept_no DESC
```

Below the query are buttons for 'Mostrar todo' (Show all), 'Número de filas:' (Number of rows: 25), and 'Filtrar filas:' (Filter rows: Buscar en esta tabla). A dropdown menu 'Ordenar según la clave:' (Sort by key: Ninguna) is shown. The table has columns: emp_no, apellido, oficio, dir, fecha_alt, salario, comision, dept_no. The data is sorted by dept_no in descending order, with 30 at the top and 20 at the bottom. Each row includes edit, copy, and delete options.

emp_no	apellido	oficio	dir	fecha_alt	salario	comision	dept_no
7499	ARROYO	VENDEDOR	7698	1990-02-20	1500	390	30
7521	SALA	VENDEDOR	7698	1991-02-22	1625	650	30
7654	MARTÍN	VENDEDOR	7698	1991-09-29	1600	1020	30
7698	NEGRO	DIRECTOR	7839	1991-05-01	3005	NULL	30
7900	JIMENO	EMPLEADO	7698	1991-12-03	1335	NULL	30
7844	TOVAR	VENDEDOR	7698	1991-09-08	1350	0	30
7369	SÁNCHEZ	EMPLEADO	7902	1990-12-17	1040	NULL	20
7902	FERNÁNDEZ	ANALISTA	7566	1991-12-03	3000	NULL	20
7876	ALONSO	EMPLEADO	7788	1991-09-23	1430	NULL	20
7788	GIL	ANALISTA	7566	1991-11-09	3000	NULL	20

- f) Datos de los empleados ordenados por número de departamento descendente y dentro de cada departamento ordenados por apellido ascendente.

The screenshot shows the MySQL Workbench interface. The database tree is identical to the previous screenshot. The main area displays a table of employee data. The SQL query at the top is:

```
Select * from emple order by dept_no desc, apellido
```

Below the query are buttons for 'Mostrar todo' (Show all), 'Número de filas:' (Number of rows: 25), and 'Filtrar filas:' (Filter rows: Buscar en esta tabla). A dropdown menu 'Ordenar según la clave:' (Sort by key: Ninguna) is shown. The table has columns: emp_no, apellido, oficio, dir, fecha_alt, salario, comision, dept_no. The data is sorted by dept_no in descending order (30 at top, 20 at bottom) and then by apellido in ascending order (ARROYO at top of dept 30, GIL at bottom of dept 20). Each row includes edit, copy, and delete options.

emp_no	apellido	oficio	dir	fecha_alt	salario	comision	dept_no
7499	ARROYO	VENDEDOR	7698	1990-02-20	1500	390	30
7900	JIMENO	EMPLEADO	7698	1991-12-03	1335	NULL	30
7654	MARTÍN	VENDEDOR	7698	1991-09-29	1600	1020	30
7698	NEGRO	DIRECTOR	7839	1991-05-01	3005	NULL	30
7521	SALA	VENDEDOR	7698	1991-02-22	1625	650	30
7844	TOVAR	VENDEDOR	7698	1991-09-08	1350	0	30
7876	ALONSO	EMPLEADO	7788	1991-09-23	1430	NULL	20
7902	FERNÁNDEZ	ANALISTA	7566	1991-12-03	3000	NULL	20
7788	GIL	ANALISTA	7566	1991-11-09	3000	NULL	20

g) Mostrar los datos de los empleados cuyo salario sea mayor que 2000000.

3

The screenshot shows the MySQL Workbench interface with the following details:

- Schemas:** On the left, the schemas are listed: `empresapruебadancabper`, `hospitaldancabper` (selected), `Nueva`, `personas`, `information_schema`, `mysql`, `performance_schema`, `phpmyadmin`, and `test`.
- Query Editor:** The SQL query is: `Select * from emple where salario>2000`.
- Table View:** The results table has columns: emp_no, apellido, oficio, dir, fecha_alt, salario, comision, dept_no. The data shows 8 rows of employees.

emp_no	apellido	oficio	dir	fecha_alt	salario	comision	dept_no
7566	JIMÉNEZ	DIRECTOR	7839	1991-04-02	2900	NULL	20
7698	NEGRO	DIRECTOR	7839	1991-05-01	3005	NULL	30
7782	CEREZO	DIRECTOR	7839	1991-06-09	2885	NULL	10
7788	GIL	ANALISTA	7566	1991-11-09	3000	NULL	20
7839	REY	PRESIDENTE	NULL	1991-11-17	4100	NULL	10
7902	FERNÁNDEZ	ANALISTA	7566	1991-12-03	3000	NULL	20

h) Mostrar los datos de los empleados cuyo oficio sea 'ANALISTA'.

The screenshot shows the MySQL Workbench interface with the following details:

- Schemas:** On the left, the schemas are listed: `'empresapruебadancabper'`.`depart` (selected), `'hospitaldancabper'`.`personas` (selected), `Nueva`, `personas`, `information_schema`, `mysql`, `performance_schema`, `phpmyadmin`, and `test`.
- Query Editor:** The SQL query is: `Select * from emple where oficio='ANALISTA'`.
- Table View:** The results table has columns: emp_no, apellido, oficio, dir, fecha_alt, salario, comision, dept_no. The data shows 2 rows of employees.

emp_no	apellido	oficio	dir	fecha_alt	salario	comision	dept_no
7788	GIL	ANALISTA	7566	1991-11-09	3000	NULL	20
7902	FERNÁNDEZ	ANALISTA	7566	1991-12-03	3000	NULL	20

i) Seleccionar el apellido y oficio de los empleados del departamento número 20.

The screenshot shows the MySQL Workbench interface with the following details:

- Schemas:** On the left, the schemas are listed: `empresapruебadancabper`, `hospitaldancabper` (selected), `Nueva`, `personas`, `information_schema`, `mysql` (selected), `performance_schema`, `phpmyadmin`, and `test`.
- Query Editor:** The SQL query is: `Select apellido, oficio from emple where dept_no=20`.
- Table View:** The results table has columns: apellido, oficio. The data shows 5 rows of employees.

apellido	oficio
SÁNCHEZ	EMPLEADO
JIMÉNEZ	DIRECTOR
GIL	ANALISTA
ALONSO	EMPLEADO
FERNÁNDEZ	ANALISTA

j) Mostrar todos los datos de los empleados ordenados por apellido.

3

The screenshot shows the phpMyAdmin interface with the database 'empresapruебadancabper' selected. A query 'Select * from emple order by apellido' is run, resulting in 14 rows of employee data. The columns are emp_no, apellido, oficio, dir, fecha_alt, salario, comision, and dept_no. The results are ordered by apellido in ascending order.

emp_no	apellido	oficio	dir	fecha_alt	salario	comision	dept_no
7876	ALONSO	EMPLEADO	7788	1991-09-23	1430	NULL	20
7499	ARROYO	VENDEDOR	7698	1990-02-20	1500	390	30
7782	CEREZO	DIRECTOR	7839	1991-06-09	2885	NULL	10
7902	FERNÁNDEZ	ANALISTA	7566	1991-12-03	3000	NULL	20
7788	GIL	ANALISTA	7566	1991-11-09	3000	NULL	20
7566	JIMÉNEZ	DIRECTOR	7839	1991-04-02	2900	NULL	20
7900	JIMENO	EMPLEADO	7698	1991-12-03	1335	NULL	30
7654	MARTÍN	VENDEDOR	7608	1991-09-29	1600	1020	30

k) Seleccionar los empleados cuyo oficio sea 'VENDEDOR'. Mostrar los datos ordenados por apellido.

The screenshot shows the phpMyAdmin interface with the database 'empresapruебadancabper' selected. A query 'Select * from emple where oficio='VENDEDOR' order by apellido' is run, resulting in 4 rows of employee data. The results are ordered by apellido in ascending order.

emp_no	apellido	oficio	dir	fecha_alt	salario	comision	dept_no
7499	ARROYO	VENDEDOR	7698	1990-02-20	1500	390	30
7654	MARTÍN	VENDEDOR	7698	1991-09-29	1600	1020	30
7521	SALA	VENDEDOR	7698	1991-02-22	1625	650	30

l) Mostrar los empleados cuyo departamento sea 10 y cuyo oficio sea 'ANALISTA'. Ordenar el resultado por apellido.

Solo hay 2 analistas y están en el departamento 20

The screenshot shows the phpMyAdmin interface with the database 'empresapruебadancabper' selected. A query 'Select * from emple where dept_no=10 and oficio='ANALISTA'' is run. A message box indicates 'MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas)' (MySQL returned an empty set (i.e. zero rows)).

Operaciones sobre los resultados de la consulta

[Crear vista](#)

The screenshot shows the MySQL Workbench interface. On the left, the database schema is visible with databases like 'empresapruebadancabper', 'hospitaldancabper', 'information_schema', 'mysql', 'performance_schema', 'phpmyadmin', and 'test'. A tree view shows tables such as 'Nueva' and 'personas' under 'hospitaldancabper'. On the right, a query results grid displays the output of the SQL query: 'Select * from emple where oficio='ANALISTA''. The results show two rows of employee data.

emp_no	apellido	oficio	dir	fecha_alt	salario	comision	dept_no
7788	GIL	ANALISTA	7566	1991-11-09	3000	NULL	20
7902	FERNANDEZ	ANALISTA	7566	1991-12-03	3000	NULL	20

- m) Mostrar los empleados que tengan un salario mayor que 200000 o que pertenezcan al departamento número 20.

The screenshot shows the MySQL Workbench interface with the query 'Select * from emple where salario>200000 or dept_no=20'. The results grid shows five rows of employee data, all belonging to department 20.

emp_no	apellido	oficio	dir	fecha_alt	salario	comision	dept_no
7369	SÁNCHEZ	EMPLEADO	7902	1990-12-17	1040	NULL	20
7566	JIMÉNEZ	DIRECTOR	7839	1991-04-02	2900	NULL	20
7788	GIL	ANALISTA	7566	1991-11-09	3000	NULL	20
7876	ALONSO	EMPLEADO	7788	1991-09-23	1430	NULL	20
7902	FERNÁNDEZ	ANALISTA	7566	1991-12-03	3000	NULL	20

- n) Ordenar los empleados por oficio, y dentro de oficio por nombre.

The screenshot shows the MySQL Workbench interface with the query 'Select * from emple order by oficio, apellido'. The results grid shows six rows of employee data, ordered first by job title ('ANALISTA', 'DIRECTOR', 'EMPLEADO') and then by last name ('FERNÁNDEZ', 'GIL', 'CEREZO', 'JIMÉNEZ', 'NEGRO', 'ALONSO').

emp_no	apellido	oficio	dir	fecha_alt	salario	comision	dept_no
7902	FERNÁNDEZ	ANALISTA	7566	1991-12-03	3000	NULL	20
7788	GIL	ANALISTA	7566	1991-11-09	3000	NULL	20
7782	CEREZO	DIRECTOR	7839	1991-06-09	2885	NULL	10
7566	JIMÉNEZ	DIRECTOR	7839	1991-04-02	2900	NULL	20
7698	NEGRO	DIRECTOR	7839	1991-05-01	3005	NULL	30
7876	ALONSO	EMPLEADO	7788	1991-09-23	1430	NULL	20

- o) Seleccionar de la tabla EMPLE los empleados cuyo apellido empiece por 'A'.

3

Select * from emple where apellido like 'A%'

emp_no	apellido	oficio	dir	fecha_alt	salario	comision	dept_no
7499	ARROYO	VENDEDOR	7698	1990-02-20	1500	390	30
7876	ALONSO	EMPLEADO	7788	1991-09-23	1430	NULL	20

- p) Seleccionar de la tabla EMPLE los empleados cuyo apellido termine por 'Z'.

Select * from emple where apellido like '%Z'

emp_no	apellido	oficio	dir	fecha_alt	salario	comision	dept_no
7369	SÁNCHEZ	EMPLEADO	7902	1990-12-17	1040	NULL	20
7566	JIMÉNEZ	DIRECTOR	7839	1991-04-02	2900	NULL	20
7902	FERNÁNDEZ	ANALISTA	7566	1991-12-03	3000	NULL	20
7934	MUÑOZ	EMPLEADO	7782	1992-01-23	1690	NULL	10

- q) Seleccionar de la tabla EMPLE aquellas filas cuyo APELLIDO empiece por 'A' y el OFICIO tenga una 'E' en cualquier posición.

Select * from emple where apellido like 'A%' and oficio like '%E%'

emp_no	apellido	oficio	dir	fecha_alt	salario	comision	dept_no
7499	ARROYO	VENDEDOR	7698	1990-02-20	1500	390	30
7876	ALONSO	EMPLEADO	7788	1991-09-23	1430	NULL	20

- r) Seleccionar los empleados cuyo salario esté entre 100000 y 200000. Utilizar el operador BETWEEN.

3

```
Select * from emple where salario between 1000 and 1700
```

Mostrar todo | Número de filas: 25 ▾ Filtrar filas: Buscar en esta tabla

Ordenar según la clave: Ninguna ▾

+ Opciones

	emp_no	apellido	oficio	dir	fecha_alt	salario	comision	dept_no
<input type="checkbox"/>	7369	SÁNCHEZ	EMPLEADO	7902	1990-12-17	1040	NULL	20
<input type="checkbox"/>	7499	ARROYO	VENDEDOR	7698	1990-02-20	1500	390	30
<input type="checkbox"/>	7521	SALA	VENDEDOR	7698	1991-02-22	1625	650	30
<input type="checkbox"/>	7654	MARTÍN	VENDEDOR	7698	1991-09-29	1600	1020	30
<input type="checkbox"/>	7844	TOVAR	VENDEDOR	7698	1991-09-08	1350	0	30
<input type="checkbox"/>	7876	ALONSO	EMPLEADO	7700	1991-09-23	1430	NULL	20

- s) Obtener los empleados cuyo oficio sea ‘VENDEDOR’ y tengan una comisión superior a 100000.

```
Select * from emple where oficio='VENDEDOR' and comision>600
```

Mostrar todo | Número de filas: 25 ▾ Filtrar filas: Buscar en esta tabla

Ordenar según la clave: Ninguna ▾

+ Opciones

	emp_no	apellido	oficio	dir	fecha_alt	salario	comision	dept_no
<input type="checkbox"/>	7521	SALA	VENDEDOR	7698	1991-02-22	1625	650	30
<input type="checkbox"/>	7654	MARTÍN	VENDEDOR	7698	1991-09-29	1600	1020	30

↑ Seleccionar todo Para los elementos que están marcados: Editar Copiar Borrar Exportar

- t) Seleccionar los datos de los empleados ordenados por número de departamento, y dentro de cada departamento ordenados por apellido.

```
Select * from emple order by dept_no, apellido
```

Mostrar todo | Número de filas: 25 ▾ Filtrar filas: Buscar en esta tabla

Ordenar según la clave: Ninguna ▾

+ Opciones

	emp_no	apellido	oficio	dir	fecha_alt	salario	comision	dept_no
<input type="checkbox"/>	7782	CEREZO	DIRECTOR	7839	1991-06-09	2885	NULL	10
<input type="checkbox"/>	7934	MUÑOZ	EMPLEADO	7782	1992-01-23	1690	NULL	10
<input type="checkbox"/>	7839	REY	PRESIDENTE	NULL	1991-11-17	4100	NULL	10
<input type="checkbox"/>	7876	ALONSO	EMPLEADO	7788	1991-09-23	1430	NULL	20
<input type="checkbox"/>	7902	FERNÁNDEZ	ANALISTA	7566	1991-12-03	3000	NULL	20

- u) Número y apellidos de los empleados cuyo apellido termine por 'Z' y tengan un salario superior a 3000.

```
Select emp_no, apellido from emple where apellido like '%Z' and salario>3000
```

Mostrar todo | Número de filas: 25 | Filtrar filas: Buscar en esta tabla

Ordenar según la clave: Ninguna

+ Opciones

	emp_no	apellido
<input type="checkbox"/> Editar Copiar Borrar	7566	JIMÉNEZ
<input type="checkbox"/> Editar Copiar Borrar	7902	FERNÁNDEZ

- v) Datos de los departamentos cuya localización empiece por 'B'.

```
Select * from depart where loc like 'B%'
```

Mostrar todo | Número de filas: 25 | Filtrar filas: Buscar en esta tabla

+ Opciones

dept_no	dnombre	loc
30	VENTAS	BARCELONA
40	PRODUCCIÓN	BILBAO

- w) Datos de los empleados cuyo oficio sea 'EMPLEADO', tengan un salario superior a 1000 y pertenezcan al departamento número 10.

```
Select * from emple where oficio='EMPLEADO' and salario>1000 and dept_no=10
```

Mostrar todo | Número de filas: 25 | Filtrar filas: Buscar en esta tabla

+ Opciones

	emp_no	apellido	oficio	dir	fecha_alt	salario	comision	dept_no
<input type="checkbox"/> Editar Copiar Borrar	7934	MUÑOZ	EMPLEADO	7782	1992-01-23	1690	NULL	10

↑ Seleccionar todo Para los elementos que están marcados: Editar Copiar Borrar Exportar

- x) Mostrar los apellidos de los empleados que no tengan comisión.

4

```
Select apellido from emple where comision is null
```

Mostrar todo | Número de filas: 25 Filt

Ordenar según la clave: Ninguna

+ Opciones
apellido
SÁNCHEZ
JIMÉNEZ
NEGRO
CEREZO
GIL
REY
ALONSO
JIMENO
FERNÁNDEZ
MUÑOZ

- y) Mostrar los apellidos de los empleados que no tengan comisión y cuyo apellido empiece por 'J'.

```
Select apellido from emple where comision is null and apellido like 'J%'
```

Mostrar todo | Número de filas: 25 Filtrar filas: Buscar en est

Ordenar según la clave: Ninguna

+ Opciones
apellido
JIMÉNEZ
JIMENO

- z) Mostrar los apellidos de los empleados cuyo oficio sea 'VENDEDOR', 'ANALISTA' o 'EMPLEADO'.

4

```
Select apellido from emple where oficio in ('VENDEDOR', 'ANALISTA', 'EMPLEADO')
```

Mostrar todo | Número de filas: 25 ▾ Filtrar filas: Buscar en esta tabla

Ordenar según la clave: Ninguna ▾

+ Opciones

apellido

- SÁNCHEZ
- ARROYO
- SALA
- MARTÍN
- GIL
- TOVAR
- ALONSO
- JIMENO
- FERNÁNDEZ
- MUÑOZ

- aa) Mostrar los apellidos de los empleados cuyo oficio no sea ni 'ANALISTA' ni 'EMPLEADO', y además tengan un salario mayor de 2000.

```
Select apellido from emple where oficio not in ('ANALISTA', 'EMPLEADO') and salario>2000
```

Mostrar todo | Número de filas: 25 ▾ Filtrar filas: Buscar en esta tabla

Ordenar según la clave: Ninguna ▾

+ Opciones

apellido

- JIMÉNEZ
- NEGRO
- CEREZO
- REY

bb) Seleccionar de la tabla EMPLE los empleados cuyo salario esté entre 2000000 y 3000000 (utilizar BETWEEN).

4

```
Select * from emple where salario between 2000 and 3000
```

Mostrar todo | Número de filas: 25 | Filtrar filas: Buscar en esta tabla

Ordenar según la clave: Ninguna

+ Opciones

	emp_no	apellido	oficio	dir	fecha_alt	salario	comision	dept_no
<input type="checkbox"/>	7566	JIMÉNEZ	DIRECTOR	7839	1991-04-02	2900	NULL	20
<input type="checkbox"/>	7782	CEREZO	DIRECTOR	7839	1991-06-09	2885	NULL	10
<input type="checkbox"/>	7788	GIL	ANALISTA	7566	1991-11-09	3000	NULL	20
<input type="checkbox"/>	7902	FERNÁNDEZ	ANALISTA	7566	1991-12-03	3000	NULL	20

cc) Seleccionar el apellido, salario y número de departamento de los empleados cuyo salario sea mayor que 200000 en los departamentos 10 ó 30.

```
Select apellido, salario, dept_no from emple where salario>2000 and (dept_no=10 or dept_no=30)
```

Mostrar todo | Número de filas: 25 | Filtrar filas: Buscar en esta tabla

Ordenar según la clave: Ninguna

+ Opciones

apellido	salario	dept_no
NEGRO	3005	30
CEREZO	2885	10
REY	4100	10

dd) Mostrar el apellido y número de los empleados cuyo salario no esté entre 100000 y 200000 (utilizar BETWEEN).

4

```
Select apellido, emp_no from emple where salario not between 1000 and 2000
```

Mostrar todo | Número de filas: 25 ▾ Filtrar filas: Buscar en esta tabla

Ordenar según la clave: Ninguna ▾

+ Opciones

← T →	▼	apellido	emp_no					
<input type="checkbox"/>		Editar		Copiar		Borrar	JIMÉNEZ	7566
<input type="checkbox"/>		Editar		Copiar		Borrar	NEGRO	7698
<input type="checkbox"/>		Editar		Copiar		Borrar	CEREZO	7782
<input type="checkbox"/>		Editar		Copiar		Borrar	GIL	7788
<input type="checkbox"/>		Editar		Copiar		Borrar	REY	7839
<input type="checkbox"/>		Editar		Copiar		Borrar	FERNÁNDEZ	7902

ee) Obtener el apellido de todos los empleados en minúscula.

```
Select lower(apellido) from emple
```

Mostrar todo | Número de filas:

Ordenar según la clave: Ninguna

+ Opciones

lower(apellido)

sánchez
arroyo
sala
jiménez
martín
negro
cerezo
gil
rey
tovar
alonso
jimeno
fernández
muñoz

ff) En una consulta concatena el apellido de cada empleado con su oficio.

4

<pre>Select concat(apellido,oficio) from emple</pre>
<input type="checkbox"/> Mostrar todo Número de filas: 25 ▾
Filtrar filas: Buscar en e
Ordenar según la clave: Ninguna ▾
- Opciones
concat(apellido,oficio)
SÁNCHEZEMPLEADO
ARROYOVENDEDOR
SALAVENDEDOR
JIMÉNEZDIRECTOR
MARTÍNVENDEDOR
NEGRODIRECTOR
CEREZODIRECTOR
GILANALISTA
REYPRESIDENTE
TOVARVENDEDOR
ALONSOEMPLEADO
JIMENOEMPLEADO
FERNÁNDEZANALISTA
MUÑOZEMPLEADO

gg) Mostrar el apellido y la longitud del apellido (función LENGTH) de todos los empleados, ordenados por la longitud de los apellidos de los empleados descendenteamente.

4

Select apellido,length(apellido) from emple order by length(apellido) desc	
<input type="checkbox"/> Mostrar todo	Número de filas: 25
	Filtrar filas: Buscar en esta tabla
Ordenar según la clave:	Ninguna
+ Opciones	
apellido	length(apellido)
FERNÁNDEZ	9
SÁNCHEZ	7
JIMÉNEZ	7
ARROYO	6
MARTÍN	6
JIMENO	6
CEREZO	6
ALONSO	6
TOVAR	5
MUÑOZ	5
NEGRO	5
SALA	4
REY	3
GIL	3

hh) Obtener el año de contratación de todos los empleados (función YEAR).

4

```
Select year(fecha_alt) from emple
```

Mostrar todo | Número de filas: 25 ▾

Ordenar según la clave: Ninguna ▾

+ Opciones

year(fecha_alt)

1990

1990

1991

1991

1991

1991

1991

1991

1991

1991

1991

1991

1991

1991

1991

1992

- ii) Mostrar los datos de los empleados que hayan sido contratados en el año 1992.

```
Select * from emple where year(fecha_alt)=1992
```

Mostrar todo | Número de filas: 25 ▾ Filtrar filas: Buscar en esta tabla

+ Opciones

emp_no	apellido	oficio	dir	fecha_alt	salario	comision	dept_no
7934	MUÑOZ	EMPLEADO	7782	1992-01-23	1690	NULL	10

Editar Copiar Borrar

- jj) Mostrar los datos de los empleados que hayan sido contratados en el mes de febrero de cualquier año (función MONTHNAME).

4

Select * from emple where monthname(fecha_alt)='February'
<input type="checkbox"/> Mostrar todo Número de filas: 25 <input type="button" value="▼"/> Filtrar filas: Buscar en esta tabla
Ordenar según la clave: Ninguna <input type="button" value="▼"/>
+ Opciones <input type="button" value="←"/> <input type="button" value="→"/> ▼ emp_no apellido oficio dir fecha_alt salario comision dept_no <input type="checkbox"/> Editar Copiar Borrar 7499 ARROYO VENDEDOR 7698 1990-02-20 1500 390 30 <input type="checkbox"/> Editar Copiar Borrar 7521 SALA VENDEDOR 7698 1991-02-22 1625 650 30

- kk) En cada departamento mostrar el apellido del empleado que tiene el salario mayor.

Select apellido,dept_no,greatest(dept_no,salario) from emple																																													
<input type="checkbox"/> Mostrar todo Número de filas: 25 <input type="button" value="▼"/> Filtrar filas: Buscar																																													
Ordenar según la clave: Ninguna <input type="button" value="▼"/>																																													
+ Opciones <table border="1"> <thead> <tr> <th>apellido</th> <th>dept_no</th> <th>greatest(dept_no,salario)</th> </tr> </thead> <tbody> <tr><td>SÁNCHEZ</td><td>20</td><td>1040</td></tr> <tr><td>ARROYO</td><td>30</td><td>1500</td></tr> <tr><td>SALA</td><td>30</td><td>1625</td></tr> <tr><td>JIMÉNEZ</td><td>20</td><td>2900</td></tr> <tr><td>MARTÍN</td><td>30</td><td>1600</td></tr> <tr><td>NEGRO</td><td>30</td><td>3005</td></tr> <tr><td>CEREZO</td><td>10</td><td>2885</td></tr> <tr><td>GIL</td><td>20</td><td>3000</td></tr> <tr><td>REY</td><td>10</td><td>4100</td></tr> <tr><td>TOVAR</td><td>30</td><td>1350</td></tr> <tr><td>ALONSO</td><td>20</td><td>1430</td></tr> <tr><td>JIMENO</td><td>30</td><td>1335</td></tr> <tr><td>FERNÁNDEZ</td><td>20</td><td>3000</td></tr> <tr><td>MUÑOZ</td><td>10</td><td>1690</td></tr> </tbody> </table>	apellido	dept_no	greatest(dept_no,salario)	SÁNCHEZ	20	1040	ARROYO	30	1500	SALA	30	1625	JIMÉNEZ	20	2900	MARTÍN	30	1600	NEGRO	30	3005	CEREZO	10	2885	GIL	20	3000	REY	10	4100	TOVAR	30	1350	ALONSO	20	1430	JIMENO	30	1335	FERNÁNDEZ	20	3000	MUÑOZ	10	1690
apellido	dept_no	greatest(dept_no,salario)																																											
SÁNCHEZ	20	1040																																											
ARROYO	30	1500																																											
SALA	30	1625																																											
JIMÉNEZ	20	2900																																											
MARTÍN	30	1600																																											
NEGRO	30	3005																																											
CEREZO	10	2885																																											
GIL	20	3000																																											
REY	10	4100																																											
TOVAR	30	1350																																											
ALONSO	20	1430																																											
JIMENO	30	1335																																											
FERNÁNDEZ	20	3000																																											
MUÑOZ	10	1690																																											

- II) Mostrar los datos de los empleados cuyo apellido empiece por 'A' y hayan sido contratados en el año 1990.

4

```
Select apellido from emple where apellido like 'A%' and year(fecha_alt)=1990
```

Mostrar todo

Número de filas:

25

Filtrar filas:

Buscar en esta tabla

+ Opciones

apellido

ARROYO

- mm) Mostrar los datos de los empleados del departamento 10 que no tengan comisión.

```
Select * from emple where dept_no=10 and comision is null
```

Mostrar todo

Número de filas:

25

Filtrar filas:

Buscar en esta tabla

Ordenar según la clave:

Ninguna

+ Opciones

	emp_no	apellido	oficio	dir	fecha_alt	salario	comision	dept_no
<input type="checkbox"/>  Editar  Copiar  Borrar	7782	CEREZO	DIRECTOR	7839	1991-06-09	2885	NULL	10
<input type="checkbox"/>  Editar  Copiar  Borrar	7839	REY	PRESIDENTE	NULL	1991-11-17	4100	NULL	10
<input type="checkbox"/>  Editar  Copiar  Borrar	7934	MUÑOZ	EMPLEADO	7782	1992-01-23	1690	NULL	10