

## UD 5-1.

### EJERCICIOS SOLUCIÓN. DISEÑO FISICO. DML.

Base de Datos  
CFGS DAW

Francisco Aldarias Raya  
[paco.aldarias@ceedcv.es](mailto:paco.aldarias@ceedcv.es)

2019/2020

Fecha 09/12/19

Versión:191209.0919

## Licencia



**Reconocimiento - NoComercial - CompartirIgual (by-nc-sa):** No se permite un uso comercial de la obra original ni de las posibles obras derivadas, la distribución de las cuales se debe hacer con una licencia igual a la que regula la obra original.

## Nomenclatura

A lo largo de este tema se utilizarán distintos símbolos para distinguir elementos importantes dentro del contenido. Estos símbolos son:



Importante



Atención



Interesante

## Revisiones

## ÍNDICE DE CONTENIDO

<b>1.EJERCICIO 1.....</b>	<b>3</b>
1.1 Solución.....	5
<b>2.EJERCICIO 2.....</b>	<b>7</b>
2.1 Solución.....	8
2.1.1 Crear la tabla Empleados conservando la integridad referencial.....	8
2.1.2 Insertar los datos de la imagen en la nueva tabla Empleados que acabamos de crear. No introduzcas el último registro.....	10
2.1.3 Borrar el departamento de Marketing de la tabla Departamentos.....	12

## UD05-1. EJERCICIOS SOLUCIÓN. DISEÑO FISICO. DML.

Para realizar la práctica podemos utilizar cualquiera de los sistemas de bases de datos: MySQL, Oracle y Postgres. Se puede hacer con uno y si se dispone de más tiempo hacerlo con el resto. Cita al inicio la BBDD que vas a utilizar ya que hay pequeñas diferencias.

### 1. EJERCICIO 1

El técnico informático de la empresa MiPC S.L., después de analizar su funcionamiento, ha decidido crear las siguientes tablas:

Tabla Proveedores.

<u>CodProveedor</u> (V10)	NombreProveedor (V30)	CodPostal (V5)
TO342	JUGUETOS, S.A.	45600
MA280	TOYPLAY, S.A.	28005
BA843	CARMELO DIAZ, S.L.	06004
SE391	ARTEAND, S.L.	41400

Tabla ProductosPed

<u>RefeProducto</u> (V10)	NombreProducto V(30)	Precio (Float)
NPP10	NAIPES PETER PARKER	3,00
P3R20	PATINETE 3 RUEDAS	22,50
AFK11	AVION FK20	31,75
PM30	PELUCHE MAYA	15,00
HM12	HOOP MUSICAL	12,80
BB75	BOLA BOOM	22,20

Tabla Pedidos

<u>NumPedido</u> (Int)	Fecha (Date)	Proveedor ▲ (V10)
1	10/06/2013	TO342
2	10/06/2013	MA280
3	12/06/2013	BA843
4	14/06/2013	TO342
5	14/06/2013	MA280

ProductosPedido

<u>NumPedido</u> ▲ (Int)	<u>RefeProducto</u> ▲ (V10)	Cantidad (Int)
1	NPP10	10
1	AFK11	12
2	P3R20	15
3	P3R20	10
3	PM30	20
3	HM12	10
4	AFK11	30
4	BB75	12
5	P3R20	18
5	NPP10	3
5	BB75	5

Crea las tablas indicadas e inserta los datos que aparecen en ellas.

(En cada campo se indica el tpo de dato que hay que utilizar al crearlo, por ejemplo (V10) es un varchar de 10)

Para insertar fechas en Oracle utilizaremos la función TO\_DATE(fecha\_string,fecha\_formato) que permite teclear la fecha en el formato que deseemos. Por ejemplo: TO\_DATE('10-12-2010','DD-MM-YYYY')

### 1.1 Solución

```
CREATE TABLE proveedores (  
  CodProveedor varchar(10),  
  NombreProveedor varchar(30),  
  CodPostal varchar(5),  
  PRIMARY KEY (CodProveedor)  
);  
  
CREATE TABLE productosped (  
  RefeProducto varchar(10),  
  NombreProducto varchar(30),  
  Prefio float,  
  PRIMARY KEY (RefeProducto)  
);  
  
CREATE TABLE pedidos (  
  NumPedido int,  
  Fecha date,  
  Proveedor varchar(10),  
  PRIMARY KEY (NumPedido),  
  FOREIGN KEY (Proveedor) REFERENCES proveedores (CodProveedor)  
);  
  
CREATE TABLE productospedido (  
  NumPedido int,  
  RefeProducto varchar(10),  
  Cantidad int,  
  PRIMARY KEY (NumPedido, RefeProducto),  
  FOREIGN KEY (NumPedido) REFERENCES pedidos (NumPedido),  
  FOREIGN KEY (RefeProducto) REFERENCES productosped (RefeProducto)  
);
```

Esta solución te debe funcionar tanto en MySQL como en Oracle.

Ahora vamos a insertar algunos registros en cada tabla, el resto lo dejo para que practiques tú.

Insertamos datos en la tabla proveedores.

```
mysql> insert into proveedores
-> values ('T0342','JUGUETOS, S.A.','45600');
Query OK, 1 row affected (0.03 sec)

mysql> insert into proveedores
-> values ('MA280','TOYPLAY, S.A.','28005');
Query OK, 1 row affected (0.03 sec)

mysql> SELECT * FROM proveedores;
+-----+-----+-----+
| CodProveedor | NombreProveedor | CodPostal |
+-----+-----+-----+
| MA280        | TOYPLAY, S.A.   | 28005     |
| T0342        | JUGUETOS, S.A.  | 45600     |
+-----+-----+-----+
2 rows in set (0.00 sec)
```

Insertamos datos en la tabla productosped.

```
mysql> insert into productosped
-> values ('NPP10','NAIPES PETER PARKER',3.0);
Query OK, 1 row affected (0.06 sec)

mysql> insert into productosped
-> values ('P3R20','PATINETE 3 RUEDAS',22.50);
Query OK, 1 row affected (0.05 sec)

mysql> insert into productosped
-> values ('AFK11','AVION FK20',31.75);
Query OK, 1 row affected (0.03 sec)
```

Es la primera vez que insertamos datos en un campo float, debes tener en cuenta que como es un número va sin comillas y además el separador decimal es el punto (fíjate que si utilizásemos la coma como separador decimal, el sistema interpretaría que le estamos dando dos valores en lugar de uno con parte decimal pues el símbolo que empleamos para separar los campos es la coma).

```
mysql> SELECT * FROM productosped;
+-----+-----+-----+
| RefeProducto | NombreProducto   | Prefio |
+-----+-----+-----+
| AFK11        | AVION FK20       | 31.75  |
| NPP10        | NAIPES PETER PARKER | 3      |
| P3R20        | PATINETE 3 RUEDAS | 22.5   |
+-----+-----+-----+
3 rows in set (0.00 sec)
```

Insertamos datos en la tabla pedidos. Recuerda que los campos tipo fecha son diferentes en MySQL y Oracle, aquí el ejemplo se ha realizado sobre MySQL, en Oracle hay que utiliza TO\_DATE('10-06-2013','DD-MM-YYYY').

Espero que también hayas observado que los campos de tipo entero también van sin comillas.

```
mysql> insert into pedidos
-> values (1,'2013-06-10','T0342');
Query OK, 1 row affected (0.05 sec)

mysql> insert into pedidos
-> values (2,'2013-06-10','MA280');
Query OK, 1 row affected (0.06 sec)

mysql> SELECT * FROM pedidos;
+-----+-----+-----+
| NumPedido | Fecha       | Proveedor |
+-----+-----+-----+
|          1 | 2013-06-10 | T0342     |
|          2 | 2013-06-10 | MA280     |
+-----+-----+-----+
2 rows in set (0.00 sec)
```

Por último insertamos datos en la tabla productospedido y mostramos el resultado.

```
mysql> source c:\src\laberinto.sql
Query OK, 0 rows affected (0.13 sec)

mysql> insert into productospedido
-> values (1,'NPP10',10);
Query OK, 1 row affected (0.03 sec)

mysql> insert into productospedido
-> values(1,'AFK11',12);
Query OK, 1 row affected (0.06 sec)

mysql> insert into productospedido
-> values(2,'P3R20',15);
Query OK, 1 row affected (0.06 sec)

mysql> select * from productospedido;
+-----+-----+-----+
| NumPedido | RefeProducto | Cantidad |
+-----+-----+-----+
|          1 | AFK11        | 12       |
|          1 | NPP10        | 10       |
|          2 | P3R20        | 15       |
+-----+-----+-----+
3 rows in set (0.00 sec)
```

## 2. EJERCICIO 2

Recordemos la tabla departamentos que hemos creado en la teoría:

```
SQL> select * from departamentos;
```

CODDPTO	NOMBRE	UBICACION
INF	Informática	Planta sótano U3
ADM	Administración	Planta quinta U2
COM	Comercial	Planta tercera U3
CONT	Contabilidad	Planta quinta U1
MKT	Marketing	Planta cuarta U5
ALM	Almacén	Planta baja U1

```
6 rows selected.
```

Vamos a crear ahora la tabla de Empleados

DNI	Nombre	Especialidad	FechaAlta	Dpto ▲
12345678A	Alberto Gil	Contable	10/12/2010	CONT
23456789B	Mariano Sanz	Informática	04/10/2011	INF
34567890C	Iván Gómez	Ventas	20/07/2012	COM
45678901D	Ana Silván	Informática	25/11/2012	INF
56789012E	María Cuadrado	Ventas	02/04/2013	COM
67890123A	Roberto Milán	Logística	05/02/2010	ALM

1. Crear la tabla Empleados conservando la integridad referencial (ON DELETE CASCADE, ON UPDATE CASCADE).
2. Insertar los datos de la imagen en la nueva tabla Empleados que acabamos de crear. No introduzcas el último registro.
3. Borrar el departamento de Marketing de la tabla Departamentos.
4. Insertar ahora el último registro de la tabla Empleados, pero en el dpto. De Marketing. ¿Qué ocurre? Razona la respuesta.
5. Insertar ahora el último registro de la tabla Empleados tal y como aparece.
6. Eliminar el departamento Comercial (COM) y comprobar que se cumple la integridad referencial, es decir, todos los empleados de ese departamento deben ser eliminados automáticamente.
7. Modificar el código del departamento de informática (INF) por (IT). Comprobar que se cumple la cláusula ON UPDATE CASCADE. Cita la BBDD que has utilizado ya que hay pequeñas diferencias.

## 2.1 Solución

### 2.1.1 Crear la tabla Empleados conservando la integridad referencial

MySQL.

La instrucción para crear la tabla será:



```

CREATE TABLE empleados(
  dni VARCHAR(10) ,
  nombre VARCHAR(30) ,
  especialidad VARCHAR(25) ,
  fechaalta DATE ,
  dpto VARCHAR(10) ,
  PRIMARY KEY (dni) ,
  FOREIGN KEY (dpto) REFERENCES departamentos(CodDpto)
    ON DELETE CASCADE ON UPDATE CASCADE
) ENGINE=InnoDB;

```

Para crear esta tabla escribiremos el contenido en un documento de texto y ejecutaremos el script correspondiente.

```

mysql> source c:\src\empleados.sql
Query OK, 0 rows affected (0.08 sec)

mysql> desc empleados;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| dni   | varchar(10) | NO | PRI | NULL | |
| nombre | varchar(30) | YES | | NULL | |
| especialidad | varchar(25) | YES | | NULL | |
| fechaalta | date | YES | | NULL | |
| dpto   | varchar(10) | YES | MUL | NULL | |
+-----+-----+-----+-----+-----+-----+
5 rows in set (0.02 sec)

```

Oracle.

Recuerda que en Oracle no podemos utilizar la cláusula ON UPDATE, por ello la instrucción a utilizar en Oracle será:

```

CREATE TABLE empleados(
  dni VARCHAR(10) ,
  nombre VARCHAR(30) ,
  especialidad VARCHAR(25) ,
  fechaalta DATE ,
  dpto VARCHAR(10) ,
  PRIMARY KEY (dni) ,
  FOREIGN KEY (dpto) REFERENCES departamentos(CodDpto)
    ON DELETE CASCADE
);

```

Como antes lo escribiremos en un archivo de texto y ejecutaremos el script. Hasta ahora hemos ejecutado el Script con @ que es el modo abreviado, pero también se pueden ejecutar con START seguido el nombre del script, como en este ejemplo.

```
SQL> START C:\SRC\EMPLEADOS.SQL
Table created.
```

Comprobamos el resultado.

```
SQL> DESC EMPLEADOS;
Name                                     Null?      Type
-----
DNI                                     NOT NULL   VARCHAR2(10)
NOMBRE                                VARCHAR2(30)
ESPECIALIDAD                          VARCHAR2(25)
FECHAALTA                             DATE
DPTO                                   VARCHAR2(10)
```

### 2.1.2 Insertar los datos de la imagen en la nueva tabla Empleados que acabamos de crear. No introduzcas el último registro.

Vamos a crear uno a mano y después crearemos un script para los cuatro siguientes, dejando el último, tal y como nos pide el enunciado:

#### MySQL.

MySQL almacena los campo de tipo fecha con el formato aaaa-mm-dd, es decir: año, mes, día. Para asignar una fecha debemos colocarla en este formato y entre comillas.

```
mysql> insert into empleados
-> values ('12345678A','Alberto Gil','Contable','2010-12-10','CONT');
Query OK, 1 row affected (0.06 sec)

mysql> select * from empleados;
+-----+-----+-----+-----+-----+
| dni   | nombre | especialidad | fechaalta | dpto |
+-----+-----+-----+-----+-----+
| 12345678A | Alberto Gil | Contable | 2010-12-10 | CONT |
+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

Como puedes observar el registro ha sido añadido.

#### Oracle.

Para insertar fechas en Oracle utilizaremos la función TO\_DATE(fecha\_string, fecha\_formato) tal y como nos indicaba el enunciado:

```
SQL> INSERT INTO empleados
  2 values ('12345678A','Alberto Gil','Contable',
  3 TO_DATE('10-12-2010','DD-MM-YYYY'),'CONT');
```

```
1 row created.
```

```
SQL> select * from empleados;
```

DNI	NOMBRE	ESPECIALIDAD	FECHAALT
12345678A	Alberto Gil	Contable	10/12/10

Vamos a añadir ahora el resto de registros menos el último. Para ello crearemos los scripts correspondientes para cada base de datos.

### MySQL.

Creamos un script con el siguiente contenido.

```
INSERT INTO empleados values
('23456789B','Mariano Sanz','Informática',TO_DATE('04-10-2011','DD-MM-YYYY'),'INF');
INSERT INTO empleados
values ('34567890C','Iván Gómez','Ventas',TO_DATE('20-07-2012','DD-MM-YYYY'),'COM');
INSERT INTO empleados
values ('45678901D','Ana Silván','Informática',TO_DATE('25-11-2012','DD-MM-YYYY'),'INF');
INSERT INTO empleados
values ('56789012E','María Cuadrado','Ventas',TO_DATE('02-04-2013','DD-MM-YYYY'),'COM');
```

Lo ejecutamos y comprobamos el resultado.

```
SQL> @ c:\src\ins_empleados.sql
```

```
1 row created.
```

```
1 row created.
```

```
1 row created.
```

```
1 row created.
```

```
SQL> select * from empleados;
```

DNI	NOMBRE	ESPECIALIDAD	FECHAALT	DPT
12345678A	Alberto Gil	Contable	10/12/10	CON
23456789B	Mariano Sanz	Informática	04/10/11	INF
34567890C	Iván Gómez	Ventas	20/07/12	COM
45678901D	Ana Silván	Informática	25/11/12	INF
56789012E	María Cuadrado	Ventas	02/04/13	COM

### 2.1.3 Borrar el departamento de Marketing de la tabla Departamentos.

#### MySQL

```
mysql> delete from departamentos  
-> where CodDpto = 'MKT';  
Query OK, 1 row affected (0.03 sec)
```

Comprobamos el resultado, el departamento de Marketing ha sido eliminado.

```
mysql> select * from departamentos;  
+-----+-----+-----+  
| CodDpto | Nombre          | Ubicacion      |  
+-----+-----+-----+  
| ADM     | Administración  | Planta quinta U2 |  
| ALM     | Almacén         | Planta baja U1   |  
| COM     | Comercial       | Planta tercera U3 |  
| CONT    | Contabilidad    | Planta quinta U1 |  
| INF     | Informática     | Planta sótano U3 |  
+-----+-----+-----+  
5 rows in set (0.00 sec)
```

#### Oracle

```
SQL> delete from departamentos  
2 where CodDpto = 'MKT';  
  
1 row deleted.
```

Comprobamos el resultado, el departamento de Marketing ha sido eliminado.