

**NOTA:** La resolución de cada uno de los ejercicios debe ser una captura de pantalla en la que se refleje la consulta y los resultados obtenidos de ella. Debéis usar la base de datos filmoteca, que os he subido junto al boletín. El diagrama de dicha base es:



### Solución

2	SET	nacionalidad=UPPER(nacionalidad)
3	WHERE	nacionalidad='España';
4		
5	SELECT	* FROM nacionalidades
6		

nacionalidades (2x32)	
id	nacionalidad
1	Alemania
2	Estados Unidos
3	Francia
4	Gran Bretaña
5	ESPAÑA
6	Checoslovaquia
7	Italia
8	Japón

## Ejercicio 2

Obtener un listado de los directores y actores ordenándolos por orden descendente del número de películas en las que han participado. En el listado debe figurar en una columna el nombre del actor o director, en otra el número de películas, en otra si es director o autor y en la última el número de letras que contienen sus nombres (sin apellidos).

## Solución

```

1 SELECT director, COUNT(*) cuenta, 'DIRECTOR' tipo,
2 CASE
3 WHEN POSITION(' ' IN director) =0 THEN LENGTH(director)
4 ELSE POSITION(' ' IN director)-1
5 END AS 'Longitud'
6 FROM directores_peliculas
7 GROUP BY 1,3
8 UNION
9 SELECT actor, COUNT(*), 'ACTOR' tipo,
10 CASE
11 WHEN POSITION(' ' IN actor) =0 THEN LENGTH(actor)
12 ELSE POSITION(' ' IN actor)-1
13 END AS 'Longitud'
14 FROM actores_peliculas
15 GROUP BY 1,3
16 ORDER BY 2 DESC, 1
17

```

Resultado #1 (4x5.969)

director	cuenta	tipo	Longitud
Zhang Rengyi	1	ACTOR	5
Zhang Yi	1	ACTOR	5
Zhao Xuri	1	ACTOR	4
Zhu Qianqian	1	ACTOR	3
Zita Johann	1	ACTOR	4
Zoe Nathanson	1	ACTOR	3
Zofia Mrozowska	1	ACTOR	5
Zouzou	1	ACTOR	6
Zsa Zsa Gabor	1	ACTOR	3
Zygmunt Malanowicz	1	ACTOR	7

### **Ejercicio 3**

Teniendo en cuenta que para dividir se utiliza “/”, para multiplicar “\*” y recordando funciones de redondeo que hemos visto, averiguar, según los datos que tenemos en nuestra base de datos de cine, cuántas películas se hicieron en cada década mostrando número de películas y década. Si hay películas sin fecha en su lugar debe poner ‘Desconocida’.

### **Solución:**

Host: 127.0.0.1 Base de datos: filmoteca Consulta Consulta #2* x	
<pre> 1 SELECT COALESCE(FLOOR(produccion/10)*10,'DESCONOCIDA') AS DECADA, COUNT(*) 2 FROM peliculas 3 GROUP BY 1 4 ORDER BY 1 5 </pre>	
Resultado #1 (2x10)	
DECADA	COUNT(*)
1910	29
1920	63
1930	238
1940	424
1950	546
1960	467
1970	459
1980	642
1990	467
DESCONOCIDA	3

#### Ejercicio 4

Mostrar en dos columnas, **Letra** y **Nº de películas**, el número de películas que comienzan por cada letra del abecedario. Las películas que no empiecen por una letra se deben agrupar en un grupo “No letra”.

#### Solución:

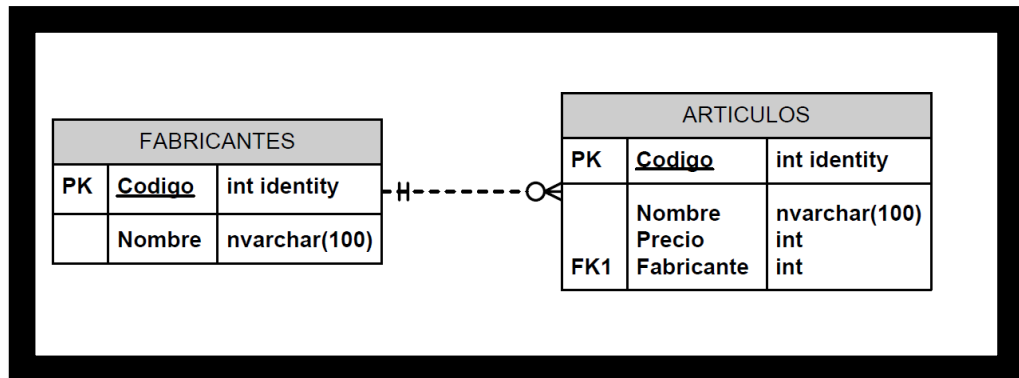
<pre> 1 CREATE VIEW titulo_letraabecedario AS 2 SELECT 3     CASE WHEN SUBSTRING(titulo,1,1) BETWEEN 'A' AND 'Z' 4         THEN SUBSTRING(titulo,1,1) 5         ELSE 'No letra' 6     END inicial 7 FROM peliculas; 8 9 SELECT inicial, COUNT(*) cuenta 10 FROM titulo_letraabecedario 11 GROUP BY 1 12 ORDER BY 1 </pre>	
Resultado #1 (2x26)	
inicial	cuenta
A	186
B	95
C	246
D	128
E	702
F	57
G	55
H	67
I	28
-	--

## EJERCICIOS BASE DE DATOS BDMONTECASTELO

**NOTA:** Para hacer estos ejercicios debemos importar el script **BDMontecastelo.sql**

## 1. La Tienda de Informática

Haz las siguientes consultas:



1.17. Añadir un nuevo artículo: Altavoces de 70 € (del fabricante 2)

```
INSERT INTO ARTICULOS( Nombre , Precio , Fabricante)
VALUES ( 'Altavoces' , 70 , 2 )
```

1.18. Cambiar el nombre del artículo 8 a 'Impresora Laser'

```
UPDATE ARTICULOS
SET Nombre = 'Impresora Laser'
WHERE Codigo = 8
```

1.19. Aplicar un descuento del 10% (multiplicar el precio por 0'9) a todos los productos.

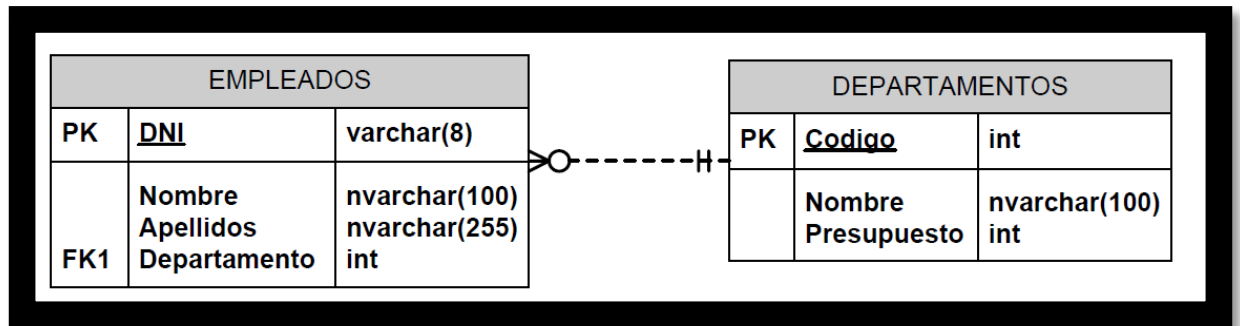
```
UPDATE ARTICULOS
SET Precio = Precio * 0.9
```

1.20. Aplicar un descuento de 10 € a todos los productos cuyo precio sea mayor o igual a 120 €

```
UPDATE ARTICULOS
SET Precio = Precio - 10
WHERE Precio >= 120
```

## 2. Empleados

Haz las siguientes consultas:



- 2.15. Añadir un nuevo departamento: 'Calidad', con presupuesto de 40.000 € y código 11. Añadir un empleado vinculado al departamento recién creado: Esther Vázquez, DNI: 89267109

```
INSERT INTO DEPARTAMENTOS
VALUES ( 11 , 'Calidad' , 40000)

INSERT INTO EMPLEADOS
VALUES ( '89267109' , 'Esther' , 'Vázquez' , 11)
```

- 2.16. Aplicar un recorte presupuestario del 10% a todos los departamentos.

```
UPDATE DEPARTAMENTOS SET Presupuesto = Presupuesto * 0.9
```

- 2.17. Reasignar a los empleados del departamento de investigación (código 77) al departamento de informática (código 14).

```
UPDATE EMPLEADOS SET Departamento = 14 WHERE Departamento = 77
```

- 2.18. Despedir a todos los empleados que trabajan para el departamento de informática (código 14).

```
DELETE FROM EMPLEADOS
WHERE Departamento = 14
```

- 2.19. Despedir a todos los empleados que trabajen para departamentos cuyo presupuesto sea superior a los 60.000 ¤.

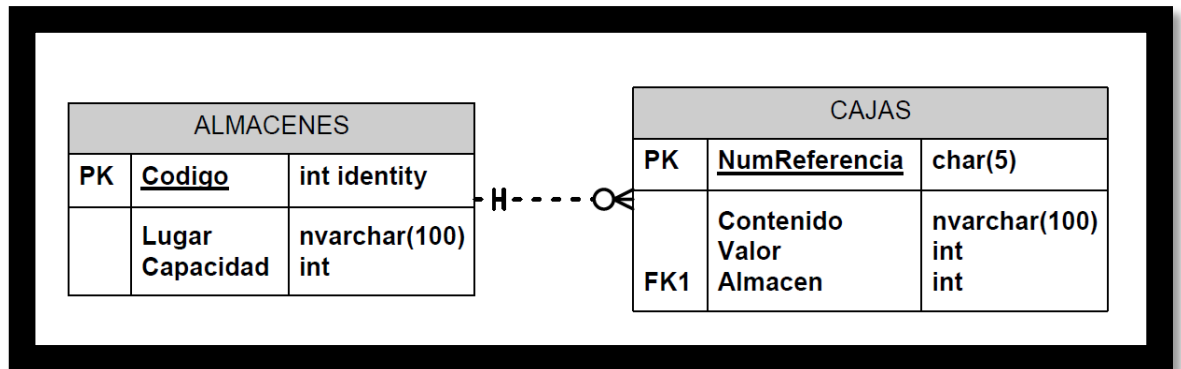
```
DELETE FROM EMPLEADOS
WHERE Departamento IN
(
    SELECT Codigo FROM DEPARTAMENTO
    WHERE Presupuesto >= 60000
)
```

2.20. Despedir a todos los empleados.

```
DELETE FROM EMPLEADOS
```

### 3. Almacenes

Haz las siguientes consultas:



3.1 Insertar un nuevo almacén en Barcelona con capacidad para 3 cajas.

```
INSERT INTO ALMACENES(Lugar,Capacidad) VALUES('Barcelona',3)
```

3.6. Insertar una nueva caja, con número de referencia 'H5RT', con contenido 'Papel', valor 200, y situada en el almacén 2.

```
INSERT INTO CAJAS  
VALUES('H5RT','Papel',200,2)
```

3.7. Rebajar el valor de todas las cajas un 15%.

```
UPDATE CAJAS SET Valor = Valor * 0.85
```

3.8. Rebajar un 20% el valor de todas las cajas cuyo valor sea superior al valor medio de todas las cajas.

```
UPDATE CAJAS SET Valor = Valor * 0.80  
WHERE Valor > (SELECT AVG(Valor) FROM CAJAS)
```

3.9. Eliminar todas las cajas cuyo valor sea inferior a 100 ¤.

```
DELETE FROM CAJAS WHERE Valor < 100
```

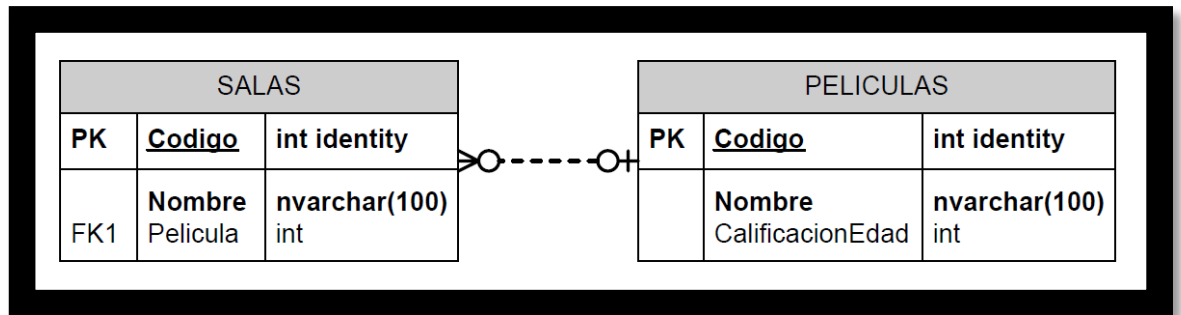
3.10. Vaciar el contenido de los almacenes que están saturados.



```
DELETE FROM CAJAS WHERE Almacen IN
(
  SELECT Codigo
  FROM ALMACENES
  WHERE Capacidad < (
    SELECT COUNT(*)
    FROM CAJAS
    WHERE Almacen = Codigo
  )
)
```

#### 4. Películas y Salas

Haz las siguientes consultas:



4.6. Añadir una nueva película 'Uno, Dos, Tres', para mayores de 7 años.

```
INSERT INTO PELICULAS(Nombre,CalificacionEdad) VALUES('Uno, Dos, Tres',7)
```

4.7. Hacer constar que todas las películas no calificadas han sido calificadas 'no recomendables para menores de 13 años'.

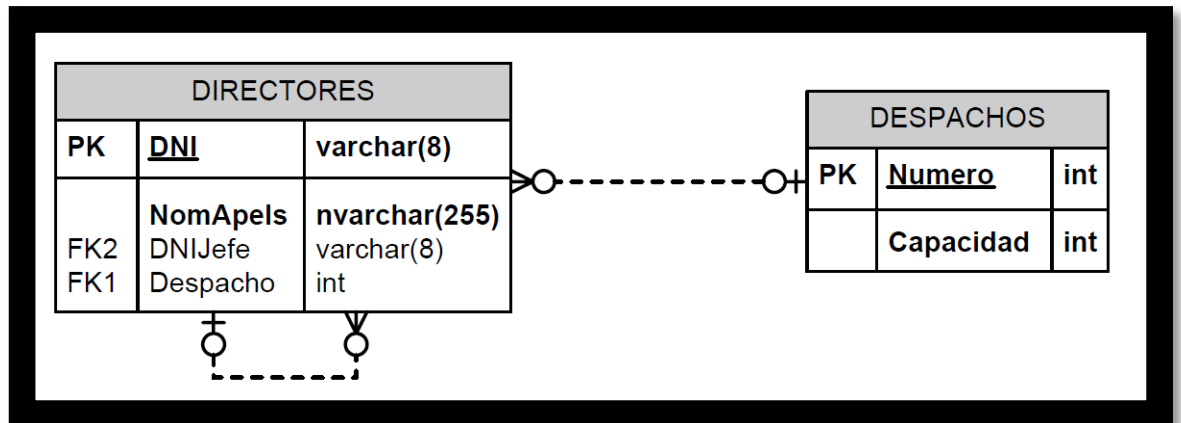
```
UPDATE PELICULAS SET CalificacionEdad=13 WHERE CalificacionEdad IS NULL
```

4.8. Eliminar todas las salas que proyectan películas recomendadas para todos los públicos.

```
DELETE FROM SALAS WHERE Pelicula IN  
(SELECT Codigo FROM PELICULAS WHERE CalificacionEdad = 0)
```

## 5. Directores

Haz las siguientes consultas:



5.5. Añadir un nuevo director llamado Paco Pérez, DNI 28301700, sin jefe, y situado en el despacho 124.

```
INSERT INTO DIRECTORES VALUES('28301700', 'Paco Pérez', NULL, 124)
```

5.6. Asignar a todos los empleados apellidados Pérez un nuevo jefe con DNI 74568521.

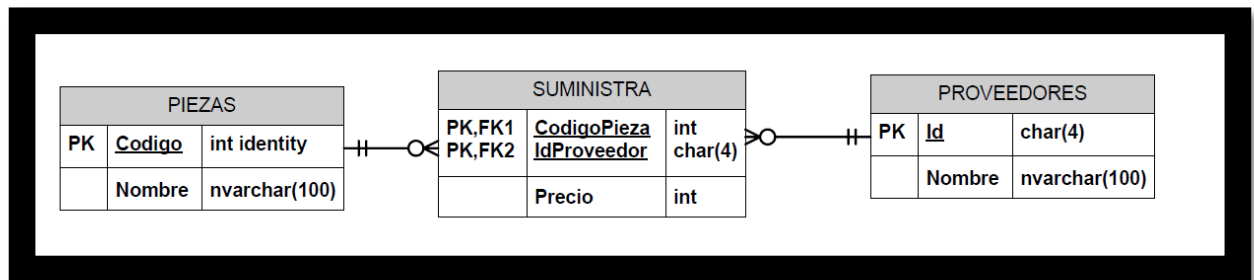
```
UPDATE DIRECTORES SET DNIJefe = '74568521' WHERE NomApels LIKE '%Pérez%'
```

5.7. Despedir a todos los directores, excepto a los que no tienen jefe.

```
DELETE FROM DIRECTORES WHERE DNIJefe IS NOT NULL
```

## 6. Piezas y Proveedores

Haz las siguientes consultas:



6.7. Hacer constar en la base de datos que la empresa "Skellington Supplies"(codigo TNBC) va a empezar a suministrarnos tuercas (código 1) a 7 pesetas cada tuerca.

```
INSERT INTO SUMINISTRA VALUES ( 'TNBC', 1, 7 )
```

6.8. Aumentar los precios en una unidad.

```
UPDATE SUMINISTRA SET Precio = Precio + 1
```

6.9. Hacer constar en la base de datos que la empresa "Susan Calvin Corp."(RBT) no va a suministrarnos ninguna pieza (aunque la empresa en sí va a seguir constando en nuestra base de datos).

```
DELETE FROM SUMINISTRA  
WHERE IdProveedor = 'RBT'
```

6.10. Hacer constar en la base de datos que la empresa "Susan Calvin Corp."(RBT) ya no va a suministrarnos clavos (código 4)

```
DELETE FROM SUMINISTRA  
WHERE IdProveedor = 'RBT'  
AND CódigoPieza = 4
```

