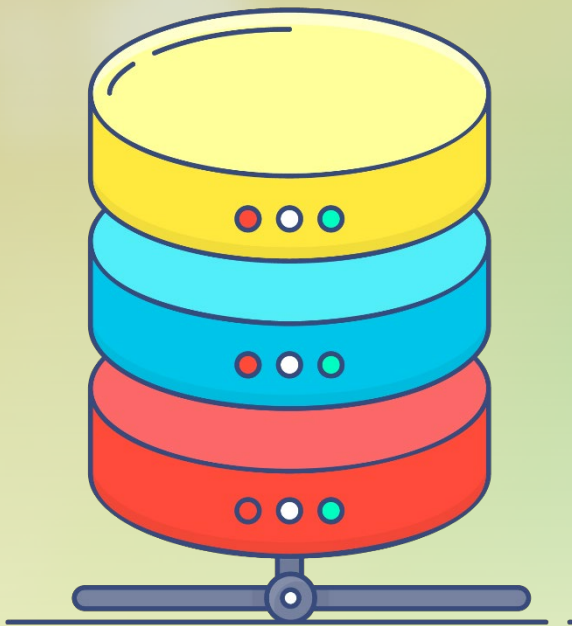


## **FUNCIONES IV: MYSQL**

---



---

**GESTIÓN DE BASES DE DATOS  
JUAN CARLOS NAVIDAD GARCÍA**

1. **Crear una función a la que se le pase el código de un artículo y me diga cuantas veces ha sido vendido. Controlar que el artículo existe.**

```
create function veces_vendido(xidprod varchar(4)) returns varchar (30) deterministic
begin
declare datos varchar(30);
declare cont int;
select count(*) into cont from productos where idprod=xidprod;
if cont >0 then
select concat(xidprod," se ha vendido ",cont," veces") into datos from productos;
else
select concat(xidprod," no se ha vendido") into datos;
end if;
return datos;
end //
```

mysql> select veces_vendido("X020") //	mysql> select veces_vendido("X000") //
+-----+   veces_vendido("X020")   +-----+	+-----+   veces_vendido("X000")   +-----+
X020 se ha vendido 1 veces	X000 no se ha vendido
+-----+	+-----+
1 row in set (0.00 sec)	1 row in set (0.00 sec)

2. **Crear una función a la que se le pase un código de cliente y me diga cuantos pedidos ha hecho. Controlar que el cliente existe.**

```
create function n_pedidos (xidcli varchar(6)) returns varchar(30) deterministic
begin
declare datos varchar(30);
declare cont int;
select count(*) into cont from clientes where idcli=xidcli;
if cont >0 then
select concat(xidcli," ha hecho ",cont," pedidos") into datos from clientes,pedidos where clientes.idcli=xidcli
and clientes.idcli=pedidos.idcli ;
else
select concat(xidcli," no existe") into datos;
end if;
return datos;
end //
```

mysql> select n_pedidos("002") //	mysql> select n_pedidos("000") //
+-----+   n_pedidos("002")   +-----+	+-----+   n_pedidos("000")   +-----+
002 ha hecho 1 pedidos	000 no existe
+-----+	+-----+
1 row in set (0.00 sec)	1 row in set (0.00 sec)

### 3. Crear una función para obtener el precio de un determinado producto si existe. En caso contrario, devolver 0.

```
create function precio (xidprod varchar(4)) returns varchar(30) deterministic
begin
declare datos varchar(30);
declare cont int;
select count(*) into cont from productos where idprod=xidprod;
if cont >0 then
select concat(xidprod," cuesta ",precio," €") into datos from productos where idprod=xidprod;
else
select concat(xidprod," = 0") into datos;
end if;
return datos;
end //
```

<pre>mysql&gt; select precio ("B001") //</pre> <table border="1"> <thead> <tr> <th>precio ("B001")</th> </tr> </thead> <tbody> <tr> <td>B001 cuesta 1.71 euros</td> </tr> </tbody> </table> <pre>1 row in set (0.00 sec)</pre>	precio ("B001")	B001 cuesta 1.71 euros	<pre>mysql&gt; select precio ("B000") //</pre> <table border="1"> <thead> <tr> <th>precio ("B000")</th> </tr> </thead> <tbody> <tr> <td>B000 = 0</td> </tr> </tbody> </table> <pre>1 row in set (0.00 sec)</pre>	precio ("B000")	B000 = 0
precio ("B001")					
B001 cuesta 1.71 euros					
precio ("B000")					
B000 = 0					

### 4. Crear una función para obtener el total de un pedido dado (la información la obtendremos de la tabla lineaspedido). Aplicarlo a todos los pedidos.

```
create function total_pedido (xidped int(11)) returns varchar(30) deterministic
begin
declare datos varchar(30);
declare xprecio decimal;
declare cont int;
select count(*) into cont from lineaspedido where idped=xidped;
if cont >0 then
select unidades*precio into xprecio from productos, lineaspedido where
lineaspedido.idped=xidped and lineaspedido.idproducto=productos.idprod;
select concat(xidped," = ", xprecio,"euros") into datos;
else
select concat(xidped," no existe") into datos;
end if;
return datos;
end //
```

```
mysql> select total_pedido (1) //
```

total_pedido (1)
1 = 3euros

```
1 row in set (0.00 sec)
```

5. Crear una función para obtener el total de pedidos gestionados por un empleado.

```
create function empleado (xidemp varchar(3)) returns varchar(30) deterministic
begin
declare datos varchar(30);
declare cont int;
select count(idemp) into cont from pedidos where idemp=xidemp;
select concat(xidemp," ha gestionado ",cont," pedidos") into datos;
return datos;
end //
```

```
mysql> select empleado("C01") //
```

empleado("C01")
C01 ha gestionado 2 pedidos

```
1 row in set (0.00 sec)
```

6. Listado de empleados, con el sueldo incrementado en 1,5€ por cada pedido que haya gestionado. Utilizar una función de apoyo.

7. Crear una función para obtener el total de caja de la empresa en un día concreto. No puede devolver NULL si ese día no hubo ventas. Debe devolver: 15 de febrero de 2022 – Total ventas: xxxxx €

```
create function ventas_dia(xdia date) returns varchar(55) deterministic
begin
declare datos varchar(55);
declare xtotal decimal;
declare cont int;
select count(xdia) into cont from pedidos where fecha=xdia;
if cont>0 then
select sum(total) into xtotal from pedidos where fecha=xdia;
select concat(xdia," – Total ventas: ",xtotal," euros") into datos;
else
select concat(xdia," – Total ventas: xxxxxx euros") into datos;
end if;
return datos;
end //
```

```
mysql> select ventas_dia('2021-01-22') //
```

```
+-----+
| ventas_dia('2021-01-22') |
+-----+
| 2021-01-22 – Total ventas: 93 euros |
+-----+
```

```
1 row in set (0.00 sec)
```

```
mysql> select ventas_dia('2023-01-22') //
```

```
+-----+
| ventas_dia('2023-01-22') |
+-----+
| 2023-01-22 – Total ventas: xxxxxx euros |
+-----+
```

```
1 row in set (0.00 sec)
```

8. Utilizando la función del punto 1, obtener las veces que se ha vendido cada producto.