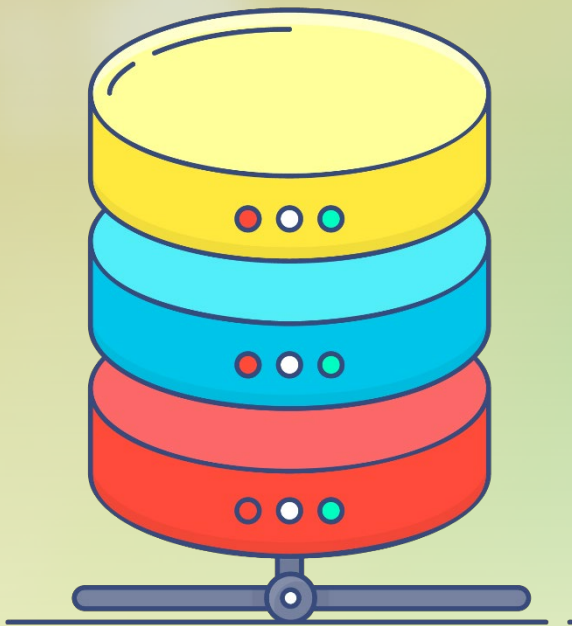


# **FUNCIONES V: MYSQL**

---



---

**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 (suma de unidades). Controlar que el artículo existe.

```
CREATE Function veces_vendido(xcodart varchar(4)) returns varchar (100) DETERMINISTIC
BEGIN
declare datos varchar(100);
declare xunidades int;
declare cont int;
select COUNT(*) into cont from ventas where codart=xcodart;
select sum(unidades) into xunidades from ventas where codart=xcodart;
if cont > 0 THEN
select distinct concat("El artículo ",descripcion," se ha vendido: ",xunidades,"
veces") into datos from artículos,ventas where artículos.codart=ventas.codart and
artículos.codart=xcodart;
ELSE
set datos=concat("El artículo ",xcodart," no existe o no se ha vendido");
END IF;
return datos;
end //
```

```
mysql> select veces_vendido('R512') //
+-----+
| veces_vendido('R512') |
+-----+
| El artículo Calcetines Unisex 38-42 P se ha vendido: 3 veces |
+-----+
1 row in set (0.00 sec)
```

```
mysql> select veces_vendido('R515') //
+-----+
| veces_vendido('R515') |
+-----+
| El artículo R515 no existe o no se ha vendido |
+-----+
1 row in set (0.00 sec)
```

2. Crear una función a la que se le pase un código de cliente y me diga la suma total de los artículos que ha comprado. Controlar que el cliente existe.

```
CREATE Function veces_comprado(xcodcli varchar(4)) returns varchar (50) DETERMINISTIC
BEGIN
declare datos varchar(50);
declare xunidades int;
declare cont int;
select COUNT(*) into cont from ventas where codcli=xcodcli;
select sum(unidades) into xunidades from ventas where codcli=xcodcli;
if cont > 0 THEN
select concat("El cliente ",xcodcli," ha comprado: ",xunidades," artículos") into
datos;
ELSE
set datos=concat("El cliente ",xcodcli," no existe o no ha comprado");
END IF;
return datos;
end //
```

```
mysql> select veces_comprado("3387") //
+-----+
| veces_comprado("3387") |
+-----+
| El cliente 3387 ha comprado: 42 artículos |
+-----+
1 row in set (0.00 sec)
```

```
mysql> select veces_comprado("0000") //
+-----+
| veces_comprado("0000") |
+-----+
| El cliente 0000 no existe o no ha comprado |
+-----+
1 row in set (0.00 sec)
```

3. Crear una función para obtener el artículo más caro que proporciona un determinado proveedor.

```
CREATE Function art_caro(xproveedor varchar(3)) returns varchar (200) DETERMINISTIC
BEGIN
declare datos varchar(200);
select concat("El producto ",descripcion," cuesta: ",PVP," Euros y es el producto más
caro del proveedor ",xproveedor) into datos from artículos where proveedor=xproveedor
order by pvp desc limit 1;
return datos;
end //
```

```
mysql> select art_caro("HEA") //
+-----+
| art_caro("HEA") |
+-----+
| El producto Raqueta Head Graphene Xt Prestige cuesta: 205.00 Euros y es el producto más caro del proveedor HEA |
+-----+
1 row in set (0.00 sec)
```

#### 4. Crear una función para obtener la suma total de los artículos vendidos en una determinada fecha.

```
CREATE Function total_fecha(xfecha date) returns varchar (100) DETERMINISTIC
BEGIN
declare datos varchar(100);
declare unidades int;
select sum(unidades) into unidades from ventas where date(fechaventa)=xfecha;
select concat("El día ",DAYNAME(xfecha)," de ",MONTHNAME(xfecha)," del
",year(xfecha)," se vendieron ",unidades," artículos") into datos;
return datos;
end //
```

```
mysql> select total_fecha('2022-09-19') //
+-----+
| total_fecha('2022-09-19') |
+-----+
| El día lunes de septiembre del 2022 se vendieron 5 artículos |
+-----+
1 row in set (0.00 sec)
```

#### 5. Crear una función para obtener el número de veces que se ha vendido cada artículo. Utilizar esta función para obtener las veces que se ha vendido cada artículo de la base de datos.

```
mysql> select veces_vendido(codart) from ventas group by codart//
+-----+
| veces_vendido(codart) |
+-----+
| El artículo Pala Padel Head Evolution se ha vendido: 1 veces |
| El artículo Raqueta Head Graphene Xt Prestige se ha vendido: 36 veces |
| El artículo Camiseta Térmica Negra L se ha vendido: 2 veces |
| El artículo Chaquetón Snow Negro XL se ha vendido: 1 veces |
| El artículo Mizuno Wave Unitus 4 se ha vendido: 7 veces |
| El artículo Zapatillas Running 40-43 se ha vendido: 1 veces |
| El artículo Zapatillas Basket 44-46 se ha vendido: 2 veces |
| El artículo Malla Gimnasio Mujer se ha vendido: 2 veces |
| El artículo Calcetines Unisex 38-42 P se ha vendido: 3 veces |
+-----+
9 rows in set (0.01 sec)
```

6. Crear una función para obtener la caja de un determinado día. Hay que tener en cuenta los descuentos aplicados.

```
CREATE Function total_caja(xfecha date) returns varchar (100) DETERMINISTIC
BEGIN
declare datos varchar(100);
declare xdescuento decimal;
select (sum(unidades))*((pvp-((descuento/100)*pvp)) into xdescuento from
ventas,artículos where date(fechaventa)=xfecha and artículos.codart=ventas.codart;
select concat("El día ",DAYNAME(xfecha)," de ",MONTHNAME(xfecha)," del
",year(xfecha)," se hicieron ",xdescuento," euros de caja") into datos;
return datos;
end //
```

```
mysql> select total_caja('2021-10-23') //
+-----+
| total_caja('2021-10-23') |
+-----+
| El día sábado de octubre del 2021 se hicieron 5166 euros de caja |
+-----+
1 row in set (0.00 sec)
```

7. Crear una función para obtener el beneficio de la tienda de un determinado día (el beneficio es el precio venta – precio compra – descuento).

```
CREATE Function beneficio(xfecha date) returns varchar (100) DETERMINISTIC
BEGIN
declare datos varchar(100);
declare xdescuento decimal;
select (sum(unidades))*((pvp-((descuento/100)*pvp))-preciocompra) into xdescuento
from ventas,artículos where date(fechaventa)=xfecha and
artículos.codart=ventas.codart;
select concat("El día ",DAYNAME(xfecha)," de ",MONTHNAME(xfecha)," del
",year(xfecha)," se hicieron ",xdescuento," euros de beneficio") into datos;
return datos;
end //
```

```
mysql> select beneficio('2021-10-23') //
+-----+
| beneficio('2021-10-23') |
+-----+
| El día sábado de octubre del 2021 se hicieron 3598 euros de beneficio |
+-----+
1 row in set (0.00 sec)
```

8. Crear una función para obtener el último artículo comprado por la tienda. Incluir precio de compra, de venta y el nombre del proveedor.

```
CREATE function ultima_compra() returns varchar (200) DETERMINISTIC
BEGIN
declare datos varchar(200);
select concat("El último artículo comprado ha sido: ",descripcion," con precio de
compra de ",preciocompra," PVP: ",pvp,". Del proveedor ",proveedor," en la fecha de
", date(fechaventa)) into datos from artículos,ventas where
artículos.codart=ventas.codart order by fechaventa desc limit 1;
return datos;
end //
```

```
mysql> select ultima_compra() //
+-----+
| ultima_compra() |
+-----+
| El último artículo comprado ha sido: Raqueta Head Graphene Xt Prestige con precio de compra de 56.00, PVP: 205.00. Del proveedor HEA en la fecha de 2022-09-19 |
+-----+
1 row in set (0.00 sec)
```