

Evaluación técnica por: Luis Abiel Bonifaz Armenta
Desarrollador de software
Repositorio GitHub: <https://github.com/Usxus/SpringBoot-Castores>
Alias: Usxus

1. CONOCIMIENTOS SQL

1) Describe el funcionamiento general de la sentencia JOIN.

JOIN se utiliza para combinar filas de dos o más tablas en una base de datos, basándose en una condición relacionada entre ellas.

2) ¿Cuáles son los tipos de JOIN y cuál es el funcionamiento de los mismos?

INNER JOIN: Devuelve solo las filas que tienen coincidencias en ambas tablas.

LEFT JOIN: Devuelve todas las filas de la tabla izquierda y solo las coincidencias de la tabla derecha.

RIGHT JOIN: Devuelve todas las filas de la tabla derecha y solo las coincidencias de la tabla izquierda.

FULL JOIN: Devuelve todas las filas de ambas tablas, sin agrupar las que no coincidan.

CROSS JOIN: Devuelve todas las combinaciones posibles de filas entre las tablas.

SELF JOIN: Une una tabla consigo misma.

3) ¿Cuál el funcionamiento general de los TRIGGER y qué propósito tienen?

Un trigger (o disparador) es un mecanismo de base de datos que se ejecuta de forma automática cuando se realiza una operación de tipo INSERT, UPDATE o DELETE sobre una tabla o vista.

4) ¿Qué es y para qué sirve un STORED PROCEDURE?

Un stored procedure (procedimiento almacenado) es un conjunto de sentencias SQL precompiladas que se almacenan y ejecutan en el servidor de la base de datos. Es una función que puede recibir parámetros, realizar operaciones lógicas, y devolver resultados o modificar datos en la base de datos.

Considerando las siguientes tablas.

Tablas.

productos	ventas
PK idProducto INT(6)	PK idVenta INT(6)
nombre VARCHAR(40)	FK1 idProducto INT(6)
precio DECIMAL(16,2)	cantidad INT(6)

Tienen los siguientes datos insertados.

idProducto	nombre	precio
1	LAPTOP	3000.00
2	PC	4000.00
3	MOUSE	100.00
4	TECLADO	150.00
5	MONITOR	2000.00
6	MICROFONO	350.00
7	AUDIFONOS	450.00

idVenta	idProducto	cantidad
1	5	8
2	1	15
3	6	13
4	6	4
5	2	3
6	5	1
7	4	5
8	2	5
9	6	2
10	1	8

Diagrama relacional.



Scripts de base de datos de conocimiento de SQL.

-- Creación de base de datos

```
create database Castores2
```

-- Usar base de datos

```
use Castores2
```

-- Creación de la tabla productos

```
CREATE TABLE productos (  
    idProducto INT IDENTITY(1,1) PRIMARY KEY,
```

```
    nombre VARCHAR(40),  
    precio DECIMAL(16,2)  
);
```

-- Registros de productos

```
INSERT INTO productos (nombre, precio)  
VALUES  
( 'LAPTO', 3000.00),  
( 'PC', 4000.00),  
( 'MOUSE', 100.00),  
( 'TECLADO', 150.00),  
( 'MONITOR', 2000.00),  
( 'MICROFONO', 350.00),  
( 'AUDIFONOS', 450.00);
```

-- Creación de la tabla ventas

```
CREATE TABLE ventas (  
    idVenta INT IDENTITY(1,1) PRIMARY KEY,  
    idProducto INT,  
    cantidad INT,  
    FOREIGN KEY (idProducto) REFERENCES productos(idProducto)  
);
```

-- Registros de ventas

```
INSERT INTO ventas (idProducto, cantidad)  
VALUES  
(5, 8),  
(1, 15),  
(6, 13),  
(6, 4),  
(2, 3),  
(5, 1),  
(4, 5),  
(2, 5),  
(6, 2),  
(1, 8);
```

5) Traer todos los productos que tengan una venta.

```
SELECT p.idProducto, v.cantidad, p.nombre, p.precio  
FROM productos p  
INNER JOIN ventas v ON p.idProducto = v.idProducto  
WHERE v.cantidad > 0;
```

6) Traer todos los productos que tengan ventas y la cantidad total de productos vendidos.

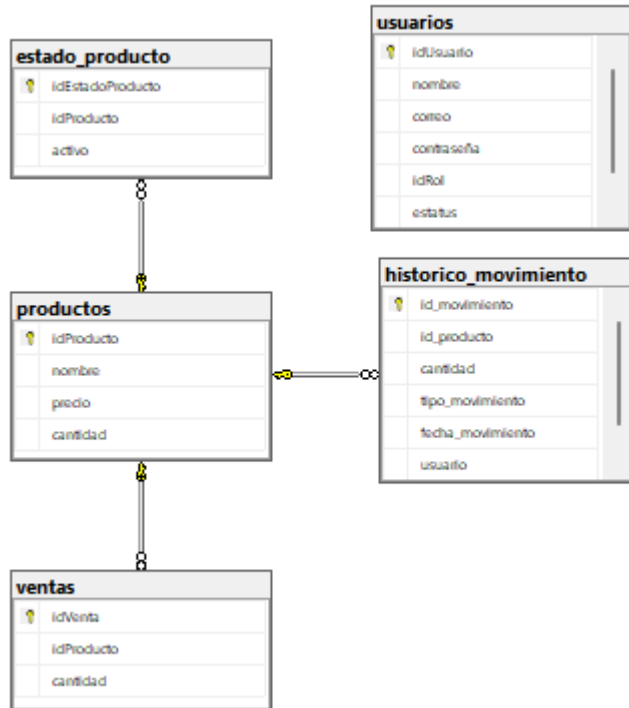
```
SELECT p.idProducto, p.nombre, SUM(v.cantidad) AS total_vendido  
FROM productos p  
INNER JOIN ventas v ON p.idProducto = v.idProducto  
GROUP BY p.idProducto, p.nombre  
HAVING SUM(v.cantidad) > 0;
```

7) Traer todos los productos (independientemente de si tienen ventas o no) y la suma total (\$) vendida por producto

```
SELECT p.idProducto, p.nombre,  
       COALESCE(SUM(v.cantidad * p.precio), 0) AS total_vendido  
FROM productos p  
LEFT JOIN ventas v ON p.idProducto = v.idProducto  
GROUP BY p.idProducto, p.nombre, p.precio;
```

2. EJERCICIO PRÁCTICO: BD

Diagrama de nuevas tablas y columnas para el nuevo escenario.



Script colocados en la carpeta llamada **SCRIPTS**, dentro del repositorio de GITHUB.
el archivo se llama [creacion_tablas.sql](#)