**TALLER PRACTICO BASES DE DATOS RELACIONALES SQL**

**PRECONDICIONES:** Para la realización de este ejercicio es necesario tener instalado el servidor XAMPP, de no tenerlo instalado en el equipo colocar el ticket a Xelerica.

**ALCANCE**

Para el desarrollo de este reto es importante estudiar acerca de las consultas básicas, ejecución de **Select, Update, Delete, Insert.**

* **OPERADORES LÓGICOS Y COMPARATIVOS:** Min, Max, And, Not, Or, Between, In, Distinct.
* **JOINS:** Joins, Inner Join, Left Join, Right Join, Full Join, Self Join
* **CLAUSULAS COMO:** Where, Like, Order By, Group By, Asc, Desc.

**Ejercicio:**

1. Realizar las lecturas del material de apoyo entregado para los diferentes temas estipulados en el alcance.

**Links De Apoyo:** [SQL SELECT Statement (w3schools.com)](https://www.w3schools.com/sql/sql_select.asp)

[SQL Tutorial - Dofactory](https://www.dofactory.com/sql/)  
**NOTA: Puedes elegir cualquiera de los dos links anteriores**

1. Después de tener instalado en el servidor XAMPP, ingresar a la URL:

<http://localhost/phpmyadmin/index.php?route=/server/databases>

**Puntos a Evaluar**

1. **\* Proceder a copiar el siguiente Script de Bases de datos:**

CREATE DATABASE tienda CHARACTER SET utf8mb4;

USE tienda;

CREATE TABLE fabricante (

codigo INT UNSIGNED AUTO\_INCREMENT PRIMARY KEY,

nombre VARCHAR(100) NOT NULL

);

CREATE TABLE producto (

codigo INT UNSIGNED AUTO\_INCREMENT PRIMARY KEY,

nombre VARCHAR(100) NOT NULL,

precio DOUBLE NOT NULL,

codigo\_fabricante INT UNSIGNED NOT NULL,

FOREIGN KEY (codigo\_fabricante) REFERENCES fabricante(codigo)

);

INSERT INTO fabricante VALUES(1, 'Asus');

INSERT INTO fabricante VALUES(2, 'Lenovo');

INSERT INTO fabricante VALUES(3, 'Hewlett-Packard');

INSERT INTO fabricante VALUES(4, 'Samsung');

INSERT INTO fabricante VALUES(5, 'Seagate');

INSERT INTO fabricante VALUES(6, 'Crucial');

INSERT INTO fabricante VALUES(7, 'Gigabyte');

INSERT INTO fabricante VALUES(8, 'Huawei');

INSERT INTO fabricante VALUES(9, 'Xiaomi');

INSERT INTO producto VALUES(1, 'Disco duro SATA3 1TB', 86.99, 5);

INSERT INTO producto VALUES(2, 'Memoria RAM DDR4 8GB', 120, 6);

INSERT INTO producto VALUES(3, 'Disco SSD 1 TB', 150.99, 4);

INSERT INTO producto VALUES(4, 'GeForce GTX 1050Ti', 185, 7);

INSERT INTO producto VALUES(5, 'GeForce GTX 1080 Xtreme', 755, 6);

INSERT INTO producto VALUES(6, 'Monitor 24 LED Full HD', 202, 1);

INSERT INTO producto VALUES(7, 'Monitor 27 LED Full HD', 245.99, 1);

INSERT INTO producto VALUES(8, 'Portátil Yoga 520', 559, 2);

INSERT INTO producto VALUES(9, 'Portátil Ideapd 320', 444, 2);

INSERT INTO producto VALUES(10, 'Impresora HP Deskjet 3720', 59.99, 3);

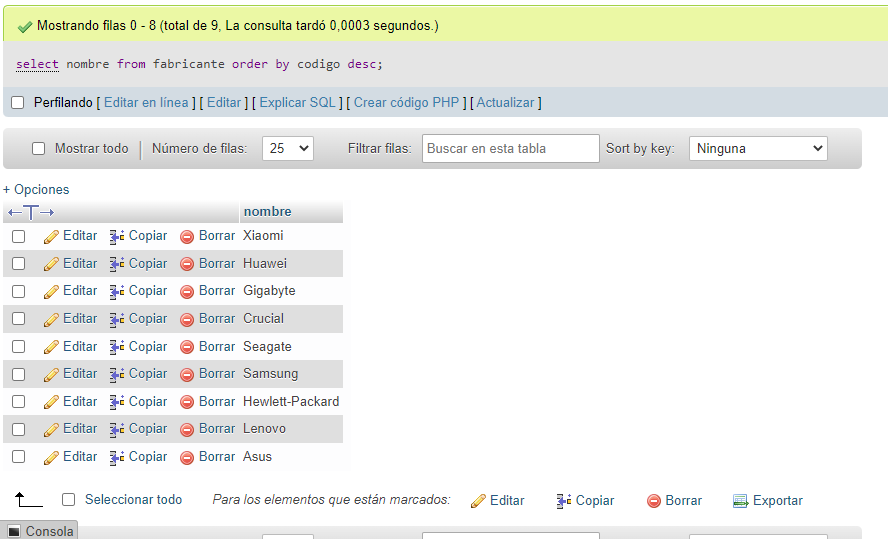
INSERT INTO producto VALUES(11, 'Impresora HP Laserjet Pro M26nw', 180, 3);

1. De Acuerdo a la Base de Datos creada anteriormente, llamada ‘Tienda’, que contiene dos tablas llamadas “Fabricante” y “Producto”. En ellas se encuentran registrados de los datos generales de cada producto de esta tienda con relación a cada uno de sus fabricantes.

**Realizar las siguientes consultas de una sola tabla:**

1. Lista los nombres de los fabricantes ordenados de forma descendente.

-select nombre from fabricante order by codigo desc;



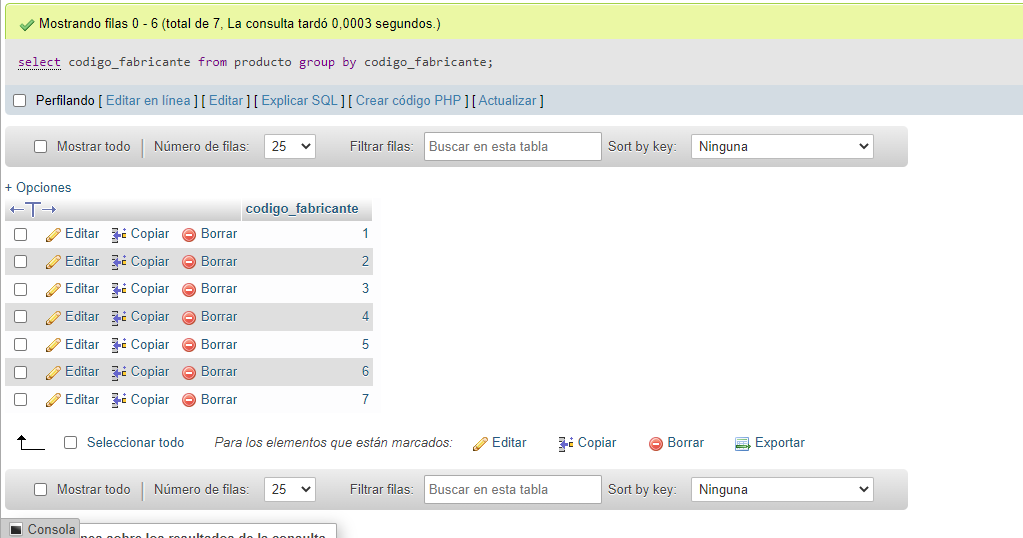
1. Lista los nombres de los productos ordenados de manera Ascendente.

-select nombre from producto order by nombre asc;



1. Lista el código de los fabricantes que tienen productos en la tabla producto, mostrando los códigos sin repetición.

-select codigo\_fabricante from producto group by codigo\_fabricante;



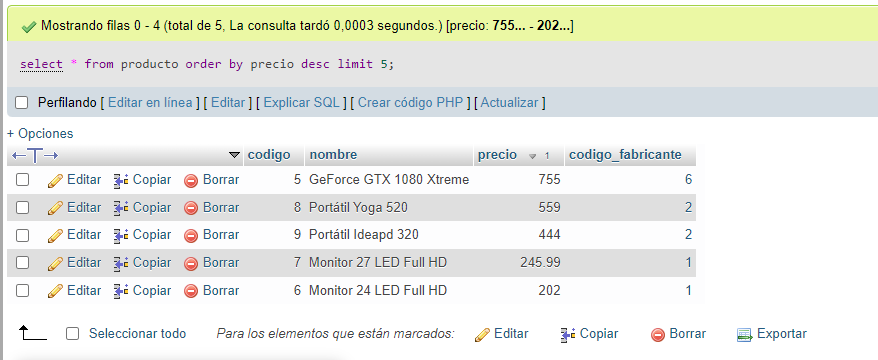
1. Traer los 3 productos que tienen menor precio.

-select \* from producto order by precio asc limit 3;



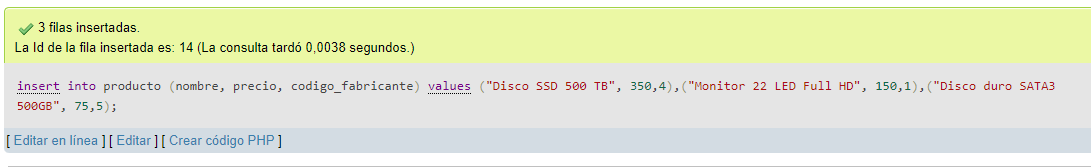
1. Traer los cinco productos de mayor precio.

-select \* from producto order by precio desc limit 5;



1. Realizar la inserción de tres nuevos productos.

-insert into producto (nombre, precio, codigo\_fabricante) values ("Disco SSD 500 TB", 350,4),("Monitor 22 LED Full HD", 150,1),("Disco duro SATA3 500GB", 75,5)





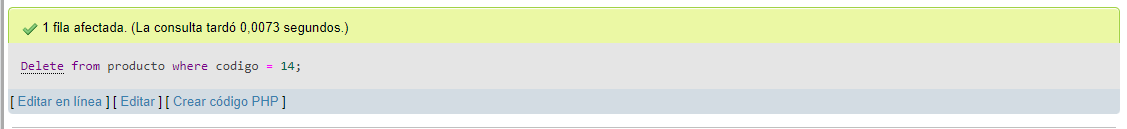
1. Consultar los productos que sea impresora HP.

-select \* from producto where nombre like "%impresora%";



1. Eliminar uno de los nuevos productos creados.

-Delete from producto where codigo = 14



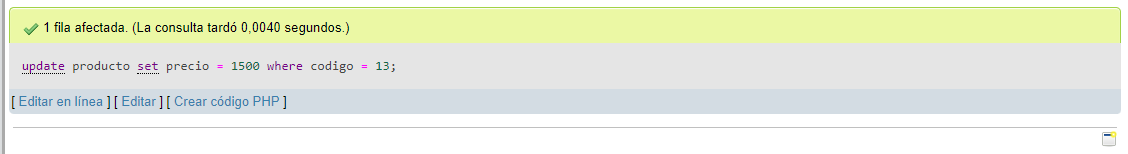


1. Actualizar el precio de uno de los productos creados.

-update producto

set precio = 1500

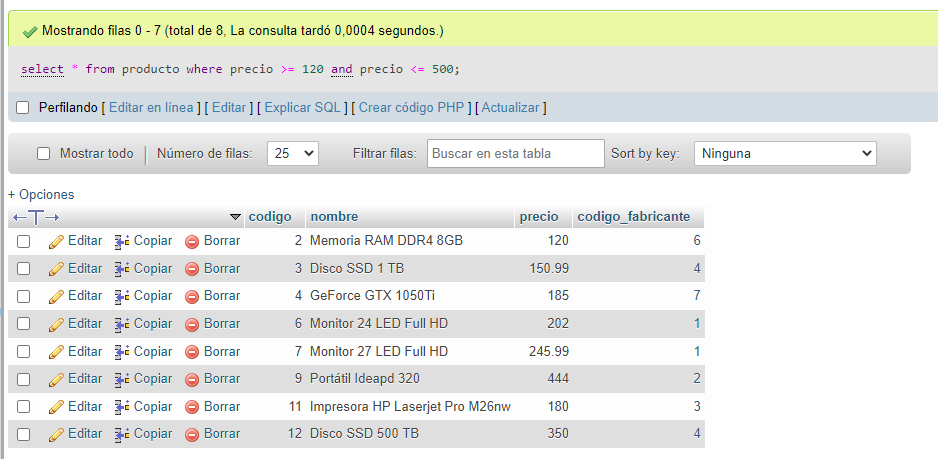
where codigo = 13





1. Traer los productos que esten dentro del rango de precios 120 y 500.

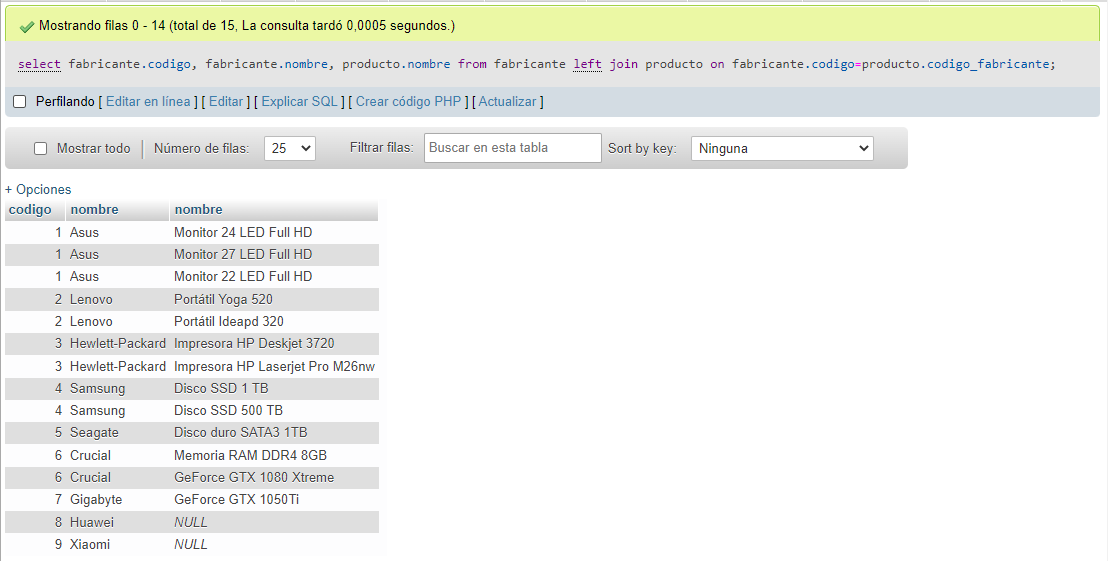
-select \* from producto where precio >= 120 and precio <= 500;



**Realiza las siguientes consultas mutitablas:**

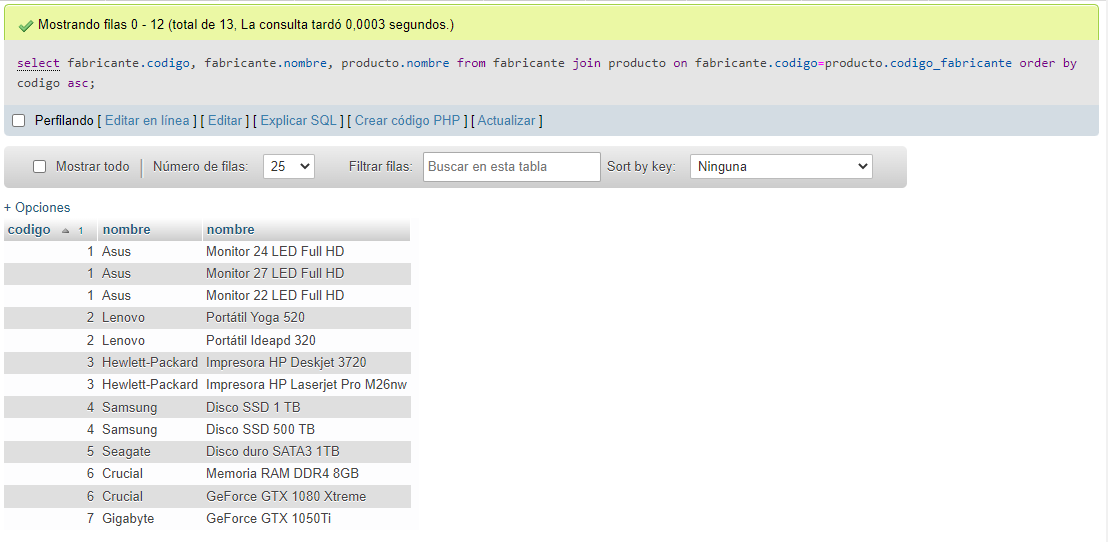
1. Consultar el nombre de cada fabricante con los productos asociados a ellos incluso mostrando los fabricantes que no tienen algún producto asignado.

-select fabricante.codigo, fabricante.nombre, producto.nombre from fabricante left join producto on fabricante.codigo=producto.codigo\_fabricante;



1. Consultar cada producto con el nombre de cada fabricante.

select fabricante.codigo, fabricante.nombre, producto.nombre from fabricante join producto on fabricante.codigo=producto.codigo\_fabricante order by codigo asc;



1. Traer los productos asociados cuyo fabricante es “Crucial” con su respectivo nombre.

-select fabricante.codigo, fabricante.nombre, producto.nombre from fabricante join producto on fabricante.codigo=producto.codigo\_fabricante where fabricante.nombre like "%crucial%" order by codigo asc;



**Recuerda, que se realizara la respectiva sustentación de este reto.  
Mucha Suerte!!**

**Control de Cambios**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Versión** | **Fecha** | **Cambios u Observaciones** | **Clasificación** | **Responsable** |
| 1.1 | 15/07/2022 | Se agrega el campo de clasificación al documento y se ajusta su píe de página. | Se asigna según la información que contenga.  (Confidencial, Restringido, Interno, Público) | Analista Creativo |
|  |  |  |  |  |