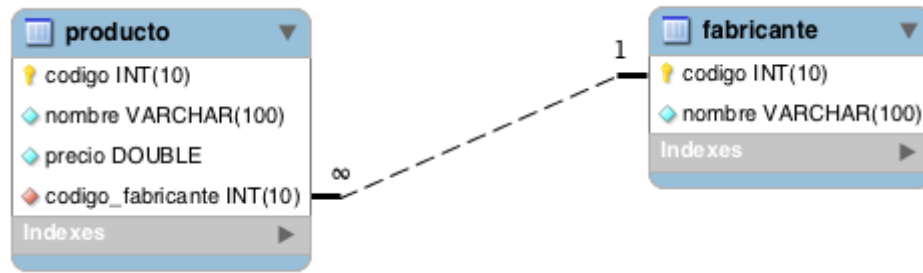


## Tienda de informática



Base de datos para MySQL

```
DROP DATABASE IF EXISTS tienda;  
CREATE DATABASE tienda CHARACTER SET utf8mb4;  
USE tienda;
```

```
CREATE TABLE fabricante (  
  id INT UNSIGNED AUTO_INCREMENT PRIMARY KEY,  
  nombre VARCHAR(100) NOT NULL  
);
```

```
CREATE TABLE producto (  
  id INT UNSIGNED AUTO_INCREMENT PRIMARY KEY,  
  nombre VARCHAR(100) NOT NULL,  
  precio DOUBLE NOT NULL,  
  id_fabricante INT UNSIGNED NOT NULL,  
  FOREIGN KEY (id_fabricante) REFERENCES fabricante(id)  
);
```

```
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);
```

### **1.1.7 Subconsultas (En la cláusula WHERE)**

#### **1.1.7.1 Con operadores básicos de comparación**

1. Devuelve todos los productos del fabricante Lenovo. (Sin utilizar INNER JOIN).
2. Devuelve todos los datos de los productos que tienen el mismo precio que el producto más caro del fabricante Lenovo. (Sin utilizar INNER JOIN).
3. Lista el nombre del producto más caro del fabricante Lenovo.
4. Lista el nombre del producto más barato del fabricante Hewlett-Packard.
5. Devuelve todos los productos de la base de datos que tienen un precio mayor o igual al producto más caro del fabricante Lenovo.
6. Lista todos los productos del fabricante Asus que tienen un precio superior al precio medio de todos sus productos.

#### **1.1.7.2 Subconsultas con ALL y ANY**

8. Devuelve el producto más caro que existe en la tabla producto sin hacer uso de MAX, ORDER BY ni LIMIT.
9. Devuelve el producto más barato que existe en la tabla producto sin hacer uso de MIN, ORDER BY ni LIMIT.
10. Devuelve los nombres de los fabricantes que tienen productos asociados. (Utilizando ALL o ANY).
11. Devuelve los nombres de los fabricantes que no tienen productos asociados. (Utilizando ALL o ANY).

#### **1.1.7.3 Subconsultas con IN y NOT IN**

12. Devuelve los nombres de los fabricantes que tienen productos asociados. (Utilizando IN o NOT IN).
13. Devuelve los nombres de los fabricantes que no tienen productos asociados. (Utilizando IN o NOT IN).

#### **1.1.7.4 Subconsultas con EXISTS y NOT EXISTS**

14. Devuelve los nombres de los fabricantes que tienen productos asociados. (Utilizando EXISTS o NOT EXISTS).
15. Devuelve los nombres de los fabricantes que no tienen productos asociados. (Utilizando EXISTS o NOT EXISTS).

#### **1.1.7.5 Subconsultas correlacionadas**

16. Lista el nombre de cada fabricante con el nombre y el precio de su producto más caro.
17. Devuelve un listado de todos los productos que tienen un precio mayor o igual a la media de todos los productos de su mismo fabricante.
18. Lista el nombre del producto más caro del fabricante Lenovo.

### **1.1.8 Subconsultas (En la cláusula HAVING)**

19. Devuelve un listado con todos los nombres de los fabricantes que tienen el mismo número de productos que el fabricante Lenovo.