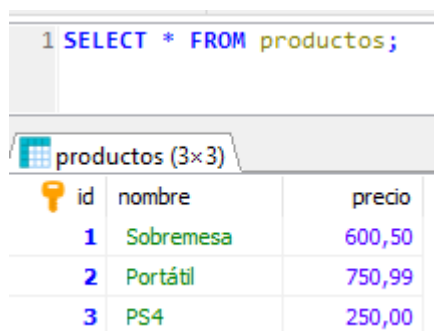


Creamos la siguiente base de datos

```
CREATE DATABASE tien21;
USE tien21;

CREATE TABLE productos (
    id INT PRIMARY KEY AUTO_INCREMENT,
    nombre VARCHAR(20),
    precio NUMERIC(5,2)
);
INSERT INTO productos (nombre, precio)
VALUES
('Sobremesa', 600.50),
('Portátil', 750.99),
('PS4', 250.00);
```



The screenshot shows a SQL query editor with the command `1 SELECT * FROM productos;` and a result window titled `productos (3x3)`. The result is a table with 3 rows and 3 columns: `id`, `nombre`, and `precio`. The first row has `1`, `Sobremesa`, and `600,50`. The second row has `2`, `Portátil`, and `750,99`. The third row has `3`, `PS4`, and `250,00`. The `id` column is marked as the primary key.

id	nombre	precio
1	Sobremesa	600,50
2	Portátil	750,99
3	PS4	250,00

1. Crear dos funciones:
 - a. Devuelve el id del producto más caro
 - b. Devuelve el precio con IVA de un producto, dado su id
2. Crear los procedimientos que :
 - a. Aplique un descuento a un producto dado un % y el identificador del producto, haz que ese descuento se refleje en la base de datos.
 - b. Inserte un producto nuevo, dados su nombre y precio
 - c. Elimine un producto dado su id
 - d. Aplica el IVA tomando el precio como base a todos los productos en la tabla (usar cursores)
3. Crear los trigger:
 - a. Uno que ponga directamente el precio con IVA sin introducirlo en el insert
 - b. Otro para la actualización de precios, que aplique el precio con IVA sobre el precio final nuevo