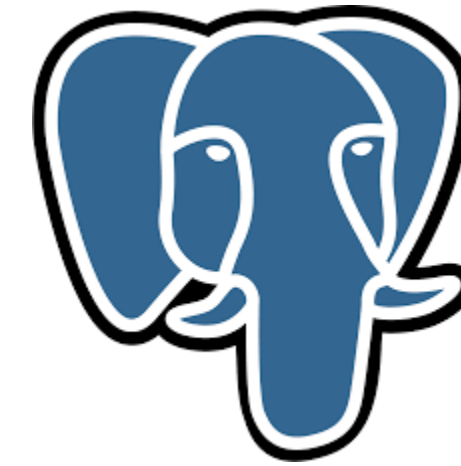
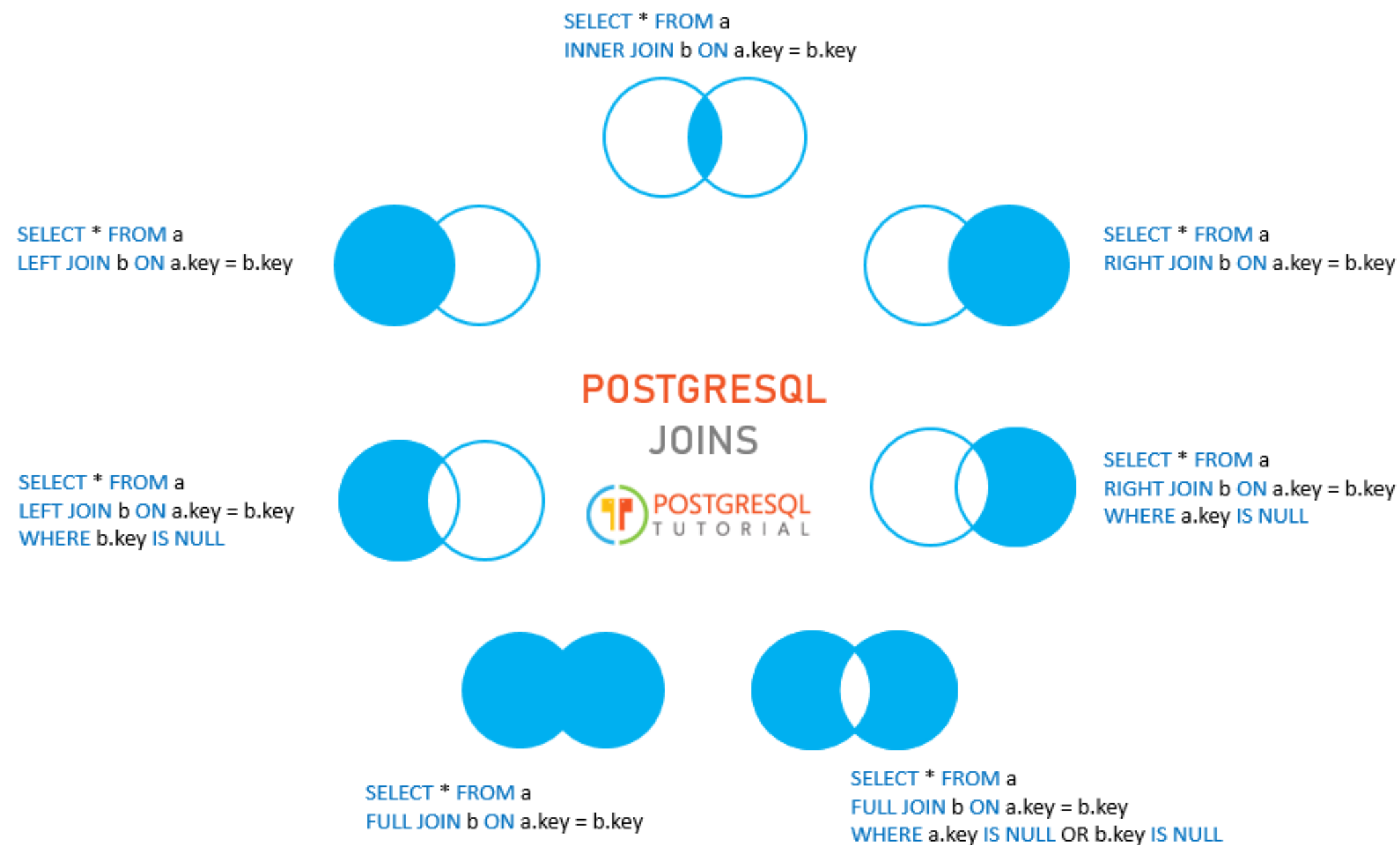


JOINS

PostgreSQL



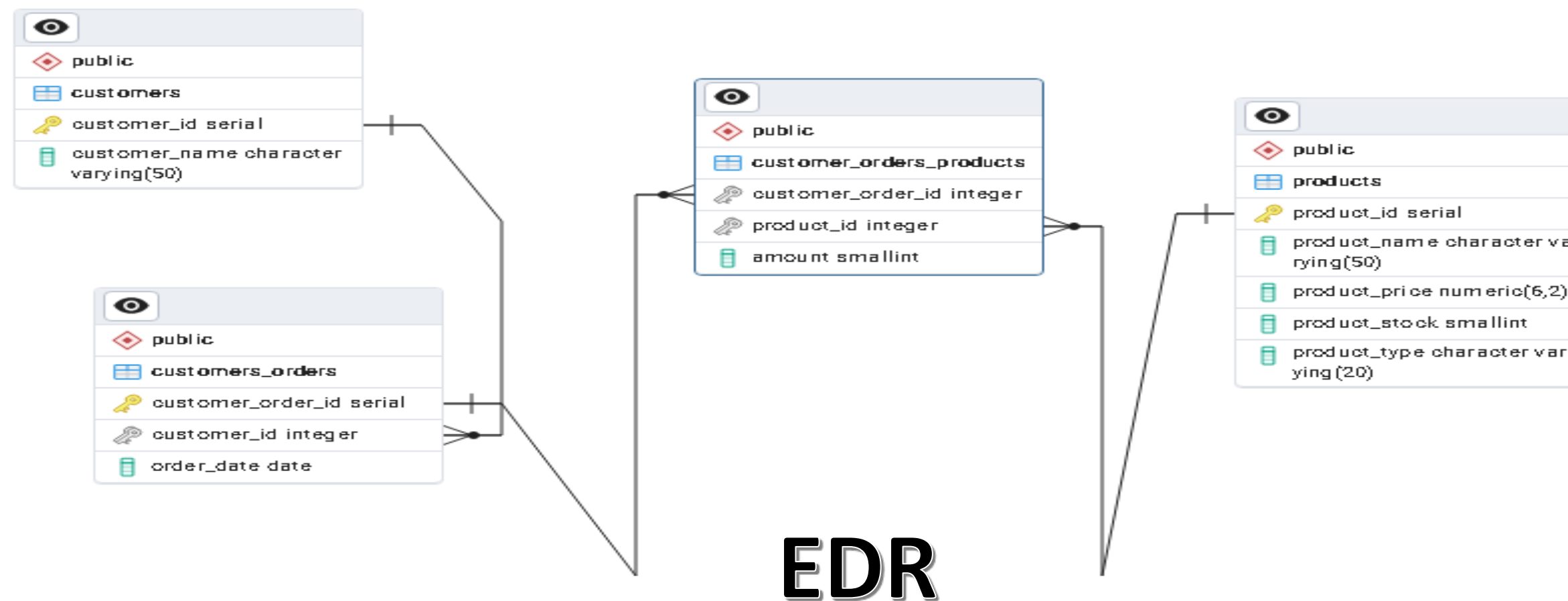
Los JOIN permiten combinar filas de dos o más tablas basadas en una relación entre ellas. Esta relación se define mediante claves primarias PK y externas FK. PostgreSQL ofrece varios tipos de JOIN, cada uno con un propósito específico.



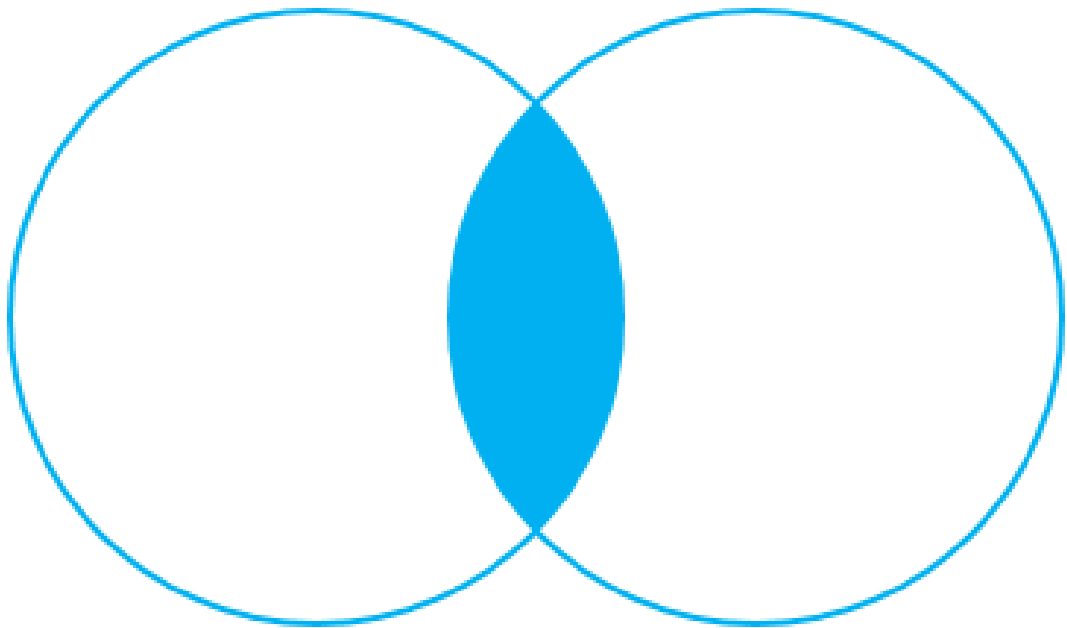
Creacion de BD y modelo EDR generado

```
CREATE TABLE customers (  
    customer_id SERIAL PRIMARY KEY,  
    customer_name VARCHAR(50)  
);  
  
CREATE TABLE products (  
    product_id SERIAL PRIMARY KEY,  
    product_name VARCHAR(50),  
    product_price NUMERIC(6,2) CHECK(product_price >= 0),  
    product_stock SMALLINT CHECK(product_stock >= 0) DEFAULT 0,  
    product_type VARCHAR(20)  
);
```

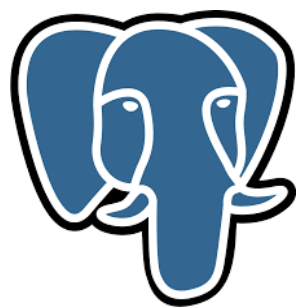
```
CREATE TABLE customers_orders(  
    customer_order_id SERIAL PRIMARY KEY,  
    customer_id INT REFERENCES  
customers(customer_id),  
    order_date DATE DEFAULT NOW()  
);  
  
CREATE TABLE customer_orders_products (  
    customer_order_id INT REFERENCES  
customers_orders(customer_order_id),  
    product_id INT REFERENCES products(product_id),  
    amount SMALLINT DEFAULT 1  
);
```



INNER JOIN: Devuelve las filas que tienen coincidencias en ambas tablas según el criterio de unión especificado. Es el tipo de JOIN más comúnmente utilizado



INNER JOIN



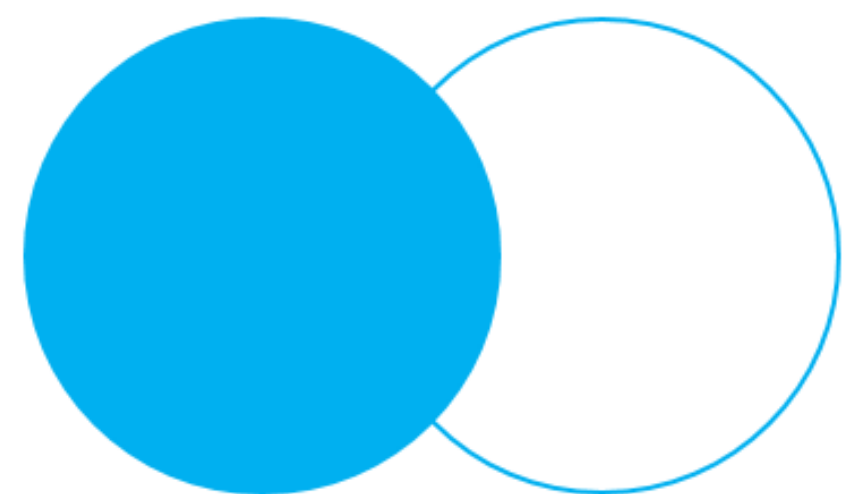
```
SELECT
  p.product_name "Nombre Producto",
  cop.amount "Cantidad",
  cop.amount * p.product_price "Costo por producto a pagar"
FROM products p
INNER JOIN customer_orders_products cop
ON p.product_id = cop.product_id;
```

	Nombre Producto character varying (50) 🔒	Cantidad smallint 🔒	Costo por producto a pagar numeric 🔒
1	Flat White	2	5.00
2	Flat White	2	5.00
3	Espresso	2	5.00
4	Espresso	2	5.00

```
SELECT
  SUM(cop.amount * p.product_price) "Total a pagar"
FROM products p
INNER JOIN customer_orders_products cop
ON p.product_id = cop.product_id;
```

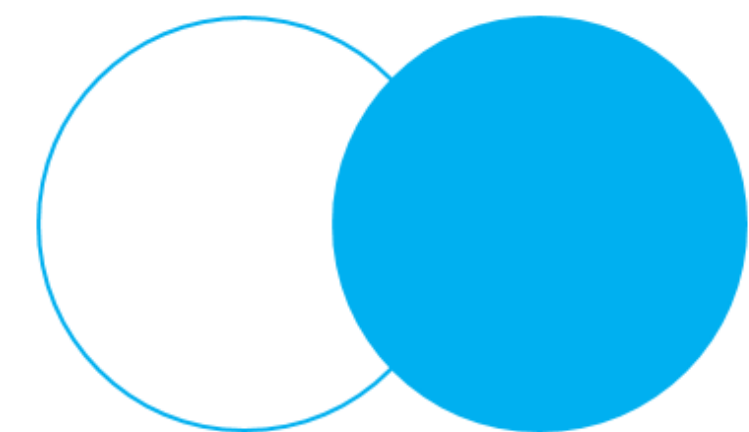
	Total a pagar numeric 🔒
1	20.00

LEFT JOIN (LEFT OUTER JOIN): Devuelve todas las filas de la tabla izquierda (primera tabla mencionada) y las filas coincidentes de la tabla derecha (segunda tabla mencionada). Si no hay coincidencias en la tabla derecha, PostgreSQL devuelve NULL.



LEFT JOIN

RIGHT JOIN (RIGHT OUTER JOIN): Similar al LEFT JOIN, pero devuelve todas las filas de la tabla derecha y las filas coincidentes de la tabla izquierda. Las filas de la tabla izquierda que no tienen coincidencias en la tabla derecha aparecen como NULL.



RIGHT JOIN



```
SELECT p.product_name
FROM products p
LEFT JOIN customer_orders_products cop
ON p.product_id = cop.product_id
```

	product_name character varying (50)
1	Flat White
2	Flat White
3	Espresso
4	Espresso
5	Scone de 4 quesos

```
SELECT p.product_name
FROM products p
RIGHT JOIN customer_orders_products cop
ON p.product_id = cop.product_id
```

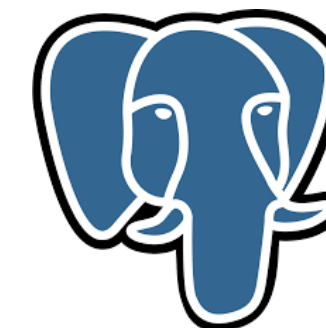
	product_name character varying (50)
1	Flat White
2	Flat White
3	Espresso
4	Espresso

JOIN entre varias tablas



```
SELECT
  c.customer_name "Cliente",
  co.order_date "Fecha",
  cop.amount "Cantidad",
  p.product_name "Producto",
  cop.amount * p.product_price "Costos productos"

FROM customers c
JOIN customers_orders co
ON c.customer_id = co.customer_id
JOIN customer_orders_products cop
ON co.customer_order_id = cop.customer_order_id
JOIN products p
ON p.product_id = cop.product_id;
```



	Cliente character varying (50) 🔒	Fecha date 🔒	Cantidad smallint 🔒	Producto character varying (50) 🔒	Costos productos numeric 🔒
1	Josefa	2024-02-20	2	Flat White	5.00
2	Josefa	2024-02-20	2	Flat White	5.00
3	Josefa	2024-02-20	2	Espresso	5.00
4	Josefa	2024-02-20	2	Espresso	5.00

