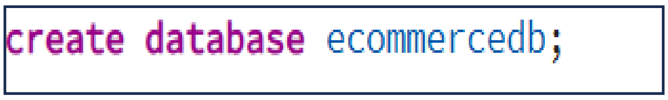
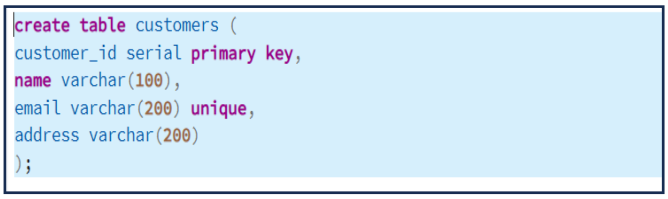
Task – 4 SQL for Data Analysis

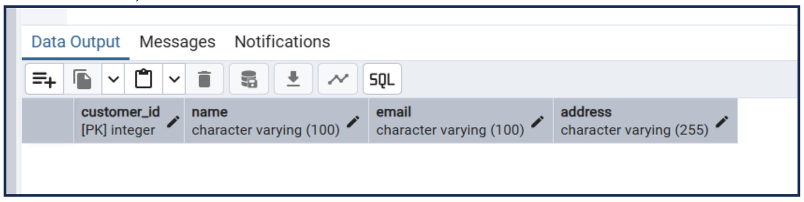
OUTPUT



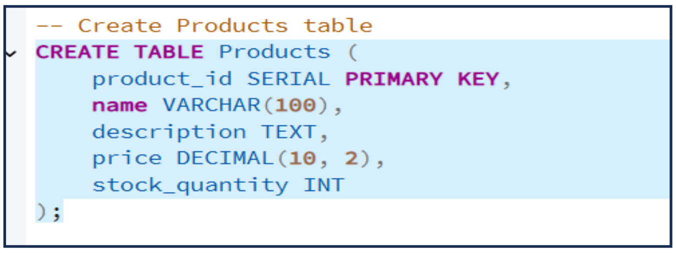
```sql
create database ecommercedb;
```



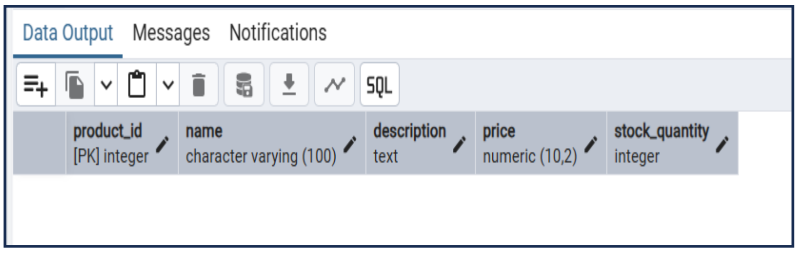
```sql
create table customers (
customer_id serial primary key,
name varchar(100),
email varchar(200) unique,
address varchar(200)
);
```



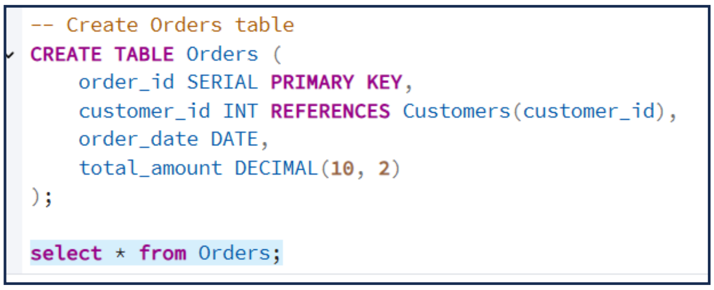
customer_id [PK] integer	name character varying (100)	email character varying (100)	address character varying (255)



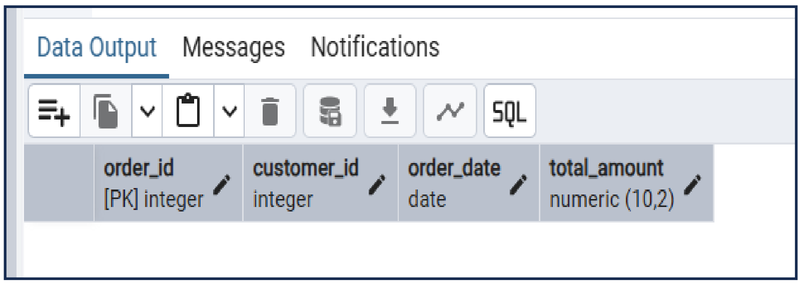
```sql
-- Create Products table
CREATE TABLE Products (
    product_id SERIAL PRIMARY KEY,
    name VARCHAR(100),
    description TEXT,
    price DECIMAL(10, 2),
    stock_quantity INT
);
```



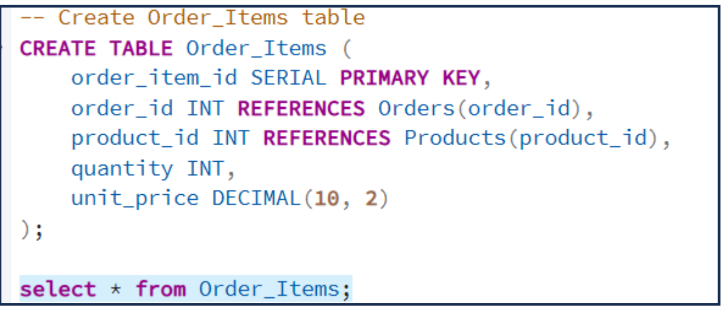
product_id [PK] integer	name character varying (100)	description text	price numeric (10,2)	stock_quantity integer



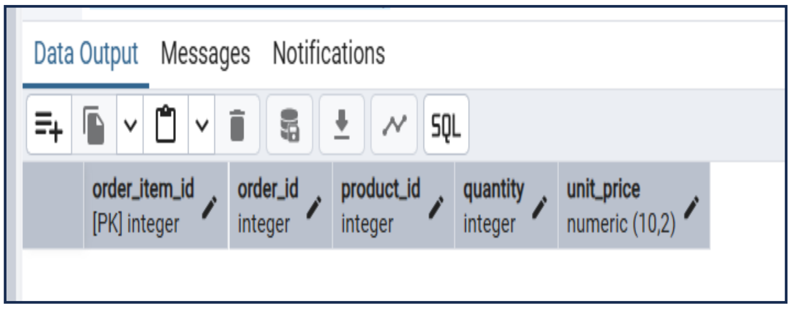
```sql
-- Create Orders table
CREATE TABLE Orders (
    order_id SERIAL PRIMARY KEY,
    customer_id INT REFERENCES Customers(customer_id),
    order_date DATE,
    total_amount DECIMAL(10, 2)
);

select * from Orders;
```



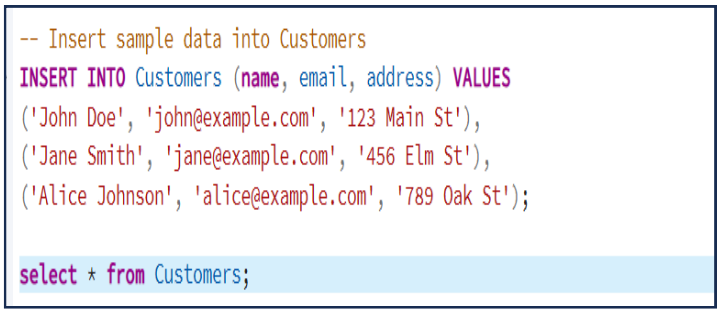
order_id [PK] integer	customer_id integer	order_date date	total_amount numeric (10,2)



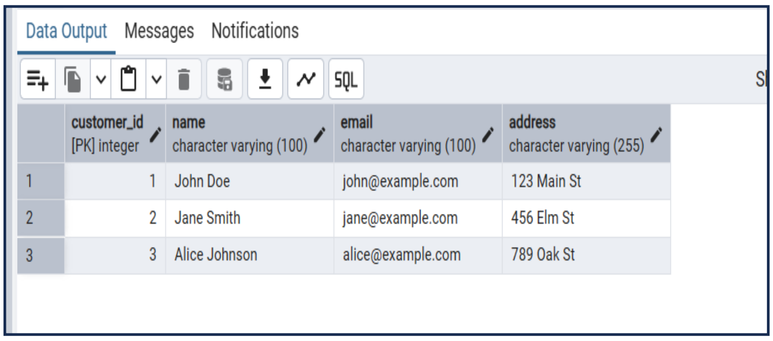
```sql
-- Create Order_Items table
CREATE TABLE Order_Items (
    order_item_id SERIAL PRIMARY KEY,
    order_id INT REFERENCES Orders(order_id),
    product_id INT REFERENCES Products(product_id),
    quantity INT,
    unit_price DECIMAL(10, 2)
);

select * from Order_Items;
```



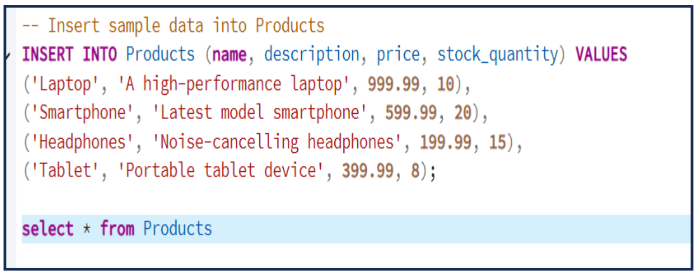
order_item_id [PK] integer	order_id integer	product_id integer	quantity integer	unit_price numeric (10,2)



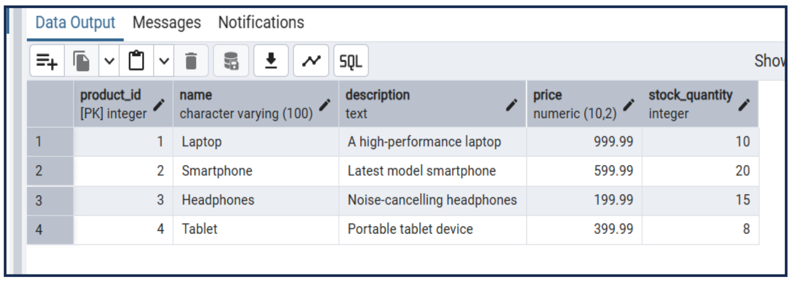
```sql
-- Insert sample data into Customers
INSERT INTO Customers (name, email, address) VALUES
('John Doe', 'john@example.com', '123 Main St'),
('Jane Smith', 'jane@example.com', '456 Elm St'),
('Alice Johnson', 'alice@example.com', '789 Oak St');

select * from Customers;
```



	customer_id [PK] integer	name character varying (100)	email character varying (100)	address character varying (255)
1	1	John Doe	john@example.com	123 Main St
2	2	Jane Smith	jane@example.com	456 Elm St
3	3	Alice Johnson	alice@example.com	789 Oak St

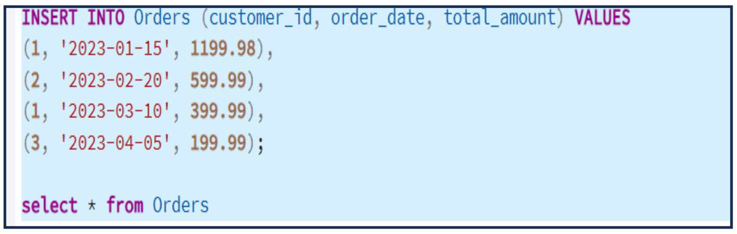


```sql
-- Insert sample data into Products
INSERT INTO Products (name, description, price, stock_quantity) VALUES
('Laptop', 'A high-performance laptop', 999.99, 10),
('Smartphone', 'Latest model smartphone', 599.99, 20),
('Headphones', 'Noise-cancelling headphones', 199.99, 15),
('Tablet', 'Portable tablet device', 399.99, 8);

select * from Products
```

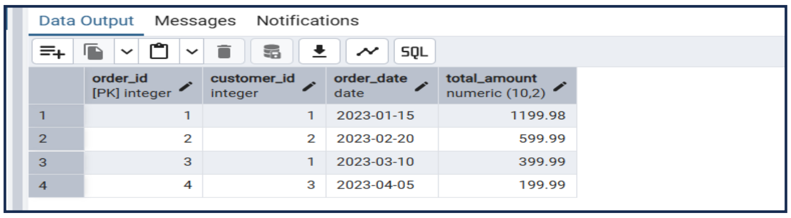


	product_id [PK] integer	name character varying (100)	description text	price numeric (10,2)	stock_quantity integer
1	1	Laptop	A high-performance laptop	999.99	10
2	2	Smartphone	Latest model smartphone	599.99	20
3	3	Headphones	Noise-cancelling headphones	199.99	15
4	4	Tablet	Portable tablet device	399.99	8

OUTPUT

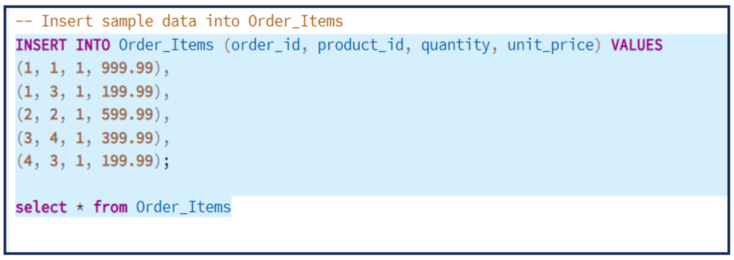


```sql
INSERT INTO Orders (customer_id, order_date, total_amount) VALUES
(1, '2023-01-15', 1199.98),
(2, '2023-02-20', 599.99),
(1, '2023-03-10', 399.99),
(3, '2023-04-05', 199.99);

select * from Orders
```

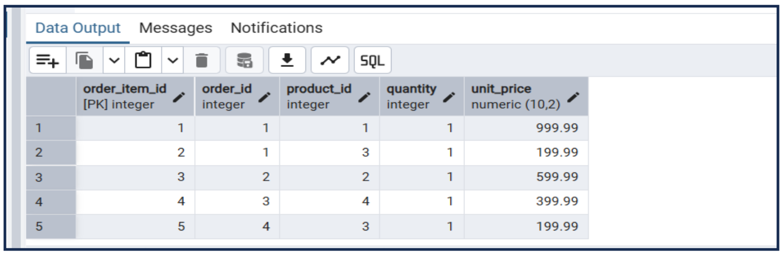


Data Output Messages Notifications

	order_id [PK] integer	customer_id integer	order_date date	total_amount numeric (10,2)
1	1	1	2023-01-15	1199.98
2	2	2	2023-02-20	599.99
3	3	1	2023-03-10	399.99
4	4	3	2023-04-05	199.99

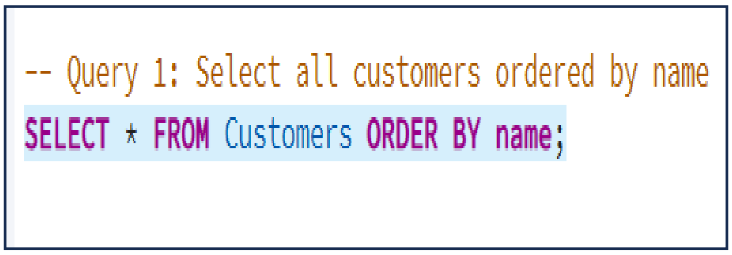


```sql
-- Insert sample data into Order_Items
INSERT INTO Order_Items (order_id, product_id, quantity, unit_price) VALUES
(1, 1, 1, 999.99),
(1, 3, 1, 199.99),
(2, 2, 1, 599.99),
(3, 4, 1, 399.99),
(4, 3, 1, 199.99);

select * from Order_Items
```

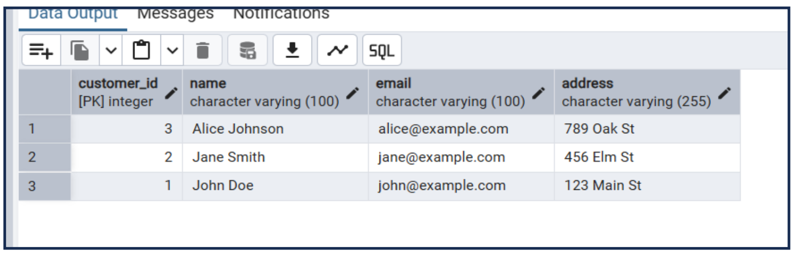


Data Output Messages Notifications

	order_item_id [PK] integer	order_id integer	product_id integer	quantity integer	unit_price numeric (10,2)
1	1	1	1	1	999.99
2	2	1	3	1	199.99
3	3	2	2	1	599.99
4	4	3	4	1	399.99
5	5	4	3	1	199.99

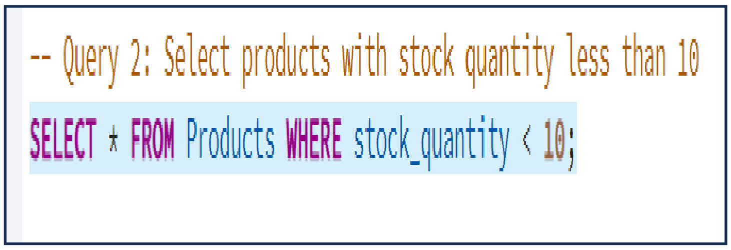


```sql
-- Query 1: Select all customers ordered by name
SELECT * FROM Customers ORDER BY name;
```

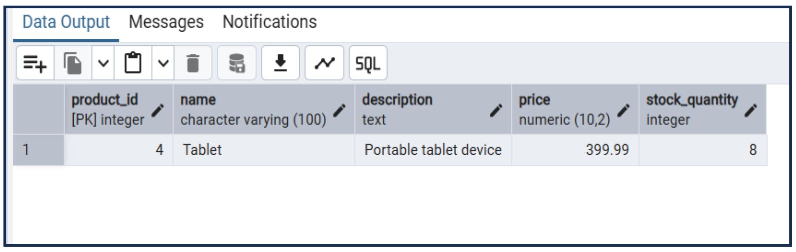


Data Output Messages Notifications

	customer_id [PK] integer	name character varying (100)	email character varying (100)	address character varying (255)
1	3	Alice Johnson	alice@example.com	789 Oak St
2	2	Jane Smith	jane@example.com	456 Elm St
3	1	John Doe	john@example.com	123 Main St

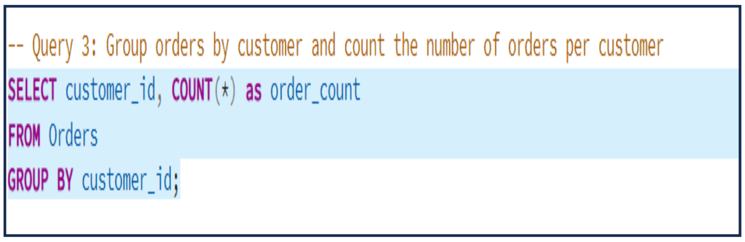


```sql
-- Query 2: Select products with stock quantity less than 10
SELECT * FROM Products WHERE stock_quantity < 10;
```

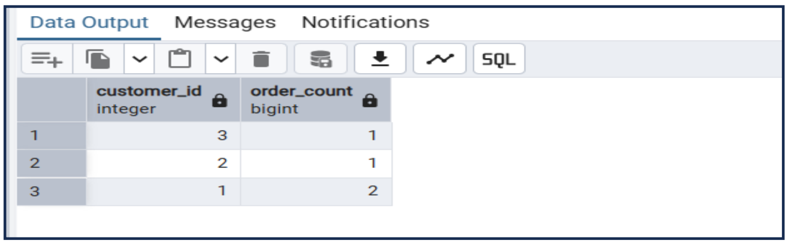


Data Output Messages Notifications

	product_id [PK] integer	name character varying (100)	description text	price numeric (10,2)	stock_quantity integer
1	4	Tablet	Portable tablet device	399.99	8

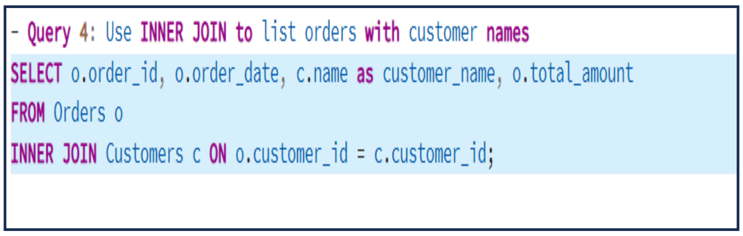


```sql
-- Query 3: Group orders by customer and count the number of orders per customer
SELECT customer_id, COUNT(*) as order_count
FROM Orders
GROUP BY customer_id;
```

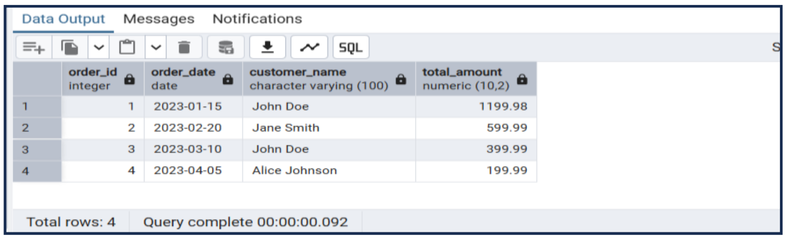


Data Output Messages Notifications

	customer_id integer	order_count bigint
1	3	1
2	2	1
3	1	2



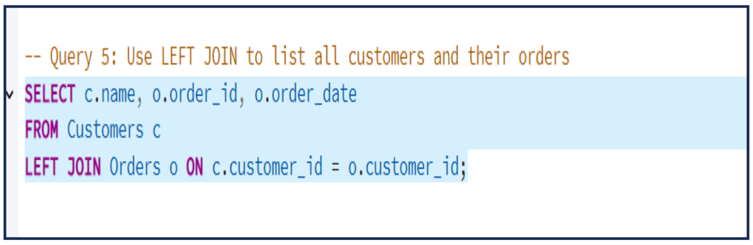
```sql
- Query 4: Use INNER JOIN to list orders with customer names
SELECT o.order_id, o.order_date, c.name as customer_name, o.total_amount
FROM Orders o
INNER JOIN Customers c ON o.customer_id = c.customer_id;
```



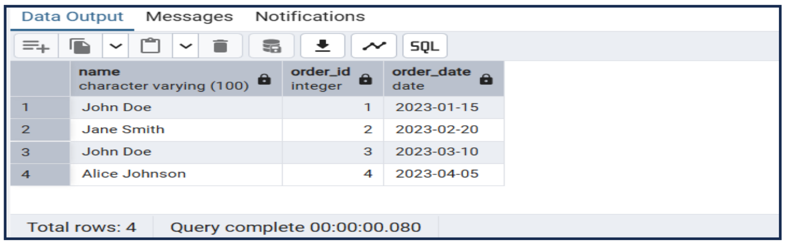
Data Output Messages Notifications

	order_id integer	order_date date	customer_name character varying (100)	total_amount numeric (10,2)
1	1	2023-01-15	John Doe	1199.98
2	2	2023-02-20	Jane Smith	599.99
3	3	2023-03-10	John Doe	399.99
4	4	2023-04-05	Alice Johnson	199.99

Total rows: 4 Query complete 00:00:00.092



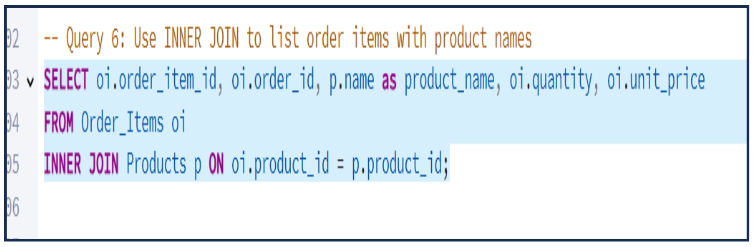
```sql
-- Query 5: Use LEFT JOIN to list all customers and their orders
SELECT c.name, o.order_id, o.order_date
FROM Customers c
LEFT JOIN Orders o ON c.customer_id = o.customer_id;
```



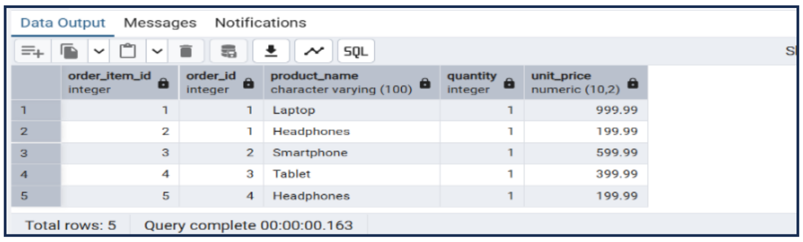
Data Output Messages Notifications

	name character varying (100)	order_id integer	order_date date
1	John Doe	1	2023-01-15
2	Jane Smith	2	2023-02-20
3	John Doe	3	2023-03-10
4	Alice Johnson	4	2023-04-05

Total rows: 4 Query complete 00:00:00.080



```sql
-- Query 6: Use INNER JOIN to list order items with product names
SELECT oi.order_item_id, oi.order_id, p.name as product_name, oi.quantity, oi.unit_price
FROM Order_Items oi
INNER JOIN Products p ON oi.product_id = p.product_id;
```

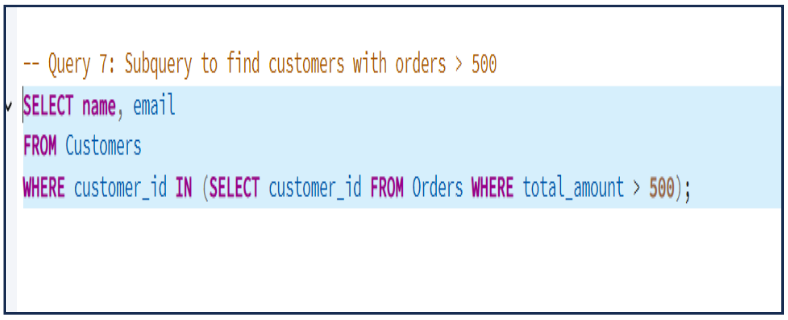


Data Output Messages Notifications

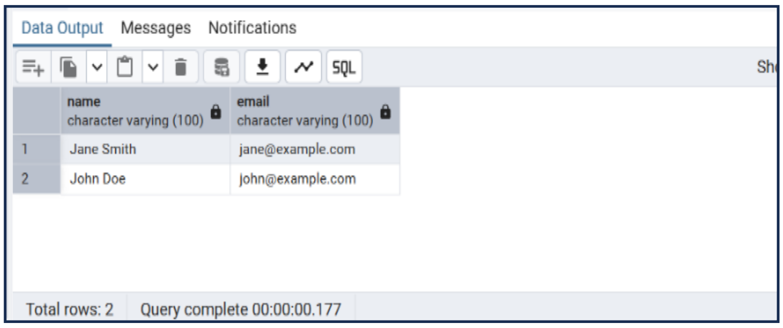
	order_item_id integer	order_id integer	product_name character varying (100)	quantity integer	unit_price numeric (10,2)
1	1	1	Laptop	1	999.99
2	2	1	Headphones	1	199.99
3	3	2	Smartphone	1	599.99
4	4	3	Tablet	1	399.99
5	5	4	Headphones	1	199.99

Total rows: 5 Query complete 00:00:00.163

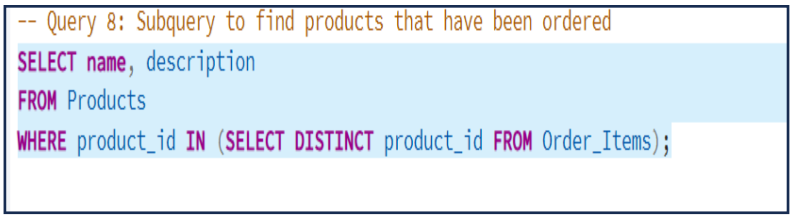
OUTPUT



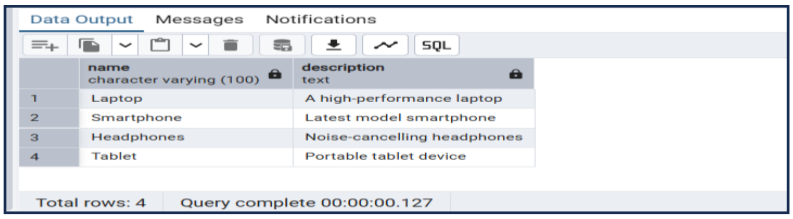
```sql
-- Query 7: Subquery to find customers with orders > 500
SELECT name, email
FROM Customers
WHERE customer_id IN (SELECT customer_id FROM Orders WHERE total_amount > 500);
```



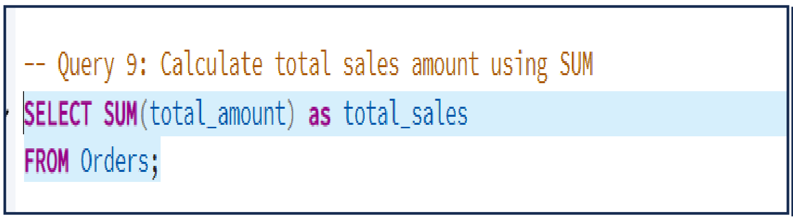
```
   name                          email
   character varying (100)       character varying (100)
1  Jane Smith                    jane@example.com
2  John Doe                      john@example.com

Total rows: 2   Query complete 00:00:00.177
```



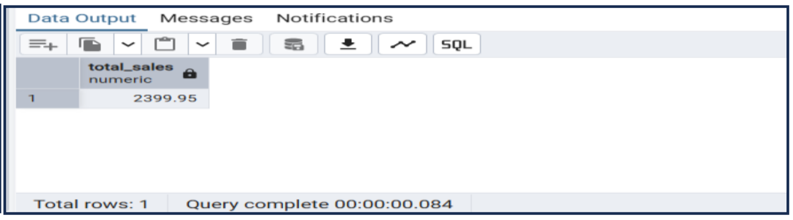
```sql
-- Query 8: Subquery to find products that have been ordered
SELECT name, description
FROM Products
WHERE product_id IN (SELECT DISTINCT product_id FROM Order_Items);
```



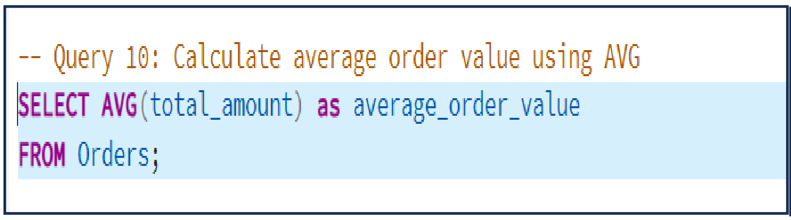
```
   name                        description
   character varying (100)     text
1  Laptop                      A high-performance laptop
2  Smartphone                  Latest model smartphone
3  Headphones                  Noise-cancelling headphones
4  Tablet                      Portable tablet device

Total rows: 4   Query complete 00:00:00.127
```



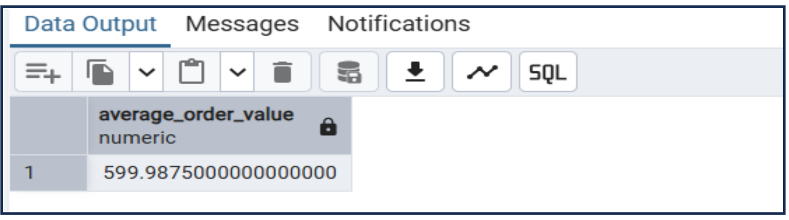
```sql
-- Query 9: Calculate total sales amount using SUM
SELECT SUM(total_amount) as total_sales
FROM Orders;
```



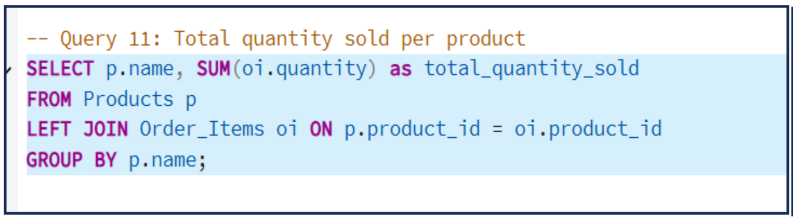
```
   total_sales
   numeric
1      2399.95

Total rows: 1   Query complete 00:00:00.084
```



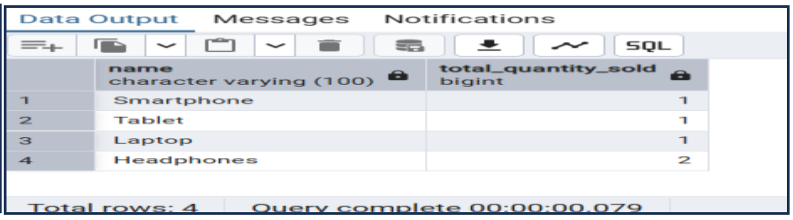
```sql
-- Query 10: Calculate average order value using AVG
SELECT AVG(total_amount) as average_order_value
FROM Orders;
```



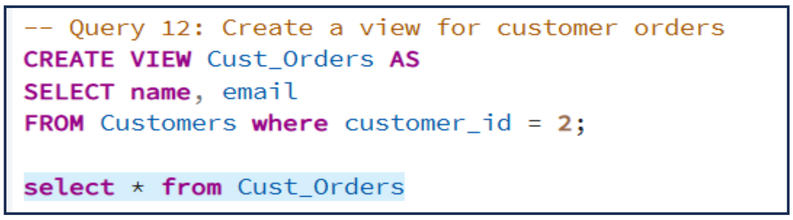
```
   average_order_value
   numeric
1  599.9875000000000000
```



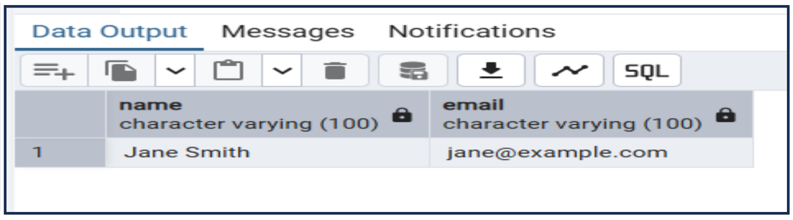
```sql
-- Query 11: Total quantity sold per product
SELECT p.name, SUM(oi.quantity) as total_quantity_sold
FROM Products p
LEFT JOIN Order_Items oi ON p.product_id = oi.product_id
GROUP BY p.name;
```



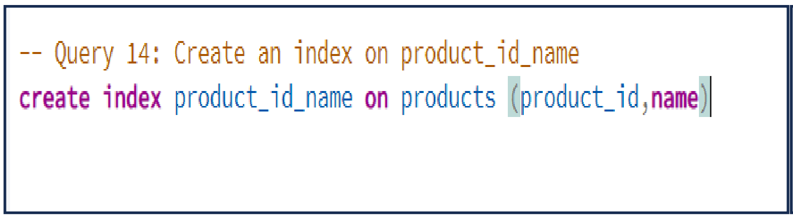
```
   name                         total_quantity_sold
   character varying (100)      bigint
1  Smartphone                                     1
2  Tablet                                         1
3  Laptop                                         1
4  Headphones                                     2

Total rows: 4   Query complete 00:00:00.079
```



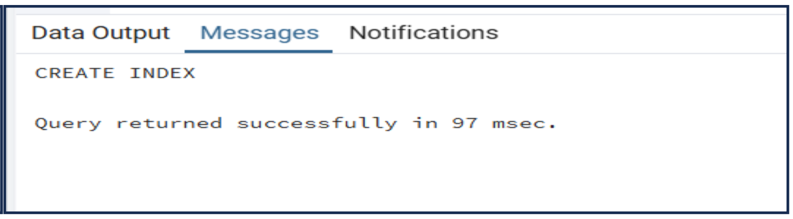
```sql
-- Query 12: Create a view for customer orders
CREATE VIEW Cust_Orders AS
SELECT name, email
FROM Customers where customer_id = 2;

select * from Cust_Orders
```



```
   name                         email
   character varying (100)      character varying (100)
1  Jane Smith                   jane@example.com
```



```sql
-- Query 14: Create an index on product_id_name
create index product_id_name on products (product_id,name)
```



```
CREATE INDEX

Query returned successfully in 97 msec.
```