Input

```sql
CREATE TABLE customers (
  customer_id INT PRIMARY KEY,
  customer_name VARCHAR(100),
  country VARCHAR(50)
);
CREATE TABLE orders (
  order_id INT PRIMARY KEY,
  customer_id INT,
  order_date DATE,
  order_total DECIMAL(10, 2),
  FOREIGN KEY (customer_id) REFERENCES customers(customer_id)
);
CREATE TABLE products (
  product_id INT PRIMARY KEY,
  product_name VARCHAR(100),
  category VARCHAR(50),
  price DECIMAL(10, 2)
);
CREATE TABLE order_items (
  order_item_id INT PRIMARY KEY,
  order_id INT,
  product_id INT,
  quantity INT,
  FOREIGN KEY (order_id) REFERENCES orders(order_id),
  FOREIGN KEY (product_id) REFERENCES products(product_id)
);
```

Run SQL

Output

SQL query successfully executed. However, the result set is empty.

Available Tables

Customers

customer_id	customer_name	country
empty		

Order_items

order_item_id	order_id	product_id	quantity
empty			

Orders

order_id	customer_id	order_date	order_total
empty			

Products

product_id	product_name	category	price
empty			

```sql
INSERT INTO customers VALUES
(1, 'Alice', 'USA'),
(2, 'Bob', 'UK'),
(3, 'Charlie', 'Canada');

INSERT INTO orders VALUES
(101, 1, '2024-01-01', 1125.00),
(102, 2, '2024-01-15', 50.00),
(103, 1, '2024-02-01', 75.00);

INSERT INTO products VALUES
(201, 'Laptop', 'Electronics', 1000.00),
(202, 'Mouse', 'Accessories', 25.00),
(203, 'Keyboard', 'Accessories', 50.00);

INSERT INTO order_items VALUES
(1, 101, 201, 1),
(2, 101, 202, 5),
(3, 102, 202, 2),
(4, 103, 203, 1);
```

Output

SQL query successfully executed. However, the result set is empty.

Available Tables

Customers

customer_id	customer_name	country
1	Alice	USA
2	Bob	UK
3	Charlie	Canada

Order_items

order_item_id	order_id	product_id	quantity
1	101	201	1
2	101	202	5
3	102	202	2
4	103	203	1

Orders

order_id	customer_id	order_date	order_total
101	1	2024-01-01	1125
102	2	2024-01-15	50
103	1	2024-02-01	75

Products

product_id	product_name	category	price
201	Laptop	Electronics	1000
202	Mouse	Accessories	25
203	Keyboard	Accessories	50

Run SQL

```
SELECT * FROM customers
WHERE country = 'USA';

SELECT customer_id, SUM(order_total) AS total_spent
FROM orders
GROUP BY customer_id
ORDER BY total_spent DESC;
```

Output

customer_id	customer_name	country
1	Alice	USA

customer_id	total_spent
1	1200
2	50

```sql
SELECT o.order_id, c.customer_name, o.order_total
FROM orders o
INNER JOIN customers c ON o.customer_id = c.customer_id;


SELECT c.customer_name, o.order_id
FROM customers c
LEFT JOIN orders o ON c.customer_id = o.customer_id;
```

Output

order_id	customer_name	order_total
101	Alice	1125
102	Bob	50
103	Alice	75

customer_name	order_id
Alice	101
Alice	103
Bob	102
Charlie	

```sql
SELECT product_name, price
FROM products
WHERE price > (
  SELECT AVG(price) FROM products
);
```

Output

product_name	price
Laptop	1000

```
|
SELECT p.product_name, SUM(oi.quantity) AS total_sold
FROM order_items oi
JOIN products p ON oi.product_id = p.product_id
GROUP BY p.product_name;


SELECT AVG(order_total) AS avg_order_value FROM orders;
```

Output

product_name	price
Laptop	1000

Run SQL

```sql
SELECT product_id,
       SUM(quantity) AS total_quantity,
       SUM(SUM(quantity)) OVER (ORDER BY product_id) AS running_total
FROM order_items
GROUP BY product_id;
```

Output

product_id	total_quantity	running_total
201	1	1
202	7	8
203	1	9

```sql
SELECT customer_id
FROM orders o
JOIN order_items oi ON o.order_id = oi.order_id
GROUP BY customer_id
HAVING COUNT(DISTINCT product_id) > 1;
```

Output

customer_id
1

Run SQL

```sql
UPDATE products
SET price = price * 1.10
WHERE category = 'Accessories';
```

Output

SQL query successfully executed. However, the result set is empty.

Customers

customer_id	customer_name	country
1	Alice	USA
2	Bob	UK
3	Charlie	Canada

Order_items

order_item_id	order_id	product_id	quantity
1	101	201	1
2	101	202	5
3	102	202	2
4	103	203	1

Orders

order_id	customer_id	order_date	order_total
101	1	2024-01-01	1125
102	2	2024-01-15	50
103	1	2024-02-01	75

Products

product_id	product_name	category	price
201	Laptop	Electronics	1000
202	Mouse	Accessories	27.500000000000004
203	Keyboard	Accessories	55.00000000000001

Input

```sql
ALTER TABLE orders ADD COLUMN order_status VARCHAR(20);
UPDATE orders
SET order_status =
  CASE
    WHEN order_total >= 1000 THEN 'Premium'
    WHEN order_total >= 100 THEN 'Standard'
    ELSE 'Basic'
  END;
```

Output

SQL query successfully executed. However, the result set is empty.

Available Tables

Customers

customer_id	customer_name	country
1	Alice	USA
2	Bob	UK
3	Charlie	Canada

Order_items

order_item_id	order_id	product_id	quantity
1	101	201	1
2	101	202	5
3	102	202	2
4	103	203	1

Orders

order_id	customer_id	order_date	order_total	order_status
101	1	2024-01-01	1125	Premium
102	2	2024-01-15	50	Basic
103	1	2024-02-01	75	Basic

Products

product_id	product_name	category	price
201	Laptop	Electronics	1000
202	Mouse	Accessories	27.500000000000004
203	Keyboard	Accessories	55.00000000000001