E-COMMERCE DATA ANALYSIS


```
mysql> use ecommercedataanalysis;
Database changed
mysql> CREATE TABLE customers (
    ->     customer_id INT PRIMARY KEY,
    ->     first_name VARCHAR(50),
    ->     last_name VARCHAR(50),
    ->     email VARCHAR(100),
    ->     created_at DATE
    -> );
Query OK, 0 rows affected (0.05 sec)

mysql> CREATE TABLE products (
    ->     product_id INT PRIMARY KEY,
    ->     product_name VARCHAR(100),
```

```
    ->     price DECIMAL(10, 2),
    ->     category VARCHAR(50),
    ->     stock INT
    -> );
Query OK, 0 rows affected (0.03 sec)


mysql> CREATE TABLE orders (
    ->     order_id INT PRIMARY KEY,
    ->     customer_id INT,
    ->     product_id INT,
    ->     quantity INT,
    ->     order_date DATE,
    ->     FOREIGN KEY (customer_id) REFERENCES
customers(customer_id),
    ->     FOREIGN KEY (product_id) REFERENCES
products(product_id)
    -> );
Query OK, 0 rows affected (0.05 sec)


mysql> INSERT INTO customers (customer_id, first_name, last_name,
email, created_at) VALUES
    -> (1, 'John', 'Doe', 'john.doe@example.com', '2024-01-01'),
    -> (2, 'Jane', 'Smith', 'jane.smith@example.com', '2024-02-
01'),
    -> (3, 'Alice', 'Johnson', 'alice.johnson@example.com', '2024-
03-01'),
    -> (4, 'Bob', 'Brown', 'bob.brown@example.com', '2024-04-01'),
    -> (5, 'Charlie', 'Davis', 'charlie.davis@example.com', '2024-
05-01');
Query OK, 5 rows affected (0.02 sec)
```

```
Records: 5  Duplicates: 0  Warnings: 0


mysql> INSERT INTO products (product_id, product_name, price,
category, stock) VALUES
    -> (1, 'Laptop', 999.99, 'Electronics', 50),
    -> (2, 'Smartphone', 699.99, 'Electronics', 100),
    -> (3, 'Tablet', 299.99, 'Electronics', 75),
    -> (4, 'Headphones', 199.99, 'Accessories', 150),
    -> (5, 'Smartwatch', 249.99, 'Accessories', 120);
Query OK, 5 rows affected (0.01 sec)
Records: 5  Duplicates: 0  Warnings: 0


mysql> INSERT INTO orders (order_id, customer_id, product_id,
quantity, order_date) VALUES
    -> (1, 1, 1, 1, '2024-01-15'),
    -> (2, 2, 2, 2, '2024-02-20'),
    -> (3, 3, 3, 1, '2024-03-25'),
    -> (4, 4, 4, 2, '2024-04-30'),
    -> (5, 5, 5, 1, '2024-05-10'),
    -> (6, 1, 2, 1, '2024-06-05'),
    -> (7, 2, 3, 3, '2024-06-10'),
    -> (8, 3, 4, 2, '2024-06-15'),
    -> (9, 4, 5, 1, '2024-06-20'),
    -> (10, 5, 1, 1, '2024-06-25');
Query OK, 10 rows affected (0.01 sec)
Records: 10  Duplicates: 0  Warnings: 0
```

❖ List of Customers

```
mysql> select * from customers;
+-------------+------------+-----------+-------------------------
-+------------+
| customer_id | first_name | last_name | email
| created_at |
+-------------+------------+-----------+-------------------------
-+------------+
|           1 | John       | Doe       | john.doe@example.com
| 2024-01-01 |
|           2 | Jane       | Smith     | jane.smith@example.com
| 2024-02-01 |
|           3 | Alice      | Johnson   | alice.johnson@example.com
| 2024-03-01 |
|           4 | Bob        | Brown     | bob.brown@example.com
| 2024-04-01 |
|           5 | Charlie    | Davis     | charlie.davis@example.com
| 2024-05-01 |
+-------------+------------+-----------+-------------------------
-+------------+
5 rows in set (0.00 sec)
```

❖ List product details

```
mysql> select * from products;
+------------+--------------+--------+-------------+-------+
| product_id | product_name | price  | category    | stock |
+------------+--------------+--------+-------------+-------+
|          1 | Laptop       | 999.99 | Electronics |    50 |
|          2 | Smartphone   | 699.99 | Electronics |   100 |
|          3 | Tablet       | 299.99 | Electronics |    75 |
|          4 | Headphones   | 199.99 | Accessories |   150 |
```

```
|          5 | Smartwatch   | 249.99 | Accessories |   120 |
+------------+--------------+--------+-------------+-------+
5 rows in set (0.00 sec)
```

❖ List of orders details

```
mysql> select * from orders;
+----------+-------------+------------+----------+------------+
| order_id | customer_id | product_id | quantity | order_date |
+----------+-------------+------------+----------+------------+
|        1 |           1 |          1 |        1 | 2024-01-15 |
|        2 |           2 |          2 |        2 | 2024-02-20 |
|        3 |           3 |          3 |        1 | 2024-03-25 |
|        4 |           4 |          4 |        2 | 2024-04-30 |
|        5 |           5 |          5 |        1 | 2024-05-10 |
|        6 |           1 |          2 |        1 | 2024-06-05 |
|        7 |           2 |          3 |        3 | 2024-06-10 |
|        8 |           3 |          4 |        2 | 2024-06-15 |
|        9 |           4 |          5 |        1 | 2024-06-20 |
|       10 |           5 |          1 |        1 | 2024-06-25 |
+----------+-------------+------------+----------+------------+
10 rows in set (0.00 sec)
```

❖ Total number of Customers

```
mysql> SELECT COUNT(*)  FROM customers;
+----------+
| COUNT(*) |
+----------+
|        5 |
+----------+
```

```
1 row in set (0.00 sec)
```

❖ Total number of Prodcts

```
mysql> SELECT COUNT(*)  FROM products;
+----------+
| COUNT(*) |
+----------+
|        5 |
+----------+
1 row in set (0.00 sec)
```

❖ Total number of order

```
mysql> SELECT COUNT(*) FROM orders;
+----------+
| COUNT(*) |
+----------+
|       10 |
+----------+
1 row in set (0.00 sec)
```

❖ Total Sales by Customer

```
mysql> SELECT c.customer_id, c.first_name, c.last_name,
SUM(p.price * o.quantity) AS total_spent
    -> FROM orders o
    -> JOIN customers c ON o.customer_id = c.customer_id
    -> JOIN products p ON o.product_id = p.product_id
    -> GROUP BY c.customer_id, c.first_name, c.last_name;
```

```
+-------------+------------+-----------+-------------+
| customer_id | first_name | last_name | total_spent |
+-------------+------------+-----------+-------------+
|           1 | John       | Doe       |     1699.98 |
|           2 | Jane       | Smith     |     2299.95 |
|           3 | Alice      | Johnson   |      699.97 |
|           4 | Bob        | Brown     |      649.97 |
|           5 | Charlie    | Davis     |     1249.98 |
+-------------+------------+-----------+-------------+
5 rows in set (0.01 sec)
```

❖ \

```
mysql> SELECT p.product_id, p.product_name, SUM(p.price *
o.quantity) AS total_sales
    -> FROM orders o
    -> JOIN products p ON o.product_id = p.product_id
    -> GROUP BY p.product_id, p.product_name;
+------------+--------------+-------------+
| product_id | product_name | total_sales |
+------------+--------------+-------------+
|          1 | Laptop       |     1999.98 |
|          2 | Smartphone   |     2099.97 |
|          3 | Tablet       |     1199.96 |
|          4 | Headphones   |      799.96 |
|          5 | Smartwatch   |      499.98 |
+------------+--------------+-------------+
5 rows in set (0.00 sec)
```

❖ Average Order Value

```
mysql> SELECT AVG(total_order_value) AS average_order_value
    -> FROM (
    ->     SELECT SUM(p.price * o.quantity) AS total_order_value
    ->     FROM orders o
    ->     JOIN products p ON o.product_id = p.product_id
    ->     GROUP BY o.order_id
    -> ) AS order_totals;
+---------------------+
| average_order_value |
+---------------------+
|          659.985000 |
+---------------------+
1 row in set (0.00 sec)
```

❖ Top 3 Best-Selling Products

```
mysql> SELECT p.product_id, p.product_name, SUM(o.quantity) AS
total_quantity_sold
    -> FROM orders o
    -> JOIN products p ON o.product_id = p.product_id
    -> GROUP BY p.product_id, p.product_name
    -> ORDER BY total_quantity_sold DESC
    -> LIMIT 3;
```

```
+------------+--------------+---------------------+
| product_id | product_name | total_quantity_sold |
+------------+--------------+---------------------+
|          3 | Tablet       |                   4 |
|          4 | Headphones   |                   4 |
|          2 | Smartphone   |                   3 |
+------------+--------------+---------------------+
3 rows in set (0.01 sec)
```

❖ Monthly Revenue

```
mysql> SELECT DATE_FORMAT(order_date, '%Y-%m') AS order_month,
COUNT(*) AS total_orders
    -> FROM orders
    -> GROUP BY order_month;
+-------------+--------------+
| order_month | total_orders |
+-------------+--------------+
| 2024-01     |            1 |
| 2024-02     |            1 |
| 2024-03     |            1 |
| 2024-04     |            1 |
| 2024-05     |            1 |
| 2024-06     |            5 |
+-------------+--------------+
6 rows in set (0.00 sec)
```

❖ Customer with the Highest Total Spend

```
mysql> SELECT c.customer_id, c.first_name, c.last_name,
SUM(p.price * o.quantity) AS total_spent
    -> FROM orders o
    -> JOIN customers c ON o.customer_id = c.customer_id
    -> JOIN products p ON o.product_id = p.product_id
    -> GROUP BY c.customer_id, c.first_name, c.last_name
    -> ORDER BY total_spent DESC
    -> LIMIT 1;
+-------------+------------+-----------+-------------+
| customer_id | first_name | last_name | total_spent |
+-------------+------------+-----------+-------------+
|           2 | Jane       | Smith     |     2299.95 |
+-------------+------------+-----------+-------------+
1 row in set (0.00 sec)
```

❖ Stock Levels of Each Product

```
mysql> SELECT product_id, product_name, stock FROM products;
+------------+--------------+-------+
| product_id | product_name | stock |
+------------+--------------+-------+
|          1 | Laptop       |    50 |
|          2 | Smartphone   |   100 |
|          3 | Tablet       |    75 |
|          4 | Headphones   |   150 |
|          5 | Smartwatch   |   120 |
+------------+--------------+-------+
```

```
5 rows in set (0.00 sec)
```

❖ Total Quantity Sold of Each Product

```
mysql> SELECT p.product_id, p.product_name, SUM(o.quantity) AS
total_quantity_sold
    -> FROM orders o
    -> JOIN products p ON o.product_id = p.product_id
    -> GROUP BY p.product_id, p.product_name;
+------------+--------------+---------------------+
| product_id | product_name | total_quantity_sold |
+------------+--------------+---------------------+
|          1 | Laptop       |                   2 |
|          2 | Smartphone   |                   3 |
|          3 | Tablet       |                   4 |
|          4 | Headphones   |                   4 |
|          5 | Smartwatch   |                   2 |
+------------+--------------+---------------------+
5 rows in set (0.00 sec)
```

❖ Average Number of Products per Order

```
mysql> SELECT AVG(product_count) AS average_products_per_order
    -> FROM (
    ->     SELECT COUNT(o.product_id) AS product_count
    ->     FROM orders o
    ->     GROUP BY o.order_id
    -> ) AS order_products;
```

```
+---------------------------+
| average_products_per_order |
+---------------------------+
|                    1.0000 |
+---------------------------+
1 row in set (0.00 sec)
```

❖ Revenue by Product Category

```
mysql> SELECT p.category, SUM(p.price * o.quantity) AS
total_revenue
    -> FROM orders o
    -> JOIN products p ON o.product_id = p.product_id
    -> GROUP BY p.category;
+-------------+---------------+
| category    | total_revenue |
+-------------+---------------+
| Electronics |       5299.91 |
| Accessories |       1299.94 |
+-------------+---------------+
2 rows in set (0.00 sec)
```