

E-Commerce Data Analysis with MySQL

Project Overview

This project analyzes an E-commerce dataset loaded into MySQL/PostgreSQL.

The goal is to extract actionable business insights, focusing on customer purchasing behavior and overall sales performance.

The following key metrics were analyzed:

- Total -selling product
 - Total revenue
 - Orders Per Customer
 - Monthly Sales
 - Total number of orders
 - Most sold product
 - Revenue by product
 - Customer purchase summary
 - Orders by date
-

Objective

To analyze an e-commerce dataset by loading order data into a MySQL database, writing SQL queries to extract insights, and presenting the results.

This project demonstrates database design, data import, querying, and reporting skills using MySQL Workbench.

Dataset

- Source: Synthetic e-commerce order data
- Table: orders
- Columns:

Column Name	Data Type	Description
order_id	INT	Unique order identifier
customer_id	INT	Unique customer identifier
product	VARCHAR(255)	Product name

Column Name	Data Type	Description
quantity	INT	Quantity of product sold
price	DECIMAL(10,2)	Price per unit
order_date	DATE	Date of the order

	A	B	C	D	E	F	G	H	I
1	order_id	customer_id	product	quantity	price	order_date			
2	1	101	Smartphone	2	299.99	15/01/2023			
3	2	102	Laptop	1	899.5	20/01/2023			
4	3	103	Headphones	3	49.99	05/02/2023			
5	4	104	Smartwatch	1	199.99	18/02/2023			
6	5	105	Keyboard	4	29.99	12/03/2023			
7	6	106	Mouse	2	19.99	15/03/2023			
8	7	101	Smartphone	1	299.99	25/03/2023			
9	8	107	Tablet	2	399	02/04/2023			
10	9	108	Monitor	1	149.5	10/04/2023			
11	10	109	Laptop	1	950	22/04/2023			
12									
13									
14									
15									
16									
17									
18									
19									

Steps Performed

Database Setup

- Installed MySQL Server & Workbench.
- Created a database named ecommerce.
- Created the orders table with the above schema.

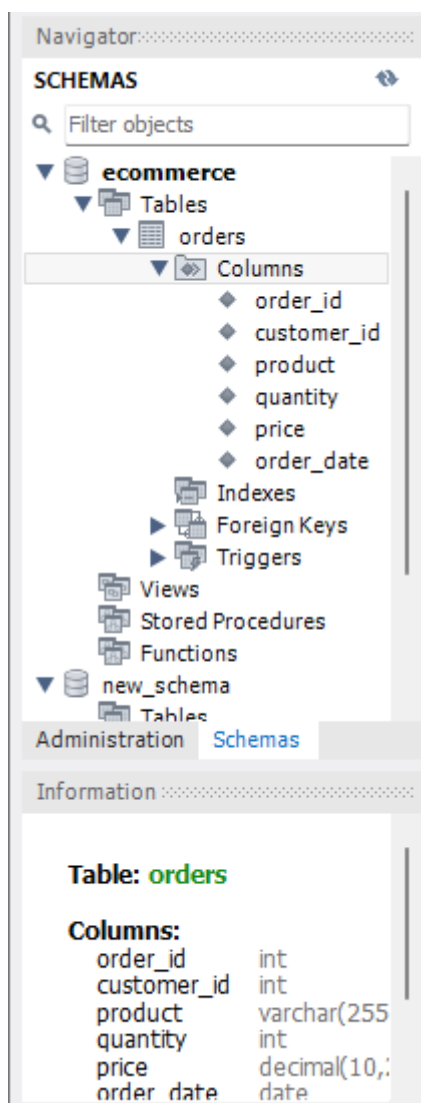
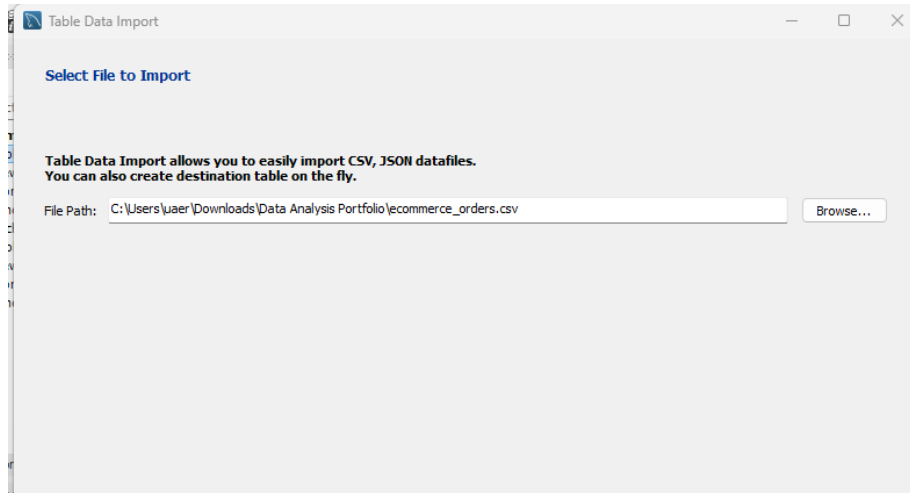
```

1  CREATE TABLE orders (
2      order_id INT,
3      customer_id INT,
4      product VARCHAR(255),
5      quantity INT,
6      price DECIMAL(10,2),
7      order_date DATE
8  );
9

```

Data Import

- Inserted sample order data into the orders table.



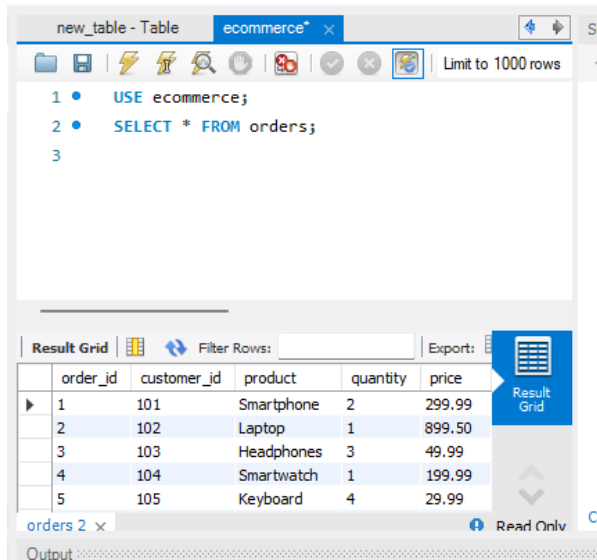
Queries Executed

- Selected all data:

sql

Copy code

```
SELECT * FROM orders;
```



The screenshot shows a SQL IDE window titled "new_table - Table" with a tab for "ecommerce". The query editor contains the following SQL code:

```
1 • USE ecommerce;  
2 • SELECT * FROM orders;  
3
```

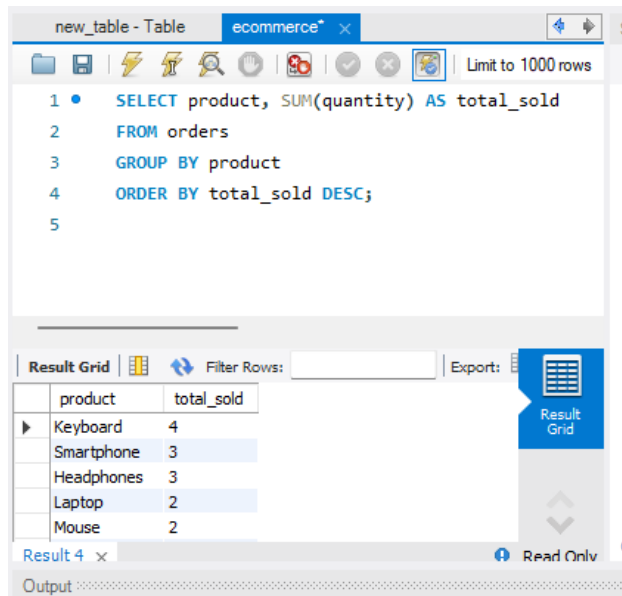
Below the query editor is the "Result Grid" tab, which displays the results of the query. The grid has five columns: "order_id", "customer_id", "product", "quantity", and "price". The results are as follows:

order_id	customer_id	product	quantity	price
1	101	Smartphone	2	299.99
2	102	Laptop	1	899.50
3	103	Headphones	3	49.99
4	104	Smartwatch	1	199.99
5	105	Keyboard	4	29.99

At the bottom of the window, there is an "Output" section and a "Read Only" button.

Insights:

- Top-selling product



The screenshot shows a SQL query editor window titled "new_table - Table" with a tab for "ecommerce*". The query is as follows:

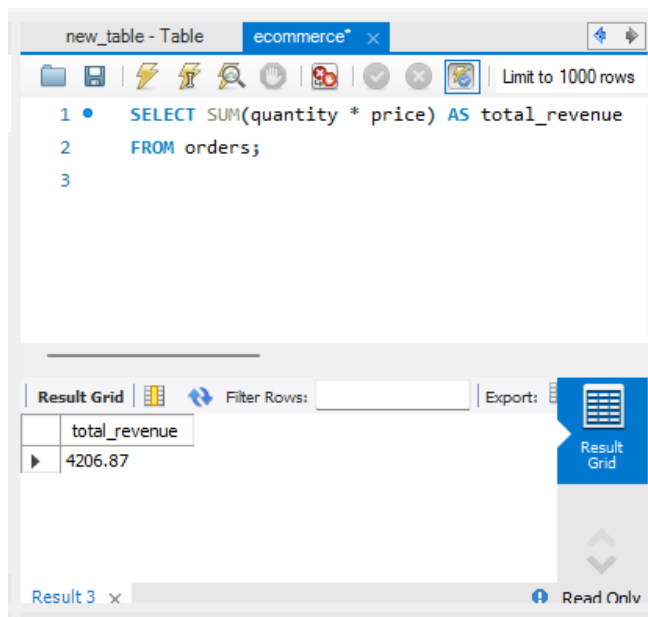
```
1 • SELECT product, SUM(quantity) AS total_sold
2 FROM orders
3 GROUP BY product
4 ORDER BY total_sold DESC;
5
```

Below the query editor, the "Result Grid" is displayed, showing the results of the query. The grid has two columns: "product" and "total_sold". The results are as follows:

product	total_sold
Keyboard	4
Smartphone	3
Headphones	3
Laptop	2
Mouse	2

The interface also includes a "Filter Rows" field, an "Export" button, and a "Result Grid" button. The status bar at the bottom indicates "Result 4" and "Read Only".

- Total revenue



The screenshot shows a SQL query editor window titled "new_table - Table" with a tab for "ecommerce*". The query is as follows:

```
1 • SELECT SUM(quantity * price) AS total_revenue
2 FROM orders;
3
```

Below the query editor, the "Result Grid" is displayed, showing the results of the query. The grid has one column: "total_revenue". The result is as follows:

total_revenue
4206.87

The interface also includes a "Filter Rows" field, an "Export" button, and a "Result Grid" button. The status bar at the bottom indicates "Result 3" and "Read Only".

- Orders Per Customer

The screenshot shows a SQL query editor window titled "new_table - Table" with a tab for "ecommerce". The query is as follows:

```
1 • SELECT customer_id, COUNT(*) AS orders_count
2 FROM orders
3 GROUP BY customer_id;
4
```

Below the query editor is the "Result Grid" showing the results of the query. The grid has two columns: "customer_id" and "orders_count". The results are as follows:

customer_id	orders_count
101	2
102	1
103	1
104	1
105	1

The interface also includes a "Filter Rows" field, an "Export" button, and a "Result Grid" button. The status bar at the bottom indicates "Result 5" and "Read Only".

- Monthly Sales

The screenshot shows a SQL query editor window titled "new_table - Table" with a tab for "ecommerce". The query is as follows:

```
1 • SELECT MONTH(order_date) AS month, SUM(quantity)
2 FROM orders
3 GROUP BY month
4 ORDER BY month;
5
```

Below the query editor is the "Result Grid" showing the results of the query. The grid has two columns: "month" and "revenue". The results are as follows:

month	revenue
1	1499.48
2	349.96
3	459.93
4	1897.50

The interface also includes a "Filter Rows" field, an "Export" button, and a "Result Grid" button. The status bar at the bottom indicates "Result 6" and "Read Only".

- Total number of orders

The screenshot shows a database query editor window titled "new_table - Table" with a tab for "ecommerce". The SQL query entered is: `1 • SELECT COUNT(*) AS total_orders FROM orders;` followed by a blank line `2`. Below the query, the "Result Grid" is displayed with a single row:

total_orders
10

. The interface includes a toolbar with icons for file operations, a "Limit to 1000 rows" indicator, and an "Export" button. The status bar at the bottom indicates "Result 7" and "Read Only".

- Most sold product

The screenshot shows a database query editor window titled "new_table - Table" with a tab for "ecommerce". The SQL query entered is: `1 • SELECT product, SUM(quantity) AS total_quantity`
`2 FROM orders`
`3 GROUP BY product`
`4 ORDER BY total_quantity DESC`
`5 LIMIT 1;` followed by a blank line `6`. Below the query, the "Result Grid" is displayed with a single row:

product	total_quantity
Keyboard	4

. The interface includes a toolbar with icons for file operations, a "Limit to 1000 rows" indicator, and an "Export" button. The status bar at the bottom indicates "Result 8" and "Read Only".

- Revenue by product

new_table - Table ecommerce* x Limit to 1000 rows

```

1 • SELECT product, SUM(quantity * price) AS revenue
2   FROM orders
3   GROUP BY product
4   ORDER BY revenue DESC;
5

```

Result Grid

	product	revenue
▶	Laptop	1849.50
	Smartphone	899.97
	Tablet	798.00
	Smartwatch	199.99
	Headphones	149.97

Result 9 x Read Only

- Customer purchase summary

new_table - Table ecommerce* x Limit to 1000 rows

```

6   FROM
7   orders
8   GROUP BY
9   customer_id
10  ORDER BY
11  total_spent DESC;
12

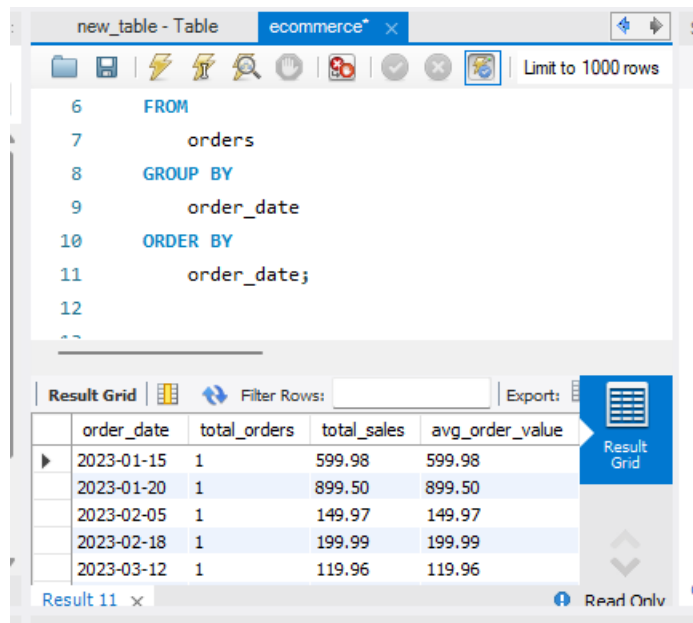
```

Result Grid

	customer_id	total_orders	total_spent	avg_order_value
▶	109	1	950.00	950.00
	101	2	899.97	449.99
	102	1	899.50	899.50
	107	1	798.00	798.00
	104	1	199.99	199.99

Result 10 x Read Only

- Orders by date



The screenshot shows the MySQL Workbench interface. The top toolbar includes icons for file operations, search, and execution. The SQL editor contains the following query:

```
6 FROM
7   orders
8 GROUP BY
9   order_date
10 ORDER BY
11   order_date;
12
```

Below the editor, the 'Result Grid' tab is active, displaying the query results. The grid has four columns: 'order_date', 'total_orders', 'total_sales', and 'avg_order_value'. The results are as follows:

	order_date	total_orders	total_sales	avg_order_value
▶	2023-01-15	1	599.98	599.98
	2023-01-20	1	899.50	899.50
	2023-02-05	1	149.97	149.97
	2023-02-18	1	199.99	199.99
	2023-03-12	1	119.96	119.96

At the bottom of the result grid, it says 'Result 11' and 'Read Only'.

Tools & Technologies Used

- MySQL 8.x
- MySQL Workbench
- CSV file (optional)

Learnings

- Hands-on experience with SQL queries
- Database design & schema creation
- Data import and export
- Generating actionable insights from raw data