# E\_commece\_SQL\_Project:

The solutions provided are valuable tools for data-driven decision-making, optimizing operations and enhancing customer experiences in the e-commerce domain. Businesses can leverage these insights to stay competitive, drive growth, and achieve their goals.

**Business Problem 1: Product Popularity Analysis**

**Use of Solution:**

* Identifying the most popular products helps businesses focus on high-demand items and allocate resources efficiently.
* Businesses can plan inventory management and restocking strategies based on product popularity.
* Promotional campaigns can be tailored around popular products to drive sales and increase revenue.

*Query to find the top 10 customers with the highest total purchase amount:*

**SQL Query:**

/\*SELECT customers.customer\_name, SUM(products.price \* order\_details.quantity) AS total\_purchase\_amount

FROM customers

JOIN orders ON customers.customer\_id = orders.customer\_id

JOIN order\_details ON orders.order\_id = order\_details.order\_id

JOIN products ON order\_details.product\_id = products.product\_id

GROUP BY customers.customer\_id, customers.customer\_name

ORDER BY total\_purchase\_amount DESC

LIMIT 10;\*/

**Result:**

|  |  |
| --- | --- |
| customer\_name | total\_purchase\_amount |
| Peter | 142000 |
| John Smith | 127000 |
| Michael Brown | 105000 |
| Lisa | 102000 |
| Emma Johnson | 60000 |
| Sophia Lee | 0 |
| William Wang | 0 |

**Business Problem 2: Order Analysis**

**Use of Solution:**

* Insights: Provides valuable insights into revenue-generating product categories.
* Strategy: Enables optimization of pricing, inventory, and marketing strategies.
* Decision-making: Empowers data-driven decision-making for growth and profitability.
* Competitive Edge: Assists in competitive analysis and performance evaluation

*Query to calculate the total revenue for each product category:*

**SQL Query:**

/\*SELECT products.category AS categories,

SUM(products.price \* order\_details.quantity) AS total\_revenue

FROM products

JOIN order\_details ON products.product\_id = order\_details.product\_id

GROUP BY categories;\*/

**Result:**

|  |  |
| --- | --- |
| categories | total\_revenue |
| Electronics | 254000 |
| Clothing | 300 |

**Business Problem 3: Revenue Analysis by City**

**Use of Solution:**

* Market Insights: Analyzing total revenue by city helps identify lucrative markets and potential areas for business growth.
* Resource Allocation: Effective allocation of marketing and sales resources to target specific cities based on revenue data.
* Pricing Optimization: Utilizing revenue analysis to optimize pricing strategies according to regional demand and purchasing power.
* Business Growth: Leveraging revenue insights to drive business growth and maximize profitability in different cities.

*Query to calculate the total revenue for each city:*

**SQL Query:**

/\*SELECT customers.customer\_name, AVG(products.price \* order\_details.quantity) AS average\_order\_value

FROM customers

JOIN orders ON customers.customer\_id = orders.customer\_id

JOIN order\_details ON orders.order\_id = order\_details.order\_id

JOIN products ON order\_details.product\_id = products.product\_id

GROUP BY customers.customer\_id, customers.customer\_name;\*/

**Result:**

|  |  |
| --- | --- |
| city | total\_revenue |
| New York | 200000 |
| Los Angeles | 40000 |
| Chicago | 0 |
| San Francisco | 0 |
| Seattle | 1000 |

**Business Problem 4: Average Order Value**

**Use of Solution:**

* Calculating the average order value for each customer helps businesses understand customer spending patterns.
* Businesses can identify high-value customers and offer personalized incentives to enhance customer loyalty.
* Analyzing average order value aids in optimizing pricing strategies and upselling opportunities.

*Query to calculate the average order value for each customer:*

**Query:**

/\*SELECT customers.customer\_name, AVG(products.price \* order\_details.quantity) AS average\_order\_value

FROM customers

JOIN orders ON customers.customer\_id = orders.customer\_id

JOIN order\_details ON orders.order\_id = order\_details.order\_id

JOIN products ON order\_details.product\_id = products.product\_id

GROUP BY customers.customer\_id, customers.customer\_name;\*/

**Result:**

|  |  |
| --- | --- |
| customer\_name | average\_order\_value |
| Peter | 50000 |
| John Smith | 63500 |
| Michael Brown | 30000 |
| Lisa | 51000 |
| Emma Johnson | 1000 |
| Sophia Lee | 0 |
| William Wang | 0 |

**Business Problem 5: Product Recommendations**

**Use of Solution:**

* Personalized product recommendations enhance customer shopping experiences and increase customer satisfaction.
* Recommending frequently co-purchased products can drive cross-selling and upselling opportunities.
* Automated marketing campaigns can be executed based on product recommendations to boost engagement and conversions.

**SQL Query:**

/\*SELECT M.product\_id AS M\_product\_id, M.product\_name AS M\_product\_name, R.product\_id AS R\_product\_id, R.product\_name AS R\_product\_name, COUNT(\*) AS frequency FROM order\_details M JOIN order\_details R ON M.order\_id = R.order\_id AND M.product\_id != R.product\_id JOIN products M\_product ON M.product\_id = M\_product.product\_id JOIN products R\_product ON R.product\_id = R\_product.product\_id GROUP BY M.product\_id, M.product\_name, R.product\_id, R.product\_name ORDER BY M\_product\_id, frequency DESC LIMIT 5;\*/

**Result:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| M\_product\_id | M\_product\_name | R\_product\_id | R\_product\_name | frequency |
| 1 | Product A | 3 | Product C | 50 |
| 1 | Product A | 5 | Product E | 48 |
| 1 | Product A | 2 | Product B | 46 |
| 1 | Product A | 4 | Product D | 44 |
| 1 | Product A | 6 | Product F | 42 |
| 2 | Product B | 1 | Product A | 46 |
| 2 | Product B | 5 | Product E | 45 |
| 2 | Product B | 4 | Product D | 43 |
| 2 | Product B | 6 | Product F | 41 |
| 2 | Product B | 3 | Product C | 40 |