Product table

1. Retrieve all columns from the product table.

mysql> select \* from products;

+------------+--------------+-------------+------------+

| product\_id | product\_name | category | unit\_price |

+------------+--------------+-------------+------------+

| 101 | Laptop | Electronics | 500.00 |

| 102 | Smartphone | Electronics | 300.00 |

| 103 | Headphones | Electronics | 30.00 |

| 104 | Keyboard | Electronics | 20.00 |

| 105 | Mouse | Electronics | 15.00 |

+------------+--------------+-------------+------------+

5 rows in set (0.06 sec)

2. Retrieve the product\_name and unit\_price from the Products table.

mysql> select product\_name,unit\_price from products;

+--------------+------------+

| product\_name | unit\_price |

+--------------+------------+

| Laptop | 500.00 |

| Smartphone | 300.00 |

| Headphones | 30.00 |

| Keyboard | 20.00 |

| Mouse | 15.00 |

+--------------+------------+

5 rows in set (0.00 sec)

3. Filter the Products table to show only products in the 'Electronics'

category.

mysql> select \* from products where category='electronics';

+------------+--------------+-------------+------------+

| product\_id | product\_name | category | unit\_price |

+------------+--------------+-------------+------------+

| 101 | Laptop | Electronics | 500.00 |

| 102 | Smartphone | Electronics | 300.00 |

| 103 | Headphones | Electronics | 30.00 |

| 104 | Keyboard | Electronics | 20.00 |

| 105 | Mouse | Electronics | 15.00 |

+------------+--------------+-------------+------------+

5 rows in set (0.12 sec)

4.Retrieve the product\_id and product\_name from the Products table for products with a unit\_price greater than $100.

mysql> select product\_id,product\_name from products where unit\_price>100;

+------------+--------------+

| product\_id | product\_name |

+------------+--------------+

| 101 | Laptop |

| 102 | Smartphone |

+------------+--------------+

2 rows in set (0.02 sec)

5. Calculate the average unit\_price of products in the Products table.

mysql> select avg(unit\_price)as average\_price from products;

+---------------+

| average\_price |

+---------------+

| 173.000000 |

+---------------+

1 row in set (0.22 sec)

6. Retrieve product\_name and unit\_price from the Products table with the Highest Unit Price

mysql> select product\_name,unit\_price from products where unit\_price=(select max(unit\_price)from products);

+--------------+------------+

| product\_name | unit\_price |

+--------------+------------+

| Laptop | 500.00 |

+--------------+------------+

1 row in set (0.21 sec)

7. Retrieve the product\_name and unit\_price from the Products table, ordering the results by unit\_price in descending order.

mysql> select product\_name,unit\_price from products order by unit\_price desc;

+--------------+------------+

| product\_name | unit\_price |

+--------------+------------+

| Laptop | 500.00 |

| Smartphone | 300.00 |

| Headphones | 30.00 |

| Keyboard | 20.00 |

| Mouse | 15.00 |

+--------------+------------+

5 rows in set (0.06 sec)

8. Retrieve the product\_name and unit\_price from the Products table, filtering the unit\_price to show only values between $20 and $600.

mysql> select product\_name,unit\_price from products where unit\_price between 20 and 600;

+--------------+------------+

| product\_name | unit\_price |

+--------------+------------+

| Laptop | 500.00 |

| Smartphone | 300.00 |

| Headphones | 30.00 |

| Keyboard | 20.00 |

+--------------+------------+

4 rows in set (0.06 sec)

9. Retrieve the product\_name and category from the Products table, ordering the results bycategory in ascending order.

mysql> select product\_name,category from products order by category asc;

+--------------+-------------+

| product\_name | category |

+--------------+-------------+

| Laptop | Electronics |

| Smartphone | Electronics |

| Headphones | Electronics |

| Keyboard | Electronics |

| Mouse | Electronics |

+--------------+-------------+

5 rows in set (0.07 sec)

Sales Table

1.Retrieve all columns from the Sales table.

mysql> select \* from sales;

+---------+------------+---------------+------------+-------------+

| sale\_id | product\_id | quantity\_sold | sale\_date | total\_price |

+---------+------------+---------------+------------+-------------+

| 1 | 101 | 5 | 2024-01-01 | 2500.00 |

| 2 | 102 | 3 | 2024-01-02 | 900.00 |

| 3 | 103 | 2 | 2024-01-02 | 60.00 |

| 4 | 104 | 4 | 2024-01-03 | 80.00 |

| 5 | 105 | 6 | 2024-01-03 | 90.00 |

+---------+------------+---------------+------------+-------------+

5 rows in set (0.00 sec)

2.Retrieve the sale\_id and sale\_date from the Sales table.

mysql> select sale\_id,sale\_date from sales;

+---------+------------+

| sale\_id | sale\_date |

+---------+------------+

| 1 | 2024-01-01 |

| 2 | 2024-01-02 |

| 3 | 2024-01-02 |

| 4 | 2024-01-03 |

| 5 | 2024-01-03 |

+---------+------------+

5 rows in set (0.00 sec)

3.Filter the Sales table to show only sales with a total\_price greater than $100

mysql> select\*from sales where total\_price>100;

+---------+------------+---------------+------------+-------------+

| sale\_id | product\_id | quantity\_sold | sale\_date | total\_price |

+---------+------------+---------------+------------+-------------+

| 1 | 101 | 5 | 2024-01-01 | 2500.00 |

| 2 | 102 | 3 | 2024-01-02 | 900.00 |

+---------+------------+---------------+------------+-------------+

2 rows in set (0.00 sec)

4.Retrieve the sale\_id and total\_price from the Sales table for sales made on January 3, 2024.

+---------+-------------+

| sale\_id | total\_price |

+---------+-------------+

| 4 | 80.00 |

| 5 | 90.00 |

+---------+-------------+

2 rows in set (0.06 sec)

5.Calculate the total revenue generated from all sales in the Sales table.

mysql> select sum(total\_price)as total\_revenue from sales;

+---------------+

| total\_revenue |

+---------------+

| 3630.00 |

+---------------+

1 row in set (0.00 sec)

6.Calculate the total quantity\_sold from the Sales table.

mysql> select sum(quantity\_sold)as total\_quantity\_sold from sales;

+---------------------+

| total\_quantity\_sold |

+---------------------+

| 20 |

+---------------------+

1 row in set (0.00 sec)

7. Retrieve the sale\_id, product\_id, and total\_price from the Sales table for sales with a quantity\_sold greater than 4.

mysql> SELECT sale\_id, product\_id, total\_price FROM Sales

-> WHERE quantity\_sold > 4;

+---------+------------+-------------+

| sale\_id | product\_id | total\_price |

+---------+------------+-------------+

| 1 | 101 | 2500.00 |

| 5 | 105 | 90.00 |

+---------+------------+-------------+

2 rows in set (0.00 sec)

8. Calculate the average total\_price of sales in the Sales table.

mysql> SELECT AVG(total\_price) AS average\_sale\_price FROM Sales;

+--------------------+

| average\_sale\_price |

+--------------------+

| 726.000000 |

+--------------------+

1 row in set (0.00 sec)