use appliances;

Create table Products(

product\_id int primary key,

product\_name varchar(100),

category varchar(50),

unit\_price decimal(10,2));

Insert into products values(101,'laptop', 'electronics', 500.00);

Insert into products values(102,'smartphone', 'electronics', 300.00);

Insert into products values(103,'headphones', 'electronics', 30.00);

Insert into products values(104,'keyboard', 'electronics', 20.00);

Insert into products values(105,'mouse', 'electronics', 15.00);

1. Retrieve all columns from the product table.

select \*from appliances.products;

3. Filter the Products table to show only products in the ’Electronic’ category.

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

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

select product\_name, unit\_price from products;

4. Retrieve the product\_id and product\_name from the Products table for products with a

unit\_price greater than $100.

select product\_id, product\_name from products where unit\_price>100.00;

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

select Avg(unit\_price)as average\_price from products;

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

unit\_price in descending order.

select product\_name,unit\_price from products order by unit\_price DESC;

8. Retrieve the product\_name and unit\_price from the Products table, filtering the unit\_price to

show only values between $20 and $600.

select product\_name, unit\_price from Products where unit\_price between 20 AND 600;

9. Retrieve the product\_name and category from the Products table, ordering the results by

category in ascending order.

select product\_name, category FROM Products order by category ASC;

use appliances;

CREATE TABLE Sales (

sale\_id INT PRIMARY KEY,

product\_id INT,

quantity\_sold INT,

sale\_date DATE,

total\_price DECIMAL(10, 2),

FOREIGN KEY (product\_id) REFERENCES Products(product\_id));

Insert into sales values(1,101, 5, '2024-01-01', 2500.00);

Insert into sales values(2,102, 3, '2024-01-02', 900.00);

Insert into sales values(3,103, 2, '2024-01-02', 60.00);

Insert into sales values(4,104, 4, '2024-01-03', 80.00);

Insert into sales values(5,105, 6, '2024-01-03', 90.00);

1. Retrieve all columns from the Sales table.

select \* from sales;

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

select sale\_id, sale\_date from Sales;

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

select \* from Sales where total\_price > 100;

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

select sale\_id, total\_price from Sales where sale\_date = '2024-01-03';

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

select SUM(total\_price) AS total\_revenue from Sales;

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

select SUM(quantity\_sold) AS total\_quantity\_sold from Sales;

7. Retrieve the sale\_id, product\_id, and total\_price from the Sales table for sales with a

quantity\_sold greater than 4.

select sale\_id, product\_id, total\_price from Sales where quantity\_sold > 4;

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

select AVG(total\_price) AS average\_total\_price from Sales;