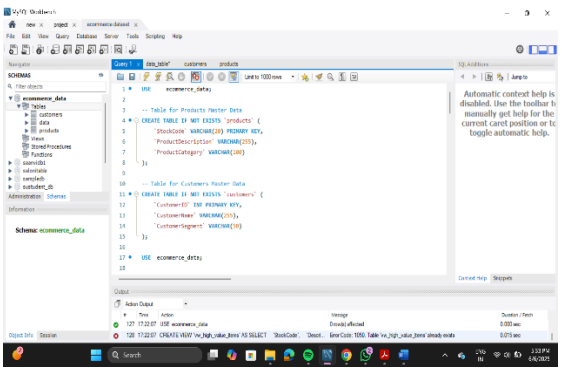
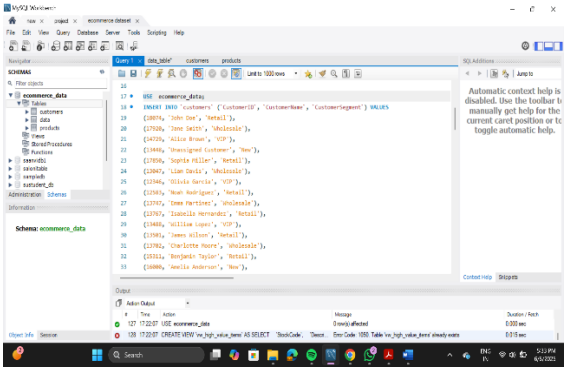
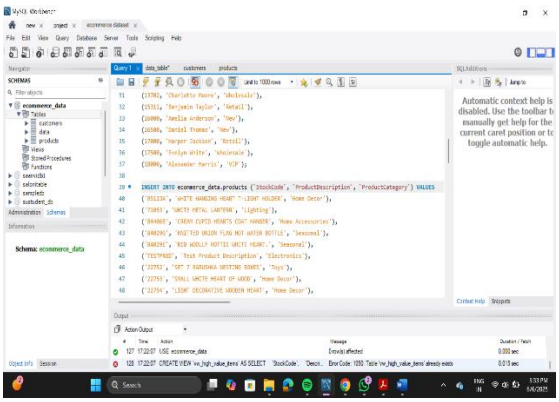
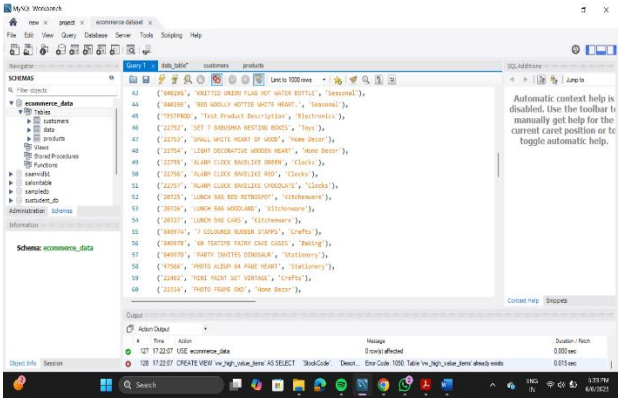
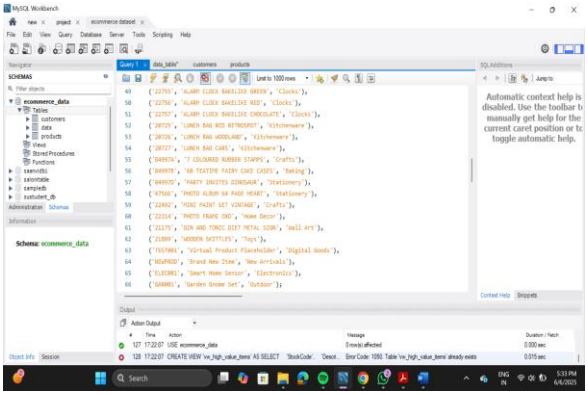
SQL QUERIES ON ECOMMECE DATASET





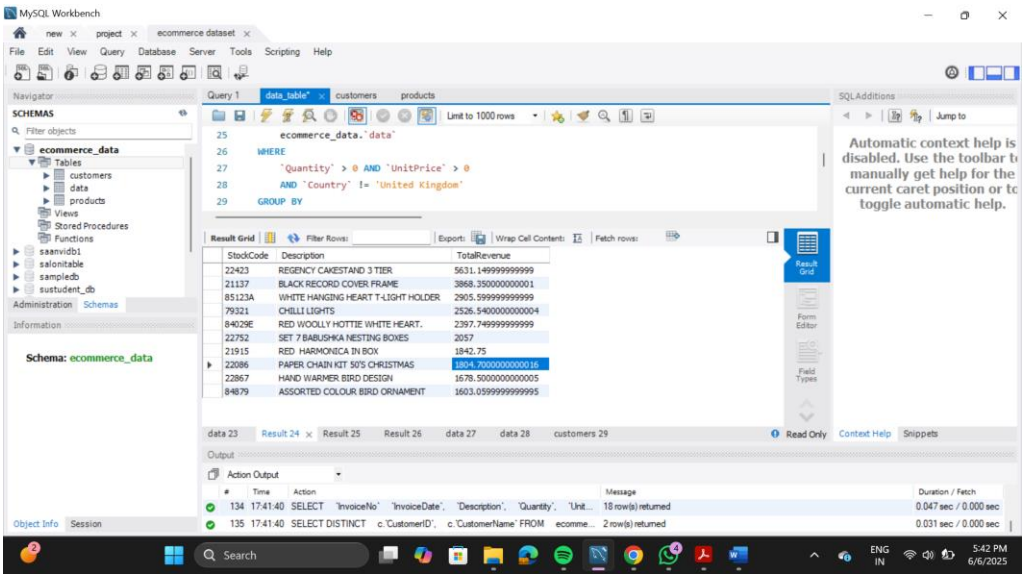






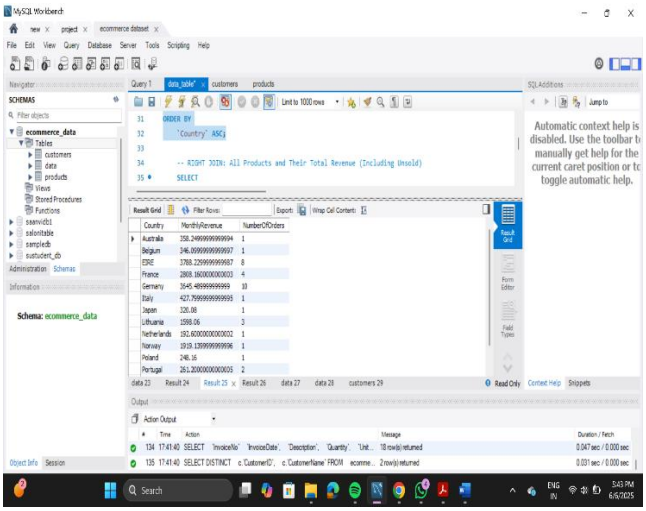
```
-- Top 10 Products by Total Sales Revenue

SELECT

    `StockCode`,

    `Description`,
```

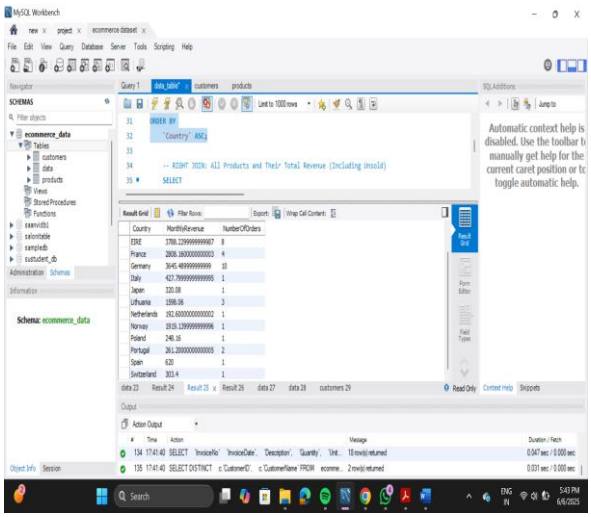
```
    SUM(`Quantity` * `UnitPrice`) AS TotalRevenue
FROM
    ecommerce_data.`data`
WHERE
    Quantity > 0 AND UnitPrice > 0
GROUP BY
    `StockCode`,
    `Description`
ORDER BY
    TotalRevenue DESC
LIMIT 10;
```




```
-- Monthly Sales Performance and Number of Unique Orders by Country (Excluding UK)
SELECT
    `Country`,
    SUM(`Quantity` * `UnitPrice`) AS MonthlyRevenue,
    COUNT(DISTINCT `InvoiceNo`) AS NumberOfOrders
```

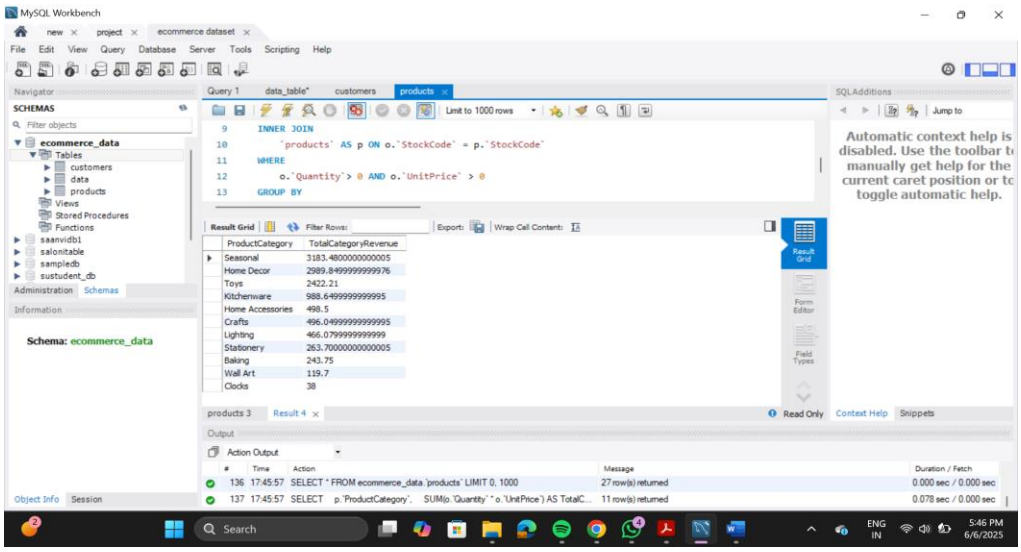
```
FROM
    ecommerce_data.`data`
WHERE
    `Quantity` > 0 AND `UnitPrice` > 0
    AND `Country` != 'United Kingdom'
GROUP BY
    `Country`
ORDER BY
    `Country` ASC;
```





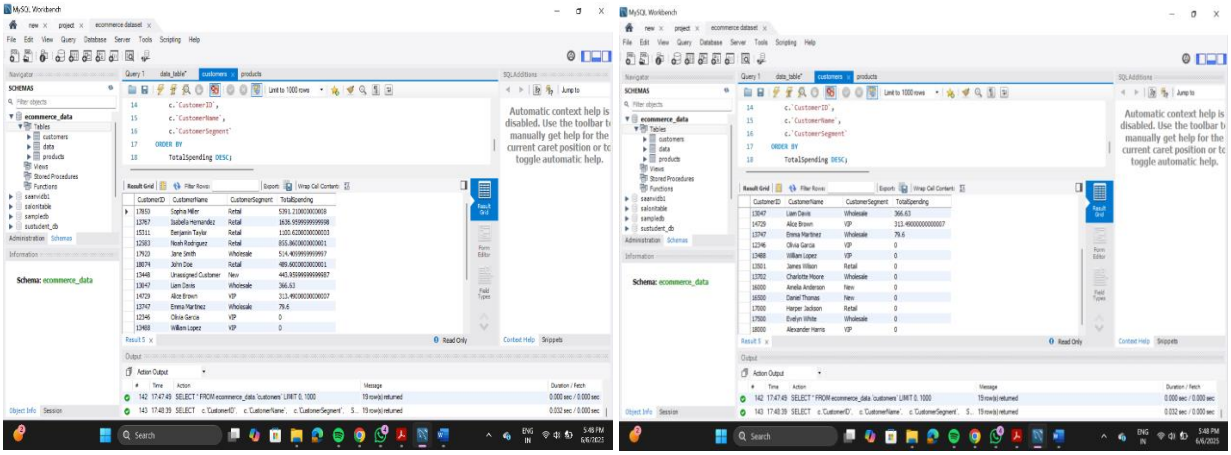
```
SELECT * FROM ecommerce_data.`products`;
-- INNER JOIN: Total Sales Revenue by Product Category
SELECT
    p.`ProductCategory`,
    SUM(o.`Quantity` * o.`UnitPrice`) AS TotalCategoryRevenue
FROM
      ecommerce_data.`data`AS o
INNER JOIN
```

```
  `products` AS p ON o.`StockCode` = p.`StockCode`
WHERE
    o.`Quantity`> 0 AND o.`UnitPrice` > 0
GROUP BY
    p.`ProductCategory`
ORDER BY
    TotalCategoryRevenue DESC;
```



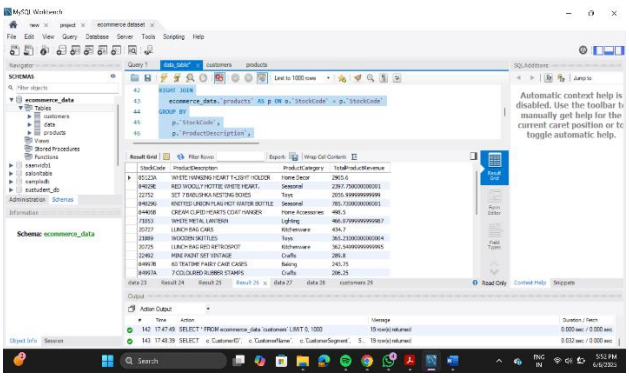

```
SELECT * FROM ecommerce_data.`customers`;
-- LEFT JOIN: All Customers and Their Total Spending
SELECT
    c.`CustomerID`,
    c.`CustomerName`,
    c.`CustomerSegment`,
    SUM(CASE WHEN o.`Quantity` > 0 AND o.`UnitPrice` > 0 THEN o.`Quantity` * o.`UnitPrice`
ELSE 0 END) AS TotalSpending
FROM
  ecommerce_data.`customers` AS c
```

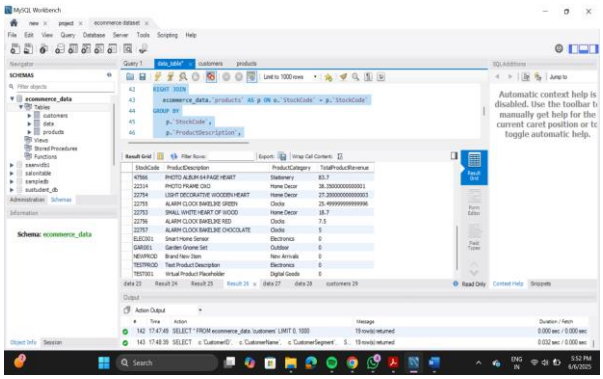
```
LEFT JOIN
        ecommerce_data.`data` AS o ON c.`CustomerID` = o.`CustomerID`
GROUP BY
    c.`CustomerID`,
    c.`CustomerName`,
    c.`CustomerSegment`
ORDER BY
    TotalSpending DESC;
```



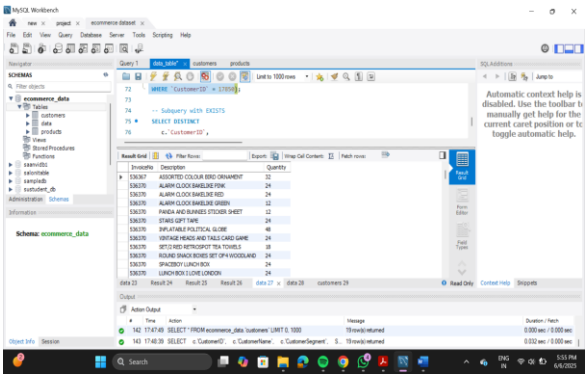
```
-- RIGHT JOIN: All Products and Their Total Revenue (Including Unsold)
    SELECT
    p.`StockCode`,
    p.`ProductDescription`,
    p.`ProductCategory`,
    SUM(CASE WHEN o.`Quantity` > 0 AND o.`UnitPrice` > 0 THEN o.`Quantity` * o.`UnitPrice`
ELSE 0 END) AS TotalProductRevenue
FROM
  ecommerce_data.`data` AS o
RIGHT JOIN
```

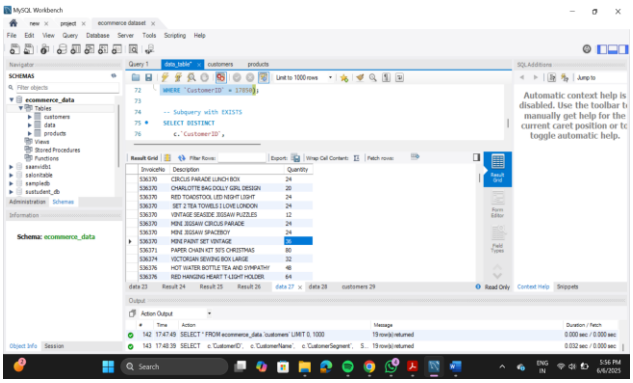
```
  ecommerce_data.`products` AS p ON o.`StockCode` = p.`StockCode`
GROUP BY
   p.`StockCode`,
   p.`ProductDescription`,
   p.`ProductCategory`
ORDER BY
   TotalProductRevenue DESC;
```

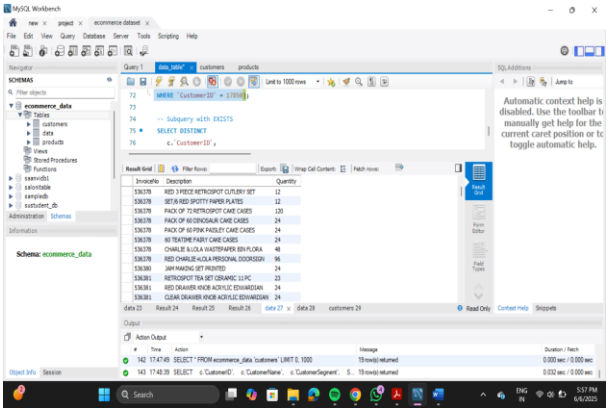




```
-- Scalar Subquery
   SELECT
   `InvoiceNo`,
   `Description`,
   `Quantity`
FROM
   ecommerce_data.`data`
WHERE
   `Quantity` > (SELECT AVG(`Quantity`) FROM ecommerce_data.`data`
WHERE `Quantity` > 0);
```

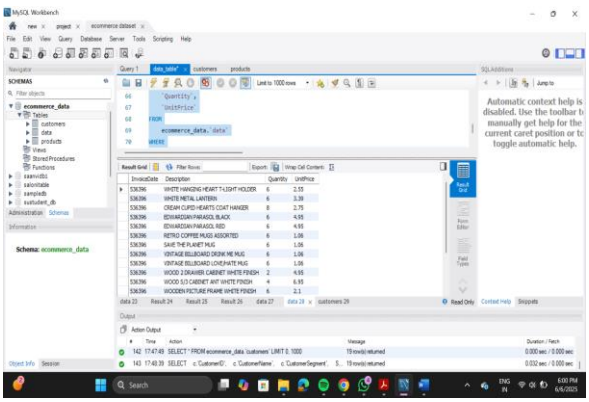


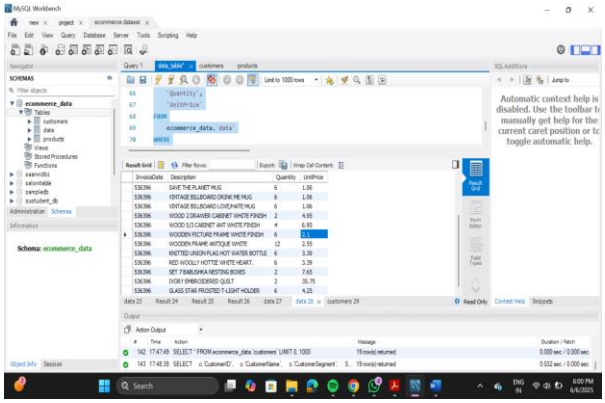




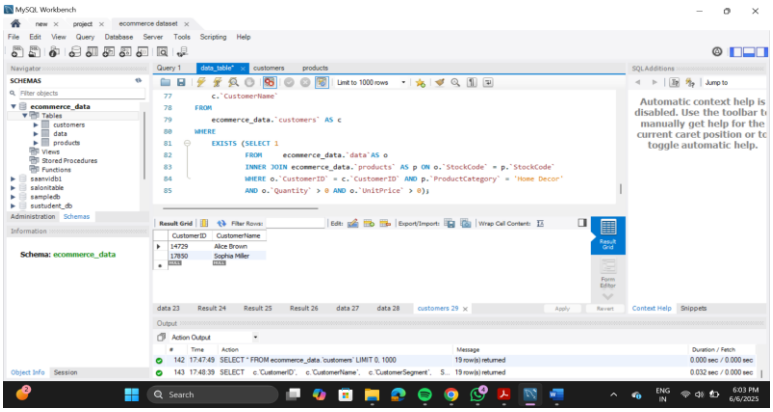
.....

```
-- Row Subquery
    SELECT
    `InvoiceNo`
    `InvoiceDate`,
    `Description`,
    `Quantity`,
    `UnitPrice`
FROM
    ecommerce_data.`data`
WHERE
    (`CustomerID`, `InvoiceDate`) = (SELECT `CustomerID`, MIN(`InvoiceDate`) FROM
ecommerce_data.`data`
 WHERE `CustomerID` = 17850);
```





```
-- Subquery with EXISTS
 SELECT DISTINCT
    c.`CustomerID`,
    c.`CustomerName`
FROM
    ecommerce_data.`customers` AS c
WHERE
    EXISTS (SELECT 1
            FROM     ecommerce_data.`data`AS o
            INNER JOIN ecommerce_data.`products` AS p ON o.`StockCode` = p.`StockCode`
            WHERE o.`CustomerID` = c.`CustomerID` AND p.`ProductCategory` = 'Home Decor'
            AND o.`Quantity` > 0 AND o.`UnitPrice` > 0);
```



VIEWS:

1. USE ecommerce_data;

```
CREATE VIEW `vw_daily_sales_summary` AS
SELECT
   DATE(`InvoiceDate`) AS SaleDate,
   SUM(`Quantity` * `UnitPrice`) AS DailyRevenue,
   COUNT(DISTINCT `InvoiceNo`) AS NumberOfOrders
FROM
   ecommerce_data.`data`
WHERE
   `Quantity` > 0 AND `UnitPrice` > 0
GROUP BY
   DATE(`InvoiceDate`)
ORDER BY
   SaleDate ASC;
```

2. USE ecommerce_data;

```
CREATE VIEW `vw_customer_spending_summary` AS
SELECT
   c.`CustomerID`,
   c.`CustomerName`,
   c.`CustomerSegment`,
```

```
   SUM(CASE WHEN o.`Quantity` > 0 AND o.`UnitPrice` > 0 THEN o.`Quantity` * o.`UnitPrice`
ELSE 0 END) AS TotalSpending,
   COUNT(DISTINCT o.`InvoiceNo`) AS NumberOfOrders,
   COUNT(DISTINCT o.`StockCode`) AS NumberOfUniqueProducts
FROM
   ecommerce_data.`customers` AS c
LEFT JOIN
   ecommerce_data.`data` AS o ON c.`CustomerID` = o.`CustomerID`
GROUP BY
   c.`CustomerID`,
   c.`CustomerName`,
   c.`CustomerSegment`;
```

3. USE ecommerce_data;

```
CREATE VIEW `vw_product_category_performance` AS
SELECT
   p.`ProductCategory`,
   SUM(o.`Quantity` * o.`UnitPrice`) AS TotalCategoryRevenue,
   COUNT(DISTINCT o.`StockCode`) AS NumberOfUniqueProductsSold
FROM ecommerce_data.`data` AS o
INNER JOIN
   ecommerce_data.`products` AS p ON o.`StockCode` = p.`StockCode`
WHERE
   o.`Quantity` > 0 AND o.`UnitPrice` > 0
GROUP BY
```

```
    p.`ProductCategory`
ORDER BY
    TotalCategoryRevenue DESC;
```

4.
```
USE  ecommerce_data;
CREATE VIEW `vw_high_value_items` AS
SELECT
    `StockCode`,
    `Description`,
    AVG(`UnitPrice`) AS `AverageUnitPrice`,
    COUNT(*) AS TotalTransactions
FROM
  ecommerce_data.`data`
WHERE
    `UnitPrice` > 100
    AND `Quantity` > 0
GROUP BY
    `StockCode`,
    `Description`
HAVING
    COUNT(*) > 5
ORDER BY
    AverageUnitPrice DESC;
```