1.First of making a database named joshinproject

Query: create database joshinmasaiproject;

USE joshinmasaiproject;

2. Then make table based on the database given :

First, create the necessary tables in your SQL database. You can start with the following SQL script:

CREATE TABLE Customers (

CustomerID INT PRIMARY KEY,

CustomerName VARCHAR(255),

Country VARCHAR(255)

);

CREATE TABLE Products (

ProductID INT PRIMARY KEY,

ProductName VARCHAR(255),

ProductCategory VARCHAR(255)

);

CREATE TABLE Transactions (

TransactionID INT PRIMARY KEY AUTO\_INCREMENT,

CustomerID INT,

ProductID INT,

PurchaseQuantity INT,

PurchasePrice DECIMAL(10, 2),

PurchaseDate DATE,

Country VARCHAR(255),

FOREIGN KEY (CustomerID) REFERENCES Customers(CustomerID),

FOREIGN KEY (ProductID) REFERENCES Products(ProductID)

);

**Step 2: Load Data into Tables**

Load the provided dataset into the respective tables. Use SQL INSERT statements or a bulk data import method if you have a large dataset.

**Step 3: Create Relationships**

Ensure the relationships between tables are correctly set up using foreign keys. This has been partially covered in the table creation script above.

**Step 4: Advanced Queries for Data Aggregation**

Create SQL queries to aggregate data as needed. Here are some example queries:

**Step 4: Re-run Aggregation Queries**

Now, re-run your aggregation queries to ensure they return the expected results.

**a.Total Purchases Per Customer**

SELECT

c.CustomerID,

c.CustomerName,

COALESCE(SUM(t.PurchaseQuantity \* t.PurchasePrice), 0) AS TotalSpent

FROM

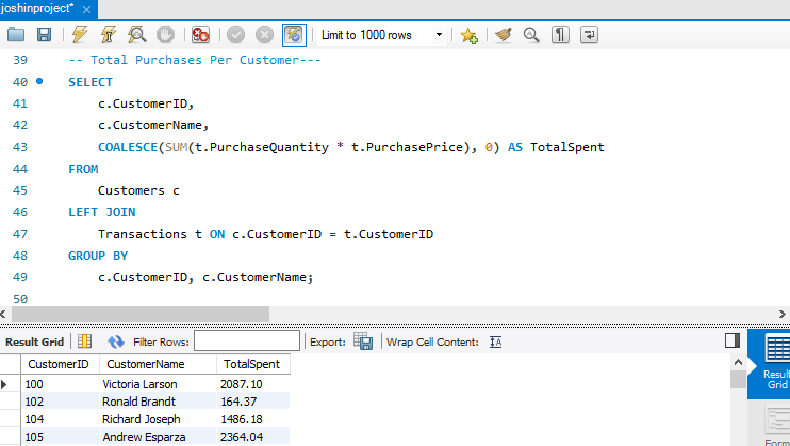
Customers c

LEFT JOIN

Transactions t ON c.CustomerID = t.CustomerID

GROUP BY

c.CustomerID, c.CustomerName;



**b. Total Sales Per Product**

SELECT

p.ProductID,

p.ProductName,

COALESCE(SUM(t.PurchaseQuantity \* t.PurchasePrice), 0) AS TotalSales

FROM

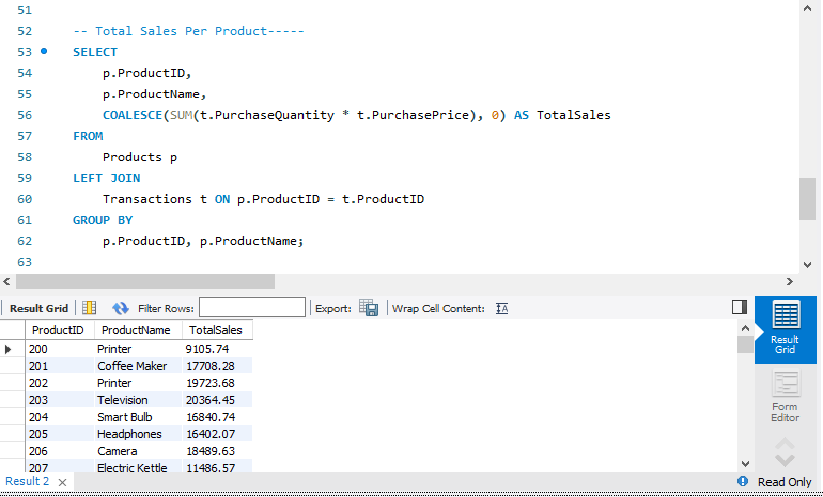
Products p

LEFT JOIN

Transactions t ON p.ProductID = t.ProductID

GROUP BY

p.ProductID, p.ProductName;



c. Total Sales Per Country

SELECT

c.Country,

COALESCE(SUM(t.PurchaseQuantity \* t.PurchasePrice), 0) AS TotalSales

FROM

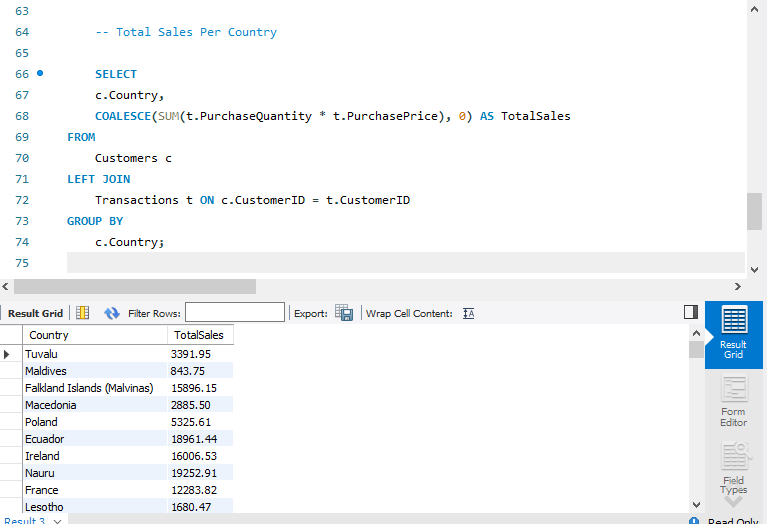
Customers c

LEFT JOIN

Transactions t ON c.CustomerID = t.CustomerID

GROUP BY

c.Country;



d. Total Purchases Per Category

SELECT

p.ProductCategory,

COALESCE(SUM(t.PurchaseQuantity \* t.PurchasePrice), 0) AS TotalSales

FROM

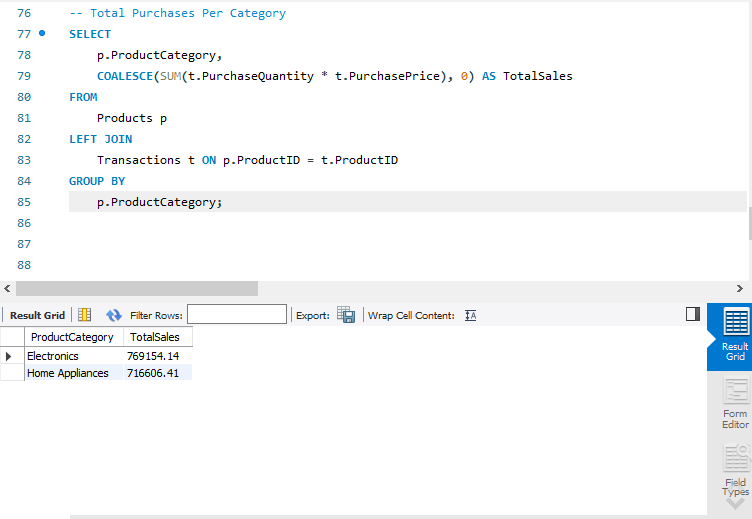
Products p

LEFT JOIN

Transactions t ON p.ProductID = t.ProductID

GROUP BY

p.ProductCategory;



e. Top 5 Customers by Total Spending

SELECT

c.CustomerID,

c.CustomerName,

SUM(t.PurchaseQuantity \* t.PurchasePrice) AS TotalSpent

FROM

Customers c

JOIN

Transactions t ON c.CustomerID = t.CustomerID

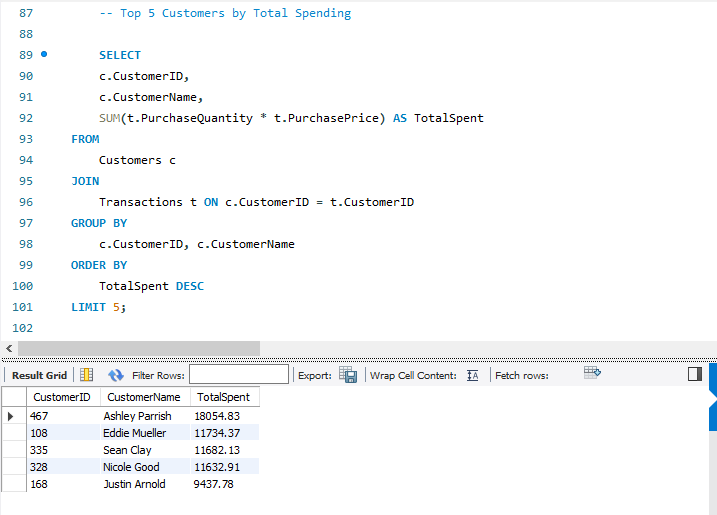
GROUP BY

c.CustomerID, c.CustomerName

ORDER BY

TotalSpent DESC

LIMIT 5;



f. **Top 5 Products by Sales**

SELECT

p.ProductID,

p.ProductName,

SUM(t.PurchaseQuantity \* t.PurchasePrice) AS TotalSales

FROM

Products p

JOIN

Transactions t ON p.ProductID = t.ProductID

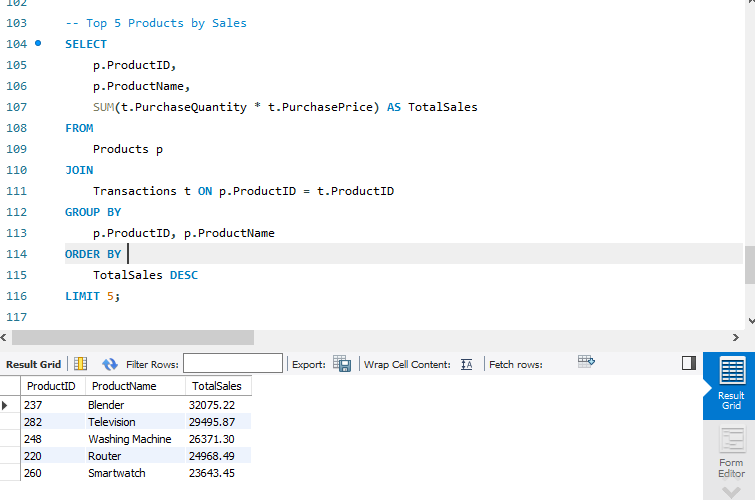
GROUP BY

p.ProductID, p.ProductName

ORDER BY

TotalSales DESC

LIMIT 5;



g. Monthly Sales Analysis

SELECT

DATE\_FORMAT(PurchaseDate, '%Y-%m') AS Month,

SUM(PurchaseQuantity \* PurchasePrice) AS TotalSales

FROM

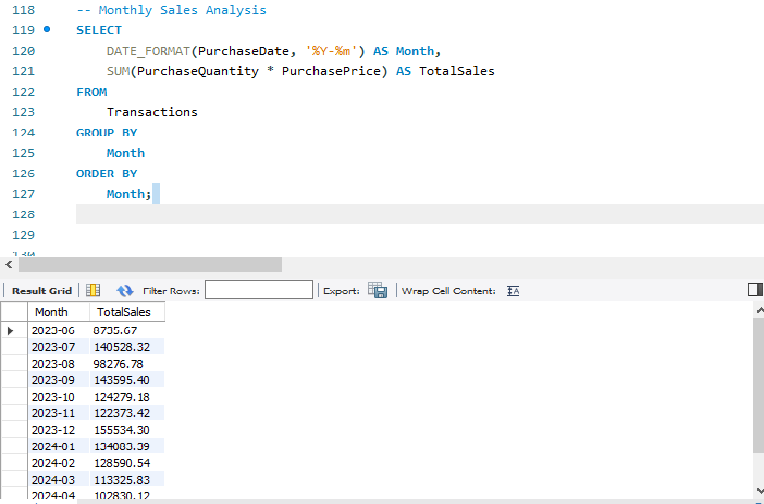
Transactions

GROUP BY

Month

ORDER BY

Month;



h. Customer Purchase Frequency

SELECT

c.CustomerID,

c.CustomerName,

COUNT(t.TransactionID) AS PurchaseCount

FROM

Customers c

JOIN

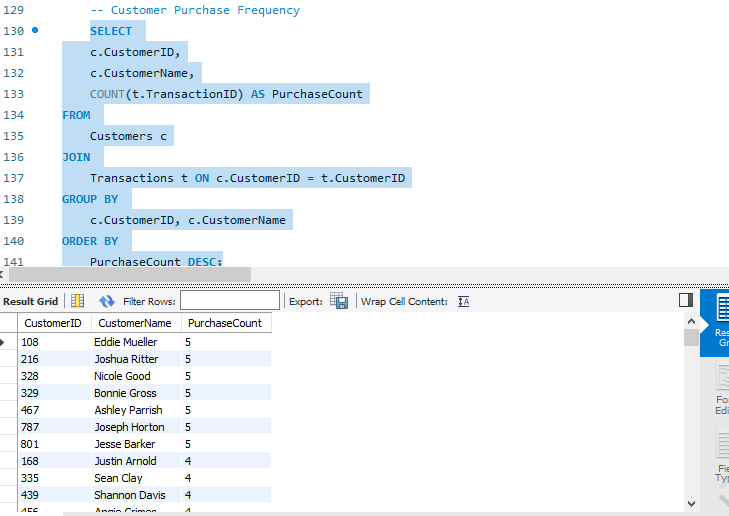
Transactions t ON c.CustomerID = t.CustomerID

GROUP BY

c.CustomerID, c.CustomerName

ORDER BY

PurchaseCount DESC;



i. Total Revenue per Country

SELECT

c.Country,

SUM(t.PurchaseQuantity \* t.PurchasePrice) AS TotalRevenue

FROM

Customers c

JOIN

Transactions t ON c.CustomerID = t.CustomerID

GROUP BY

c.Country

ORDER BY

TotalRevenue DESC;

