**SQL Project: Monday Coffee Data Analysis**

**Objective**

The goal of this project is to analyse Monday Coffee's sales data, a

company that has been selling its products online since January 2023, and

to recommend the top three major cities in India for opening new coffee

shop locations based on consumer demand and sales performance.

**Activity 1: Data Loading**

**How do you create the tables for customers, products, and sales in the database?**

**Customers Table**

CREATE TABLE Customers (

CustomerID INT PRIMARY KEY AUTO\_INCREMENT,

FirstName VARCHAR(50),

LastName VARCHAR(50),

Email VARCHAR(100) UNIQUE,

Phone VARCHAR(20),

Address VARCHAR(255),

City VARCHAR(50),

Country VARCHAR(50),

CreatedAt TIMESTAMP DEFAULT CURRENT\_TIMESTAMP

);

**Products Table**

CREATE TABLE Products (

ProductID INT PRIMARY KEY AUTO\_INCREMENT,

ProductName VARCHAR(100) NOT NULL,

Category VARCHAR(50),

Price DECIMAL(10,2) NOT NULL,

StockQuantity INT DEFAULT 0,

CreatedAt TIMESTAMP DEFAULT CURRENT\_TIMESTAMP

);

**Sales Table**

CREATE TABLE Sales (

SaleID INT PRIMARY KEY AUTO\_INCREMENT,

CustomerID INT,

ProductID INT,

Quantity INT NOT NULL,

SaleDate TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

TotalAmount DECIMAL(10,2) GENERATED ALWAYS AS (Quantity \* (SELECT Price FROM Products WHERE Products.ProductID = Sales.ProductID)) STORED,

**Foreign Key Relations**

FOREIGN KEY (CustomerID) REFERENCES Customers(CustomerID),

FOREIGN KEY (ProductID) REFERENCES Products(ProductID)

);

**Activity 2: Data Cleaning & Preprocessing**

● **How can you identify null values in your dataset?**

Check nulls in each column of Customers

SELECT

SUM(CASE WHEN FirstName IS NULL THEN 1 ELSE 0 END) AS Null\_FirstName,

SUM(CASE WHEN LastName IS NULL THEN 1 ELSE 0 END) AS Null\_LastName,

SUM(CASE WHEN Email IS NULL THEN 1 ELSE 0 END) AS Null\_Email

FROM Customers;

For Sales table

SELECT \*

FROM Sales

WHERE CustomerID IS NULL OR ProductID IS NULL OR Quantity IS NULL;

**● How can you check for duplicate entries in the customers table?**

To find exact duplicates (based on Email, FirstName, LastName)

SELECT Email, FirstName, LastName, COUNT(\*) AS DupCount

FROM Customers

GROUP BY Email, FirstName, LastName

HAVING COUNT(\*) > 1;

**● How do you check for mismatches between total\_amount and the calculated value of**

**price × quantity?**

SELECT s.SaleID, s.TotalAmount, (p.Price \* s.Quantity) AS ExpectedAmount

FROM Sales s

JOIN Products p ON s.ProductID = p.ProductID

WHERE s.TotalAmount <> (p.Price \* s.Quantity);

**Activity 3: Data Transformation & Integration**

Once the data was clean, we proceeded to join the tables to create a comprehensive view for

analysis.

**● How do you create a comprehensive sales report with customer and product details?**

SELECT

s.SaleID,

s.SaleDate,

c.CustomerID,

CONCAT(c.FirstName, ' ', c.LastName) AS CustomerName,

c.Email,

p.ProductName,

p.Category,

s.Quantity,

p.Price,

(s.Quantity \* p.Price) AS CalculatedAmount,

s.TotalAmount

FROM Sales s

JOIN Customers c ON s.CustomerID = c.CustomerID

JOIN Products p ON s.ProductID = p.ProductID

ORDER BY s.SaleDate DESC;

**Activity 4: Data Analysis & Aggregation**

**(a) Total Sales per City**

**● What are the total sales per city?**

SELECT c.City, SUM(s.TotalAmount) AS TotalSales

FROM Sales s

JOIN Customers c ON s.CustomerID = c.CustomerID

GROUP BY c.City

ORDER BY TotalSales DESC;

**(b) Total Transactions per City**

**● How many total transactions occurred per city?**

SELECT c.City, COUNT(s.SaleID) AS TotalTransactions

FROM Sales s

JOIN Customers c ON s.CustomerID = c.CustomerID

GROUP BY c.City;

**(c) Unique Customers per City**

**● How many unique customers are there in each city?**

SELECT c.City, COUNT(DISTINCT s.CustomerID) AS UniqueCustomers

FROM Sales s

JOIN Customers c ON s.CustomerID = c.CustomerID

GROUP BY c.City;

**(d) Average Order Value per City**

**● What is the average order value per city?**

SELECT c.City, AVG(s.TotalAmount) AS AvgOrderValue

FROM Sales s

JOIN Customers c ON s.CustomerID = c.CustomerID

GROUP BY c.City;

**(e) Product Demand per City**

**● What is the demand for each product in different cities?**

SELECT c.City, p.ProductName, SUM(s.Quantity) AS TotalDemand

FROM Sales s

JOIN Customers c ON s.CustomerID = c.CustomerID

JOIN Products p ON s.ProductID = p.ProductID

GROUP BY c.City, p.ProductName

ORDER BY c.City, TotalDemand DESC;

**(f) Monthly Sales Trend**

**● What is the monthly sales trend?**

SELECT DATE\_FORMAT(s.SaleDate, '%Y-%m') AS Month, SUM(s.TotalAmount) AS MonthlySales

FROM Sales s

GROUP BY DATE\_FORMAT(s.SaleDate, '%Y-%m')

ORDER BY Month;

**(g) Customer Rating Analysis**

**● What is the average product rating per city based on customer purchases?**

SELECT c.City, AVG(s.Rating) AS AvgProductRating

FROM Sales s

JOIN Customers c ON s.CustomerID = c.CustomerID

GROUP BY c.City

ORDER BY AvgProductRating DESC;

**Activity 5: Decision-Making & Recommendations**

**(a) Top Cities Selection**

**● How do you identify the top 3 cities based on sales, unique customers, and order count?**

Top 3 Cities by Sales

SELECT c.City, SUM(s.TotalAmount) AS TotalSales

FROM Sales s

JOIN Customers c ON s.CustomerID = c.CustomerID

GROUP BY c.City

ORDER BY TotalSales DESC

LIMIT 3;

Top 3 Cities by Unique Customers

SELECT c.City, COUNT(DISTINCT s.CustomerID) AS UniqueCustomers

FROM Sales s

JOIN Customers c ON s.CustomerID = c.CustomerID

GROUP BY c.City

ORDER BY UniqueCustomers DESC

LIMIT 3;

Top 3 Cities by Order Count

SELECT c.City, COUNT(s.SaleID) AS OrderCount

FROM Sales s

JOIN Customers c ON s.CustomerID = c.CustomerID

GROUP BY c.City

ORDER BY OrderCount DESC

LIMIT 3;

**(b) Final Recommendations**

**● What are the final recommendations for expanding Monday Coffee shops?**

Based on the analysis (Activities 1–4), the **recommendations** would look like:

**Expand in High-Potential Cities**

Focus on the **top 3 cities** that show strong performance across **sales, customers, and transactions**.

These cities not only generate revenue but also have a loyal and diverse customer base.

**Leverage Product Demand Insights**

Expand menu offerings for **high-demand products** in each city (identified in Activity 4e).

Localize inventory management to match city-specific preferences.

**Enhance Customer Experience**

Target cities with **high average order value** by offering premium coffee experiences and loyalty programs.

Focus on cities with **strong ratings** (Activity 4g) to build reputation further.

**Seasonal & Trend Alignment**

Use **monthly sales trend analysis** to identify peak seasons (Activity 4f).

Align promotions, staffing, and inventory with these demand cycles.

**Data-Driven Expansion Strategy**

Roll out **pilot stores** in top cities first, measure performance, then scale gradually.

Keep monitoring **customer feedback (ratings)** to adapt offerings.