To walk you through the project, we will break it down into the following steps. I'll provide explanations and SQL code where necessary:

**Step 1: Data Wrangling (Data Inspection and Cleaning)**

1. **Inspect the data**: This helps understand the data's structure, types of values, and any potential issues like missing or NULL values.

sql

Copy code

-- Inspect the data

SELECT TOP 10 \*

FROM WalmartSalesData;

1. **Detect NULL values**: This will identify any missing or NULL values, which can be addressed using appropriate techniques (though in this case, the table has NOT NULL constraints).

sql

Copy code

-- Select columns with NULL values

SELECT \*

FROM WalmartSalesData

WHERE [column\_name] IS NULL;

1. **Replace NULL values** (if there are any): Example: For categorical data, you might replace NULL values with "Unknown" or for numerical data with 0.

sql

Copy code

-- Update NULL values in a column

UPDATE WalmartSalesData

SET [column\_name] = 'Unknown'

WHERE [column\_name] IS NULL;

**Step 2: Build a Database and Insert Data**

Assuming you already have the data, let’s create the table and insert the data into the database.

1. **Create a database** (if not already created):

sql

Copy code

CREATE DATABASE WalmartSalesDB;

1. **Create the table**: Here's an example table structure based on common columns that you might have:

sql

Copy code

CREATE TABLE WalmartSalesData (

TransactionID INT PRIMARY KEY,

Date DATE NOT NULL,

Branch VARCHAR(100) NOT NULL,

ProductCategory VARCHAR(100) NOT NULL,

Sales DECIMAL(10, 2) NOT NULL,

Profit DECIMAL(10, 2) NOT NULL,

Quantity INT NOT NULL,

CustomerType VARCHAR(50) NOT NULL,

PaymentType VARCHAR(50) NOT NULL,

-- other columns here (TotalCost, etc.)

);

1. **Insert the data**: This would be done either via bulk insertion or manually:

sql

Copy code

-- Example insertion

INSERT INTO WalmartSalesData (TransactionID, Date, Branch, ProductCategory, Sales, Profit, Quantity, CustomerType, PaymentType)

VALUES (1, '2023-09-15', 'Mandalay', 'Groceries', 250.50, 50.00, 5, 'Regular', 'Cash');

**Step 3: Feature Engineering**

1. **Add a time\_of\_day column**: This will categorize the time of sales (Morning, Afternoon, Evening). You might first need to split the time from the date.

sql

Copy code

-- Add the column first

ALTER TABLE WalmartSalesData ADD time\_of\_day VARCHAR(50);

-- Update the time\_of\_day based on sales time

UPDATE WalmartSalesData

SET time\_of\_day = CASE

WHEN DATEPART(HOUR, [Date]) BETWEEN 6 AND 12 THEN 'Morning'

WHEN DATEPART(HOUR, [Date]) BETWEEN 12 AND 17 THEN 'Afternoon'

ELSE 'Evening'

END;

1. **Add a day\_name column**: Extract the day of the week.

sql

Copy code

-- Add the column

ALTER TABLE WalmartSalesData ADD day\_name VARCHAR(50);

-- Update the day\_name based on the transaction date

UPDATE WalmartSalesData

SET day\_name = DATENAME(WEEKDAY, [Date]);

1. **Add a month\_name column**: Extract the month from the date.

sql

Copy code

-- Add the column

ALTER TABLE WalmartSalesData ADD month\_name VARCHAR(50);

-- Update the month\_name based on the transaction date

UPDATE WalmartSalesData

SET month\_name = DATENAME(MONTH, [Date]);

**Step 4: Exploratory Data Analysis (EDA)**

You’ll now start analyzing the data based on the features created. Some common analysis could include:

1. **Top performing branches**:

sql

Copy code

-- Find the total sales per branch

SELECT Branch, SUM(Sales) AS TotalSales

FROM WalmartSalesData

GROUP BY Branch

ORDER BY TotalSales DESC;

1. **Sales trend of different products**:

sql

Copy code

-- Get the sales trend for each product category

SELECT ProductCategory, month\_name, SUM(Sales) AS MonthlySales

FROM WalmartSalesData

GROUP BY ProductCategory, month\_name

ORDER BY month\_name;

1. **Customer behavior (based on time of day)**:

sql

Copy code

-- Analyze sales per time\_of\_day

SELECT time\_of\_day, COUNT(\*) AS TransactionCount, SUM(Sales) AS TotalSales

FROM WalmartSalesData

GROUP BY time\_of\_day

ORDER BY TotalSales DESC;

1. **Busiest day of the week for each branch**:

sql

Copy code

-- Find the busiest day of the week by branch

SELECT Branch, day\_name, COUNT(\*) AS TransactionCount

FROM WalmartSalesData

GROUP BY Branch, day\_name

ORDER BY TransactionCount DESC;

1. **Month with the most sales and profit**:

sql

Copy code

-- Analyze sales and profit by month

SELECT month\_name, SUM(Sales) AS TotalSales, SUM(Profit) AS TotalProfit

FROM WalmartSalesData

GROUP BY month\_name

ORDER BY TotalSales DESC;

These steps should help you get started with your analysis and give you insights into customer behavior, sales trends, and optimization strategies. Let me know if you need further clarifications or code adjustments!