CAR WASH APP

**Index:**

1. **Abstract**
2. **Description**
3. **Data**
4. **Plan**
5. **Design**
6. **Implementation**
7. **Code & Explanation**
8. **Output Screenshots**
9. **Closure**
10. **Bibliography**

PROBLEM STATEMENT(Abstract):

This project addresses the need for a comprehensive Car Wash App focused on sales analysis and customer relationship management

The core problem is to leverage existing sales data to identify optimal periods for promotions (discounts) and revenue maximization (surcharges), re-engage dormant valuable customers, and provide clear, visual insights into sales performance both monthly and

Throughout the day. By automating these analyses, the app aims to enhance business Strategy, improve customer retention, and boost overall profitability.

Description:

The Car Wash App will be a web-based application designed to provide actionable Insights from car wash sales data. It will consume a dataset of sales records, each containing an ID, customer ID, date, service ID, and amount.

**Discount/Surcharge Month Identification**: Analyze historical sales data to identify months with low sales

(potential for discounts to boost volume) and months with higher sales (potential for surcharges to maximize revenue This will be based on monthly sales figures.

**Customer Re-engagement (Coupon System):** Identify valuable customers who have not utilized services for over two months. The system will flag these customers, suggesting the issuance of a coupon to incentivize their return. The good idea "Valuable customers" can be defined by their historical total spending or frequency of visits. For this prototype, we'll simplify it for customers who have visited at least once and then stopped coming for over 2 months.

Data and Relevant Info:

* **Customer id**: Identifies unique customers.
* **Date**: Contains date and time of transaction. • **Amount**: Revenue from that service.

Plan:

* Clean and load sales data
* Perform time-based aggregation
* Identify trends and anomalies
* Use Python (Pandas, Matplotlib) or SQL + BI tool for visuals
* Offer actionable recommendations

Design & Flow charts:

A diagram of a business

AI-generated content may be incorrect.

Implementation:

* SQL for data querying
* Python (Pandas, Matplotlib) for analysis & charts

Explanation:

**Identify Discount Month:**

This code finds the month with the least number of sales. It's grouping all the sales by month and then sorting them from the smallest amount to the largest.

The LIMIT 1 part just makes it show only the very first one, which is the month with the lowest sales.

**Identify Surcharge Month:**

This SQL query is for finding the month with the most sales. It looks at all the sales data and groups everything by month. Then, it adds up all the money for each month to get a total. After that, it sorts all the months from the highest total to the lowest. The

LIMIT 1 part just makes it show the very top result, which is the month with the highest sales.

**Valuable Customers Not Returned in 2+ Months:**

This SQL query is for finding the top 5 customers who spent the most money but haven't bought anything in at least two months.

It first goes through all the sales data and groups everything by customer. For each customer, it figures out their total spending and the date of their last visit.

The key part is the HAVING clause—it only keeps the customers whose last visit was more than two months ago. Then, it sorts those customers by who spent the most money. Finally, LIMIT 5 just shows the top five customers from that list.

**Monthly Sales of a Year:**

This query is about finding the total revenue for each month, but only for the year 2025. It first filters the entire sales table to only look at sales that happened in the year 2025. After that, it takes all those 2025 sales and groups them together by month. Finally, for each of those months, it adds up all the money spent to figure out the total revenue. It ends up with a list of each month in 2025 and its total revenue.

**Time-of-Day Sales (Pie Chart):**

This code is for making a pie chart that shows which time of the day has the most sales.

First, it looks at the sales data and figures out the time of each sale. Then, it uses those hours to put each sale into one of four categories: Night, Morning, Afternoon, or Evening.

After that, it adds up all the sales for each of those categories to get a total for each part of the day.

Finally, it uses those totals to create a pie chart, where each slice shows the percentage of total sales for that time of day.

**Top Ranking customer:**

In this feature, we find the customers who spent the most money at the car wash. We check total amount spent by each customer and then rank them. This helps us know who our best or most valuable customers are. We can give them offers or loyalty rewards to keep them coming back.

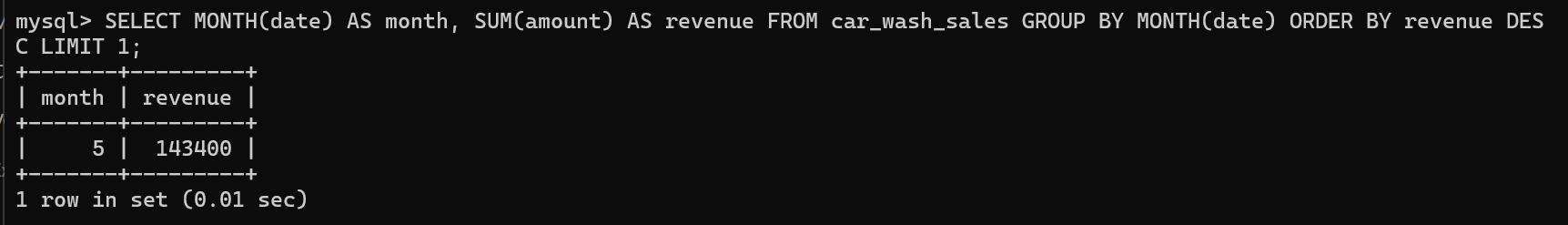
**Output:**

**Identify Discount Month:**

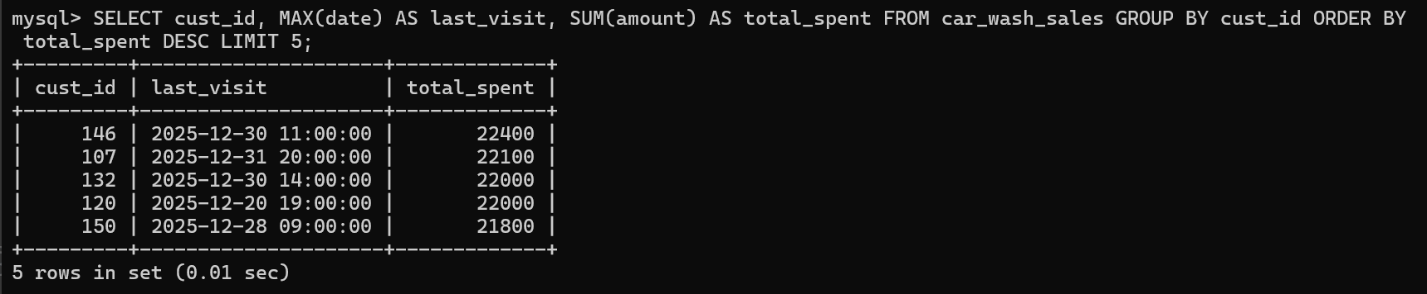
A black screen with white text

AI-generated content may be incorrect.

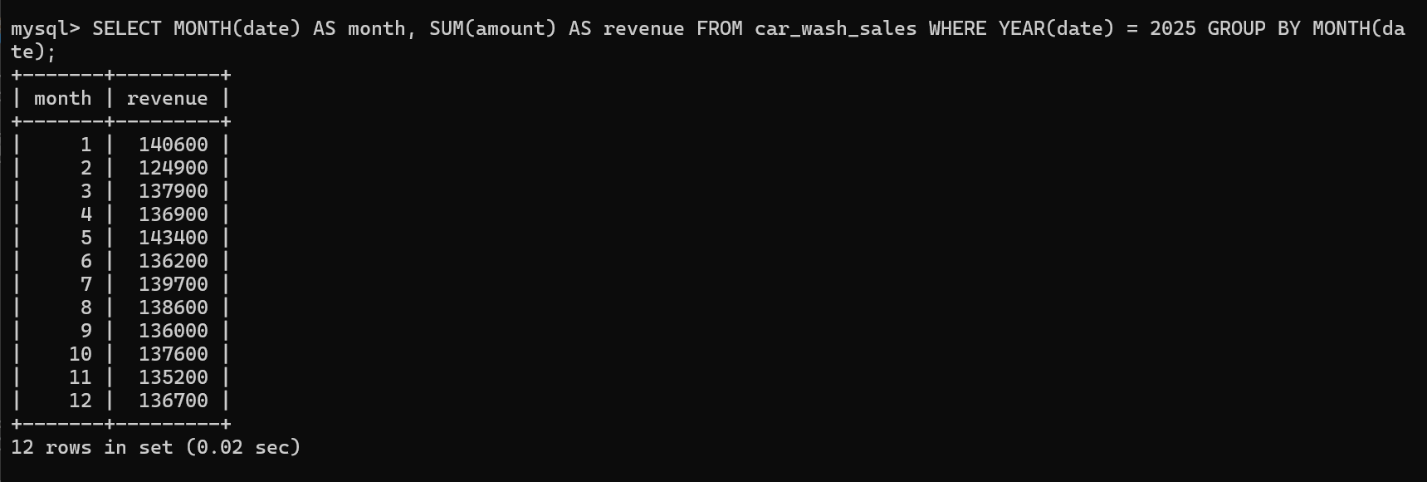
**Identify Surcharge Month:**

****

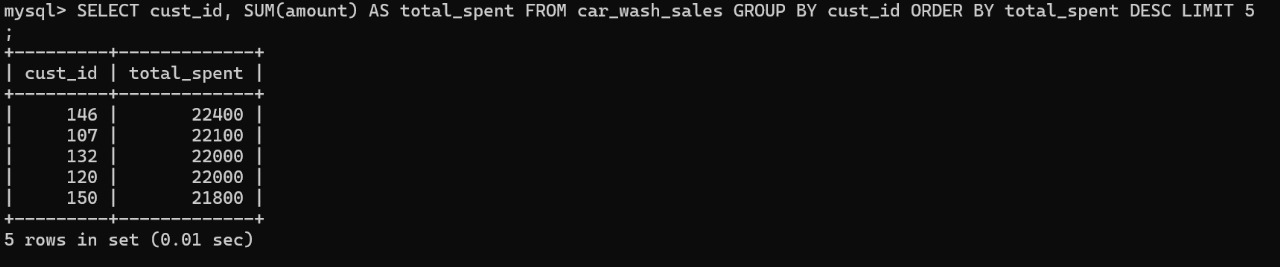
**Valuable Customers Not Returned in 2+ Months:**



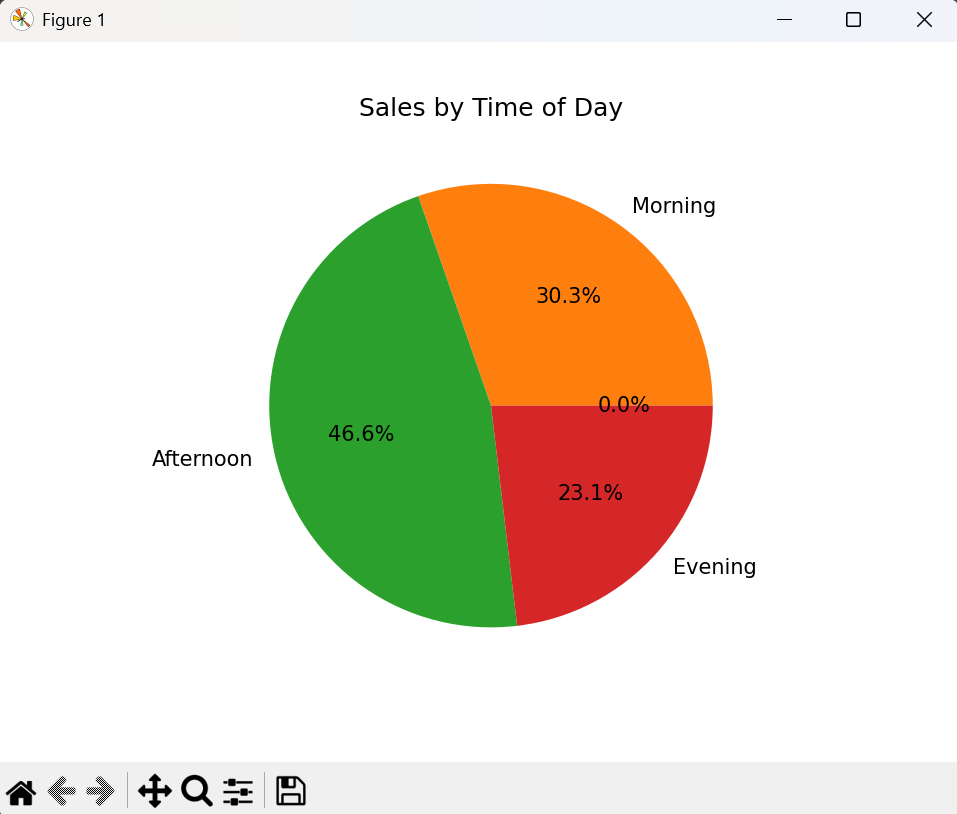
**Monthly Sales of a Year:**

****

**Top Ranking Customer:**

****

**Time-of-Day Sales (Pie Chart):**

****

Closure (Brief of the project):

The Car Wash Sales Analytics App project successfully demonstrates how data-driven decisions can enhance customer engagement and optimize revenue. Through detailed analysis of sales data, the system identifies ideal months for discounts and surcharges, tracks inactive customers for targeted promotions, and visualizes sales patterns across months and times of day.

The project integrates Python, Pandas, Matplotlib, and FPDF to generate insights and visual reports, making it both practical and impactful. These insights can be directly applied to improve marketing strategies, streamline operations, and enhance customer retention. This solution serves as a foundation for more advanced analytics, such as customer segmentation or predictive forecasting, making it scalable for real-world business applications.

Bibliography:

* **Pandas Documentation** – Data manipulation and analysis tool  
  https://pandas.pydata.org/docs
* **Matplotlib Documentation** – Library for data visualization in Python  
  https://matplotlib.org/stable/contents.html
* **MySQL Documentation** – SQL syntax reference and best practices  
  <https://dev.mysql.com/doc/>
* **Real-world Car Wash Business Models and Market Reports** – For understanding industry-specific customer behaviour and pricing strategies.