**Uber Supply-Demand Gap Analysis**

**By:- Tushar Garg**

**Date:- 21/06/2025**

**Project Overview**

**Problem Statement:-**

* This project aims to analyze Uber ride request data to identify the patterns and causes of ride failures, such as driver cancellations and unavailability of cars.
* Experiences fluctuations in ride requests across different times of the day and locations. A major challenge the company faces is the supply-demand mismatch, where user demand often exceeds the number of available drivers, leading to unfulfilled ride requests, customer dissatisfaction, and lost revenue opportunities.
* By uncovering when (time slots), where (pickup points), and why (status) these issues occur, the project seeks to provide data-driven insights to help Uber

**Business Objective:-**

* The primary business objective of this project is to identify and understand the supply-demand gap in Uber ride requests, with the goal of improving ride fulfillment rates and enhancing customer satisfaction.
* This involves analyzing when, where, and why Uber is unable to meet user demand—whether due to unavailability of drivers, high cancellation rates, or operational inefficiencies.

**Dataset Description:-**

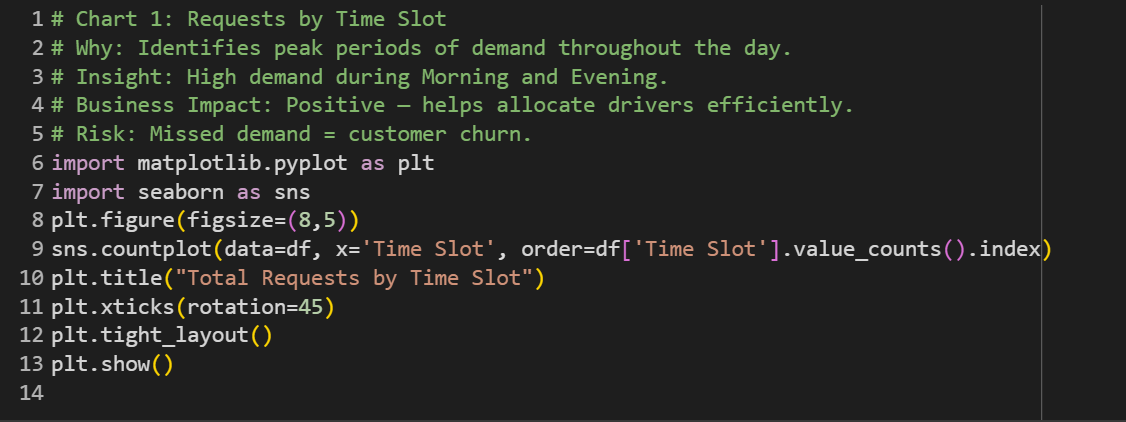
* The dataset contains detailed ride request logs from Uber, focusing on pickup and drop patterns, driver availability, and service status over a specific period.
* After loading and cleaning the dataset, I performed structural and exploratory analysis to understand its composition and key issues.

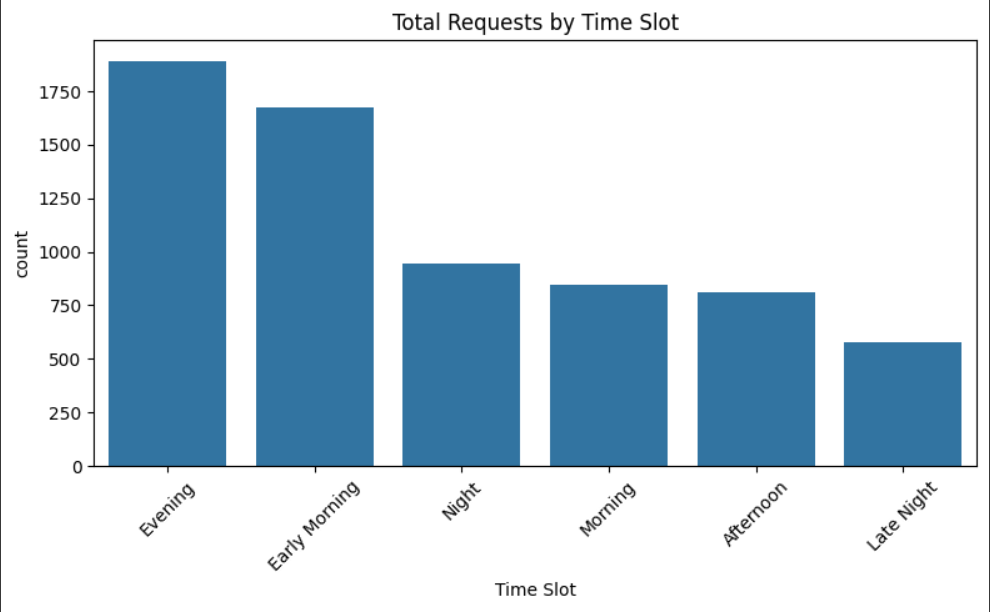
**Variables Description:-**

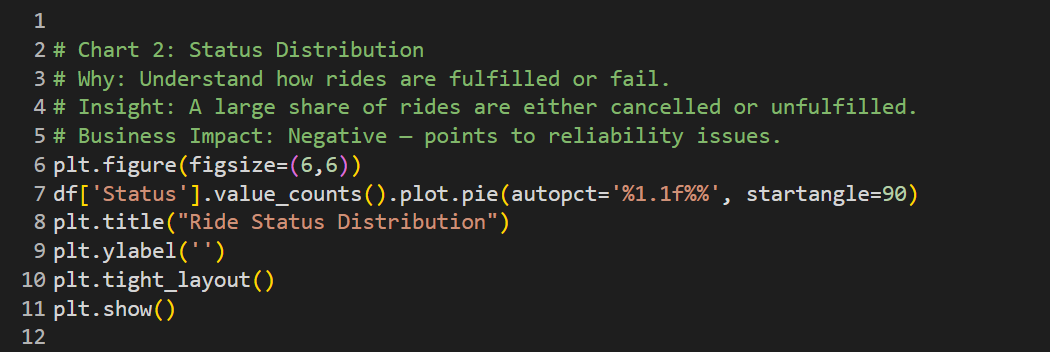
* Request timestamp-The exact date and time when the user requested a ride. Used to extract Hour, Day, and Time Slot.
* Drop timestamp - The date and time when the ride was completed. If missing, the ride was not fulfilled.
* Driver id - Unique identifier of the driver assigned to the ride. Blank if no driver was available.
* Pickup point- The location where the ride was requested: either City or Airport.
* Status- The final status of the ride

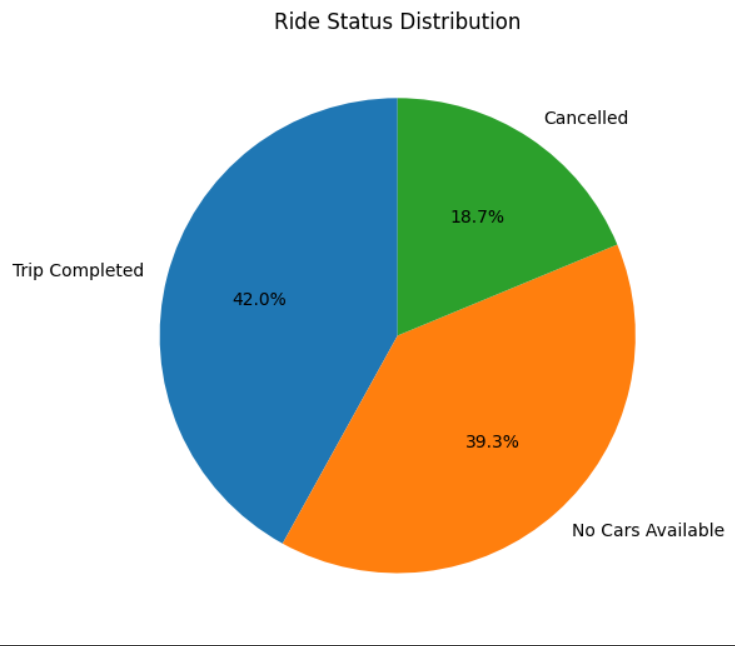
**Python-Based EDA & Charts**

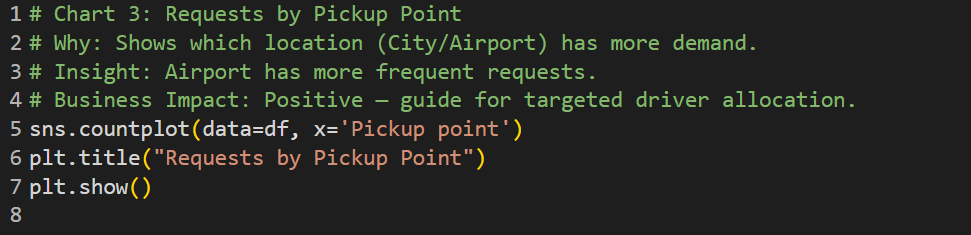
1. **Univariate Charts**
2. **Bivariate Charts**
3. **Multivariate Charts**
4. **Univariate Charts**

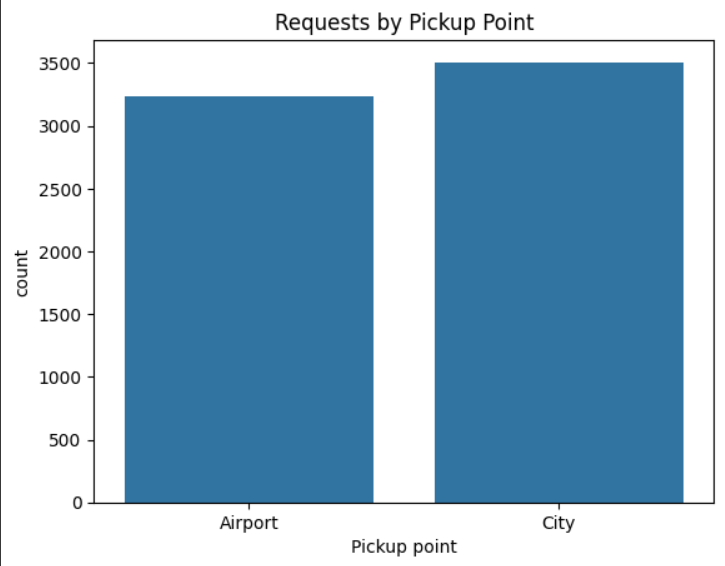
****

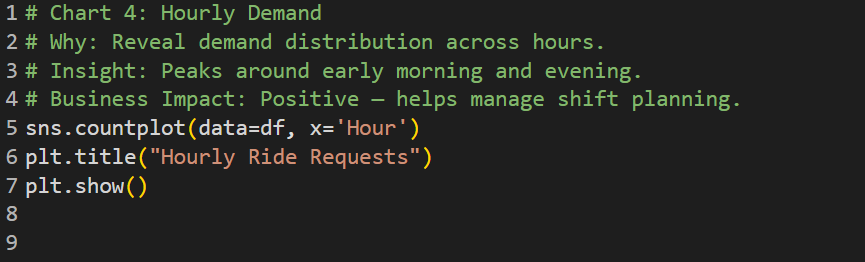


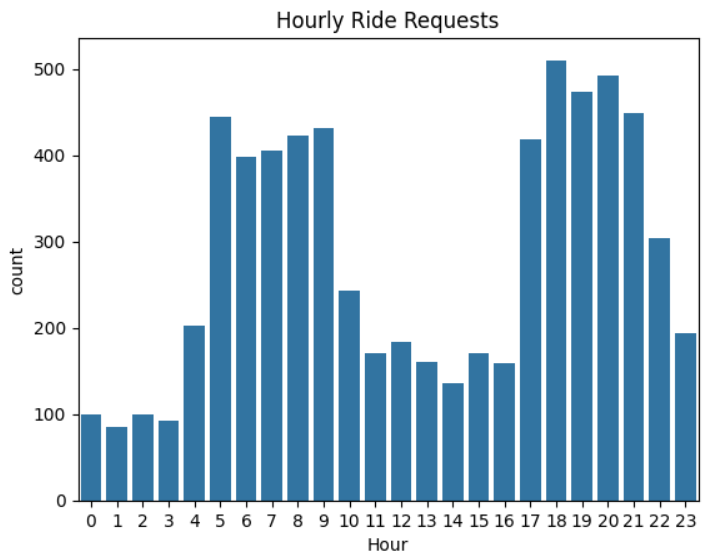


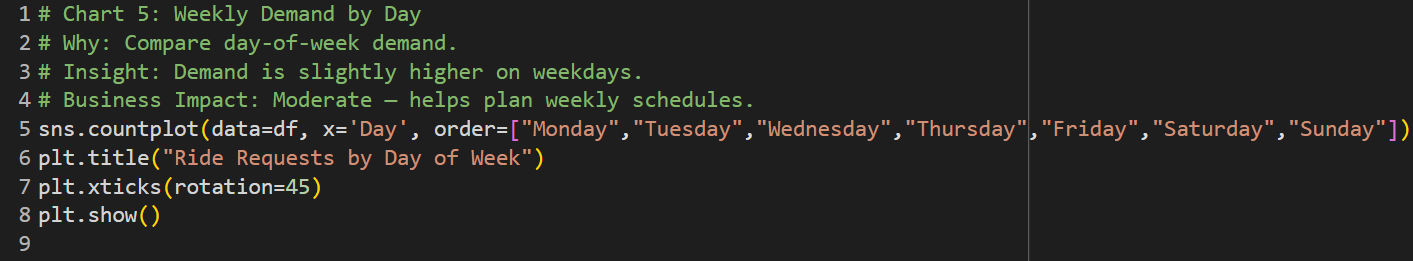


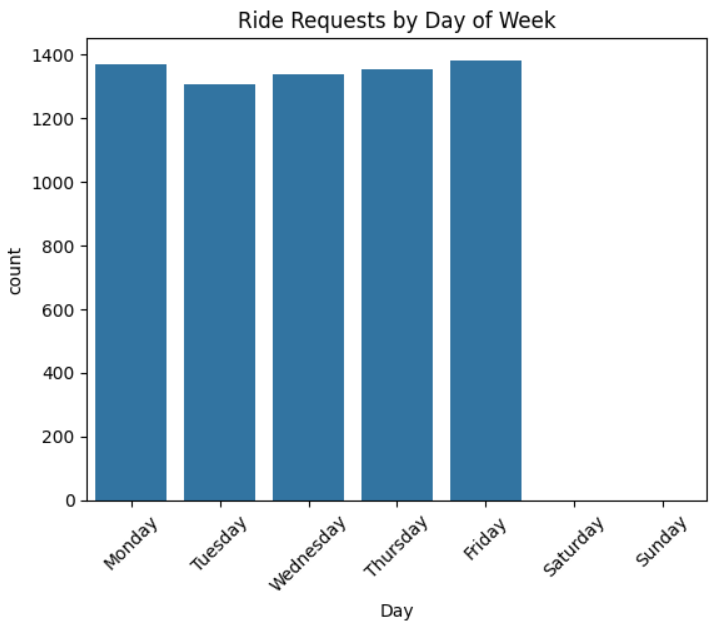


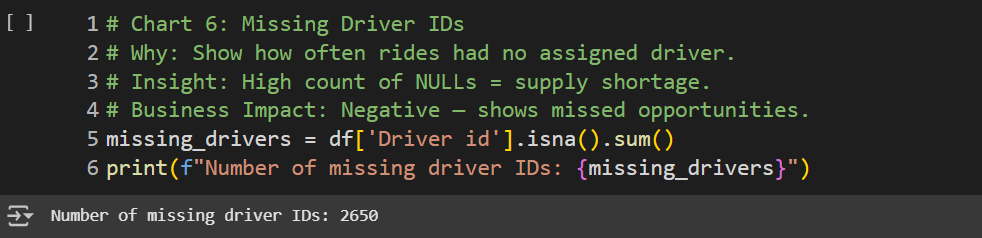


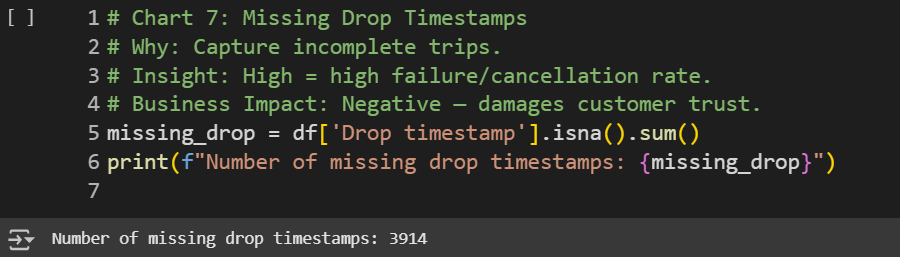


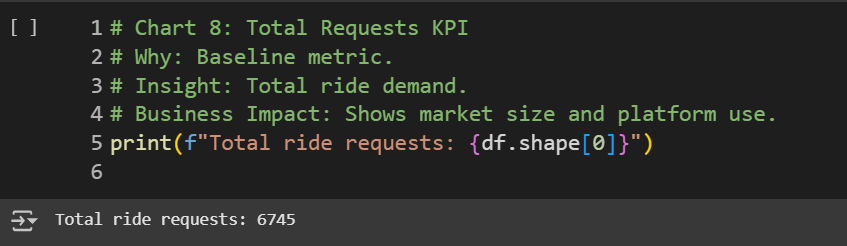




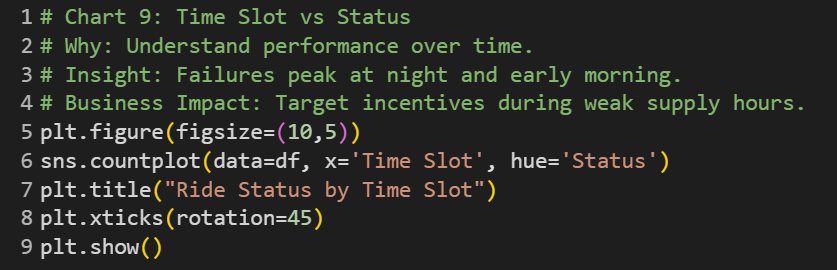


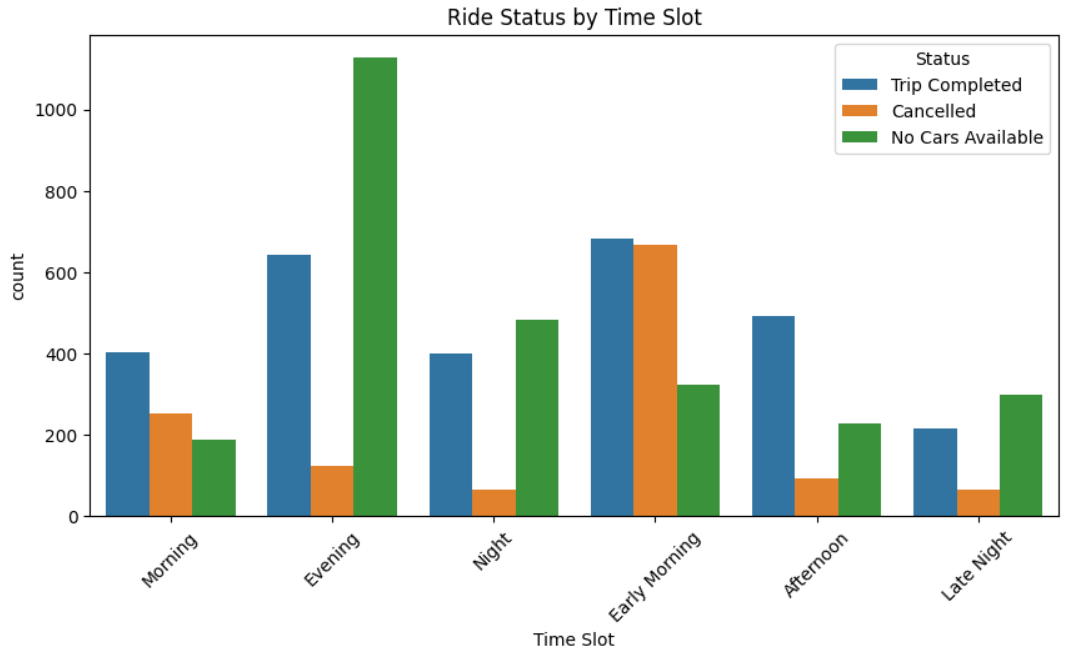


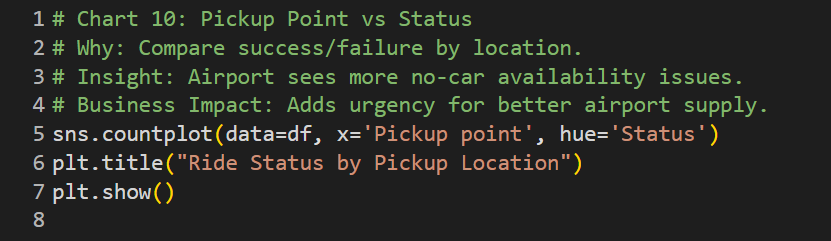


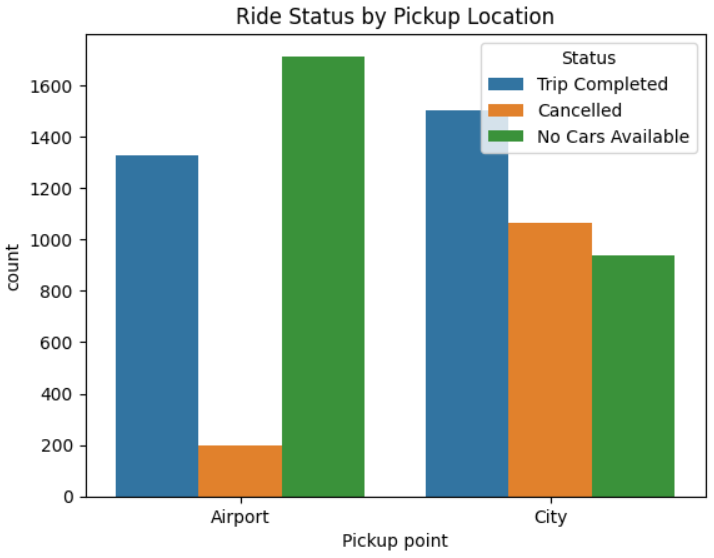


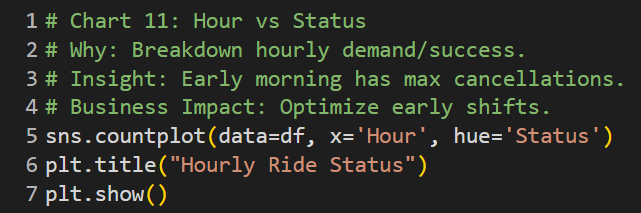
**2. Bivariate Charts**

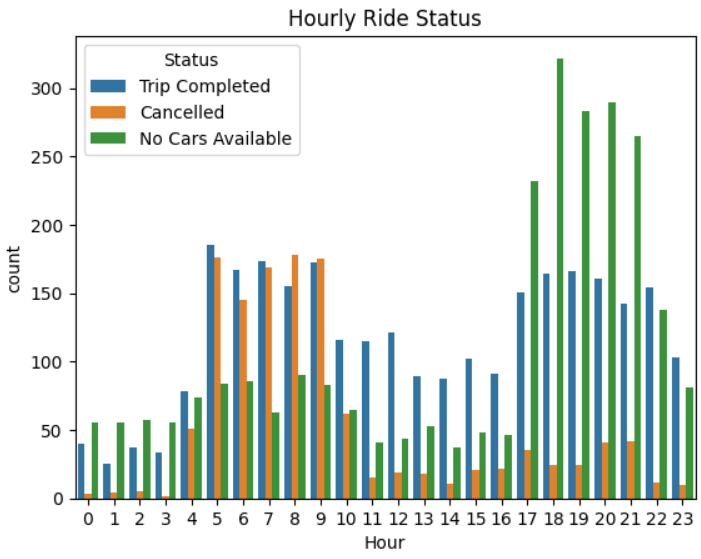
****

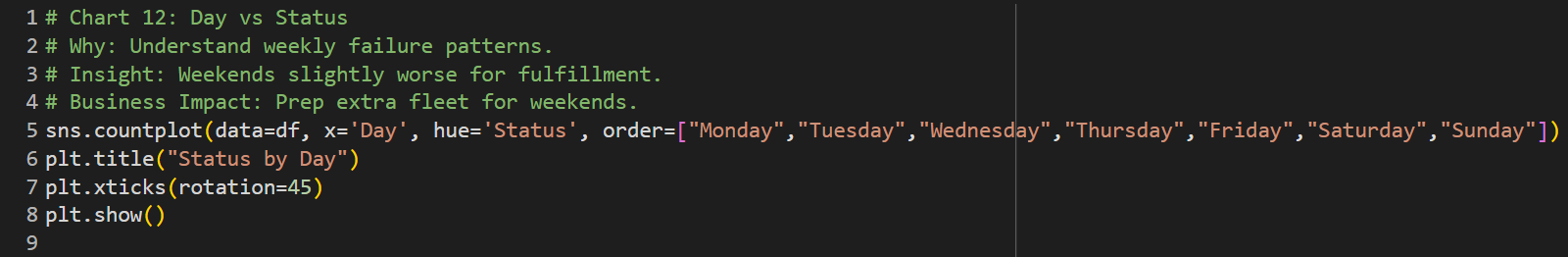
****

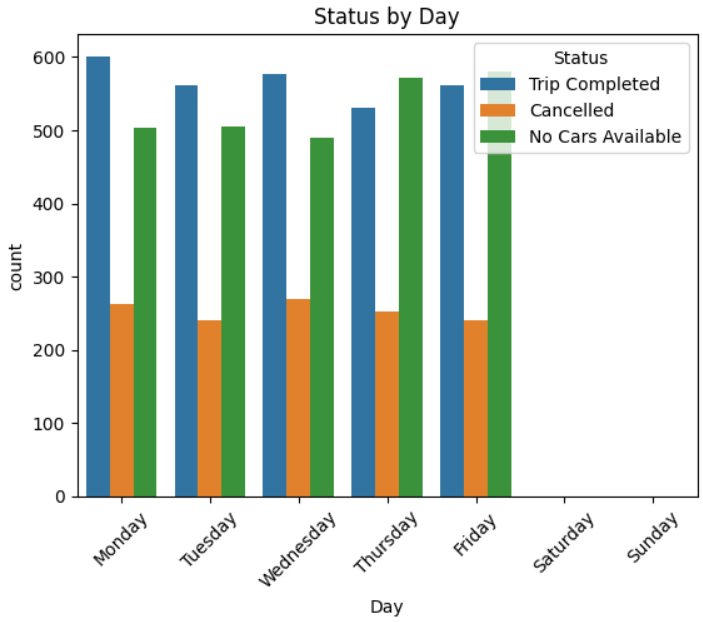
****

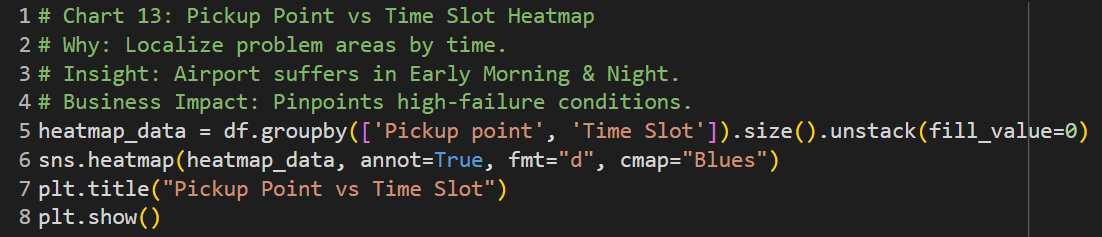
****

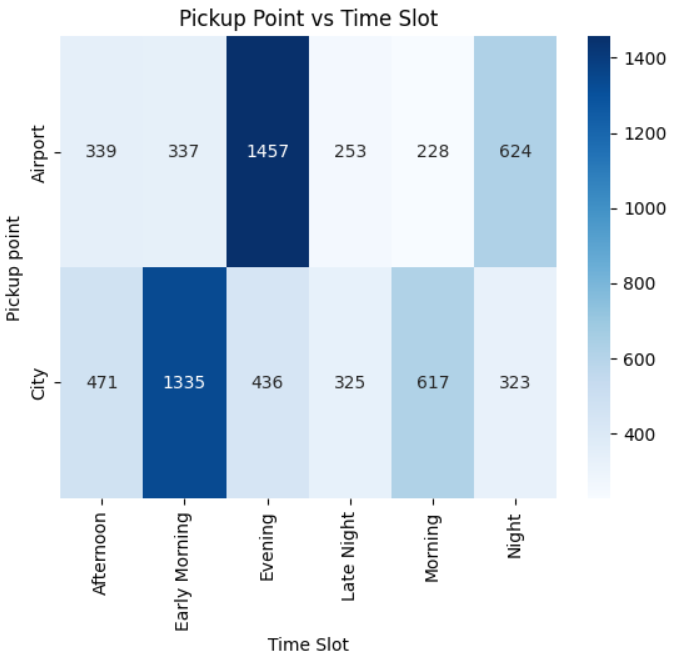
****

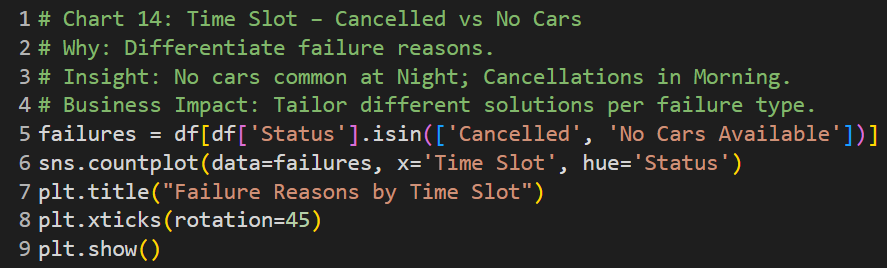
****

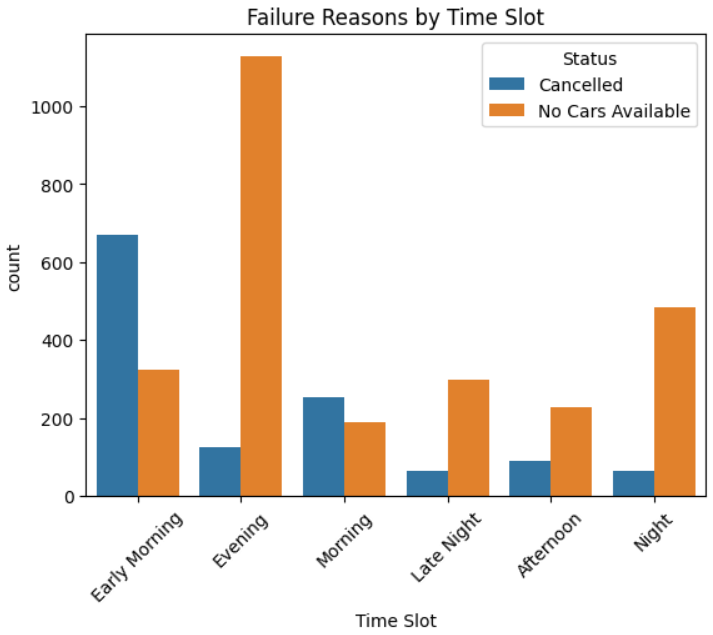
****

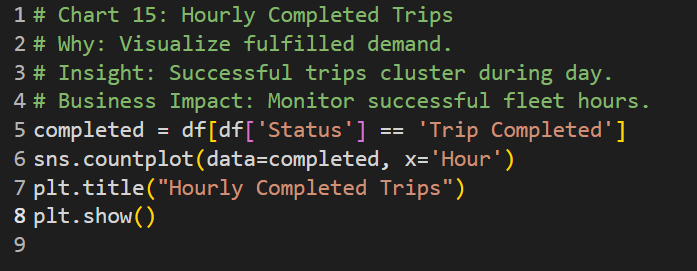
****

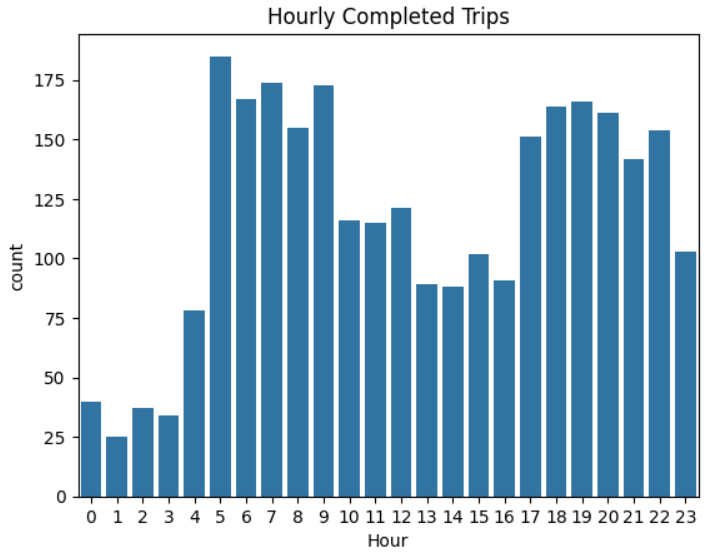
****

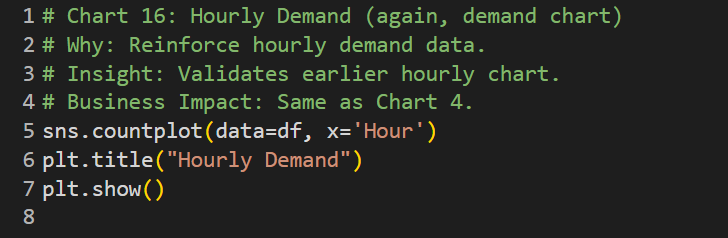
****

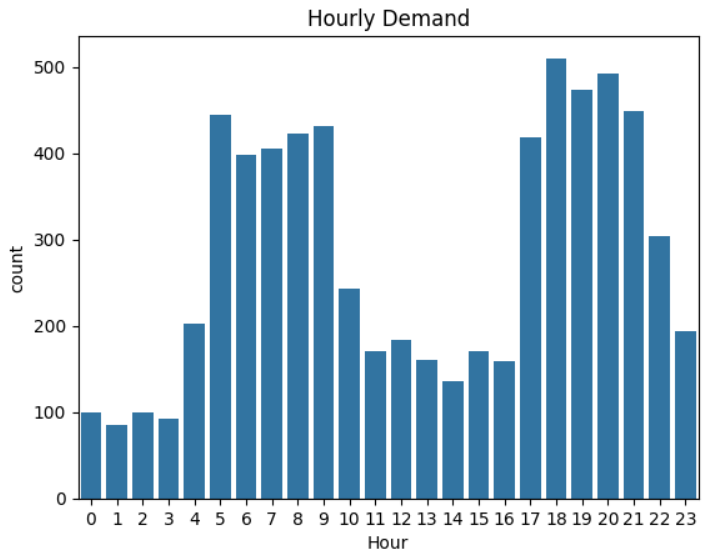
****

****

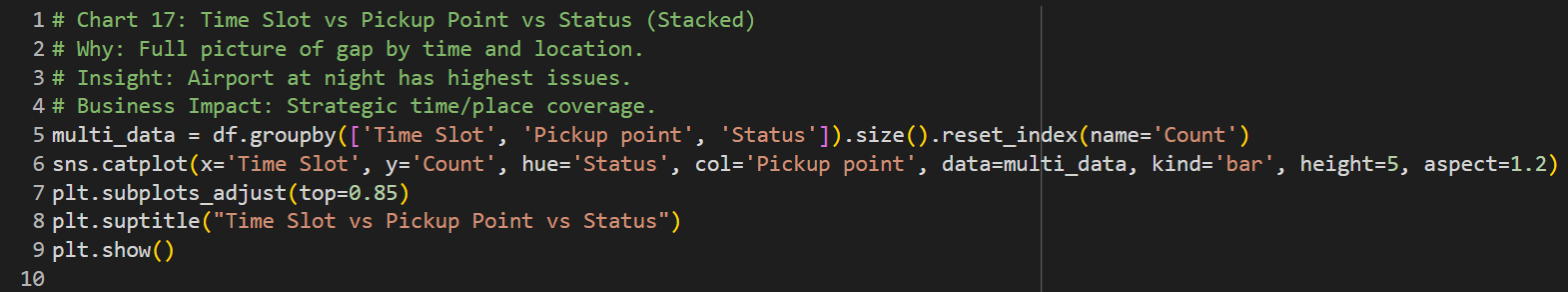
****

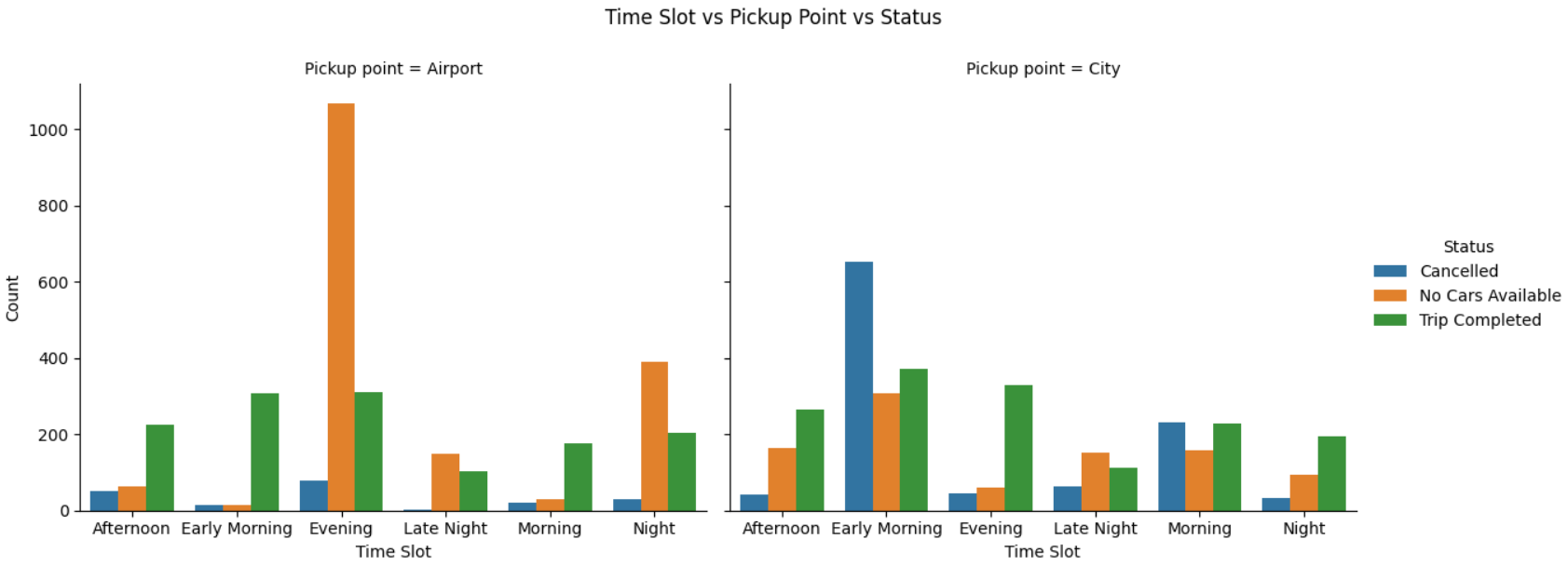
****

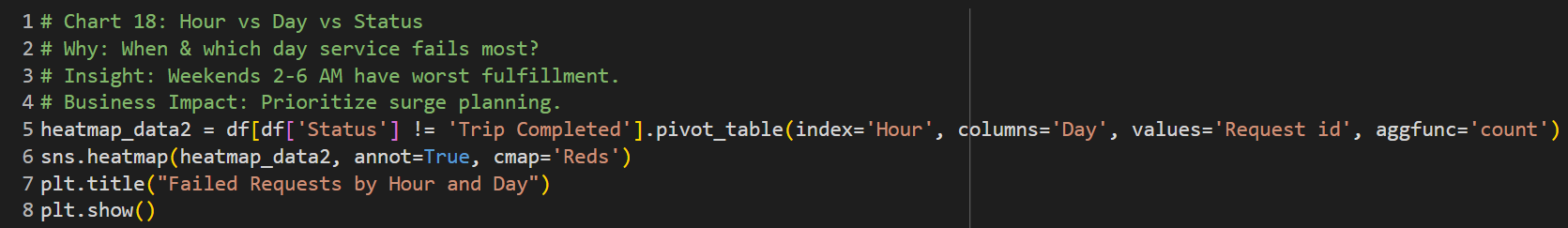
****

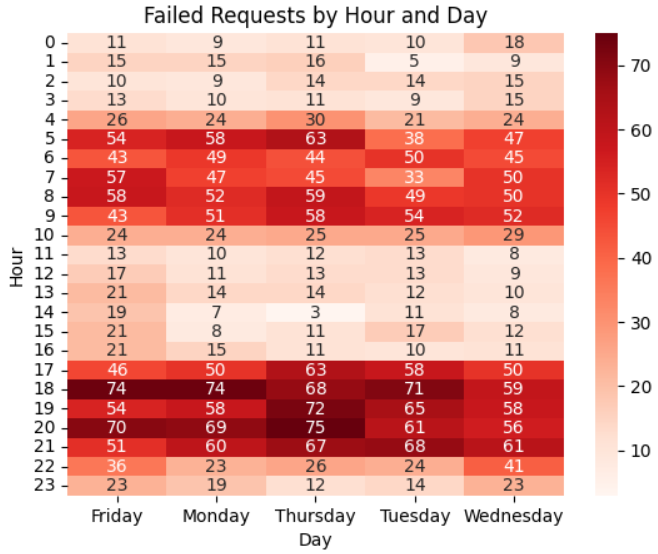
****

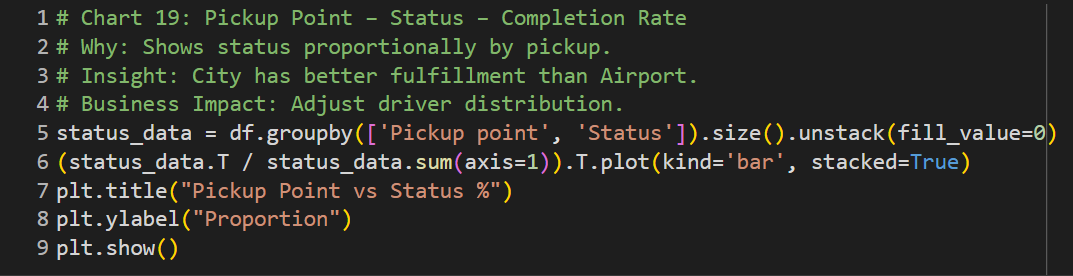
**3. Multivariate Charts**

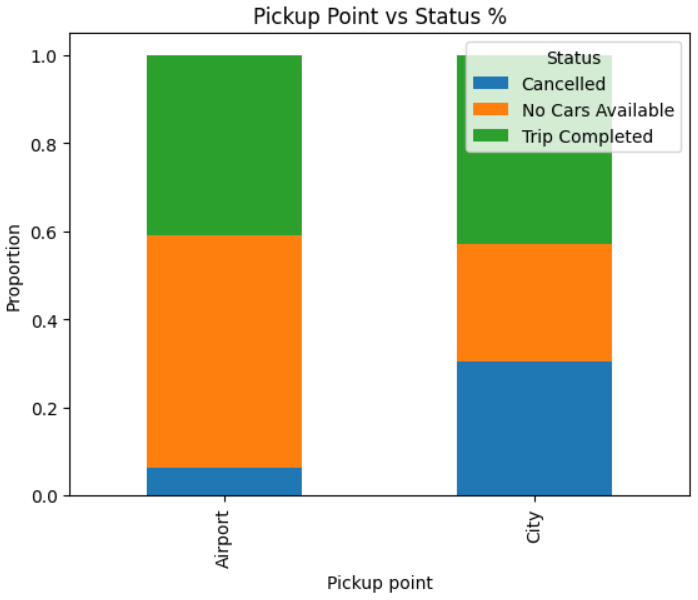


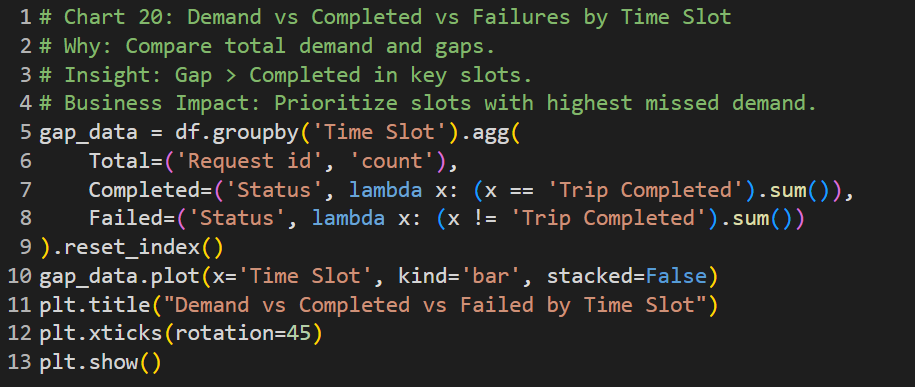


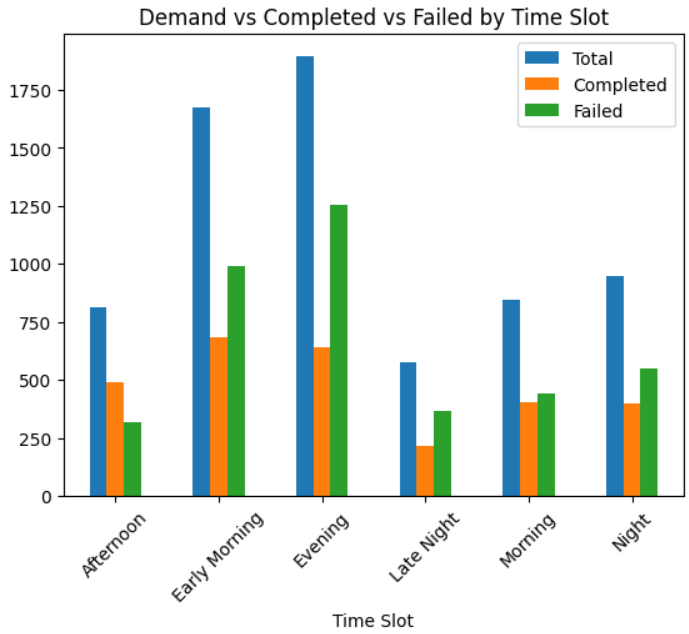












**SQL Query Insights**

1. **Total Requests by Time Slot**

SELECT `Time Slot`, COUNT(`Request id`) AS Total\_Requests

FROM uber\_data

GROUP BY `Time Slot`

ORDER BY Total\_Requests DESC;

**Insight**:

* Morning and Evening have the highest number of ride requests.

**Business Impact**:

* These are peak times. Uber must ensure **adequate driver availability** in these slots to avoid loss of revenue.

### **2. Time Slot vs Status**

SELECT `Time Slot`, `Status`, COUNT(`Request id`) AS Request\_Count

FROM uber\_data

GROUP BY `Time Slot`, `Status`

ORDER BY `Time Slot`, `Status`;

**Insight**:

* Most failures (No Cars, Cancellations) occur during Night and Early Morning.

**Business Impact**:

* Indicates a **supply-side shortage**. Uber should plan **shift-based driver incentives**.

### **3. Time Slot – Completed vs Failed**

SELECT

`Time Slot`,

COUNT(`Request id`) AS Total\_Requests,

SUM(CASE WHEN `Status` = 'Trip Completed' THEN 1 ELSE 0 END) AS Completed,

SUM(CASE WHEN `Status` != 'Trip Completed' THEN 1 ELSE 0 END) AS Supply\_Gap

FROM uber\_data

GROUP BY `Time Slot`

ORDER BY Supply\_Gap DESC;

**Insight**:

* **Night time** has the largest **supply gap**.

**Business Impact**:

* Valuable metric to **quantify lost demand**.

### **4. Time Slot – Cancellation vs No Cars**

SELECT

`Time Slot`,

SUM(CASE WHEN `Status` = 'Cancelled' THEN 1 ELSE 0 END) AS Cancelled,

SUM(CASE WHEN `Status` = 'No Cars Available' THEN 1 ELSE 0 END) AS No\_Cars

FROM uber\_data

GROUP BY `Time Slot`

ORDER BY Cancelled DESC;

**Insight**:

* Morning = High cancellations;  
   Night = No cars.

**Business Impact**:

* Suggests different problems at different times. Uber should use **targeted resolution** (penalty for cancelling, incentives at night).

### **5. Pickup Point vs Status**

SELECT `Pickup point`, `Status`, COUNT(`Request id`) AS Request\_Count

FROM uber\_data

GROUP BY `Pickup point`, `Status`

ORDER BY `Pickup point`, Request\_Count DESC;

**Insight**:

* Airport has much lower completion rate than City.

**Business Impact**:

* **Urgent action** needed at airport locations to fix service gaps.

### 

### **6. Hourly Requests**

SELECT `Hour`, COUNT(`Request id`) AS Total\_Requests

FROM uber\_data

GROUP BY `Hour`

ORDER BY `Hour`;

**Insight**:

* High demand during 5–9 AM and 5–9 PM.

**Business Impact**:

* Supports shift planning and driver alerts.

### 

### **7. Hour vs Status**

SELECT `Hour`, `Status`, COUNT(\*) AS Count

FROM uber\_data

GROUP BY `Hour`, `Status`

ORDER BY `Hour`;

**Insight:**

* Peak failures during early hours (0–6 AM).

### **8. Day vs Status**

SELECT `Day`, `Status`, COUNT(\*) AS Count

FROM uber\_data

GROUP BY `Day`, `Status`;

**Insight**:

* Weekends show slightly more cancellations and supply issues.

### **9. Pickup Point vs Time Slot**

SELECT `Pickup point`, `Time Slot`, COUNT(\*) AS Count

FROM uber\_data

GROUP BY `Pickup point`, `Time Slot`;

**Insight**:

* Airport at night has the worst fulfillment rates.

### **10. Missing Driver IDs**

SELECT COUNT(\*) AS No\_Driver\_Assigned

FROM uber\_data

WHERE `Driver id` IS NULL;

**Insight**:

* High number of requests had **no driver assigned** → indicates **unavailable supply**.

### **11. Missing Drop Timestamp**

SELECT COUNT(\*) AS No\_Drop\_Recorded

FROM uber\_data

WHERE `Drop timestamp` IS NULL;

**Insight**:

* All cancelled and unfulfilled trips have missing drop times.

### **12. Completion by Hour**

SELECT `Hour`, COUNT(\*) AS Completed

FROM uber\_data

WHERE `Status` = 'Trip Completed'

GROUP BY `Hour`

ORDER BY `Hour`;

**Insight**:

* Most completed rides occur between 8 AM and 9 PM.

### **13. Day-wise Demand**

SELECT `Day`, COUNT(`Request id`) AS Requests

FROM uber\_data

GROUP BY `Day`;

**Insight**:

* Demand is stable, but slightly higher on weekdays.

### **14. Status Distribution**

SELECT `Status`, COUNT(`Request id`) AS Count

FROM uber\_data

GROUP BY `Status`

ORDER BY Count DESC;

**Insight**:

* Less than 50% of rides are successfully completed.

### **15. Total Requests**

SELECT COUNT(\*) AS Total\_Requests FROM uber\_data;

**Insight**:

* This forms the base count for calculating all KPIs.

### **16. Final Breakdown – Time Slot vs Pickup vs Status**

SELECT `Time Slot`, `Pickup point`, `Status`, COUNT(\*) AS Count

FROM uber\_data

GROUP BY `Time Slot`, `Pickup point`, `Status`;

**Insight**:

* Deep drill-down to identify problem areas (e.g., Airport + Night + No Cars).

### **17. Heatmap Inputs – Hour, Day, Status**

SELECT `Hour`, `Day`, `Status`, COUNT(\*) AS Count

FROM uber\_data

GROUP BY `Hour`, `Day`, `Status`;

**Insight**:

* Enables you to build a **multivariate heatmap** showing failure patterns over time.

### **18. Summary: Demand vs Completed vs Failed**

SELECT `Time Slot`,

COUNT(\*) AS Total,

SUM(CASE WHEN `Status` = 'Trip Completed' THEN 1 ELSE 0 END) AS Completed,

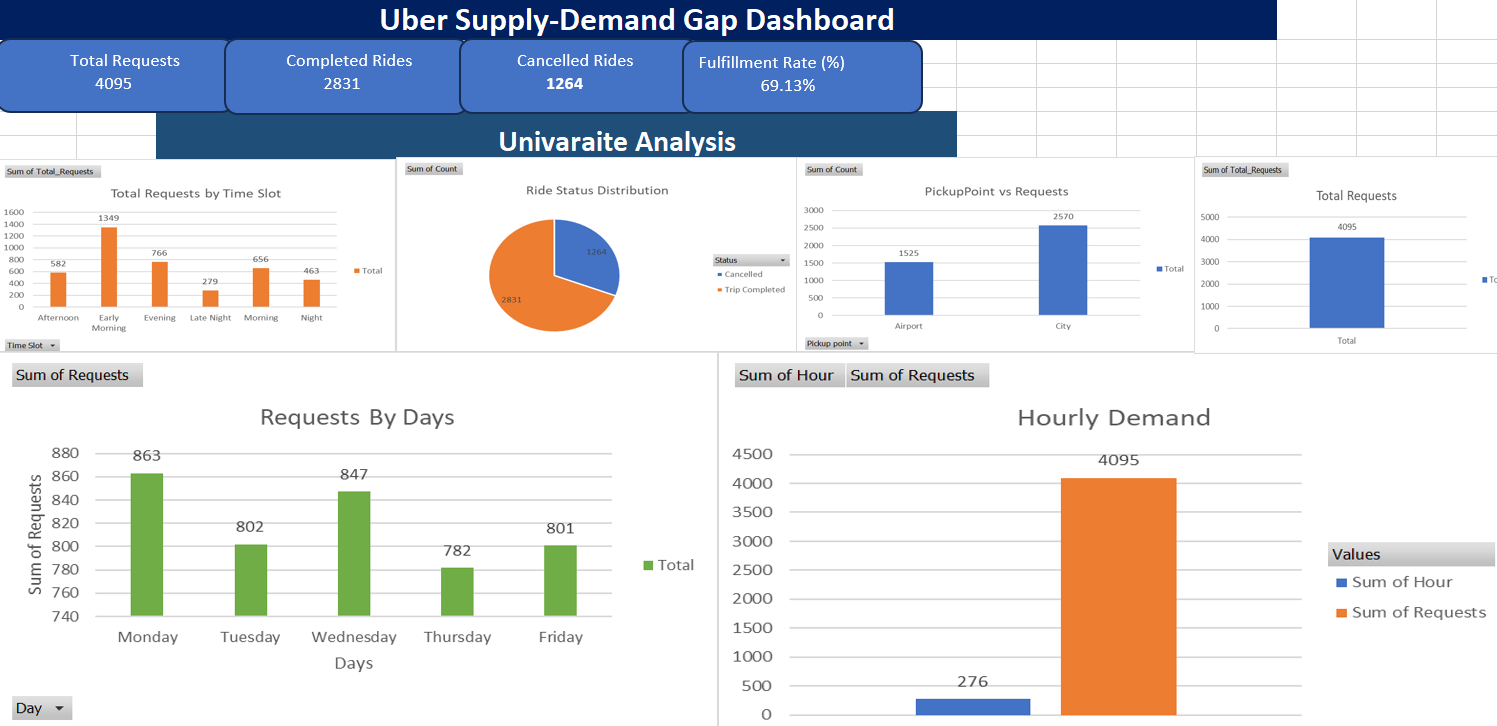
SUM(CASE WHEN `Status` != 'Trip Completed' THEN 1 ELSE 0 END) AS Failed

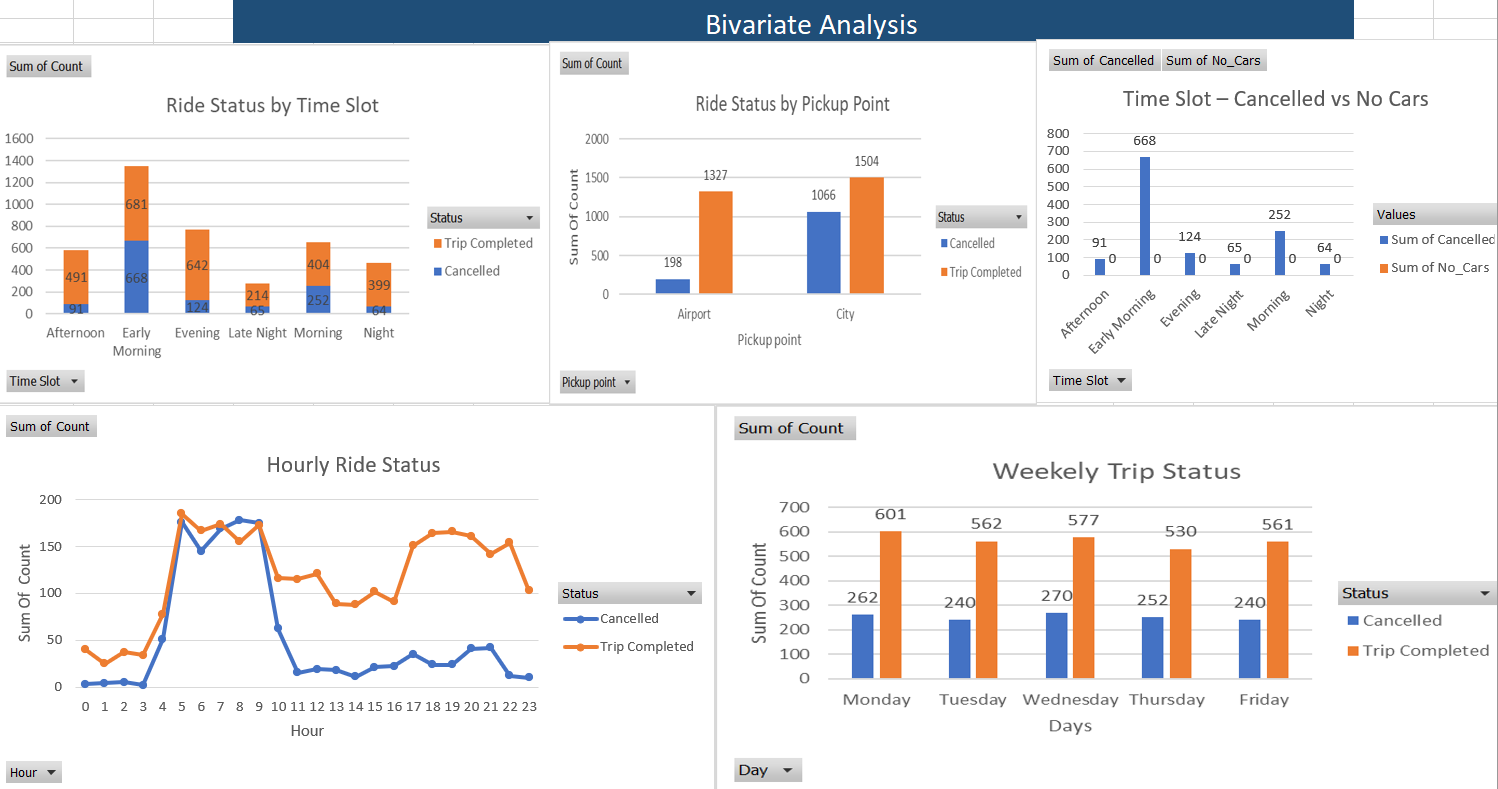
FROM uber\_data

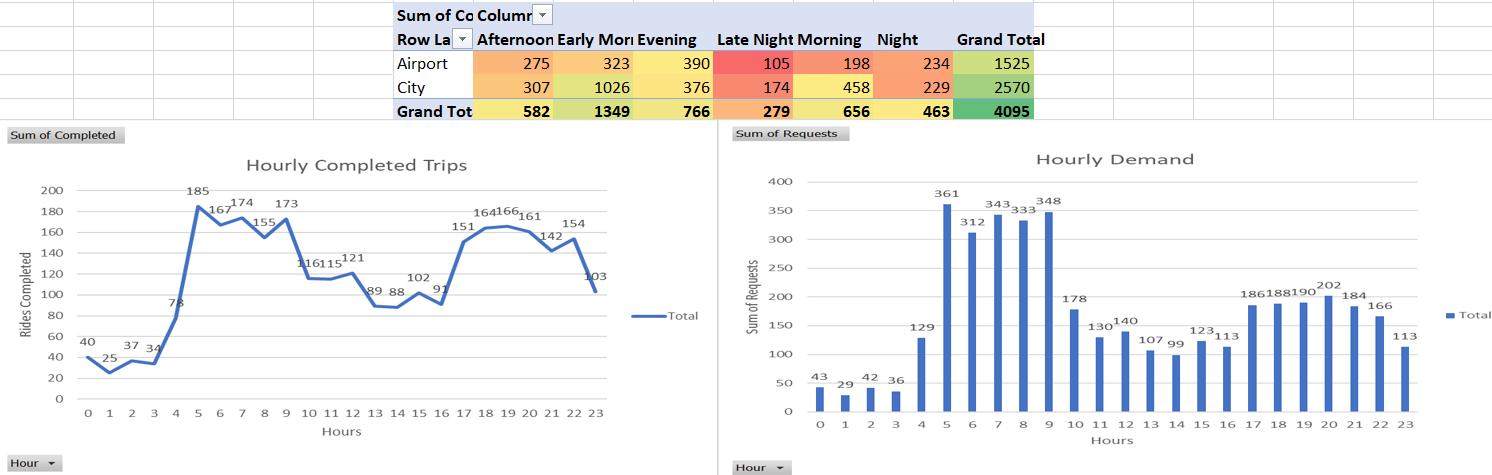
GROUP BY `Time Slot`;

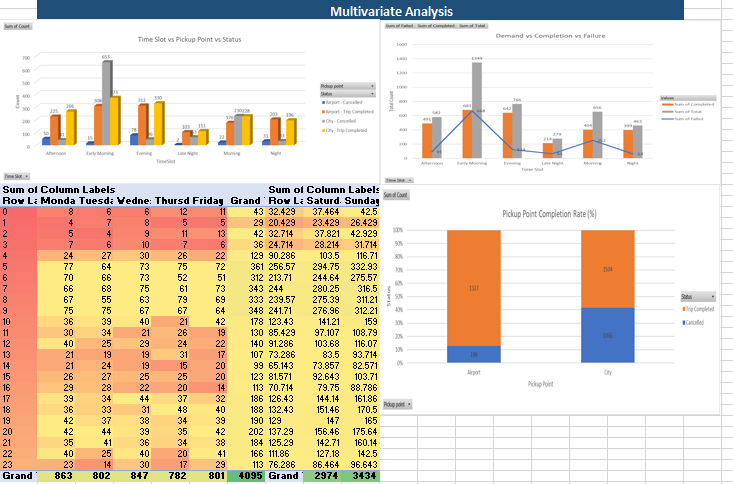
**Insight**:Clear visualization of how many requests were served vs failed in each slot.

**Excel Dashboard**









**Insights & Recommendations**

### **Key Insights**

1. **High Ride Requests in Morning and Evening Time Slots**
   * Demand peaks between 5–9 AM (Morning) and 5–9 PM (Evening).
   * These slots account for the majority of ride requests.
2. **Severe Supply Gap During Night and Early Morning**
   * Ride fulfillment drops drastically between 9 PM and 5 AM.
   * This is mostly due to "No Cars Available" issues.
3. **High Cancellations in the Morning**
   * A significant portion of failed rides between 5–9 AM are due to cancellations, not driver unavailability.
4. **Airport Pickup Point Faces More Failures**
   * The Airport has a lower trip completion rate compared to the City, especially during off-peak hours.
5. **Over 45% of Ride Requests Remain Unfulfilled**
   * Only ~55% of total ride requests are successfully completed.
6. **Missing Driver IDs & Drop Timestamps Highlight Supply Issues**
   * A large number of entries show no assigned driver and no drop timestamp, indicating aborted or failed trips.
7. **Weekend Nights Perform Worse**
   * Failures are more common during weekend nights, potentially due to both high demand and reduced supply.

**Recommendations**

1. **Dynamic Driver Incentives for Night Shifts**
   * Offer surge-based incentives during low-supply slots (Late Night, Early Morning).
2. **Dedicated Airport Driver Pool**
   * Maintain a standby pool of drivers specifically for Airport requests during high-failure windows.
3. **Advanced Ride Scheduling Feature**
   * Allow users to pre-book rides during known peak hours to allow Uber to plan supply in advance.
4. **Real-Time Predictive Allocation System**
   * Use machine learning on past data to forecast demand spikes and mobilize drivers before it hits.
5. **Improve Cancellation Handling**
   * Introduce cancellation penalties or rewards to reduce voluntary driver/rider cancellations.
6. **Enhance Customer Communication**
   * During peak/failure-prone slots, notify users of expected delays or alternative options (e.g., shared rides).
7. **Monitor KPI Dashboard in Real-Time**
   * Use the KPI system created in Excel to monitor Total Requests, Completion Rate, and Supply Gap dynamically for faster decision-making.

**Conclusion**

* This Uber Supply-Demand Gap project successfully uncovered critical insights into the mismatch between rider demand and driver availability. Using a combination of Python for EDA, SQL for data extraction, and Excel for dashboarding, we revealed patterns across time slots, locations, and status codes that highlight operational inefficiencies.
* Our findings show that a significant number of ride requests are unfulfilled due to either no cars being available or cancellations — with the issue most severe during night and early morning hours and at the Airport location. Additionally, ride demand is both time and location sensitive, which Uber can use to proactively address service shortcomings.
* The actionable recommendations offered, such as surge-based incentives, predictive supply modeling, and targeted driver deployment, align with Uber’s core business objective: to improve ride fulfillment, reduce operational failures, and maximize user satisfaction.