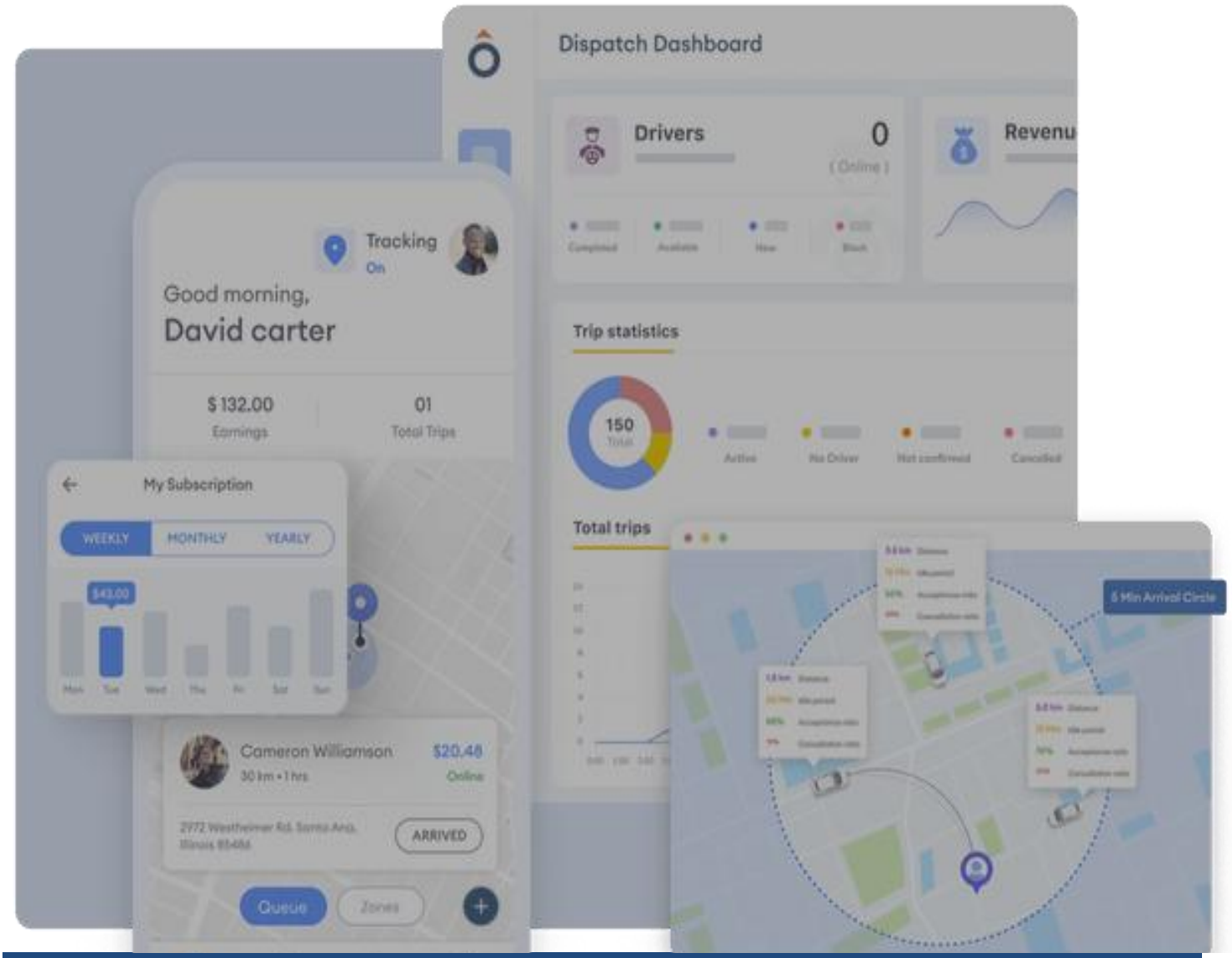


Uber Supply-Demand Gap Analysis

Tools: using Excel, SQL, and Python

6/22/25

Prajaktha Poojary

1. PROJECT OBJECTIVE

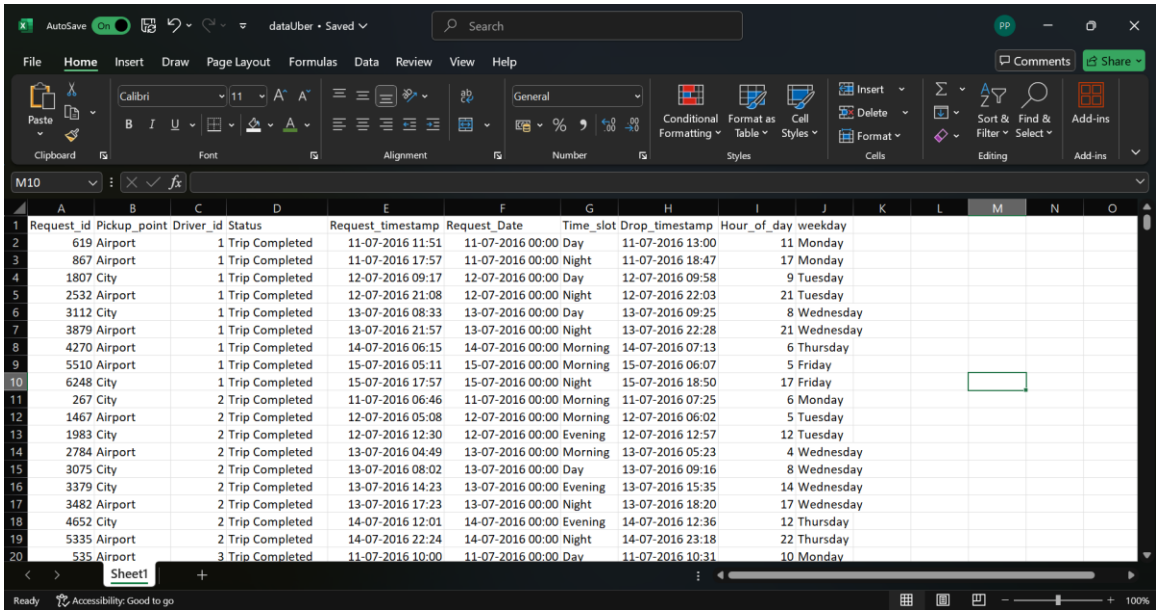
The main goal of this project is to analyze Uber ride request data to identify gaps between supply and demand using Excel, SQL, and Python. The goal is to derive actionable insights to help Uber improve driver allocation and reduce ride failures.

2. EXCEL SUMMARY

File name: uber_excel_cleaned.xlsx

🧠 Data Cleaning Done in Excel:

- Removed or fixed missing values (like Drop Timestamps for cancelled rides)
- Converted Request_timestamp and Drop_timestamp to proper datetime format
- Extracted new columns using formulas:
    - Hour_of_day using =HOUR()
    - Weekday using =TEXT(date,"dddd")
    - Time_slot using IF() and nested conditions



Request_id	Pickup_point	Driver_id	Status	Request_timestamp	Request_Date	Time_slot	Drop_timestamp	Hour_of_day	weekday
619	Airport	1	Trip Completed	11-07-2016 11:51	11-07-2016 00:00	Day	11-07-2016 13:00	11	Monday
867	Airport	1	Trip Completed	11-07-2016 17:57	11-07-2016 00:00	Night	11-07-2016 18:47	17	Monday
1807	City	1	Trip Completed	12-07-2016 09:17	12-07-2016 00:00	Day	12-07-2016 09:58	9	Tuesday
2532	Airport	1	Trip Completed	12-07-2016 21:08	12-07-2016 00:00	Night	12-07-2016 22:03	21	Tuesday
3112	City	1	Trip Completed	13-07-2016 08:33	13-07-2016 00:00	Day	13-07-2016 09:25	8	Wednesday
3879	Airport	1	Trip Completed	13-07-2016 21:57	13-07-2016 00:00	Night	13-07-2016 22:28	21	Wednesday
4270	Airport	1	Trip Completed	14-07-2016 06:15	14-07-2016 00:00	Morning	14-07-2016 07:13	6	Thursday
5510	Airport	1	Trip Completed	15-07-2016 05:11	15-07-2016 00:00	Morning	15-07-2016 06:07	5	Friday
6248	City	1	Trip Completed	15-07-2016 17:57	15-07-2016 00:00	Night	15-07-2016 18:50	17	Friday
267	City	2	Trip Completed	11-07-2016 06:46	11-07-2016 00:00	Morning	11-07-2016 07:25	6	Monday
1467	Airport	2	Trip Completed	12-07-2016 05:08	12-07-2016 00:00	Morning	12-07-2016 06:02	5	Tuesday
1983	City	2	Trip Completed	12-07-2016 12:30	12-07-2016 00:00	Evening	12-07-2016 12:57	12	Tuesday
2784	Airport	2	Trip Completed	13-07-2016 04:49	13-07-2016 00:00	Morning	13-07-2016 05:23	4	Wednesday
3075	City	2	Trip Completed	13-07-2016 08:02	13-07-2016 00:00	Day	13-07-2016 09:16	8	Wednesday
3379	City	2	Trip Completed	13-07-2016 14:23	13-07-2016 00:00	Evening	13-07-2016 15:35	14	Wednesday
3482	Airport	2	Trip Completed	13-07-2016 17:23	13-07-2016 00:00	Night	13-07-2016 18:20	17	Wednesday
4652	City	2	Trip Completed	14-07-2016 12:01	14-07-2016 00:00	Evening	14-07-2016 12:36	12	Thursday
5335	Airport	2	Trip Completed	14-07-2016 22:24	14-07-2016 00:00	Night	14-07-2016 23:18	22	Thursday
535	Airport	3	Trip Completed	11-07-2016 10:00	11-07-2016 00:00	Day	11-07-2016 10:31	10	Monday

---

📊 Pivot Tables Created:

- **Status by Pickup Point** – to see how many rides were completed, cancelled, or had no cars from City and Airport
- **Hourly Requests by Status** – to identify peak hours for each status
- **Weekday-wise Request Volume** – to check daily demand trends
- **Time Slot-wise Requests** – to analyze demand distribution across morning, evening, night, etc.

📈 **Dashboard Features:**

- Added all charts in a single sheet called Dashboard
- Used slicers for **Pickup Point** and **Weekday** to make the dashboard interactive
- Aligned charts like:
    - Bar Chart (Ride Status Distribution)
    - Pie Chart (Pickup Point Split)
    - Column Chart (Hourly Demand)
    - Line Chart (Weekday Trends)

💡 **Insights from Excel Dashboard:**

- Most ride failures happened at the **Airport during night**
- **Morning hours (7 AM – 10 AM)** saw the highest number of cancellations
- **Weekdays have more ride requests** than weekends

---

3. SQL SUMMARY

- File name: uber_queries.sql

Table Name: uber_requests

Key Queries:

📥 **What I Did:**

- Imported the cleaned CSV (cleanedReport.csv) into MySQL
- Created a table named uber_requests with appropriate data types

- Used SQL queries to explore patterns in ride request data

📌 **Important SQL Queries Used:**

- **Total number of requests by Status**

```
SELECT Status, COUNT(*) FROM uber_requests GROUP BY Status;
```

- **Failed requests (Cancelled/No Cars) by Hour of Day**

```
SELECT Hour_of_day, COUNT(*)
FROM uber_requests
WHERE Status IN ('Cancelled', 'No Cars Available')
GROUP BY Hour_of_day;
```

- **Sql query for Pickup Point-wise No Cars Available count**

```
SELECT Pickup_point, COUNT(*)
FROM uber_requests
WHERE Status = 'No Cars Available'
GROUP BY Pickup_point;
```

- **Ride status distribution on each weekday**

```
SELECT weekday, Status, COUNT(*)
FROM uber_requests
GROUP BY weekday, Status;
```

💡 **Insights from SQL Analysis:**

- Most cancellations happen between **7 AM and 10 AM**
- **Airport** has the highest number of "No Cars Available" responses
- **Weekdays** show a higher number of failed rides compared to weekends

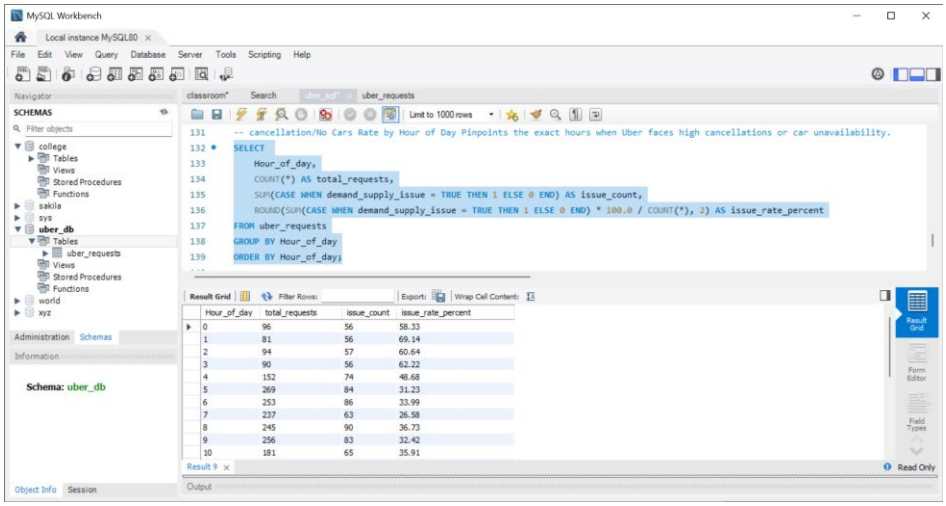


Fig: Cancellation/No cars rate by hour of day

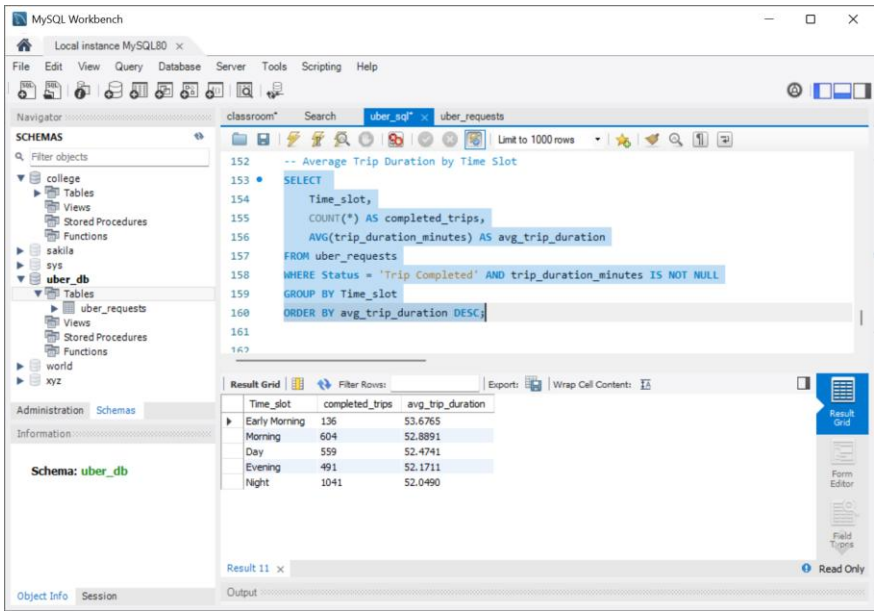


Fig:average Trip Duration by Time Slot

4. PYTHON EDA SUMMARY

File name: Uber_EDA_UBM_Visualizations.ipynb
Tools Used: Pandas, Seaborn, Matplotlib, NumPy

🖌️ Data Cleaning & Feature Engineering:

- Loaded the cleaned dataset using pandas.read_csv()

- Converted Request_timestamp and Drop_timestamp to datetime objects
- Created new columns for better analysis:
    - Hour_of_day – extracted using .dt.hour
    - Weekday – using .dt.day_name()
    - Trip_duration – calculated from request and drop time (for completed trips)
    - is_weekend – flagged Saturday/Sunday
    - Issue – flagged cancelled or "No Cars Available"
    - Time_slot_filled – filled missing/unavailable time slots based on hour

---

📈 **Charts Created:**

- Total of **20+ charts** categorized under:
    - **Univariate Analysis** (e.g., Status, Hour, Pickup Point)
    - **Bivariate Analysis** (e.g., Status vs Hour, Pickup vs Issue)
    - **Multivariate Analysis** (e.g., Status vs Hour vs Pickup Point, Heatmaps)

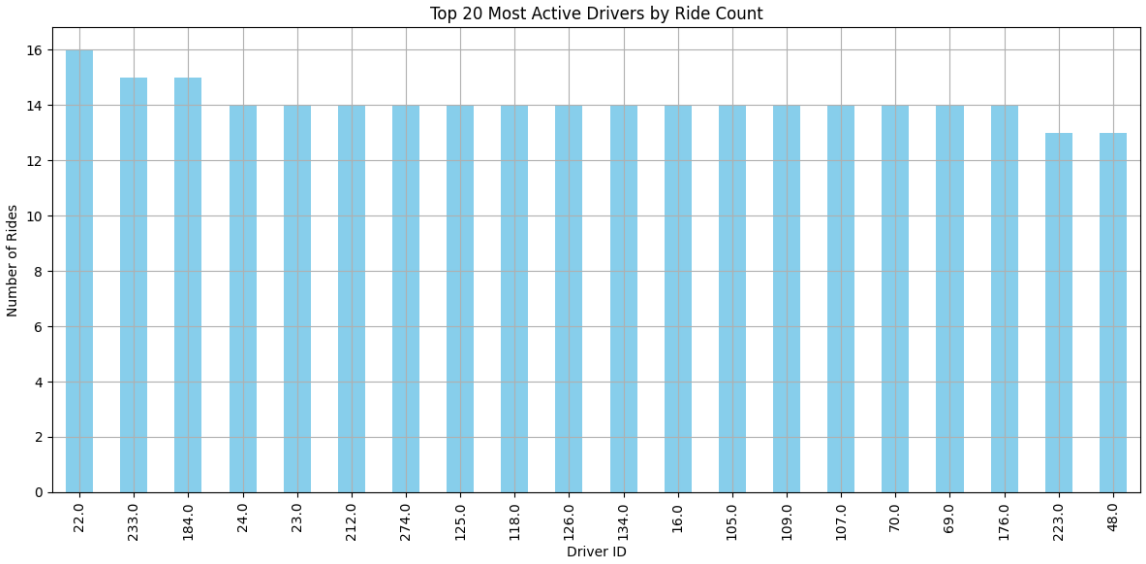


Fig: Most active drivers

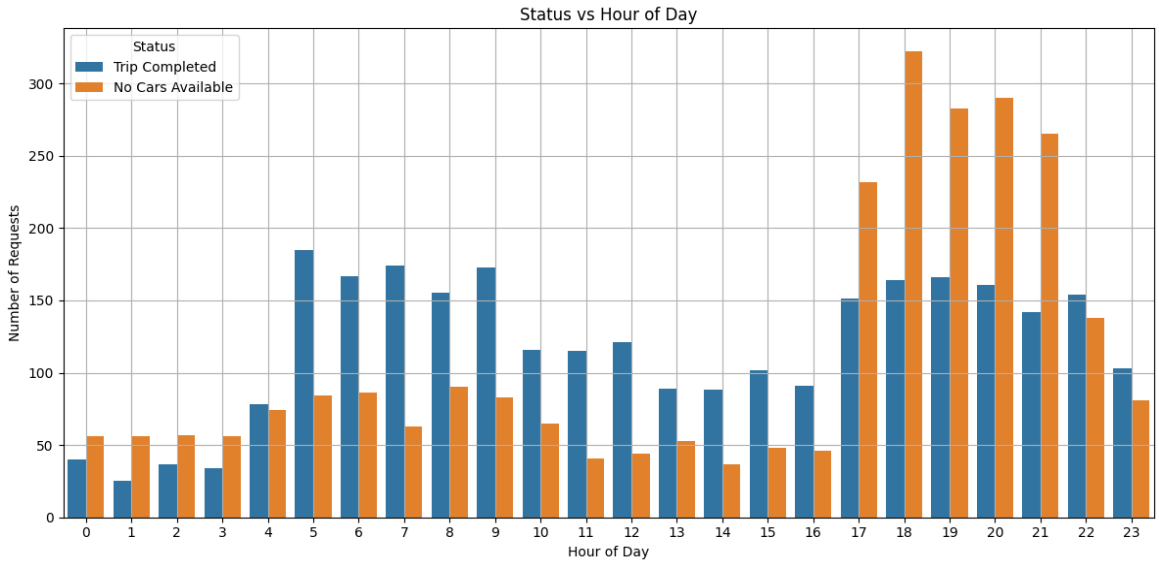


Fig: Status vs Hour of days

💡 **Insights from Python EDA:**

- Peak demand is during **Morning (7–10 AM)** and **Evening (5–8 PM)**
- **Airport** faces the highest "No Cars Available" issues, especially at Night
- **Trip durations** are longer from the Airport and during Day/Evening slots
- **Most ride failures** happen on **weekdays during morning rush hours**

📊 **Libraries Used:**

- pandas for data wrangling
- seaborn and matplotlib.pyplot for visualizations
- warnings and datetime for data handling

5. RECOMMENDATIONS

1. Increase night-shift drivers, especially at the Airport
2. Offer bonuses for early morning and weekend availability
3. Introduce predictive driver positioning based on historical peaks
4. Improve app messaging during high-failure time slots

6. CONCLUSION

This project successfully combines Excel, SQL, and Python to deliver end-to-end data analysis for Uber. The insights clearly show where Uber faces demand-supply mismatches and provide practical recommendations to improve service and user satisfaction.