

UBER REQUEST DATA ANALYSIS

Let's start with the Excel part, where I focused on initial data cleaning and dashboard creation.

◆ Data Cleaning in Excel

After loading the raw dataset into Excel, I began by:

-Converting timestamps in the Request timestamp and Drop timestamp columns to proper datetime formats using Excel's "Format Cells" and "Text to Columns" features.

-I ensured that missing values in Drop timestamp and Driver id were flagged and interpreted correctly — marking these as incomplete trips.

-I extracted new columns such as "Request Hour" and "Request Date" using Excel's built-in formulas like =HOUR() and =TEXT() for temporal analysis.

-I created a new column called Trip Completed using an IF formula to differentiate completed vs. uncompleted trips.

These transformations helped structure the raw data into a more analyzable format.

A1	B	C	D	E	F	G	H	I	J	K	L
Request_Id	Pickup_Po	Driver_Id	Status	Request_Timestamp	Drop_Timestamp	Request_Da	Request_Hour	Drop_Hour	Trip_Duration(in minutes)		
1											
2	619 Airport	1	Trip Comp	11-07-2016 11:51	11-07-2016 13:00	11-07-2016	11	13	69		
3	867 Airport	1	Trip Comp	11-07-2016 17:57	11-07-2016 18:47	11-07-2016	17	18	50		
4	1807 City	1	Trip Comp	12-07-2016 09:17	12-07-2016 09:58	12-07-2016	9	9	41		
5	2532 Airport	1	Trip Comp	12-07-2016 21:08	12-07-2016 22:03	12-07-2016	21	22	55		
6	3112 City	1	Trip Comp	13-07-2016 08:33	13-07-2016 09:25	13-07-2016	8		9 52.51667		
7	3879 Airport	1	Trip Comp	13-07-2016 21:57	13-07-2016 22:28	13-07-2016	21	22	31.51667		
8	4270 Airport	1	Trip Comp	14-07-2016 06:15	14-07-2016 07:13	14-07-2016	6		7 57.71667		
9	5510 Airport	1	Trip Comp	15-07-2016 05:11	15-07-2016 06:07	15-07-2016	5		6 56		
10	6248 City	1	Trip Comp	15-07-2016 17:57	15-07-2016 18:50	15-07-2016	17	18	53.4		
11	267 City	2	Trip Comp	11-07-2016 06:46	11-07-2016 07:25	11-07-2016	6		7 39		
12	1467 Airport	2	Trip Comp	12-07-2016 05:08	12-07-2016 06:02	12-07-2016	5		6 54		
13	1983 City	2	Trip Comp	12-07-2016 12:30	12-07-2016 12:57	12-07-2016	12	12	27		
14	2784 Airport	2	Trip Comp	13-07-2016 04:49	13-07-2016 05:23	13-07-2016	4		5 33.71667		
15	3075 City	2	Trip Comp	13-07-2016 08:02	13-07-2016 09:16	13-07-2016	8		9 73.43333		
16	3379 City	2	Trip Comp	13-07-2016 14:23	13-07-2016 15:35	13-07-2016	14		15 72.26667		
17	3482 Airport	2	Trip Comp	13-07-2016 17:23	13-07-2016 18:20	13-07-2016	17	18	57.55		
18	4652 City	2	Trip Comp	14-07-2016 12:01	14-07-2016 12:36	14-07-2016	12		12 35.73333		
19	5335 Airport	2	Trip Comp	14-07-2016 22:24	14-07-2016 23:18	14-07-2016	22		23 54.65		
20	535 Airport	3	Trip Comp	11-07-2016 10:00	11-07-2016 10:31	11-07-2016	10		10 31		
21	960 Airport	3	Trip Comp	11-07-2016 18:45	11-07-2016 19:23	11-07-2016	18		19 38		
22	1934 Airport	3	Trip Comp	12-07-2016 11:17	12-07-2016 12:23	12-07-2016	11		12 66		
23	2083 Airport	3	Trip Comp	12-07-2016 15:46	12-07-2016 16:40	12-07-2016	15		16 54		
24	2211 Airport	3	Trin Compn	12-07-2016 18:00	12-07-2016 18:28	12-07-2016	18		18 28		

Bonda Naveen Kumar

◆ Dashboard in Excel

I then built an interactive Excel dashboard using:

-Pivot Tables to summarize data like the number of requests by status, by hour, and by pickup location.

-I added Pivot Charts such as stacked bar charts, line plots, and donut charts to visualize:

- Trip status over time
- Cancellations by hour
- Pickup point distribution

Pivot Tables:

The image displays three vertically stacked Pivot Tables from Microsoft Excel, illustrating the data summarization capabilities of the software.

Pivot Table 1 (Top): Summarizes trip status. The Row Labels are "Cancelled", "No Cars Available", and "Trip Completed". The Sum of Request_Id values are 3949069, 9501514, and 9378847 respectively. The Grand Total is 22829430.

Row Labels	Sum of Request_Id
Cancelled	3949069
No Cars Available	9501514
Trip Completed	9378847
Grand Total	22829430

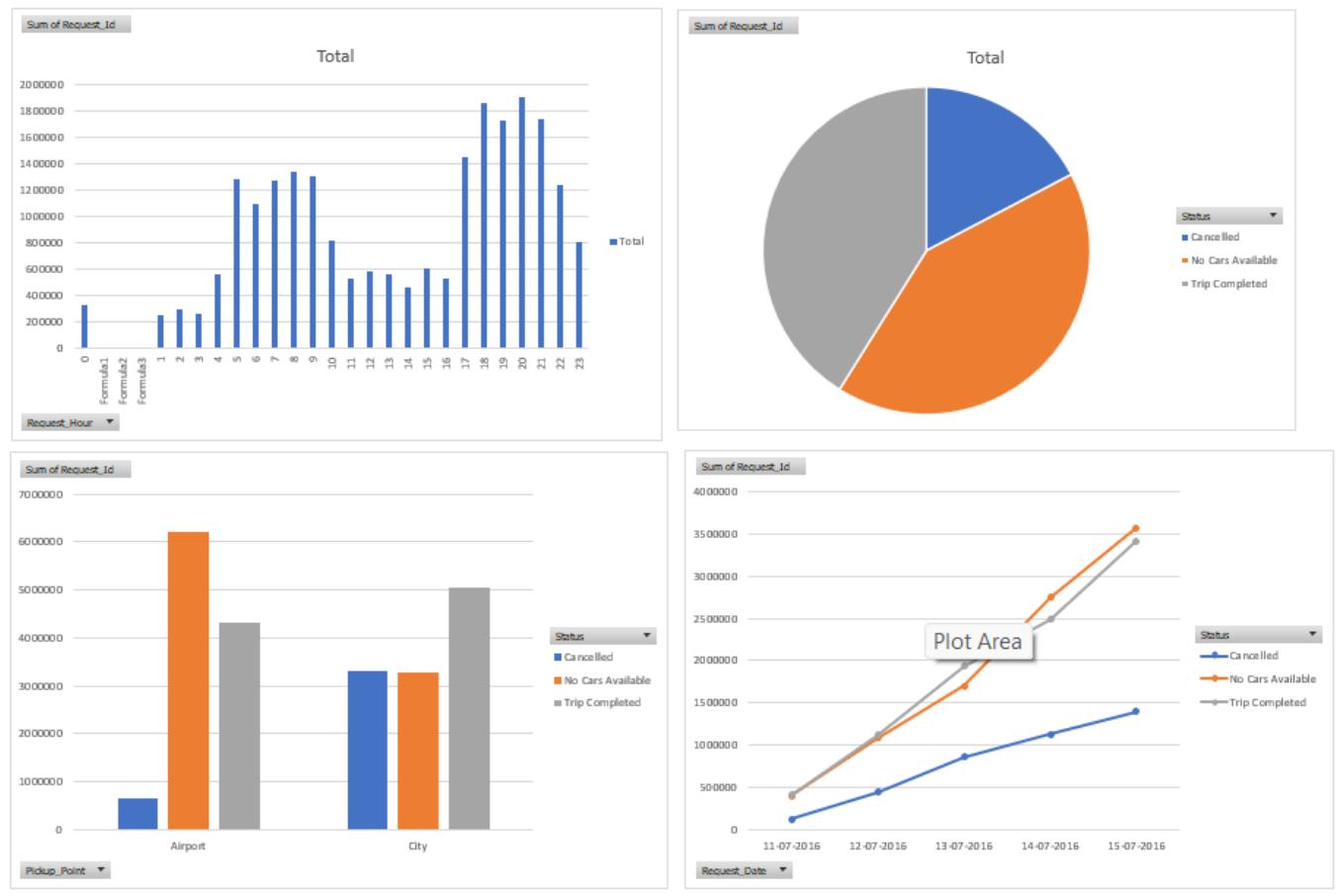
Pivot Table 2 (Middle): Summarizes pickup location. The Row Labels are "Airport" and "City". The Column Labels are "Cancelled", "No Cars Available", "Trip Completed", and "Grand Total". The values are: Airport (Cancelled: 651439, No Cars Available: 6219812, Trip Completed: 4317499, Grand Total: 11188750); City (Cancelled: 3297630, No Cars Available: 3281702, Trip Completed: 5061348, Grand Total: 11640680). The Grand Total for the entire row is 22829430.

Sum of Request_Id	Column Labels	Cancelled	No Cars Available	Trip Completed	Grand Total
Row Labels					
Airport		651439	6219812	4317499	11188750
City		3297630	3281702	5061348	11640680
Grand Total		3949069	9501514	9378847	22829430

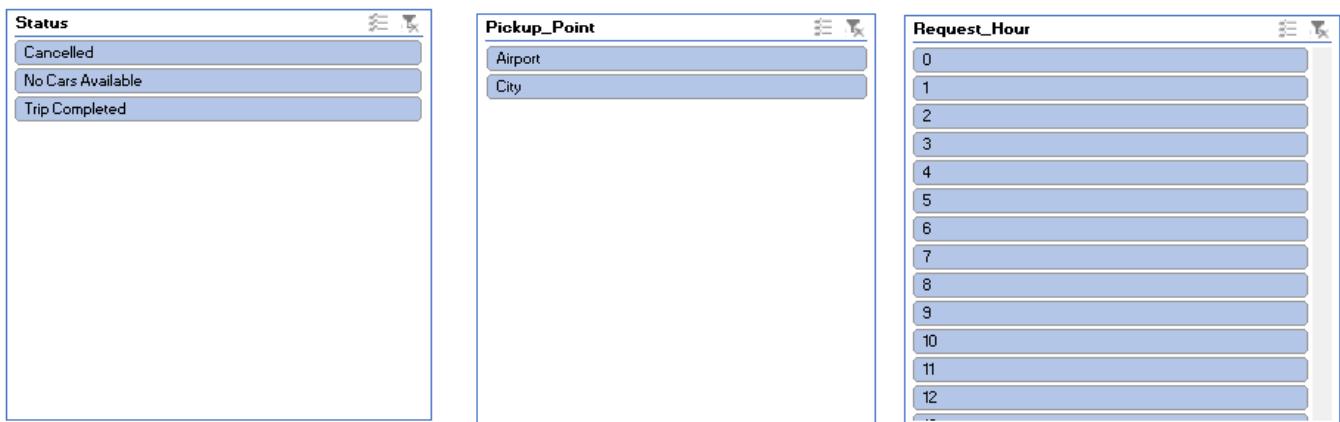
Pivot Table 3 (Bottom): Summarizes pickup date. The Row Labels are dates from 11-07-2016 to 15-07-2016. The Column Labels are "Cancelled", "No Cars Available", "Trip Completed", and "Grand Total". The values are: 11-07-2016 (Cancelled: 123431, No Cars Available: 399927, Trip Completed: 411670, Grand Total: 935028); 12-07-2016 (Cancelled: 440062, No Cars Available: 1085497, Trip Completed: 1123108, Grand Total: 2648667); 13-07-2016 (Cancelled: 857662, No Cars Available: 1698258, Trip Completed: 1931431, Grand Total: 4487351); 14-07-2016 (Cancelled: 1129011, No Cars Available: 2746118, Trip Completed: 2492299, Grand Total: 6367428); 15-07-2016 (Cancelled: 1398903, No Cars Available: 3571714, Trip Completed: 3420339, Grand Total: 8390956). The Grand Total for the entire row is 22829430.

Sum of Request_Id	Column Labels	Cancelled	No Cars Available	Trip Completed	Grand Total
Row Labels					
11-07-2016		123431	399927	411670	935028
12-07-2016		440062	1085497	1123108	2648667
13-07-2016		857662	1698258	1931431	4487351
14-07-2016		1129011	2746118	2492299	6367428
15-07-2016		1398903	3571714	3420339	8390956
Grand Total		3949069	9501514	9378847	22829430

Bonda Naveen Kumar



To make it dynamic, I used slicers and filters that allow the user to change the view by pickup point or hour of day.



This dashboard gives a quick, visual overview of when and where service issues occur, helping decision-makers act faster.

SQL Analysis in DB Browser :

Moving on to the SQL part, I imported the cleaned data into SQLite using DB Browser and wrote queries to extract business insights.

◆ Data Preparation

I loaded the dataset into two tables:

-uber_requests: Containing core trip data

-driver_info: For linking driver details

The screenshot shows the DB Browser for SQLite interface. The title bar says "DB Browser for SQLite - C:\Users\naveen\Downloads\Uber_Request.db". The menu bar includes File, Edit, View, Tools, Help. The toolbar has buttons for New Database, Open Database, Write Changes, Revert Changes, Undo, Open Project, Save Project, Attach Database, and Close Database. Below the toolbar is a menu bar with Database Structure, Browse Data, Edit Pragmas, Execute SQL, Create Table, Create Index, Modify Table, Delete Table, Print, and Refresh. The main area shows the database structure. Under "Tables (1)", there is one entry: "uber_data". Under "uber_data", there are 11 columns listed with their types: Request_Id (INTEGER), Pickup_Point (TEXT), Driver_Id (INTEGER), Status (TEXT), Request_Timestamp (TEXT), Drop_Timestamp (TEXT), Request_Date (TEXT), Request_Hour (INTEGER), field9 (TEXT), Drop_Hour (INTEGER), and Trip_Duration(in minutes) (INTEGER). To the right of the table structure is the "Schema" pane, which contains the CREATE TABLE SQL statement for "uber_data":

```
CREATE TABLE "uber_data" ( "Request_Id" INTEGER, "Pickup_Point" TEXT, "Driver_Id" INTEGER, "Status" TEXT, "Request_Timestamp" TEXT, "Drop_Timestamp" TEXT, "Request_Date" TEXT, "Request_Hour" INTEGER, "field9" TEXT, "Drop_Hour" INTEGER, "Trip_Duration(in minutes)" INTEGER )
```

Below the table structure, there are links for Indices (0), Views (0), and Triggers (0).

◆ Key SQL Queries and Insights

Here are some examples of the queries and what I found:

```
SELECT
    COUNT(CASE WHEN Status = 'Trip Completed' THEN 1 END) * 1.0 / COUNT(*) * 100 AS completion_rate_percent
FROM uber_data;
```

completion_rate_percent
41.9718309659155

Bonda Naveen Kumar

```
1   SELECT "Request_Hour", COUNT(*) AS total_requests
2   FROM uber_data
3   GROUP BY "Request_Hour"
4   ORDER BY "Request_Hour";
5
```

	Request_Hour	total_requests
18	17	418
19	18	510
20	19	473
21	20	492
22	21	449
23	22	304
24	23	194

```
1   SELECT
2   "Pickup_Point",
3   COUNT(*) AS total_requests,
4   SUM(CASE WHEN Status = 'Cancelled' THEN 1 ELSE 0 END) AS cancelled_requests,
5   ROUND(SUM(CASE WHEN Status = 'Cancelled' THEN 1.0 ELSE 0 END) * 100.0 / COUNT(*), 2) AS cancellation_rate_percent
6   FROM uber_data
7   GROUP BY "Pickup_Point";
8
```

	Pickup_Point	total_requests	cancelled_requests	cancellation_rate_percent
1	Airport	3238	198	6.11
2	city	3507	1066	30.4

```
1   SELECT
2   "Request_Hour",
3   SUM(CASE WHEN Status = 'Trip Completed' THEN 1 ELSE 0 END) AS completed,
4   SUM(CASE WHEN Status = 'Cancelled' THEN 1 ELSE 0 END) AS cancelled,
5   SUM(CASE WHEN Status = 'No Cars Available' THEN 1 ELSE 0 END) AS no_car,
6   COUNT(*) AS total_requests
7   FROM uber_data
8   GROUP BY "Request_Hour"
9   ORDER BY "Request_Hour";
10
```

	Request_Hour	completed	cancelled	no_car	total_requests
6	5	185	176	84	445
7	6	167	145	86	398
8	7	174	169	63	406
9	8	155	178	90	423
10	9	173	175	83	431
11	10	116	62	85	243
12	11	115	15	41	171

Bonda Naveen Kumar

```
1  SELECT
2      "Pickup_Point",
3      "Request_Hour",
4      COUNT(*) AS cancelled_requests
5  FROM uber_data
6  WHERE Status = 'Cancelled'
7  GROUP BY "Pickup_Point", "Request_Hour"
8  ORDER BY cancelled_requests DESC
9  LIMIT 10;
10
```

	Pickup_Point	Request_Hour	cancelled_requests
1	City	8	176
2	City	5	172
3	City	9	167
4	City	7	164
5	City	6	141
6	City	10	53
7	City	4	49

↳

```
1  SELECT
2      "Pickup_Point",
3      SUM(CASE WHEN Status = 'Trip Completed' THEN 1 ELSE 0 END) AS completed,
4      SUM(CASE WHEN Status = 'Cancelled' THEN 1 ELSE 0 END) AS cancelled,
5      SUM(CASE WHEN Status = 'No Cars Available' THEN 1 ELSE 0 END) AS no_car
6  FROM uber_data
7  GROUP BY "Pickup_Point";
8
```

	Pickup_Point	completed	cancelled	no_car
1	Airport	1327	198	1713
2	city	1504	1066	937

```
1  SELECT
2      CASE
3          WHEN "Trip_Duration(in minutes)" <= 10 THEN '0-10 min'
4          WHEN "Trip_Duration(in minutes)" <= 20 THEN '11-20 min'
5          WHEN "Trip_Duration(in minutes)" <= 30 THEN '21-30 min'
6          ELSE '30+ min'
7      END AS duration_bucket,
8      COUNT(*) AS trip_count
9  FROM uber_data
10 WHERE Status = 'Trip Completed'
11 GROUP BY duration_bucket
12 ORDER BY trip_count DESC;
13
```

	duration_bucket	trip_count
1	30+ min	2713
2	21-30 min	118

The SQL queries helped validate the patterns I would later visualize in Python.

Exploratory Data Analysis in Python :

Finally, I conducted a full Exploratory Data Analysis (EDA) in Python using Pandas, Seaborn, and Matplotlib. This is where I followed the UBM rule: Univariate, Bivariate, and Multivariate analysis.

◆ Data Wrangling

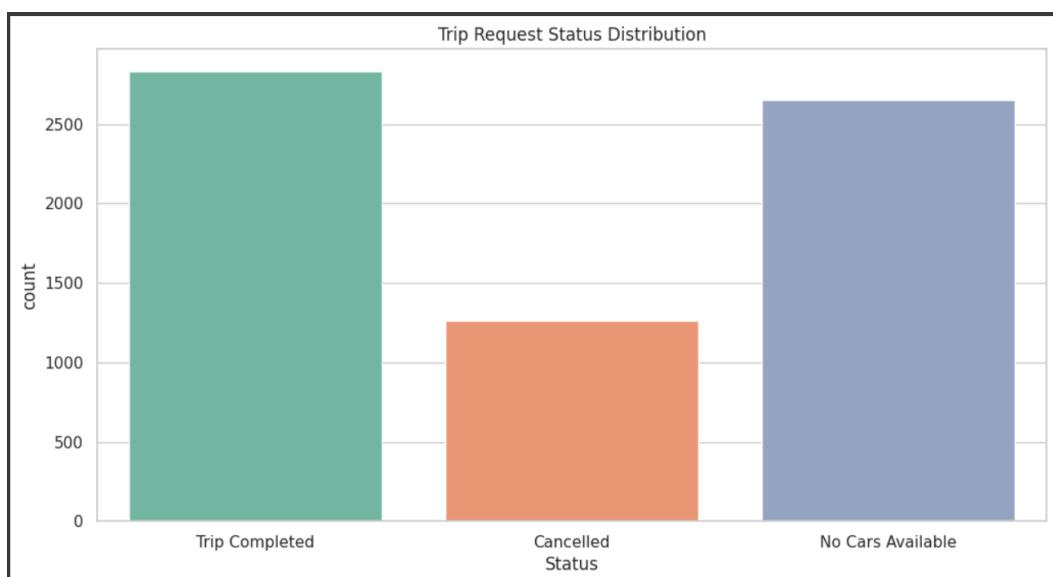
Before analysis, I:

- Converted timestamps
- Created features like Request Hour, Trip Completed, and Trip Duration
- Handled null values and standardized column formats

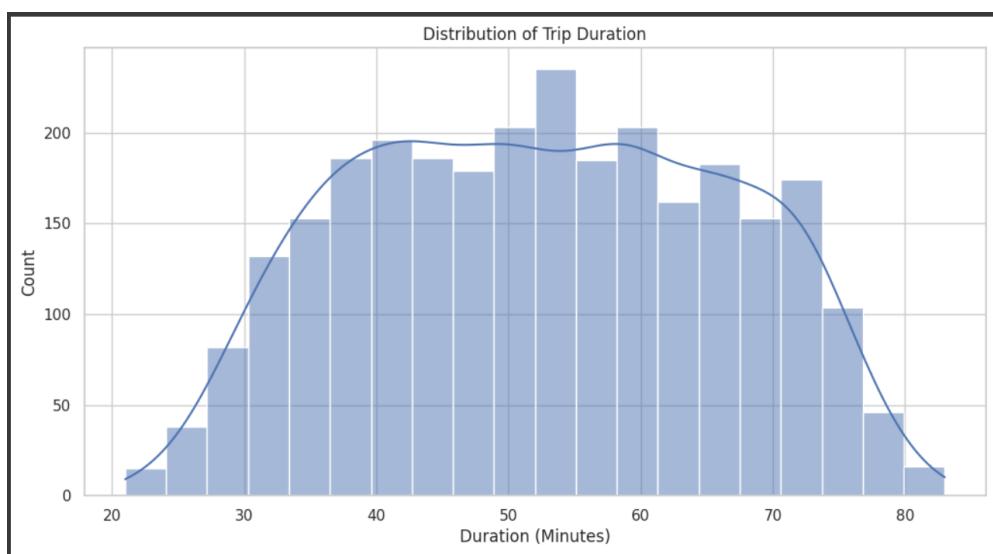
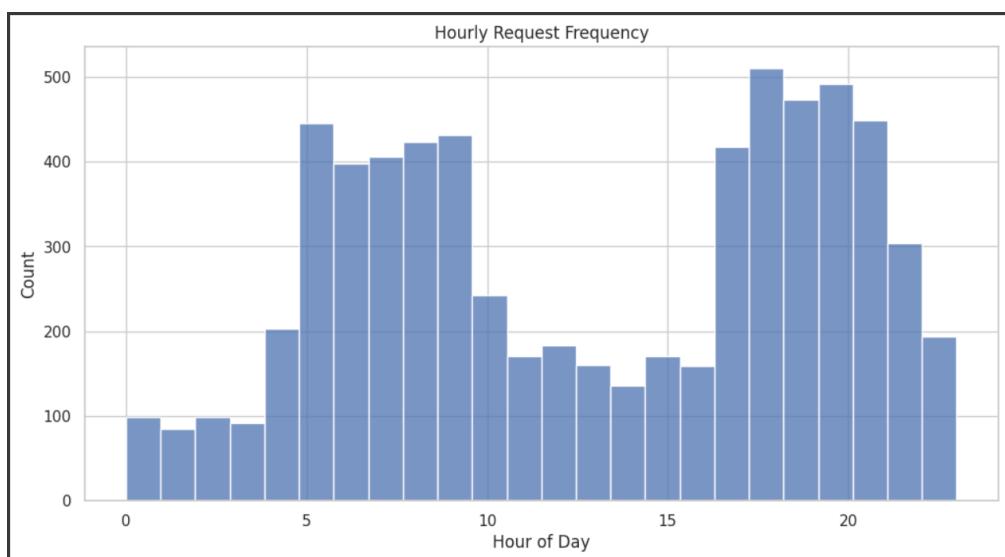
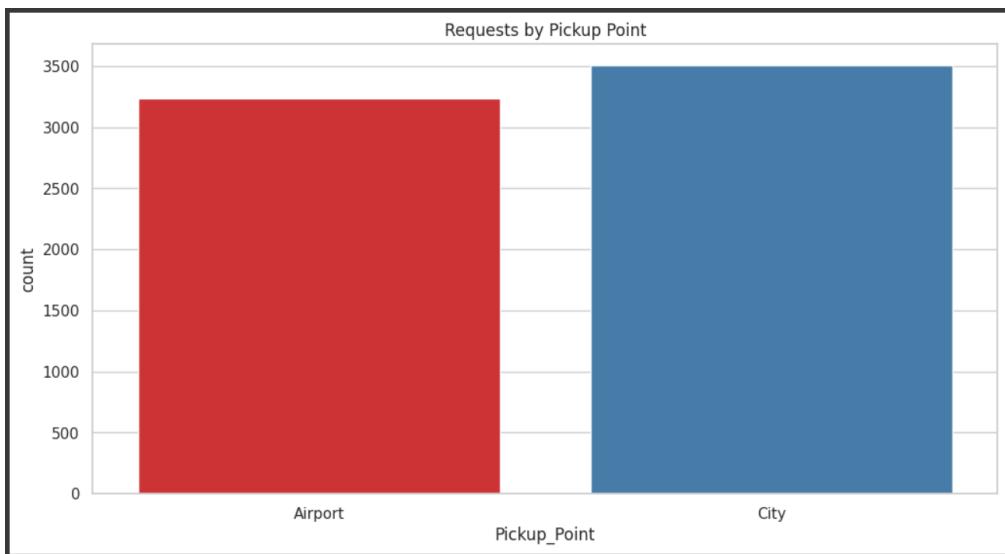
◆ Univariate Analysis

Here I analyzed individual variables.

- A bar chart of statuses showed that nearly half of all requests were either cancelled or unfulfilled, not completed.
- A histogram of request hours revealed two peak times — morning (5–10 AM) and evening (5–10 PM).
- Trip durations mostly lasted under 60 minutes.



Bonda Naveen Kumar



Bonda Naveen Kumar



◆ Bivariate Analysis

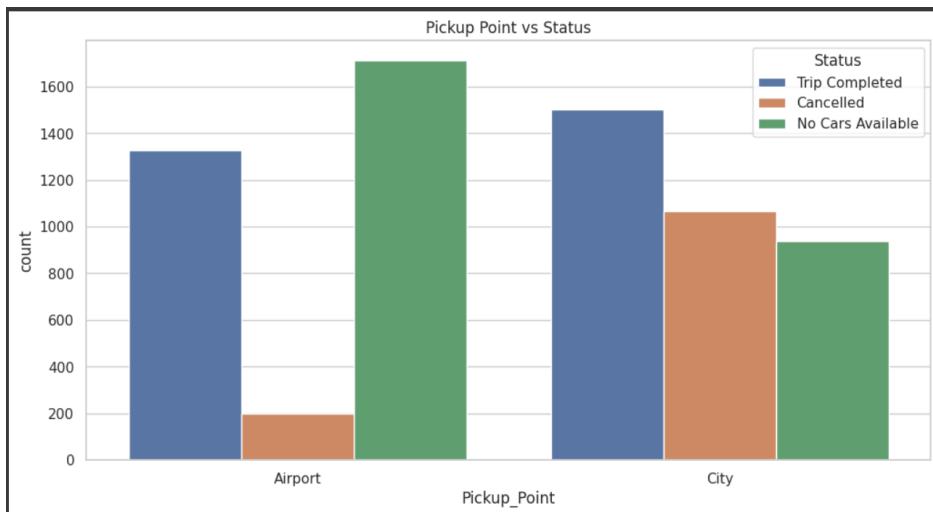
This explored relationships between two variables:

- Pickup Point vs. Status showed that:
- City requests are cancelled more
- Airport requests fail due to car unavailability

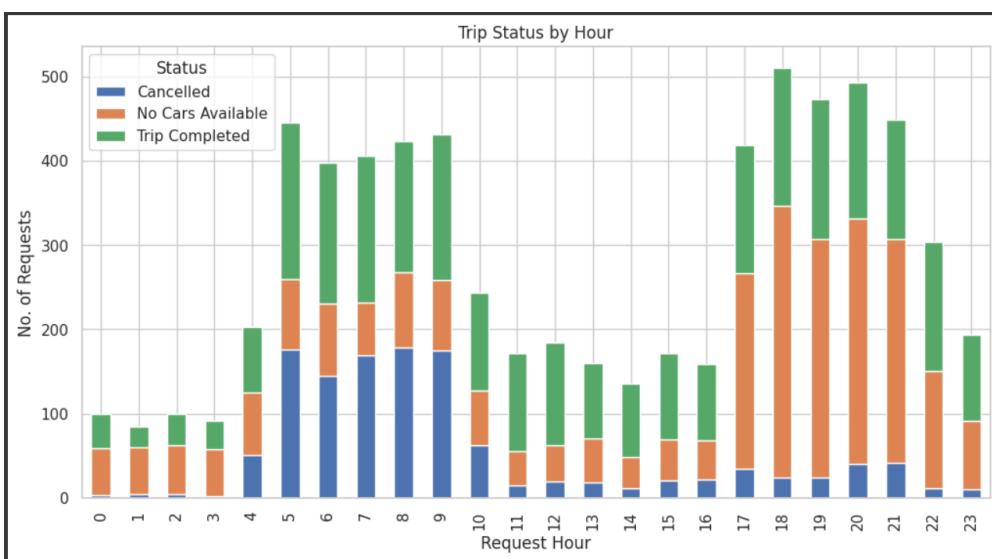
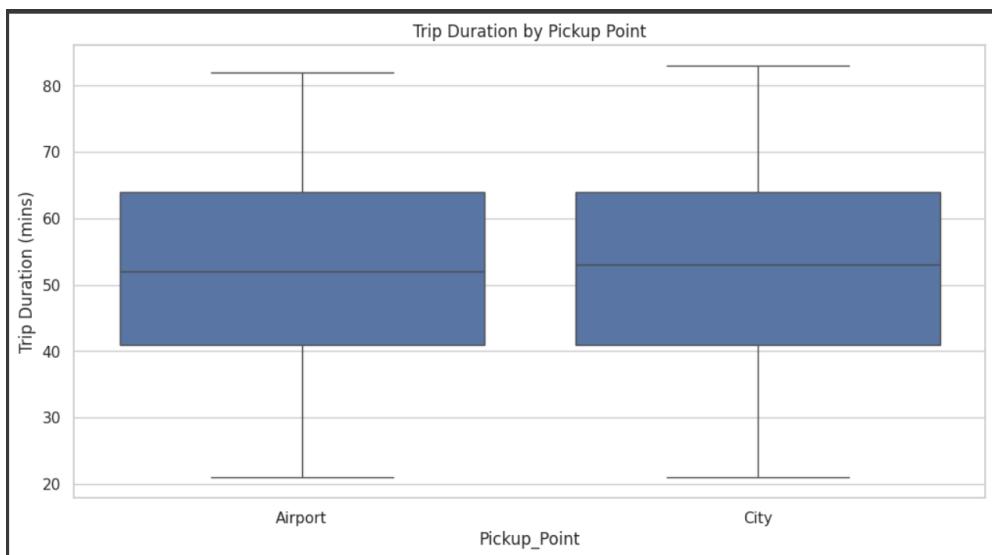
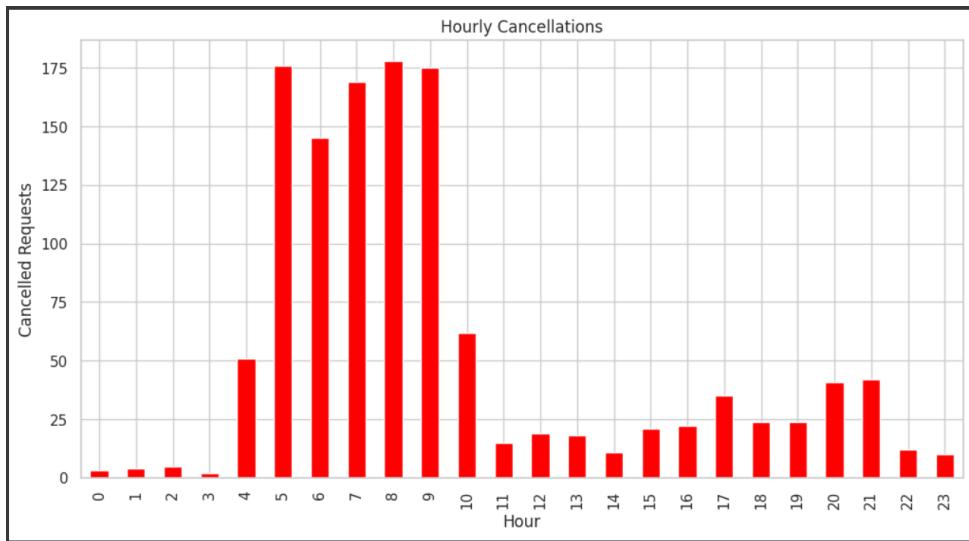
- A boxplot of Trip Duration by Pickup Point showed Airport trips tend to be longer — possibly discouraging drivers.

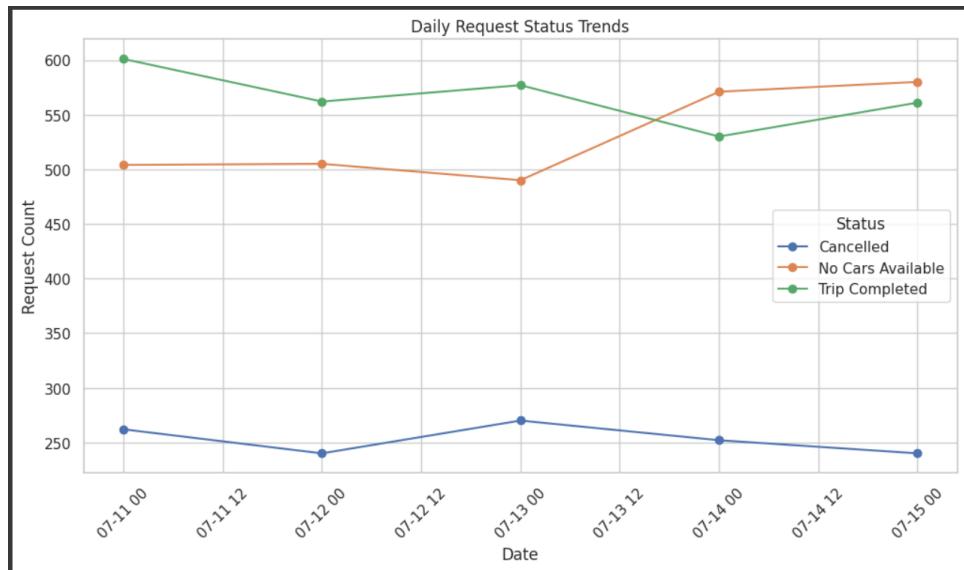
A stacked bar chart of status by hour revealed that:

- Cancellations peak in the morning
- No cars are available at night



Bonda Naveen Kumar





◆ Multivariate Analysis

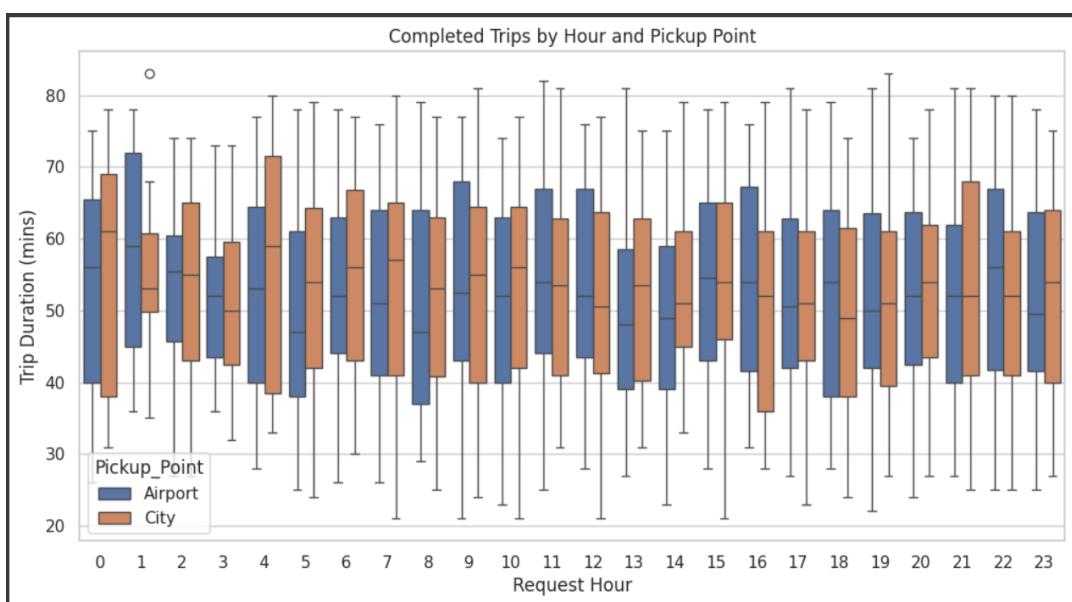
I then analyzed interactions among three or more variables:

A heatmap of cancellations showed:

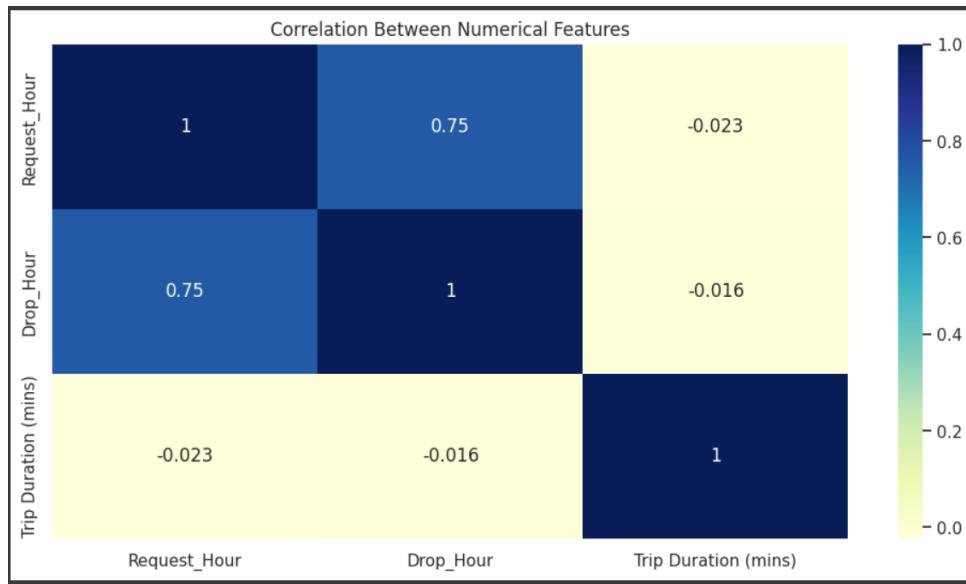
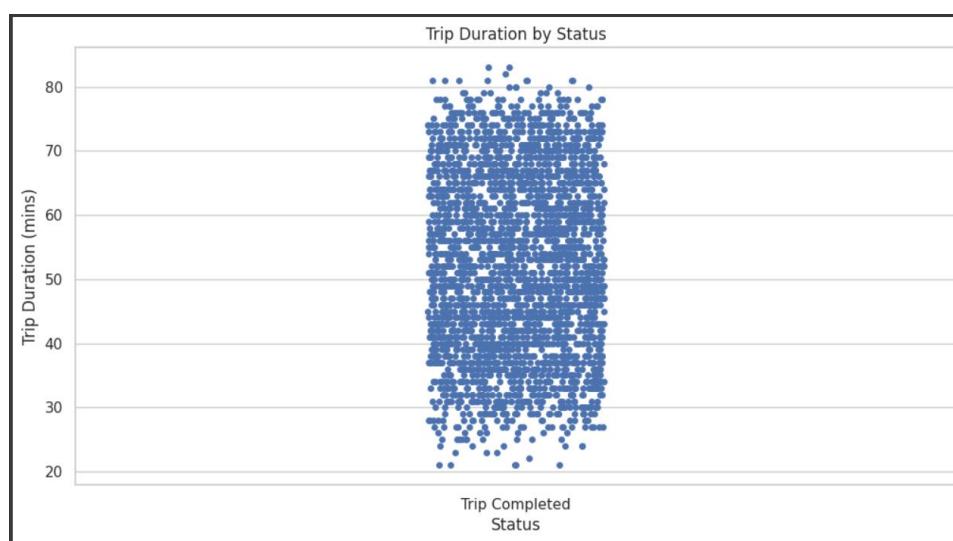
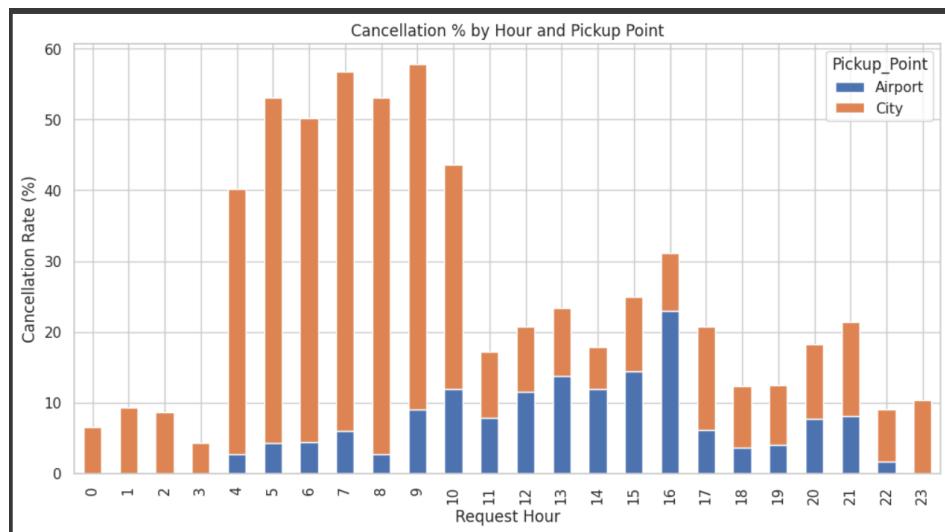
-Early morning City cancellations and night-time Airport unavailability as hot spots.

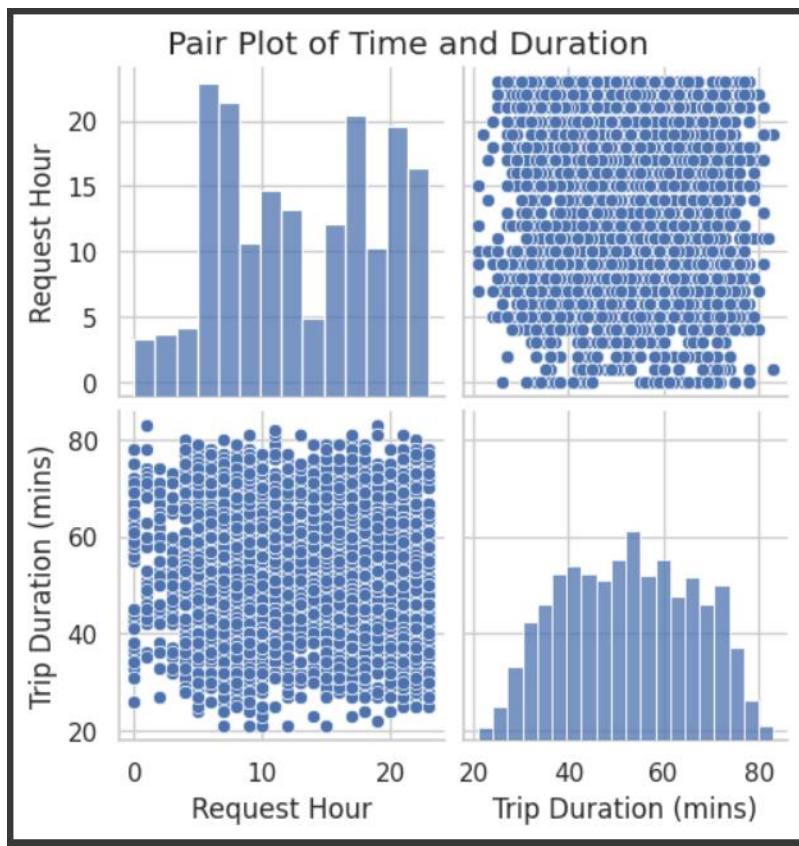
-Violin and swarm plots confirmed that longer trips are less preferred.

-A facet grid of statuses by pickup and hour provided a visual drill-down of when and where failures occur.



Bonda Naveen Kumar





Final Insights & Recommendations :

- City cancellations during morning hours suggest a need for driver incentives in those hours.
- No cars available at Airport at night calls for better night-shift driver scheduling.
- Duration-based driver reluctance could be tackled through trip-based bonus structures.

These insights can help Uber reduce failed requests and improve operational efficiency.

Solution to Business Objective :

- **Optimize Driver Deployment Based on Time and Location Trends**
- **Implement Driver Incentive Programs to Reduce Cancellations**

Bonda Naveen Kumar

- **Introduce Predictive Supply Reallocation Algorithms**
- **Enhance User Communication and Transparency**
- **Deploy Operational Heatmaps to Inform Strategy**

Conclusion :

In conclusion, I approached this project in three layers — Excel for quick overviews and dashboards, SQL for deep querying and verification, and Python EDA for in-depth pattern discovery and visualization. Each layer supported the next and helped me derive strong, actionable business recommendations for Uber.