CHICAGO TAXI TRIPS ANALYSIS

Overview

This report addresses in-depth analysis of the **Chicago Taxi Trips dataset** using **bigquery-public-data.chicago_taxi_trips.taxi_trips** dataset in Google BigQuery and Python.

Objectives

PART 1

1) a) Which three distinct taxi companies had the largest month-over-month increase in trips.
   b) Which three distinct taxi companies had the largest month-over-month decrease in fare-per-mile.

2) Provide an executive summary.

PART 2

3) a) Perform an additional analysis
   b) Include at least one visualization.

Part 1

1(a) Largest Month-Over-Month Increase in Trips

Approach

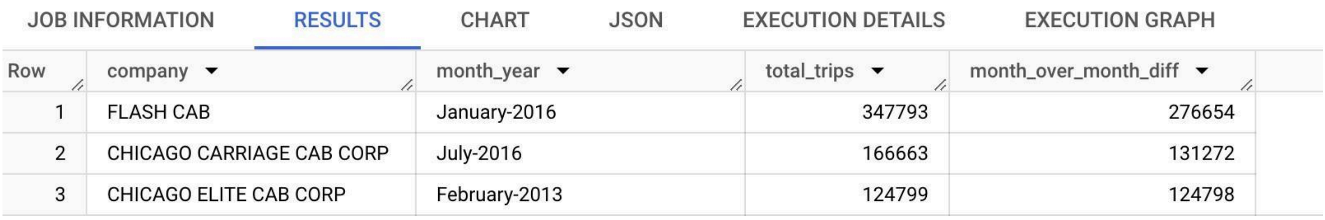
1. **Data Cleaning**: Excluded rows where the company is ‘NULL’. Normalized company names by removing trailing punctuation, converting to uppercase
2. **Data Grouping:** I first grouped trips by (company, year, month) to calculate the total number of trips in each month for each company.
3. **Window Function:** Using a LAG window function, I retrieved the previous month’s total trips for each company.
4. **Month-Over-Month Difference:** I computed current_month_trips - previous_month_trips. A positive result indicates an increase, the greater the value, the bigger the jump in trips from one month to the next.
5. **Selecting the Largest Increase per Company**: To ensure distinct companies, I used a ranking function (e.g., ROW_NUMBER()) partitioned by the company to isolate each company’s single largest jump.
6. **Final Sorting and Limit:** Finally, I sorted these largest per-company increases in descending order and limited to the top three, revealing which companies saw the biggest month-over-month surge in trips.

Query

```
 -- Data cleaning ( Removing punctuations, converting to uppercase, trim space at the
end)

WITH
 cleaned_data AS (
 SELECT
   CASE
     WHEN UPPER(company) LIKE '%CHICAGO ELITE CAB CORP%' THEN 'CHICAGO ELITE CAB CORP'
     WHEN UPPER(company) LIKE '%CHICAGO CARRIAGE CAB CORP%' THEN 'CHICAGO CARRIAGE CAB
CORP'
     ELSE TRIM(REGEXP_REPLACE(UPPER(company), r'[^\p{L}\p{N}\s]+$', ''))
 END AS cleaned_company,
   fare,
   trip_miles,
   trip_start_timestamp
 FROM `bigquery-public-data.chicago_taxi_trips.taxi_trips`
 WHERE company IS NOT NULL -- exclude null values
   ),

 --Calculating total number of trips for each company by month and year
 monthly_trips AS (
 SELECT
   cleaned_company AS company,
   EXTRACT (month FROM trip_start_timestamp) AS trip_month,
   EXTRACT (year FROM trip_start_timestamp) AS trip_year,
   COUNT(*) AS total_trips
 FROM cleaned_data
 GROUP BY
   company,
   trip_month,
   trip_year ),

 --Calculating previous month's total trip count
 previous_month_trips AS (
 SELECT
   *,
   LAG(total_trips) OVER(PARTITION BY company ORDER BY trip_year, trip_month) AS
prev_month_total_trips
 FROM monthly_trips mt)

 --Calculating month over month change in total trips and sorting it by the highest
values
SELECT
 company,
 t.month_year,
 t.total_trips,
 t.month_over_month_diff
```

```
FROM (
 SELECT
   company,
   FORMAT_DATE('%B-%Y', DATE(trip_year,trip_month,1)) AS month_year,
   total_trips,
   (total_trips - prev_month_total_trips) AS month_over_month_diff,
   ROW_NUMBER() OVER(PARTITION BY company ORDER BY (total_trips -
prev_month_total_trips)DESC) AS rn
 FROM previous_month_trips pmt
 WHERE prev_month_total_trips IS NOT NULL) t
WHERE t.rn=1
ORDER BY month_over_month_diff
DESC LIMIT 3 --to get the top 3
results
```

Results



Row	company	month_year	total_trips	month_over_month_diff
1	FLASH CAB	January-2016	347793	276654
2	CHICAGO CARRIAGE CAB CORP	July-2016	166663	131272
3	CHICAGO ELITE CAB CORP	February-2013	124799	124798

Interpretation

- ‘Flash Cab’ saw the largest month-over-month jump of ‘276,654’ trips in August 2019
- ‘Chicago Carriage Cab Corp’ saw the second largest month-over-month jump with ‘131,272’ trips in July 2016
- ‘Chicago Elite Cab Corp’ had the third largest increase from ‘1’ to ‘24,798’ trips in just a month.

1(b) Largest Month-Over-Month Decrease in Fare-Per-Mile

Approach

1. **Data Cleaning**: Excluded rows where the company is 'NULL'. Normalized company names by removing trailing punctuation, converting to uppercase
2. **Fare-per-Mile Calculation:** For each (company, year, month), I calculated average fare-per-mile—in my case, by averaging (fare/trip_miles) for each ride (an unweighted approach).
3. **Window Function:** Similar to Part I(a), I used a LAG window function partitioned by company to capture the previous month's average fare-per-mile.
4. **Difference Calculation**: By subtracting prev_month_fare_per_mile from the current month's fare-per-mile, I obtained the month-over-month change. Negative results mean the fare-per-mile decreased.
5. **Identifying the Biggest Negative Changes:** I ranked each company's monthly differences in ascending order (most negative first). This ensures I only pick the single worst drop per company.
6. **Top Three Distinct Companies:** From those single biggest drops per company, I sorted across all companies in ascending order and took the top three, indicating the largest fare-per-mile declines among distinct providers.

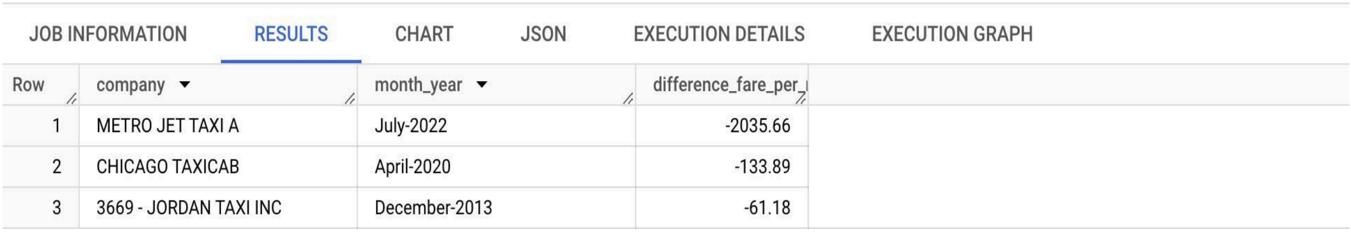
Query

```
 -- Data cleaning ( Removing puncuations, converting to upper case, trim space at the
end)

WITH
 cleaned_data AS (
 SELECT
   CASE
     WHEN UPPER(company) LIKE '%CHICAGO ELITE CAB CORP%' THEN 'CHICAGO ELITE CAB CORP'
     WHEN UPPER(company) LIKE '%CHICAGO CARRIAGE CAB CORP%' THEN 'CHICAGO CARRIAGE CAB
CORP'                                               ELSE TRIM(REGEXP_REPLACE(UPPER(company),
                                                    r'[^\p{L}\p{N}\s]+$', ''))

 END

   AS cleaned_company,
   fare,
   trip_miles,
   trip_start_timestamp
 FROM
   `bigquery-public-data.chicago_taxi_trips.taxi_trips`
 WHERE
   company IS NOT NULL -- exclude null values
   ),

 --Calculating cost per mile for each ride
 cost_per_mile AS (
 SELECT
   cleaned_company AS company,
   (fare/trip_miles) AS fare_per_mile,
   EXTRACT (month
   FROM
     trip_start_timestamp) AS trip_month,
   EXTRACT (year
   FROM
     trip_start_timestamp) AS trip_year,
   *
 FROM
   cleaned_data
 WHERE
   trip_miles>0  -- to avoid division by zero
   ),

 --Calculating avg cost per mile for each month
 monthly_cost_per_mile AS (
 SELECT
   company,
   trip_month,
   trip_year,
   SUM(fare_per_mile) AS total_fare_per_mile_monthly,
   ROUND(SUM(fare_per_mile)/COUNT(*),5) AS avg_fare_per_mile_monthly,
   COUNT(*) AS total_rides
 FROM
```

```
   cost_per_mile cpm
 GROUP BY
   company,
   trip_month,
   trip_year ),

 -- sorting the companies
 sorting_order AS (
 SELECT
   *,
   DENSE_RANK() OVER(PARTITION BY company ORDER BY trip_year, trip_month) AS dr,
   LAG(avg_fare_per_mile_monthly) OVER(PARTITION BY company ORDER BY trip_year,
trip_month) AS previous_month_avg_fare_per_mile
 FROM
   monthly_cost_per_mile mcpm ),

 --Calculating the difference
 monthly_difference AS (
 SELECT
   *,
   ROUND(avg_fare_per_mile_monthly - previous_month_avg_fare_per_mile,5) AS
difference_in_avgs
 FROM
   sorting_order so
 WHERE
   previous_month_avg_fare_per_mile IS NOT NULL ),

 --picking largest negative difference for each company
 highest_diff AS (
 SELECT
   *,
   ROW_NUMBER() OVER(PARTITION BY company ORDER BY difference_in_avgs ASC) AS
 rn FROM
   monthly_difference )

 --3 companies with largest month-over-month decrease
```

```
SELECT
 company,
 FORMAT_DATE('%B-%Y', DATE(trip_year,trip_month,1)) AS month_year,
 ROUND(difference_in_avgs,2) AS difference_fare_per_mile
FROM
 highest_diff
WHERE
 rn = 1
ORDER BY
 difference_in_avgs ASC
LIMIT
 3
```

Results



Row	company	month_year	difference_fare_per_
1	METRO JET TAXI A	July-2022	-2035.66
2	CHICAGO TAXICAB	April-2020	-133.89
3	3669 - JORDAN TAXI INC	December-2013	-61.18

Interpretation

- ‘Metro Jet Taxi A’ saw the largest month-over-month drop of ‘2035.66’ in fare per mile in July 2022.
- ‘Chicago Taxicab’ saw the second largest decrease in a month with ‘133.89’ in fare per mile in April 2020.
- ‘3669-Jordan Taxi INC’ had the third largest decrease of ‘61.18’ fare-per-mile in December 2013.

2. Executive

Summary Objective

- Identify three distinct taxi companies with the largest month-over-month increase in trips.
- Identify three distinct taxi companies with the largest month-over-month decrease in fare-per-mile.

Key Findings

1. **Largest Month-Over-Month Increase in Trips**
   - **Flash Cab**: Saw the biggest single-month jump, adding ‘276,654’ trips in August 2019.
   - **Chicago Carriage Cab Corp**: Recorded a significant monthly increase of around ‘131,272’ trips in July 2016.
   - **Chicago Elite Cab Corp**: Notably rose from a low baseline to ‘24,798’ trips in one February 2013.
2. **Largest Month-Over-Month Decrease in Fare-Per-Mile**
   - **Metro Jet Taxi A**: Experienced the steepest fare-per-mile drop (−$2035.66) in July 2022.
   - **Chicago Taxicab**: Showed the second-largest decrease (−$133.89) in April 2020.
   - **3669-Jordan Taxi INC**: Displayed the third-largest drop (−$61.18) in December 2013.

Business Insights & Considerations

- **Trip Surges**: A large jump in trips may signal successful promotions, seasonality, or improved service availability.
- **Fare-Per-Mile Drops**: Substantial decreases could reflect new pricing strategies, discounts, or competitive pressures.

Part 2

3 (a) & 3 (b) Additional Analysis & Visualization
Busiest Days and Hour of the Week in the last 2 years
Objective

After identifying month-over-month spikes in total trips and drops in fare-per-mile, I wanted to explore the weekly demand patterns by analyzing total trips by both day-of-week and hour-of-day in the last 2 years. This dual-dimensional analysis helps identify not only which days are busiest but also the specific hours that drive demand. This narrower focus can provide tangible insights for resource planning, pricing strategies, and driver availability.

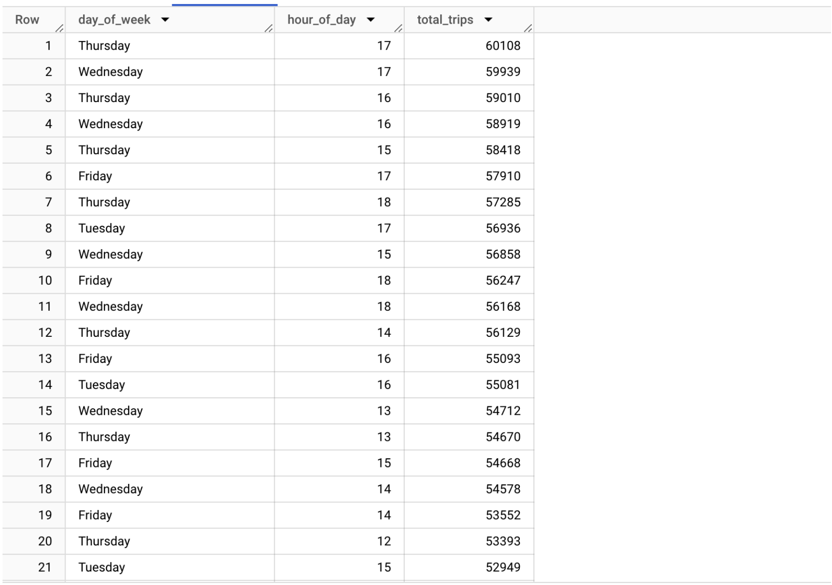
Approach

1. **Data Filtering and Cleaning:** Exclude rows with null company values and invalid data (trips with zero or negative mileage) and trips older than two years from the current date.
2. Restrict the analysis to the last two years to capture recent trends.
3. **Aggregation:** Use BigQuery's date functions to extract the full weekday name and the hour of day from the trip start timestamp.
4. Group by day-of-week and hour-of-day, and calculate the total number of trips for each combination.
5. **Visualization Preparation:** Pivot the resulting table so that days of the week form the rows and hours of the day form the columns.
6. This pivoted data is then used to create a heatmap, highlighting the busiest time slots.

Query

```
 -- Calculating trips by day of week, hour of day

WITH
 day_of_week_trips AS (
 SELECT
   FORMAT_TIMESTAMP('%A', trip_start_timestamp) AS day_of_week,
   EXTRACT(HOUR
   FROM
     trip_start_timestamp) AS hour_of_day,
   COUNT(*) AS total_trips
 FROM
   `bigquery-public-data.chicago_taxi_trips.taxi_trips`
 WHERE
   company IS NOT NULL -- exclude null values
   AND trip_miles>0
   AND trip_start_timestamp BETWEEN TIMESTAMP(DATE_SUB(DATE(CURRENT_TIMESTAMP()),
INTERVAL 2 YEAR))
   AND CURRENT_TIMESTAMP()
 GROUP BY
   day_of_week,
   hour_of_day )
SELECT
 day_of_week,
 hour_of_day,
 total_trips
FROM
 day_of_week_trips
ORDER BY
 total_trips DESC;
```

Results

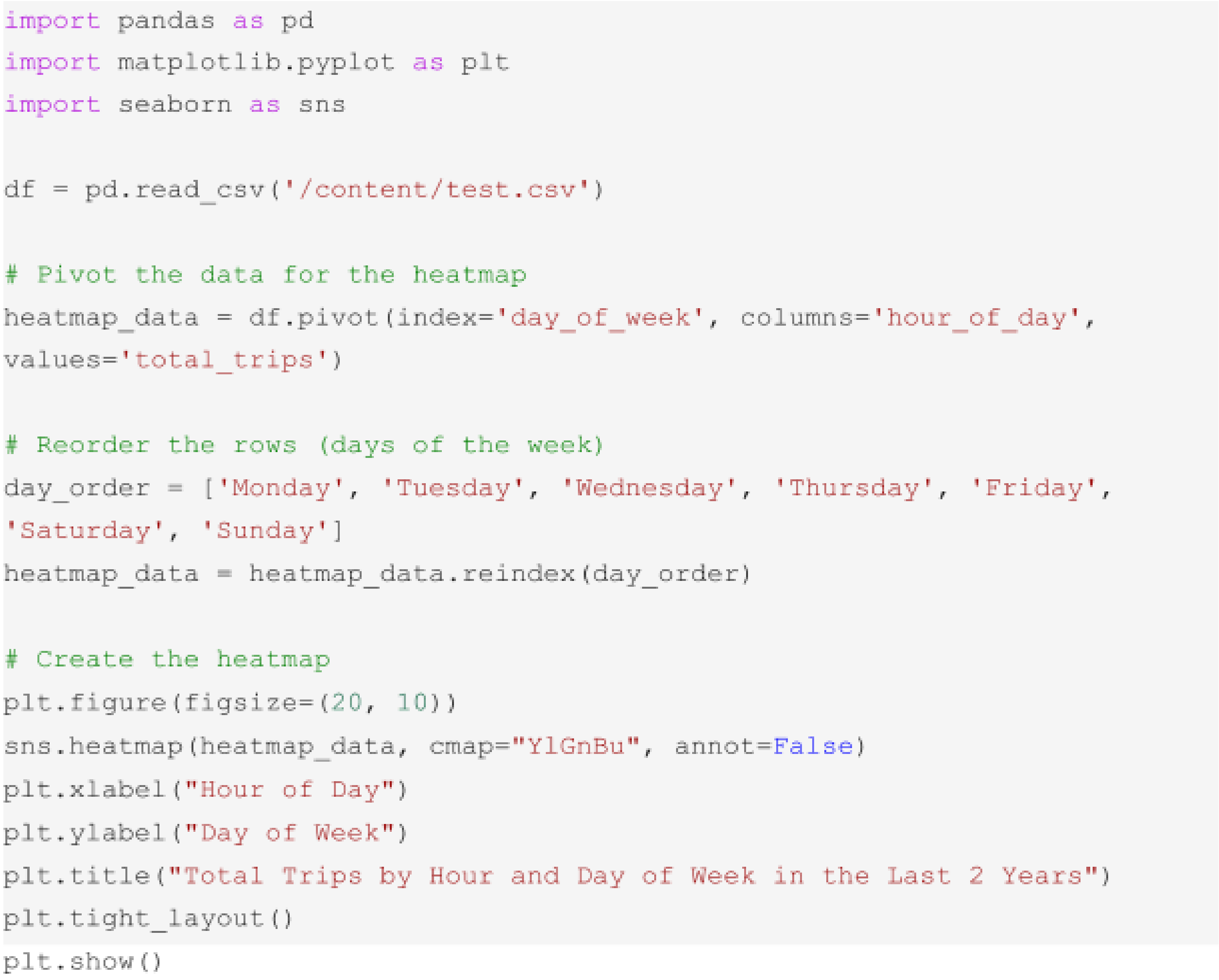


Row	day_of_week	hour_of_day	total_trips
1	Thursday	17	60108
2	Wednesday	17	59939
3	Thursday	16	59010
4	Wednesday	16	58919
5	Thursday	15	58418
6	Friday	17	57910
7	Thursday	18	57285
8	Tuesday	17	56936
9	Wednesday	15	56858
10	Friday	18	56247
11	Wednesday	18	56168
12	Thursday	14	56129
13	Friday	16	55093
14	Tuesday	16	55081
15	Wednesday	13	54712
16	Thursday	13	54670
17	Friday	15	54668
18	Wednesday	14	54578
19	Friday	14	53552
20	Thursday	12	53393
21	Tuesday	15	52949

Interpretation

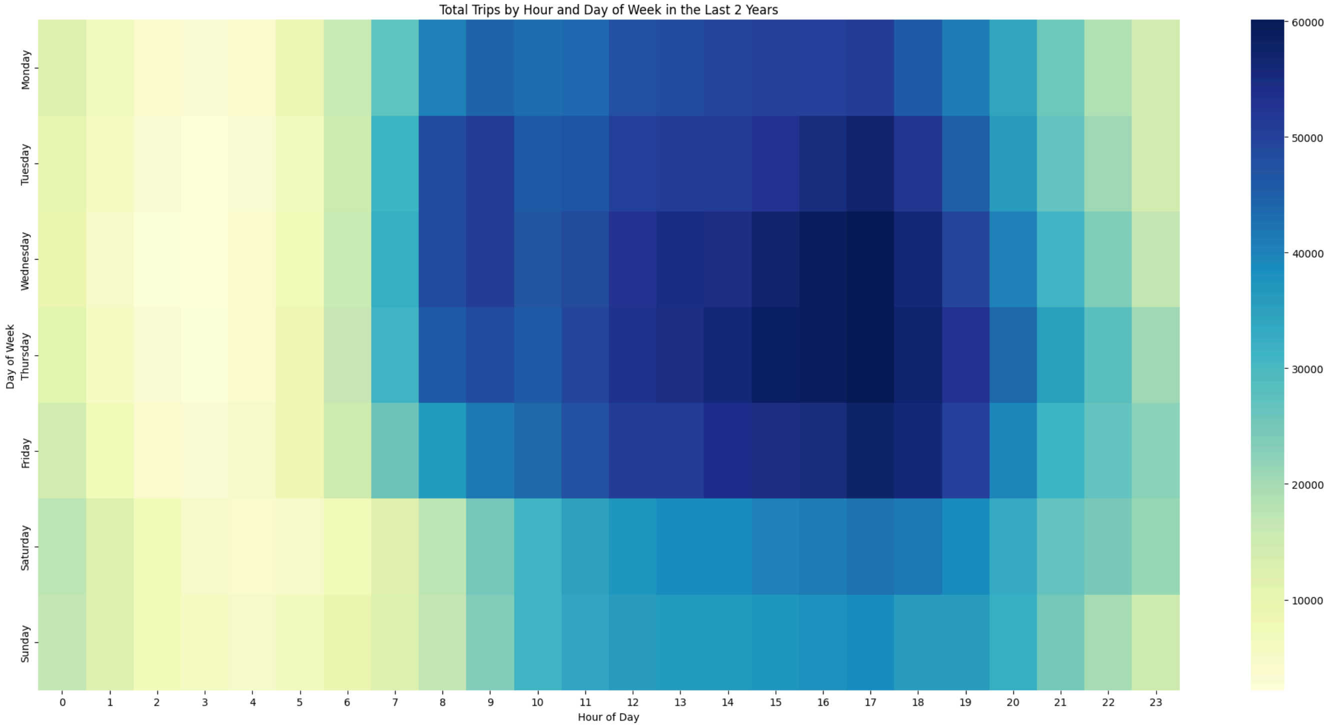
- A peak in trip volume is observed on Thursdays at 5 PM (17:00), with a total of 60,108 trips.
- This peak, followed by Wednesday's high demand, presents a key opportunity for resource optimization.
- The overall weekday trend, particularly on Thursday, Wednesday, and Friday, of higher trip volume compared to other days underscores the importance of strategic driver deployment during these periods.
- The comparatively lower weekend demand necessitates further investigation to understand contributing factors and explore potential strategies for service optimization.

Python code for visualization using Seaborn library



```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

df = pd.read_csv('/content/test.csv')

# Pivot the data for the heatmap
heatmap_data = df.pivot(index='day_of_week', columns='hour_of_day',
values='total_trips')

# Reorder the rows (days of the week)
day_order = ['Monday', 'Tuesday', 'Wednesday', 'Thursday', 'Friday',
'Saturday', 'Sunday']
heatmap_data = heatmap_data.reindex(day_order)

# Create the heatmap
plt.figure(figsize=(20, 10))
sns.heatmap(heatmap_data, cmap="YlGnBu", annot=False)
plt.xlabel("Hour of Day")
plt.ylabel("Day of Week")
plt.title("Total Trips by Hour and Day of Week in the Last 2 Years")
plt.tight_layout()
plt.show()
```

Visualization



Understanding the Heatmap

A heatmap is a visual tool that uses color to represent data values across two dimensions. In this analysis:

- X-Axis (Horizontal): Represents the hour of the day (from 0 to 23). Each column shows a specific hour, such as 8 AM, 3 PM, or 11 PM in the 24 hour clock format.
- Y-Axis (Vertical): Represents the day of the week, with rows for Monday through Sunday. This lets you see daily patterns.
- Color Scale: The colors indicate the total number of taxi trips. Darker shades (deep blues in our chosen color scheme) represent higher trip volumes, while lighter colors show lower volumes.

How to Read It:

- Look across the heatmap to see which hours on which days have the darkest colors. For instance, if Thursday afternoon and evenings are dark, that means those times have the highest taxi demand.
- Compare different days: a row that is consistently darker across most hours indicates a busier day overall.

Business Insights & Recommendations

Insights

- The heatmap clearly shows that Thursday and Wednesday experience the highest trip volumes. The trip counts peak between 12 PM and 6 PM. This aligns with commuter traffic and post-work activities.
- Similarly, Monday, Tuesday and Friday exhibits a strong surge during the same late afternoon to early evening period.
- Weekends have a milder but steady volume from late morning (around 10 AM) through early evening (6 PM), possibly reflecting brunch outings, shopping, or weekend errands.
- Across all days, 1–5 AM remains comparatively light (pale hues), suggesting minimal overnight demand.

Recommendations

1. **Driver Allocation & Scheduling**
    - **Prioritize Evenings**: Increase driver availability during peak hours (12–6 PM) on weekdays to reduce wait times and capitalize on high demand.
2. **Promotions**
    - **Off-Peak Incentives**: Offer discounts or loyalty rewards in early morning hours (1–5 AM) or midweek mornings (7–10 AM) to attract riders during slow periods.
3. **Marketing**
    - **Weekend Entertainment Tie-Ins**: Collaborate with local restaurants or shopping centers on Saturday/Sunday to promote special taxi deals. ridership or direct booking links.
4. **Continuous Monitoring**
    - **Regularly monitor** the heatmap to spot changes in rider behavior (seasonal shifts, new events, or external factors).

Conclusion

This project explored key trends in the Chicago Taxi Trips dataset. In Part I, we identified the companies with the largest month-over-month increases in trips and the steepest drops in fare-per-mile—trends that hint at seasonal demand shifts and pricing strategy changes.

In Part II, our day-and-hour analysis revealed that Wednesday and Thursday evenings are the busiest, while early mornings remain quiet. This insight can help optimize driver scheduling, dynamic pricing, and targeted promotions.

Overall, these findings offer a clear view of when taxi demand peaks and dips, providing actionable guidance to improve operations and enhance customer satisfaction.