



GOODCABS CAB SERVICES



Presented by Aditi Teli

Project Challenge by Codebasics

Table of Contents

- Business Overview
- Problem Statement
- Primary & Secondary Analysis
- Ad-Hoc Analysis
- Dashboard



What Do You Know About Us?

Goodcabs, a cab service company established two years ago, has gained a strong foothold in the Indian market by focusing on tier-2 cities. Unlike other cab service providers, Goodcabs is committed to supporting local drivers, helping them make a sustainable living in their hometowns while ensuring excellent service to passengers. With operations in ten tier-2 cities across India, Goodcabs has set ambitious performance targets for 2024 to drive growth and improve passenger satisfaction.





Problem Statement

The Goodcabs management team aims to evaluate the company's performance across key metrics, including trip volume, passenger satisfaction, repeat passenger rate, trip distribution, and the balance between new and repeat passengers. These insights are critical for operational decision-making and need to be prepared on an urgent basis. To ensure quality and accuracy, the analytics manager has provided supporting notes for the analysis.

This initiative seeks to provide insights that can directly inform strategies to improve service quality, customer retention, and overall operational efficiency.

Primary & Secondary Analysis

1. Top and Bottom Performing Cities based on number of trips

```
In [14]: # Top 3 cities:  
top_3_cities = citywise_trip_count.nlargest(3, "total_trips")  
top_3_cities
```

```
Out[14]:   city_id  city_name  total_trips  
          7      RJ01     Jaipur    76888  
          9      UP01     Lucknow   64299  
          2      GJ01     Surat     54843
```

```
In [15]: # Bottom 3 cities:  
  
bottom_3_cities = citywise_trip_count.nsmallest(3, "total_trips")  
bottom_3_cities
```

```
Out[15]:   city_id  city_name  total_trips  
          4      KA01     Mysore    16238  
          8      TN01     Coimbatore  21104  
          0      AP01     Visakhapatnam 28366
```

- Jaipur, Lucknow and Surat are Top 3 cities with highest trips
- Mysore, Coimbatore and Vishakhapatnam are Bottom 3 cities



2. Average Fare per trip by city

```
In [22]: # Highest Average fare trip:
```

```
highest_avg_fare_city = city_metric.loc[city_metric["average_fare_per_trip"].idxmax()]
```

```
In [23]: highest_avg_fare_city
```

```
Out[23]: city_id          RJ01  
         city_name        Jaipur  
         average_fare_per_trip  483.918128  
         average_trip_distance  30.023125  
         Name: 7, dtype: object
```

```
In [24]: # Smallest Average fare trip:
```

```
lowest_avg_per_trip = city_metric.loc[city_metric["average_fare_per_trip"].idxmin()]  
lowest_avg_per_trip
```

```
Out[24]: city_id          GJ01  
         city_name        Surat  
         average_fare_per_trip  117.272925  
         average_trip_distance  10.997247  
         Name: 2, dtype: object
```

- Jaipur is the city with highest average fare trip
- Surat is the city with lowest average fare trip



3. Average rating by city and passenger type

```
In [29]: # Identify cities with highest and Lowest average passenger ratings

highest_avg_passenger_rating = city_ratings.loc[city_ratings["avg_passenger_rating"].idxmax()]
lowest_avg_passenger_rating = city_ratings.loc[city_ratings["avg_driver_rating"].idxmin()]

print("Highest average passenger rating:\n", highest_avg_passenger_rating)
print("Lowest average passenger rating:\n", lowest_avg_passenger_rating)

Highest average passenger rating:
  city_id              KL01
  city_name            Kochi
  passenger_type       new
  avg_passenger_rating    8.987394
  avg_driver_rating      8.98535
  Name: 10, dtype: object
Lowest average passenger rating:
  city_id              GJ01
  city_name            Surat
  passenger_type       repeated
  avg_passenger_rating    5.995511
  avg_driver_rating      6.479441
  Name: 5, dtype: object
```

- Kochi has the higher average passenger rating whereas Surat has lowest average passenger rating

4. Average rating by city and driver

```
In [30]: # Identify cities with highest and lowest average driver rating:
```

```
highest_avg_driver_rating = city_ratings.loc[city_ratings["avg_driver_rating"].idxmax()]
lowest_avg_driver_rating = city_ratings.loc[city_ratings["avg_driver_rating"].idxmin()]

print("Highest average driver rating\n", highest_avg_driver_rating)
print("Lowest average driver rating\n", lowest_avg_driver_rating)
```

```
Highest average driver rating
  city_id          AP01
  city_name        Visakhapatnam
  passenger_type   repeated
  avg_passenger_rating  7.989628
  avg_driver_rating    8.992701
Name: 1, dtype: object
Lowest average driver rating
  city_id          GJ01
  city_name        Surat
  passenger_type   repeated
  avg_passenger_rating  5.995511
  avg_driver_rating    6.479441
Name: 5, dtype: object
```

- Average driver rating is highest in Vishakhapatnam and Surat has lowest average driver rating.

5. Peak and Low Demand Months by city

```
In [37]: # Identify the month with highest and lowest total trips for each city:
```

```
peak_demand_months = city_month_trips.loc[city_month_trips.groupby('city_id')['total_trips'].idxmax()]
low_demand_months = city_month_trips.loc[city_month_trips.groupby('city_id')['total_trips'].idxmin()]
```

```
In [38]: # Reorder columns for clarity
```

```
peak_demand_months = peak_demand_months[['city_id", "city_name", "month_name", "total_trips"]]
low_demand_months = low_demand_months[['city_id", "city_name", "month_name", "total_trips']]
```

```
In [39]: # Display results
```

```
peak_demand_months
```

```
Out[39]:
```

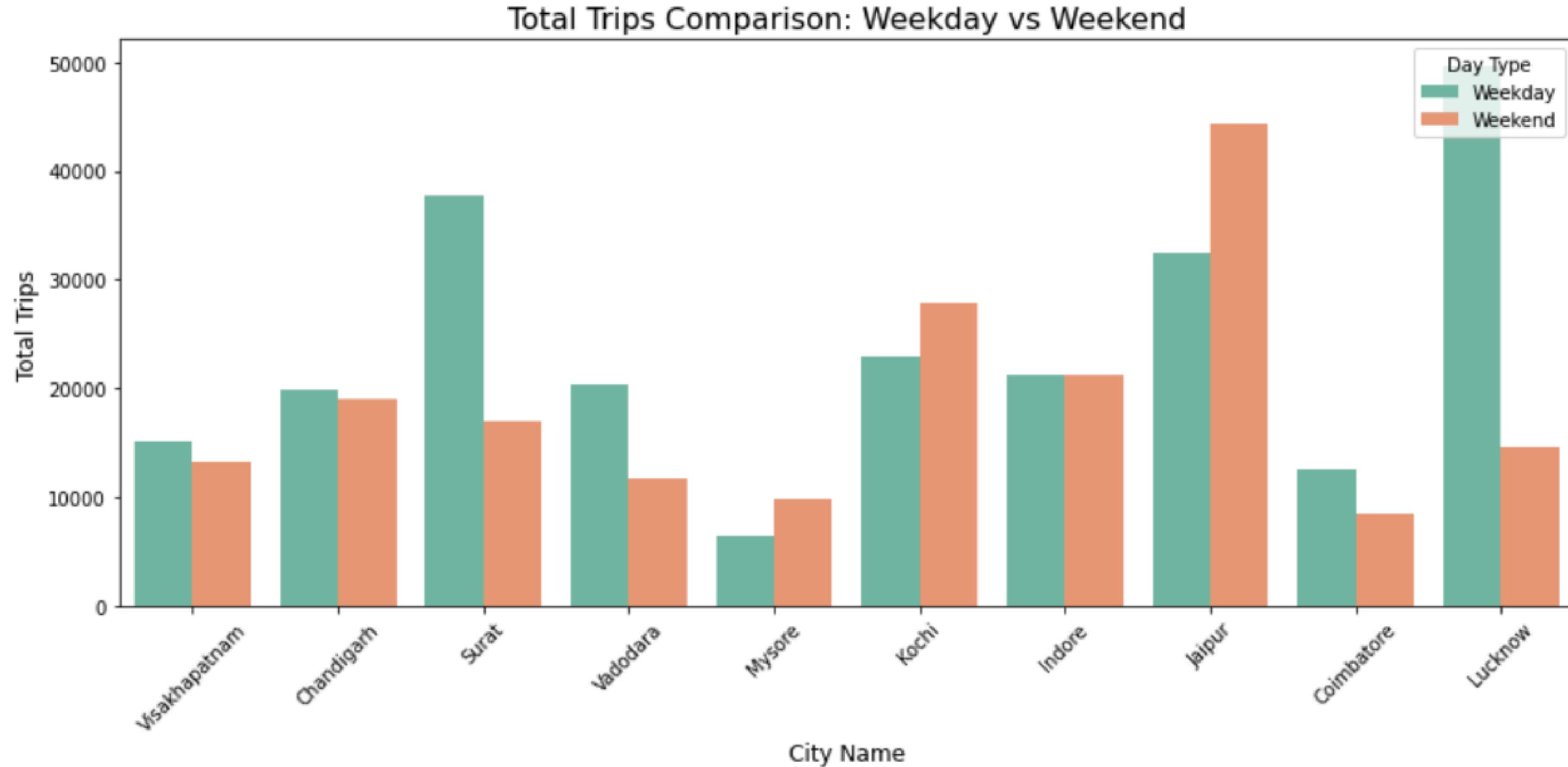
	city_id	city_name	month_name	total_trips
0	AP01	Visakhapatnam	April	4938
7	CH01	Chandigarh	February	7387
12	GJ01	Surat	April	9831
18	GJ02	Vadodara	April	5941
29	KA01	Mysore	May	3007
35	KL01	Kochi	May	10014
41	MP01	Indore	May	7787
43	RJ01	Jaipur	February	15872
52	TN01	Coimbatore	March	3680
55	UP01	Lucknow	February	12060

```
In [40]: low_demand_months
```

```
Out[40]:
```

	city_id	city_name	month_name	total_trips
2	AP01	Visakhapatnam	January	4468
6	CH01	Chandigarh	April	5566
14	GJ01	Surat	January	8358
21	GJ02	Vadodara	June	4685
26	KA01	Mysore	January	2485
33	KL01	Kochi	June	6399
39	MP01	Indore	June	6288
45	RJ01	Jaipur	June	9842
51	TN01	Coimbatore	June	3158
59	UP01	Lucknow	May	9705

6. Weekend vs Weekday Trip Demand City

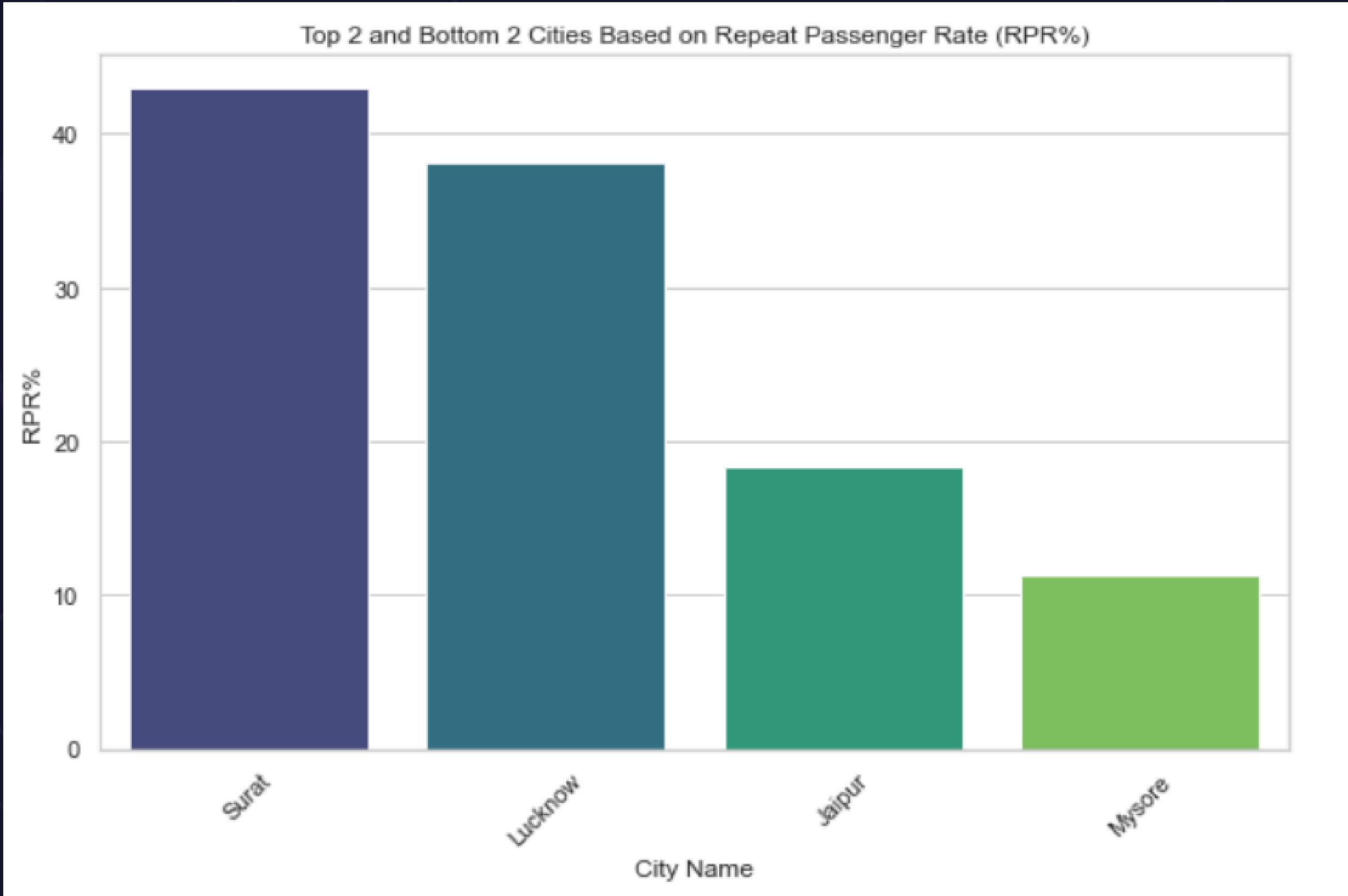




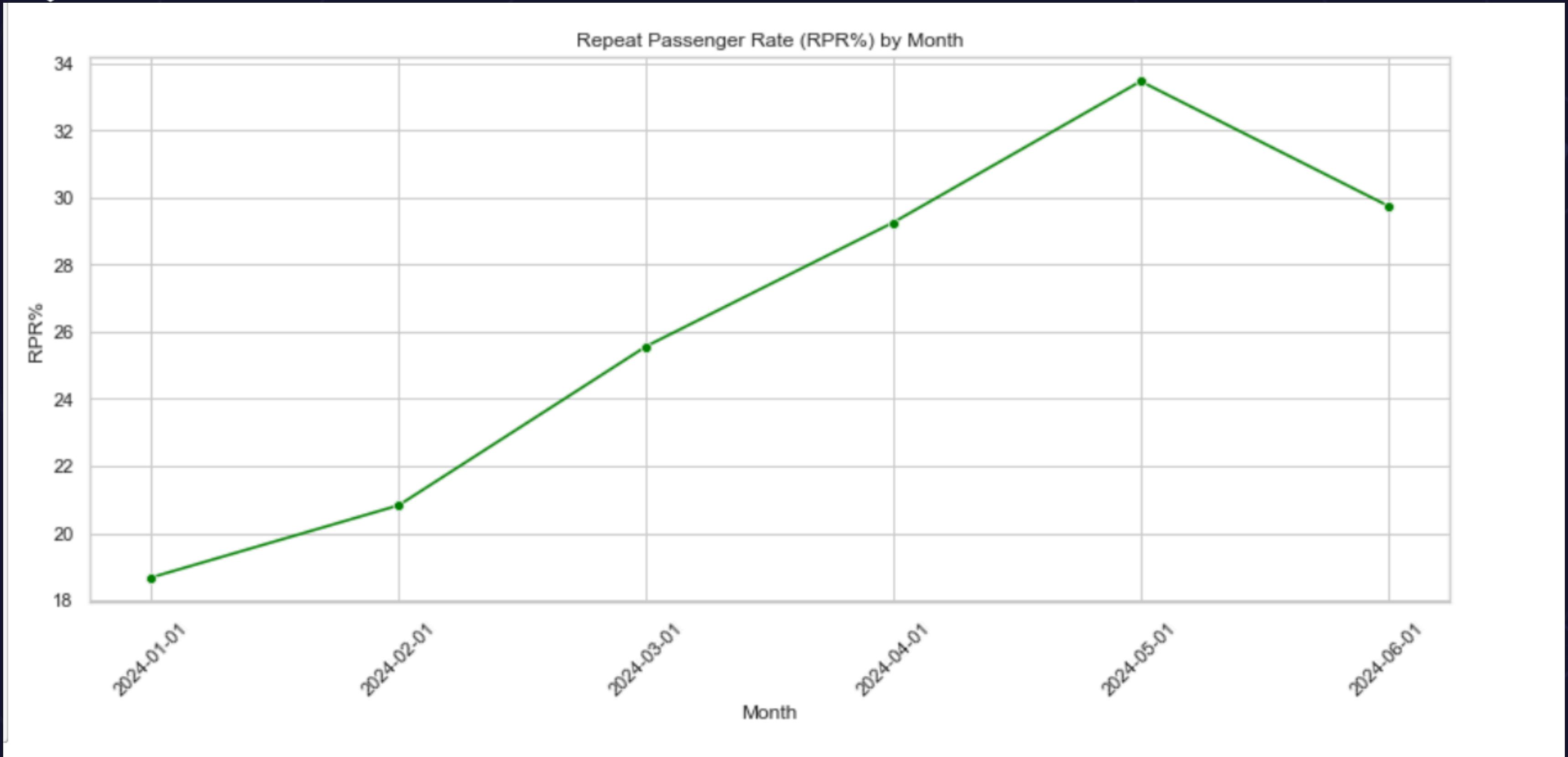
8. Identify cities contributing most to higher trip frequencies:



9. Highest and Lowest Repeat Passenger Rate (RPR%) by City



9. Repeat Passenger Rate (RPR%) by month.



DASHBOARD

All Cities

Selected City

425.90K

Total Trips

108M

Total Revenue

7.66

Average Passenger Rat...

7.83

Average Driver Rating

25.73

RPR%

Select City

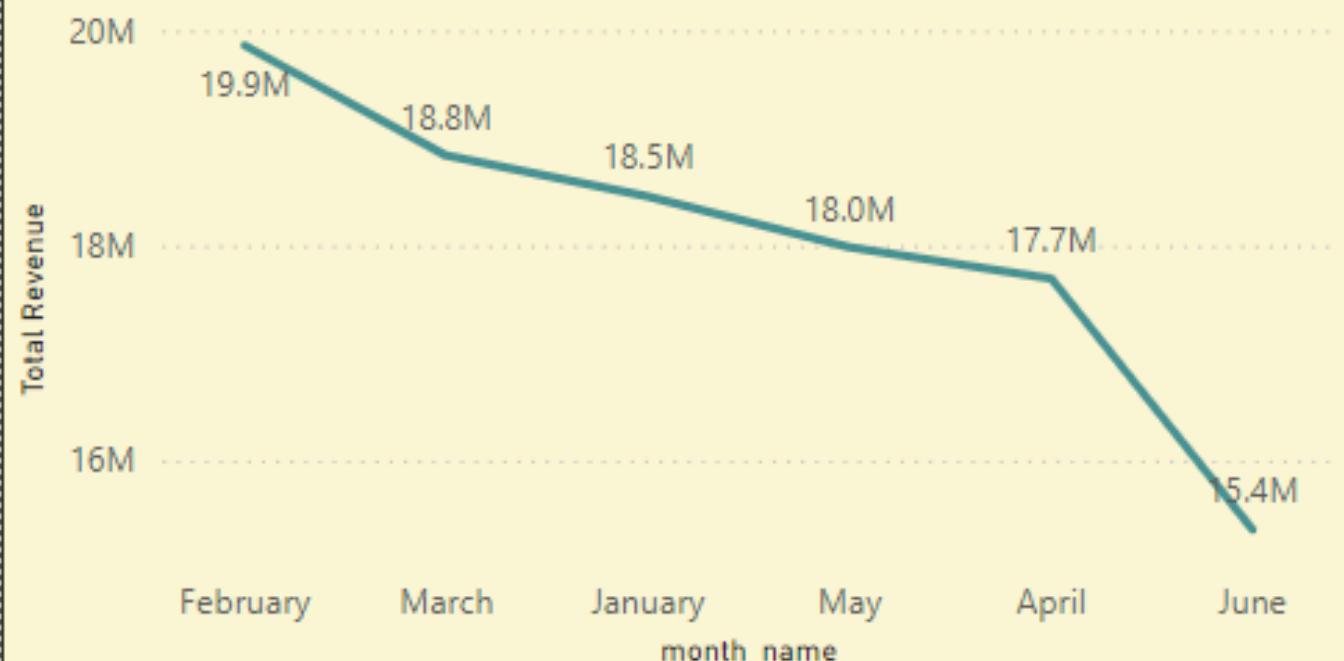
city_name

All



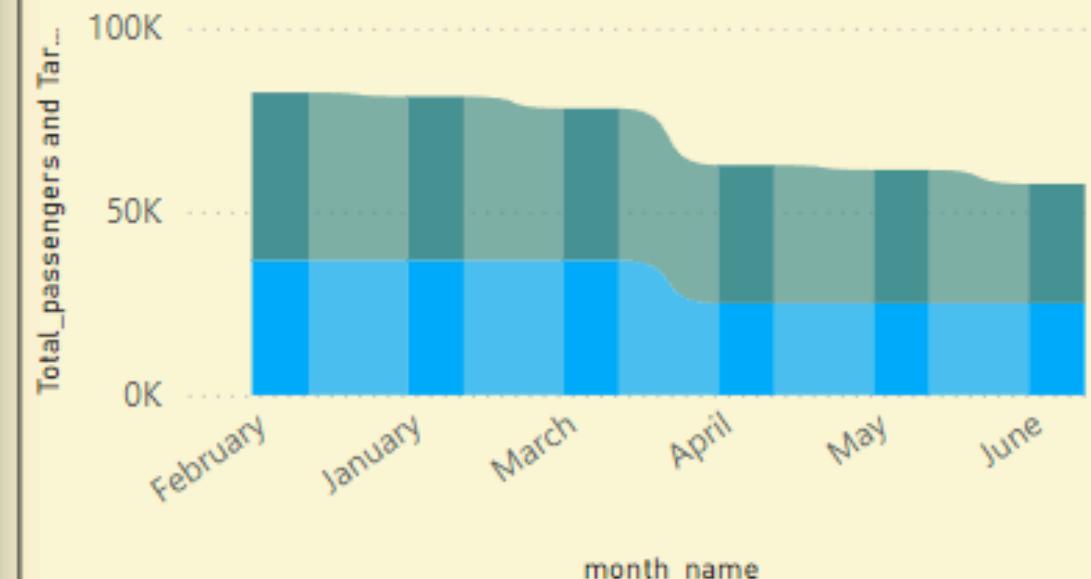
...

Monthly Total Revenue

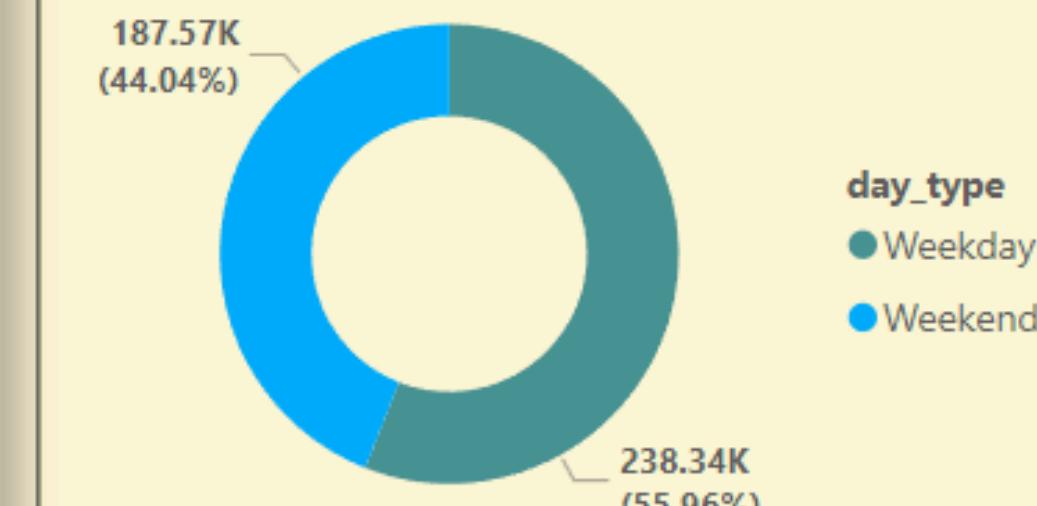


Actual vs. Target Passengers

Total_passenger Target_passenger

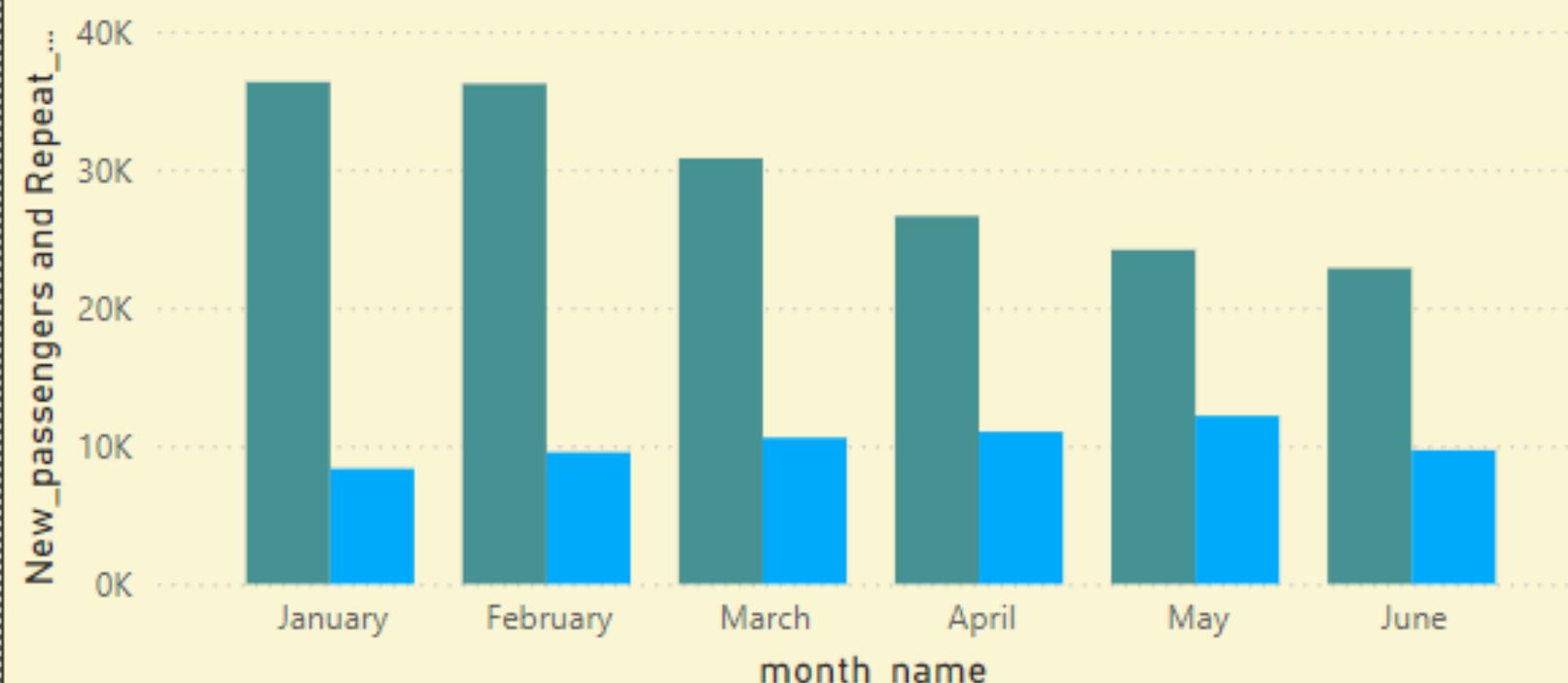


Distribution of total trip by Day type



New Passengers vs. Repeat Passengers

New_passenger Repeat_passenger



city_name	Total Distance_travelled (km)	Average of fare_amount	Average of distance_travelled(km)	Average Ratings Status
-----------	-------------------------------	------------------------	-----------------------------------	------------------------

Jaipur	2308418	483.92	30.02	Exceeded
Kochi	1220167	335.25	24.07	Exceeded
Mysore	267877	249.71	16.50	Exceeded
Chandigarh	916783	283.69	23.52	Missed
Coimbatore	316121	166.98	14.98	Missed
Indore	700629	179.84	16.50	Missed
Lucknow	804571	147.18	12.51	Missed
Surat	603122	117.27	11.00	Missed
Vadodara	368867	118.57	11.52	Missed
Visakhapatnam	639765	282.67	22.55	Missed

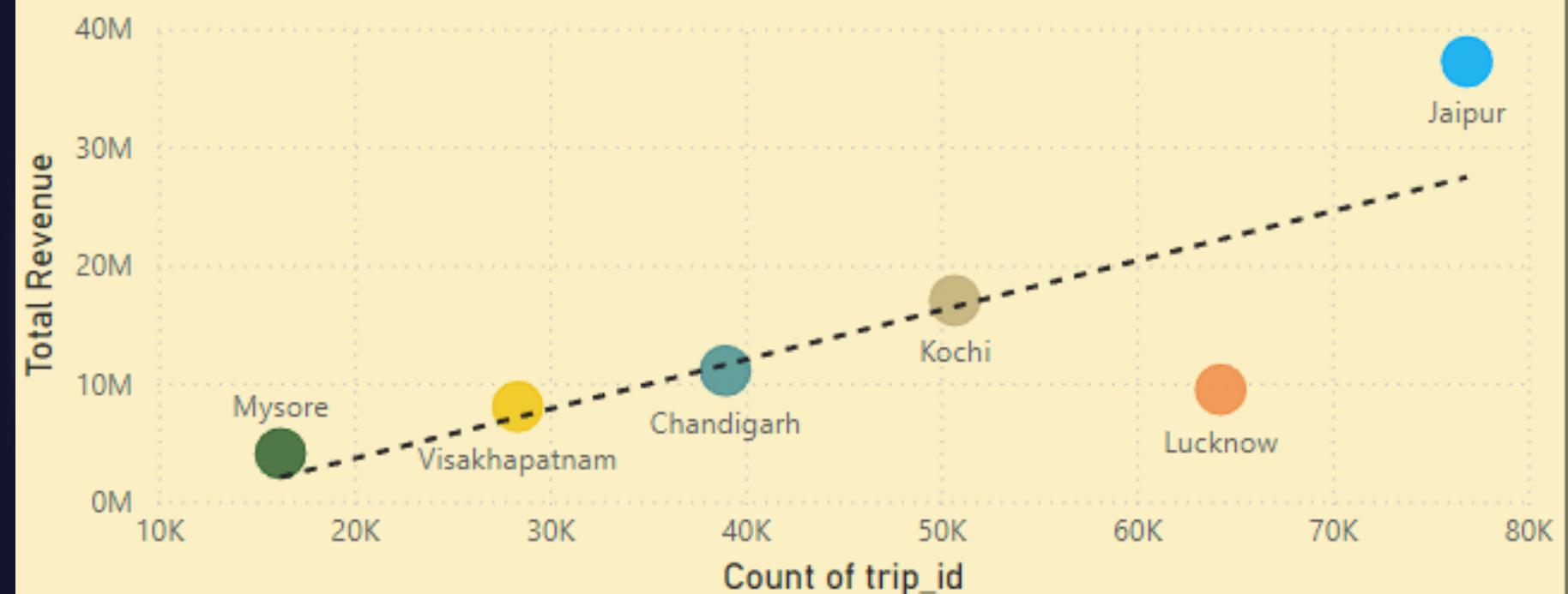
city_category

Tourist

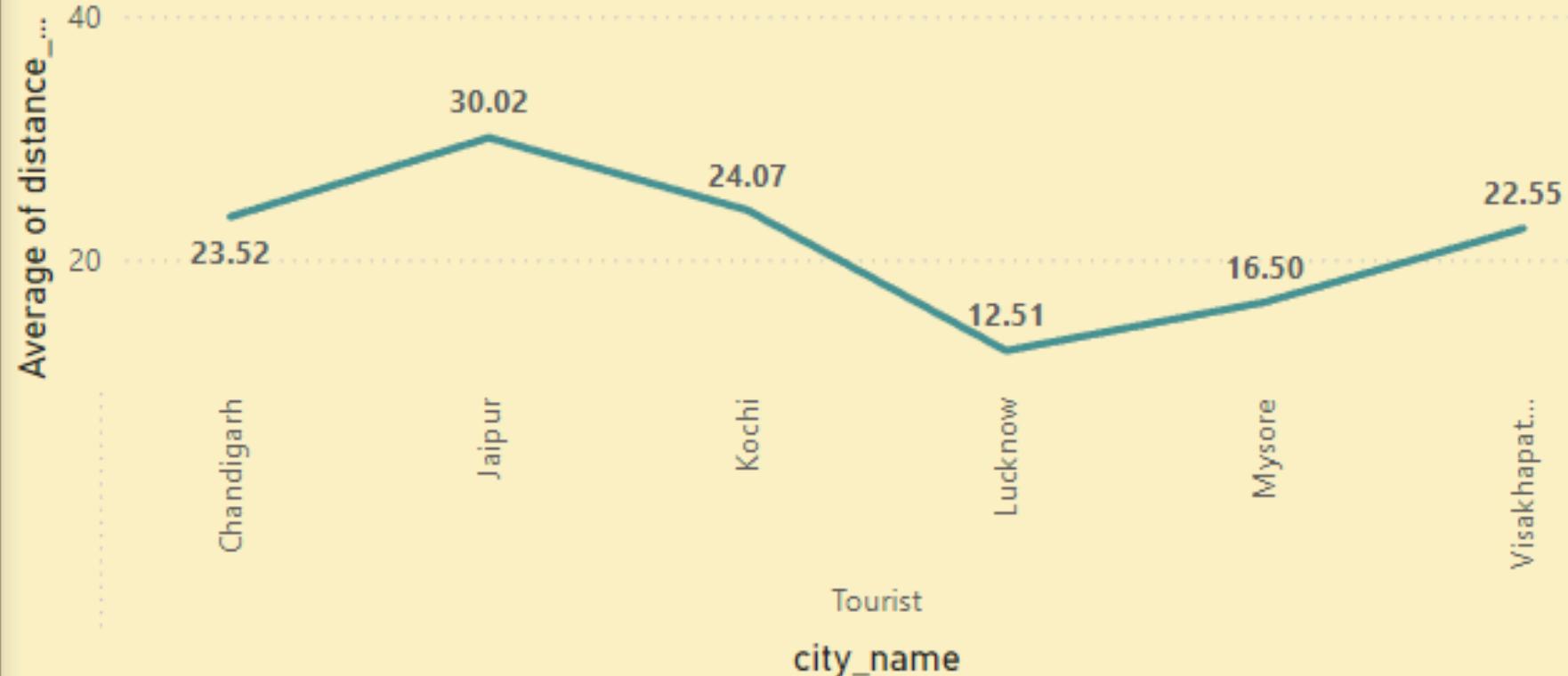


Correlation between Total trips vs. Total revenue

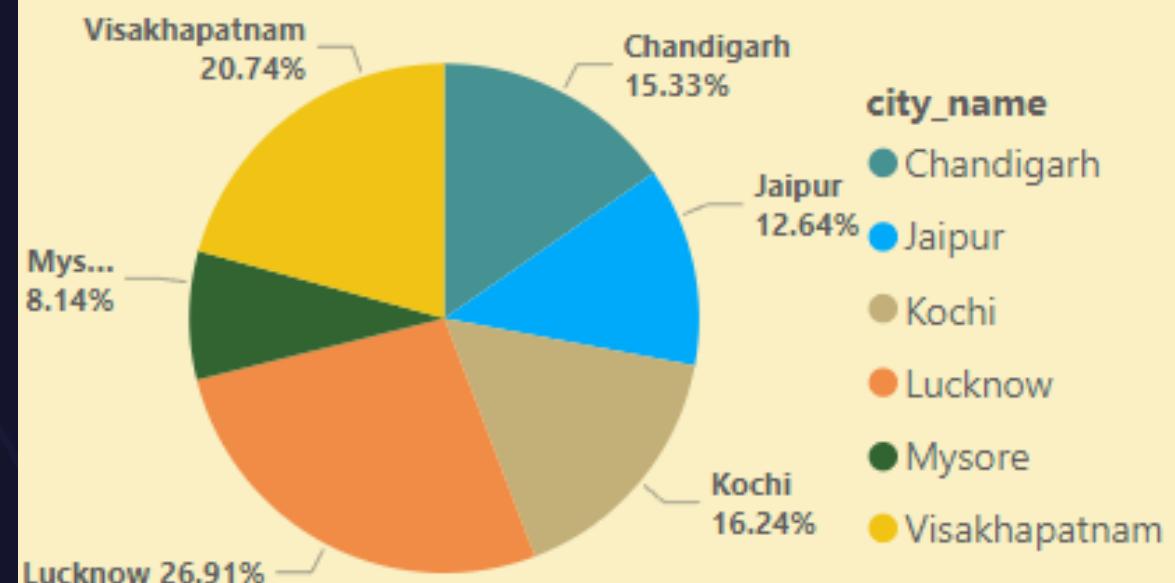
city_name ● Chandigarh ● Jaipur ● Kochi ● Lucknow ● Mysore ● Visakhapatnam



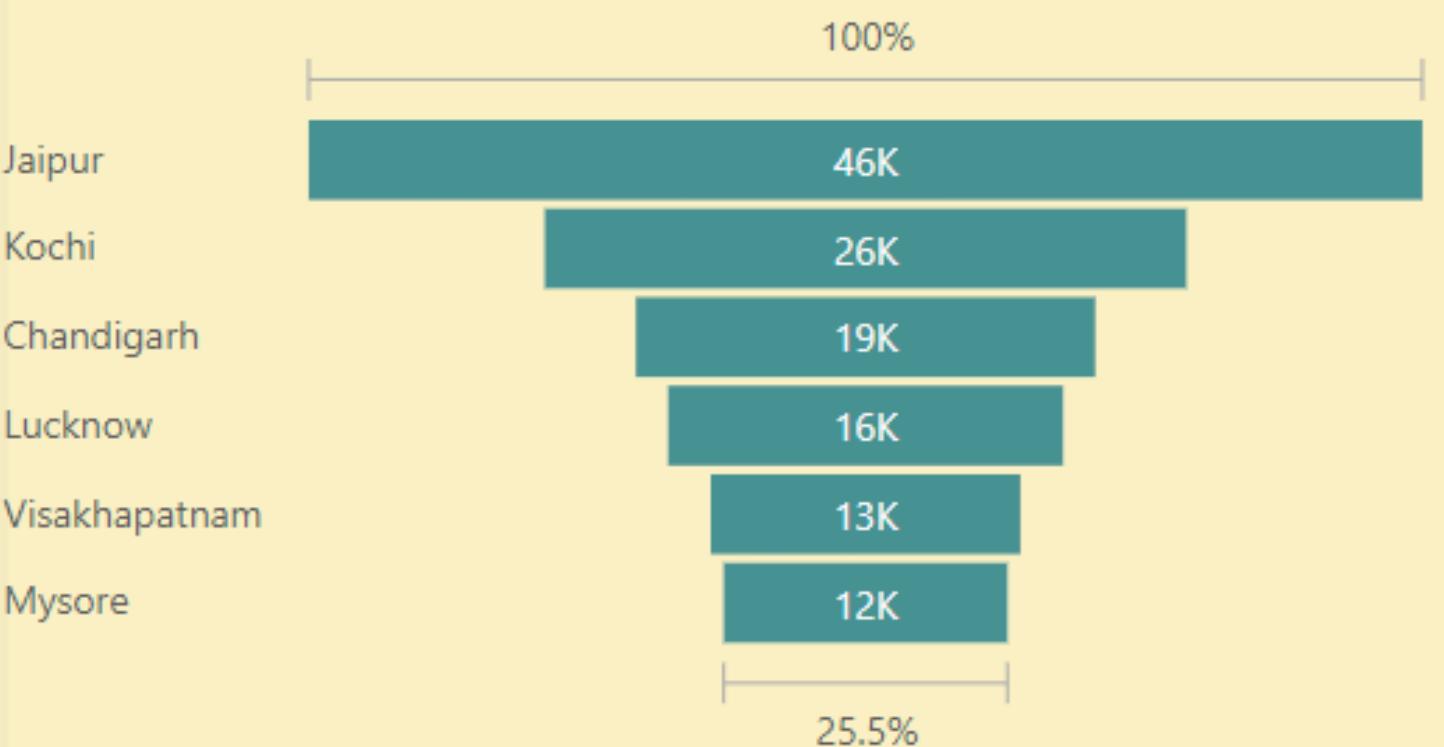
Average of distance_travelled (km)



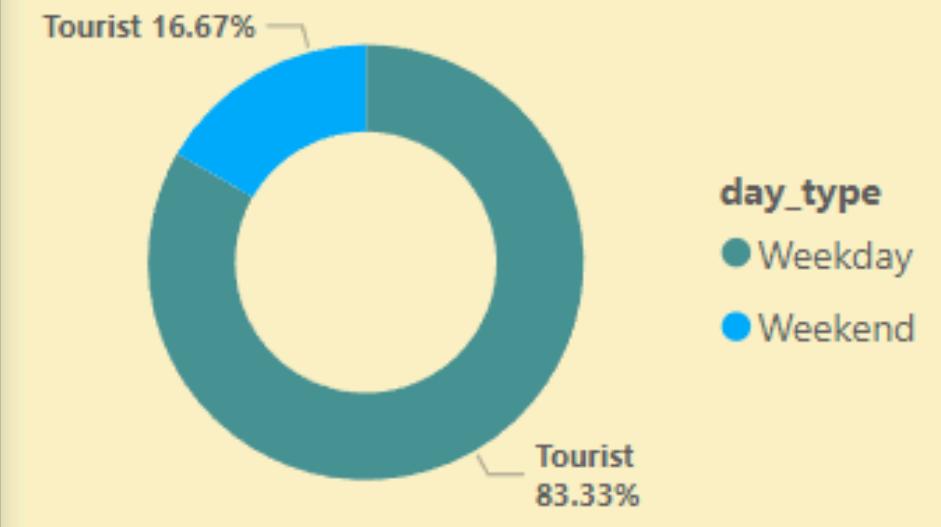
RPR% by city_name



New Passengers



Distribution of total trip by Day type



AD HOC ANALYSIS

Generate a report that displays the total trips, average fare per kilometer, average fare per trip, and the percentage contribution of each city's trip to the overall trips.



```
SELECT
    c.city_name AS City,
    COUNT(f.trip_id) AS Total_Trips,
    SUM(f.fare_amount) / SUM(f.distance_travelled_km) AS Avg_Fare_Per_KM,
    SUM(f.fare_amount) / COUNT(f.trip_id) AS Avg_Fare_Per_Trip,
    COUNT(f.trip_id) * 100.0 / (SELECT COUNT(trip_id) FROM fact_trips) AS Percentage_Contribution
FROM fact_trips f
JOIN dim_city c ON f.city_id = c.city_id
GROUP BY c.city_name
ORDER BY Total_Trips DESC;
```

	City	Total_Trips	Avg_Fare_Per_KM	Avg_Fare_Per_Trip	Percentage_Contribution
▶	Jaipur	76888	16.1182	483.9181	18.05294
	Lucknow	64299	11.7622	147.1804	15.09710
	Surat	54843	10.6638	117.2729	12.87688
	Kochi	50702	13.9305	335.2451	11.90459
	Indore	42456	10.8977	179.8386	9.96847
	Chandigarh	38981	12.0622	283.6870	9.15255
	Vadodara	32026	10.2942	118.5662	7.51955
	Visakhapatnam	28366	12.5332	282.6723	6.66020
	Coimbatore	21104	11.1476	166.9822	4.95512
	Mysore	16238	15.1366	249.7072	3.81261

Generate a report that evaluates the target performance for trips at the monthly and city level. For each city and month, compare the actual total trips with the target trips and categorise the performance as follows:



1. If actual trips are greater than target trips, mark it as "Above target"
2. If actual trips are less than or equal to target trips, mark it as "Below target"

Additionally, calculate the % difference between the actual and target trips to quantify the performance gap.

```
SELECT
    c.city_name AS City,
    f.city_id,
    f.month,
    SUM(f.total_passengers) AS Actual_Trips,
    t.total_target_trips AS Target_Trips,
    CASE
        WHEN SUM(f.total_passengers) > t.total_target_trips THEN 'Above target'
        WHEN SUM(f.total_passengers) < t.total_target_trips THEN 'Below target'
        ELSE 'Equals Target'
    END AS Performance_Status,
    ROUND((SUM(f.total_passengers) - t.total_target_trips) * 100.0 / t.total_target_trips, 2) AS
    Percentage_Difference
FROM trips_db.fact_passenger_summary f
JOIN trips_db.dim_city c
    ON f.city_id = c.city_id
JOIN targets_db.monthly_target_trips t
    ON f.city_id = t.city_id AND f.month = t.month
GROUP BY c.city_name, f.city_id, f.month, t.total_target_trips
ORDER BY f.month, c.city_name;
```

	City	city_id	month	Actual_Trips	Target_Trips	Performance_Status	Percentage_Difference
▶	Chandigarh	CH01	2024-01-01	4640	7000	Below target	-33.71
	Coimbatore	TN01	2024-01-01	2214	3500	Below target	-36.74
	Indore	MP01	2024-01-01	3876	7000	Below target	-44.63
	Jaipur	RJ01	2024-01-01	11845	13000	Below target	-8.88
	Kochi	KL01	2024-01-01	5660	7500	Below target	-24.53
	Lucknow	UP01	2024-01-01	4896	13000	Below target	-62.34
	Mysore	KA01	2024-01-01	2129	2000	Above target	6.45
	Surat	GJ01	2024-01-01	3616	9000	Below target	-59.82
	Vadodara	GJ02	2024-01-01	2633	6000	Below target	-56.12
	Visakhapatnam	AP01	2024-01-01	3163	4500	Below target	-29.71
	Chandigarh	CH01	2024-02-01	4957	7000	Below target	-29.19
	Coimbatore	TN01	2024-02-01	1993	3500	Below target	-43.06
	Indore	MP01	2024-02-01	3981	7000	Below target	-43.13
	Jaipur	RJ01	2024-02-01	12450	13000	Below target	-4.23
	Kochi	KL01	2024-02-01	5372	7500	Below target	-28.37
	Lucknow	UP01	2024-02-01	5188	13000	Below target	-60.09
	Mysore	KA01	2024-02-01	2290	2000	Above target	14.50
	Surat	GJ01	2024-02-01	3567	9000	Below target	-60.37
	Vadodara	GJ02	2024-02-01	2756	6000	Below target	-54.07
	Visakhapatnam	AP01	2024-02-01	3170	4500	Below target	-29.56
	Chandigarh	CH01	2024-03-01	4100	7000	Below target	-41.43
	Coimbatore	TN01	2024-03-01	1965	3500	Below target	-43.86
	Indore	MP01	2024-03-01	3833	7000	Below target	-45.24
	Jaipur	RJ01	2024-03-01	9257	13000	Below target	-28.79
	Kochi	KL01	2024-03-01	6213	7500	Below target	-17.16
	Lucknow	UP01	2024-03-01	4781	13000	Below target	-63.22
	Mysore	KA01	2024-03-01	2194	2000	Above target	9.70



City-Level Repeat Passenger Trip Frequency Report.

```
SELECT
    c.city_name AS City_Name,
    ROUND(SUM(CASE WHEN r.trip_count = 2 THEN r.repeat_passenger_count ELSE 0 END) * 100.0 /
SUM(r.repeat_passenger_count), 2) AS `2 Trips`,
    ROUND(SUM(CASE WHEN r.trip_count = 3 THEN r.repeat_passenger_count ELSE 0 END) * 100.0 /
SUM(r.repeat_passenger_count), 2) AS `3 Trips`,
    ROUND(SUM(CASE WHEN r.trip_count = 4 THEN r.repeat_passenger_count ELSE 0 END) * 100.0 /
SUM(r.repeat_passenger_count), 2) AS `4 Trips`,
    ROUND(SUM(CASE WHEN r.trip_count = 5 THEN r.repeat_passenger_count ELSE 0 END) * 100.0 /
SUM(r.repeat_passenger_count), 2) AS `5 Trips`,
    ROUND(SUM(CASE WHEN r.trip_count = 6 THEN r.repeat_passenger_count ELSE 0 END) * 100.0 /
SUM(r.repeat_passenger_count), 2) AS `6 Trips`,
    ROUND(SUM(CASE WHEN r.trip_count = 7 THEN r.repeat_passenger_count ELSE 0 END) * 100.0 /
SUM(r.repeat_passenger_count), 2) AS `7 Trips`,
    ROUND(SUM(CASE WHEN r.trip_count = 8 THEN r.repeat_passenger_count ELSE 0 END) * 100.0 /
SUM(r.repeat_passenger_count), 2) AS `8 Trips`,
    ROUND(SUM(CASE WHEN r.trip_count = 9 THEN r.repeat_passenger_count ELSE 0 END) * 100.0 /
SUM(r.repeat_passenger_count), 2) AS `9 Trips`,
    ROUND(SUM(CASE WHEN r.trip_count = 10 THEN r.repeat_passenger_count ELSE 0 END) * 100.0 /
SUM(r.repeat_passenger_count), 2) AS `10 Trips`
FROM trips_db.dim_repeat_trip_distribution r
JOIN trips_db.dim_city c
    ON r.city_id = c.city_id
GROUP BY c.city_name
ORDER BY c.city_name;
```

	City_Name	2 Trips	3 Trips	4 Trips	5 Trips	6 Trips	7 Trips	8 Trips	9 Trips	10 Trips
▶	Chandigarh	32.31	19.25	15.74	12.21	7.42	5.48	3.47	2.33	1.79
	Coimbatore	11.21	14.82	15.56	20.62	17.64	10.47	6.15	2.31	1.22
	Indore	34.34	22.69	13.40	10.34	6.85	5.24	3.26	2.38	1.51
	Jaipur	50.14	20.73	12.12	6.29	4.13	2.52	1.90	1.20	0.97
	Kochi	47.67	24.35	11.81	6.48	3.91	2.11	1.65	1.21	0.81
	Lucknow	9.66	14.77	16.20	18.42	20.18	11.33	6.43	1.91	1.10
	Mysore	48.75	24.44	12.73	5.82	4.06	1.76	1.42	0.54	0.47
	Surat	9.76	14.26	16.55	19.75	18.45	11.89	6.24	1.74	1.35
	Vadodara	9.87	14.17	16.52	18.06	19.08	12.86	5.78	2.05	1.61
	Visakhapatnam	51.25	24.96	9.98	5.44	3.19	1.98	1.39	0.88	0.92

Identify Cities with Highest and Lowest Total New Passengers



```
WITH CityNewPassengers AS (
    SELECT
        c.city_name AS City_Name,
        SUM(f.new_passengers) AS Total_New_Passengers
    FROM trips_db.fact_passenger_summary f
    JOIN trips_db.dim_city c
        ON f.city_id = c.city_id
    GROUP BY c.city_name
),
RankedCities AS (
    SELECT
        City_Name,
        Total_New_Passengers,
        CASE
            WHEN RANK() OVER (ORDER BY Total_New_Passengers DESC) <= 3 THEN 'Top 3'
            WHEN RANK() OVER (ORDER BY Total_New_Passengers ASC) <= 3 THEN 'Bottom 3'
            ELSE NULL
        END AS City_Category
    FROM CityNewPassengers
)
SELECT
    City_Name,
    Total_New_Passengers,
    City_Category
FROM RankedCities
WHERE City_Category IS NOT NULL
ORDER BY City_Category, Total_New_Passengers DESC;
```

	City_Name	Total_New_Passengers	City_Category
▶	Surat	11626	Bottom 3
	Vadodara	10127	Bottom 3
	Coimbatore	8514	Bottom 3
	Jaipur	45856	Top 3
	Kochi	26416	Top 3
	Chandigarh	18908	Top 3



Identify Month with Highest Revenue for each city.

```
WITH CityMonthlyRevenue AS (
    SELECT
        c.city_name AS City_Name,
        d.month_name AS Month_Name,
        SUM(f.fare_amount) AS Revenue
    FROM trips_db.fact_trips f
    JOIN trips_db.dim_city c ON f.city_id = c.city_id
    JOIN trips_db.dim_date d ON f.date = d.date
    GROUP BY c.city_name, d.month_name
),
CityTotalRevenue AS (
    SELECT
        City_Name,
        SUM(Revenue) AS Total_Revenue
    FROM CityMonthlyRevenue
    GROUP BY City_Name
),
HighestRevenueMonth AS (
    SELECT
        cmr.City_Name,
        cmr.Month_Name AS Highest_Revenue_Month,
        cmr.Revenue,
        ROUND((cmr.Revenue / ctr.Total_Revenue) * 100, 2) AS Percentage_Contribution
    FROM CityMonthlyRevenue cmr
    JOIN CityTotalRevenue ctr ON cmr.City_Name = ctr.City_Name
    WHERE cmr.Revenue = (
        SELECT MAX(Revenue)
        FROM CityMonthlyRevenue cmr2
        WHERE cmr2.City_Name = cmr.City_Name
    )
)
SELECT
    City_Name,
    Highest_Revenue_Month,
    Revenue,
    Percentage_Contribution
FROM HighestRevenueMonth
ORDER BY City_Name;
```

	City_Name	Highest_Revenue_Month	Revenue	Percentage_Contribution
▶	Chandigarh	February	2108290	19.07
	Coimbatore	April	612431	17.38
	Indore	May	1380996	18.09
	Jaipur	February	7747202	20.82
	Kochi	May	3333746	19.61
	Lucknow	February	1777269	18.78
	Mysore	May	745170	18.38
	Surat	April	1154909	17.96
	Vadodara	April	706250	18.60
	Visakhapatnam	April	1390682	17.34

Repeat Passenger Rate Analysis



```
WITH MonthlyMetrics AS (
    SELECT
        c.city_name AS City_Name,
        f.month AS Month,
        SUM(f.total_passengers) AS Total_Passengers,
        SUM(f.repeat_passengers) AS Repeat_Passengers,
        ROUND(SUM(f.repeat_passengers) * 100.0 / SUM(f.total_passengers), 2) AS
    Monthly_Repeat_Passenger_Rate
    FROM trips_db.fact_passenger_summary f
    JOIN trips_db.dim_city c ON f.city_id = c.city_id
    GROUP BY c.city_name, f.month
),
CityWideMetrics AS (
    SELECT
        c.city_name AS City_Name,
        SUM(f.total_passengers) AS Total_Passengers,
        SUM(f.repeat_passengers) AS Repeat_Passengers,
        ROUND(SUM(f.repeat_passengers) * 100.0 / SUM(f.total_passengers), 2) AS
    City_Repeat_Passenger_Rate
    FROM trips_db.fact_passenger_summary f
    JOIN trips_db.dim_city c ON f.city_id = c.city_id
    GROUP BY c.city_name
)
SELECT
    m.City_Name,
    m.Month,
    m.Total_Passengers,
    m.Repeat_Passengers,
    m.Monthly_Repeat_Passenger_Rate,
    cwm.City_Repeat_Passenger_Rate
FROM MonthlyMetrics m
JOIN CityWideMetrics cwm ON m.City_Name = cwm.City_Name
ORDER BY m.City_Name, m.Month;
```

City_Name	Month	Total_Passengers	Repeat_Passengers	Monthly_Repeat_Passenger_Rate	City_Repeat_Passenger_Rate
Chandigarh	2024-01-01	4640	720	15.52	21.14
Chandigarh	2024-02-01	4957	853	17.21	21.14
Chandigarh	2024-03-01	4100	872	21.27	21.14
Chandigarh	2024-04-01	3285	789	24.02	21.14
Chandigarh	2024-05-01	3699	969	26.20	21.14
Chandigarh	2024-06-01	3297	867	26.30	21.14
Coimbatore	2024-01-01	2214	392	17.71	23.05
Coimbatore	2024-02-01	1993	346	17.36	23.05
Coimbatore	2024-03-01	1965	427	21.73	23.05
Coimbatore	2024-04-01	1722	480	27.87	23.05
Coimbatore	2024-05-01	1543	504	32.66	23.05
Coimbatore	2024-06-01	1628	402	24.69	23.05
Indore	2024-01-01	3876	1033	26.65	32.68
Indore	2024-02-01	3981	1103	27.71	32.68
Indore	2024-03-01	3833	1091	28.46	32.68
Indore	2024-04-01	3646	1295	35.52	32.68
Indore	2024-05-01	3591	1563	43.53	32.68
Indore	2024-06-01	3152	1131	35.88	32.68
Jaipur	2024-01-01	11845	1422	12.01	17.43
Jaipur	2024-02-01	12450	1661	13.34	17.43
Jaipur	2024-03-01	9257	1840	19.88	17.43
Jaipur	2024-04-01	7856	1736	22.10	17.43
Jaipur	2024-05-01	7174	1842	25.68	17.43
Jaipur	2024-06-01	6956	1181	16.98	17.43
Kochi	2024-01-01	5660	795	14.05	22.40
Kochi	2024-02-01	5372	1005	18.71	22.40
Kochi	2024-03-01	6213	1348	21.70	22.40



INSIGHTS GENERATED



Goodcabs operates in 10 cities, categorized into 6 tourist destinations and 4 business hubs, with a significant focus on tourist cities. Below is a detailed breakdown, incorporating city descriptions:

Tourist Cities:

1. Jaipur (RJ01)

Known as the "Pink City," Jaipur is a historic and cultural hub famous for its royal palaces, forts, and vibrant bazaars. It attracts significant tourist footfall due to its heritage sites like Hawa Mahal and Amer Fort.

Insight: Jaipur records the highest total trips among tourist cities and also leads in revenue generation. Its success is driven by the highest average fare per trip (₹16.11/km) and strong new passenger additions (46K). Passenger as well as driver rating is 8.58 and 8.99 respectively. Around 57% of the trips are on weekends. In the month of April, May and June Total passengers exceeded target passenger. However, repeat passenger rate is low since its the tourist focused city and passenger comes for vacation only.



2. Lucknow (UP01)

Lucknow (UP01)

The capital of Uttar Pradesh, Lucknow is celebrated for its rich cultural heritage, historical monuments, and traditional Awadhi cuisine. Key attractions include Bara Imambara and Rumi Darwaza.

Insight: Lucknow shows moderate performance in terms of trips and revenue. In terms of passengers over the past six months total passengers exceeds target passengers. The passengers and driver ratings needs to be improved which is 6.49 and 6.62 respectively. Around 77.17% of the trips are taken on weekdays. January , February and March are the months were the total passengers travel more to this city.

3. Kochi (KL01)

Located in Kerala, Kochi is a prominent tourist destination, known for its backwaters, Chinese fishing nets, and spice markets.



Passenger and driver rating is 8.52 and 8.99 respectively. Around 54.8% trips are on weekends. Repeat passengers are lesser as compared to new passengers addition.

4. Mysore (KA01)

- Known for its grand Mysore Palace and cultural heritage, Mysore is a key tourist spot in Karnataka, attracting visitors for its architecture and Dasara festival celebrations.
- Insight: Mysore has the lowest trips in the tourist segment, impacting its revenue. Despite a strong average fare per kilometer (₹15.13), it struggles with low new passenger additions (12K) and the lowest RPR% (8.14%).

5. Visakhapatnam (AP01)

- A port city in Andhra Pradesh, Visakhapatnam is renowned for its scenic beaches, lush green hills, and cultural landmarks like the INS Kurusura Submarine Museum.



Insight: Visakhapatnam stands out with the highest driver rating, reflecting strong driver performance and customer satisfaction.

6. Chandigarh (CH01)

- A union territory and planned city, Chandigarh is known for its modern architecture, Rock Garden, and Sukhna Lake, making it both a tourist and administrative center.
- Insight: Chandigarh exhibits steady performance across metrics but lacks standout achievements in trips or revenue.



Business Cities

1. Surat (GJ01)

- Known as the diamond and textile capital of India, Surat is a bustling business hub in Gujarat with strong industrial and commercial activity.
- Insight: Surat leads in trips among business cities but shows the lowest average fare per trip and average ratings for passengers and drivers. It does, however, have the highest RPR% (33.2%), indicating strong passenger retention. Around 68% of the trips are taken on weekday.
- However, passenger and driver rating is not satisfactory. The repeat passenger rate is around 42%.

2. Indore (MP01)

- A commercial capital of Madhya Pradesh, Indore is famous for its food culture and thriving trade in textiles and pharmaceuticals.



Insight: Indore displays moderate performance across all metrics, with opportunities to increase repeat trips and revenue. RPR% is around 32% which indicates that customers are not really happy with the service. However, the passenger and driver rating is quite satisfactory.

3. Coimbatore (TN01)

- A major industrial hub in Tamil Nadu, Coimbatore specializes in textiles, engineering goods, and technology services.
- **Insight:** Coimbatore has the lowest trips among business cities, impacting its overall revenue contribution. It needs focused strategies to improve trip frequency and customer acquisition. The RPR% is 23% showing the signs that customers are taking less repeat trips.
- Passenger and driver rating is 7.88 and 7.69 respectively.



4. Vadodara (GJ02)

- Known for its industries and cultural heritage, Vadodara is both a business hub and a city with historical landmarks like the Laxmi Vilas Palace.
- Insight: Vadodara has balanced metrics but lacks standout performance. It falls behind in both new passenger additions and repeat passenger rates, requiring attention to boost growth.



Key Insights Across Cities

- Jaipur dominates the tourist segment in trips and revenue, driven by high average fares and passenger additions.
- Surat leads the business segment in trips but underperforms in revenue and ratings.
- Mysore and Coimbatore are lagging cities in their respective segments, requiring targeted improvements in trips and passenger engagement.
- The average fare per kilometer is a significant driver for revenue in tourist cities like Jaipur and Mysore.
- Repeat passenger rates highlight disparities: Surat excels with 33.2%, while Mysore lags with 8.14%.
- Passenger satisfaction metrics (ratings) indicate room for service quality improvements, particularly in Surat and Indore.



Key Insights Across Cities

- Jaipur dominates the tourist segment in trips and revenue, driven by high average fares and passenger additions.
- Surat leads the business segment in trips but underperforms in revenue and ratings.
- Mysore and Coimbatore are lagging cities in their respective segments, requiring targeted improvements in trips and passenger engagement.
- The average fare per kilometer is a significant driver for revenue in tourist cities like Jaipur and Mysore.
- Repeat passenger rates highlight disparities: Surat excels with 33.2%, while Mysore lags with 8.14%.
- Passenger satisfaction metrics (ratings) indicate room for service quality improvements, particularly in Surat and Indore.



Recommendations for Business Improvement

1. Maintain Customer Retention Rate: Enhance Repeat Passenger Engagement

- Challenge: Surat, a business hub, has the highest trips but struggles with low repeat passenger rates.
- Solution:
 - Introduce programs for frequent passengers, offering discounts or rewards for repeat bookings.
 - Use targeted marketing campaigns to remind users about trip history and encourage rebooking.

2. Optimize Fare Structures

Challenge: Surat has the lowest average fare per trip, while Jaipur thrives with higher average fares.

Solution:

Analyze pricing elasticity in Surat and other cities with low fares. Consider small increases while maintaining competitiveness.



3. Focus on Underperforming Cities

- Challenge: Coimbatore (business segment) and Mysore (tourist segment) have the lowest trips.
- Solution:
 - Increase marketing efforts in these cities by promoting services at key points like airports, IT parks, and tourist spots.
 - Partner with local businesses (Coimbatore) or tourism boards (Mysore) to create bundled travel packages.

4. Improve Passenger Experience

Challenge: Kochi has a higher passenger rating, while Surat lags behind.

Solution:

Conduct driver training programs in cities like Surat to improve service quality. Use customer feedback surveys to address specific concerns regarding ride quality or driver behavior.



5. Driver Incentive Programs

- Challenge: Driver ratings are highest in Visakhapatnam but low in Surat.
- Solution:
 - Introduce performance-based incentives for drivers to maintain high standards.
 - Provide recognition (certificates, cash rewards) for drivers who consistently receive high ratings.

6. Promote High-Performing Cities

- Challenge: Jaipur leads in total revenue and trips but lags in repeat passenger rates.
- Solution:
 - Focus on customer retention strategies by offering seasonal discounts and family/group travel packages.
 - Partner with hotels and tour operators to promote Jaipur as a bundled service destination.



7. Focus on Revenue Growth Drivers

- Challenge: Jaipur and Mysore have high average fares per km, but Mysore underperforms in revenue.
- Solution:
 - Implement demand-based pricing during peak seasons in Mysore.
 - Increase visibility in tourist hotspots and partner with local businesses for promotions.

8. City-Specific Strategies

Surat: Address the service quality issues and focus on repeat customers by improving driver and passenger experiences.

Indore: Promote services in commercial zones and IT parks to boost trips from the business community.

Kochi: Leverage its higher passenger rating by expanding premium service offerings targeting high-income customers.

Visakhapatnam: Build on the high driver rating by marketing the city as a quality-focused travel destination.



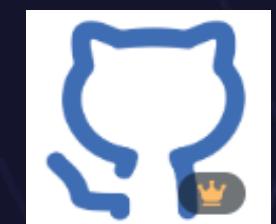
Target underperforming cities with aggressive promotions.

Retain passengers with loyalty programs and better experiences.
Incentivize drivers to improve service quality.

Adjust fares to optimize revenue without discouraging passengers.



Thank you!



AditiTeli-99 - Repositories

AditiTeli-99 has 14 repositories available. Follow their code on GitHub.

GitHub



<https://www.linkedin.com/in/aditi-teli-7a2504245/>

