GOOD CABS ANALYSIS

Objective

Good cabs ,a cab service company established two years ago, has gained strong foothold in the Indian market by focusing on tier-2 cities.

unlike other cab service providers,Goodcab is committed to supporting local drivers, helping them make a sustainable living in their hometowns while ensuring excellent service to passengers.

with operation in ten tier -2 cities across India, Good cabs has set ambitious performance targets for 2024 to drive growth and improve passenger satisfaction.

SQL PROJECT

To analyze:-

- Trip volume by cities.
- Passengers satisfaction levels
- Repeat passengers rate
- Revenue analysis
- Target new trips analysis
- Target new passengers analysis
- Ratio new vs. repeat passengers

Total trips

- Total trips

```
SELECT
    COUNT(trip_id) AS total_trips
FROM
    fact_trips;
```

total_trips
89037

Average fare per km

```sql
SELECT
    round(AVG(fare_amount / NULLIF(`distance_travelled(km)`, 0)),2) AS avg_fare_per_km
FROM
    fact_trips
WHERE
    `distance_travelled(km)` IS NOT NULL
    AND `distance_travelled(km)` > 0;
```

	avg_fare_per_km
▶	12.80

Average fare per trip

- Avg_fare per trip

```
SELECT
    round(SUM(fare_amount) / COUNT(DISTINCT trip_id),2) AS avg_fare_per_trip
FROM
    fact_trips
WHERE
    fare_amount IS NOT NULL;
```

	avg_fare_per_trip
▶	263.50

Fare per trip for each city

```sql
SELECT
    c.city_name,
    SUM(f.fare_amount) AS total_fare_amount,
    COUNT(f.trip_id) AS total_trips,
    ROUND(SUM(f.fare_amount) / COUNT(f.trip_id), 2) AS fare_per_trip
FROM
    fact_trips f
JOIN
    dim_city c
ON
    f.city_id = c.city_id
GROUP BY
    c.city_name
ORDER BY
    fare_per_trip DESC;
```

city_name	total_fare_amount	total_trips	fare_per_trip
Jaipur	7822351	16133	484.87
Kochi	3578149	10670	335.35
Visakhapatnam	1690634	5967	283.33
Chandigarh	2267003	8016	282.81
Mysore	867060	3477	249.37
Indore	1623788	9005	180.32
Coimbatore	737328	4428	166.51
Lucknow	1955703	13299	147.06
Vadodara	790844	6662	118.71
Surat	1336767	11380	117.47

Percentage wise contribution of trips by each city

```sql
SELECT
    city.city_name,
  round((COUNT(trip_id) * 100.0 / SUM(COUNT(trip_id)) OVER ()),2) AS percent_contribution_to_total_trips
FROM
    fact_trips
JOIN
    dim_city city ON fact_trips.city_id = city.city_id
GROUP BY
    city_name
    order by
    percent_contribution_to_total_trips desc;
```

city_name	percent_contribution_to_total_trips
Jaipur	18.39
Lucknow	16.43
Kochi	13.93
Surat	13.21
Indore	8.93
Chandigarh	7.86
Vadodara	7.68
Visakhapatnam	6.25
Coimbatore	4.29
Mysore	3.04

Jaipur has highest rate of contribution

Comparing average fare per trip with average distance per city

Jaipur has highest fare amount per trip and max distance travelled per trip

- Average fare per trip and Average distance to each city(so we can compare highest and lowest fare per trip to access the pricing efficiency across the

```
SELECT
    c.city_name,
    round(AVG(ft.fare_amount),2) AS avg_fare_per_trip,
    round(avg(ft.distance_travelled),2) as avg_dist_per_city
FROM
    fact_trips ft
JOIN
    dim_city c ON ft.city_id = c.city_id
WHERE
    ft.fare_amount IS NOT NULL
GROUP BY
    c.city_name
    order by avg_fare_per_trip desc;
```

city_name	avg_fare_per_trip	avg_dist_per_city
Jaipur	487.38	30.35
Kochi	339.45	24.15
Visakhapatnam	278.27	22.27
Chandigarh	276.31	23.09
Mysore	252.27	16.70
Indore	178.30	16.32
Coimbatore	165.47	14.98
Lucknow	150.46	12.91
Vadodara	118.27	11.40
Surat	116.13	10.90

Average ratings :-
average passenger and driver ratings for each city and segmented by passenger type (new vs. repeated)

```
SELECT c.city_name,ft.passenger_type,
    ROUND(AVG(ft.passenger_rating), 2) AS avg_passenger_ratings,
    ROUND(AVG(ft.driver_rating), 2) AS avg_driver_rating
    FROM
    fact_trips ft
   join dim_city c
   on
   c.city_id=ft.city_id
   where ft.passenger_type="new" and
   passenger_rating is not null
   and driver_rating is not null
   group by c.city_name,ft.passenger_type
   order by avg_passenger_ratings desc;
```

city_name	passenger_type	avg_passenger_ratings	avg_driver_rating
Jaipur	new	8.99	8.98
Kochi	new	8.98	9.00
Mysore	new	8.98	8.98
Visakhapatnam	new	8.96	8.99
Chandigarh	new	8.50	8.00
Indore	new	8.49	7.98
Coimbatore	new	8.45	7.95
Surat	new	8.00	6.98
Lucknow	new	7.97	6.99
Vadodara	new	7.97	6.97

New vs. repeated

```
SELECT c.city_name,ft.passenger_type,
    ROUND(AVG(ft.passenger_rating), 2) AS avg_passenger_ratings,
    ROUND(AVG(ft.driver_rating), 2) AS avg_driver_rating
    FROM
    fact_trips ft
   join dim_city c
   on
   c.city_id=ft.city_id
   where ft.passenger_type="repeated" and
   passenger_rating is not null
   and driver_rating is not null
   group by c.city_name,ft.passenger_type
   order by avg_passenger_ratings desc;
```

city_name	passenger_type	avg_passenger_ratings	avg_driver_rating
Kochi	repeated	8.00	8.99
Mysore	repeated	7.98	8.94
Visakhapatnam	repeated	7.97	8.99
Jaipur	repeated	7.97	8.99
Chandigarh	repeated	7.54	7.51
Coimbatore	repeated	7.47	7.48
Indore	repeated	7.47	7.49
Vadodara	repeated	5.99	6.51
Surat	repeated	5.99	6.48
Lucknow	repeated	5.98	6.46

Repeat passengers rate

```sql
SELECT
    c.city_name,
    SUM(f.repeat_passengers) AS total_repeat_passengers,
    SUM(f.total_passengers) AS total_passengers,
    ROUND((SUM(f.repeat_passengers) / SUM(f.total_passengers)) * 100, 2) AS repeat_passenger_rate
FROM
    fact_passenger_summary f
JOIN
    dim_city c
ON
    f.city_id = c.city_id
GROUP BY
    c.city_name
ORDER BY
    repeat_passenger_rate DESC;
```

city_name	total_repeat_passengers	total_passengers	repeat_passenger_rate
Surat	8638	20264	42.63
Lucknow	9597	25857	37.12
Indore	7216	22079	32.68
Vadodara	4346	14473	30.03
Visakhapatnam	5108	17855	28.61
Coimbatore	2551	11065	23.05
Kochi	7626	34042	22.40
Chandigarh	5070	23978	21.14
Jaipur	9682	55538	17.43
Mysore	1477	13158	11.23

Total trips by city

```sql
SELECT
    c.city_name, COUNT(t.trip_id) AS total_trips
FROM
    fact_trips t
        JOIN
    dim_city c ON c.city_id = t.city_id
GROUP BY city_name;
```

city_name	total_trips
Lucknow	13299
Vadodara	6662
Coimbatore	4428
Kochi	10670
Visakhapatnam	5967
Chandigarh	8016
Surat	11380
Jaipur	16133
Mysore	3477
Indore	9005

Repeat passengers frequency according to city (city wise contribution of repeat passengers)

```sql
SELECT c.city_name,
    SUM(CASE WHEN d.trip_count = '9-Trips' THEN d.repeat_passenger_count ELSE 0 END) AS `9-Trips`,
    SUM(CASE WHEN d.trip_count = '8-Trips' THEN d.repeat_passenger_count ELSE 0 END) AS `8-Trips`,
    SUM(CASE WHEN d.trip_count = '7-Trips' THEN d.repeat_passenger_count ELSE 0 END) AS `7-Trips`,
    SUM(CASE WHEN d.trip_count = '6-Trips' THEN d.repeat_passenger_count ELSE 0 END) AS `6-Trips`,
    SUM(CASE WHEN d.trip_count = '5-Trips' THEN d.repeat_passenger_count ELSE 0 END) AS `5-Trips`,
    SUM(CASE WHEN d.trip_count = '4-Trips' THEN d.repeat_passenger_count ELSE 0 END) AS `4-Trips`,
    SUM(CASE WHEN d.trip_count = '3-Trips' THEN d.repeat_passenger_count ELSE 0 END) AS `3-Trips`,
    SUM(CASE WHEN d.trip_count = '2-Trips' THEN d.repeat_passenger_count ELSE 0 END) AS `2-Trips`,
    SUM(CASE WHEN d.trip_count = '10-Trips' THEN d.repeat_passenger_count ELSE 0 END) AS `10-Trips`
FROM dim_repeat_trip_distribution d
    JOIN
    dim_city c
ON
    d.city_id = c.city_id
GROUP BY
    c.city_name
ORDER BY
    c.city_name;
```

output

city_name	9-Trips	8-Trips	7-Trips	6-Trips	5-Trips	4-Trips	3-Trips	2-Trips	10-Trips
Chandigarh	118	176	278	376	619	798	976	1638	91
Coimbatore	59	157	267	450	526	397	378	286	31
Indore	172	235	378	494	746	967	1637	2478	109
Jaipur	116	184	244	400	609	1173	2007	4855	94
Kochi	92	126	161	298	494	901	1857	3635	62
Lucknow	183	617	1087	1937	1768	1555	1417	927	106
Mysore	8	21	26	60	86	188	361	720	7
Surat	150	539	1027	1594	1706	1430	1232	843	117
Vadodara	89	251	559	829	785	718	616	429	70
Visakhapatnam	45	71	101	163	278	510	1275	2618	47

Repeat passengers frequency(in %) according to city (city wise contribution of repeat passengers)

```sql
SELECT
    c.city_name,
    ROUND( (SUM(CASE WHEN d.trip_count = '9-Trips' THEN d.repeat_passenger_count ELSE 0 END) /
           SUM(d.repeat_passenger_count)) * 100, 2) AS `9-Trips`,
    ROUND( (SUM(CASE WHEN d.trip_count = '8-Trips' THEN d.repeat_passenger_count ELSE 0 END) /
           SUM(d.repeat_passenger_count)) * 100, 2) AS `8-Trips`,
    ROUND( (SUM(CASE WHEN d.trip_count = '7-Trips' THEN d.repeat_passenger_count ELSE 0 END) /
           SUM(d.repeat_passenger_count)) * 100, 2) AS `7-Trips`,
    ROUND( (SUM(CASE WHEN d.trip_count = '6-Trips' THEN d.repeat_passenger_count ELSE 0 END) /
           SUM(d.repeat_passenger_count)) * 100, 2) AS `6-Trips`,
    ROUND( (SUM(CASE WHEN d.trip_count = '5-Trips' THEN d.repeat_passenger_count ELSE 0 END) /
           SUM(d.repeat_passenger_count)) * 100, 2) AS `5-Trips`,
    ROUND( (SUM(CASE WHEN d.trip_count = '4-Trips' THEN d.repeat_passenger_count ELSE 0 END) /
           SUM(d.repeat_passenger_count)) * 100, 2) AS `4-Trips`,
    ROUND( (SUM(CASE WHEN d.trip_count = '3-Trips' THEN d.repeat_passenger_count ELSE 0 END) /
           SUM(d.repeat_passenger_count)) * 100, 2) AS `3-Trips`,
    ROUND( (SUM(CASE WHEN d.trip_count = '2-Trips' THEN d.repeat_passenger_count ELSE 0 END) /
```

Repeat passengers frequency(in %) according to city (city wise contribution of repeat passengers)

```
17      ROUND( (SUM(CASE WHEN d.trip_count = '2-Trips' THEN d.repeat_passenger_count ELSE 0 END) /
18              SUM(d.repeat_passenger_count)) * 100, 2) AS `2-Trips`,
19      ROUND( (SUM(CASE WHEN d.trip_count = '10-Trips' THEN d.repeat_passenger_count ELSE 0 END) /
20              SUM(d.repeat_passenger_count)) * 100, 2) AS `10-Trips`
21  FROM
22      dim_repeat_trip_distribution d
23  JOIN
24      dim_city c
25  ON
26      d.city_id = c.city_id
27  GROUP BY
28      c.city_name
29  ORDER BY
30      c.city_name;
```

Repeat passengers frequency(in %) according to city (city wise contribution of repeat passengers)

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

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

Highest and lowest cities by new passengers

```
,
WITH city_passenger_summary AS (
    SELECT
        c.city_name,
        SUM(f.new_passengers) AS total_new_passengers
    FROM
        fact_passenger_summary f
    JOIN
        dim_city c
    ON
        c.city_id = f.city_id
    GROUP BY
        c.city_name
)
SELECT
    city_name,
    total_new_passengers,
    RANK() OVER (ORDER BY total_new_passengers DESC) AS rank_status,
    CASE
        WHEN RANK() OVER (ORDER BY total_new_passengers DESC) <= 3 THEN 'Top 3'
        WHEN RANK() OVER (ORDER BY total_new_passengers DESC) >= 8 THEN 'Bottom 3'
        ELSE 'Other'
    END AS city_category
FROM
    city_passenger_summary
ORDER BY
   rank_status;
```

Analysis:-Highest and lowest cities by new passengers

city_name	total_new_passengers	rank_status	city_category
Jaipur	45856	1	Top 3
Kochi	26416	2	Top 3
Chandigarh	18908	3	Top 3
Lucknow	16260	4	Other
Indore	14863	5	Other
Visakhapatnam	12747	6	Other
Mysore	11681	7	Other
Surat	11626	8	Bottom 3
Vadodara	10127	9	Bottom 3
Coimbatore	8514	10	Bottom 3

Target analysis between target new passengers vs. passengers

```sql
SELECT
c.city_name,
SUM(t.total_passengers) AS total_passengers,
SUM(t.new_passengers) AS new_passengers,
SUM(m.target_new_passengers) AS target_new_passengers,
ROUND(
    ((SUM(t.new_passengers) - SUM(m.target_new_passengers)) / SUM(m.target_new_passengers)) * 100,
) AS percentage_difference,
CASE
    WHEN SUM(t.new_passengers) >= SUM(m.target_new_passengers) THEN 'Target Achieved'
    ELSE 'Target Missed'
END AS target_status
FROM
    fact_passenger_summary t
JOIN
   monthly_target_new_passengers m
ON
    t.city_id = m.city_id
JOIN
    dim_city c
ON
    t.city_id = c.city_id
GROUP BY
    c.city_name;
```

Output:-

city_name	total_passengers	new_passengers	target_new_passengers	percentage_difference	target_status
Surat	121584	69756	63000	10.72	Target Achieved
Vadodara	86838	60762	59400	2.29	Target Achieved
Lucknow	155142	97560	93600	4.23	Target Achieved
Indore	132474	89178	84600	5.41	Target Achieved
Visakhapatnam	107130	76482	81000	-5.58	Target Missed
Chandigarh	143868	113448	126000	-9.96	Target Missed
Mysore	78948	70086	72000	-2.66	Target Missed
Coimbatore	66390	51084	45000	13.52	Target Achieved
Kochi	204252	158496	162000	-2.16	Target Missed
Jaipur	333228	275136	324000	-15.08	Target Missed

Target analysis between avg_rating by passengers and ratings by targeted passengers

```sql
SELECT
    c.city_name,
    round(AVG(f.passenger_rating),2) AS avg_passenger_rating,
    AVG(r.target_avg_passenger_rating) AS target_avg_passenger_rating,
    CASE
        WHEN AVG(f.passenger_rating) >= AVG(r.target_avg_passenger_rating) THEN 'Target Achieved'
        ELSE 'Target Missed'
    END AS target_status
FROM
    fact_trips f
JOIN
    city_target_passenger_rating r
ON
    f.city_id = r.city_id
JOIN
    dim_city c
ON
     f.city_id = r.city_id
 JOIN
     dim_city c
 ON
     c.city_id = r.city_id
 GROUP BY
     c.city_name;
```

Output:-

city_name	avg_passenger_rating	target_avg_passenger_rating	target_status
Lucknow	6.49	7.25	Target Missed
Vadodara	6.61	7.5	Target Missed
Coimbatore	7.87	8.25	Target Missed
Kochi	8.51	8.5	Target Achieved
Visakhapatnam	8.43	8.5	Target Missed
Chandigarh	7.99	8	Target Missed
Surat	6.43	7	Target Missed
Jaipur	8.58	8.25	Target Achieved
Mysore	8.70	8.5	Target Achieved
Indore	7.83	8	Target Missed

Average fare per km and average distance travelled by each city

```sql
SELECT
    c.city_name,
    round(AVG(f.distance_travelled),2) AS avg_distance_travelled,
    round(AVG(f.fare_amount / f.distance_travelled),2) AS avg_fare_per_km
FROM
    fact_trips f
JOIN
    dim_city c
ON
    f.city_id = c.city_id
WHERE
    f.distance_travelled > 0
GROUP BY
    c.city_name
ORDER BY
    c.city_name;
```

city_name	avg_distance_travelled	avg_fare_per_km
Chandigarh	23.55	12.12
Coimbatore	14.95	11.30
Indore	16.57	11.05
Jaipur	30.10	16.24
Kochi	24.04	14.16
Lucknow	12.52	12.13
Mysore	16.49	15.36
Surat	11.00	10.92
Vadodara	11.54	10.53
Visakhapatnam	22.58	12.72

Revenue analysis :- new vs. Repeated passengers

```sql
SELECT
    month_name,
SUM(CASE WHEN passenger_type = 'New' THEN fare_amount ELSE 0 END) AS new_passenger_revenue,
SUM(CASE WHEN passenger_type = 'Repeated' THEN fare_amount ELSE 0 END) AS repeated_passenger_revenue
FROM (
    SELECT
        f.fare_amount,
        f.passenger_type,
        MONTHNAME(STR_TO_DATE(f.date, '%d/%m/%Y')) AS month_name,
        DATE_FORMAT(STR_TO_DATE(f.date, '%d/%m/%Y'), '%Y-%m') AS month_order
    FROM fact_trips f
) AS subquery
GROUP BY
    month_name, month_order
ORDER BY
    month_order;
```

month_name	new_passenger_revenue	repeated_passenger_revenue
January	2317228	1511137
February	2353213	1821561
March	1881698	2034585
April	1686718	2061030
May	1519689	2292561
June	1450248	1739959

Revenue analysis:-weekdays vs. weekends revenue

```sql
SELECT
    c.city_name,
    SUM(CASE
        WHEN DAYOFWEEK(STR_TO_DATE(f.`DATE`, '%d/%m/%Y')) IN (1, 7) THEN fare_amount
        ELSE 0
    END) AS WeekendRevenue,
    SUM(CASE
        WHEN DAYOFWEEK(STR_TO_DATE(f.`DATE`, '%d/%m/%Y')) NOT IN (1, 7) THEN fare_amount
        ELSE 0
    END) AS WeekdayRevenue,
    CASE
        WHEN SUM(CASE
            WHEN DAYOFWEEK(STR_TO_DATE(f.`DATE`, '%d/%m/%Y')) IN (1, 7) THEN fare_amount
            ELSE 0
        END) > SUM(CASE
            WHEN DAYOFWEEK(STR_TO_DATE(f.`DATE`, '%d/%m/%Y')) NOT IN (1, 7) THEN fare_amount
            ELSE 0
         END) THEN 'Higher Revenue on Weekends'
         ELSE 'Higher Revenue on Weekdays'
     END AS RevenueComparison
FROM
    fact_trips f
JOIN
    dim_city c
ON
    f.city_id = c.city_id
GROUP BY
    c.city_name;
```

Output:-

city_name	WeekendRevenue	WeekdayRevenue	RevenueComparison
Lucknow	524782	1430921	Higher Revenue on Weekdays
Vadodara	323046	467798	Higher Revenue on Weekdays
Coimbatore	358640	378688	Higher Revenue on Weekdays
Kochi	2170195	1407954	Higher Revenue on Weekends
Visakhapatnam	963902	726732	Higher Revenue on Weekends
Chandigarh	1267435	999568	Higher Revenue on Weekends
Surat	503272	833495	Higher Revenue on Weekdays
Jaipur	5098306	2724045	Higher Revenue on Weekends
Mysore	588375	278685	Higher Revenue on Weekends
Indore	919771	704017	Higher Revenue on Weekends

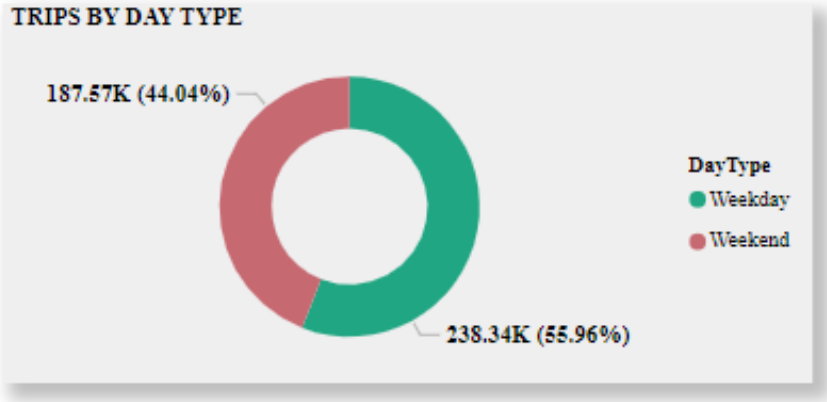
Highest revenue generated on weekend so we conclude that there is high demand on weekends for tourism activities and again cities like Jaipur,kochi in highest range.

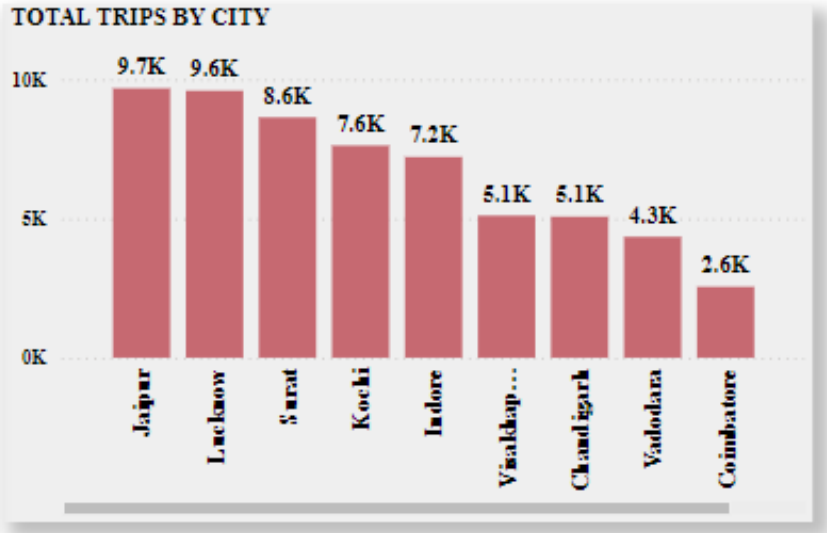
POWER BI DASHBOARD

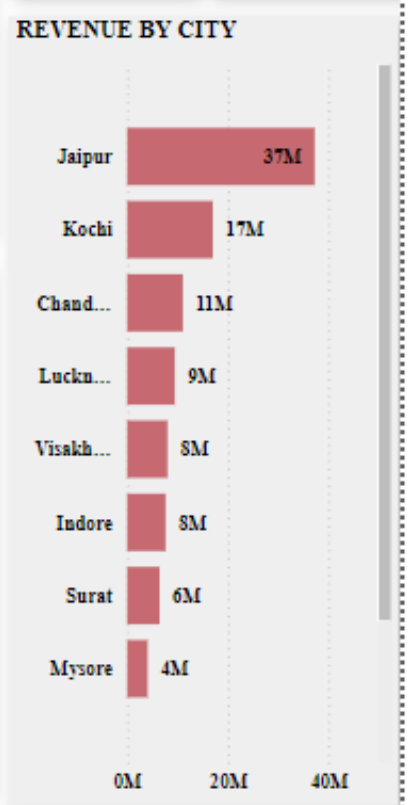
GOOD CABS ANALYSIS

108M	8M	426K	254.02	238K
TOTAL REVENUE	TOTAL DISTAN...	TOTAL TRIPS	AVG FARE	TOTAL PASSENG...

CITY	TOTAL TRIPS	TOTAL TARGET TRIPS	TARGET TRIPS STAT
Chandigarh	38981	39000	TARGET MISSED
Coimbatore	21104	21000	TARGET ACHIVE
Indore	42456	43500	TARGET MISSED
Jaipur	76888	67500	TARGET ACHIVE
Kochi	50702	49500	TARGET ACHIVE
Lucknow	64299	72000	TARGET MISSED
Mysore	16238	13500	TARGET ACHIVE
Surat	54843	57000	TARGET MISSED
Vadodara	32026	37500	TARGET MISSED
Visakhapatnam	28366	28500	TARGET MISSED







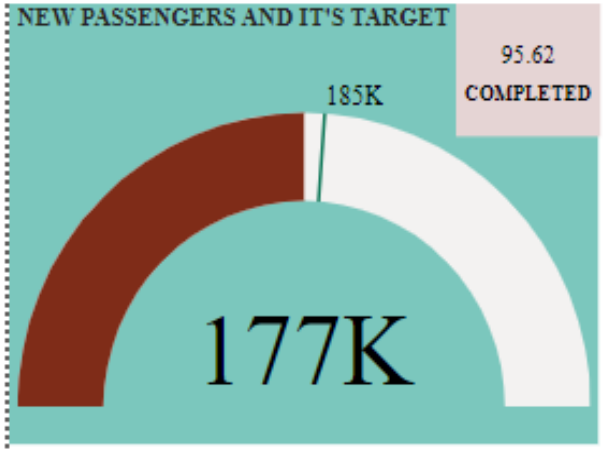
TRIPS ANALYSIS

8	61K	13.28	177K
AVERGAE RATINGS	REPEAT PASSENGERS	Average Fare Per KM	NEW PASSEGERS

MONTH: All

CITY: All

NEW PASSENGERS AND IT'S TARGET

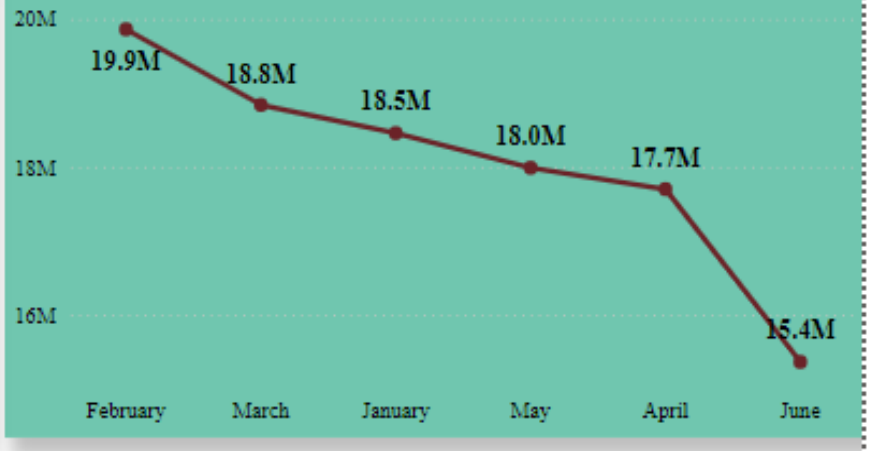


95.62
COMPLETED

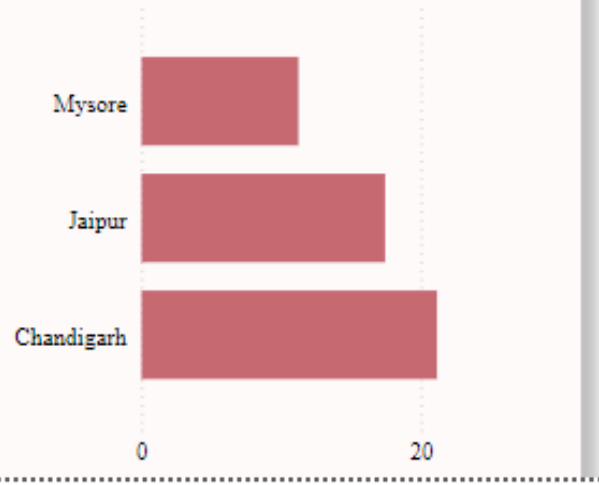
TRIPS BY DAY TYPE BY EACH CITY

CITY NAME	Weekday Trips	WeekEND Trips
Chandigarh	19914	19067
Coimbatore	12576	8528
Indore	21198	21258
Jaipur	32491	44397
Kochi	22915	27787
Lucknow	49617	14682
Mysore	6424	9814

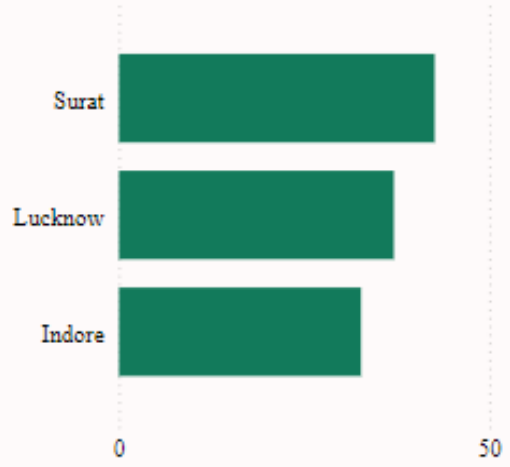
TOTAL REVENUE BY MONTH



RPR BY CITIES (BOTTOM 3)



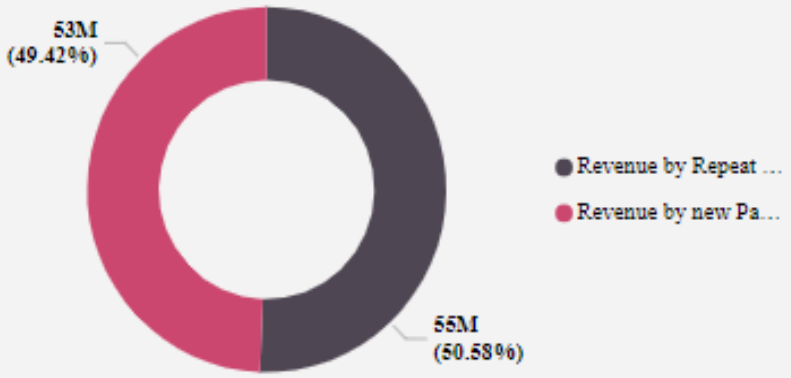
RPR BY CITIES (TOP 3)



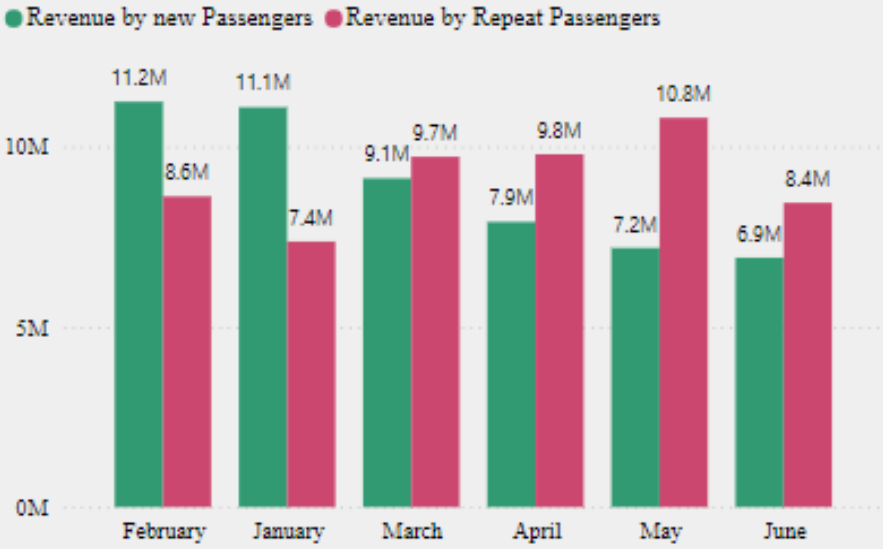
CITY NAME	AVERGAE PASSE_RATINGS	AVERAGE TARGET PASS_RATINGS	TARGET RATINGS STA
Surat	6	7.00	TARGET MISSED
Lucknow	6	7.25	TARGET MISSED
Vadodara	7	7.50	TARGET MISSED
Indore	8	8.00	TARGET MISSED
Coimbatore	8	8.25	TARGET MISSED
Chandigarh	8	8.00	TARGET MISSED
Visakhapatnam	8	8.50	TARGET MISSED
Kochi	9	8.50	TARGET ACHIVED

REVENUE ANALYSIS

REVENUE BY PASSENGERS TYPE



MONTHLY REVENUE COMPARISION BY PASSENGER TYPE



CITY NAME	NEW PASSENGERS	TARGET	TARGET STATU
Coimbatore	8514	7500	Target Achieved
Vadodara	10127	9900	Target Achieved
Surat	11626	10500	Target Achieved
Mysore	11681	12000	Target Missed
Visakhapatnam	12747	13500	Target Missed
Indore	14863	14100	Target Achieved
Lucknow	16260	15600	Target Achieved
Chandigarh	18908	21000	Target Missed
Kochi	26416	27000	Target Missed

REVENUE BY DAY TYPE

CITY NAME	Weekday	Weekend
Chandigarh	4732653	6325748
Coimbatore	1798952	1725040
Indore	3282698	4352530
Jaipur	12915283	24292214
Kochi	6784213	10213383
Lucknow	6879267	2584284
Mysore	1288320	2766425

CITY NAME	AVERAGE DISTANCE TRAVELLED
Chandigarh	23.52
Coimbatore	14.98
Indore	16.50
Jaipur	30.02
Kochi	24.07
Lucknow	12.51
Mysore	16.50
Surat	11.00
Vadodara	11.52

REPEAT PASSENGERS FREQUENCY ANALYSIS

city_name	10-Trips	2-Trips	3-Trips	4-Trips	5-Trips	6-Trips	7-Trips	8-Trips	9-Trips
Chandigarh	91	1638	976	798	619	376	278	176	118
Coimbatore	31	286	378	397	526	450	267	157	59
Indore	109	2478	1637	967	746	494	378	235	172
Jaipur	94	4855	2007	1173	609	400	244	184	116
Kochi	62	3635	1857	901	494	298	161	126	92
Lucknow	106	927	1417	1555	1768	1937	1087	617	183
Mysore	7	720	361	188	86	60	26	21	8
Surat	117	843	1232	1430	1706	1594	1027	539	150
Vadodara	70	429	616	718	785	829	559	251	89
Visakhapatnam	47	2618	1275	510	278	163	101	71	45

INSIGHTS

- Revenue:-Jaipur is highest revenue generated city.
- Trip volume:-Jaipur, lucknow and surat has highest trip volume.
- Passengers ratings:-Mysore, Jaipur,kochi has highest passengers rating(passengers satisfaction is good here)
- Average fare per km:- surat , Vadodara and Indore has lowest average fare amount so need to check and allocate the profitable amount as Surat is industrial city.
- New passengers target:-Jaipur (-15.08),Chandigarh(-9.96), Vishakhapatnam(-5.58) here target missed.so need to focus on this as Jaipur is the most revenue generated city.

INSIGHTS

- New passengers rate in Jaipur is high as compare to repeat passengers, though the Jaipur is lagging for targeted new passengers but new passengers rate is high as compare to repeat may be this is because Jaipur is tourist place.
- Kochi and Chandigarh also high rate of new passengers.
- Surat, Vadodara and Coimbatore has minimum rate of new passengers(bottom 3 cities)
- Cities like Jaipur, kochi and Vishakhapatnam passengers taken 2-trips in maximum quantity.

Insights

- To enhance awareness and increase trip volumes, we can implement localized marketing campaigns and targeted promotions. Cities such as Vishakhapatnam, Coimbatore, and Mysore present significant opportunities for growth and can benefit from tailored offers and discounts designed to attract new riders and drive engagement.

Further analysis and recommendations

What factors (such as quality of service, competitive pricing, or city demographics) might contribute to higher or lower repeat passenger rates in different cities? Are there correlations with socioeconomic or lifestyle patterns in these cities?

Repeat passengers rate may be depend on city as well as good service or may be low fare amount per trip, according to analysis surat, lucknow and Indore has high repeat passengers rate, this is may be due to the cities are industrial. So the daily travelers are more as compare to tourist cities. May be the fare amount is less for surat, lucknow and low fare amount cities as compared to other cities

Further analysis and recommendations

How do tourism seasons or local events (festivals, conferences) impact Goodcabs's demand patterns? Would tailoring marketing efforts to these events increase trip volume in tourism-oriented cities?

Based on the total trip counts for each city, Jaipur, Lucknow, and Surat have the highest number of trips. This suggests that these cities are popular tourist destinations, and as a result, we can infer that the demand for Good Cabs is higher in these cities.

Further analysis and recommendations

What emerging mobility trends (such as electric vehicle adoption, green energy use) are impacting the cab service market in tier-2 cities? Should Goodcabs consider integrating electric vehicles or eco-friendly initiatives to stay competitive?

Good cabs should indeed consider integrating electric vehicles and other eco-friendly initiatives into their operations. By doing so, the company can capitalize on the growing consumer demand for sustainable transportation options, It can attract environmentally conscious passengers.

Further analysis and recommendations

Are there opportunities for Goodcabs to partner with local businesses (such as hotels, malls, or event venues) to boost demand and improve customer loyalty? Could these partnerships drive more traffic, especially in tourism-heavy or high-footfall areas?

It will be the great opportunity for goodcabs to tie-up with local hotels and malls to drive more traffic,goodcab can offer ride packages at the time of hotel booking, it may helpful for tourists as well. Goodcabs provide shuttle services for events and conferences.

Further analysis and recommendations

To make goodcabs more data-driven and improve its performance across key metrics (such as repeat passenger rate, customer satisfaction, new passengers and trip volume), what additional data should Goodcabs collect? Consider data that could provide deeper insights into customer behavior, operational efficiency, and market trends.

Goodcabs should analyze the trends and demand areas like airports, tourist spots, should analyze customer’s ride history like there frequent routes or preference Competitor pricing and offers.


THANK
YOU!