Project Title: *Cab Booking System – Data Analysis*

Name: Praneeth

Course: SQL

Introduction

A Cab Booking System is an application that allows customers to book rides, track drivers, and make payments seamlessly. It connects customers with available drivers, making transportation faster and more convenient.

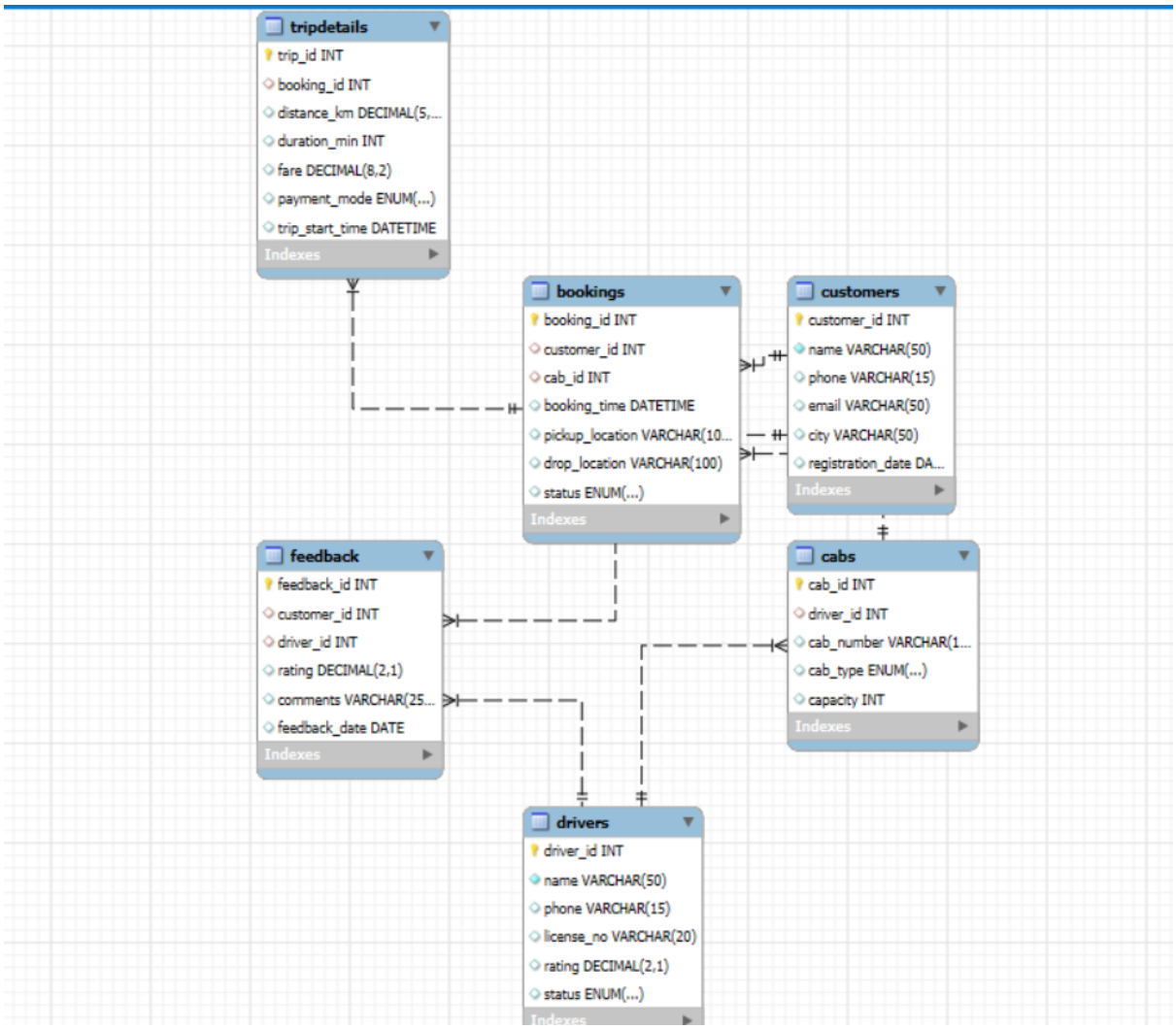
Data analysis plays a crucial role in improving the efficiency of such a system. By analyzing customer booking patterns, cancellation trends, and driver performance, companies can enhance customer satisfaction, optimize revenue generation, and improve driver utilization. This helps the business make better decisions, such as adjusting pricing during peak hours, identifying top customers, and rewarding high-performing drivers.

In this project, we designed a relational database schema for the cab booking system, inserted sample data, and wrote SQL queries to analyze important trends such as customer behavior, booking frequency, cancellations, and driver ratings. These insights can help improve operations and provide a better overall experience for both customers and drivers.

Objective:

A cab service company wants to enhance its operations by analyzing customer bookings, driver performance, and trip details. The company aims to: - Monitor ongoing and completed bookings. - Track customer preferences and behavior. - Evaluate driver efficiency and performance. - Analyze revenue trends based on fares and trip details. - Identify operational bottlenecks and areas for improvement. By structuring and querying the database, stakeholders can make data-driven decisions to improve the efficiency of the cab booking system.

Database Schema



Cab Booking System for Data Analysis

Problem Statement:

Customer and Booking Analysis

1.Identify customers who have completed the most bookings. What insights can you draw about their behavior?

```
SELECT
   c.customer_id,
   c.name,
   COUNT(b.booking_id) AS completed_bookings
FROM Customers c
JOIN Bookings b ON c.customer_id = b.customer_id
WHERE b.status = 'Completed'
GROUP BY c.customer_id, c.name
ORDER BY completed_bookings DESC;
```

customer_id	name	completed_bookings
9	Rohit Yadav	2
18	Manish Gupta	2
28	Snehal Pawar	2
6	Neha Patel	2
25	Bhavna Desai	2
3	Amit Verma	1
4	Sneha Reddy	1

2. Find customers who have canceled more than 30% of their total bookings. What could be the reason for frequent cancellations?

```
SELECT
   customer_id,
   COUNT(*) AS total_bookings,
```

```
  SUM(CASE WHEN status = 'Cancelled' THEN 1 ELSE 0 END) AS cancelled_bookings,
  (SUM(CASE WHEN status = 'Cancelled' THEN 1 ELSE 0 END) / COUNT(*)) * 100 AS cancellation_rate
FROM Bookings
GROUP BY customer_id
HAVING cancellation_rate > 30
ORDER BY cancellation_rate DESC;
```

customer_id	total_bookings	cancelled_bookings	cancellation_rate
2	1	1	100.0000
11	2	1	100.0000
37	1	1	100.0000
47	1	1	100.0000
7	2	1	50.0000
8	2	1	50.0000
20	2	1	50.0000

3. Determine the busiest day of the week for bookings. How can the company optimize cab availability on peak days?

```
SELECT
  DAYNAME(booking_time) AS day_of_week,
  COUNT(*) AS total_bookings
FROM Bookings
GROUP BY day_of_week
ORDER BY total_bookings DESC
limit 1;
```

day_of_week	total_bookings
Tuesday	21

Driver Performance & Efficiency

1.Identify drivers who have received an average rating below 3.0 in the past three months. What strategies can be implemented to improve their performance?

```
SELECT
  d.driver_id,
  d.name AS driver_name,
  ROUND(AVG(f.rating),2) AS avg_rating
FROM Drivers d
JOIN Feedback f ON d.driver_id = f.driver_id
WHERE f.feedback_date >= DATE_SUB(CURDATE(), INTERVAL 3 MONTH)
GROUP BY d.driver_id, d.name
HAVING avg_rating < 3.0
ORDER BY avg_rating ASC;
```

driver_id	driver_name	avg_rating	total_feedbacks
8	Praveen Yadav	3.70	3
9	Arvind Joshi	3.73	3
2	Suresh Mehta	3.75	2
5	Dinesh Shetty	3.80	3.80
15	Nitin Jain	3.80	2
25	Farhan Khan	3.85	2
22	Anil Saini	3.95	2
19	Suraj Malhotra	4.00	2
26	Mithun Joshi	4.05	2
29	Vishal Gupta	4.05	2

2. Find the top 5 drivers who have completed the longest trips in terms of distance. What does this say about their working patterns?

```
SELECT
   d.driver_id,
   d.name AS driver_name,
   SUM(t.distance_km) AS total_distance,
   COUNT(t.trip_id) AS total_trips
FROM Drivers d
JOIN Bookings b ON d.driver_id = b.cab_id   -- if cab is linked to driver
JOIN TripDetails t ON b.booking_id = t.booking_id
WHERE b.status = 'Completed'
GROUP BY d.driver_id, d.name
ORDER BY total_distance DESC
LIMIT 5;
```

driver_id	driver_name	total_distance	total_trips
11	Mahesh Choudhary	80.00	4
13	Harish Pandey	74.20	4
18	Siddharth Mehta	67.80	4
15	Nitin Jain	61.10	4
19	Suraj Malhotra	51.00	2

3. Identify drivers with a high percentage of canceled trips. Could this indicate driver unreliability?

```
SELECT
   d.driver_id,
   d.name AS driver_name,
   COUNT(CASE WHEN b.status = 'Cancelled' THEN 1 END) AS cancelled_trips,
   COUNT(*) AS total_trips,
```

```
  ROUND((COUNT(CASE WHEN b.status = 'Cancelled' THEN 1 END) *
100.0 / COUNT(*)), 2) AS cancel_percentage
FROM Drivers d
JOIN Cabs c ON d.driver_id = c.driver_id
JOIN Bookings b ON c.cab_id = b.cab_id
GROUP BY d.driver_id, d.name
HAVING cancel_percentage > 30
ORDER BY cancel_percentage DESC;
```

driver_id	driver_name	cancelled_trips	total_trips	cancel_percentage
2	Suresh Mehta	1	1	100.00
7	Imran Khan	1	1	100.00
14	Gopal Reddy	3	5	60.00
17	Mohan Das	2	5	40.00

Revenue & Business Metrics

1.Calculate the total revenue generated by completed bookings in the last 6 months. How has the revenue trend changed over time?

2. Identify the top 3 most frequently traveled routes based on PickupLocation and DropoffLocation. Should the company allocate more cabs to these routes?

```
SELECT
  b.pickup_location,
  b.drop_location,
  COUNT(*) AS trip_count
FROM Bookings b
WHERE b.status = 'Completed'
```

```
GROUP BY b.pickup_location, b.drop_location
ORDER BY trip_count DESC
LIMIT 3;
```

Result Grid | Filter Rows:

pickup_location	drop_location	trip_count
MG Road	Airport	3
HSR Layout	Airport	2
BTM Layout	Koramangala	2

- **Yes, the company should allocate more cabs** to these high-demand routes.

It might also:

- Introduce **shared ride options** for busy routes.
- Offer **discounts or surge pricing** depending on demand.
- Improve **driver availability** during peak times on these routes.

3. Determine if higher-rated drivers tend to complete more trips and earn higher fares. Is there a direct correlation between driver ratings and earnings?

```
SELECT
  d.driver_id,
  d.name AS driver_name,
  ROUND(AVG(f.rating), 2) AS avg_rating,
  COUNT(t.trip_id) AS total_trips,
  SUM(t.fare) AS total_earnings
FROM Drivers d
```

```
JOIN Cabs c ON d.driver_id = c.driver_id
JOIN Bookings b ON c.cab_id = b.cab_id
JOIN TripDetails t ON b.booking_id = t.booking_id
LEFT JOIN Feedback f ON d.driver_id = f.driver_id
WHERE b.status = 'Completed'
GROUP BY d.driver_id, d.name
ORDER BY avg_rating DESC;
```

driver_id	driver_name	avg_rating	total_trips	total_earnings
13	Harish Pandey	4.87	12	4290.00
10	Kishore Naik	4.83	3	450.00
1	Ramesh Kumar	4.60	2	1100.00
20	Raj Malviya	4.35	8	1400.00
16	Arjun Iyer	4.35	4	1120.00
17	Mohan Das	4.35	4	800.00
6	Rahul Patil	4.20	3	960.00
11	Mahesh Choudhary	4.20	12	4470.00
14	Gopal Reddy	Mahesh Choudhary		1890.00
3	Ajay Rao	4.07	3	540.00
19	Suraj Malhotra	4.00	4	1780.00
15	Nitin Jain	3.80	8	2380.00
9	Arvind Joshi	3.73	3	1350.00

- If top-rated drivers have **higher trip counts & earnings**, there may be a **positive correlation**:
- They likely get **repeat customers**.
- They might receive **priority bookings** due to better ratings.
- If there is **no clear correlation**, you might need to:
- Investigate if good drivers are underutilized.
- Adjust cab assignment algorithm to favor high-rated drivers.

Operational Efficiency & Optimization

1.Analyze the average waiting time (difference between booking time and trip start time) for different pickup locations. How can this be optimized to reduce delays?

```
SELECT
   b.pickup_location,
   ROUND(AVG(TIMESTAMPDIFF(MINUTE, b.booking_time, t.trip_start_time)), 2) AS avg_waiting_time_minutes,
   COUNT(*) AS total_trips
FROM Bookings b
JOIN TripDetails t ON b.booking_id = t.booking_id
WHERE b.status = 'Completed'
GROUP BY b.pickup_location
ORDER BY avg_waiting_time_minutes DESC;
```

pickup_location	avg_waiting_time_minutes	total_trips
KR Puram	17.00	1
HSR Layout	14.00	4
Whitefield	14.00	2
Marathahalli	11.60	5
Yeshwanthpur	10.50	2
Majestic	10.00	1
Banashankari	10.00	1
Rajajinagar	9.50	2
Indiranagar	8.80	5
MG Road	8.33	3

2.Identify the most common reasons for trip cancellations from customer feedback. What actions can be taken to reduce cancellations?

```
SELECT
   f.comments AS cancellation_reason,
```

```
  COUNT(*) AS total_cancellations,
  ROUND(COUNT(*) * 100.0 /
    (SELECT COUNT(*)
     FROM Feedback fb
     JOIN Bookings b2 ON fb.customer_id = b2.customer_id
     WHERE b2.status = 'Cancelled'), 2) AS percentage
FROM Feedback f
JOIN Bookings b ON f.customer_id = b.customer_id
WHERE b.status = 'Cancelled'
GROUP BY f.comments
ORDER BY total_cancellations DESC
LIMIT 5;
```

cancellation_reason	total_cancellations	percentage
Driver was late, but trip was okay	1	5.26
Driver was polite	1	5.26
Trip slightly delayed	1	5.26
Driver was okay	1	5.26
Quick trip, satisfied	1	5.26

3.Find out whether shorter trips (low-distance) contribute significantly to revenue. Should the company encourage more short-distance rides?

```
SELECT
  CASE
    WHEN t.distance_km < 5 THEN 'Short Trip (<5 km)'
    ELSE 'Long Trip (>=5 km)'
  END AS trip_category,
  COUNT(*) AS total_trips,
```

```
  SUM(t.fare) AS total_revenue,
  ROUND(SUM(t.fare) * 100.0 / (SELECT SUM(fare) FROM TripDetails), 2)
AS revenue_percentage
FROM TripDetails t
JOIN Bookings b ON t.booking_id = b.booking_id
WHERE b.status = 'Completed'
GROUP BY trip_category;
```

trip_category	total_trips	total_revenue	revenue_percentage
Long Trip (>=5 km)	38	11880.00	77.19

Comparative & Predictive Analysis

1.Compare the revenue generated from 'Sedan' and 'SUV' cabs. Should the company invest more in a particular vehicle type?

```
SELECT
  c.cab_type,
  ROUND(SUM(t.fare), 2) AS total_revenue,
  COUNT(*) AS total_trips,
  ROUND(AVG(t.fare), 2) AS avg_fare_per_trip
FROM Cabs c
JOIN Bookings b ON c.cab_id = b.cab_id
JOIN TripDetails t ON b.booking_id = t.booking_id
WHERE b.status = 'Completed'
GROUP BY c.cab_type
ORDER BY total_revenue DESC;
```

cab_type	total_revenue	total_trips	avg_fare_per_trip
Sedan	4640.00	13	356.92
SUV	3820.00	12	318.33
Mini	3420.00	13	263.08

2.Predict which customers are likely to stop using the service based on their last booking date and frequency of rides. How can customer retention be improved?

```
SELECT
   c.customer_id,
   c.name AS customer_name,
   COUNT(b.booking_id) AS total_bookings,
   MAX(b.booking_time) AS last_booking_date,
   DATEDIFF(CURDATE(), MAX(b.booking_time)) AS
days_since_last_booking
FROM Customers c
LEFT JOIN Bookings b ON c.customer_id = b.customer_id
GROUP BY c.customer_id, c.name
HAVING days_since_last_booking > 30  -- no trips in the last 30 days
  OR total_bookings < 3          -- very few total rides
ORDER BY days_since_last_booking DESC;
```

customer_id	customer_name	total_bookings	last_booking_date	days_since_last_booking
12	Anjali Das	1	2024-06-06 09:00:00	467
15	Ritika Sharma	1	2024-06-06 18:45:00	467
11	Varun Nair	1	2024-07-03 08:45:00	440
13	Naveen Kumar	1	2024-07-03 12:00:00	440
14	Meera Joshi	1	2024-07-03 14:15:00	440
16	Aditya Rao	1	2024-07-04 07:00:00	439
18	Manish Gupta	2	2024-07-04 08:50:00	439
20	Rajeev Sinha	2	2024-07-04 15:30:00	439
21	Aakash Jain	1	2024-07-04 17:45:00	439
22	Divya Rani	2	2024-08-05 06:30:00	407

3.Analyze whether weekend bookings differ significantly from weekday bookings.Should the company introduce dynamic pricing based on demand?

```
SELECT
  CASE
    WHEN DAYOFWEEK(b.booking_time) IN (1,7) THEN 'Weekend'  -- 1=Sunday, 7=Saturday
    ELSE 'Weekday'
  END AS day_type,
  COUNT(b.booking_id) AS total_bookings,
  ROUND(SUM(t.fare), 2) AS total_revenue,
  ROUND(AVG(t.fare), 2) AS avg_fare_per_trip
FROM Bookings b
JOIN TripDetails t ON b.booking_id = t.booking_id
WHERE b.status = 'Completed'
GROUP BY day_type
ORDER BY total_bookings DESC;
```

day_type	total_bookings	total_revenue	avg_fare_per_trip
Weekday	31	9590.00	309.35
Weekend	7	2290.00	327.14