OLA Data Analyst Project

ChatGPT Prompt to Create Data

Please create a spreadsheet with 1 lac rows, for Bengaluru city.

Give the following columns.

1. Date
2. Time
3. Booking ID
4. Booking Status
5. Customer ID
6. Vehicle Type
   - Auto
   - Prime Plus
   - Prime Sedan
   - Mini
   - Bike
   - e Bike
   - Prime SUV
7. Pickup Location (Create dummy location points Take any 50 areas from Bangalore)
8. Drop Location (Take from dummy pickup locations)
9. Average V_TAT (Time taken to arrive at the vehicle)
10. Average C_TAT (Time taken to arrive the Customer)
11. Cancelled Rides by Customer
12. Reason for cancelling by Customer
    - Driver is not moving towards pickup location
    - Driver asked to cancel
    - AC is not working (Only for 4-wheelers)
    - Change of plans
    - Wrong Address
13. Cancelled Rides by Driver
    - Personal & Car related issues
    - Customer related issue
    - The customer was coughing/sick
    - More than permitted people in there
14. Incomplete Rides
15. Incomplete Rides Reason
    - Customer Demand

- Vehicle Breakdown
- Other Issue

16. Booking Value
17. Ride Distance
18. Driver Ratings
19. Customer Rating

SQL Questions

1. Retrieve all successful bookings:

2. Find the average ride distance for each vehicle type:

3. Get the total number of cancelled rides by customers:

4. List the top 5 customers who booked the highest number of rides:

5. Get the number of rides cancelled by drivers due to personal and car-related issues:

6. Find the maximum and minimum driver ratings for Prime Sedan bookings:

7. Retrieve all rides where payment was made using UPI:

8. Find the average customer rating per vehicle type:

9. Calculate the total booking value of rides completed successfully:

10. List all incomplete rides along with the reason:

Data Columns

1. Date
2. Time
3. Booking_ID
4. Booking_Status
5. Customer_ID
6. Vehicle_Type
7. Pickup_Location
8. Drop_Location
9. V_TAT
10. C_TAT
11. Cancelled_Rides_by_Customer
12. Cancelled_Rides_by_Driver
13. Incomplete_Rides
14. Incomplete_Rides_Reason
15. Booking_Value
16. Payment_Method
17. Ride_Distance
18. Driver_Ratings
19. Customer_Rating

SQL Question & Answer

Create database Ola;

use Ola;

1. Retrieve all successful bookings:

Create View Successful_Bookings As

SELECT * FROM bookings WHERE Booking_Status = 'Success';

View of Retrieve all successful bookings:

SELECT * FROM Successful_Bookings;

2. Find the average ride distance for each vehicle type:

Create View avg_ride_distance_for_each_vehicle As

SELECT Vehicle_Type, AVG(Ride_Distance) as avg_distance FROM bookings GROUP BY Vehicle_Type;

View of Find the average ride distance for each vehicle type:

SELECT * FROM avg_ride_distance_for_each_vehicle;

3. Get the total number of cancelled rides by customers:

Create View cancelled_rides_by_customers As

SELECT COUNT(*) FROM bookings WHERE Booking_Status = 'cancelled by Customer';

View of Get the total number of cancelled rides by customers:

SELECT * FROM cancelled_rides_by_customers;

4. List the top 5 customers who booked the highest number of rides:

Create View top_5_customers As

```
SELECT Customer_ID, COUNT(Booking_ID) as total_rides FROM
bookings GROUP BY Customer_ID ORDER BY total_rides DESC LIMIT
5;
```

View of List the top 5 customers who booked the highest number of rides:

```
SELECT * FROM top_5_customers;
```

5. Get the number of rides cancelled by drivers due to personal and car-related issues:

```
Create View rides_cancelled_by_drivers_P_and_C_related_issues As

 SELECT COUNT(*) FROM bookings WHERE
canceled_Rides_by_Driver = 'Personal & Car related issue';
```

View of Get the number of rides cancelled by drivers due to personal and car-related issues:

```
SELECT * FROM rides_cancelled_by_drivers_P_and_C_related_issues;
```

6. Find the maximum and minimum driver ratings for Prime Sedan bookings:

```
Create View max_min_driver_rating As

SELECT MAX(Driver_Ratings) as max_rating, MIN(Driver_Ratings) as
min_rating FROM bookings WHERE Vehicle_Type = 'Prime Sedan';
```

View of Find the maximum and minimum driver ratings for Prime Sedan bookings:

```
SELECT * FROM max_min_driver_rating;
```

7. Retrieve all rides where payment was made using UPI:

```
Create View UPI_payment As

SELECT * FROM bookings WHERE Payment_Method = 'UPI';
```

View of Retrieve all rides where payment was made using UPI:

```
SELECT * FROM UPI_payment;
```

8. Find the average customer rating per vehicle type:

```
Create View average_customer_rating_per_vehicle As

SELECT Vehicle_Type, AVG(Customer_Rating) as
avg_customer_rating FROM bookings GROUP BY Vehicle_Type;
```

View of Find the average customer rating per vehicle type:

```
SELECT * FROM average_customer_rating_per_vehicle;
```

9. Calculate the total booking value of rides completed successfully:

```
Create View total_booking_value_of_rides_completed_successfully As

SELECT SUM(Booking_Value) as totalsuccessful__value FROM
bookings WHERE Booking_Status = 'Success';
```

View of Calculate the total booking value of rides completed successfully:

```
SELECT * FROM
total_booking_value_of_rides_completed_successfully;
```

10. List all incomplete rides along with the reason:

Create View incomplete_rides_along_with_the_reason As

SELECT Booking_ID, Incomplete_Rides_Reason FROM bookings
WHERE Incomplete_Rides='Yes';

View of List all incomplete rides along with the reason:

SELECT * FROM incomplete_rides_along_with_the_reason;