SQL Project

Presented by RITLESH KUMAR

HELLO !

- My name is Ritlesh Kumar, and this project is based on a real-world OLA ride booking dataset containing 70,000 records.
- Using SQL queries, I explored booking performance, cancellation reasons, customer activity, vehicle-wise metrics, payment trends, and driver ratings to solve practical business problems and generate meaningful insights.
-

DATA INFO

This project uses a relational OLA ride booking database consisting of multiple interconnected tables, including Bookings, Customers, Drivers, Vehicles, Payments, and Ratings.

The dataset contains 70,000 ride records and captures key details such as booking date and time, ride status, pickup and drop locations, vehicle type, payment method, ride distance, fare amount, and customer & driver ratings.

It is designed to enable efficient analysis of ride success and cancellations, customer behavior, vehicle-wise performance, payment trends, and overall business performance using SQL.

OLA

QUESTION

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:

OLA

1.Retrieve all successful bookings

```sql
Create view Successfull_Bookings As
SELECT * FROM bookings
WHERE Booking_status = 'Success';

select * from Successfull_Bookings;
```

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

Date	Time	Booking_ID	Booking_Status	Customer_ID	Vehicle_Type	Pickup_Location	Drop_Location	V_TAT	C_TAT	Canceled_Rides_by_Customer	Canceled_Rides_by_Driver	Incomplete_Rides	Incomplete_
25-07-2024 22:20	22:20:00	CNR2940424040	Success	CID225428	Bike	Magadi Road	Varthur	203	30	NULL	NULL	No	NULL
30-07-2024 19:59	19:59:00	CNR2982357879	Success	CID270156	Prime SUV	Sahakar Nagar	Varthur	238	130	NULL	NULL	No	NULL
02-07-2024 09:02	09:02:00	CNR1797421769	Success	CID939555	Mini	Rajajinagar	Chamarajpet	252	80	NULL	NULL	No	NULL
13-07-2024 04:42	04:42:00	CNR8787177882	Success	CID802429	Mini	Kadugodi	Vijayanagar	231	90	NULL	NULL	No	NULL

2.Find the average ride distance for each vehicle type

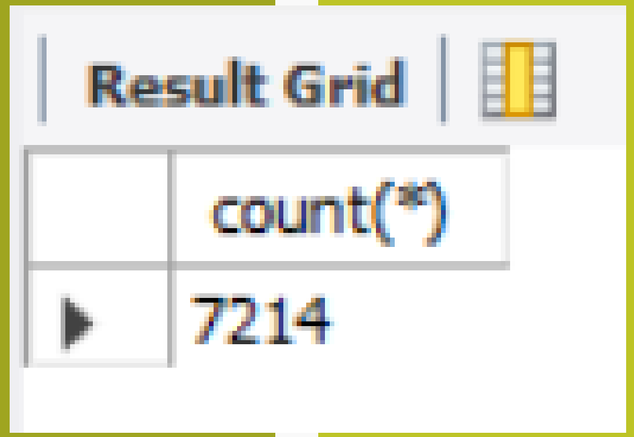
```sql
create view ride_distance_for_each_vehicle As
SELECT Vehicle_Type, AVG(Ride_Distance)
as avg_distance FROM bookings
GROUP BY Vehicle_Type;

select * from ride_distance_for_each_vehicle;
```

Result Grid | Filter Rows:

Vehicle_Type	avg_distance
Prime Sedan	15.7127
Bike	15.7467
Prime SUV	15.2022
eBike	15.6303
Mini	15.5738
Prime Plus	15.3720
Auto	6.2100

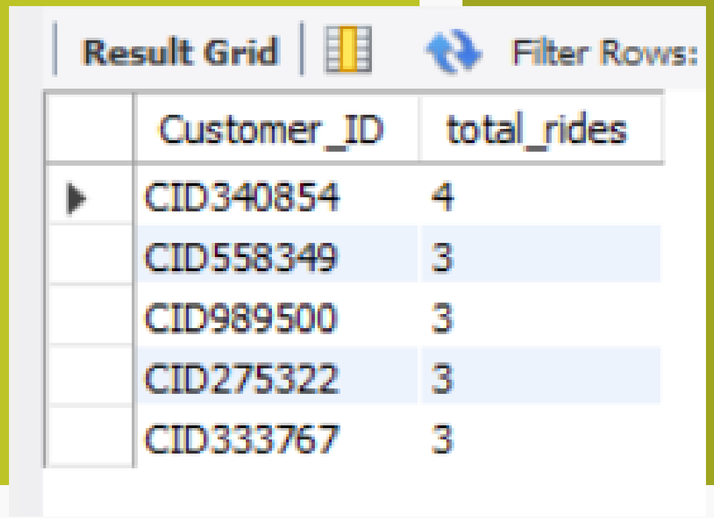
3.Get the total number of cancelled rides by customers

```sql
create view cancelled_rides_by_customers As
SELECT count(*) FROM bookings
WHERE Booking_Status = 'canceled by customer';

select * from cancelled_rides_by_customers;
```



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

```sql
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;

SELECT * FROM Top_5_Customers;
```



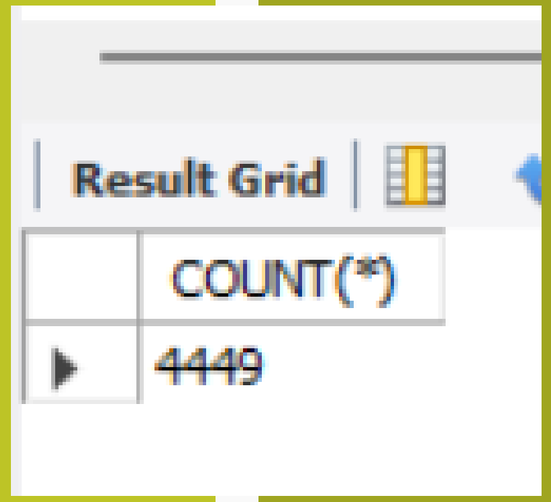
Result Grid | Filter Rows:

Customer_ID	total_rides
CID340854	4
CID558349	3
CID989500	3
CID275322	3
CID333767	3

OLA

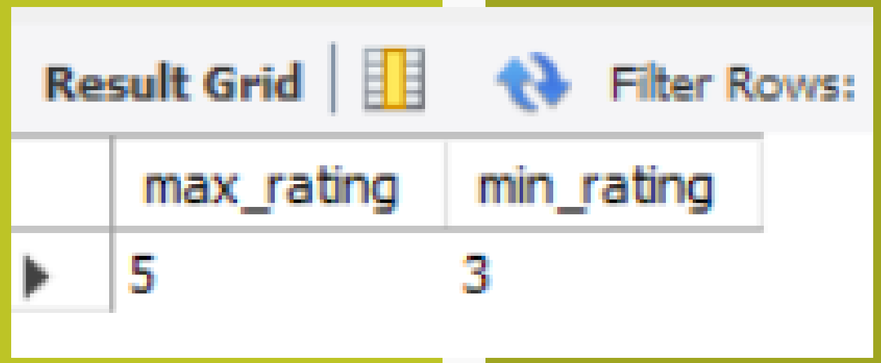
5. Get the number of rides cancelled by drivers due to personal and car-related issues

```sql
create view Ride_Canceled_by_Drivers_P_C_Issues AS
SELECT COUNT(*) FROM bookings
WHERE Canceled_Rides_by_Driver = 'Personal & Car related issue';

SELECT * FROM Ride_Canceled_by_Drivers_P_C_Issues;
```



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

```sql
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';

SELECT * FROM Max_Min_Driver_Rating;
```



	max_rating	min_rating
▶	5	3

7. Retrieve all rides where payment was made using UPI

```sql
#7. Retrieve all rides where payment was made using UPI:
CREATE VIEW UPI_Payment As
SELECT * FROM bookings
WHERE Payment_Method = 'UPI';

SELECT * FROM UPI_Payment;
```

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

Booking_Status	Customer_ID	Vehicle_Type	Pickup_Location	Drop_Location	V_TAT	C_TAT	Canceled_Rides_by_Customer	Canceled_Rides_by_Driver	Incomplete_Rides	Incomplete_Rides_Reason	Booking_Value	Payment_Method
Success	CID270156	Prime SUV	Sahakar Nagar	Varthur	238	130	NULL	NULL	No	NULL	386	UPI
Success	CID802429	Mini	Kadugodi	Vijayanagar	231	90	NULL	NULL	No	NULL	173	UPI
Success	CID540929	Auto	Cox Town	Yelahanka	126	35	NULL	NULL	No	NULL	330	UPI
Success	CID167642	Bike	Indiranagar	MG Road	70	95	NULL	NULL	No	NULL	378	UPI
Success	CID640151	Bike	Magadi Road	HSR Layout	126	95	NULL	NULL	No	NULL	343	UPI

8. Find the average customer rating per vehicle type

```sql
CREATE VIEW AVG_Cust_Rating As
SELECT Vehicle_Type, AVG(Customer_Rating) as avg_customer_rating
FROM bookings
GROUP BY Vehicle_Type;

SELECT * FROM AVG_Cust_Rating;
```

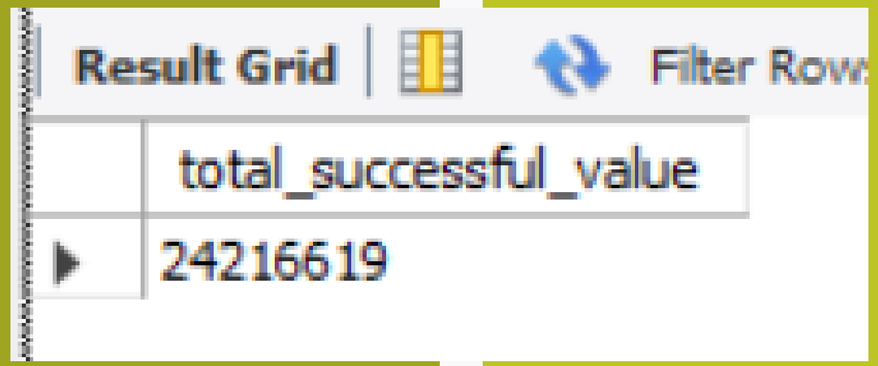
Result Grid | Filter Rows:

Vehicle_Type	avg_customer_rating
Prime Sedan	4.000231374363722
Bike	3.991264260040628
Prime SUV	3.9975113122171915
eBike	3.9887519500780066
Mini	3.997006083893701
Prime Plus	4.007608695652183
Auto	3.9978692955954718

OLA

9. Calculate the total booking value of rides completed successfully

```sql
CREATE VIEW total_successful_ride_value As
SELECT SUM(Booking_Value) as total_successful_value
FROM bookings
WHERE Booking_status = 'Success';

 SELECT * FROM total_successful_ride_value;
```



10. List all incomplete rides along with the reason

```sql
CREATE VIEW Incomplete_Rides_Reason As
 SELECT Booking_ID, Incomplete_Rides_Reason
 FROM bookings
 WHERE Incomplete_Rides ='Yes';

select * from Incomplete_Rides_Reason;
```

Result Grid | Filter Rows:

Booking_ID	Incomplete_Rides_Reason
CNR5176704322	Customer Demand
CNR9312632867	Vehicle Breakdown
CNR7924302885	Customer Demand
CNR1640228587	Other Issue
CNR7623690602	Other Issue
CNR9590311980	Customer Demand
CNR5863244684	Customer Demand

OLA

Navigator

SCHEMAS

Filter objects

- ola
  - Tables
    - bookings
  - Views
  - Stored Procedures
  - Functions

Administration | Schemas

Information

No object selected

Query 1 | SQL File 4*

Limit to 50000 rows

```sql
#1. Retrieve all successful bookings:
select * from Successfull_Bookings;

#2. Find the average ride distance for each vehicle type:
select * from ride_distance_for_each_vehicle;

#3. 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:
SELECT * FROM Top_5_Customers;

#5. Get the number of rides cancelled by drivers due to personal and car-related issues:
SELECT * FROM Ride_Canceled_by_Drivers_P_C_Issues;

#6. 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:
SELECT * FROM UPI_Payment;

#8. Find the average customer rating per vehicle type:
SELECT * FROM AVG_Cust_Rating;

#9. Calculate the total booking value of rides completed successfully:
 SELECT * FROM total_successful_ride_value;

 #10. List all incomplete rides along with the reason:
 select * from Incomplete_Rides_Reason;
```

OLA

THANKYOU

for reviewing my project.
I welcome any feedback or recommendations to further enhance my skills and understanding.

Email

ritleshkumar03@gmail.com

LinkedIN

https://www.linkedin.com/in/ritlesh-kumar-740405280/

Phone

6201493881