SQL QUERIES

1. Retrieve all successful bookings:

```
CREATE VIEW SUCCESSFUL_BOOKINGS as

SELECT * FROM bookings WHERE Booking_Status = 'Success';

SELECT * FROM SUCCESSFUL_BOOKINGS;
```

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

```
CREATE VIEW average_ride_distance_for_each_vehicle AS

SELECT Vehicle_Type AS "Vehicle Typw", AVG(Ride_Distance) AS "Avg Ride Distance" FROM bookings GROUP BY Vehicle_Type;

SELECT * FROM average_ride_distance_for_each_vehicle;
```

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

```
CREATE VIEW rides_cancelled_by_customers AS

SELECT COUNT(*) AS "Total number of rides cancelled by Customers" FROM bookings WHERE Booking_Status ='Canceled by Customer';

SELECT * FROM rides_cancelled_by_customers;
```

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

```
CREATE VIEW top_5_customers AS

SELECT TOP 5 Customer_ID, COUNT(Booking_ID) AS Total_Bookings FROM bookings GROUP BY Customer_ID ORDER BY Total_Bookings DESC;

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 AS

SELECT COUNT(*) as "COUNT" FROM bookings WHERE Canceled_Rides_by_Driver ='Personal & Car related issue';

SELECT * FROM rides_Cancelled_by_drivers;
```

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

```
CREATE VIEW Max_Min_Rating_by_Prime_Sedan_bookings AS

SELECT MAX(Driver_Ratings) AS "MAX RATING IN PRIME SEDAN",
MIN(Driver_Ratings) AS "MIN RATING IN PRIME SEDAN" FROM bookings Where
Vehicle_Type='Prime Sedan';

SELECT * FROM Max_Min_Rating_by_Prime_Sedan_bookings;
```

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

```
CREATE VIEW Payments_by_UPI AS

SELECT * FROM bookings WHERE Payment_Method='UPI';

SELECT * FROM Payments_by_UPI;

SELECT count(*) FROM Payments_by_UPI;
```

8. Find the average customer rating per vehicle type:

```
CREATE VIEW Average_customer_rating_as_per_vehicle AS

SELECT Vehicle_Type, AVG(Customer_Rating) AS "Avg Customer Rating" FROM bookings
GROUP BY Vehicle_Type;

SELECT * FROM Average_customer_rating_as_per_vehicle;
```

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

```
CREATE VIEW Total_booking_value_of_rides_successful AS

SELECT SUM(Booking_Value) AS "Total Successful Value" FROM Bookings WHERE
Booking_Status = 'Success';

SELECT * FROM Total_booking_value_of_rides_successful;
```

10. List all incomplete rides along with the reason:

```
CREATE VIEW Incomplete_rides AS

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

SELECT * FROM Incomplete_rides;
```