SQL Questions:

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 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;
```

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

```
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:

```
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;
```

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(*) 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_and_min_driver_ratings_for_Prime_Sedan as
select max(Driver_Ratings) as max_rating,
min(Driver_Ratings) as min_rating
from bookings where Vehicle_Type = 'Prime Sedan';

select * from max_and_min_driver_ratings_for_Prime_Sedan;
```

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;
```

8. Find the average customer rating per vehicle type:

```
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;
```

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

```
create view total_successful_ride_value as
select sum(Booking_Value) as total_successful_ride_value
from bookings
where Booking_Status = 'Success';

select * from total_successful_ride_value;
```

10. List all incomplete rides along with the reason:

```
create view Incomplete_Rides_Reason as
select Booking_ID,Incomplete_Rides_Reason
from bookings
where Incomplete_Rides = 'Yes' ;
select * from Incomplete_Rides_Reason;
```