Deadline 5

Arnav Agarwal   2021235
Himanshu Sachdeva   2021256

Triggers

1)Trigger to update the distance and set an OTP for each ride after it has been been requested.

```
delimiter |
CREATE TRIGGER update_distance Before INSERT ON ride
FOR EACH ROW
    Begin
    set NEW.Distance=(select sqrt(power((l1.X_Coordinate-l2.X_Coordinate),2)+power((l1.Y_Coordinate-l2.Y_Coordinate),2))
    from location l1,location l2
    where NEW.SourceName=l1.Location_Name And NEW.DestinationName=l2.Location_Name);
    set NEW.OTP=(SELECT FLOOR(RAND() * 10000));
end
|
delimiter ;
```

2)Trigger to delete a ride from proposed booking after booking has been confirmed.

```
delimiter |
create trigger book_ride after Update on ride for each row
        Begin
            Delete from proposed_booking pb
            where NEW.Ride_Id=pb.Ride_Id And New.Status='Booked' And Old.Status!='Booked';
end
|
delimiter ;
```

3)Trigger to insert records corresponding to a ride into the proposed booking table. This means that the ride has been proposed to drivers nearby who may accept or reject that ride.

```
Create Trigger insert_into_pb after insert on ride
for each row
insert into proposed_booking(Driver_Id,Ride_Id)
Select Driver_Id,NEW.Ride_Id from vehicle where vehicle.Type=NEW.Vehicle_Type;
```

OLAP

1)An OLAP query(pivot) that shows the Number of Rides in each month corresponding to each area and vehicle type.

```sql
SELECT
    SourceName,Vehicle_Type,
    Count(Case when Month(Date)=1 then Ride_Id end) as 'January',
    Count(Case when Month(Date)=2 then Ride_Id end) as 'February',
    Count(Case when Month(Date)=3 then Ride_Id end) as 'March',
    Count(Case when Month(Date)=4 then Ride_Id end) as 'April',
    Count(Case when Month(Date)=5 then Ride_Id end) as 'May',
    Count(Case when Month(Date)=6 then Ride_Id end) as 'June',
    Count(Case when Month(Date)=6 then Ride_Id end) as 'June',
    Count(Case when Month(Date)=7 then Ride_Id end) as 'July',
    Count(Case when Month(Date)=8 then Ride_Id end) as 'August',
    count(Case when Month(Date)=9 then Ride_Id end) as 'September',
    Count(Case when Month(Date)=10 then Ride_Id end) as 'October',
    count(Case when Month(Date)=11 then Ride_Id end) as 'November',
    Count(Case when Month(Date)=12 then Ride_Id end) as 'December'
    from ride
group by SourceName,Vehicle_Type
order by SourceName;
```

2)An OLAP query(rollup) that shows total revenue generated by each vehicle type and further by each vehicle of that type.

```sql
SELECT
    Vehicle_Type,
    Driver_Id,
    SUM(Cost) AS TotalCost
FROM ride
where Status='COMPLETED'
Group By Vehicle_Type, Driver_Id, Customer_Id With ROllUP;
```

3)A OLAP query(Rollup) to show total revenue in each year and further in each month of that year.

```sql
Select Year(Date) as Y,Month(Date) as M,Date,Sum(Cost) from ride
where Status='COMPLETED'
Group by Year(Date),Month(Date),Date with Rollup;
```

4) An OLAP query(Slice) to find the number of customers and average trips taken by these customers belonging to Hauz Khas grouped according to age group(decade).

```sql
SELECT floor(Age/10)*10 as Age_Group, COUNT(Customer_id) as customer_count, AVG(No_of_Trips) as avg_trips
FROM customer
WHERE Home_Location = 'Hauz Khas'
GROUP BY floor(Age/10)*10;
```

Embedded Queries

Host Language:JavaScript

1)This query displays the rides that have been proposed to a driver ordered by distance between the source of the ride and current location of the driver.

```javascript
let sql = `Select Driver_Id,v4.s SourceName,v4.cl CurrentLocation, v4.Ride_id,v4.date,
  sqrt(power((a.X_Coordinate-b.X_Coordinate),2)+power((a.Y_Coordinate-b.Y_Coordinate),2))
  as Distance from v4
  Join location a on a.Location_Name=v4.s
  Join location b on b.Location_Name=v4.cl
  where Driver_Id=94 and v4.date not in
  (Select ride.Date from ride where Driver_id=94 and Status='Booked')
  order by Distance;`
```

2) We allow a user to sign up to the app and accept details of the customer. This query inserts a customer with appropriate data into the customer relation.

```
let sql =
  "INSERT INTO customer (Age, Name, Wallet, Phone_Number, Email_Address, Rating, No_of_Trips, Ho
  req.body.Age +
  "', '" +
  req.body.Name +
  "', '" +
  req.body.Wallet +
  "', '" +
  req.body.Phone_Number +
  "', '" +
  req.body.Email_Address +
  "', '" +
  req.body.Rating +
  "', '" +
  req.body.No_of_Trips +
  "', '" +
  req.body.Home_Location +
  "')"
```

GitHub link to the entire front end of the application:

https://github.com/lightninghimanshu/ComfortCruise_DBMS_Group26.git