

CLASSES:

Admin module:

- 1.Login
- 2.Update profile
- 3.View employee
- 4.Add employee
- 5.Feedbacks

Driver module:

- 1.Signup
- 2.Login
- 3.Update profile
- 4.Accept ride
- 5.Feedback

Customer module:

- 1.Signup
- 2.Login
- 3.Update profile
- 4.Book ride
- 5.Feedback

DATABASE TABLES:

1.Customer:

Fname
Lname
Mobile
Email
Gender
State
Password
PortNumber

//4.create table customer(Fname varchar2(30),Lname varchar2(30),Mobile int unique,Email varchar2(30) unique,Gender char(8),State char(20),Pass varchar2(30),portNumber int primary key);

2.Admin:

Email

Password

//7.create table admin(email varchar(30) primary key,password varchar(30));

3.Driver:

Fname

Lname

Car_ID

Gender

Dob

City

address

totalDrivingExp

Lisence ID

State

MobileNo

Email

Password

Approved

Availability

XP

//1.create table driver(fname char(20),lname char(20),car_ID varchar(20),gender char(6),dob date,city char(30),address varchar(100),DrivingExp int,lisence_ID varchar(10),nationality char(30),mobile int unique,email varchar(30) primary key,password varchar(30),approved char(5),availability char(5),xp int);

4.Car:

ownerEmail

carID

Company

Model

Capacity

Ac

FarePerKM

//2.create table car(ownerEmail varchar(30) references driver(email),carId varchar(10) primary key,company varchar(20),model varchar(30),capacity int,ac char(5),farePerKM int);

5.DriverFeedback:

email

SNo

Feedback

Ratings(Rating from 1-5)

//5.create table driverfeedback(email varchar(30) references driver(email),feedback clob,ratings int,feedbackNo primary key int);

6.CustomerFeedback:

email

SNo

Feedback

Ratings(Rating from 1-5)

//6.create table customerfeedback(email varchar(30) references customer(email),feedback clob,ratings int,feedbackNo int primary key);

To insert into customerfeedback , {call: exec(email,feedback,rating)}

FUNCTION:

SQL> create or replace function getcustomerlastfeedbackno

2 return number

3 is

4 ans number;

5 r customerfeedback%rowtype;

6 cursor c is

7 select * from customerfeedback;

8 begin

9 ans:=1;

10 open c;

11 loop

12 fetch c into r;

13 exit when c%notFound;

14 ans:=r.feedbackno;

15 end loop;

16 close c;

17 ans:=ans+1;

18 return ans;

```
19 end;  
20 /
```

PROCEDURE 2:

```
SQL> create or replace procedure insertCustomerFeedback(email varchar,feedback clob,ratings  
int) as  
2 begin  
3 insert into customerfeedback values(email,feedback,ratings,getcustomerlastfeedbackno());  
4 end;  
5 /
```

7.Ride:

Email (customer's)
NoOfPerson
Pickup
Dest
DriverAss
startKM
OTP

```
//3.create table ride(email varchar(30) references customer(email),noOfSeats int,pickup  
varchar(30),dest varchar(30),driverAssigned char(5),startKM int,otp int primary key);
```

//Procedure

```
SQL>create or replace procedure insertRide(email varchar,noOfSeats int,pickup  
varchar,dest varchar,driverAssigned char,startKM int,otp int) as  
2 begin  
3 insert into ride values(email,noOfSeats,pickup,dest,driverAssigned,startKM,otp);  
4 end;  
5 /
```

level 1: Documentation works

File Edit View Insert Format Tools Add-ons Help Last edit was seconds ago

Run SQL Command Line

```
SQL> desc ride;
Name                                     Null?   Type
-----
GENDER                                   CHAR(8)
STATE                                   CHAR(20)
PASS                                    VARCHAR2(30)
PORTNUMBER                             NOT NULL NUMBER(38)

SQL> create or replace procedure insertRide(email varchar,noOfSeats int,pickup varchar,dest varchar,driverAssigned char,
startKM int,otp int) as
2 begin
3 insert into ride values(email,noOfSeats,pickup,dest,driverAssigned,startKM,otp);
4 end;
5 /

Procedure created.

SQL> commit;

Commit complete.

SQL>
```

4 end;
5 /

58% 16:59 31-08-2021