University of Texas at Arlington

CSE 3330 Database Project 2 Part 3

CARZZ 3330

Aditya Chaudhari (1001747134)

Utkarsh Singh(1001722244)

HONOR CODE

I pledge, on my honor, to uphold UT Arlington's tradition of academic integrity, a tradition that values
hard work and honest effort in the pursuit of academic excellence.
I promise that I will submit only work that I personally create or that I contribute to group collaborations,
and I will appropriately reference any work from other sources. I will follow the highest standards of
integrity and uphold the spirit of the Honor Code. The project files and code can be found in the GitHub repositories:

https://github.com/UT07 (master branch)

https://github.com/adityachaudhari1412

Tools Used for project:

Database: MySQL

Programming Language: HTML, CSS, Javascript, babel javascript

Database tool: MySQLWorkbench

The data was already present in MySQL server from previous part, we made a node.js API to fetch and post the relevant data according to requirements.

Importing the Database and Table Creation

To create Databases and importing the provided one and creating the tables, columns were fetched from Project Part 2 queries only. The project can be found in GitHub Repository provided along with the documentation.

https://github.com/UT07

https://github.com/adityachaudhari1412

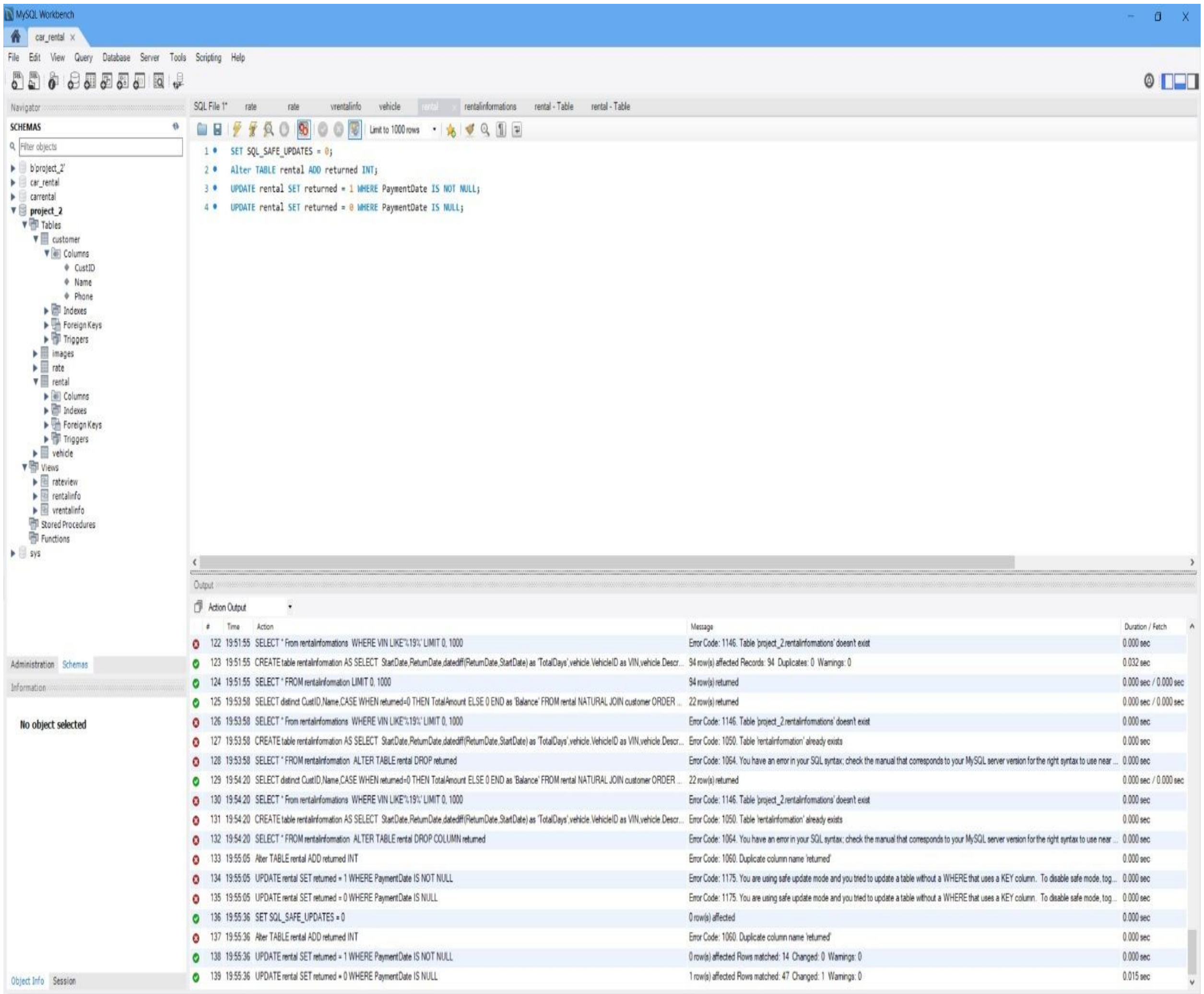
and can be found on following website created:

https://carz3330.herokuapp.com/

Task 1:

Query 1:

```
Alter TABLE rental ADD returned INT;

UPDATE rental SET returned = 1 WHERE PaymentDate IS NOT NULL;

UPDATE rental SET retuned = 0 WHERE PaymentDate IS NULL;
```



```sql
SET SQL_SAFE_UPDATES = 0;
Alter TABLE rental ADD returned INT;
UPDATE rental SET returned = 1 WHERE PaymentDate IS NOT NULL;
UPDATE rental SET returned = 0 WHERE PaymentDate IS NULL;
```

```
#    Time      Action                                                                                                                                   Message                                                                                                                    Duration / Fetch
122  19:51:55  SELECT * From rentalinfomations  WHERE VIN LIKE'%19%' LIMIT 0, 1000                                                                      Error Code: 1146. Table 'project_2.rentalinfomations' doesn't exist                                                       0.000 sec
123  19:51:55  CREATE table rentalinfomation AS SELECT  StartDate,ReturnDate,datediff(ReturnDate,StartDate) as 'TotalDays',vehicle.VehicleID as VIN,vehicle.Descr...  94 row(s) affected Records: 94  Duplicates: 0  Warnings: 0                                                  0.032 sec
124  19:51:55  SELECT * FROM rentalinfomation LIMIT 0, 1000                                                                                             94 row(s) returned                                                                                                         0.000 sec / 0.000 sec
125  19:53:58  SELECT distinct CustID,Name,CASE WHEN returned=0 THEN TotalAmount ELSE 0 END as 'Balance' FROM rental NATURAL JOIN customer ORDER ...    22 row(s) returned                                                                                                         0.000 sec / 0.000 sec
126  19:53:58  SELECT * From rentalinfomations  WHERE VIN LIKE'%19%' LIMIT 0, 1000                                                                      Error Code: 1146. Table 'project_2.rentalinfomations' doesn't exist                                                       0.000 sec
127  19:53:58  CREATE table rentalinfomation AS SELECT  StartDate,ReturnDate,datediff(ReturnDate,StartDate) as 'TotalDays',vehicle.VehicleID as VIN,vehicle.Descr...  Error Code: 1050. Table 'rentalinfomation' already exists                                                   0.000 sec
128  19:53:58  SELECT * FROM rentalinfomation  ALTER TABLE rental DROP returned                                                                          Error Code: 1064. You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near ...  0.000 sec
129  19:54:20  SELECT distinct CustID,Name,CASE WHEN returned=0 THEN TotalAmount ELSE 0 END as 'Balance' FROM rental NATURAL JOIN customer ORDER ...    22 row(s) returned                                                                                                         0.000 sec / 0.000 sec
130  19:54:20  SELECT * From rentalinfomations  WHERE VIN LIKE'%19%' LIMIT 0, 1000                                                                      Error Code: 1146. Table 'project_2.rentalinfomations' doesn't exist                                                       0.000 sec
131  19:54:20  CREATE table rentalinfomation AS SELECT  StartDate,ReturnDate,datediff(ReturnDate,StartDate) as 'TotalDays',vehicle.VehicleID as VIN,vehicle.Descr...  Error Code: 1050. Table 'rentalinfomation' already exists                                                   0.000 sec
132  19:54:20  SELECT * FROM rentalinfomation  ALTER TABLE rental DROP COLUMN returned                                                                   Error Code: 1064. You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near ...  0.000 sec
133  19:55:05  Alter TABLE rental ADD returned INT                                                                                                      Error Code: 1060. Duplicate column name 'returned'                                                                         0.000 sec
134  19:55:05  UPDATE rental SET returned = 1 WHERE PaymentDate IS NOT NULL                                                                            Error Code: 1175. You are using safe update mode and you tried to update a table without a WHERE that uses a KEY column.  To disable safe mode, tog...  0.000 sec
135  19:55:05  UPDATE rental SET returned = 0 WHERE PaymentDate IS NULL                                                                                Error Code: 1175. You are using safe update mode and you tried to update a table without a WHERE that uses a KEY column.  To disable safe mode, tog...  0.000 sec
136  19:55:36  SET SQL_SAFE_UPDATES = 0                                                                                                                0 row(s) affected                                                                                                          0.000 sec
137  19:55:36  Alter TABLE rental ADD returned INT                                                                                                      Error Code: 1060. Duplicate column name 'returned'                                                                         0.000 sec
138  19:55:36  UPDATE rental SET returned = 1 WHERE PaymentDate IS NOT NULL                                                                            0 row(s) affected Rows matched: 14  Changed: 0  Warnings: 0                                                                0.000 sec
139  19:55:36  UPDATE rental SET returned = 0 WHERE PaymentDate IS NULL                                                                                1 row(s) affected Rows matched: 47  Changed: 1  Warnings: 0                                                                0.015 sec
```

Query 2:

CREATE VIEW vRentalInfo AS

SELECT DISTINCT StartDate,ReturnDate,datediff(ReturnDate,StartDate) as 'TotalDays',vehicle.VehicleID as VIN,vehicle.Description as 'Vehicle', CASE WHEN

vehicle.Type=1 THEN 'Compact' WHEN vehicle.Type=2 THEN 'Medium' WHEN

vehicle.Type=3 THEN 'Large' WHEN vehicle.Type=4 THEN 'SUV' WHEN vehicle.Type=5

THEN 'Truck' ELSE 'Van' END AS Type, CASE WHEN vehicle.Category=0 THEN

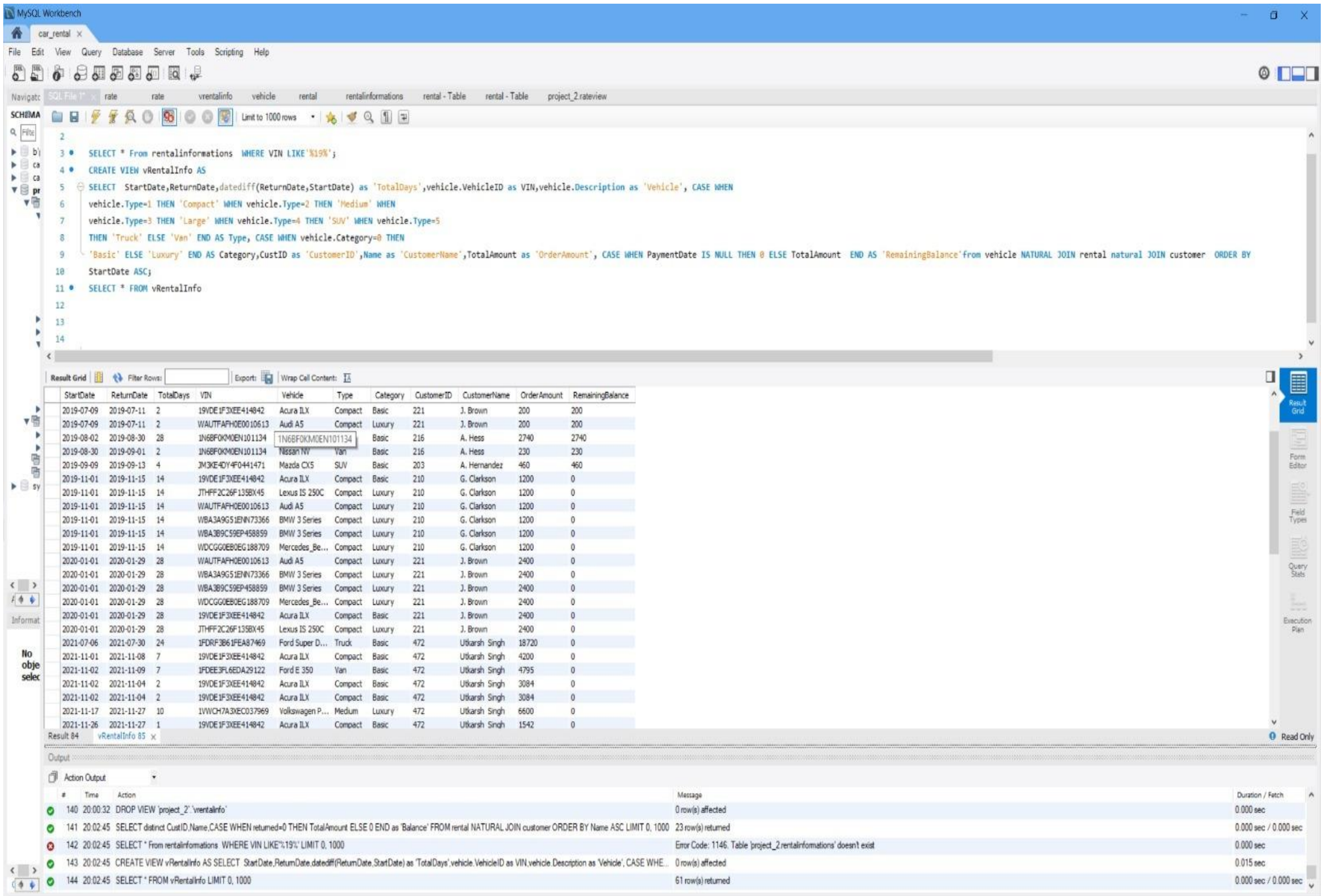
'Basic' ELSE 'Luxury' END AS Category,CustID as 'CustomerID',Name as 'CustomerName', TotalAmount as 'OrderAmount', CASE WHEN PaymentDate IS NULL THEN 0 ELSE TotalAmount END AS 'Remaining Balance'from vehicle NATURAL JOIN rental natural JOIN customer ORDER BY

StartDate ASC;

SELECT * FROM vRentalinfo;

SELECT distinct CustID,Name,CASE WHEN returned=0 THEN TotalAmount ELSE 0 END as 'Balance' FROM rental NATURAL JOIN customer;

SELECT distinct CustID,Name,CASE WHEN returned=0 THEN TotalAmount ELSE 0 END as 'Balance' FROM rental NATURAL JOIN customer ORDER BY Name ASC;


MySQL Workbench
car_rental
File Edit View Query Database Server Tools Scripting Help
SQL File 1* rate rate vrentalinfo vehicle rental rentalinformations rental - Table rental - Table project_2.rateview
Limit to 1000 rows
Result Grid Filter Rows: Export: Wrap Cell Content:
Result Grid Form Editor Field Types Query Stats Execution Plan
Read Only
Output
Action Output

```
SELECT * From rentalinformations  WHERE VIN LIKE'%19%';
CREATE VIEW vRentalInfo AS
SELECT  StartDate,ReturnDate,datediff(ReturnDate,StartDate) as 'TotalDays',vehicle.VehicleID as VIN,vehicle.Description as 'Vehicle', CASE WHEN
vehicle.Type=1 THEN 'Compact' WHEN vehicle.Type=2 THEN 'Medium' WHEN
vehicle.Type=3 THEN 'Large' WHEN vehicle.Type=4 THEN 'SUV' WHEN vehicle.Type=5
THEN 'Truck' ELSE 'Van' END AS Type, CASE WHEN vehicle.Category=0 THEN
'Basic' ELSE 'Luxury' END AS Category,CustID as 'CustomerID',Name as 'CustomerName',TotalAmount as 'OrderAmount', CASE WHEN PaymentDate IS NULL THEN 0 ELSE TotalAmount  END AS 'RemainingBalance'from vehicle NATURAL JOIN rental natural JOIN customer  ORDER BY
StartDate ASC;
SELECT * FROM vRentalInfo
```

StartDate	ReturnDate	TotalDays	VIN	Vehicle	Type	Category	CustomerID	CustomerName	OrderAmount	RemainingBalance
2019-07-09	2019-07-11	2	19VDE1F3XEE414842	Acura ILX	Compact	Basic	221	J. Brown	200	200
2019-07-09	2019-07-11	2	WAUTFAFH0E0010613	Audi A5	Compact	Luxury	221	J. Brown	200	200
2019-08-02	2019-08-30	28	1N6BF0KM0EN101134	1N6BF0KM0EN101134		Basic	216	A. Hess	2740	2740
2019-08-30	2019-09-01	2	1N6BF0KM0EN101134	Nissan NV	Van	Basic	216	A. Hess	230	230
2019-09-09	2019-09-13	4	JM3KE4DY4F0441471	Mazda CX5	SUV	Basic	203	A. Hernandez	460	460
2019-11-01	2019-11-15	14	19VDE1F3XEE414842	Acura ILX	Compact	Basic	210	G. Clarkson	1200	0
2019-11-01	2019-11-15	14	JTHFF2C26F135BX45	Lexus IS 250C	Compact	Luxury	210	G. Clarkson	1200	0
2019-11-01	2019-11-15	14	WAUTFAFH0E0010613	Audi A5	Compact	Luxury	210	G. Clarkson	1200	0
2019-11-01	2019-11-15	14	WBA3A9G51ENN73366	BMW 3 Series	Compact	Luxury	210	G. Clarkson	1200	0
2019-11-01	2019-11-15	14	WBA3B9C59EP458859	BMW 3 Series	Compact	Luxury	210	G. Clarkson	1200	0
2019-11-01	2019-11-15	14	WDCGG0EB0EG188709	Mercedes_Be...	Compact	Luxury	210	G. Clarkson	1200	0
2020-01-01	2020-01-29	28	WAUTFAFH0E0010613	Audi A5	Compact	Luxury	221	J. Brown	2400	0
2020-01-01	2020-01-29	28	WBA3A9G51ENN73366	BMW 3 Series	Compact	Luxury	221	J. Brown	2400	0
2020-01-01	2020-01-29	28	WBA3B9C59EP458859	BMW 3 Series	Compact	Luxury	221	J. Brown	2400	0
2020-01-01	2020-01-29	28	WDCGG0EB0EG188709	Mercedes_Be...	Compact	Luxury	221	J. Brown	2400	0
2020-01-01	2020-01-29	28	19VDE1F3XEE414842	Acura ILX	Compact	Basic	221	J. Brown	2400	0
2020-01-01	2020-01-29	28	JTHFF2C26F135BX45	Lexus IS 250C	Compact	Luxury	221	J. Brown	2400	0
2021-07-06	2021-07-30	24	1FDRF3B61FEA87469	Ford Super D...	Truck	Basic	472	Utkarsh Singh	18720	0
2021-11-01	2021-11-08	7	19VDE1F3XEE414842	Acura ILX	Compact	Basic	472	Utkarsh Singh	4200	0
2021-11-02	2021-11-09	7	1FDEE3FL6EDA29122	Ford E 350	Van	Basic	472	Utkarsh Singh	4795	0
2021-11-02	2021-11-04	2	19VDE1F3XEE414842	Acura ILX	Compact	Basic	472	Utkarsh Singh	3084	0
2021-11-02	2021-11-04	2	19VDE1F3XEE414842	Acura ILX	Compact	Basic	472	Utkarsh Singh	3084	0
2021-11-17	2021-11-27	10	1VWCH7A3XEC037969	Volkswagen P...	Medium	Luxury	472	Utkarsh Singh	6600	0
2021-11-26	2021-11-27	1	19VDE1F3XEE414842	Acura ILX	Compact	Basic	472	Utkarsh Singh	1542	0

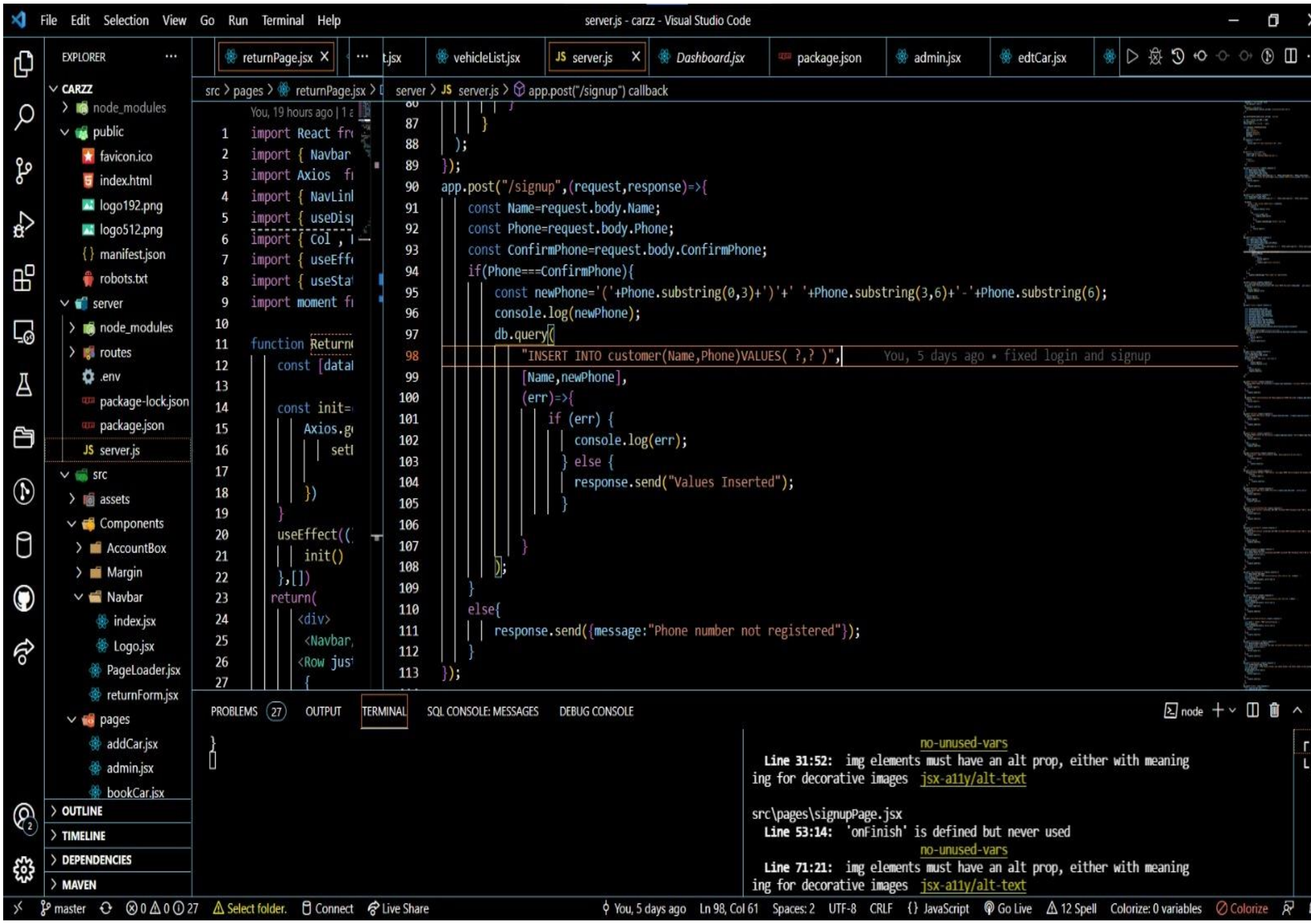
Result 84   vRentalInfo 85

```
#    Time      Action                                                                                                                       Message                                                          Duration / Fetch
140  20:00:32  DROP VIEW 'project_2'.'vrentalinfo'                                                                                          0 row(s) affected                                                0.000 sec
141  20:02:45  SELECT distinct CustID,Name,CASE WHEN returned=0 THEN TotalAmount ELSE 0 END as 'Balance' FROM rental NATURAL JOIN customer ORDER BY Name ASC LIMIT 0, 1000   23 row(s) returned   0.000 sec / 0.000 sec
142  20:02:45  SELECT * From rentalinformations  WHERE VIN LIKE'%19%' LIMIT 0, 1000                                                          Error Code: 1146. Table 'project_2.rentalinformations' doesn't exist   0.000 sec
143  20:02:45  CREATE VIEW vRentalInfo AS SELECT  StartDate,ReturnDate,datediff(ReturnDate,StartDate) as 'TotalDays',vehicle.VehicleID as VIN,vehicle.Description as 'Vehicle', CASE WHE...   0 row(s) affected   0.015 sec
144  20:02:45  SELECT * FROM vRentalInfo LIMIT 0, 1000                                                                                      61 row(s) returned                                               0.000 sec / 0.000 sec
```

Task-2:

SS from GUI

1)



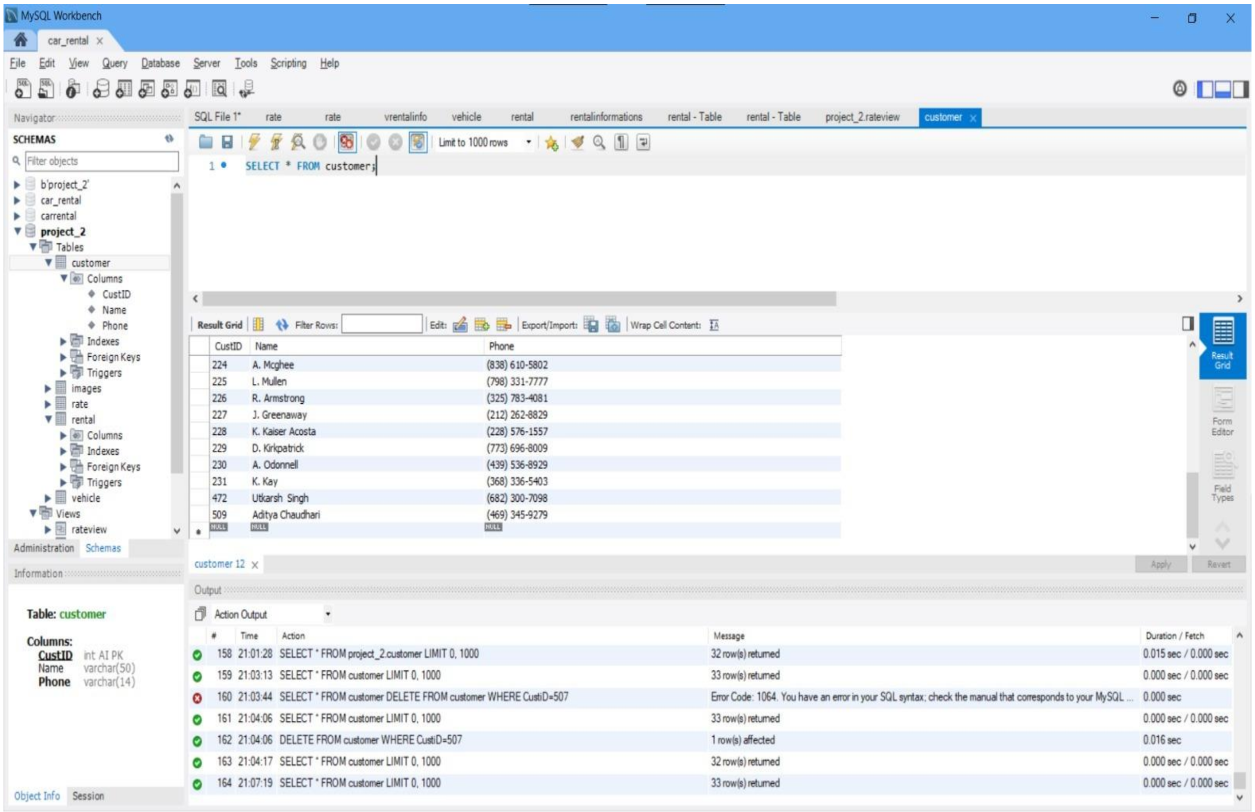
```
import React fr
import { Navbar
import Axios  f
import { NavLinl
import { useDis
import { Col , 
import { useEff
import { useSta
import moment f

function Return
  const [datal

  const init=
    Axios.g
      setI

    })
  }
  useEffect((
    init()
  },[])
  return(
    <div>
      <Navbar
      <Row jus
        {
```

```
      }
  );
});
app.post("/signup",(request,response)=>{
  const Name=request.body.Name;
  const Phone=request.body.Phone;
  const ConfirmPhone=request.body.ConfirmPhone;
  if(Phone===ConfirmPhone){
    const newPhone='('+Phone.substring(0,3)+')'+' '+Phone.substring(3,6)+'-'+Phone.substring(6);
    console.log(newPhone);
    db.query(
      "INSERT INTO customer(Name,Phone)VALUES( ?,? )",       You, 5 days ago • fixed login and signup
      [Name,newPhone],
      (err)=>{
        if (err) {
          console.log(err);
        } else {
          response.send("Values Inserted");
        }

      }
    );
  }
  else{
    response.send({message:"Phone number not registered"});
  }
});
```

```
}
```

```
                       no-unused-vars
  Line 31:52:  img elements must have an alt prop, either with meaning
ing for decorative images  jsx-a11y/alt-text

src\pages\signupPage.jsx
  Line 53:14:  'onFinish' is defined but never used
                       no-unused-vars
  Line 71:21:  img elements must have an alt prop, either with meaning
ing for decorative images  jsx-a11y/alt-text
```

Backend execution for task 1 where wea re adding a new customer via signup form using AXIOS to send a post request and execute the SQL query.



```
SELECT * FROM customer;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

CustID	Name	Phone
224	A. Mcghee	(838) 610-5802
225	L. Mullen	(798) 331-7777
226	R. Armstrong	(325) 783-4081
227	J. Greenaway	(212) 262-8829
228	K. Kaiser Acosta	(228) 576-1557
229	D. Kirkpatrick	(773) 696-8009
230	A. Odonnell	(439) 536-8929
231	K. Kay	(368) 336-5403
472	Utkarsh Singh	(682) 300-7098
509	Aditya Chaudhari	(469) 345-9279
NULL	NULL	NULL

customer 12

Information

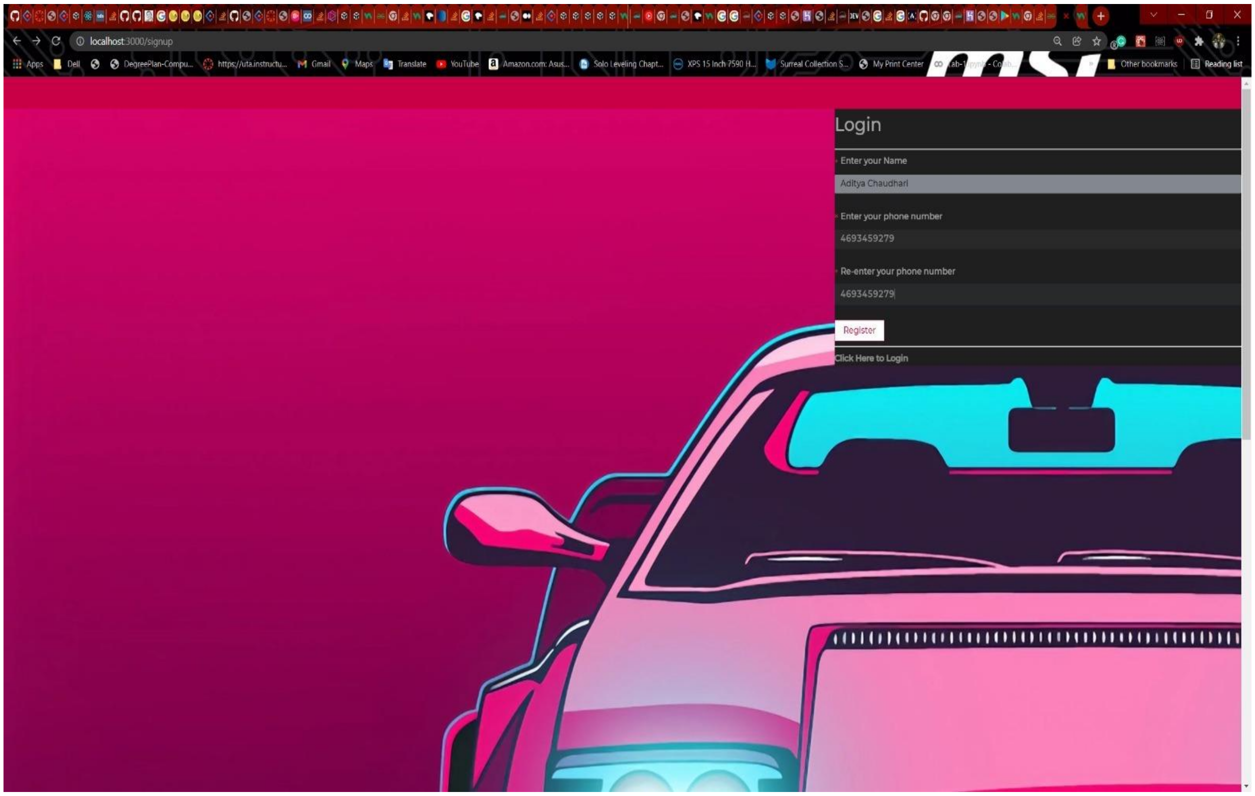
Table: customer

Columns:
CustID int AI PK
Name varchar(50)
Phone varchar(14)

Output

Action Output

```
#    Time      Action                                                     Message                                                                                         Duration / Fetch
158  21:01:28  SELECT * FROM project_2.customer LIMIT 0, 1000              32 row(s) returned                                                                              0.015 sec / 0.000 sec
159  21:03:13  SELECT * FROM customer LIMIT 0, 1000                        33 row(s) returned                                                                              0.000 sec / 0.000 sec
160  21:03:44  SELECT * FROM customer DELETE FROM customer WHERE CustiD=507 Error Code: 1064. You have an error in your SQL syntax; check the manual that corresponds to your MySQL ...  0.000 sec
161  21:04:06  SELECT * FROM customer LIMIT 0, 1000                        33 row(s) returned                                                                              0.000 sec / 0.000 sec
162  21:04:06  DELETE FROM customer WHERE CustiD=507                       1 row(s) affected                                                                               0.016 sec
163  21:04:17  SELECT * FROM customer LIMIT 0, 1000                        32 row(s) returned                                                                              0.000 sec / 0.000 sec
164  21:07:19  SELECT * FROM customer LIMIT 0, 1000                        33 row(s) returned                                                                              0.000 sec / 0.000 sec
```


localhost:3000/signup
Apps Dell DegreePlan-Compu... https://uta.instructu... Gmail Maps Translate YouTube Amazon.com: Asus... Solo Leveling Chapt... XPS 15 Inch 7590 H... Surreal Collection S... My Print Center Other bookmarks Reading list

Login

Enter your Name

Aditya Chaudhari

Enter your phone number

4693459279

Re-enter your phone number

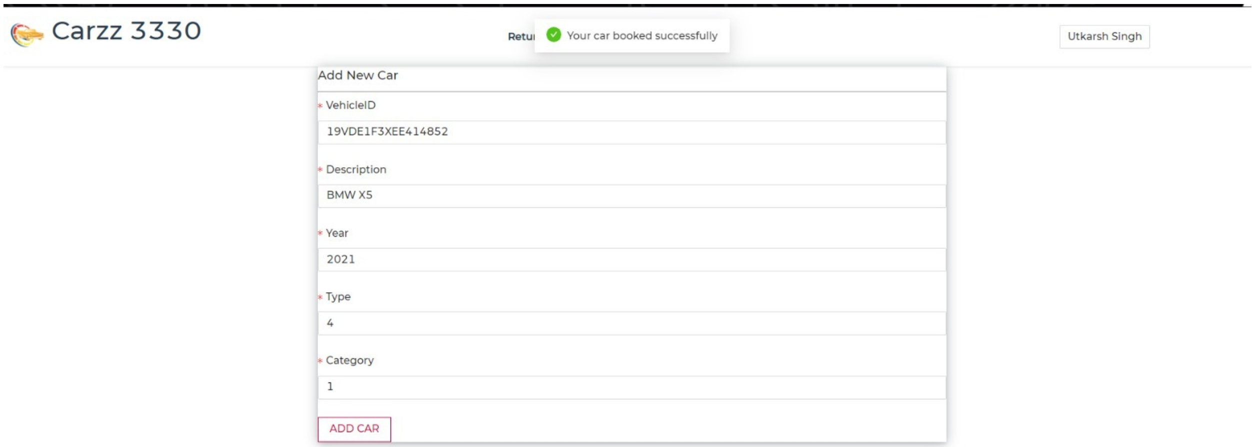
4693459279

Register

Click Here to Login

Front end GUI for sign up.

2) Front end gui for adding new car



Your car booked successfully

Add New Car

* VehicleID

19VDE1F3XEE414852

* Description

BMW X5

* Year

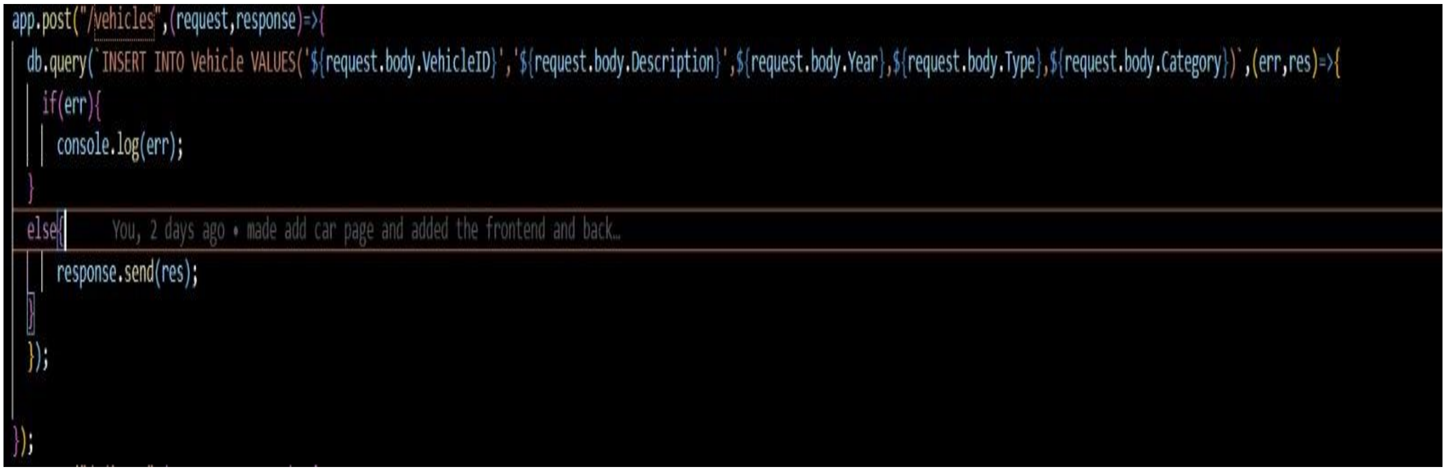
2021

* Type

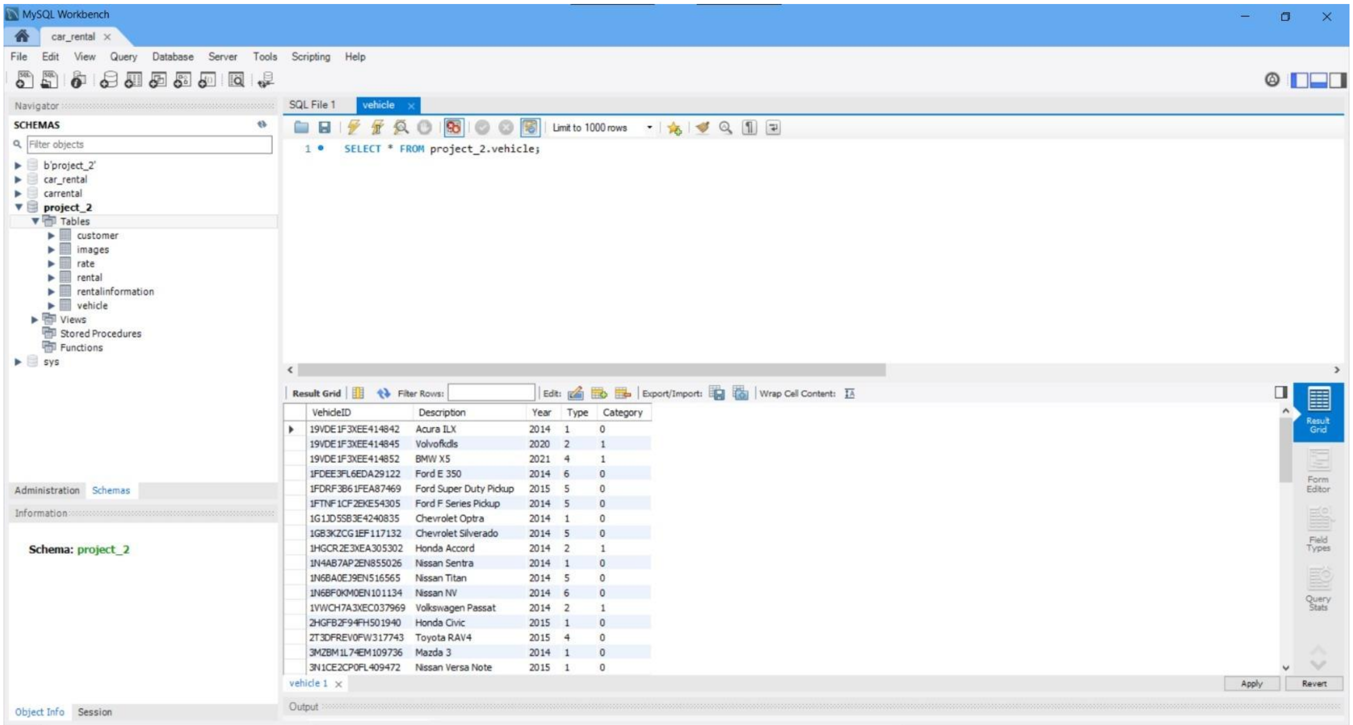
4

* Category

1



```
app.post("/vehicles",(request,response)=>{
  db.query(`INSERT INTO Vehicle VALUES('${request.body.VehicleID}','${request.body.Description}',${request.body.Year},${request.body.Type},${request.body.Category})`,(err,res)=>{
    if(err){
      console.log(err);
    }
  else{      You, 2 days ago • made add car page and added the frontend and back…
      response.send(res);
  }
  });

});
```



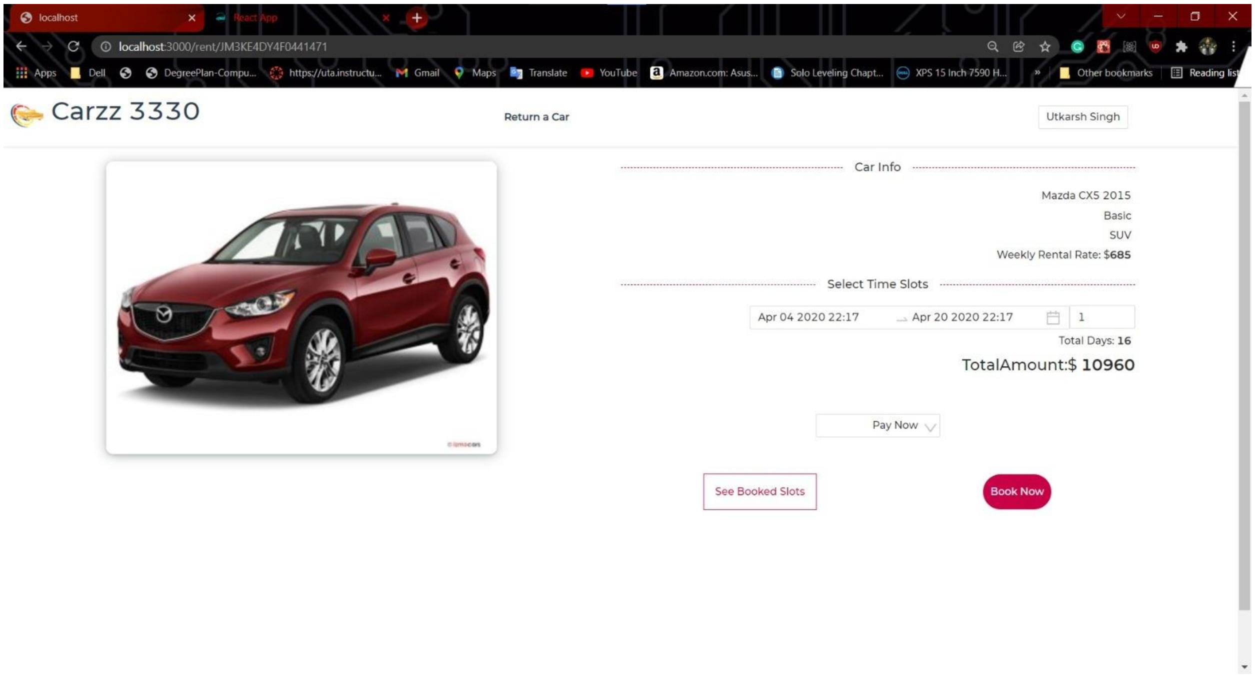
Schema: project_2

```
SELECT * FROM project_2.vehicle;
```

VehicleID	Description	Year	Type	Category
19VDE1F3XEE414842	Acura ILX	2014	1	0
19VDE1F3XEE414845	Volvofkdls	2020	2	1
19VDE1F3XEE414852	BMW X5	2021	4	1
1FDEE3FL6EDA29122	Ford E 350	2014	6	0
1FDRF3B61FEA87469	Ford Super Duty Pickup	2015	5	0
1FTNF1CF2EKE54305	Ford F Series Pickup	2014	5	0
1G1JD5SB3E4240835	Chevrolet Optra	2014	1	0
1GB3KZCG1EF117132	Chevrolet Silverado	2014	5	0
1HGCR2E3XEA305302	Honda Accord	2014	2	1
1N4AB7AP2EN855026	Nissan Sentra	2014	1	0
1N6BA0EJ9EN516565	Nissan Titan	2014	5	0
1N6BF0KM0EN101134	Nissan NV	2014	6	0
1VWCH7A3XEC037969	Volkswagen Passat	2014	2	1
2HGFB2F94FH501940	Honda Civic	2015	1	0
2T3DFREV0FW317743	Toyota RAV4	2015	4	0
3MZBM1L74EM109736	Mazda 3	2014	1	0
3N1CE2CP0FL409472	Nissan Versa Note	2015	1	0

executing post request through axios to run sql query to add a new car

3) Front end gui for booking the car

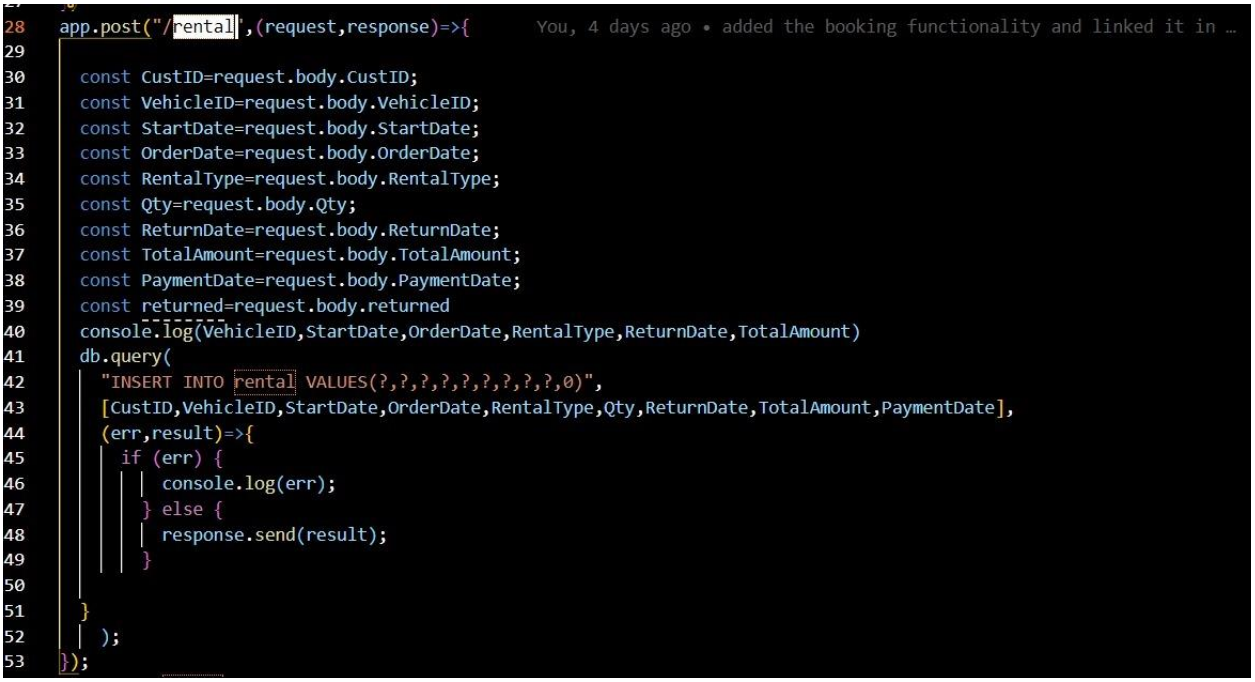

localhost
React App
localhost:3000/rent/JM3KE4DY4F0441471
Apps Dell DegreePlan-Compu... https://uta.instructu... Gmail Maps Translate YouTube Amazon.com: Asus... Solo Leveling Chapt... XPS 15 Inch 7590 H... Other bookmarks Reading list
Carzz 3330
Return a Car
Utkarsh Singh
Car Info
Select Time Slots
Pay Now
See Booked Slots
Book Now

Mazda CX5 2015
Basic
SUV
Weekly Rental Rate: $685

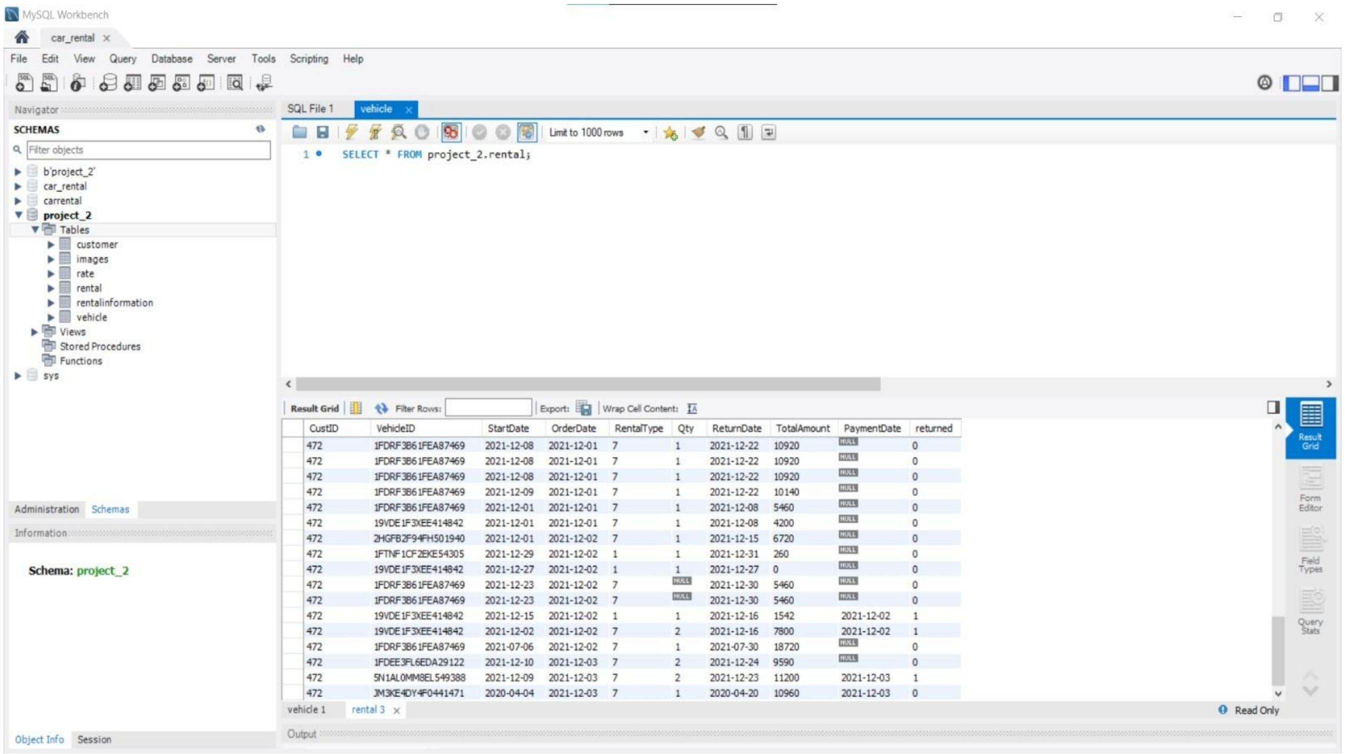
Apr 04 2020 22:17 → Apr 20 2020 22:17 | 1

Total Days: 16

TotalAmount:$ 10960



```
app.post("/rental",(request,response)=>{       You, 4 days ago • added the booking functionality and linked it in …

  const CustID=request.body.CustID;
  const VehicleID=request.body.VehicleID;
  const StartDate=request.body.StartDate;
  const OrderDate=request.body.OrderDate;
  const RentalType=request.body.RentalType;
  const Qty=request.body.Qty;
  const ReturnDate=request.body.ReturnDate;
  const TotalAmount=request.body.TotalAmount;
  const PaymentDate=request.body.PaymentDate;
  const returned=request.body.returned
  console.log(VehicleID,StartDate,OrderDate,RentalType,ReturnDate,TotalAmount)
  db.query(
    "INSERT INTO rental VALUES(?,?,?,?,?,?,?,?,?,0)",
    [CustID,VehicleID,StartDate,OrderDate,RentalType,Qty,ReturnDate,TotalAmount,PaymentDate],
    (err,result)=>{
      if (err) {
        console.log(err);
      } else {
        response.send(result);
      }

  }
    );
});
```

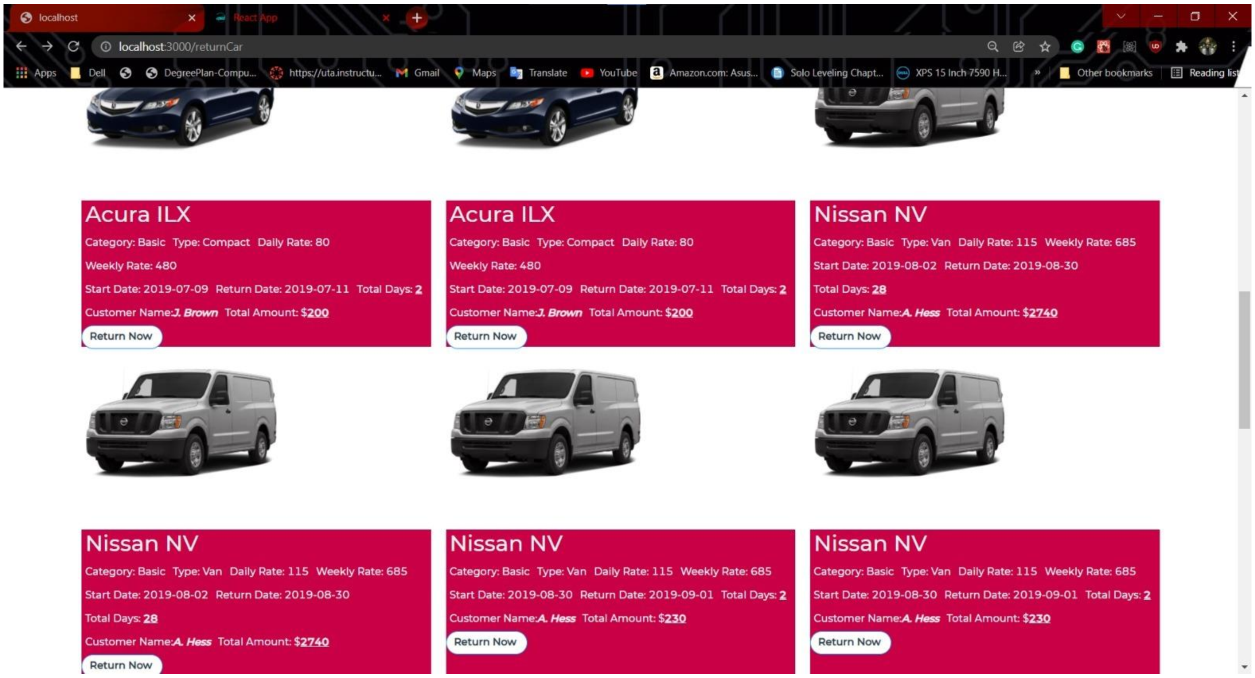


Schema: project_2

```
1 ● SELECT * FROM project_2.rental;
```

CustID	VehicleID	StartDate	OrderDate	RentalType	Qty	ReturnDate	TotalAmount	PaymentDate	returned
472	1FDRF3B61FEA87469	2021-12-08	2021-12-01	7	1	2021-12-22	10920	NULL	0
472	1FDRF3B61FEA87469	2021-12-08	2021-12-01	7	1	2021-12-22	10920	NULL	0
472	1FDRF3B61FEA87469	2021-12-08	2021-12-01	7	1	2021-12-22	10920	NULL	0
472	1FDRF3B61FEA87469	2021-12-09	2021-12-01	7	1	2021-12-22	10140	NULL	0
472	1FDRF3B61FEA87469	2021-12-01	2021-12-01	7	1	2021-12-08	5460	NULL	0
472	19VDE1F3XEE414842	2021-12-01	2021-12-01	7	1	2021-12-08	4200	NULL	0
472	2HGFB2F94FH501940	2021-12-01	2021-12-02	7	1	2021-12-15	6720	NULL	0
472	1FTNF1CF2EKE54305	2021-12-29	2021-12-02	1	1	2021-12-31	260	NULL	0
472	19VDE1F3XEE414842	2021-12-27	2021-12-02	1	1	2021-12-27	0	NULL	0
472	1FDRF3B61FEA87469	2021-12-23	2021-12-02	7	NULL	2021-12-30	5460	NULL	0
472	1FDRF3B61FEA87469	2021-12-23	2021-12-02	7	NULL	2021-12-30	5460	NULL	0
472	19VDE1F3XEE414842	2021-12-15	2021-12-02	1	1	2021-12-16	1542	2021-12-02	1
472	19VDE1F3XEE414842	2021-12-02	2021-12-02	7	2	2021-12-16	7800	2021-12-02	1
472	1FDRF3B61FEA87469	2021-07-06	2021-12-02	7	1	2021-07-30	18720	NULL	0
472	1FDEE3FL6EDA29122	2021-12-10	2021-12-03	7	2	2021-12-24	9590	NULL	0
472	5N1AL0MM8EL549388	2021-12-09	2021-12-03	7	2	2021-12-23	11200	2021-12-03	1
472	JM3KE4DY4F0441471	2020-04-04	2021-12-03	7	1	2020-04-20	10960	2021-12-03	0

4) Frontend gui for returning the car part-1 (Retrieving all the cars that needs to be returned)



Acura ILX
Category: Basic Type: Compact Daily Rate: 80
Weekly Rate: 480
Start Date: 2019-07-09 Return Date: 2019-07-11 Total Days: 2
Customer Name:J. Brown Total Amount: $200
Return Now

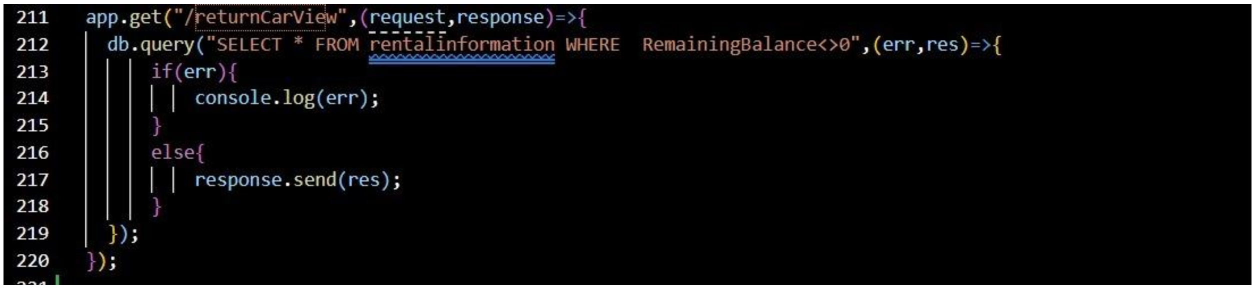
Acura ILX
Category: Basic Type: Compact Daily Rate: 80
Weekly Rate: 480
Start Date: 2019-07-09 Return Date: 2019-07-11 Total Days: 2
Customer Name:J. Brown Total Amount: $200
Return Now

Nissan NV
Category: Basic Type: Van Daily Rate: 115 Weekly Rate: 685
Start Date: 2019-08-02 Return Date: 2019-08-30
Total Days: 28
Customer Name:A. Hess Total Amount: $2740
Return Now

Nissan NV
Category: Basic Type: Van Daily Rate: 115 Weekly Rate: 685
Start Date: 2019-08-02 Return Date: 2019-08-30
Total Days: 28
Customer Name:A. Hess Total Amount: $2740
Return Now

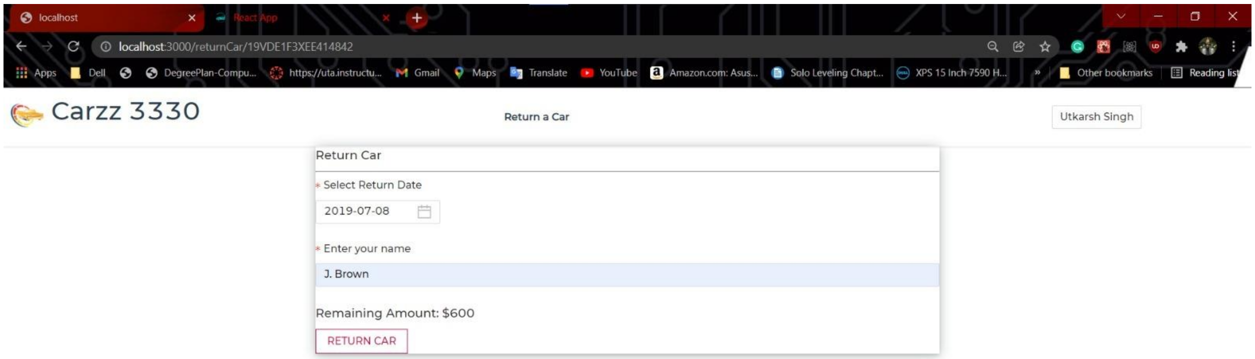
Nissan NV
Category: Basic Type: Van Daily Rate: 115 Weekly Rate: 685
Start Date: 2019-08-30 Return Date: 2019-09-01 Total Days: 2
Customer Name:A. Hess Total Amount: $230
Return Now

Nissan NV
Category: Basic Type: Van Daily Rate: 115 Weekly Rate: 685
Start Date: 2019-08-30 Return Date: 2019-09-01 Total Days: 2
Customer Name:A. Hess Total Amount: $230
Return Now



```
app.get("/returnCarView",(request,response)=>{
  db.query("SELECT * FROM rentalinformation WHERE  RemainingBalance<>0",(err,res)=>{
      if(err){
          console.log(err);
      }
      else{
          response.send(res);
      }
  });
});
```

Part-2(Returning the selected car when return now button is pressed.)



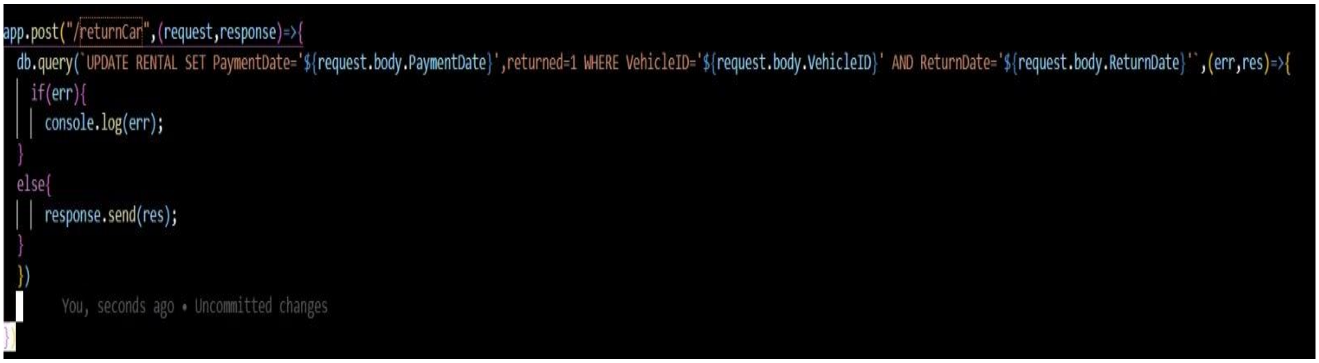
Return Car

* Select Return Date
2019-07-08

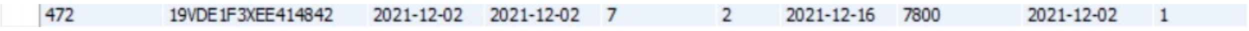
* Enter your name
J. Brown

Remaining Amount: $600

RETURN CAR

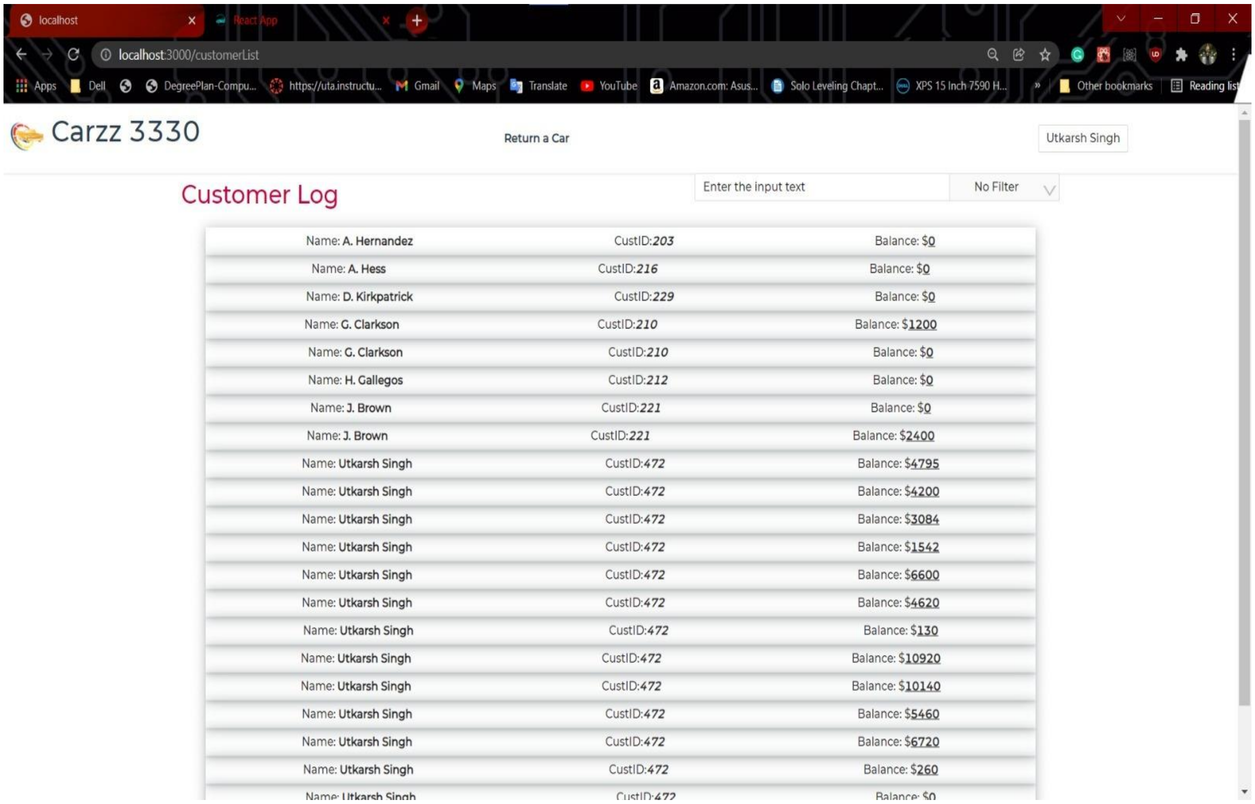


```
app.post("/returnCar",(request,response)=>{
  db.query(`UPDATE RENTAL SET PaymentDate='${request.body.PaymentDate}',returned=1 WHERE VehicleID='${request.body.VehicleID}' AND ReturnDate='${request.body.ReturnDate}'`,(err,res)=>{
    if(err){
      console.log(err);
    }
  else{
      response.send(res);
    }
  })
      You, seconds ago • Uncommitted changes
})
```



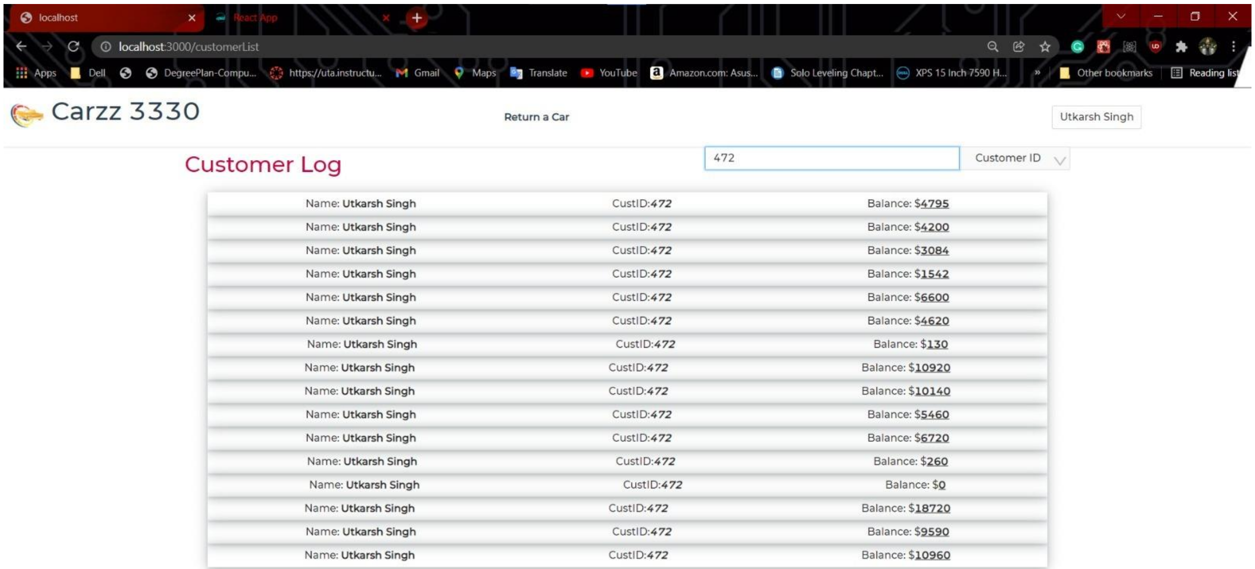
472	19VDE1F3XEE414842	2021-12-02	2021-12-02	7	2	2021-12-16	7800	2021-12-02	1

5) a) Front end gui for every customer ids remaining balance returning every balance.



Customer Log

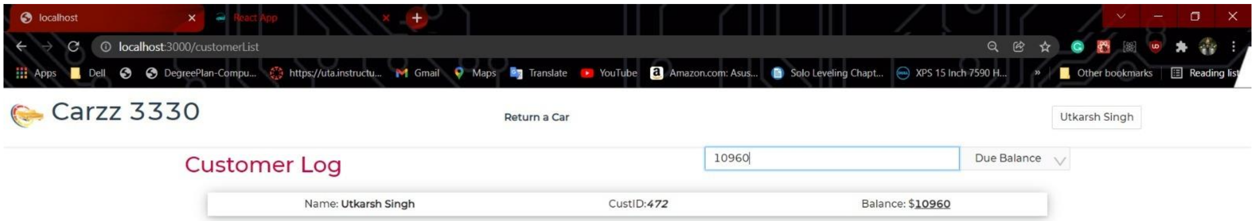
Name: **A. Hernandez**	CustID:***203***	Balance: $0
Name: **A. Hess**	CustID:***216***	Balance: $0
Name: **D. Kirkpatrick**	CustID:***229***	Balance: $0
Name: **G. Clarkson**	CustID:***210***	Balance: $1200
Name: **G. Clarkson**	CustID:***210***	Balance: $0
Name: **H. Gallegos**	CustID:***212***	Balance: $0
Name: **J. Brown**	CustID:***221***	Balance: $0
Name: **J. Brown**	CustID:***221***	Balance: $2400
Name: **Utkarsh Singh**	CustID:***472***	Balance: $4795
Name: **Utkarsh Singh**	CustID:***472***	Balance: $4200
Name: **Utkarsh Singh**	CustID:***472***	Balance: $3084
Name: **Utkarsh Singh**	CustID:***472***	Balance: $1542
Name: **Utkarsh Singh**	CustID:***472***	Balance: $6600
Name: **Utkarsh Singh**	CustID:***472***	Balance: $4620
Name: **Utkarsh Singh**	CustID:***472***	Balance: $130
Name: **Utkarsh Singh**	CustID:***472***	Balance: $10920
Name: **Utkarsh Singh**	CustID:***472***	Balance: $10140
Name: **Utkarsh Singh**	CustID:***472***	Balance: $5460
Name: **Utkarsh Singh**	CustID:***472***	Balance: $6720
Name: **Utkarsh Singh**	CustID:***472***	Balance: $260
Name: **Utkarsh Singh**	CustID:***472***	Balance: $0



Customer Log

472 | Customer ID

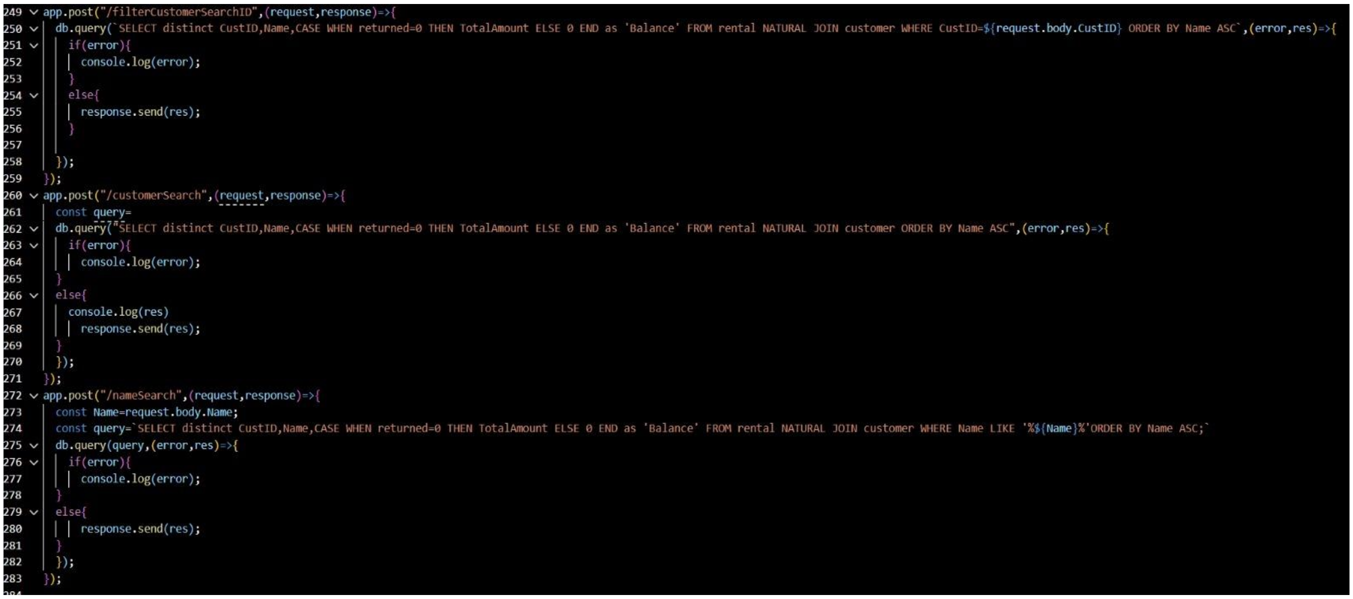
Name: **Utkarsh Singh**	CustID:***472***	Balance: $**4795**
Name: **Utkarsh Singh**	CustID:***472***	Balance: $**4200**
Name: **Utkarsh Singh**	CustID:***472***	Balance: $**3084**
Name: **Utkarsh Singh**	CustID:***472***	Balance: $**1542**
Name: **Utkarsh Singh**	CustID:***472***	Balance: $**6600**
Name: **Utkarsh Singh**	CustID:***472***	Balance: $**4620**
Name: **Utkarsh Singh**	CustID:***472***	Balance: $**130**
Name: **Utkarsh Singh**	CustID:***472***	Balance: $**10920**
Name: **Utkarsh Singh**	CustID:***472***	Balance: $**10140**
Name: **Utkarsh Singh**	CustID:***472***	Balance: $**5460**
Name: **Utkarsh Singh**	CustID:***472***	Balance: $**6720**
Name: **Utkarsh Singh**	CustID:***472***	Balance: $**260**
Name: **Utkarsh Singh**	CustID:***472***	Balance: $**0**
Name: **Utkarsh Singh**	CustID:***472***	Balance: $**18720**
Name: **Utkarsh Singh**	CustID:***472***	Balance: $**9590**
Name: **Utkarsh Singh**	CustID:***472***	Balance: $**10960**



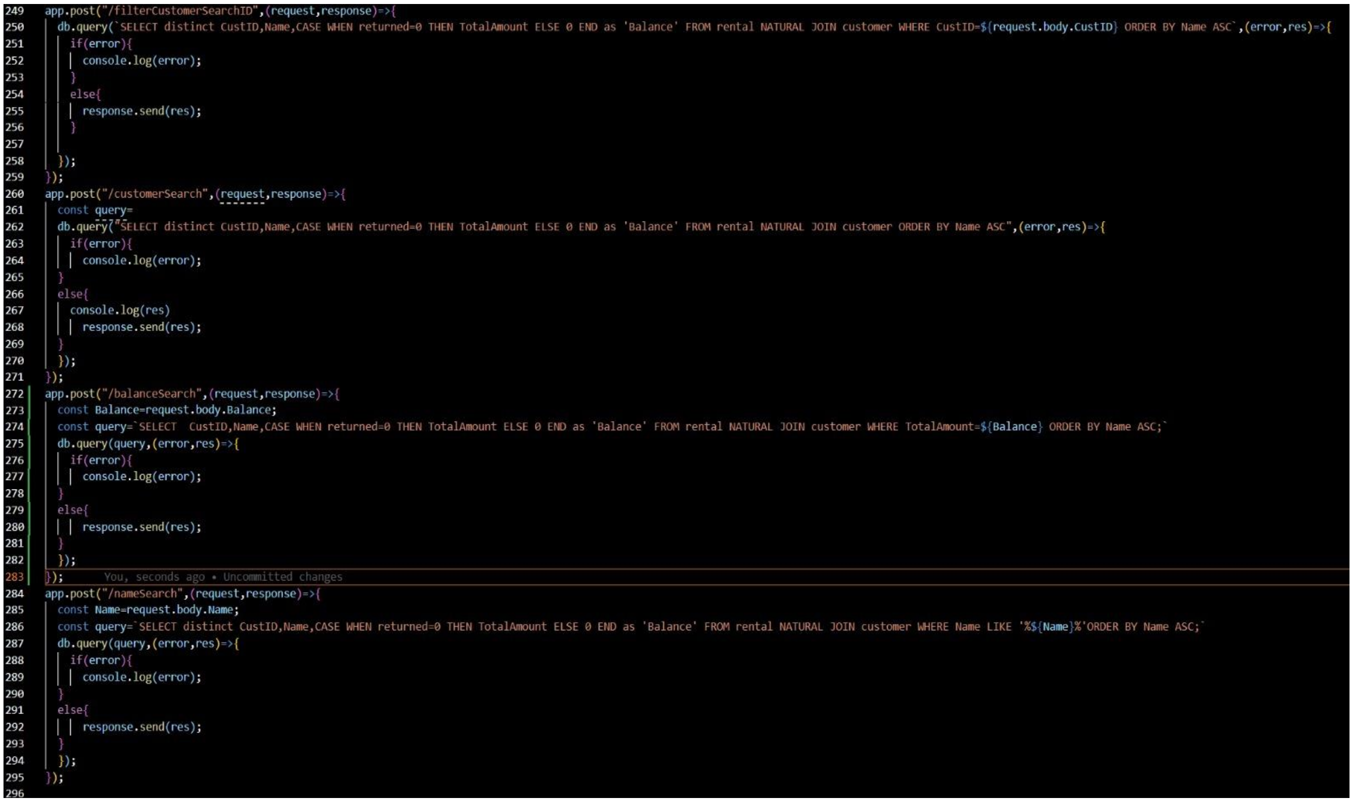
Customer Log

10960 | Due Balance

Name: **Utkarsh Singh**	CustID:***472***	Balance: $**10960**

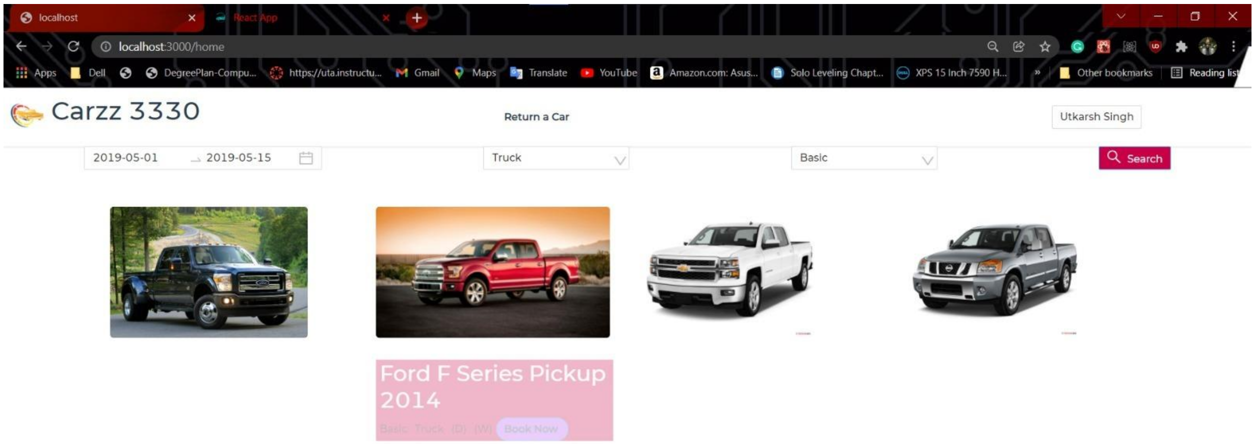


```
app.post("/filterCustomerSearchID",(request,response)=>{
  db.query(`SELECT distinct CustID,Name,CASE WHEN returned=0 THEN TotalAmount ELSE 0 END as 'Balance' FROM rental NATURAL JOIN customer WHERE CustID=${request.body.CustID} ORDER BY Name ASC`,(error,res)=>{
    if(error){
      console.log(error);
    }
    else{
      response.send(res);
    }

  });
});
app.post("/customerSearch",(request,response)=>{
  const query=
  db.query("SELECT distinct CustID,Name,CASE WHEN returned=0 THEN TotalAmount ELSE 0 END as 'Balance' FROM rental NATURAL JOIN customer ORDER BY Name ASC",(error,res)=>{
    if(error){
      console.log(error);
  }
  else{
    console.log(res)
      response.send(res);
  }
  });
});
app.post("/nameSearch",(request,response)=>{
  const Name=request.body.Name;
  const query=`SELECT distinct CustID,Name,CASE WHEN returned=0 THEN TotalAmount ELSE 0 END as 'Balance' FROM rental NATURAL JOIN customer WHERE Name LIKE '%${Name}%'ORDER BY Name ASC;`
  db.query(query,(error,res)=>{
    if(error){
      console.log(error);
  }
  else{
      response.send(res);
  }
  });
});
```



```
app.post("/filterCustomerSearchID",(request,response)=>{
  db.query(`SELECT distinct CustID,Name,CASE WHEN returned=0 THEN TotalAmount ELSE 0 END as 'Balance' FROM rental NATURAL JOIN customer WHERE CustID=${request.body.CustID} ORDER BY Name ASC`,(error,res)=>{
    if(error){
      console.log(error);
    }
    else{
      response.send(res);
    }

  });
});
app.post("/customerSearch",(request,response)=>{
  const query=
  db.query("SELECT distinct CustID,Name,CASE WHEN returned=0 THEN TotalAmount ELSE 0 END as 'Balance' FROM rental NATURAL JOIN customer ORDER BY Name ASC",(error,res)=>{
    if(error){
      console.log(error);
  }
  else{
    console.log(res)
      response.send(res);
  }
  });
});
app.post("/balanceSearch",(request,response)=>{
  const Balance=request.body.Balance;
  const query=`SELECT  CustID,Name,CASE WHEN returned=0 THEN TotalAmount ELSE 0 END as 'Balance' FROM rental NATURAL JOIN customer WHERE TotalAmount=${Balance} ORDER BY Name ASC;`
  db.query(query,(error,res)=>{
    if(error){
      console.log(error);
  }
  else{
      response.send(res);
  }
  });
});       You, seconds ago • Uncommitted changes
app.post("/nameSearch",(request,response)=>{
  const Name=request.body.Name;
  const query=`SELECT distinct CustID,Name,CASE WHEN returned=0 THEN TotalAmount ELSE 0 END as 'Balance' FROM rental NATURAL JOIN customer WHERE Name LIKE '%${Name}%'ORDER BY Name ASC;`
  db.query(query,(error,res)=>{
    if(error){
      console.log(error);
  }
  else{
      response.send(res);
  }
  });
});
```

b) Front End gui for searching the vehicke by category by type stardate and return date.We have also



2019-05-01 → 2019-05-15

Truck

Basic

Ford F Series Pickup 2014

Basic Truck (D) (W)

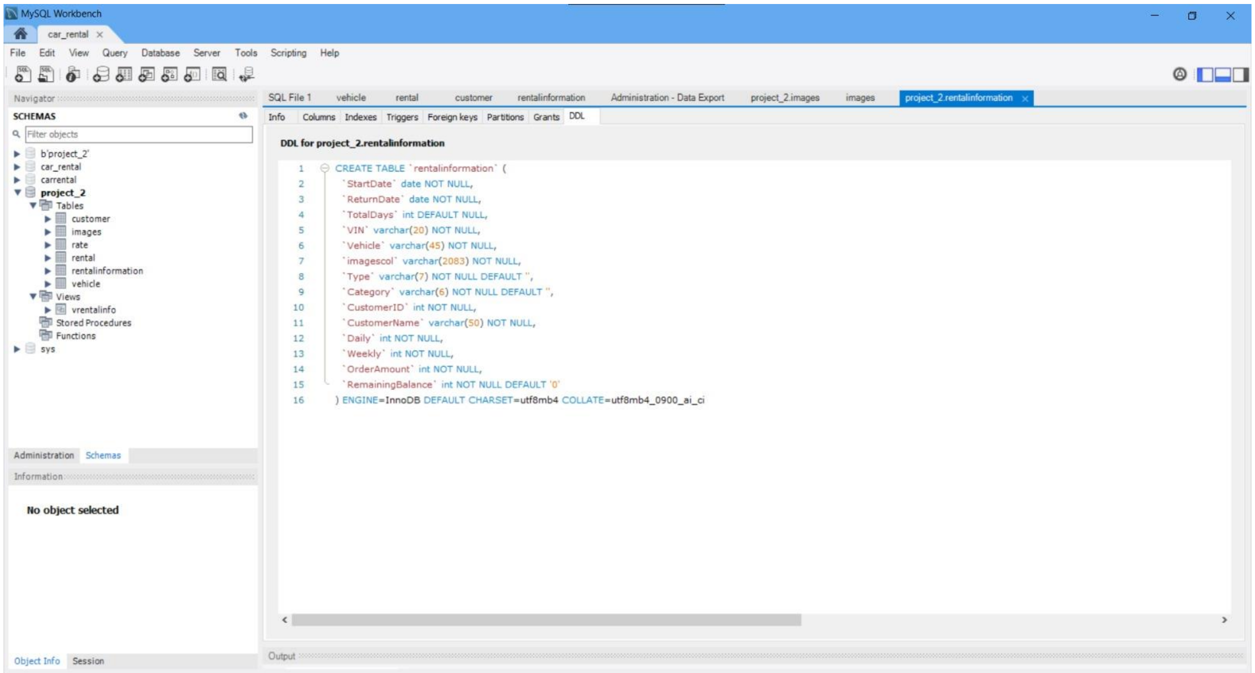
Changes in Original Database

Created an extra table images as follows:-

```
 CREATE TABLE `images` (

  `VehicleID` varchar(20) NOT NULL,

  `imagescol` varchar(2083) NOT NULL,

  KEY `VehicleID_idx` (`VehicleID`),

  CONSTRAINT `VehicleIDS` FOREIGN KEY (`VehicleID`) REFERENCES `vehicle` (`VehicleID`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci
```

- **Imported the data using import data wizard and took images from internet**
- **Have a foreign key relationship with VehicleID of Vehicle Table**

Created the table rentalinformation as follows:-



DDL for project_2.rentalinformation

```
CREATE TABLE `rentalinformation` (
  `StartDate` date NOT NULL,
  `ReturnDate` date NOT NULL,
  `TotalDays` int DEFAULT NULL,
  `VIN` varchar(20) NOT NULL,
  `Vehicle` varchar(45) NOT NULL,
  `imagescol` varchar(2083) NOT NULL,
  `Type` varchar(7) NOT NULL DEFAULT '',
  `Category` varchar(6) NOT NULL DEFAULT '',
  `CustomerID` int NOT NULL,
  `CustomerName` varchar(50) NOT NULL,
  `Daily` int NOT NULL,
  `Weekly` int NOT NULL,
  `OrderAmount` int NOT NULL,
  `RemainingBalance` int NOT NULL DEFAULT '0'
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci
```

This table is similar to view vRentalinfo but has natural join with rate table and images table to get the daily price information and images while getting the view in return a car webpage.

Contritions:

Task 1:- Aditya Chaudhari

Task2:-

Login and Sign up(Frontend and Backend) : Aditya Chaudhari

Dashboard (Frontend)-Aditya Chaudhari

Dashboard (Backend)- Utkarsh Singh

Vehicle Search (frontend and backend)- Utkarsh Singh

Vehicle Availability (frontend and backend)- Utkarsh Singh

Car Booking (frontend and backend)- Utkarsh Singh

Admin Home(Frontend)-Aditya Chaudhari

Admin Home(Backend)- Utkarsh Singh

Customer Log (Frontend and Backend)- Aditya Chaudhari

Customer Log (Frontend and Backend)- Aditya Chaudhari

Customer Search – Utkarsh Singh

Add Car (Frontend and Backend)- Aditya Chaudhari

Vehicle Return Page (Frontend and Backend)- Utkarsh Singh

Return Car Form (Frontend and Backend)- Aditya Chaudhari