**Car Rental Application**

**Team Members:**

**Rutuja Dukhande(1001730748)**

**Priyanka Patil(1001981851)**

**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.

**Task 1**

1. **Add an extra column ‘Returned’ to the RENTAL table. Values will be 0-for non-returned cars, and 1-for returned. Then update the ‘Returned’ column with '1' for all records that they have a payment date and with '0' for those that they do not have payment date**.

**Answer**: Query: ALTER TABLE rental

ADD COLUMN Returned INT(11)

UPDATE rental

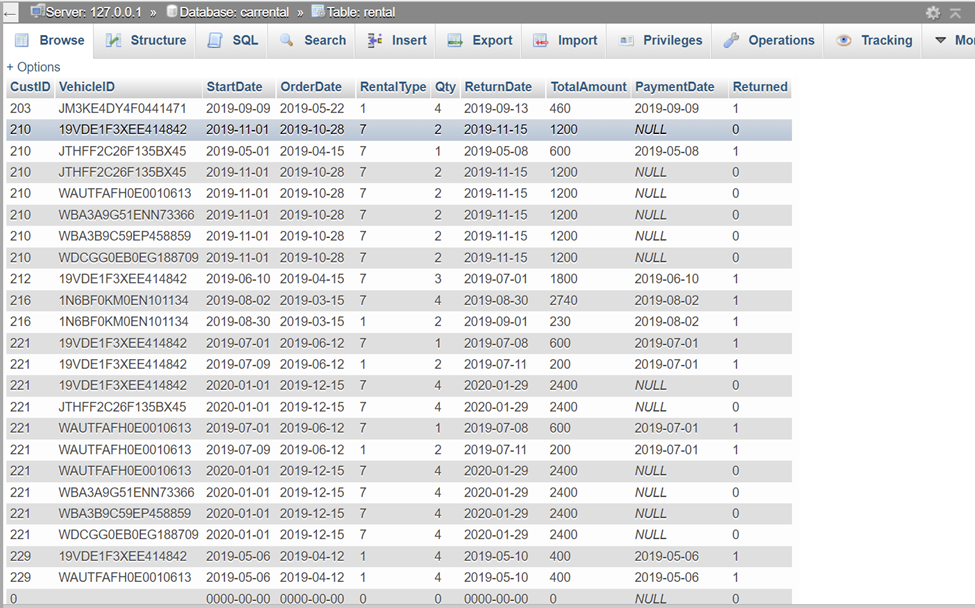
SET Returned =

( CASE WHEN rental.PaymentDate IS NOT NULL THEN 1

WHEN rental.PaymentDate IS NULL THEN 0

END)

**Screenshot:**

****

Graphical user interface, application

Description automatically generated

1. **Create a view vRentalInfo that retrieves all information per rental.**

**Answer:**

CREATE VIEW vRentalInfo AS

SELECT rental.OrderDate, rental.StartDate, rental.ReturnDate, rental.RentalType\*rental.Qty AS TotalDays, rental.TotalAmount AS OrderAmount,

CASE

WHEN rental.PaymentDate IS NULL THEN rental.TotalAmount

WHEN rental.PaymentDate IS NOT NULL THEN 0

END As PaymentDate, 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'

WHEN vehicle.Type = 6 THEN 'VAN'

END As Type,

CASE

WHEN vehicle.Category = 0 THEN 'Basic'

WHEN vehicle.Category = 1 THEN 'Luxury'

END As Category, customer.CustID AS CustomerID, customer.Name AS CustomerName

FROM rental, vehicle, customer

WHERE customer.CustID = rental.CustID AND rental.VehicleID = vehicle.VehicleID

ORDER BY rental.StartDate ASC

**Screenshot:**

Graphical user interface, text, application

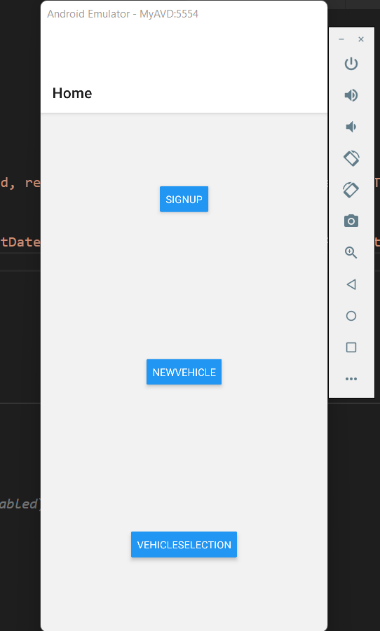
Description automatically generated

Graphical user interface

Description automatically generated

**Task 2**

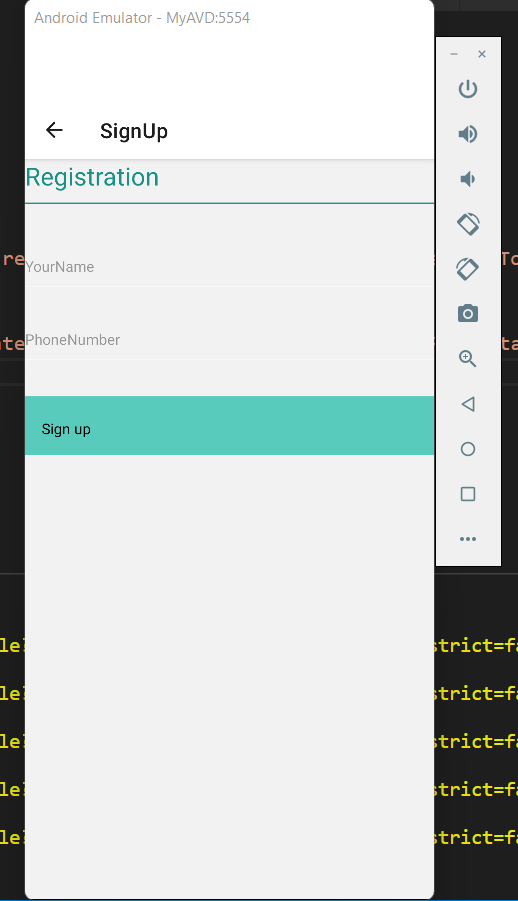
**UI**

****

1. **The first requirement is to add information about a new customer. Do not provide the customer ID in your query. Submit your editable SQL query that your code executes.**

**Answer:** INSERT INTO customer(Name, Phone) VALUES ('$custName','$custPhone');

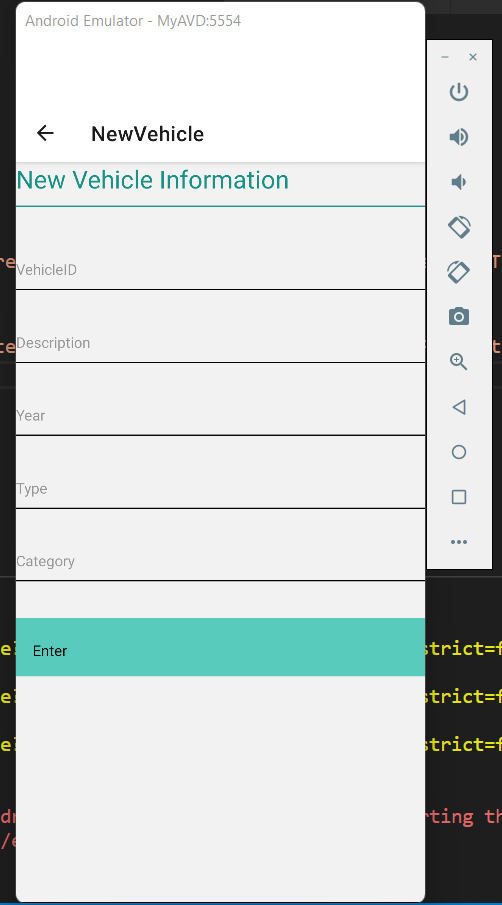
where $custName and $custPhone is user input.



1. **The second requirement is to add all the information about a new vehicle. Submit your editable SQL query that your code executes.**

**Answer:** INSERT INTO vehicle(VehicleID, Description, Year, Type, Category) VALUES('$vehicleId', '$vehicleDesc', '$vehicleYear', '$vehicleType', '$vehicleCat');

Where '$vehicleId', '$vehicleDesc', '$vehicleYear', '$vehicleType', '$vehicleCat' are User – Input



1. **The third requirement is to add all the information about a new rental reservation (this must find a free vehicle of the appropriate type and category for a specific rental period). We assume that the customer has the right either to pay at the order or return date. Submit your editable SQL queries (select available vehicles & insert rental) that your code executes.**

**Answer:**

***To select the free vehicle of the appropriate type and category entered by the User end:***

Query🡪 SELECT rental.VehicleID FROM rental JOIN vehicle on vehicle.VehicleID = rental.VehicleID

WHERE vehicle.Type = '".$Type."' AND vehicle.Category = '".$Category."' AND rental.Returned = 1 LIMIT 1;

***To Insert the Data of rented vehicle:***

Query🡪 INSERT INTO rental (CustID, VehicleID, StartDate, OrderDate, RentalType, Qty, ReturnDate) VALUES('".$custId."','".$row[0]."','".$startDate."','".$OrderDate."','".$RentalType."','".$Qty."','".$ReturnDate."');

***To update the TotalAmount, Returned, PaymentDate Column:***

Query🡪 UPDATE rental SET rental.TotalAmount=

CASE

WHEN $Type = 1 AND $Category = 0 AND rental.RentalType= 7 THEN rental.Qty\*480

WHEN $Type = 1 AND $Category = 0 AND rental.RentalType= 1 THEN rental.Qty\*80

WHEN $Type = 1 AND $Category = 1 AND rental.RentalType= 7 THEN rental.Qty\*600

WHEN $Type = 1 AND $Category = 1 AND rental.RentalType= 1 THEN rental.Qty\*105

WHEN $Type = 2 AND $Category = 0 AND rental.RentalType= 7 THEN rental.Qty\*530

WHEN $Type = 2 AND $Category = 0 AND rental.RentalType= 1 THEN rental.Qty\*90

WHEN $Type = 2 AND $Category = 1 AND rental.RentalType= 7 THEN rental.Qty\*660

WHEN $Type = 2 AND $Category = 1 AND rental.RentalType= 1 THEN rental.Qty\*116

WHEN $Type = 3 AND $Category = 0 AND rental.RentalType= 7 THEN rental.Qty\*600

WHEN $Type = 3 AND $Category = 0 AND rental.RentalType= 1 THEN rental.Qty\*100

WHEN $Type = 3 AND $Category = 1 AND rental.RentalType= 7 THEN rental.Qty\*710

WHEN $Type = 3 AND $Category = 1 AND rental.RentalType= 1 THEN rental.Qty\*126

WHEN $Type = 4 AND $Category = 0 AND rental.RentalType= 7 THEN rental.Qty\*685

WHEN $Type = 4 AND $Category = 0 AND rental.RentalType= 1 THEN rental.Qty\*115

WHEN $Type = 4 AND $Category = 1 AND rental.RentalType= 7 THEN rental.Qty\*800

WHEN $Type = 4 AND $Category = 1 AND rental.RentalType= 1 THEN rental.Qty\*142

WHEN $Type = 5 AND $Category = 0 AND rental.RentalType= 7 THEN rental.Qty\*780

WHEN $Type = 5 AND $Category = 0 AND rental.RentalType= 1 THEN rental.Qty\*130

WHEN $Type = 5 AND $Category = 1 AND rental.RentalType= 7 THEN rental.Qty\*685

WHEN $Type = 5 AND $Category = 1 AND rental.RentalType= 1 THEN rental.Qty\*115

WHEN $Type = 6 AND $Category = 0 AND rental.RentalType= 7 THEN rental.Qty\*900

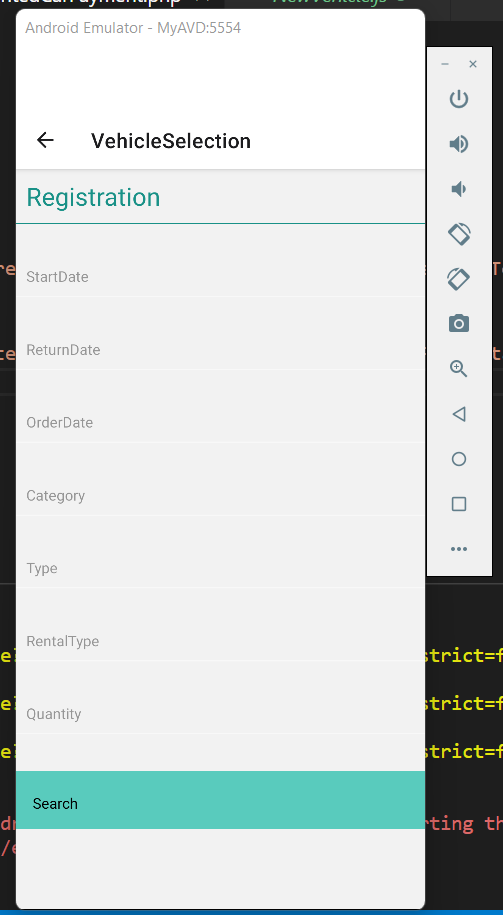
WHEN $Type = 6 AND $Category = 0 AND rental.RentalType= 1 THEN rental.Qty\*150

WHEN $Type = 6 AND $Category = 1 AND rental.RentalType= 7 THEN rental.Qty\*800

WHEN $Type = 6 AND $Category = 1 AND rental.RentalType= 1 THEN rental.Qty\*135 END

Query🡪 UPDATE rental SET rental.PaymentDate = '".$ReturnDate."', rental.Returned =1

Query🡪 UPDATE rental SET rental.PaymentDate = NULL, rental.Returned = 0



1. **The fourth requirement is to handle the return of a rented car. This transaction should print the total customer payment due for that rental, enter it in the database and update the returned attribute accordingly. You need to be able to retrieve a rental by the return date, customer name (the table needs the id), and vehicle info. Submit your editable SQL queries (retrieve & update rental) that your code executes**

**Answer:**

***To show payment due with car information***

Query🡪 "SELECT rental.CustID, rental.VehicleId, rental.StartDate,rental.ReturnDate, rental.TotalAmount FROM rental WHERE rental.CustID='".$custId."' AND rental.ReturnDat='".$ReturnDate."'";

***Query under Pay Button***

Query🡪 "UPDATE rental SET rental.PaymentDate='".$ReturnDate."',Returned='1' WHERE rental.CustID='".$custId."'";

1. **a)** **List for every customer the ID, name, and if there is any remaining balance. The user has the right to search either by a customer’s ID, name, part of the name, or to run the query with no filters/criteria. The amount needs to be in US dollars. For customers with zero (0) or NULL balance, you need to return zero dollars ($0.00). Make sure that your query returns meaningful attribute names. In the case that the user decides not to provide any filters, order the results based on the balance amount. Make sure that you return all records. Submit your editable SQL query that your code executes.**

**Answer:** CREATE VIEW custinfo2 AS

SELECT customer.CustID, customer.Name,

CASE

WHEN rental.TotalAmount IS NULL THEN CONCAT('$0.00')

WHEN rental.TotalAmount IS NOT NULL THEN CONCAT('$',rental.TotalAmount)

END AS PaymentDue

FROM customer, rental

WHERE customer.CustID=rental.CustID AND rental.CustID = “210”;

b) **List for every vehicle the VIN, the description, and the average DAILY price. The user has the right either to search by the VIN, vehicle’s description, part of the description, or to run the query with no filters/criteria. An example criterion would be all ‘BMW’ vehicles. The amount needs to be in US dollars. The average DAILY price derives from the rental table, and the amount needs to have two decimals as well as the dollar ‘$’ sign. For vehicles that they do not have any rentals, you need to substitute the NULL value with a ‘Non-Applicable’ text. Make sure that your query returns meaningful attribute names. In the case that the user decides not to provide any filters, order the results based on the average daily price. Submit your editable SQL query that your code executes.**

**Answer**:CREATE VIEW mytest4 AS

SELECT vehicle.VehicleID, vehicle.Description,

CASE

WHEN vehicle.VehicleID = rental.VehicleID THEN CONCAT('$',ROUND((rental.TotalAmount/(rental.Qty\*rental.RentalType)), 2))

WHEN vehicle.VehicleID != rental.VehicleID THEN CONCAT('NOT APPLICABLE')

END AS DailyPrice

FROM rental, vehicle

WHERE rental.VehicleID = vehicle.VehicleID