Project 2

Names: Patrick Arzoumanian, Spencer Whitehead, Nghia Lam

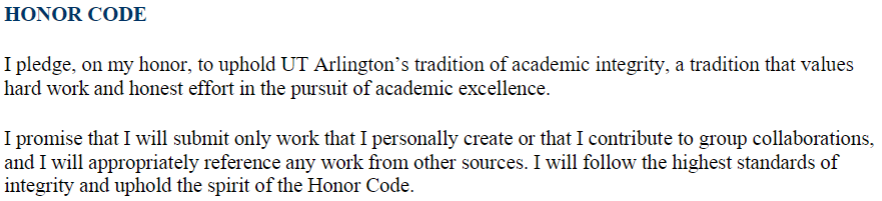
****

Table of Contents

[**Task 1**](#_3llgp49ihgqr) **3**

[Query 1:](#_hrglw0cw9lc9) 3

[Query 2:](#_mxgp81xm1r24) 4

[**Task 2:**](#_kz4shd84amtg) **5**

[Requirement 1:](#_7wbnqlsehmy) 5

[Requirement 2:](#_srzimqy5qgpe) 5

[Requirement 3:](#_woiqyltrzwsk) 5

[Requirement 4:](#_3h2qjuo2wjym) 6

[Requirement 5a:](#_s5ye5b1we1c6) 6

[Requirement 5b:](#_upmjete6y84q) 7

## Task 1

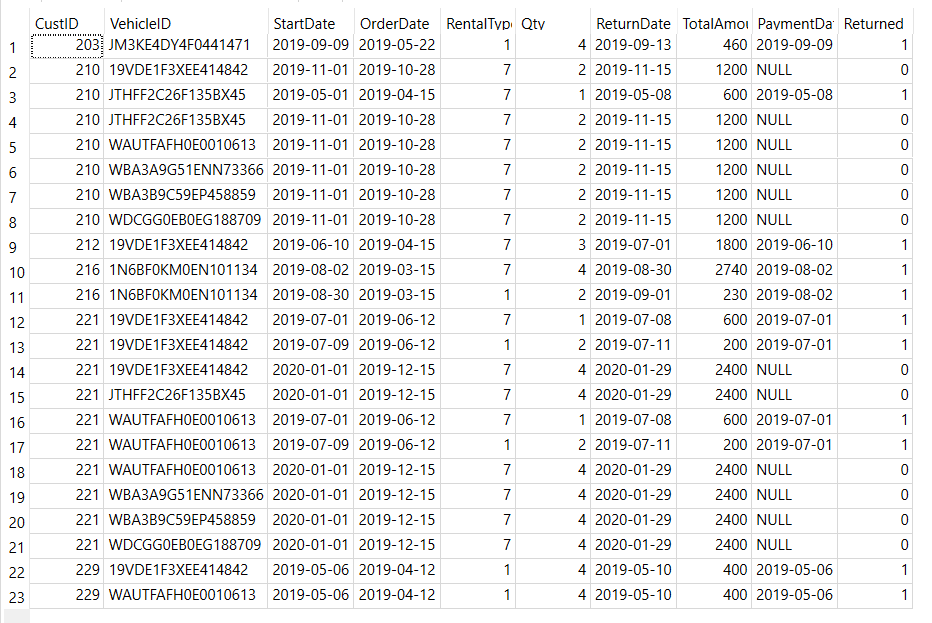
### Query 1:

ALTER TABLE RENTAL ADD Returned INT;

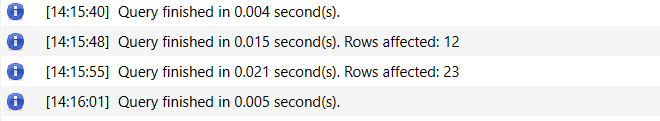
UPDATE RENTAL SET Returned = 0 WHERE PaymentDate = 'NULL';

UPDATE RENTAL SET Returned = 1 WHERE PaymentDate != 'NULL;

SELECT \* FROM RENTAL';



Action response:



### Query 2:

CREATE VIEW vRentalInfo AS

SELECT OrderDate, StartDate, ReturnDate, TotalDays, VehicleID as VIN, Description AS Vehicle,

CASE

WHEN Type = 1 THEN 'Compact'

WHEN Type = 2 THEN 'Medium'

WHEN Type = 3 THEN 'Large'

WHEN Type = 4 THEN 'SUV'

WHEN Type = 5 THEN 'Truck'

WHEN Type = 6 THEN 'VAN'

END Type,

CASE

WHEN Category = 1 THEN 'Luxury'

ELSE 'Basic'

END Category,

CustID AS CustomerID, Name AS CustomerName, TotalAmount AS OrderAmount, RentalBalance

FROM VEHICLE NATURAL JOIN CUSTOMER NATURAL JOIN RENTAL NATURAL JOIN

(SELECT VehicleID, StartDate, JULIANDAY(ReturnDate) - JULIANDAY(StartDate) AS TotalDays

FROM RENTAL) NATURAL JOIN

(SELECT VehicleID, StartDate,

CASE

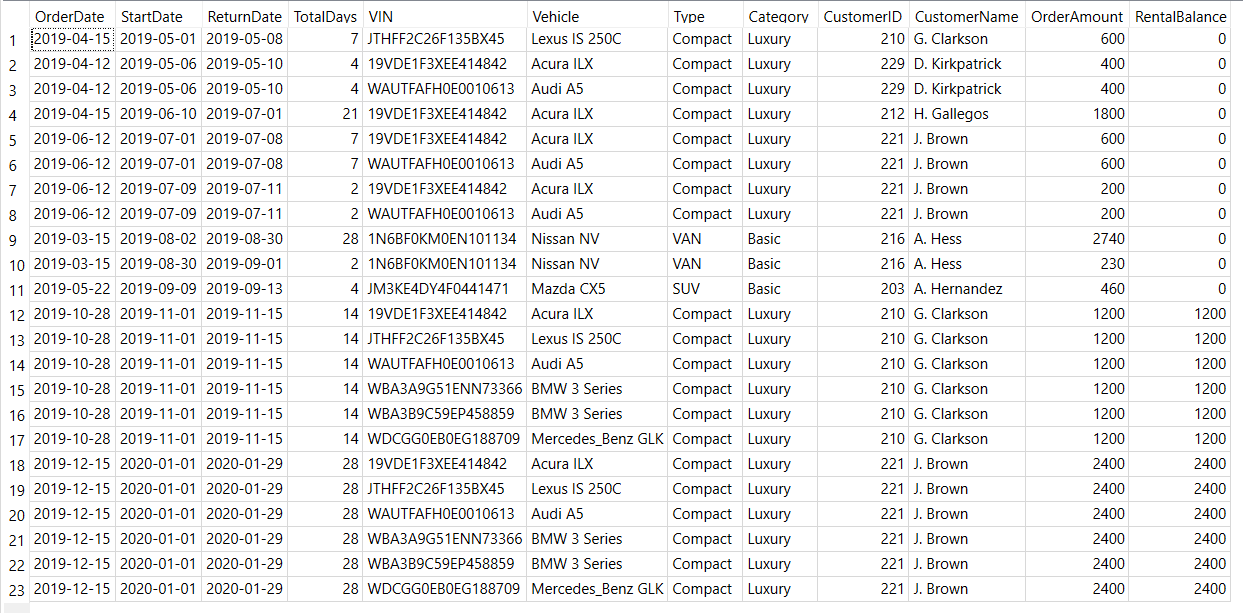
WHEN PaymentDate = 'NULL' THEN TotalAmount

ELSE 0

END RentalBalance

FROM RENTAL)

ORDER BY StartDate;  
  
SELECT \* FROM vRentalInfo;



Rows returned: 23

Action response:



## Task 2:

### Requirement 1:

INSERT INTO CUSTOMER(Name, Phone) VALUES (:name, :phone)

### Requirement 2:

INSERT INTO VEHICLE VALUES (:VehicleID, :Description, :Year, :Category, :Type )

### Requirement 3:

Query for retrieving vehicleID:

SELECT VehicleID

FROM VEHICLE

WHERE Type = :Type AND Category = :Category AND VehicleID NOT IN

(SELECT VehicleID

FROM RENTAL

WHERE StartDate BETWEEN :StartDate AND :ReturnDate OR

ReturnDate BETWEEN :StartDate AND :ReturnDate)",

Queries for getting rate:

SELECT Weekly FROM RATE WHERE Type = :Type AND Category = :Category

SELECT Daily FROM RATE WHERE Type = :Type AND Category = :Category

Query for adding rental:

INSERT INTO RENTAL VALUES (:CustID, :VehicleID, :StartDate, :OrderDate, :RentalType, :Qty, :ReturnDate, :TotalAmount, :PaymentDate, :Returned )

### Requirement 4:

Query for getting custID:

SELECT CustID FROM CUSTOMER WHERE Name = ?

Query for getting totalAmount:

SELECT TotalAmount FROM RENTAL WHERE CustID = ? AND VehicleID = ? AND ReturnDate = ?

Query for setting rental to paid:

UPDATE RENTAL SET Returned = 1, PaymentDate = ? WHERE CustID = ? AND VehicleID = ? AND ReturnDate = ?

### Requirement 5a:

Query for searching by custID:

SELECT \*

FROM (SELECT CustID, Name, '$' ||

SUM(CASE WHEN Returned = 0 THEN TotalAmount ELSE 0 END) || '.00' Balance

FROM CUSTOMER C NATURAL JOIN RENTAL R

GROUP BY CustID

UNION

SELECT CustID, Name, '$0.00' Balance

FROM CUSTOMER

WHERE CustID NOT IN

(SELECT CustID

FROM CUSTOMER NATURAL JOIN RENTAL

GROUP BY CustID)

GROUP BY CustID)

WHERE CustID = :CustID

Query for searching by name, or list all:

SELECT \*

FROM (SELECT CustID, Name, '$' ||

SUM(CASE WHEN Returned = 0 THEN TotalAmount ELSE 0 END) || '.00' Balance

FROM CUSTOMER C NATURAL JOIN RENTAL R

GROUP BY CustID

UNION

SELECT CustID, Name, '$0.00' Balance

FROM CUSTOMER

WHERE CustID NOT IN

(SELECT CustID

FROM CUSTOMER NATURAL JOIN RENTAL

GROUP BY CustID)

GROUP BY CustID)

WHERE Name LIKE ?

### Requirement 5b:

Query for searching by vehicleID:

SELECT \*

FROM (SELECT VehicleID, Description, '$' || ROUND(AVG(TotalAmount / (RentalType \* Qty)), 1)

|| '0' DailyRate

FROM VEHICLE NATURAL JOIN RENTAL

GROUP BY VehicleID

UNION

SELECT VehicleID, Description, 'Non-Applicable'

FROM VEHICLE

WHERE VehicleID NOT IN (SELECT VehicleID FROM RENTAL))

WHERE VehicleID = :VehicleID

GROUP BY VehicleID

Query for searching by description, or list all:

SELECT \*

FROM (SELECT VehicleID, Description, '$' || ROUND(AVG(TotalAmount / (RentalType \* Qty)), 1)

|| '0' DailyRate

FROM VEHICLE NATURAL JOIN RENTAL

GROUP BY VehicleID

UNION

SELECT VehicleID, Description, 'Non-Applicable'

FROM VEHICLE

WHERE VehicleID NOT IN (SELECT VehicleID FROM RENTAL))

WHERE Description LIKE ?

GROUP BY VehicleID