Car Rental Project

Phase 2

CSE 5330 - 04 Database Systems

Team Number: 12

Team members:

Name	UTA ID	NetID
Sri Harish Pinnimti	1001865949	sxp5949
Yogesh Kalapala	1001879640	yxk9640

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.

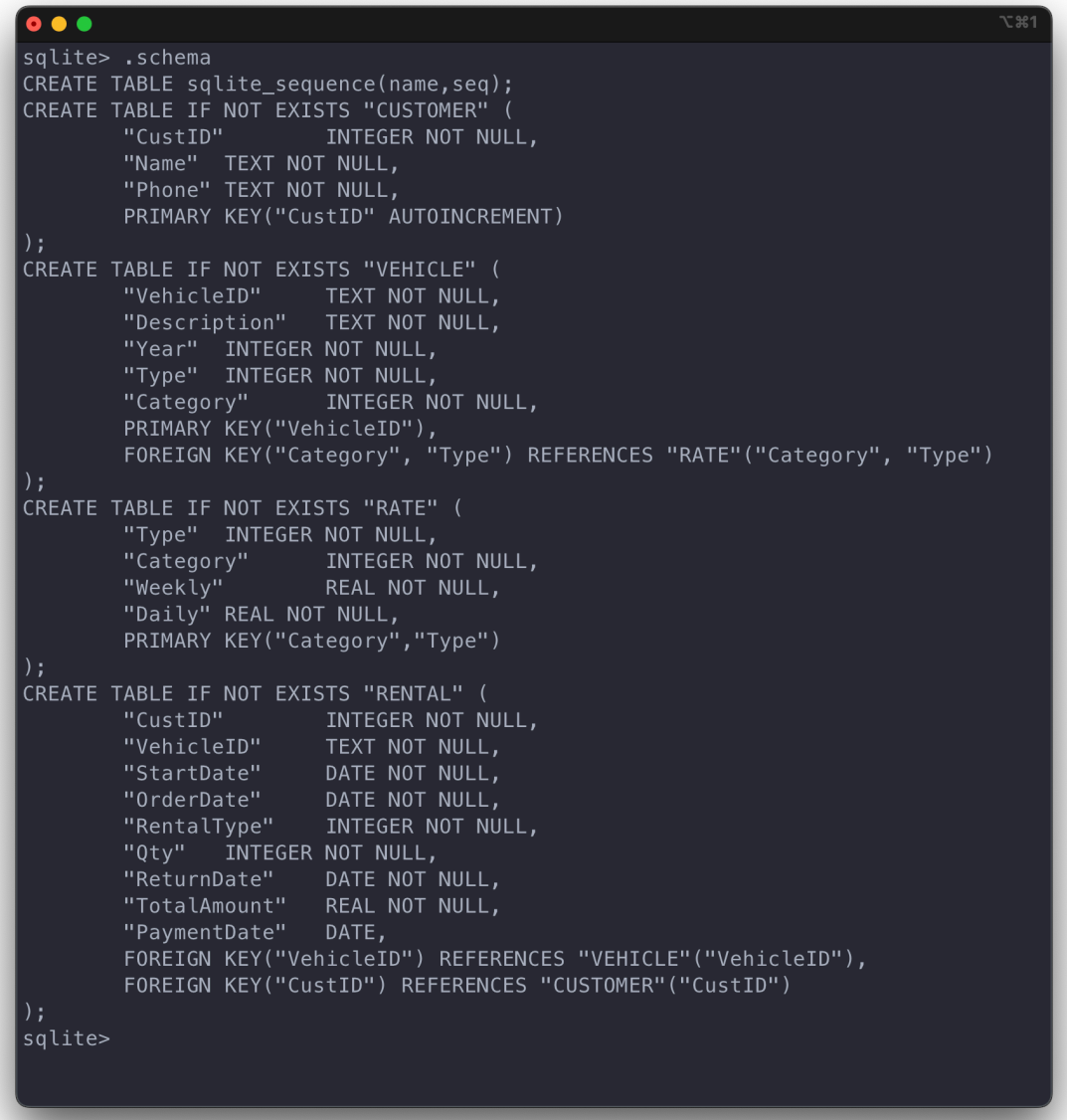
Task 1

Create the following 4 tables for the CarRental database.

Commands to create tables:

```
CREATE TABLE IF NOT EXISTS "CUSTOMER" (
     "CustID"   INTEGER NOT NULL,
     "Name"     TEXT NOT NULL,
     "Phone"    TEXT NOT NULL,
     PRIMARY KEY("CustID" AUTOINCREMENT)
);
CREATE TABLE IF NOT EXISTS "VEHICLE" (
     "VehicleID"     TEXT NOT NULL,
     "Description"   TEXT NOT NULL,
     "Year"     INTEGER NOT NULL,
     "Type"     INTEGER NOT NULL,
     "Category" INTEGER NOT NULL,
     PRIMARY KEY("VehicleID"),
     FOREIGN KEY("Category", "Type") REFERENCES "RATE"("Category",
"Type")
);
CREATE TABLE IF NOT EXISTS "RATE" (
     "Type"     INTEGER NOT NULL,
     "Category" INTEGER NOT NULL,
     "Weekly"   REAL NOT NULL,
     "Daily"    REAL NOT NULL,
     PRIMARY KEY("Category","Type")
);
CREATE TABLE IF NOT EXISTS "RENTAL" (
     "CustID"   INTEGER NOT NULL,
     "VehicleID"     TEXT NOT NULL,
     "StartDate"     DATE NOT NULL,
     "OrderDate"     DATE NOT NULL,
     "RentalType"    INTEGER NOT NULL,
     "Qty" INTEGER NOT NULL,
     "ReturnDate"    DATE NOT NULL,
     "TotalAmount"   REAL NOT NULL,
```

```
    "PaymentDate"   DATE,
    FOREIGN KEY("VehicleID") REFERENCES "VEHICLE"("VehicleID"),
    FOREIGN KEY("CustID") REFERENCES "CUSTOMER"("CustID")
);
```



```
sqlite> .schema
CREATE TABLE sqlite_sequence(name,seq);
CREATE TABLE IF NOT EXISTS "CUSTOMER" (
        "CustID"        INTEGER NOT NULL,
        "Name"  TEXT NOT NULL,
        "Phone" TEXT NOT NULL,
        PRIMARY KEY("CustID" AUTOINCREMENT)
);
CREATE TABLE IF NOT EXISTS "VEHICLE" (
        "VehicleID"     TEXT NOT NULL,
        "Description"   TEXT NOT NULL,
        "Year"  INTEGER NOT NULL,
        "Type"  INTEGER NOT NULL,
        "Category"      INTEGER NOT NULL,
        PRIMARY KEY("VehicleID"),
        FOREIGN KEY("Category", "Type") REFERENCES "RATE"("Category", "Type")
);
CREATE TABLE IF NOT EXISTS "RATE" (
        "Type"  INTEGER NOT NULL,
        "Category"      INTEGER NOT NULL,
        "Weekly"        REAL NOT NULL,
        "Daily" REAL NOT NULL,
        PRIMARY KEY("Category","Type")
);
CREATE TABLE IF NOT EXISTS "RENTAL" (
        "CustID"        INTEGER NOT NULL,
        "VehicleID"     TEXT NOT NULL,
        "StartDate"     DATE NOT NULL,
        "OrderDate"     DATE NOT NULL,
        "RentalType"    INTEGER NOT NULL,
        "Qty"   INTEGER NOT NULL,
        "ReturnDate"    DATE NOT NULL,
        "TotalAmount"   REAL NOT NULL,
        "PaymentDate"   DATE,
        FOREIGN KEY("VehicleID") REFERENCES "VEHICLE"("VehicleID"),
        FOREIGN KEY("CustID") REFERENCES "CUSTOMER"("CustID")
);
sqlite>
```

“CustID” in CUSTOMER is made “AUTOINCREMENT” due to Task 3 Q1.
“Daily” and “Weekly” in RATE and “TotalAmount” in RENTAL are made “REAL” due to Task 3 Q3.

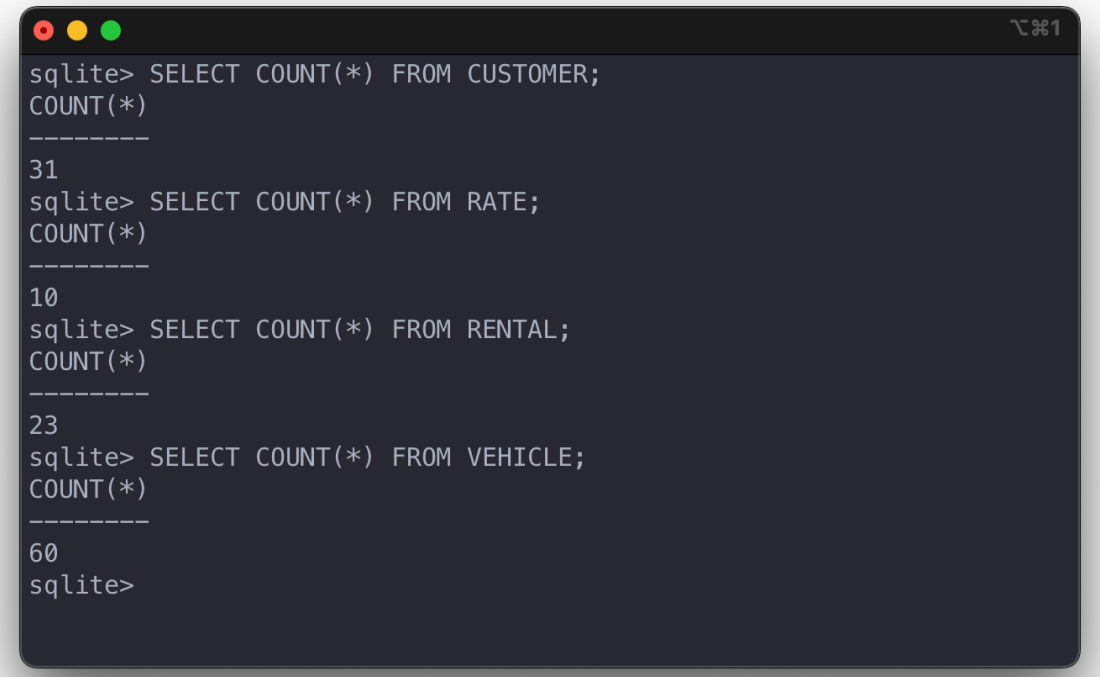
Task 2

Load the data from the text files into the corresponding tables.

For importing CSV data, we have used the “Import from CSV” option in “DB Browser For SQLite”

Commands to show the total number of records per table:

```
SELECT COUNT(*) FROM CUSTOMER;
SELECT COUNT(*) FROM RATE;
SELECT COUNT(*) FROM RENTAL;
SELECT COUNT(*) FROM VEHICLE;
```

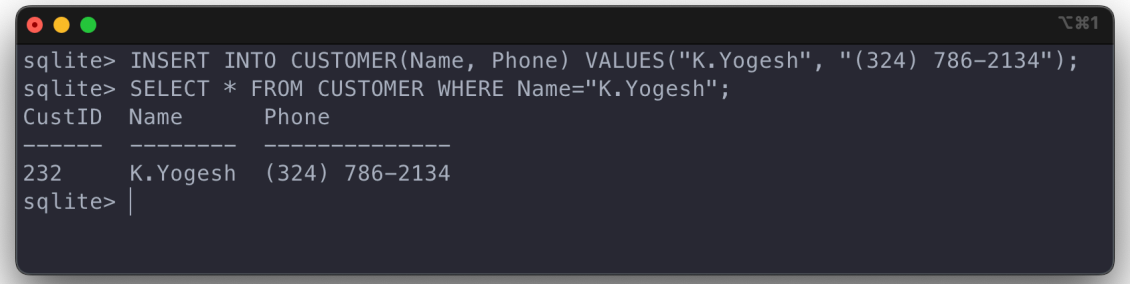


Task 3

Execute the following queries on the database tables:

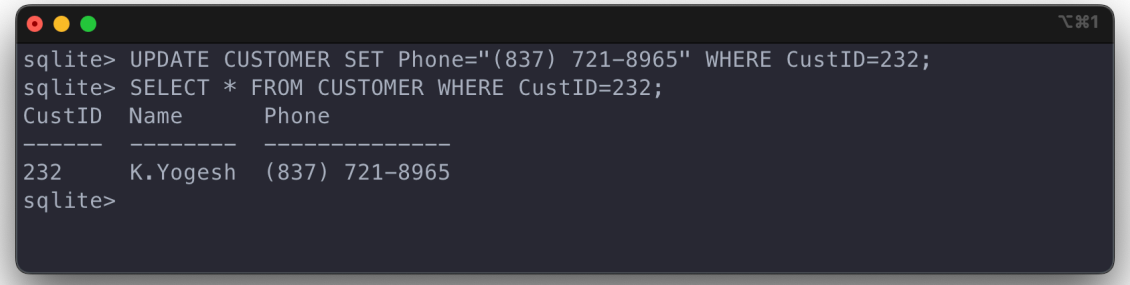
Question 1: Insert yourself as a New Customer. Do not provide the CustomerID in your query

```
INSERT INTO CUSTOMER(Name, Phone) VALUES("K.Yogesh", "(324)
786-2134”);
```



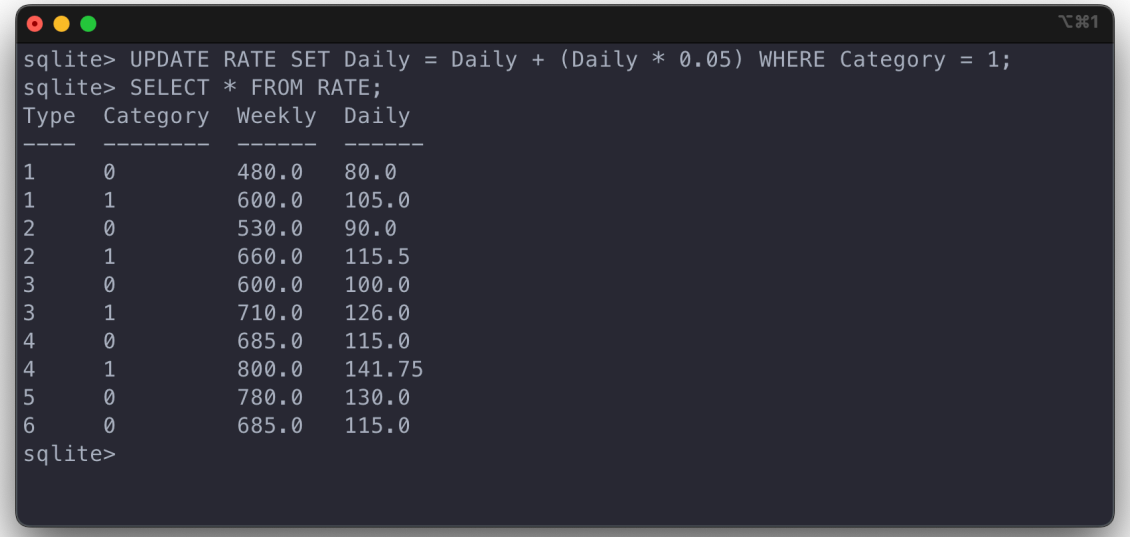
Question 2: Update your phone number to (837) 721-8965

```
UPDATE CUSTOMER SET Phone="(837) 721-8965" WHERE CustID=232;
```



Question 3: Increase only daily rates for luxury vehicles by 5%

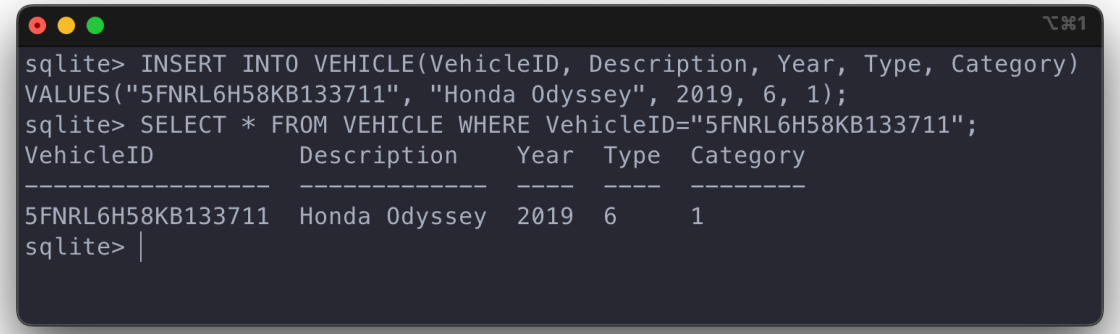
```
UPDATE RATE SET Daily = Daily + (Daily * 0.05) WHERE Category = 1;
```



```
sqlite> UPDATE RATE SET Daily = Daily + (Daily * 0.05) WHERE Category = 1;
sqlite> SELECT * FROM RATE;
Type  Category  Weekly  Daily
----  --------  ------  ------
1     0         480.0   80.0
1     1         600.0   105.0
2     0         530.0   90.0
2     1         660.0   115.5
3     0         600.0   100.0
3     1         710.0   126.0
4     0         685.0   115.0
4     1         800.0   141.75
5     0         780.0   130.0
6     0         685.0   115.0
sqlite>
```

Question 4-a: Insert a new luxury van with the following info: Honda Odyssey 2019, vehicle id: 5FNRL6H58KB133711

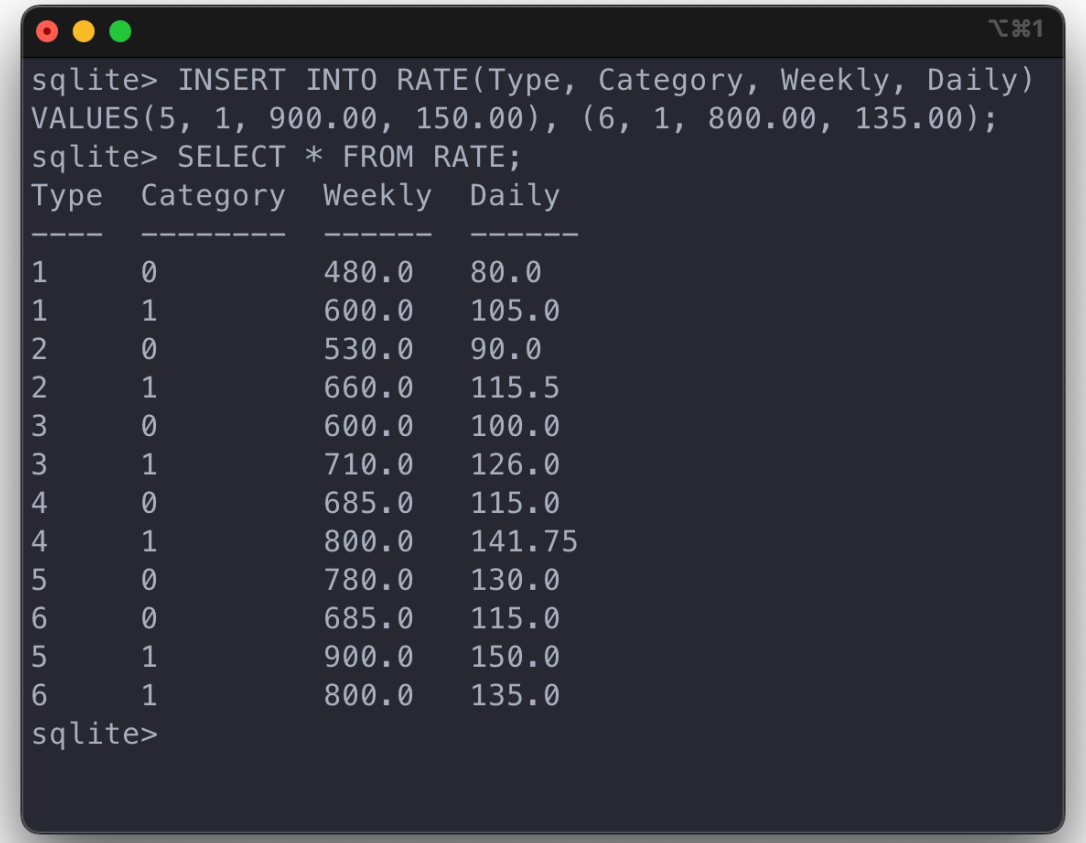
```
INSERT INTO VEHICLE(VehicleID, Description, Year, Type, Category)
VALUES("5FNRL6H58KB133711", "Honda Odyssey", 2019, 6, 1);
```



```
sqlite> INSERT INTO VEHICLE(VehicleID, Description, Year, Type, Category)
VALUES("5FNRL6H58KB133711", "Honda Odyssey", 2019, 6, 1);
sqlite> SELECT * FROM VEHICLE WHERE VehicleID="5FNRL6H58KB133711";
VehicleID          Description    Year  Type  Category
-----------------  -------------  ----  ----  --------
5FNRL6H58KB133711  Honda Odyssey  2019  6     1
sqlite>
```

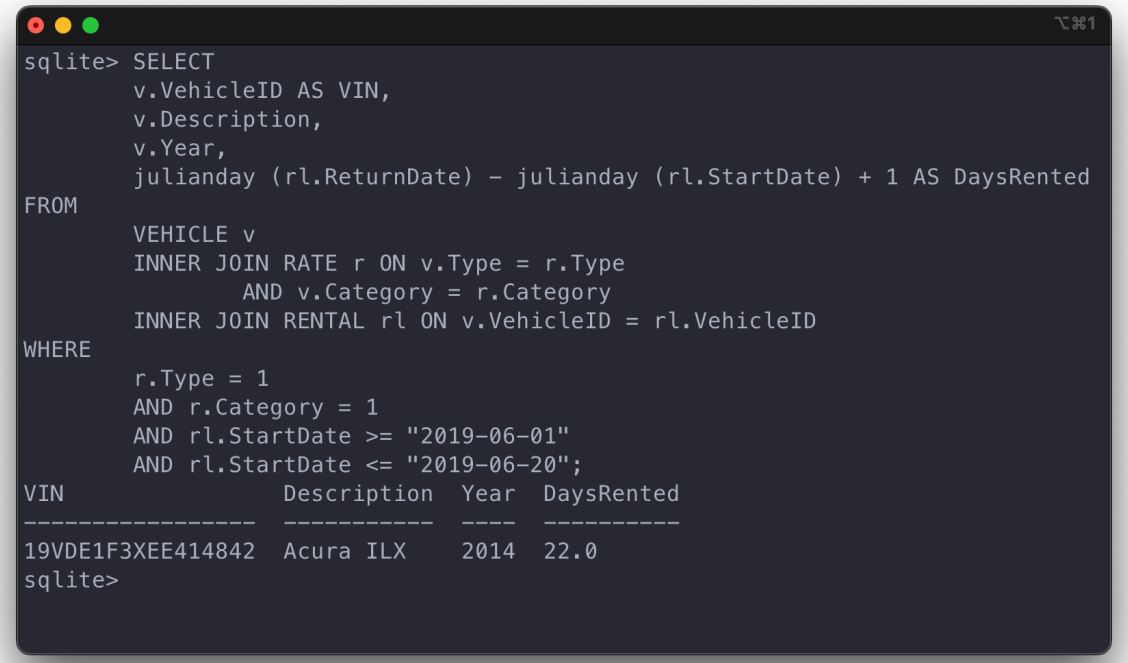
Question 4-b: You also need to insert the following rates:

```
INSERT INTO RATE (TYPE, Category, Weekly, Daily)
          VALUES(5, 1, 900.00, 150.00), (6, 1, 800.00, 135.00);
```



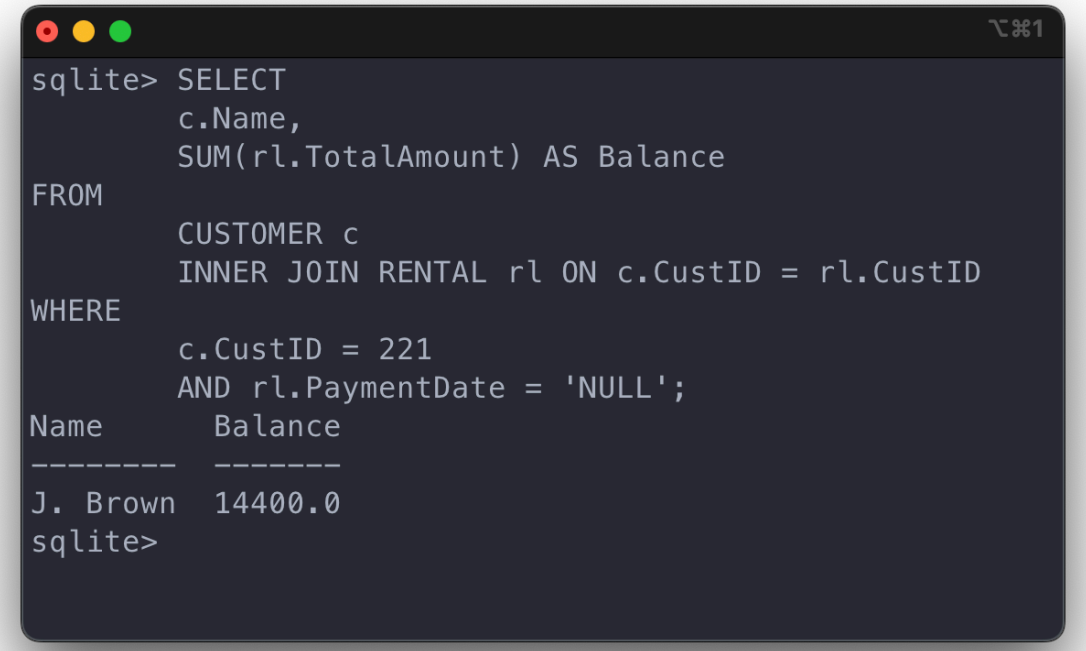
Question 5: Return all Compact(1) & Luxury(1) vehicles that were available for rent from June 01, 2019 until June 20, 2019. List VechicleID as VIN, Description, year, and how many days have been rented so far. You need to change the weeks into days

```
SELECT
      v.VehicleID AS VIN,
      v.Description,
      v.Year,
      julianday (rl.ReturnDate) - julianday (rl.StartDate) + 1 AS
DaysRented
FROM
      VEHICLE v
      INNER JOIN RATE r ON v.Type = r.Type
            AND v.Category = r.Category
      INNER JOIN RENTAL rl ON v.VehicleID = rl.VehicleID
WHERE
      r.Type = 1
      AND r.Category = 1
      AND rl.StartDate ⩾ "2019-06-01"
      AND rl.StartDate ⩽ “2019-06-20";
```



Question 6: Return a list with the remaining balance for the customer with the id ‘221’. List customer name, and the balance

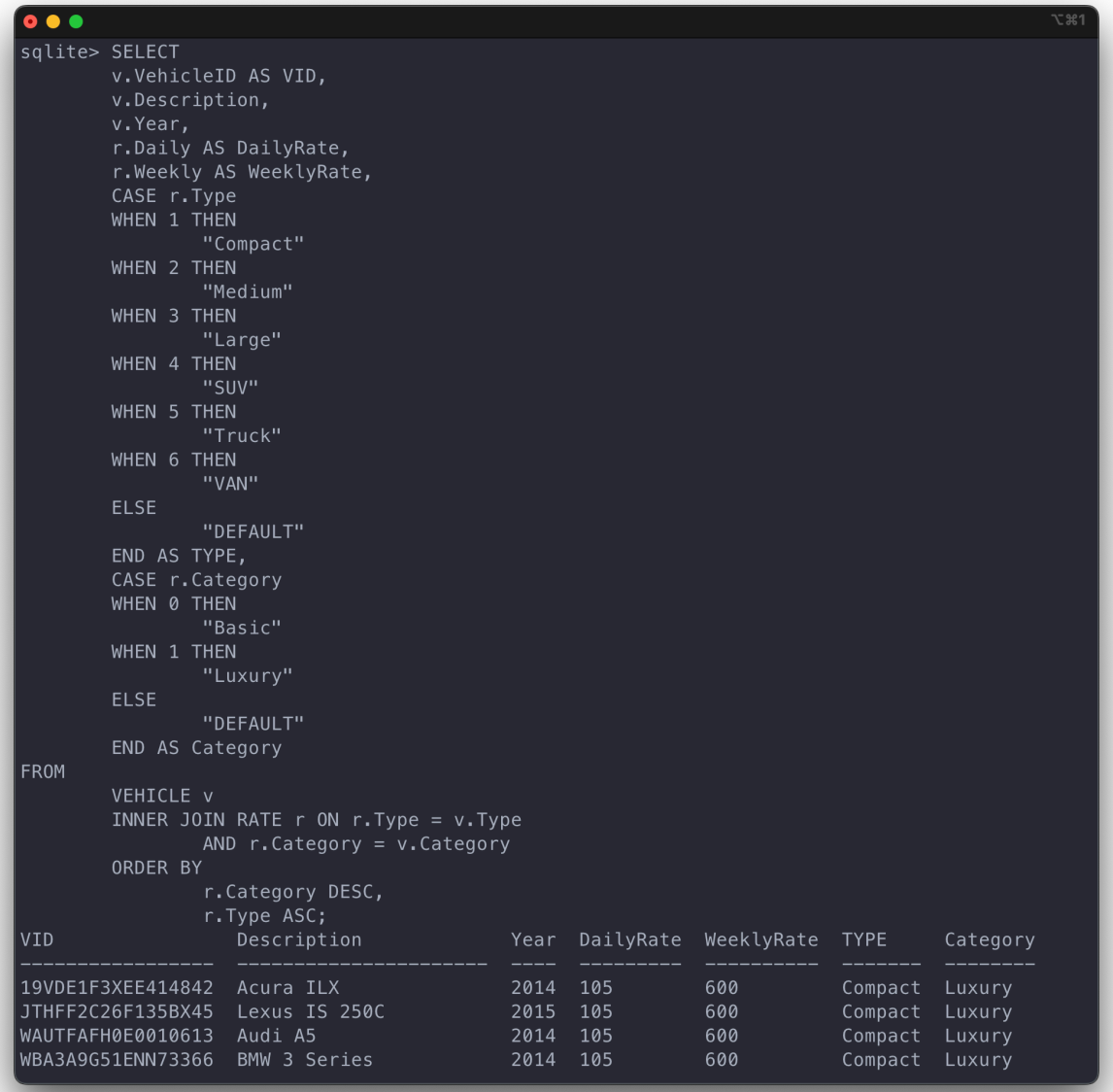
```
SELECT
      c.Name,
      SUM(rl.TotalAmount) AS Balance
FROM
      CUSTOMER c
      INNER JOIN RENTAL rl ON c.CustID = rl.CustID
WHERE
      c.CustID = 221
      AND rl.PaymentDate = ‘NULL';
```



Question 7: Create a report that will return all vehicles. List the VehicleID as VIN, Description, Year, Type, Category, and Weekly and Daily rates. For the vehicle Type and Category, you need to use the SQL Case statement to substitute the numbers with text. Order your results based on Category (first Luxury and then Basic) and Type based on the Type number, not the text

```
SELECT
     v.VehicleID AS VID,
     v.Description,
     v.Year,
     r.Daily AS DailyRate,
     r.Weekly AS WeeklyRate,
     CASE r.Type
     WHEN 1 THEN
          "Compact"
     WHEN 2 THEN
          "Medium"
     WHEN 3 THEN
          "Large"
     WHEN 4 THEN
          "SUV"
     WHEN 5 THEN
          "Truck"
     WHEN 6 THEN
          "VAN"
     ELSE
          "DEFAULT"
     END AS TYPE,
     CASE r.Category
     WHEN 0 THEN
          "Basic"
     WHEN 1 THEN
          "Luxury"
     ELSE
          "DEFAULT"
     END AS Category
FROM
     VEHICLE v
     INNER JOIN RATE r ON r.Type = v.Type
          AND r.Category = v.Category
```

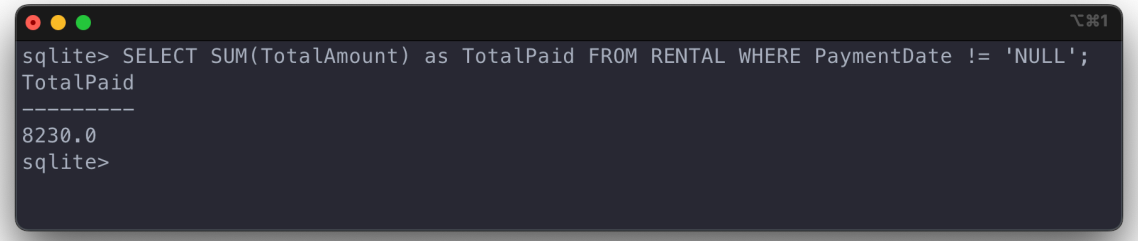
```
    ORDER BY
          r.Category DESC,
          r.Type ASC;
```



```
sqlite> SELECT
        v.VehicleID AS VID,
        v.Description,
        v.Year,
        r.Daily AS DailyRate,
        r.Weekly AS WeeklyRate,
        CASE r.Type
        WHEN 1 THEN
                "Compact"
        WHEN 2 THEN
                "Medium"
        WHEN 3 THEN
                "Large"
        WHEN 4 THEN
                "SUV"
        WHEN 5 THEN
                "Truck"
        WHEN 6 THEN
                "VAN"
        ELSE
                "DEFAULT"
        END AS TYPE,
        CASE r.Category
        WHEN 0 THEN
                "Basic"
        WHEN 1 THEN
                "Luxury"
        ELSE
                "DEFAULT"
        END AS Category
FROM
        VEHICLE v
        INNER JOIN RATE r ON r.Type = v.Type
                AND r.Category = v.Category
        ORDER BY
                r.Category DESC,
                r.Type ASC;
VID                 Description             Year  DailyRate  WeeklyRate  TYPE     Category
------------------  ----------------------  ----  ---------  ----------  -------  --------
19VDE1F3XEE414842   Acura ILX               2014  105        600         Compact  Luxury
JTHFF2C26F135BX45   Lexus IS 250C           2015  105        600         Compact  Luxury
WAUTFAFH0E0010613   Audi A5                 2014  105        600         Compact  Luxury
WBA3A9G51ENN73366   BMW 3 Series            2014  105        600         Compact  Luxury
```

Question 8: What is the total of money that customers paid to us until today?

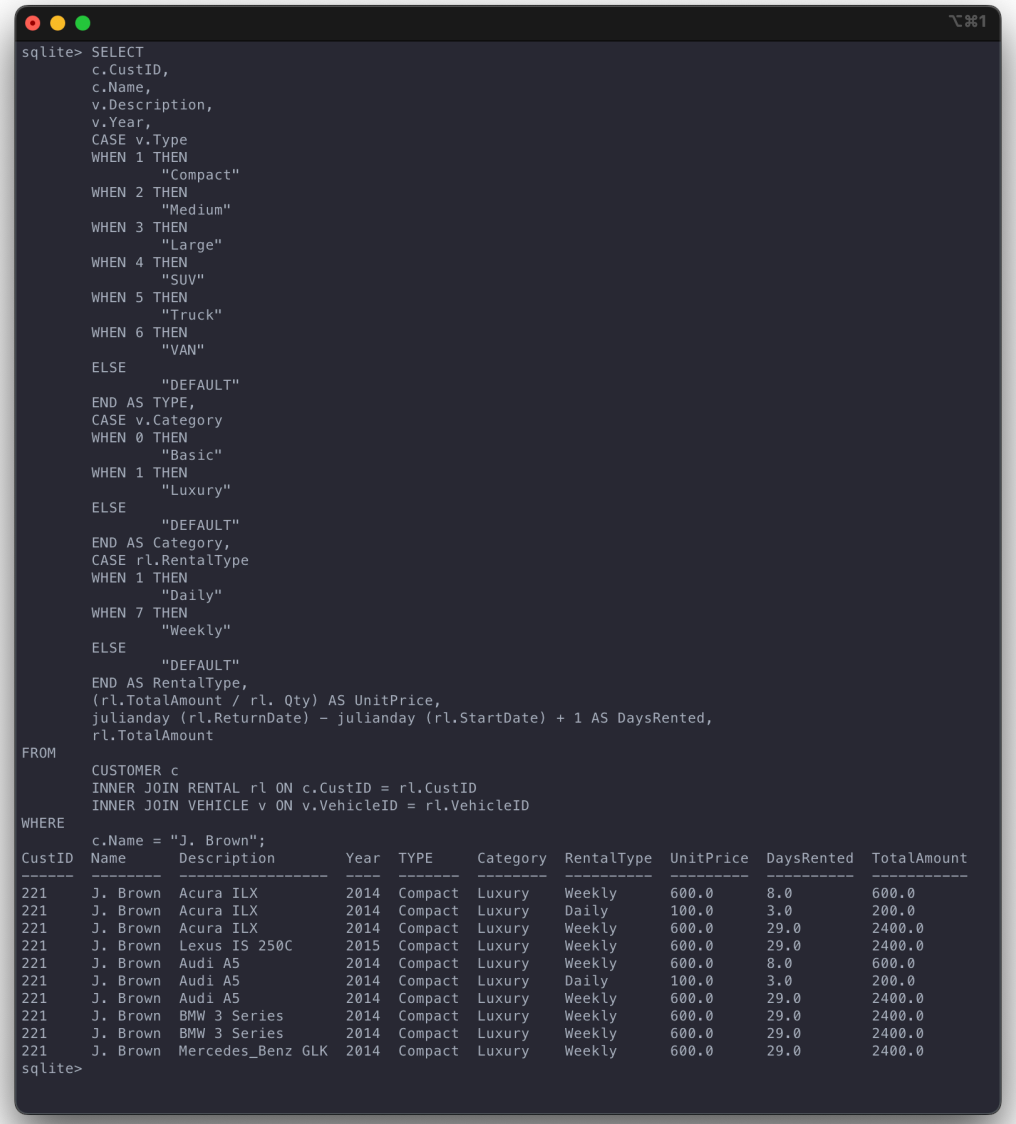
```
SELECT SUM(TotalAmount) as TotalPaid FROM RENTAL WHERE PaymentDate ≠
‘NULL';
```



Question 9-a: Create a report for the J. Brown customer with all vehicles he rented. List the description, year, type, and category. Also, calculate the unit price for every rental, the total duration mention if it is on weeks or days, the total amount, and if there is any payment. Similarly, as in Question 7, you need to change the numeric values to the corresponding text. Order the results by the StartDate.

```
SELECT
     c.CustID,
     c.Name,
     v.Description,
     v.Year,
     CASE v.Type
     WHEN 1 THEN
          "Compact"
     WHEN 2 THEN
          "Medium"
     WHEN 3 THEN
          "Large"
     WHEN 4 THEN
          "SUV"
     WHEN 5 THEN
          "Truck"
     WHEN 6 THEN
          "VAN"
     ELSE
          "DEFAULT"
     END AS TYPE,
     CASE v.Category
     WHEN 0 THEN
          "Basic"
     WHEN 1 THEN
          "Luxury"
     ELSE
          "DEFAULT"
     END AS Category,
     CASE rl.RentalType
     WHEN 1 THEN
          "Daily"
     WHEN 7 THEN
          "Weekly"
```

```
      ELSE
            "DEFAULT"
      END AS RentalType,
      (rl.TotalAmount / rl. Qty) AS UnitPrice,
      julianday (rl.ReturnDate) - julianday (rl.StartDate) + 1 AS
DaysRented,
      rl.TotalAmount
FROM
      CUSTOMER c
      INNER JOIN RENTAL rl ON c.CustID = rl.CustID
      INNER JOIN VEHICLE v ON v.VehicleID = rl.VehicleID
WHERE
      c.Name = "J. Brown”;
```

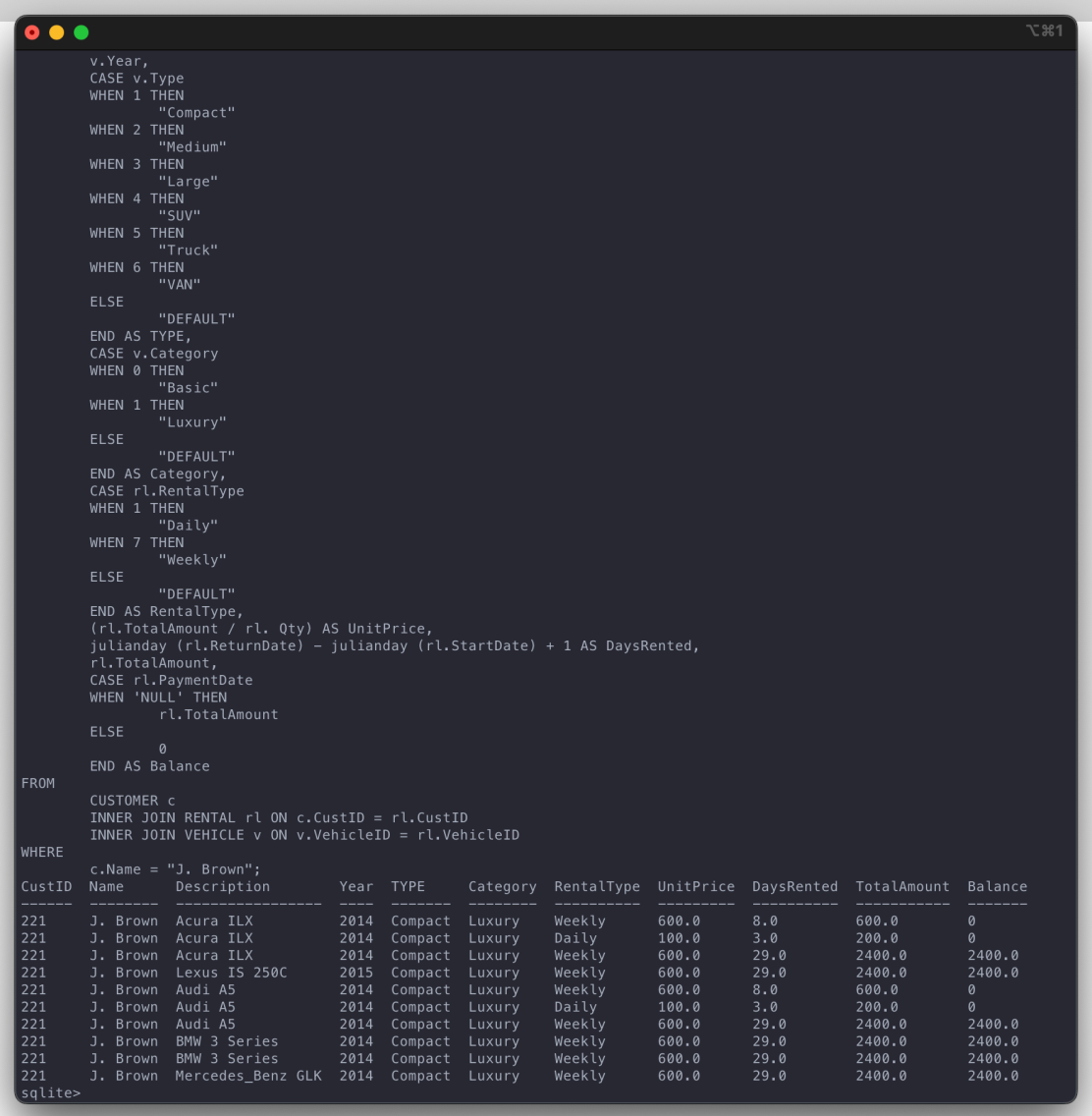


```
sqlite> SELECT
        c.CustID,
        c.Name,
        v.Description,
        v.Year,
        CASE v.Type
        WHEN 1 THEN
                "Compact"
        WHEN 2 THEN
                "Medium"
        WHEN 3 THEN
                "Large"
        WHEN 4 THEN
                "SUV"
        WHEN 5 THEN
                "Truck"
        WHEN 6 THEN
                "VAN"
        ELSE
                "DEFAULT"
        END AS TYPE,
        CASE v.Category
        WHEN 0 THEN
                "Basic"
        WHEN 1 THEN
                "Luxury"
        ELSE
                "DEFAULT"
        END AS Category,
        CASE rl.RentalType
        WHEN 1 THEN
                "Daily"
        WHEN 7 THEN
                "Weekly"
        ELSE
                "DEFAULT"
        END AS RentalType,
        (rl.TotalAmount / rl. Qty) AS UnitPrice,
        julianday (rl.ReturnDate) - julianday (rl.StartDate) + 1 AS DaysRented,
        rl.TotalAmount
FROM
        CUSTOMER c
        INNER JOIN RENTAL rl ON c.CustID = rl.CustID
        INNER JOIN VEHICLE v ON v.VehicleID = rl.VehicleID
WHERE
        c.Name = "J. Brown";
CustID  Name      Description        Year  TYPE     Category  RentalType  UnitPrice  DaysRented  TotalAmount
------  --------  -----------------  ----  -------  --------  ----------  ---------  ----------  -----------
221     J. Brown  Acura ILX          2014  Compact  Luxury    Weekly      600.0      8.0         600.0
221     J. Brown  Acura ILX          2014  Compact  Luxury    Daily       100.0      3.0         200.0
221     J. Brown  Acura ILX          2014  Compact  Luxury    Weekly      600.0      29.0        2400.0
221     J. Brown  Lexus IS 250C      2015  Compact  Luxury    Weekly      600.0      29.0        2400.0
221     J. Brown  Audi A5            2014  Compact  Luxury    Weekly      600.0      8.0         600.0
221     J. Brown  Audi A5            2014  Compact  Luxury    Daily       100.0      3.0         200.0
221     J. Brown  Audi A5            2014  Compact  Luxury    Weekly      600.0      29.0        2400.0
221     J. Brown  BMW 3 Series       2014  Compact  Luxury    Weekly      600.0      29.0        2400.0
221     J. Brown  BMW 3 Series       2014  Compact  Luxury    Weekly      600.0      29.0        2400.0
221     J. Brown  Mercedes_Benz GLK  2014  Compact  Luxury    Weekly      600.0      29.0        2400.0
sqlite>
```

Question 9-b: For the same customer return the current balance

```
SELECT
     c.CustID,
     c.Name,
     v.Description,
     v.Year,
     CASE v.Type
     WHEN 1 THEN
          "Compact"
     WHEN 2 THEN
          "Medium"
     WHEN 3 THEN
          "Large"
     WHEN 4 THEN
          "SUV"
     WHEN 5 THEN
          "Truck"
     WHEN 6 THEN
          "VAN"
     ELSE
          "DEFAULT"
     END AS TYPE,
     CASE v.Category
     WHEN 0 THEN
          "Basic"
     WHEN 1 THEN
          "Luxury"
     ELSE
          "DEFAULT"
     END AS Category,
     CASE rl.RentalType
     WHEN 1 THEN
          "Daily"
     WHEN 7 THEN
          "Weekly"
     ELSE
          "DEFAULT"
     END AS RentalType,
```

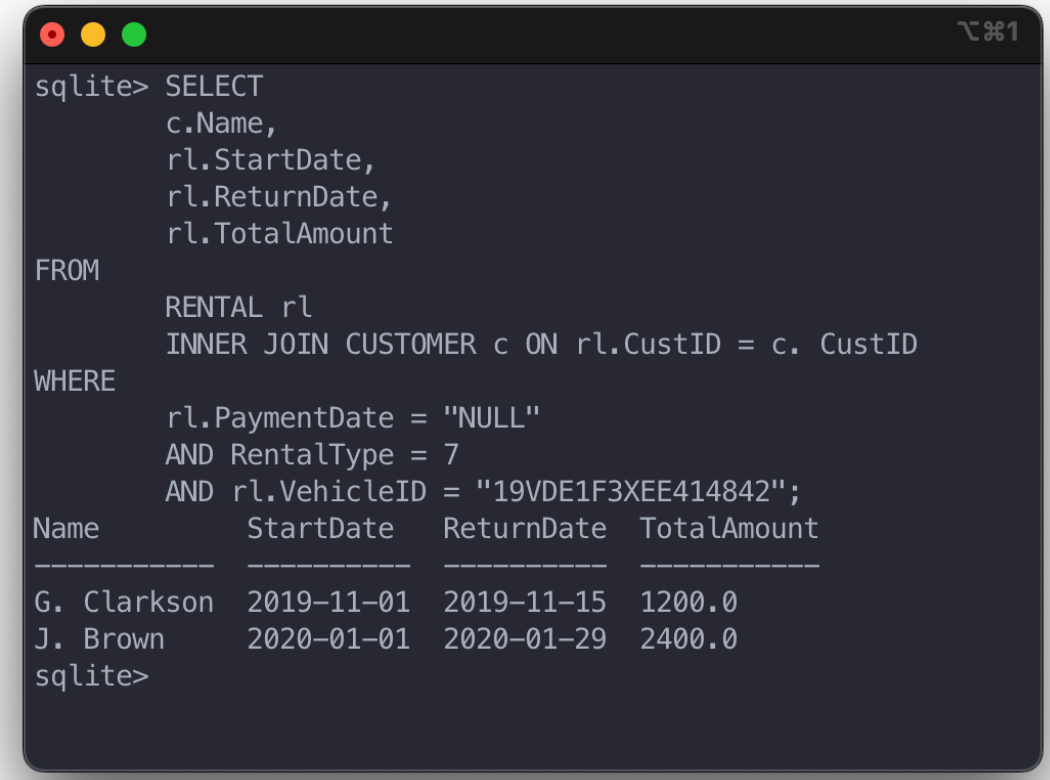
```
      (rl.TotalAmount / rl. Qty) AS UnitPrice,
      julianday (rl.ReturnDate) - julianday (rl.StartDate) + 1 AS
DaysRented,
      rl.TotalAmount,
      CASE rl.PaymentDate
      WHEN 'NULL' THEN
            rl.TotalAmount
      ELSE
            0
      END AS Balance
FROM
      CUSTOMER c
      INNER JOIN RENTAL rl ON c.CustID = rl.CustID
      INNER JOIN VEHICLE v ON v.VehicleID = rl.VehicleID
WHERE
      c.Name = "J. Brown”;
```



```
       v.Year,
       CASE v.Type
       WHEN 1 THEN
              "Compact"
       WHEN 2 THEN
              "Medium"
       WHEN 3 THEN
              "Large"
       WHEN 4 THEN
              "SUV"
       WHEN 5 THEN
              "Truck"
       WHEN 6 THEN
              "VAN"
       ELSE
              "DEFAULT"
       END AS TYPE,
       CASE v.Category
       WHEN 0 THEN
              "Basic"
       WHEN 1 THEN
              "Luxury"
       ELSE
              "DEFAULT"
       END AS Category,
       CASE rl.RentalType
       WHEN 1 THEN
              "Daily"
       WHEN 7 THEN
              "Weekly"
       ELSE
              "DEFAULT"
       END AS RentalType,
       (rl.TotalAmount / rl. Qty) AS UnitPrice,
       julianday (rl.ReturnDate) - julianday (rl.StartDate) + 1 AS DaysRented,
       rl.TotalAmount,
       CASE rl.PaymentDate
       WHEN 'NULL' THEN
              rl.TotalAmount
       ELSE
              0
       END AS Balance
FROM
       CUSTOMER c
       INNER JOIN RENTAL rl ON c.CustID = rl.CustID
       INNER JOIN VEHICLE v ON v.VehicleID = rl.VehicleID
WHERE
       c.Name = "J. Brown";
CustID  Name      Description        Year  TYPE     Category  RentalType  UnitPrice  DaysRented  TotalAmount  Balance
------  --------  -----------------  ----  -------  --------  ----------  ---------  ----------  -----------  -------
221     J. Brown  Acura ILX          2014  Compact  Luxury    Weekly      600.0      8.0         600.0        0
221     J. Brown  Acura ILX          2014  Compact  Luxury    Daily       100.0      3.0         200.0        0
221     J. Brown  Acura ILX          2014  Compact  Luxury    Weekly      600.0      29.0        2400.0       2400.0
221     J. Brown  Lexus IS 250C      2015  Compact  Luxury    Weekly      600.0      29.0        2400.0       2400.0
221     J. Brown  Audi A5            2014  Compact  Luxury    Weekly      600.0      8.0         600.0        0
221     J. Brown  Audi A5            2014  Compact  Luxury    Daily       100.0      3.0         200.0        0
221     J. Brown  Audi A5            2014  Compact  Luxury    Weekly      600.0      29.0        2400.0       2400.0
221     J. Brown  BMW 3 Series       2014  Compact  Luxury    Weekly      600.0      29.0        2400.0       2400.0
221     J. Brown  BMW 3 Series       2014  Compact  Luxury    Weekly      600.0      29.0        2400.0       2400.0
221     J. Brown  Mercedes_Benz GLK  2014  Compact  Luxury    Weekly      600.0      29.0        2400.0       2400.0
sqlite>
```

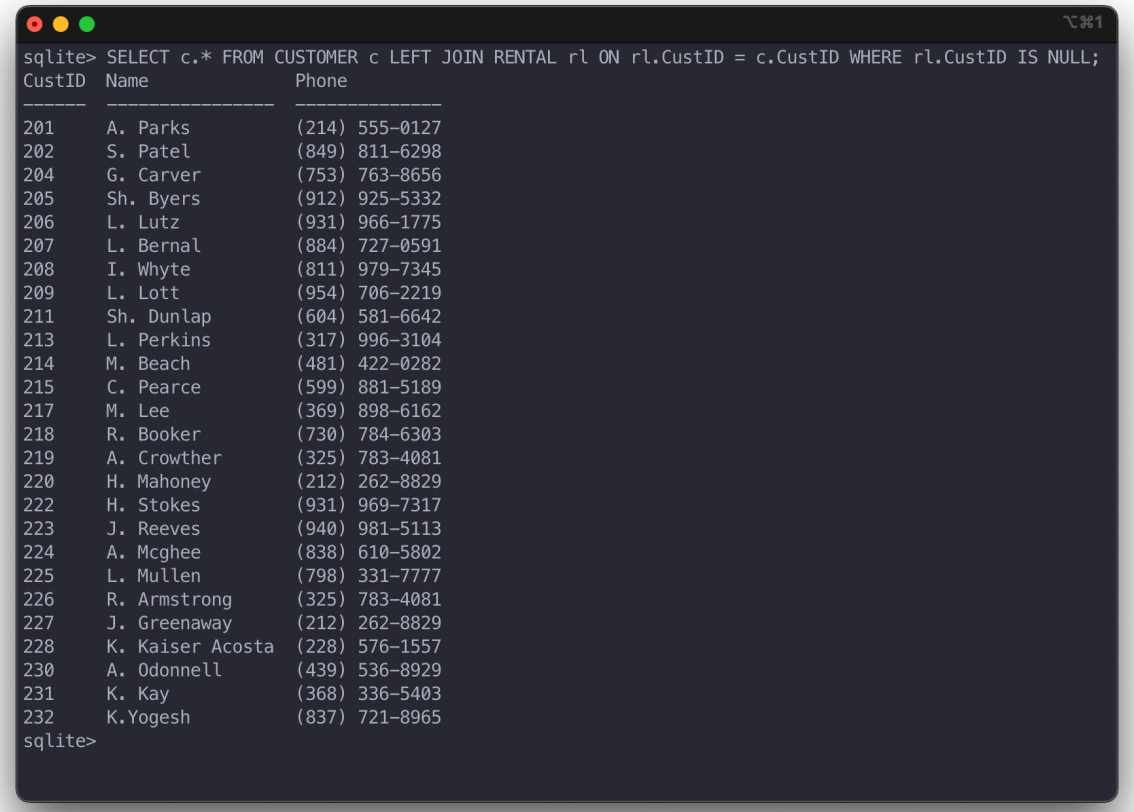
Question 10: Retrieve all weekly rentals for the vechicleID '19VDE1F3XEE414842' that are not paid yet. List the Customer Name, the start and return date, and the amount

```
SELECT
      c.Name,
      rl.StartDate,
      rl.ReturnDate,
      rl.TotalAmount
FROM
      RENTAL rl
      INNER JOIN CUSTOMER c ON rl.CustID = c. CustID
WHERE
      rl.PaymentDate = "NULL"
      AND RentalType = 7
      AND rl.VehicleID = “19VDE1F3XEE414842";
```



Question 11: Return all customers that they never rent a vehicle.

```
SELECT
      c.*
FROM
      CUSTOMER c
      LEFT JOIN RENTAL rl ON rl.CustID = c.CustID
WHERE
      rl.CustID IS NULL;
```



```
sqlite> SELECT c.* FROM CUSTOMER c LEFT JOIN RENTAL rl ON rl.CustID = c.CustID WHERE rl.CustID IS NULL;
CustID  Name              Phone
------  ----------------  --------------
201     A. Parks          (214) 555-0127
202     S. Patel          (849) 811-6298
204     G. Carver         (753) 763-8656
205     Sh. Byers         (912) 925-5332
206     L. Lutz           (931) 966-1775
207     L. Bernal         (884) 727-0591
208     I. Whyte          (811) 979-7345
209     L. Lott           (954) 706-2219
211     Sh. Dunlap        (604) 581-6642
213     L. Perkins        (317) 996-3104
214     M. Beach          (481) 422-0282
215     C. Pearce         (599) 881-5189
217     M. Lee            (369) 898-6162
218     R. Booker         (730) 784-6303
219     A. Crowther       (325) 783-4081
220     H. Mahoney        (212) 262-8829
222     H. Stokes         (931) 969-7317
223     J. Reeves         (940) 981-5113
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
230     A. Odonnell       (439) 536-8929
231     K. Kay            (368) 336-5403
232     K.Yogesh          (837) 721-8965
sqlite>
```

Question 12: Return all rentals that the customer paid on the StartDate. List Customer Name, Vehicle Description, StartDate, ReturnDate, and TotalAmount. Order by Customer Name

```
SELECT
      c.Name,
      v.Description,
      rl.StartDate,
      rl.ReturnDate,
      rl.TotalAmount
FROM
      RENTAL rl
      INNER JOIN CUSTOMER c ON rl.CustID = c.CustID
      INNER JOIN VEHICLE v ON v.VehicleID = rl.VehicleID
WHERE
      rl.StartDate = rl.PaymentDate
ORDER BY
      c.Name ASC;
```

```
sqlite> SELECT
        c.Name,
        v.Description,
        rl.StartDate,
        rl.ReturnDate,
        rl.TotalAmount
FROM
        RENTAL rl
        INNER JOIN CUSTOMER c ON rl.CustID = c.CustID
        INNER JOIN VEHICLE v ON v.VehicleID = rl.VehicleID
WHERE
        rl.StartDate = rl.PaymentDate
ORDER BY
        c.Name ASC;
Name            Description  StartDate   ReturnDate  TotalAmount
--------------  -----------  ----------  ----------  -----------
A. Hernandez    Mazda CX5    2019-09-09  2019-09-13  460.0
A. Hess         Nissan NV    2019-08-02  2019-08-30  2740.0
D. Kirkpatrick  Acura ILX    2019-05-06  2019-05-10  400.0
D. Kirkpatrick  Audi A5      2019-05-06  2019-05-10  400.0
H. Gallegos     Acura ILX    2019-06-10  2019-07-01  1800.0
J. Brown        Acura ILX    2019-07-01  2019-07-08  600.0
J. Brown        Audi A5      2019-07-01  2019-07-08  600.0
sqlite>
```