College of Engineering, Trivandrum

Department of Computer Science and Engineering

CS333 APPLICATION SOFTWARE DEVELOPMENT LAB

LABORATORY REPORT 4

Learning Basic SQL queries(Part 2)

Student Name	Student ID
1. Justine Biju(S5)	170445(Roll No:37)

Submission Date : 29/07/2019

1 Introduction

There are many SQL queries that help support out basic queries that SQL offers. Few of what we are going to discuss are listed below:

1. ALTER
   (a) The ALTER TABLE statement is used to add, delete, or modify columns in an existing table.
   (b) The ALTER TABLE statement is also used to add and drop various constraints on an existing table.
2. RENAME
   (a) We use this clause along with ALTER to rename a column name.
   (b) TO clause is also used to specify the new name.
3. SELECT DISTINCT
   (a) The SELECT DISTINCT statement is used to return only distinct (different) values.
   (b) The output ignores redundant data.
4. SQL IN
   (a) The IN operator allows you to specify multiple values in a WHERE clause.
   (b) The IN operator is a shorthand for multiple OR conditions.
5. SQL BETWEEN
   (a) The BETWEEN operator selects values within a given range. The values can be numbers, text, or dates.
   (b) The BETWEEN operator is inclusive: begin and end values are included.
6. SQL ALIASES
   (a) SQL aliases are used to give a table, or a column in a table, a temporary name.
   (b) Aliases are often used to make column names more readable.
   (c) An alias only exists for the duration of the query.
7. SQL AND
   (a) The AND operator displays a record if all the conditions separated by AND are TRUE.
8. SQL OR
   (a) The OR operator displays a record if any of the conditions separated by OR is TRUE.

All these SQL queries are supported by PostgreSQL and can be implemented in a similar fashion.

2 Implementation in PostgreSQL

Let's take a look at the database we are going to work with.

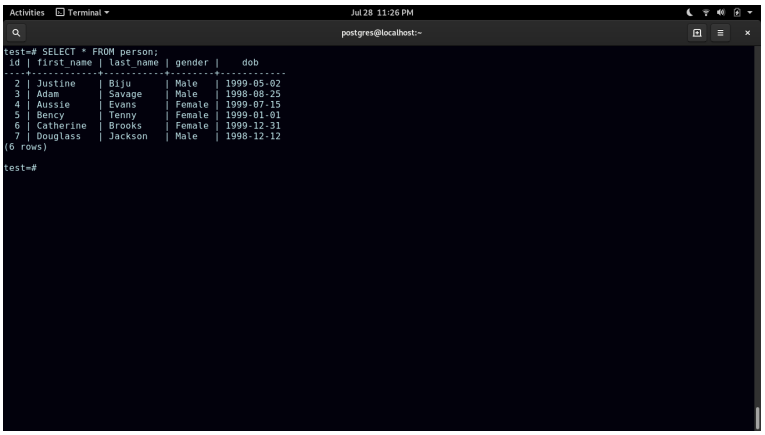


Figure 1: Entities in the table 'person'

1. Lets use the ALTER query to add an email column to the database. This can be done using the following query:

```
ALTER TABLE person
ADD email VARCHAR(50);

SELECT * FROM person;
```

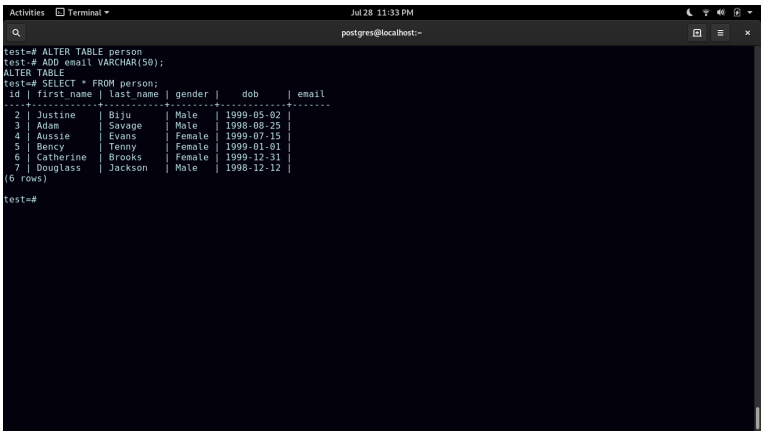


Figure 2: Adding 'email' column

2. Now lets use ALTER RENAME to rename the 'id' column to 'p_id'

```
ALTER TABLE person
RENAME id to p_id;

SELECT * FROM person;
```

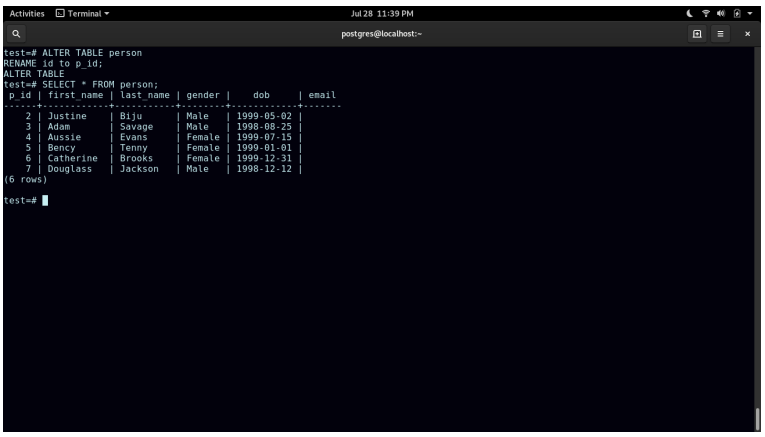


Figure 3: Renaming 'id' to 'pid'

3. We can select different entities from a database using the SELECT DISTINCT query.

```
SELECT DISTINCT gender
FROM person;
```

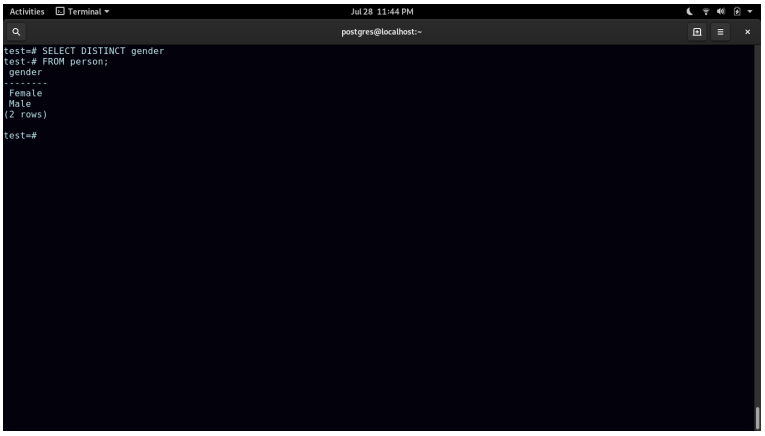


Figure 4: Distinct 'gender' in DB

4. We can use IN to check records with multiple values.

```
SELECT *
FROM person
WHERE first_name IN ('Justine', 'Douglass');
```

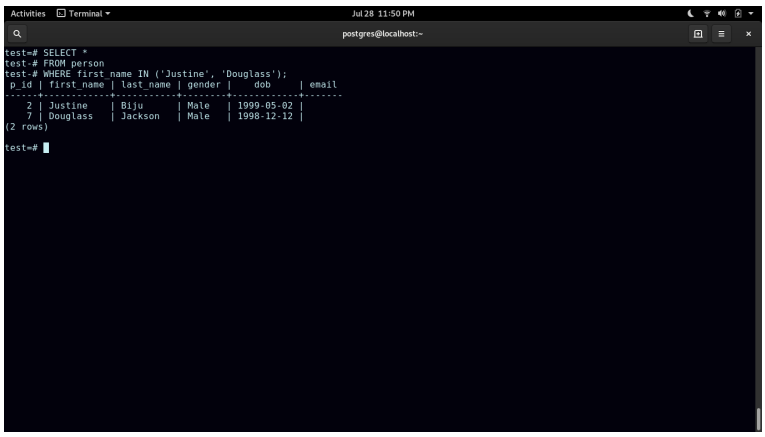


Figure 5: Using the IN query

5. We can use BETWEEN clause in SELECT to display records that have values in a particular range.

```
SELECT *
FROM person
WHERE dob BETWEEN '1999-01-01' AND '1999-07-01';
```

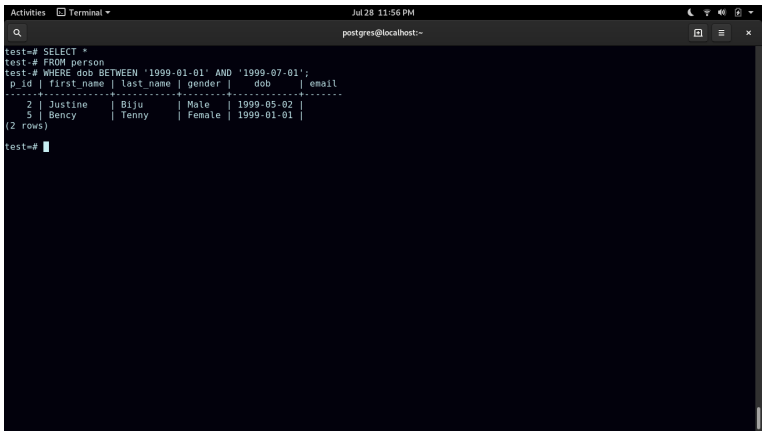


Figure 6: Using BETWEEN clause

6. We can use aliases to make the database to a user.

```
SELECT first_name AS NAME,
dob AS "Date of Birth"
FROM person;
```

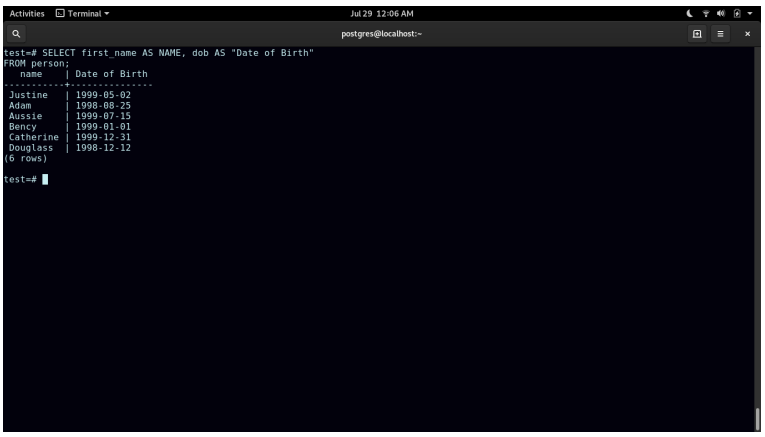


Figure 7: Using an alias for column name

7. We can use AND and/or OR clause to group conditional statements.

```
SELECT *
FROM person
WHERE (first_name = 'Justine' OR
 first_name = 'Adam') AND
 (dob BETWEEN '1999-01-01' AND
 '1999-12-31');
```

Figure 8: Using AND and OR clause

3 Questions

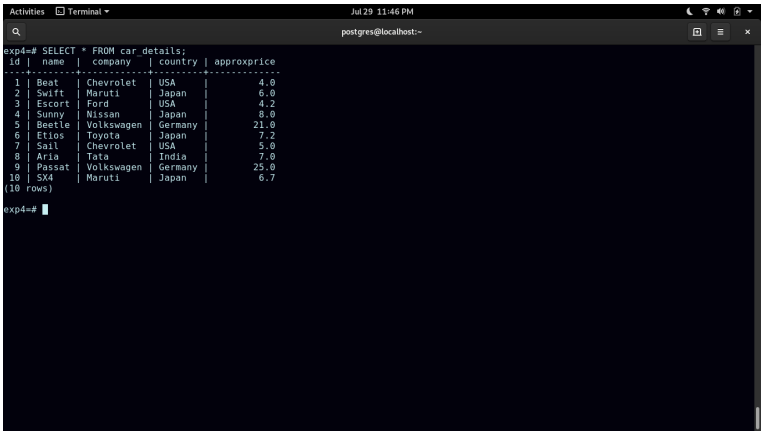


Figure 9: Entries in 'car_details'

1. List the names of all companies as mentioned in the database

```
SELECT DISTINCT company
FROM car_details;
```

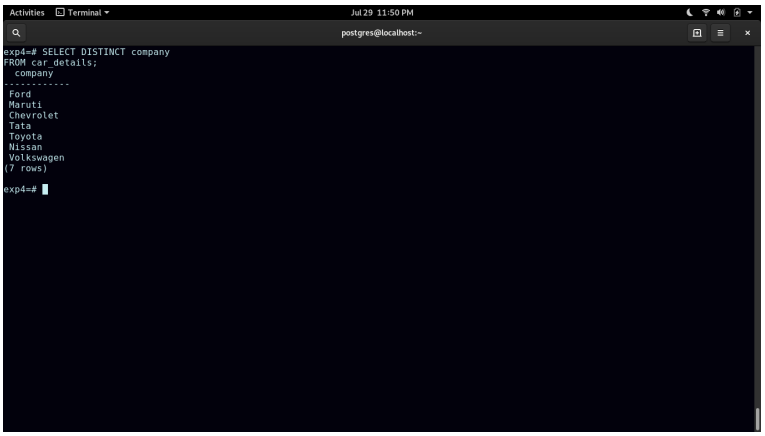


Figure 10: Question 1

2. List the names of all countries having car production companies

```
SELECT DISTINCT country
FROM car_details;
```

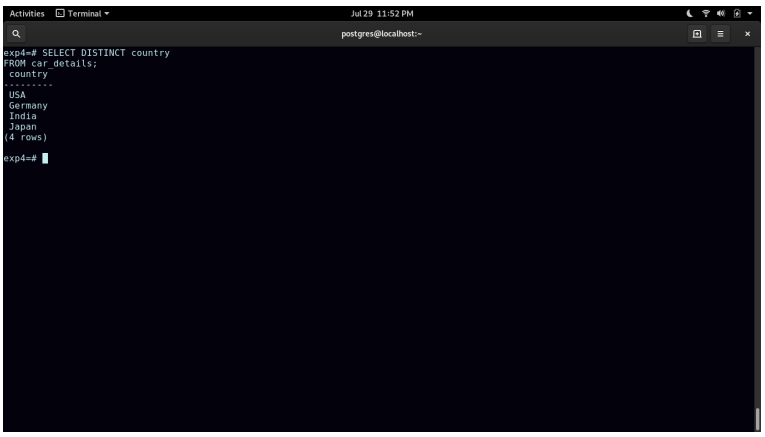


Figure 11: Question 2

3. List the details of all cars within a price range 4 to 7 lakhs

```
SELECT * FROM car_details
WHERE approxprice BETWEEN
 4.0 AND 7.0;
```

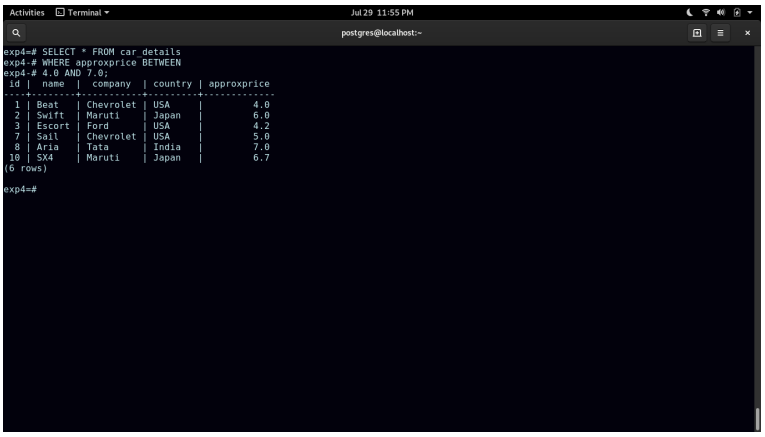


Figure 12: Question 3

4. List the name and company of all cars originating from Japan and having price¡=6 lakhs

```
SELECT name, company
FROM car_details
WHERE country = 'Japan' AND
 approxprice <= 6.0;
```

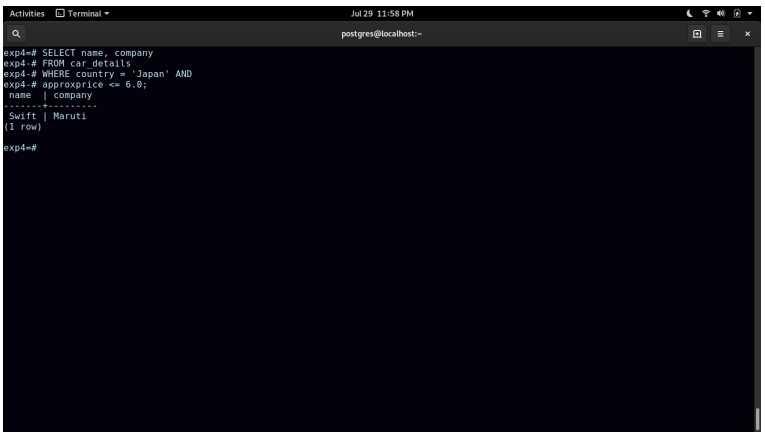


Figure 13: Question 4

5. List the names and the companies of all cars either from Nissan or having a price greater than 20 lakhs.

```
SELECT name, company
FROM car_details
WHERE company = 'Nissan' OR
 approxprice > 20.0;
```

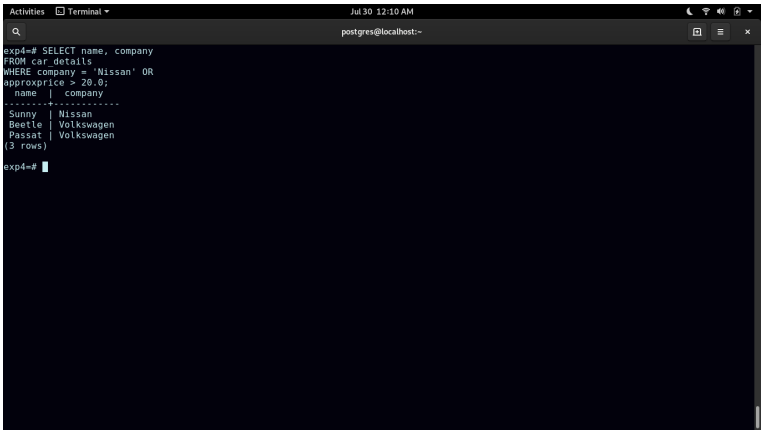


Figure 14: Question 5

6. List the names of all cars produced by (Maruti,Ford).Use SQL IN statement.

```
SELECT name
FROM car_details
WHERE company in ('Maruti', 'Ford');
```

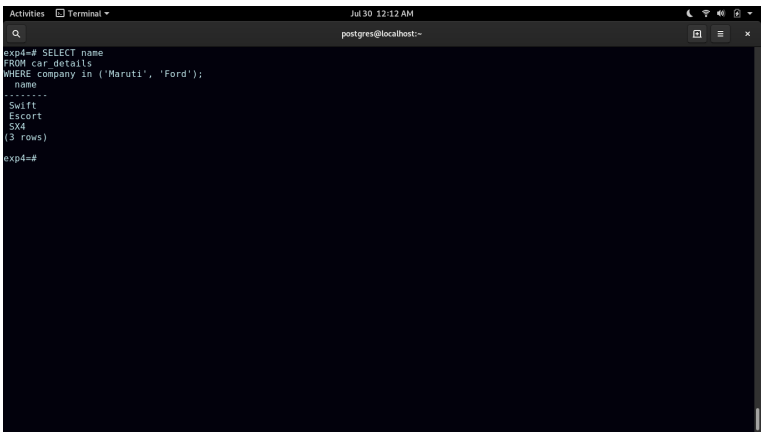


Figure 15: Question 6

7. Alter the table cars to add a new field year (model release year).Update the year column for all the rows in the database.

```
ALTER TABLE car_details
ADD year INT DEFAULT 2015;

SELECT * FROM car_details;
```

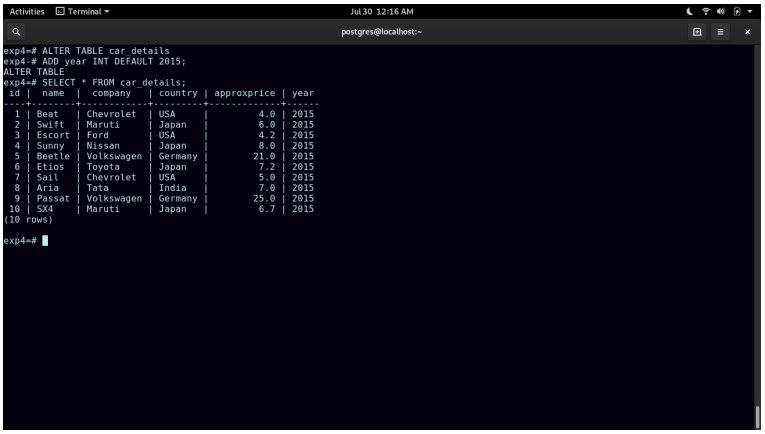


Figure 16: Question 7

8. Display the names of all cars as Car_name (while displaying the name attribute should be listed as car_aliases)

```
SELECT name AS Car_name
FROM car_details;
```

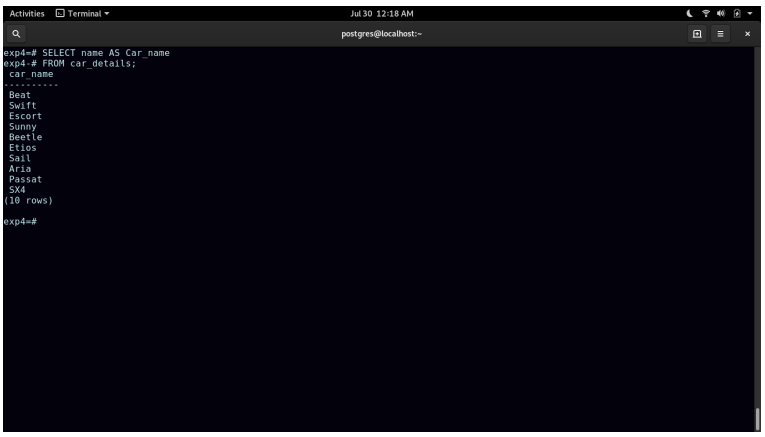


Figure 17: Question 8

9. Rename the attribute name to car_name

```
ALTER TABLE car_details
RENAME name TO car_name;

SELECT * FROM car_details;
```

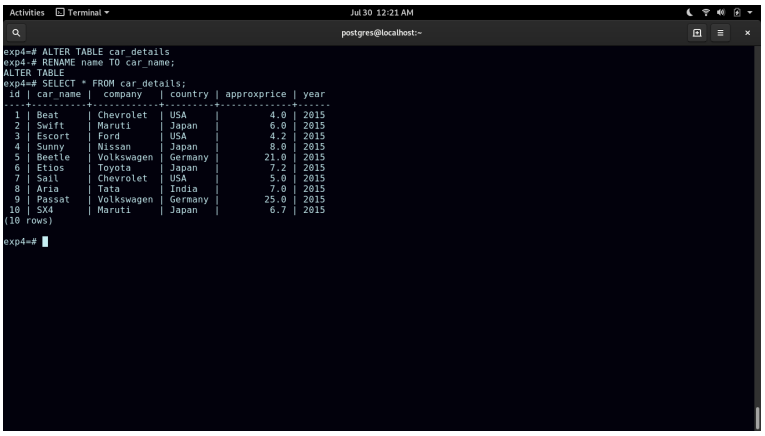


Figure 18: Question 9

10. List the car manufactured by Toyota(to be displayed as cars_Toyota)

```
SELECT car_name AS CARS_TOYOTA
FROM car_details
WHERE company = 'Toyota';
```

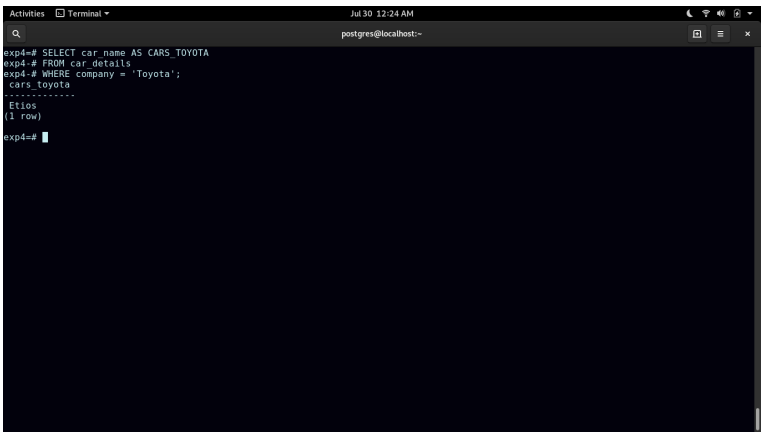


Figure 19: Question 10

11. List the details of all cars in alphabetical order

```
SELECT * FROM car_details
ORDER BY car_name;
```

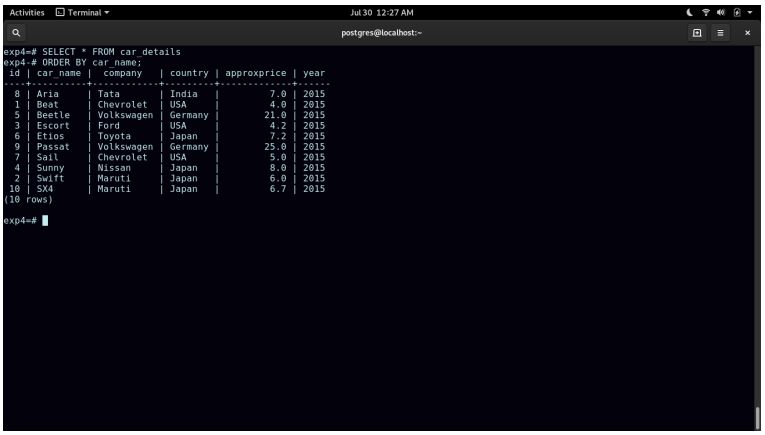
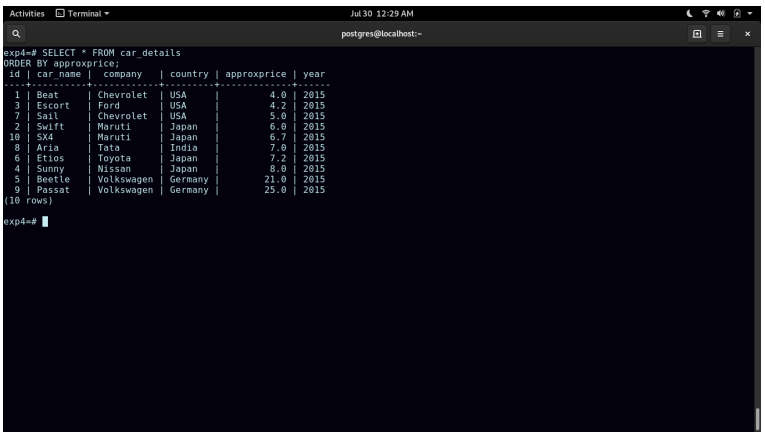


Figure 20: Question 11

12. List the details of all cars from cheapest to costliest.

```
SELECT * FROM car_details
ORDER BY approxprice;
```



```
exp4=# SELECT * FROM car_details
ORDER BY approxprice;
 id | car_name |  company   | country | approxprice | year
----+----------+------------+---------+-------------+------
  1 | Beat     | Chevrolet  | USA     |         4.0 | 2015
  3 | Escort   | Ford       | USA     |         4.2 | 2015
  7 | Sail     | Chevrolet  | USA     |         5.0 | 2015
  2 | Swift    | Maruti     | Japan   |         6.0 | 2015
 10 | SX4      | Maruti     | Japan   |         6.7 | 2015
  8 | Aria     | Tata       | India   |         7.0 | 2015
  6 | Etios    | Toyota     | Japan   |         7.2 | 2015
  4 | Sunny    | Nissan     | Japan   |         8.0 | 2015
  5 | Beetle   | Volkswagen | Germany |        21.0 | 2015
  9 | Passat   | Volkswagen | Germany |        25.0 | 2015
(10 rows)

exp4=#
```

Figure 21: Question 12

4 Result

- Successfully implemented ALTER, RENAME, SELECT DISTINCT, SQL IN, BETWEEN, AND and OR queries on PostgreSQL.