**>**

**ACTIVITY 6:** **Functions and Aggregate Functions**

**Objectives:**

1. To know how to query data using SELECT and SELECT DISTINCT
2. To know how to invoke SQL commands using aggregate functions
3. To learn how to use Order by, Group by and Having statements

**Materials:**

PC or Laptop

WAMP/XAMPP Installer

Web Browser or CLI

**Background**

This activity helps you learn how to query data from the MySQL database server. You learned from your previous activities a simple SELECT statement that allows you to query data from a single table. Additionally,

* **SELECT** – show you how to use simple SELECT statement to query the data from a single table.
* **SELECT DISTINCT** – learn how to use the DISTINCT operator in the SELECT statement to eliminate duplicate rows in a result set.

When querying data from a table, you may get duplicate rows. In order to remove these duplicate rows, you use the **DISTINCT** clause in the **SELECT** statement.

The syntax of using the **DISTINCT** clause is as follows:

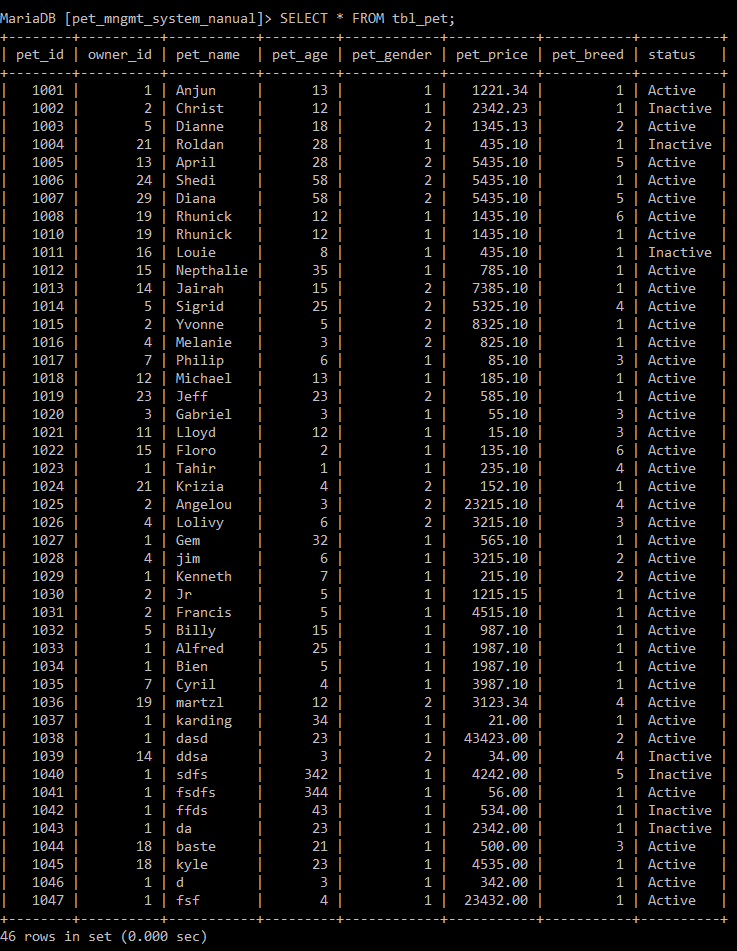
***SELECT DISTINCT columns FROM table\_name WHERE where\_conditions;***

***Example:*** ***select distinct(pet\_gender) from tbl\_pet where pet\_age=2;***

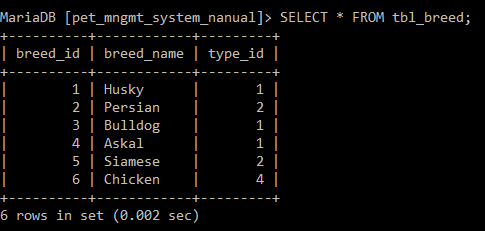
**TASK:**

1. Display all records of every table in your database.

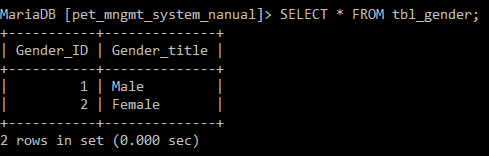
**tbl\_pet**



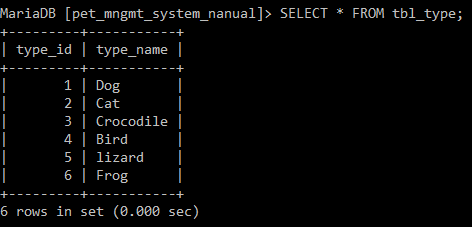
**tbl\_breed**



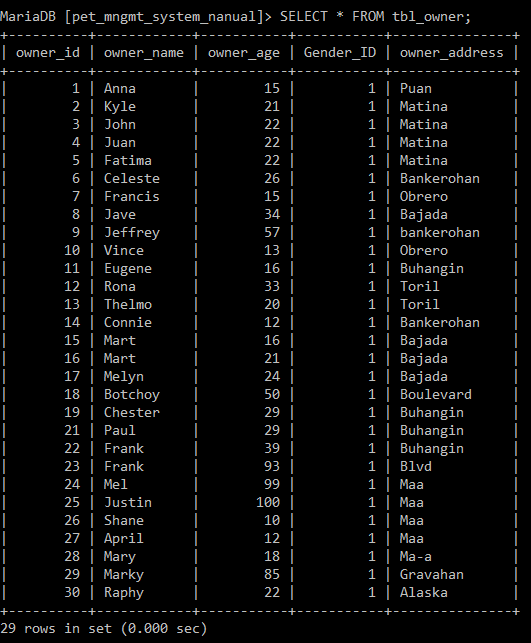
**tbl\_breed**



**tbl\_type**

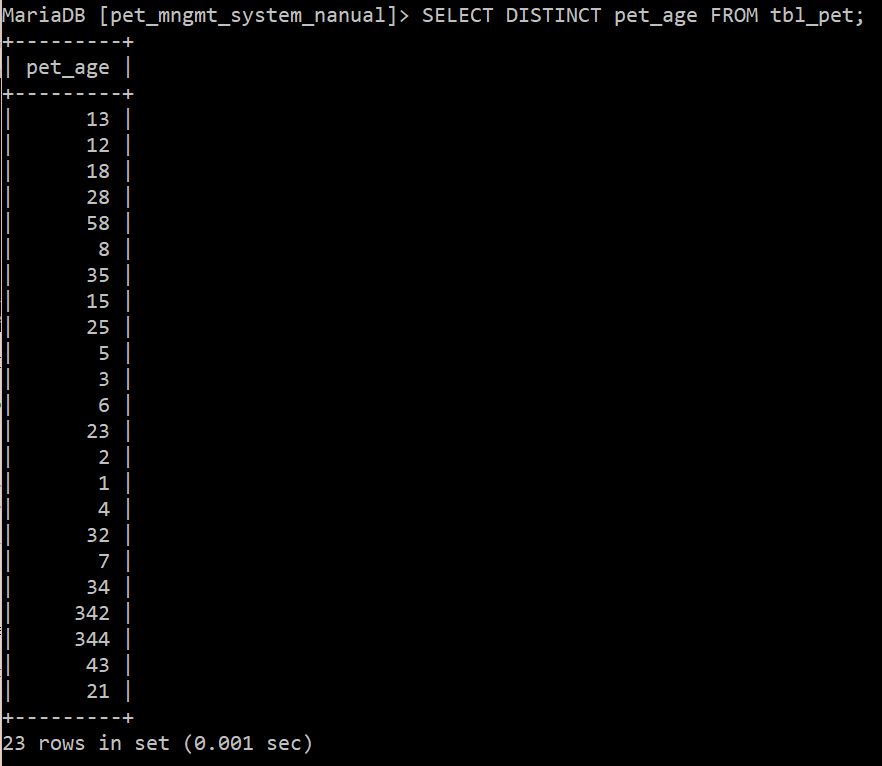


**tbl\_owner**



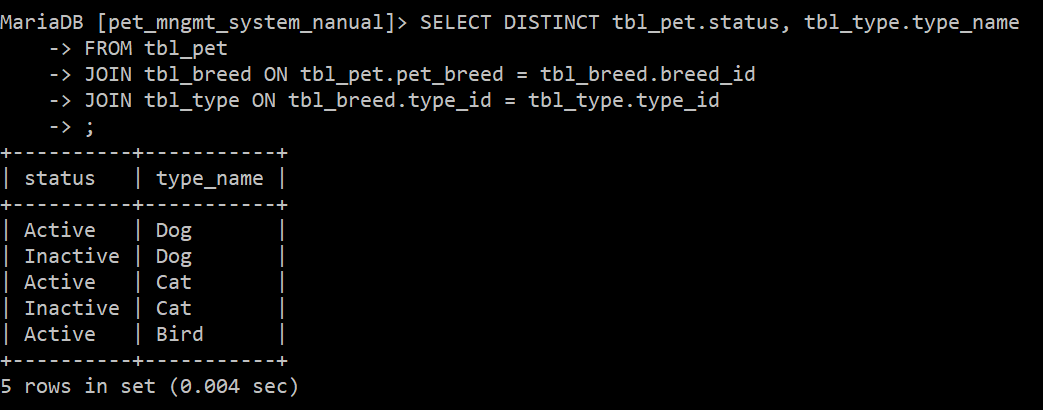
1. Display the ***ages of pet*** without duplication.

**My command prompt:**



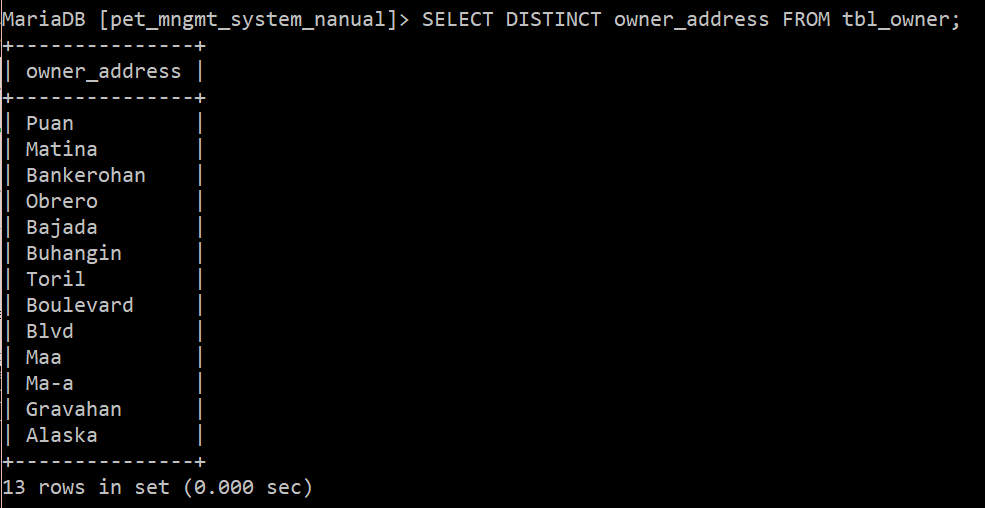
1. Display all combined unique ***pet’s status and type*** without duplicating its values.

**My command prompt:**



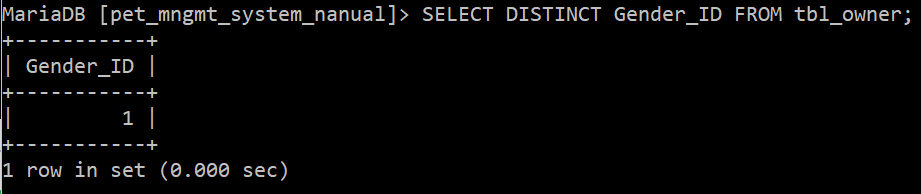
1. Display how all addresses in your ***tbl\_owner*** having no same values.

**My command prompt:**



1. Display how many gender types in your ***tbl\_owner***

**My command prompt:**



But of course, you don’t stop from querying your data in your database. It is also good if you can filter your data in a way it is summarized to present and project data well. Here are some of the SQL filter you must learn:

* **WHERE –** learn how to use the WHERE clause to filter rows based on specified conditions.
* **IN –** show you how to use the IN operator in the WHERE clause to determine if a value matches any value in a list or a subquery.

Text

Description automatically generated

* **BETWEEN –** show you how to query data based on a range using BETWEEN operator.

A picture containing diagram

Description automatically generated

* **LIKE –** provide you with technique to query data based on a specific pattern. LIKE provides two wildcard characters that allows you to construct patterns. These two wildcards are:
  1. Percentage (% ) wildcard matches a sequence of any character including space.

*Suppose you want to find pets whose last name starts with the letter ‘a’, you can use the following query.*

A picture containing text

Description automatically generated

*Or starts with letter ‘****c’***

A picture containing text

Description automatically generated

*Or to find pets whose first name ends with* ***‘a’****, you can execute the following query:*

A screenshot of a computer

Description automatically generated with low confidence

* 1. Underscore ( \_ ) wildcard matches any single character.

A picture containing timeline

Description automatically generated

*You can combine two wildcard characters ‘%’ and ‘\_’ to construct a pattern. For example, you can find pets whose last name starts with any single character, followed by character a, and ends with any characters as the following query.*

A picture containing text, scoreboard

Description automatically generated

* **LIMIT –** use LIMIT to constrain the number of rows returned by SELECT statement

A picture containing diagram

Description automatically generated

* **IS NULL –** test whether a value is NULL or not by using IS NULL operator.

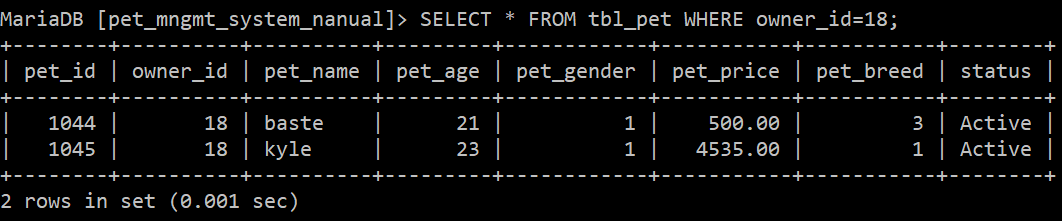
A picture containing calendar

Description automatically generated

**TASK:**

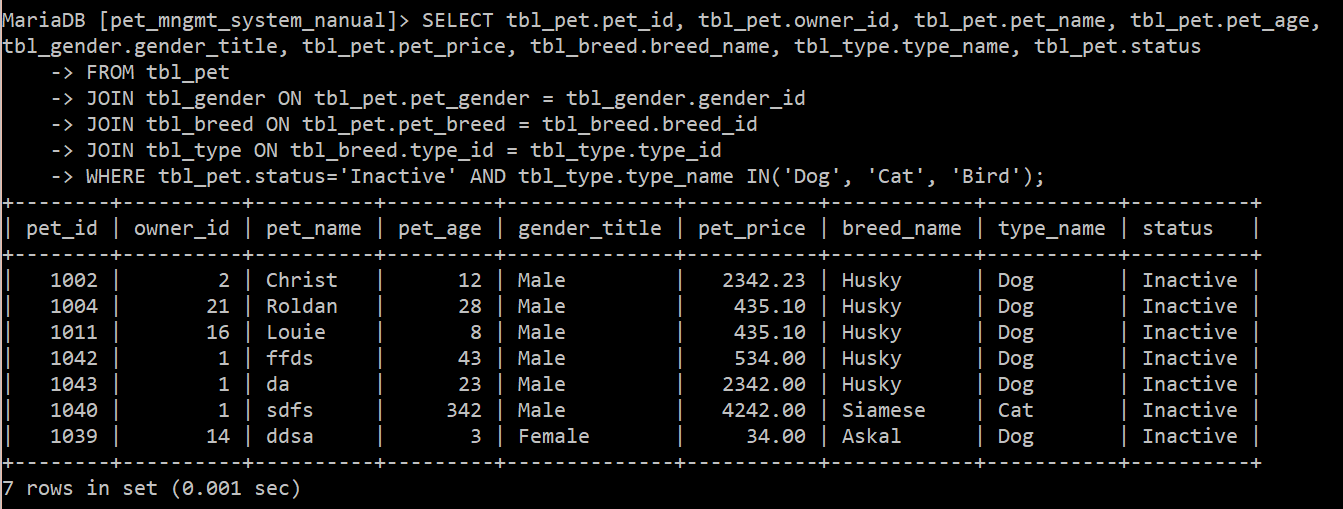
1. **WHERE**. Display all records of your pet where owner id is 18.

**My command prompt:**



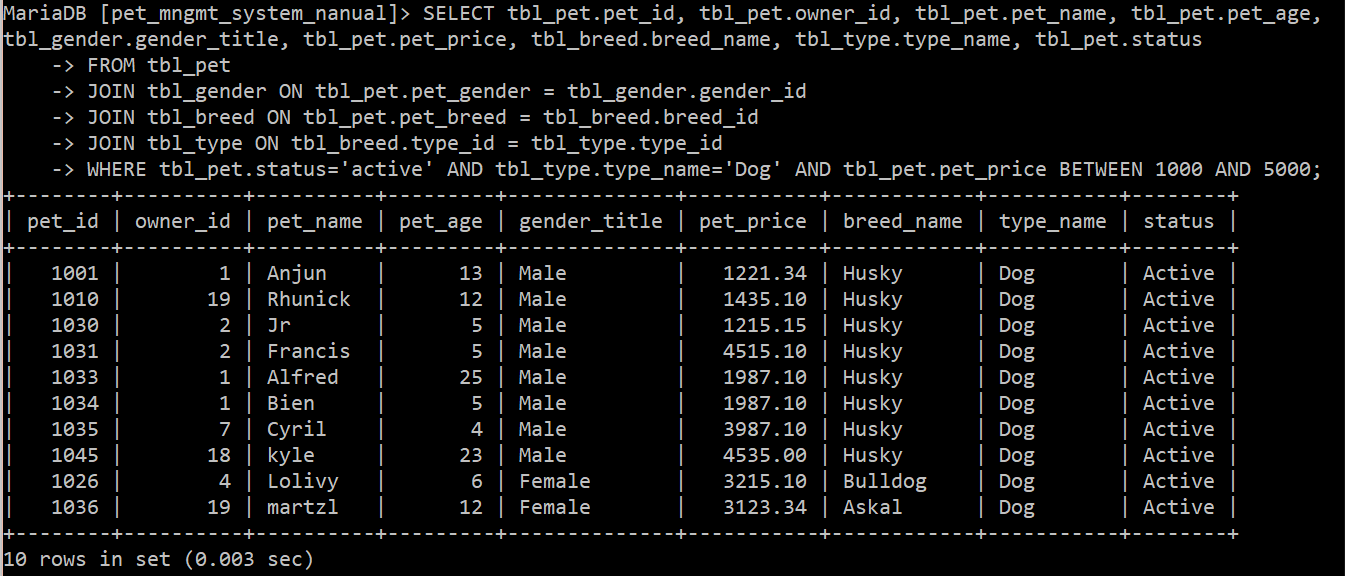
1. **IN**. Display all pets which are dogs, cats, and bird but status must be ‘Inactive’.

**My command prompt:**



1. **BETWEEN.** Display all pets whose price ranges from 1000 to 5000. The pet must be dog and is active.

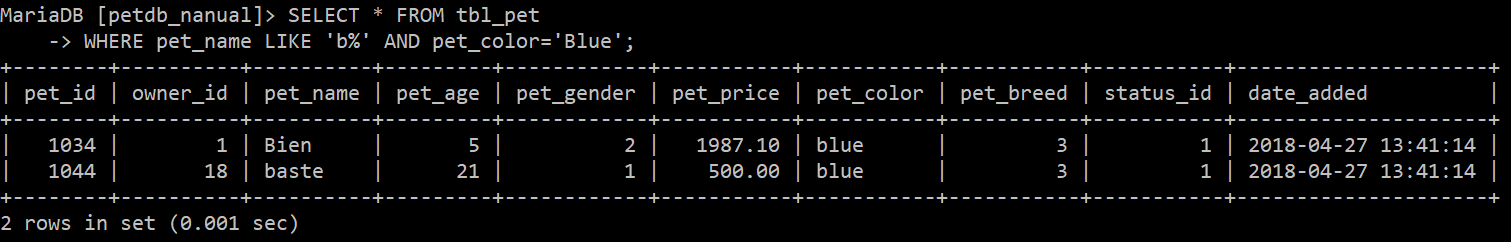
**My command prompt:**



1. **LIKE.** Display all pets whose name starts with letter **b** where color must be blue.

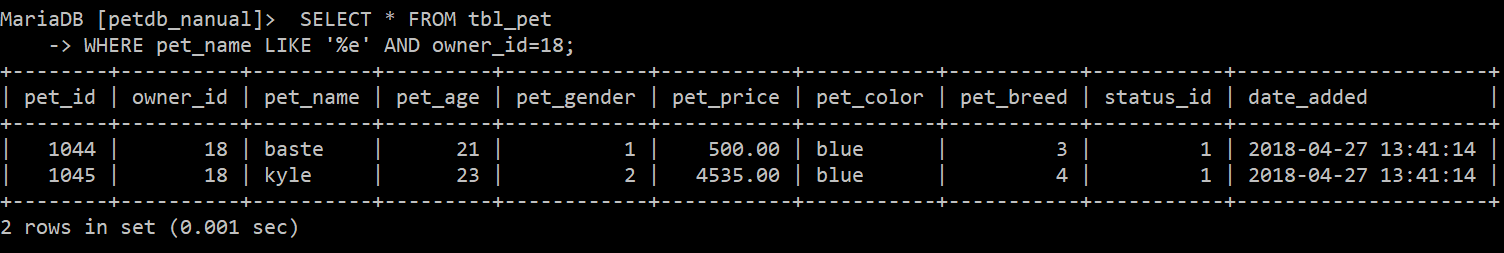
**My command prompt:**

**NOTE:** From here and onwards, I switched to using the ‘petDB’ database because the .sql file in the *Pet Management System-OOP Tutorial* folder (see downloadables) does not have a ‘pet-color’ column in its ‘tbl\_pet’ table.



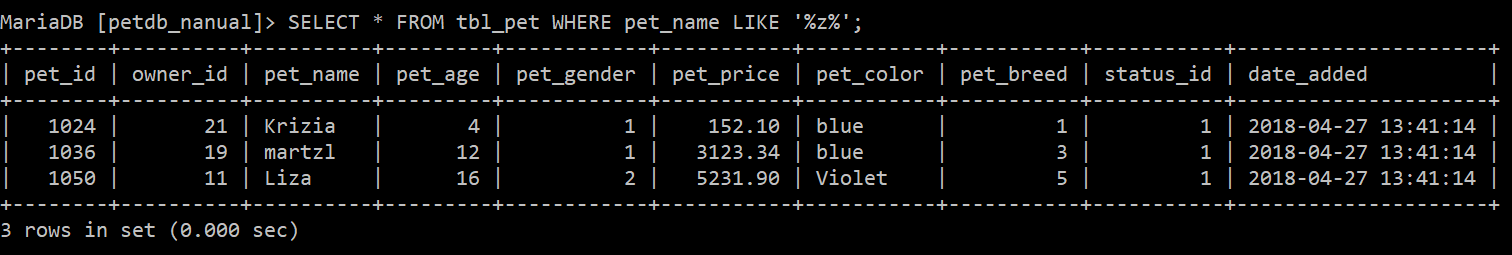
1. **LIKE.** Display all pets whose name ends with letter **e** which is owned by owner\_id 18.

**My command prompt:**



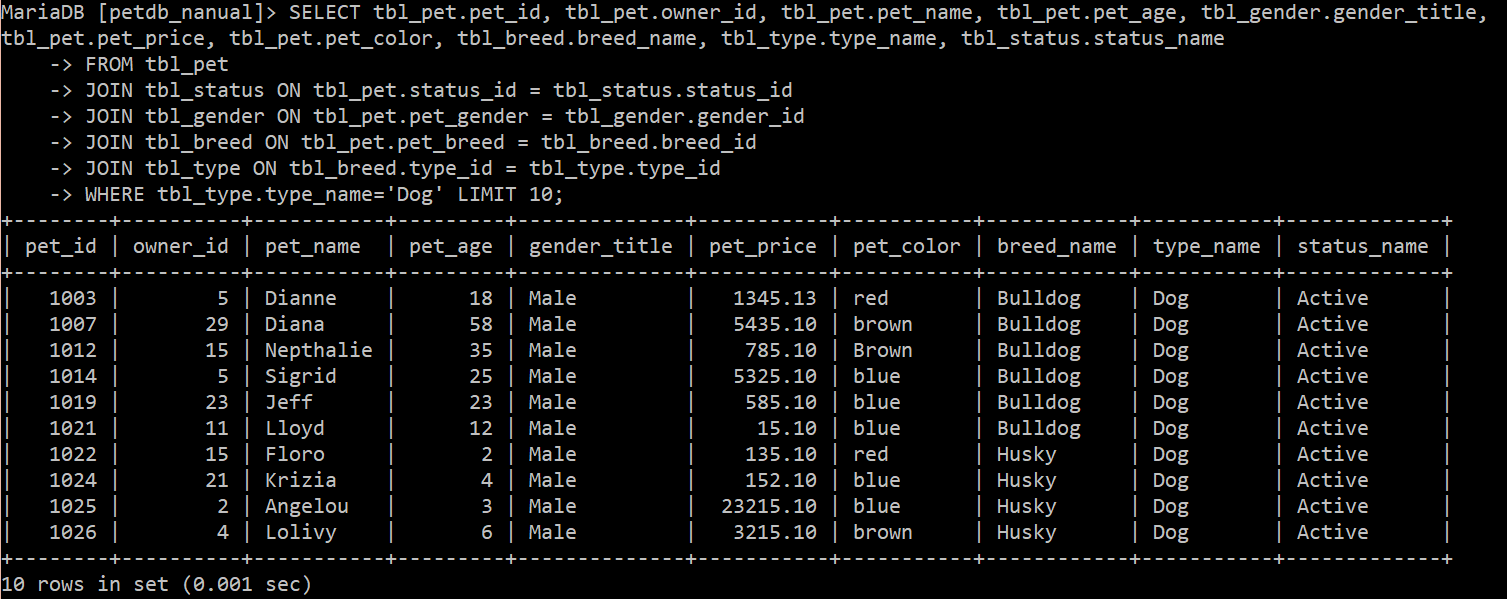
1. **LIKE.** Display all pets whose name contains **z**.

**My command prompt:**

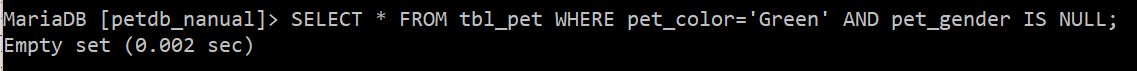


1. **LIMIT.** Display ONLY 10 information of pet, but for all types which are dogs.

**My command prompt:**



1. **IS NULL.** Find NO gender pet which color is green.



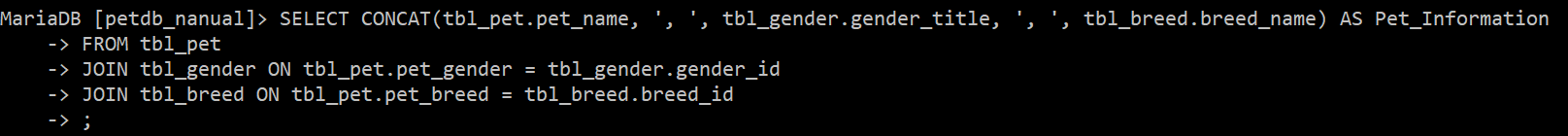
**Functions and Aggregate Functions**

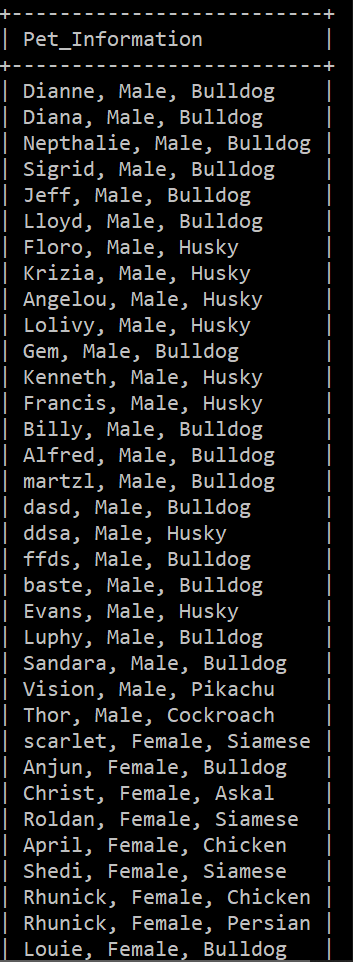
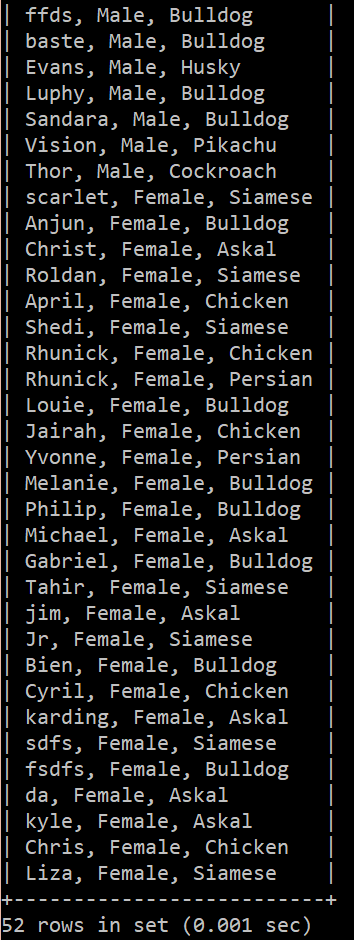
**CONCAT()*.*It is a function that is used to add two or more strings or values.**

***Example: Select CONCAT(pet\_id, "   ", pet\_name) as pet\_id\_and\_name from tbl\_pet;***

1. Display in the screen the name, gender, and breed of pets in column "Pet\_Information".

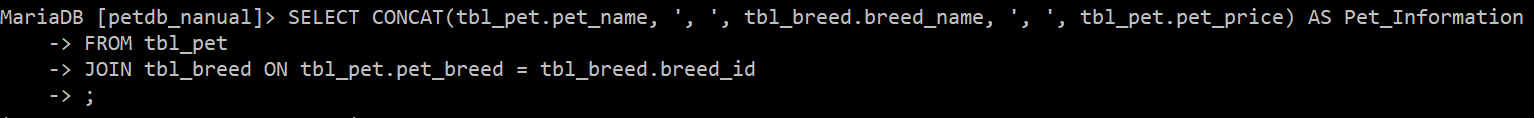
**My command prompt:**

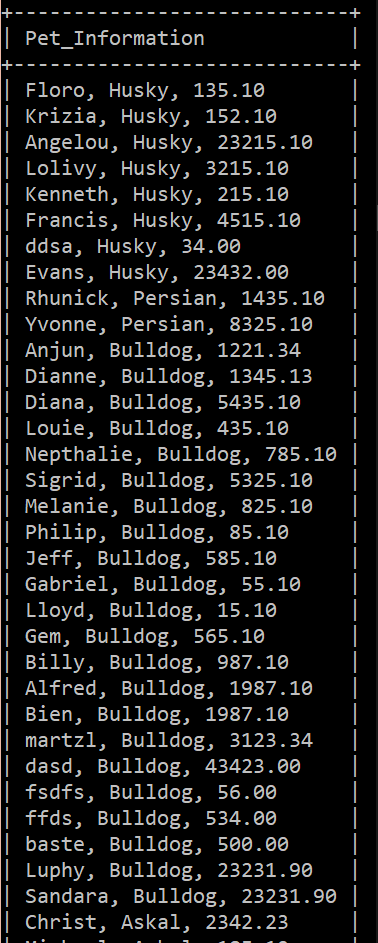
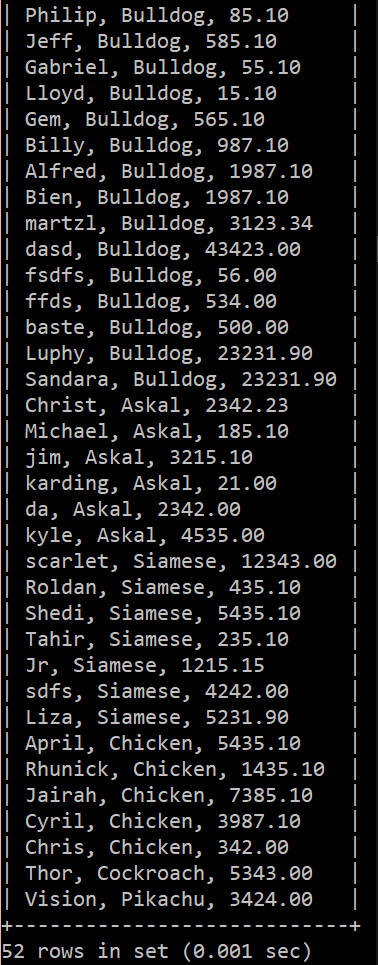


1. Display in the screen the name, breed, and price of pets in column "Pet\_Information".

**My command prompt:**

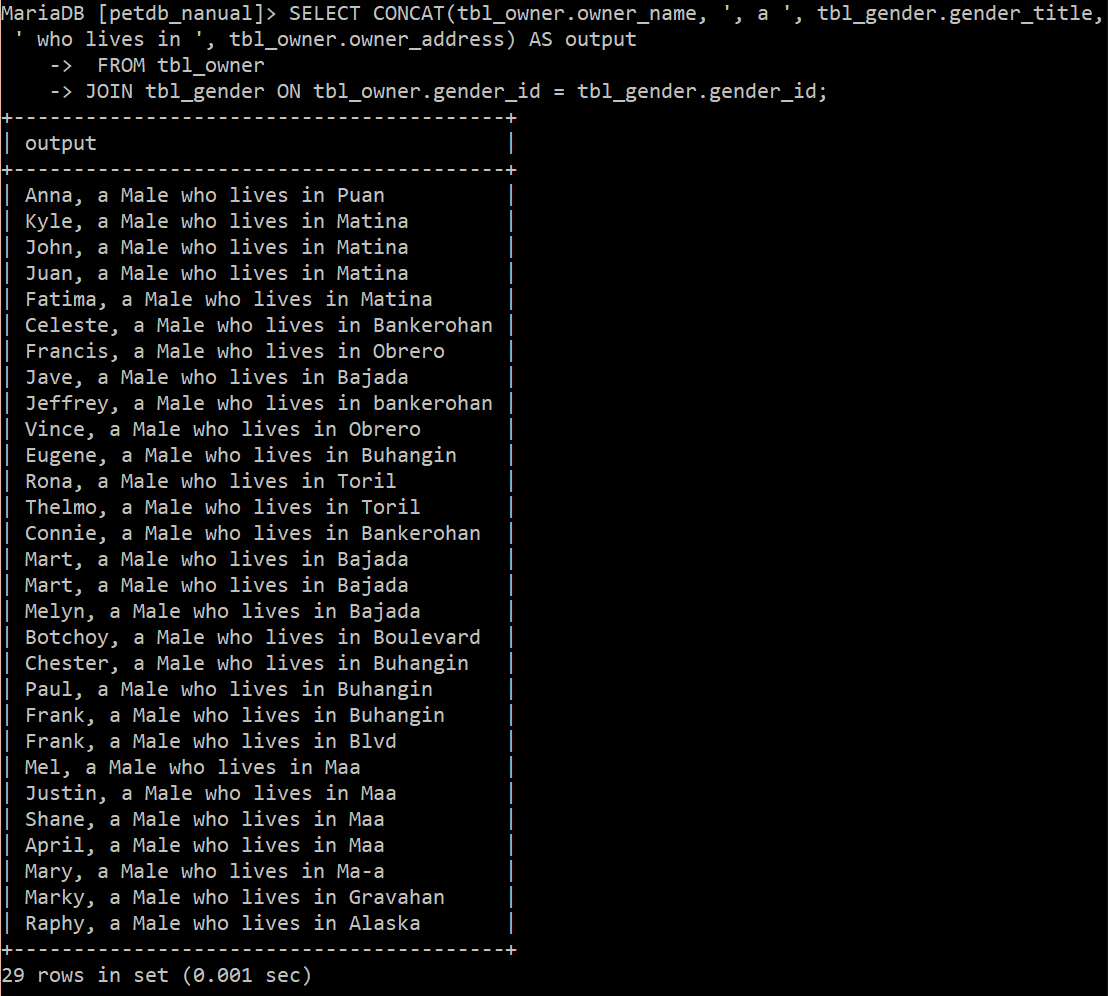


1. Display an output of the owner table name, **"a"** gender **"who lives in"** address.

***example: Pedro, a male who lives in Davao city.***

**My command prompt:**



**Aggregate Functions**

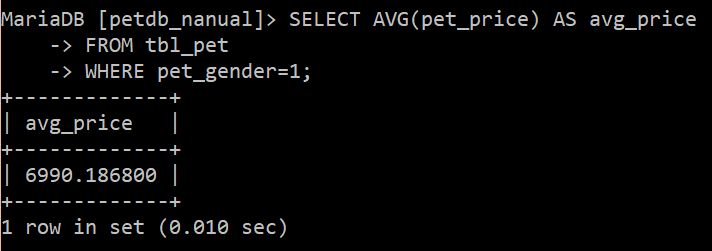
An aggregate function allows you to perform a calculation on a set of values to return a single scalar value. We often use aggregate functions with the GROUP BY and HAVING clauses of the SELECT statement. The following are the most commonly used SQL aggregate functions:

1. **AVG** – calculates the average of a set of values.
2. **COUNT** – counts rows in a specified table or view.
3. **MIN** – gets the minimum value in a set of values.
4. **MAX** – gets the maximum value in a set of values.
5. **SUM** – calculates the sum of values.

**TASK:**

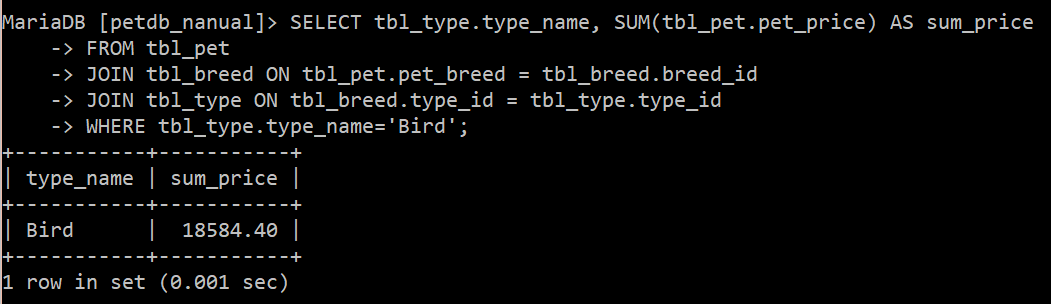
1. **AVG**. Show how much is the average price value of all pets whose gender is male.

**My command prompt:**



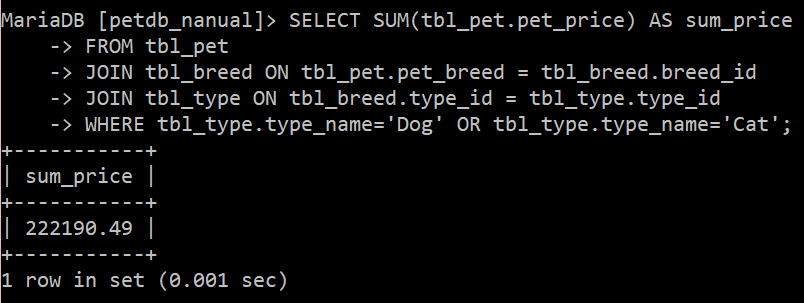
1. **SUM**. Show how much sum of all pets’ price whose type is bird. Show the pet\_type and then the total sum.

**My command prompt:**



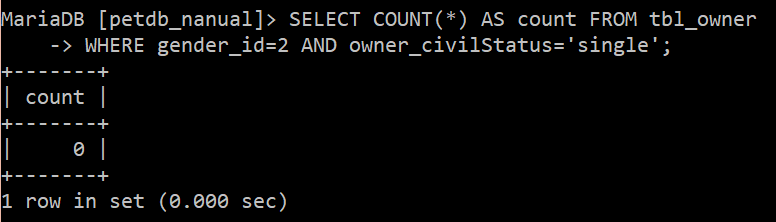
1. **SUM**. Show how much is the total sum of all dogs and cats. Show the total sum of two types.

**My command prompt:**



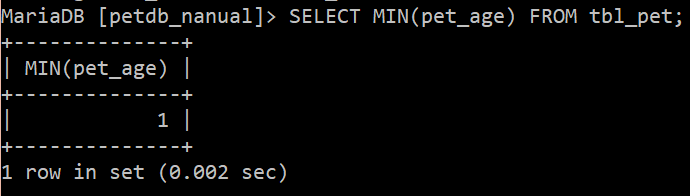
1. **COUNT**. Show how many female single in your owner table.

**My command prompt:**



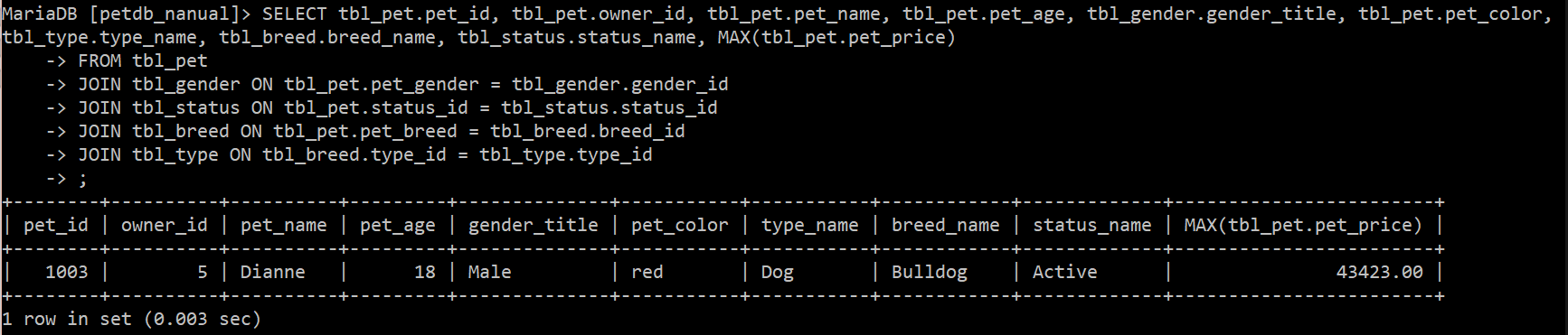
1. **MIN**. Find the youngest pet by showing the age only.

**My command prompt:**



1. **MAX**. Find the most expensive pet.

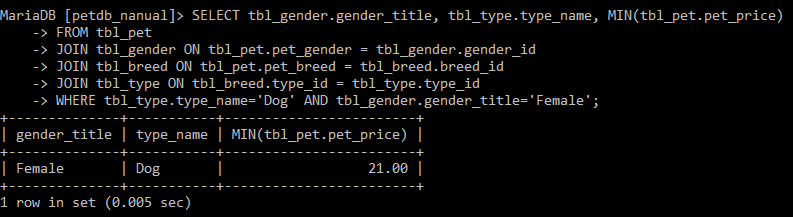
**My command prompt:**



***NOTE that below ad hoc queries may show you wrong output/display, but don’t bother as of the moment since aggregate requires subqueries and group by clause. You can check the output by checking its values.***

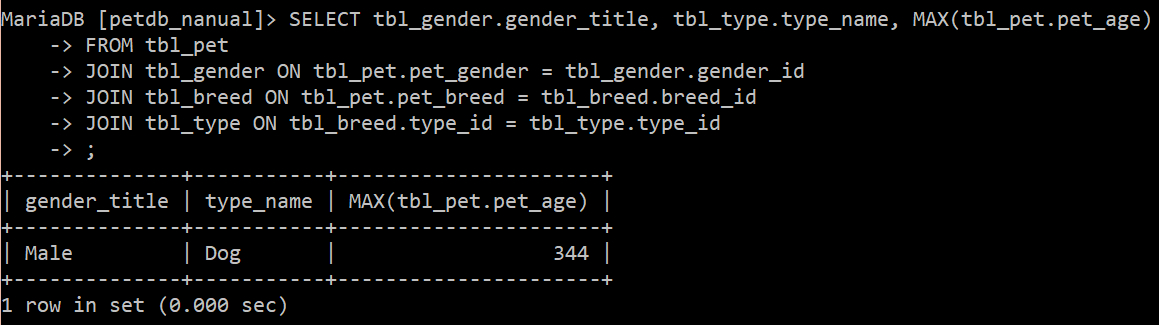
1. **MIN**. Find the cheapest female dog. Show the gender, type, and price.

**My command prompt:**



1. **MAX**. Find the eldest male dog. Show the gender, type and age.

**My command prompt:**

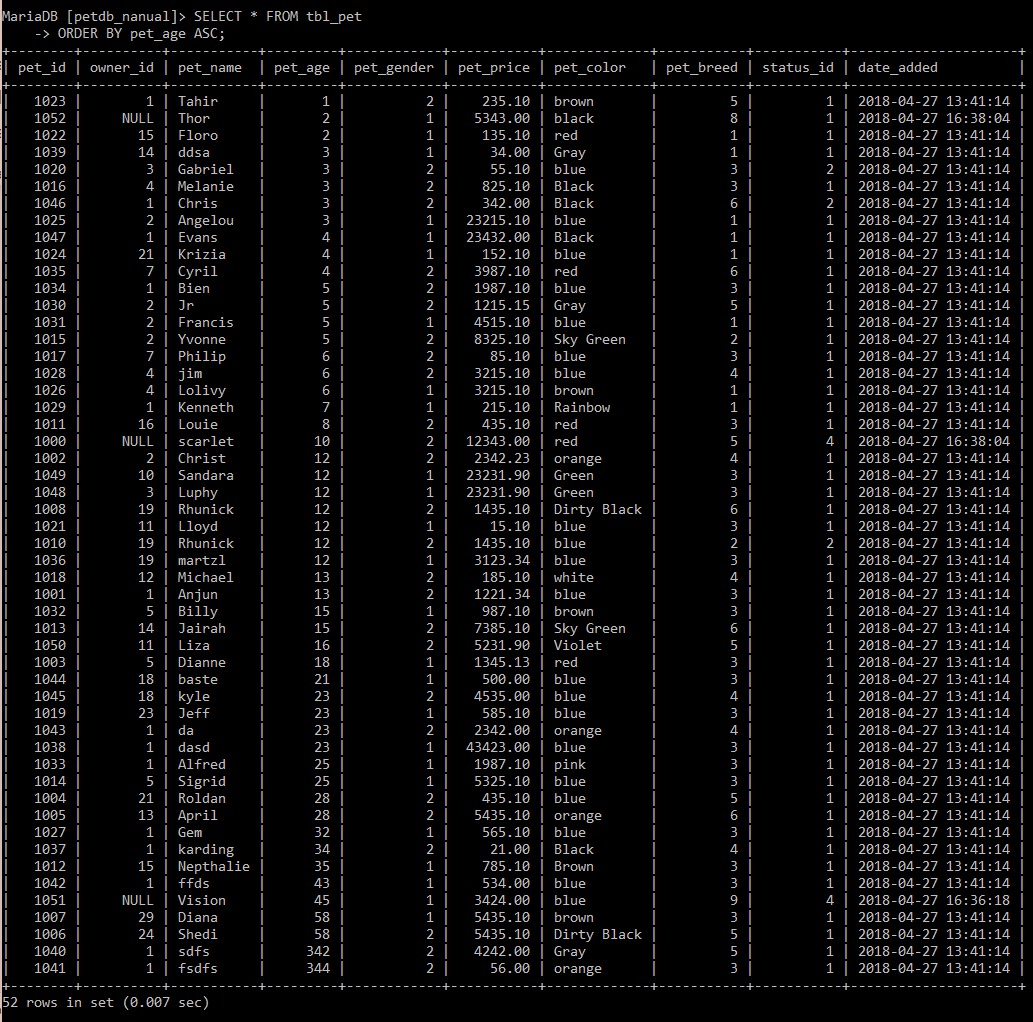


1. **ORDER BY** – show you how to sort the result set using ORDER BY clause. The custom sort order with the FIELD function will be also covered.
2. **GROUP BY** clause – show you how to group rows into groups based on columns or expressions.
3. **HAVING** – filter the groups by a specific condition.

**TASK:**

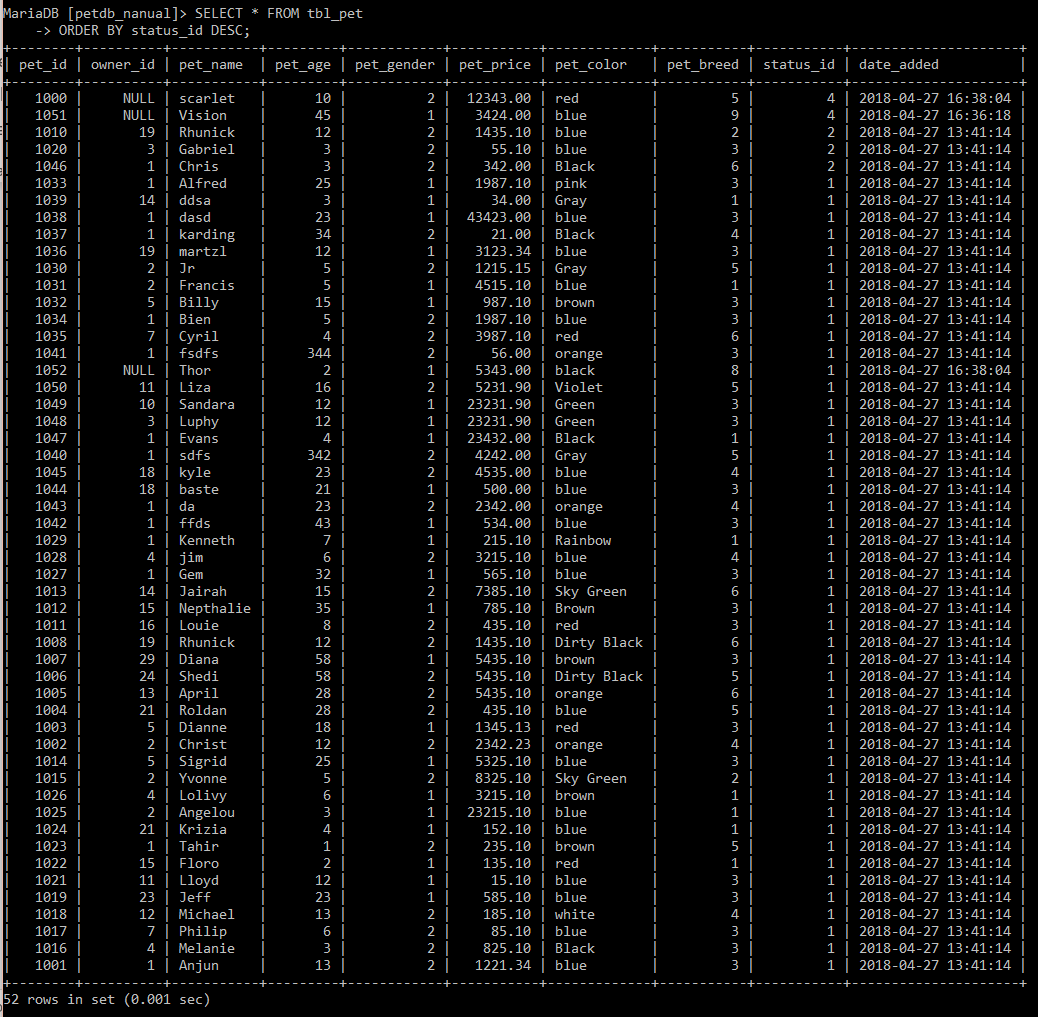
1. Display all pet’s information order by age from lowest to highest.

**My command prompt:**



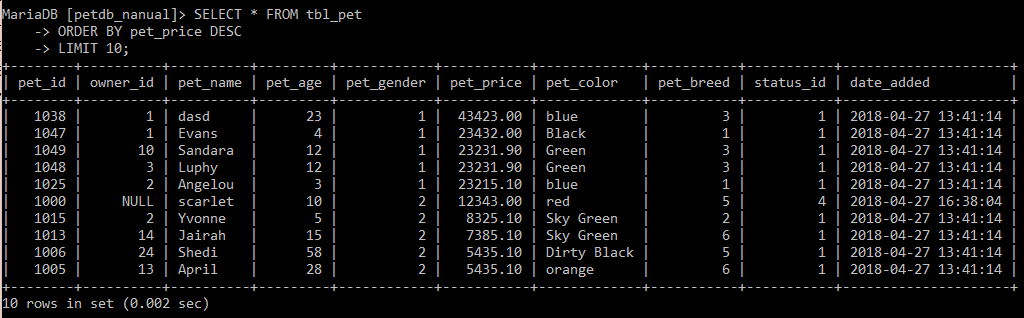
1. Display all pet’s information order by status in descending order.

**My command prompt:**



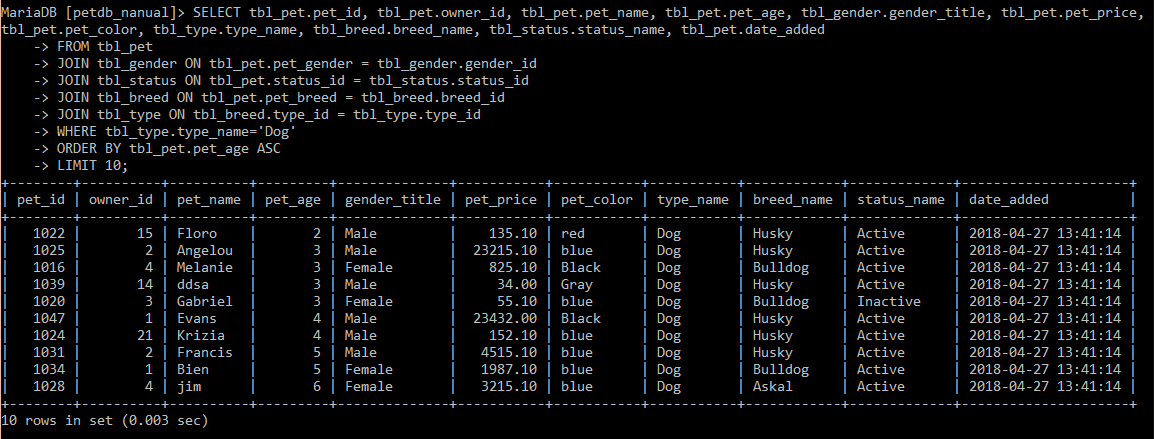
1. Display the information of top 10 most expensive pets.

**My command prompt:**



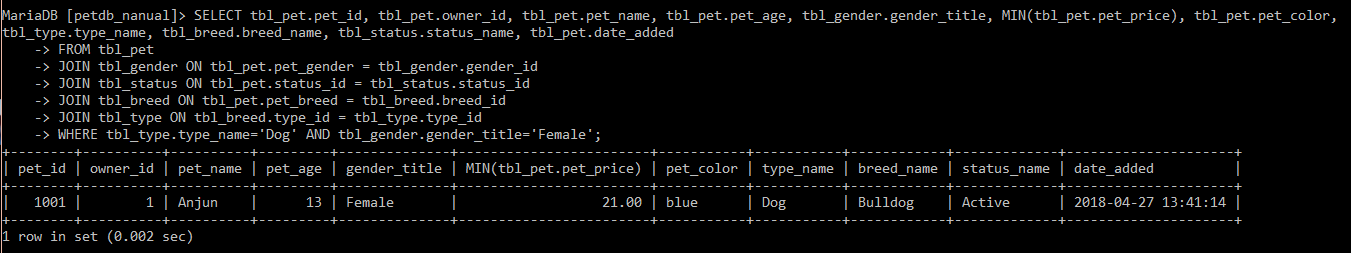
1. Display the information of top 10 youngest pets but only those who are dogs.

**My command prompt:**



1. Find the information of the cheapest female dog.

**My command prompt:**



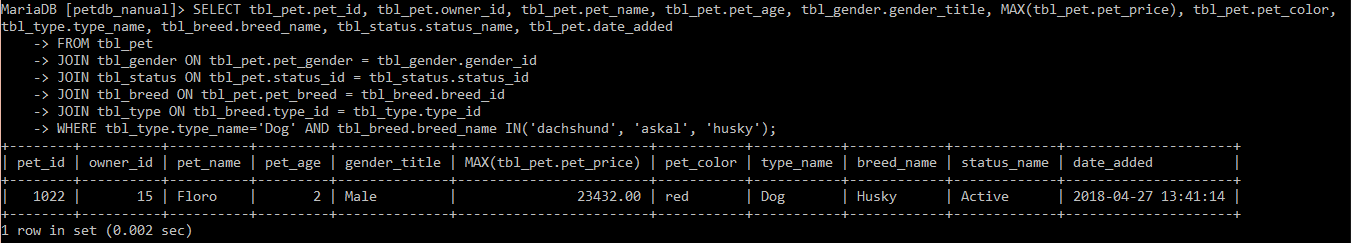
1. Find the information of the youngest dog which/who are inactive.

**My command prompt:**



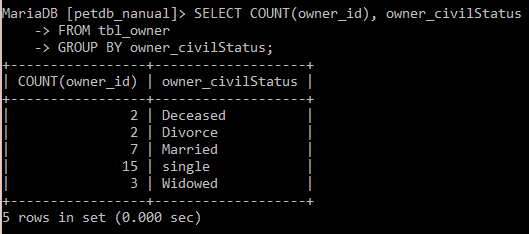
1. Find the most expensive pet whose type is dog, but breed must be either dachshund, askal, and/or Siberian husky.

**My command prompt:**



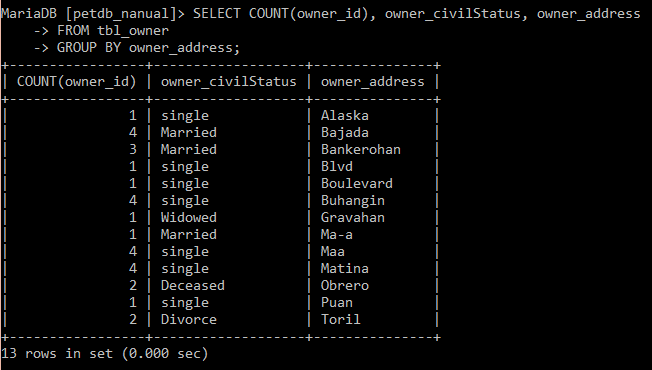
1. Show each how many owners of each civil status.

**My command prompt:**



1. Show each how many owners of each civil status by city.

**My command prompt:**



1. Show how many pets per status.

**My command prompt:**

