Lab: CREATE, ALTER, TRUNCATE, DROP into Tables in MySQL using phpMyAdmin

Estimated time needed: 20 minutes

In this lab, you will learn how to create tables and load data in the MySQL database service using the phpMyAdmin graphical user interface (GUI) tool.

Software Used in this Lab

In this lab, you will use MySQL. MySQL is a Relational Database Management System (RDBMS) designed to efficiently store, manipulate, and retrieve data.

To complete this lab you will utilize MySQL relational database service available as part of IBM Skills Network Labs (SN Labs) Cloud IDE. SN Labs is a virtual lab environment used in this course.

Database Used in this Lab

Mysql_learners database has been used in this lab.

Objectives

After completing this lab, you will be able to use phpMyAdmin with MySQL to:

- Create a database.
- Create a new table in a database.
- Add, delete, or modify columns in an existing table.
- Remove all rows from an existing table without deleting the table itself.
- Delete an existing table in a database

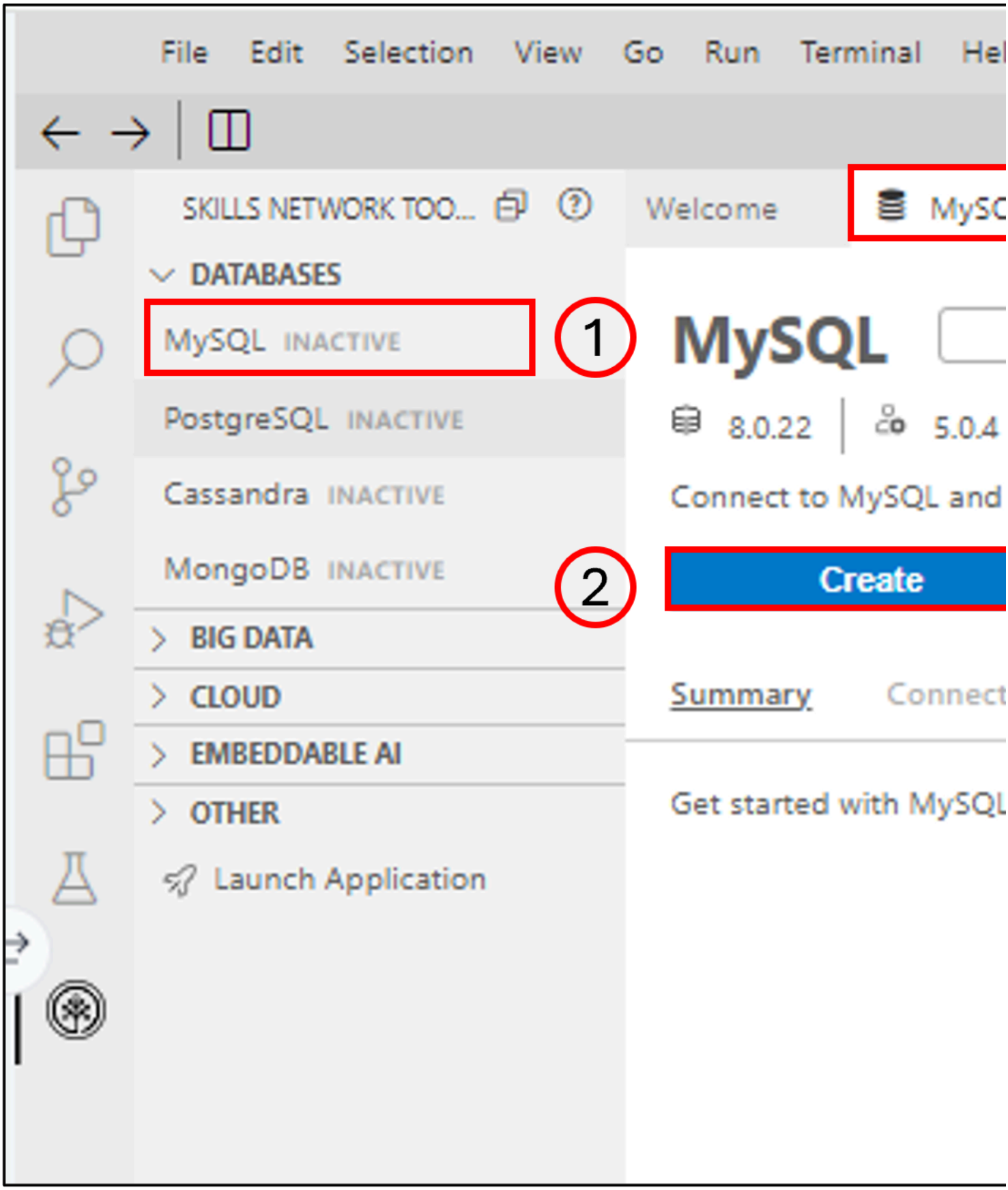
Exercise

In this exercise through different tasks, you will learn how to create tables and load data in the MySQL database service using the phpMyAdmin graphical user interface (GUI) tool.

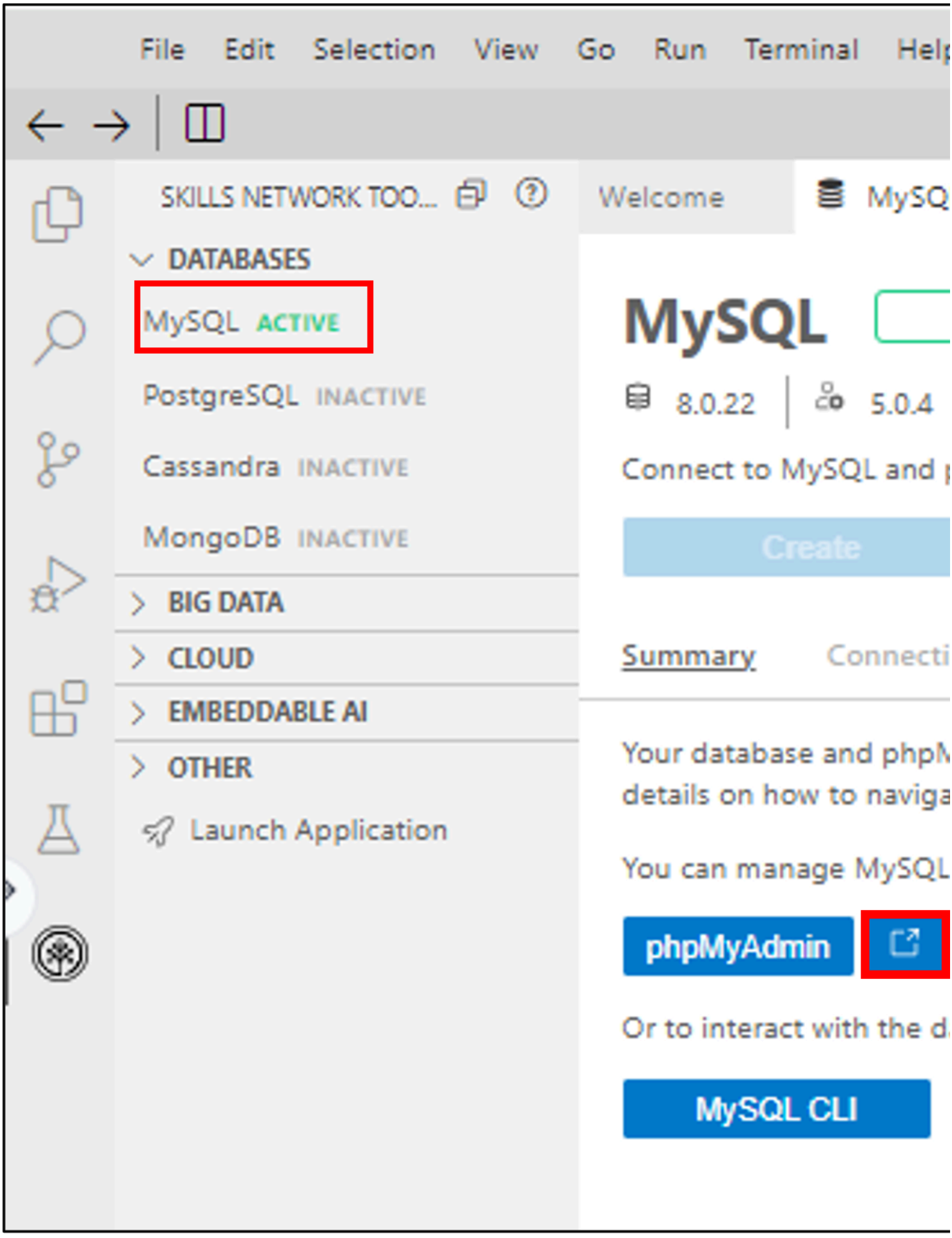
Task A: Create a database

1. Click on **Skills Network Toolbox**. In **Database** section, click **MySQL**.

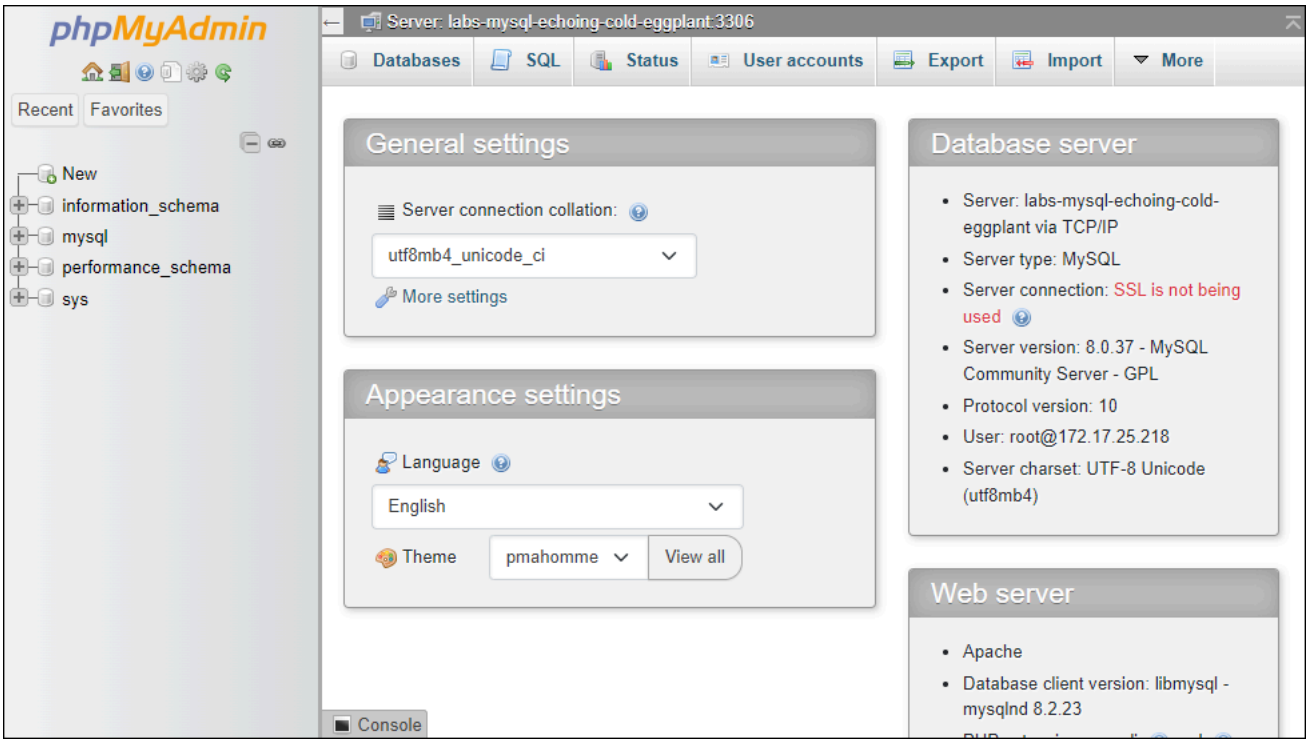
To start the MySQL click **Create**.



2. Once **MySQL** has started, click on **phpMyAdmin button** to open **phpMyAdmin** in new window.

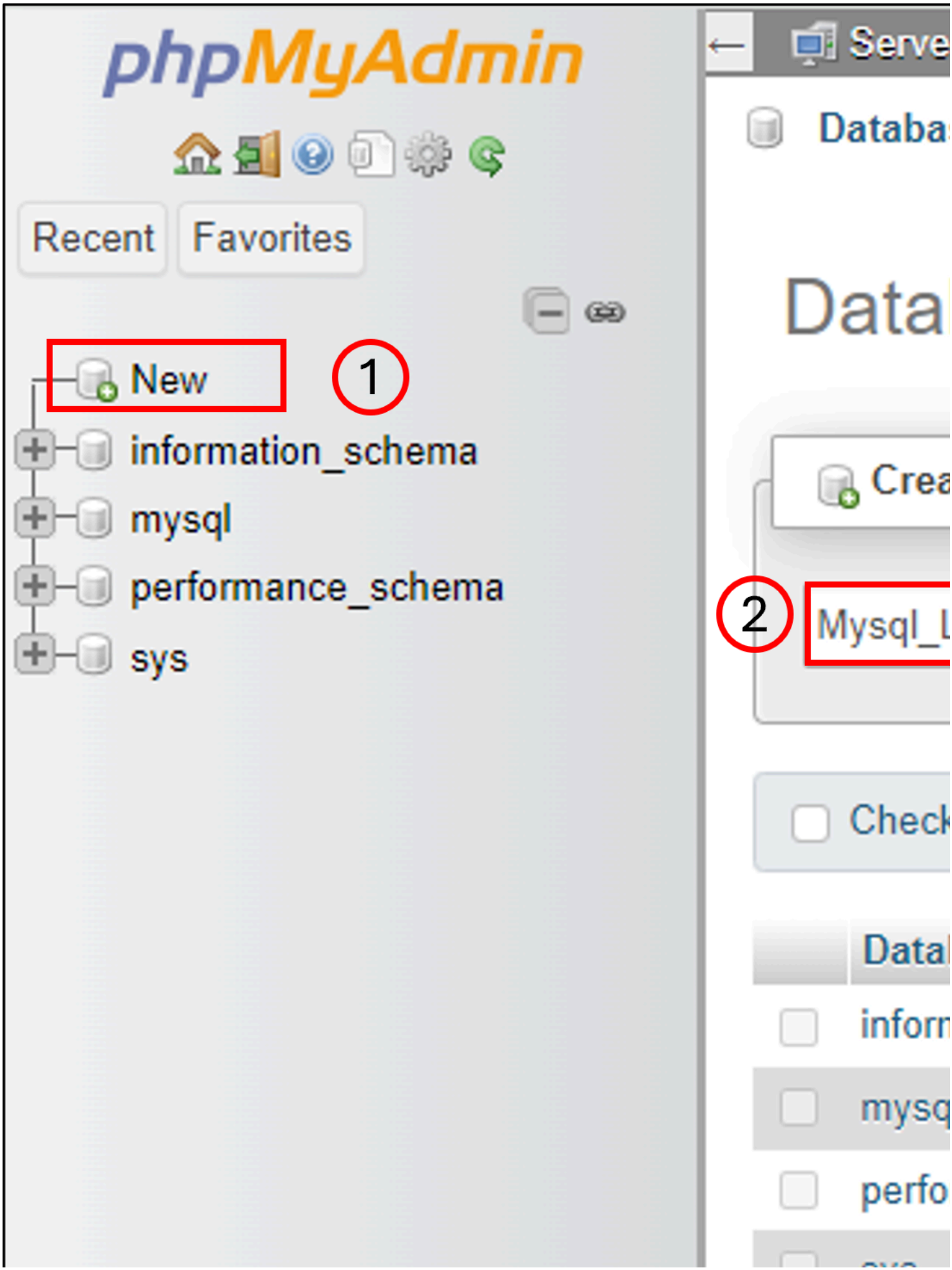


3. You will see the phpMyAdmin GUI tool.



4. In the tree-view, click **New** to create a new empty database. Then enter **Mysql_Learners** as the name of the database and select **utf8mb4_0900_ai_ci** and click **Create**.

UTF-8 is the most commonly used character encoding for content or data.
Proceed to Task B.


phpMyAdmin
Recent
Favorites
New
1
information_schema
mysql
performance_schema
sys
Serve
Databa
Data
Crea
2
Mysql_l
Check
Data
inforn
mysq
perfo

In this lab, you will learn some commonly used DDL (Data Definition Language) statements of SQL. First you will learn the CREATE statement, which is used to create a new table in a database. Next, you will learn the ALTER statement which is used to add, delete, or modify columns in an existing table. Then, you will learn the TRUNCATE statement which is used to remove all rows from an existing table without deleting the table itself. Lastly, you will learn the DROP statement which is used to delete an existing table in a database.

How does the syntax of a CREATE statement look?

```
CREATE TABLE table_name (
    column1 datatype,
    column2 datatype,
    column3 datatype,
   ....
);
```

How does the syntax of an ALTER statement look?

```
ALTER TABLE table_name
ADD COLUMN column_name data_type column_constraint;
ALTER TABLE table_name
DROP COLUMN column_name;
ALTER TABLE table_name
ALTER COLUMN column_name SET DATA TYPE data_type;
ALTER TABLE table_name
CHANGE current_column_name new_column_name;
```

How does the syntax of a TRUNCATE statement look?

```
TRUNCATE TABLE table_name;
```

How does the syntax of a DROP statement look?

```
DROP TABLE table_name;
```

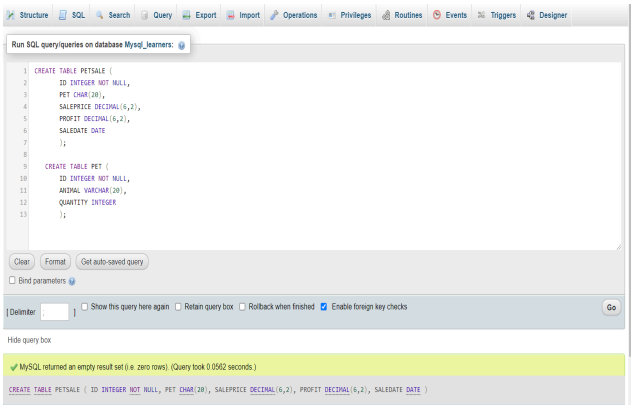
In this exercise, you will use the CREATE statement to create two new tables using Db2.

1. You need to create two tables, **PETSALE** and **PET**. To create the two tables PETSALE and PET, copy the code below and paste it to the textarea of the **SQL** page. Click **Go**.

```
CREATE TABLE PETSALE (
```

```
    ID INTEGER NOT NULL,
    PET CHAR(20),
    SALEPRICE DECIMAL(6,2),
    PROFIT DECIMAL(6,2),
    SALEDATE DATE
    );

CREATE TABLE PET (
    ID INTEGER NOT NULL,
    ANIMAL VARCHAR(20),
    QUANTITY INTEGER
    );
```

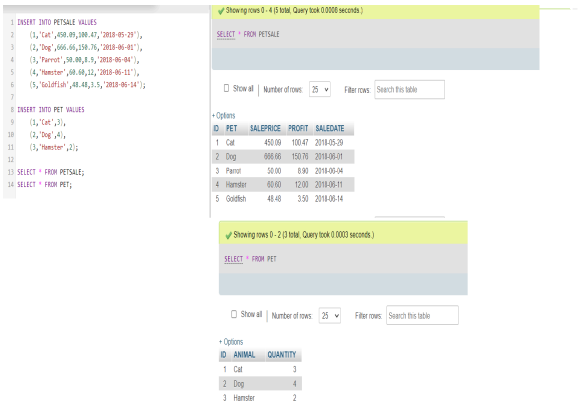


2. Now insert some records into the two newly created tables and show all the records of the two tables. Copy the code below and paste it to the textarea of the **SQL** page. Click **Go**.

```
INSERT INTO PETSALE VALUES
    (1,'Cat',450.09,100.47,'2018-05-29'),
    (2,'Dog',666.66,150.76,'2018-06-01'),
    (3,'Parrot',50.00,8.9,'2018-06-04'),
    (4,'Hamster',60.60,12,'2018-06-11'),
    (5,'Goldfish',48.48,3.5,'2018-06-14');

INSERT INTO PET VALUES
    (1,'Cat',3),
    (2,'Dog',4),
    (3,'Hamster',2);

SELECT * FROM PETSALE;
SELECT * FROM PET;
```

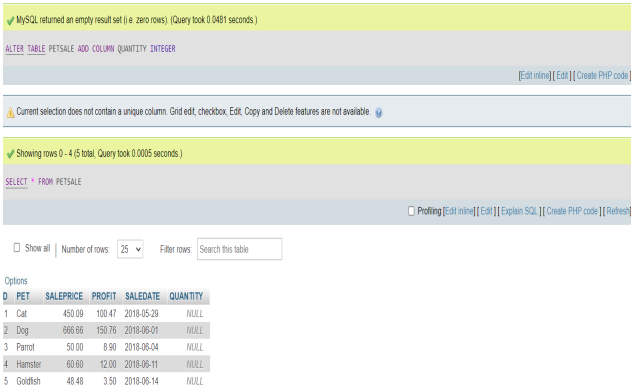


In this exercise, you will use the ALTER statement to add, delete, or modify columns in two of the existing tables created in exercise 1.

Task A: ALTER using ADD COLUMN

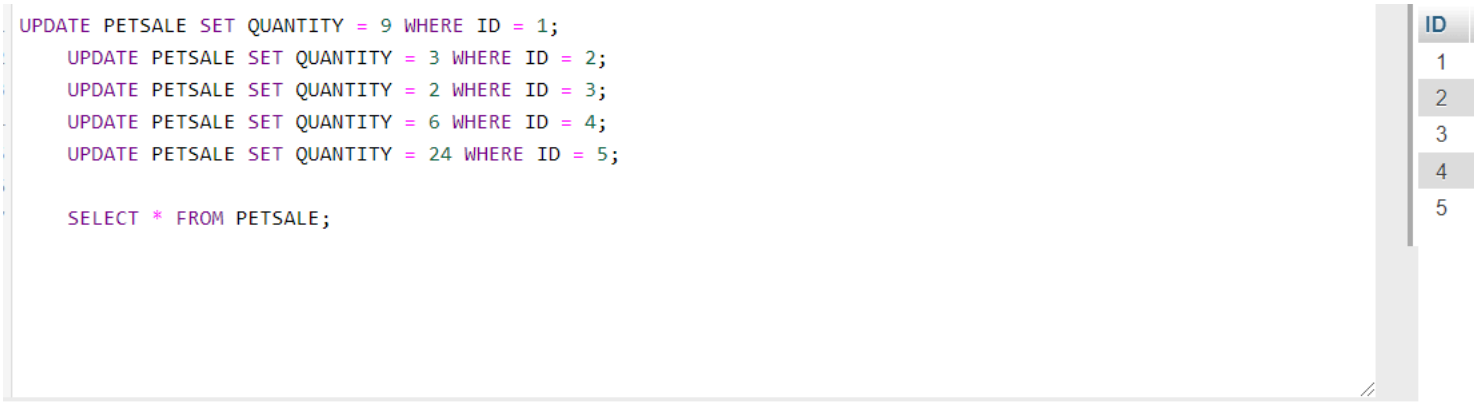
1. Add a new **QUANTITY** column to the **PETSALE** table and show the altered table. Copy the code below and paste it to the textarea of the **SQL** page. Click **Go**..

```
ALTER TABLE PETSALE
ADD COLUMN QUANTITY INTEGER;
SELECT * FROM PETSALE;
```



2. Now update the newly added **QUANTITY** column of the **PETSALE** table with some values and show all the records of the table. Copy the code below and paste it to textarea of the **SQL** page. Click **Go**.

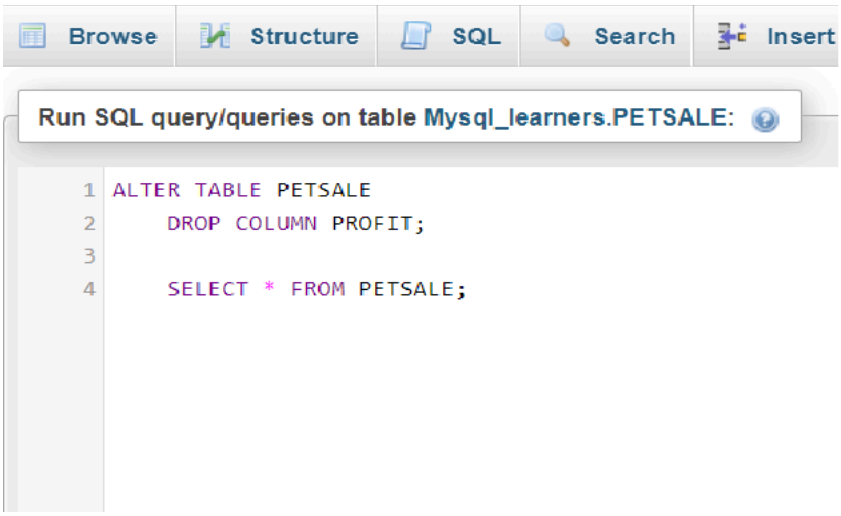
```
UPDATE PETSALE SET QUANTITY = 9 WHERE ID = 1;
UPDATE PETSALE SET QUANTITY = 3 WHERE ID = 2;
UPDATE PETSALE SET QUANTITY = 2 WHERE ID = 3;
UPDATE PETSALE SET QUANTITY = 6 WHERE ID = 4;
UPDATE PETSALE SET QUANTITY = 24 WHERE ID = 5;
SELECT * FROM PETSALE;
```

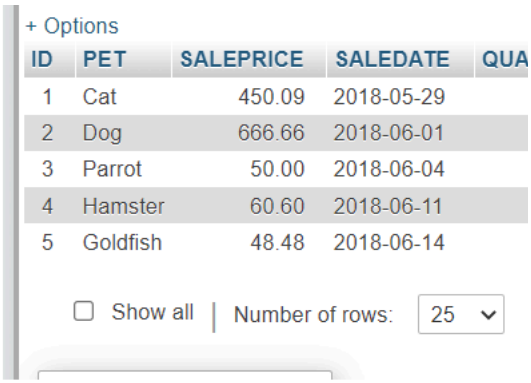


Task B: ALTER using DROP COLUMN

1. Delete the **PROFIT** column from the **PETSALE** table and show the altered table. Copy the code below and paste it to the textarea of the **SQL** page. Click **Go**.

```
ALTER TABLE PETSALE
DROP COLUMN PROFIT;
SELECT * FROM PETSALE;
```

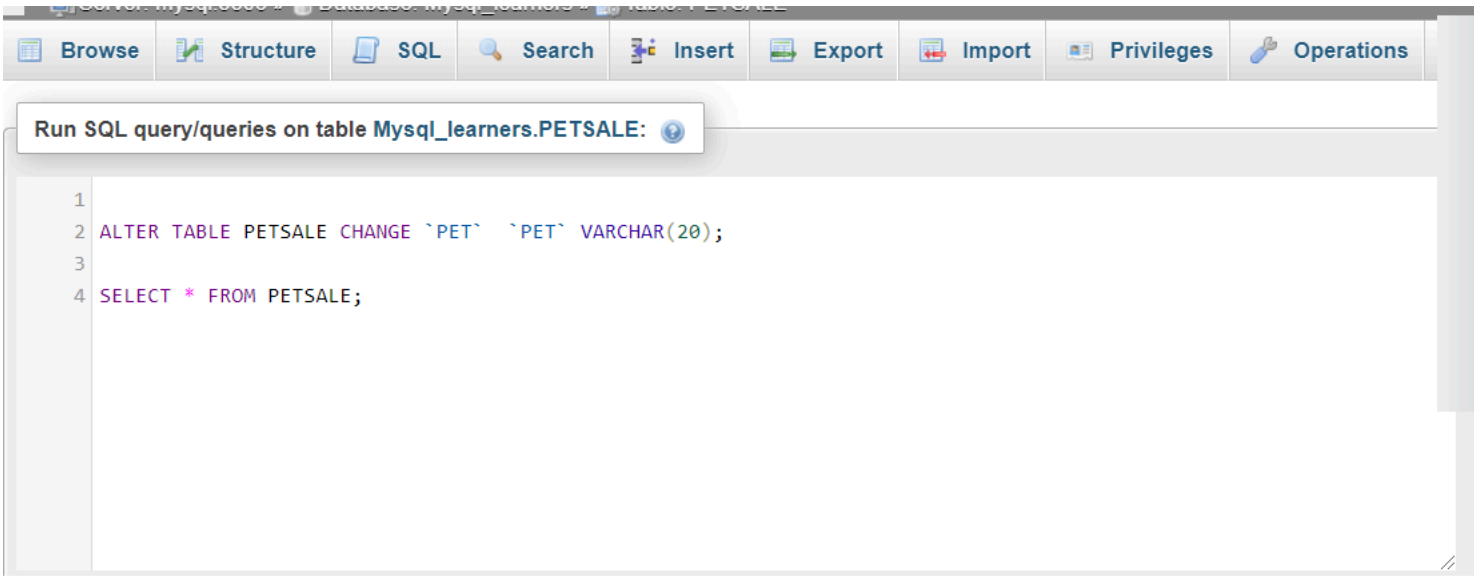


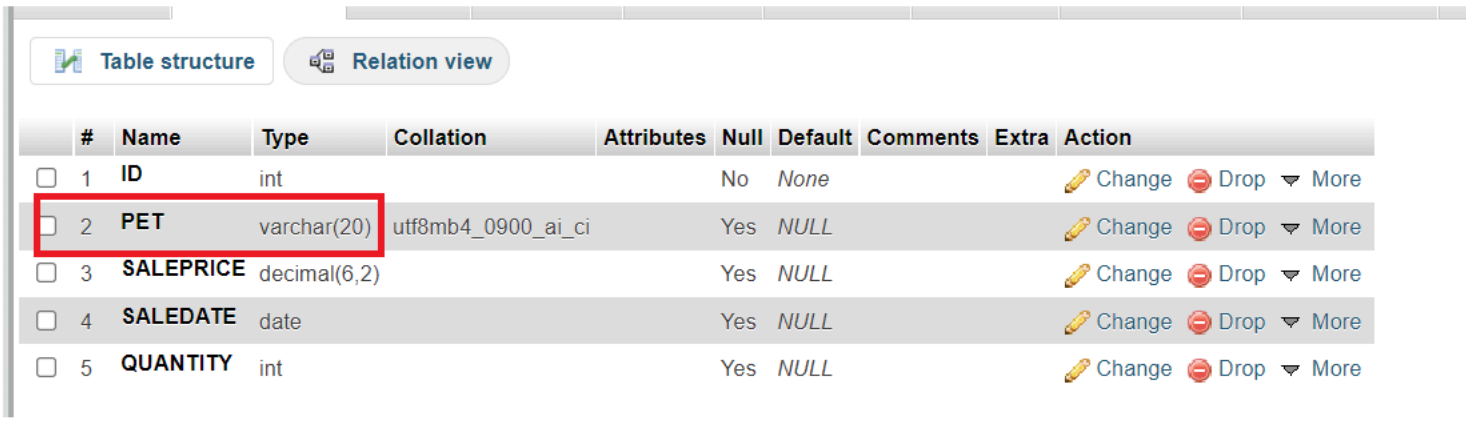


Task C: ALTER using ALTER COLUMN

1. Change the data type to **VARCHAR(20)** type of the column **PET** of the table **PETSALE** and show the altered table. Copy the code below and paste it to the textarea of the **SQL** page. Click **Go**.

```
ALTER TABLE PETSALE CHANGE PET  PET VARCHAR(20);
SELECT * FROM PETSALE;
```

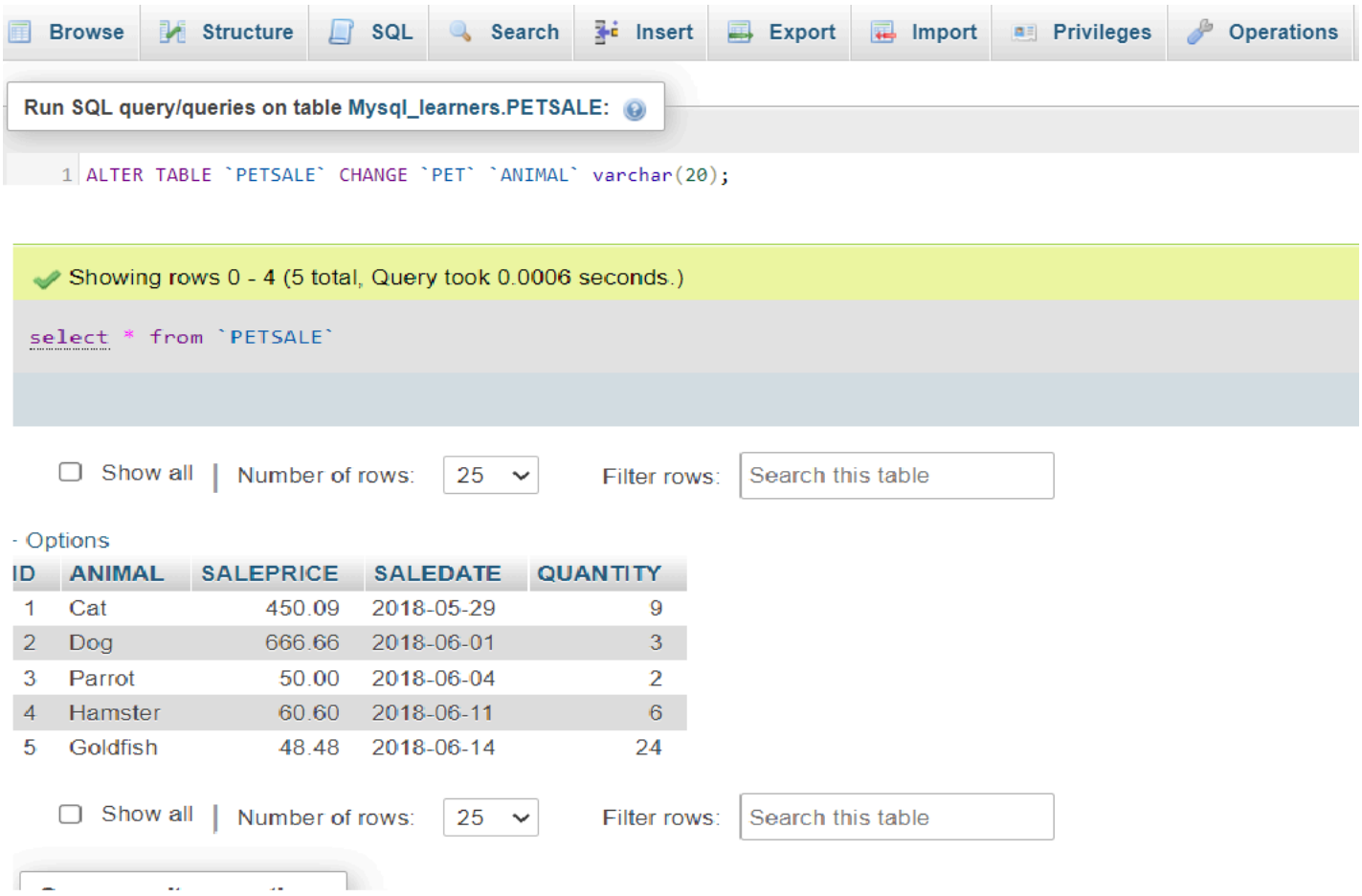




Task D: ALTER using RENAME COLUMN

1. Rename the column **PET** to **ANIMAL** of the **PETSALE** table and show the altered table. Copy the code below and paste it to the textarea of the **SQL** page. Click **Go**.

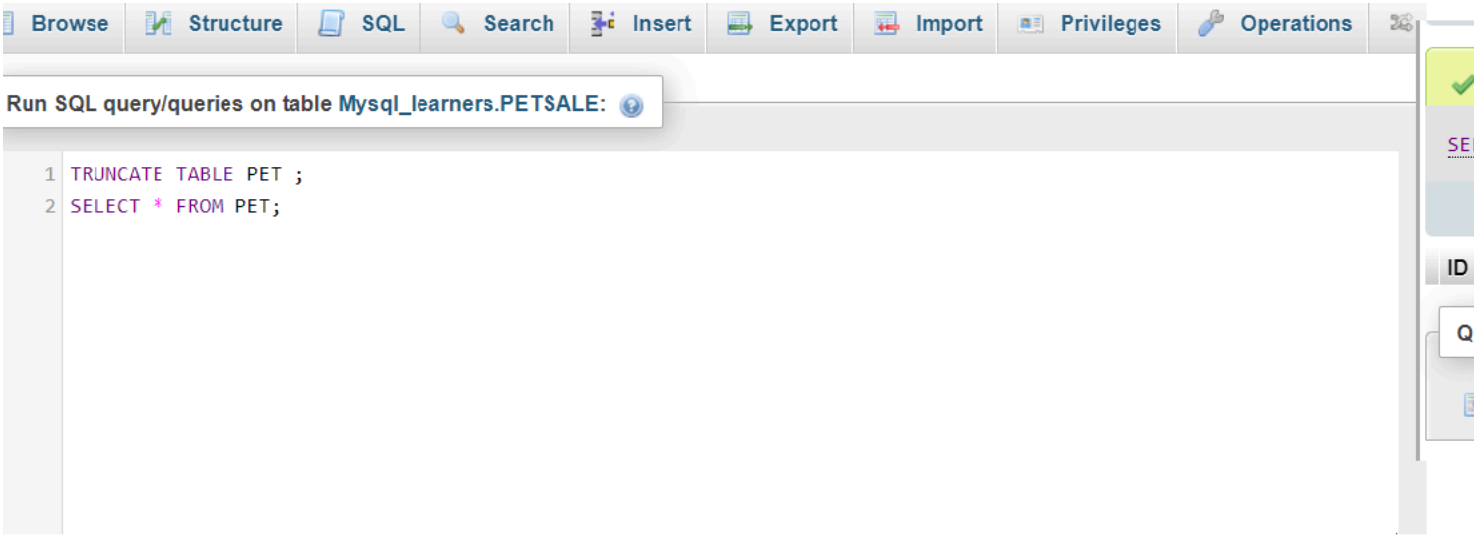
```
ALTER TABLE `PETSALE` CHANGE `PET` `ANIMAL` varchar(20);
SELECT * FROM PETSALE;
```



In this exercise, you will use the TRUNCATE statement to remove all rows from an existing table created in exercise 1 without deleting the table itself.

1. Remove all rows from the **PET** table and show the empty table. Copy the code below and paste it to the textarea of the **SQL** page. Click **Go**.

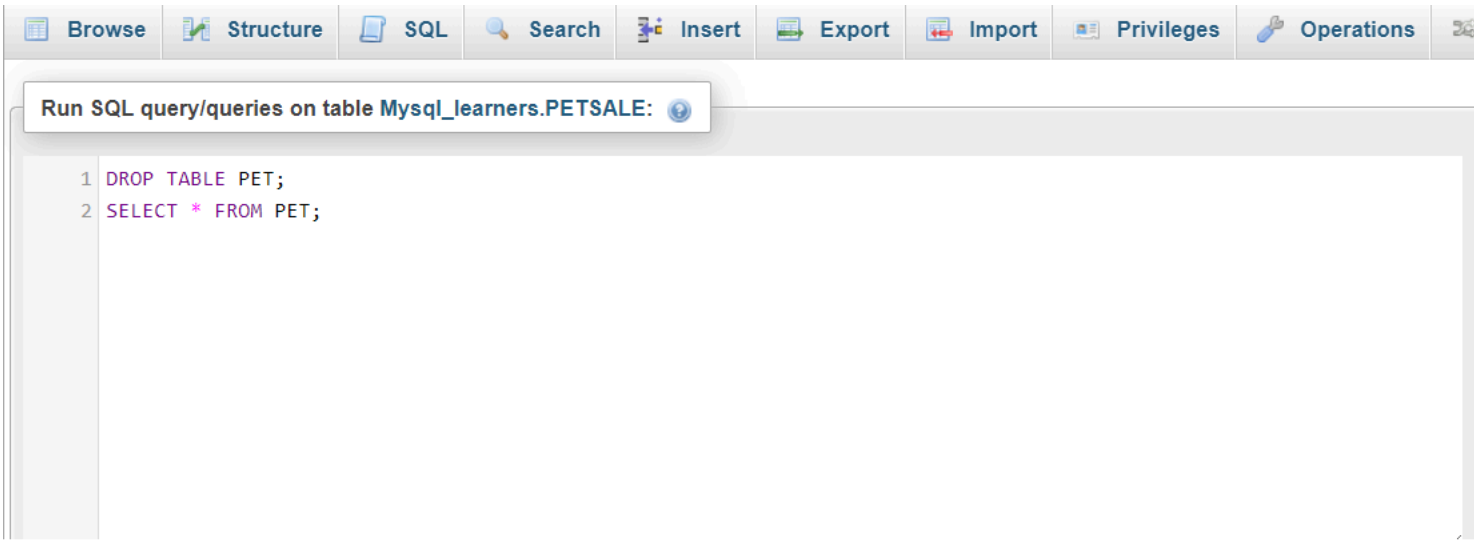
```
TRUNCATE TABLE PET ;
SELECT * FROM PET;
```



In this exercise, you will use the DROP statement to delete an existing table created in exercise 1.

1. Delete the **PET** table and verify if the table still exists or not (SELECT statement won't work if a table doesn't exist). Copy the code below and paste it to the textarea of the **SQL** page. Click **Go**.

```
DROP TABLE PET;
SELECT * FROM PET;
```



Congratulations! You have completed this lab, and you are ready for the next topic.

Lakshmi Holla

Malika Singla