

# Hands-on 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

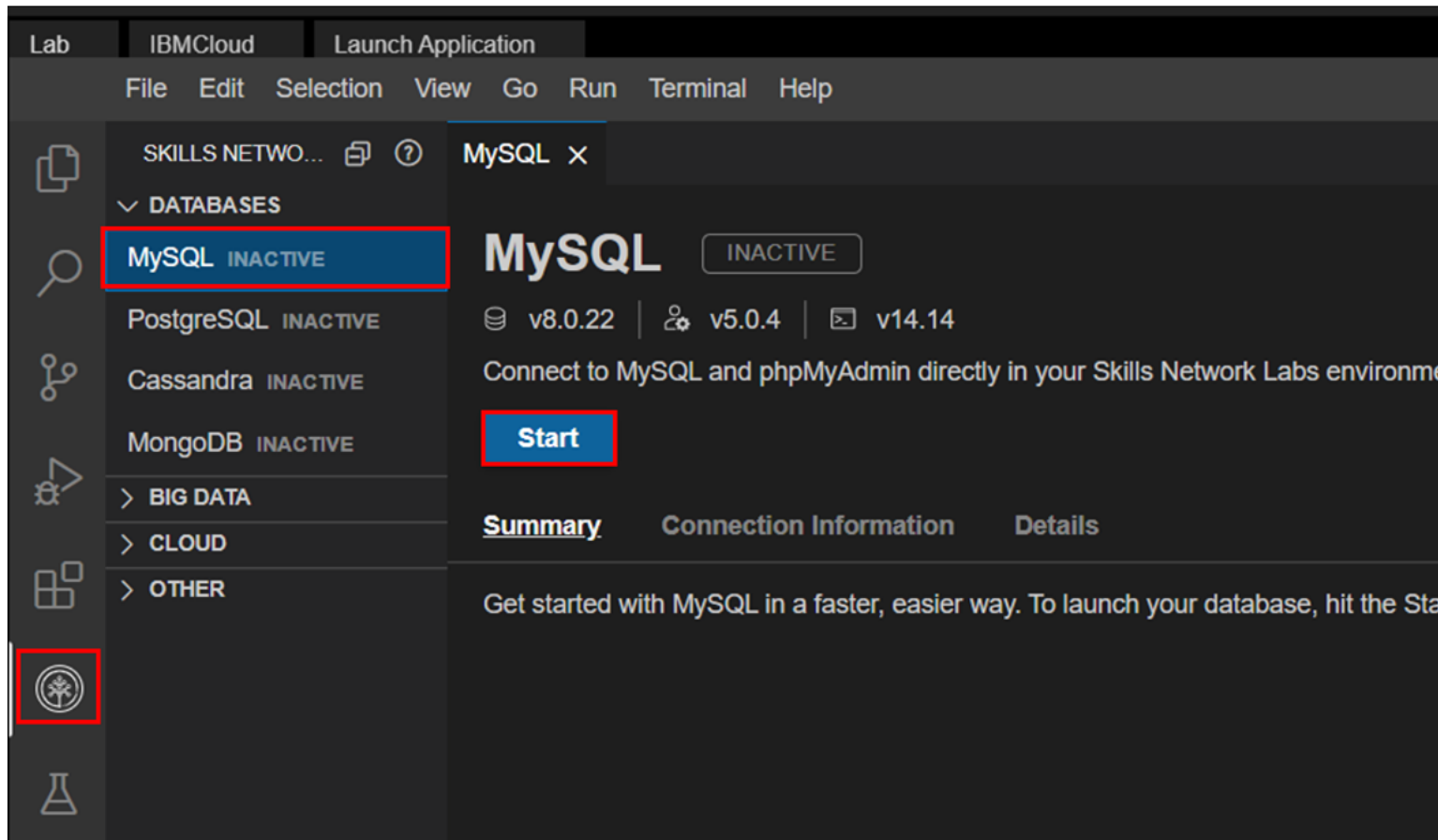
## Exercise

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.

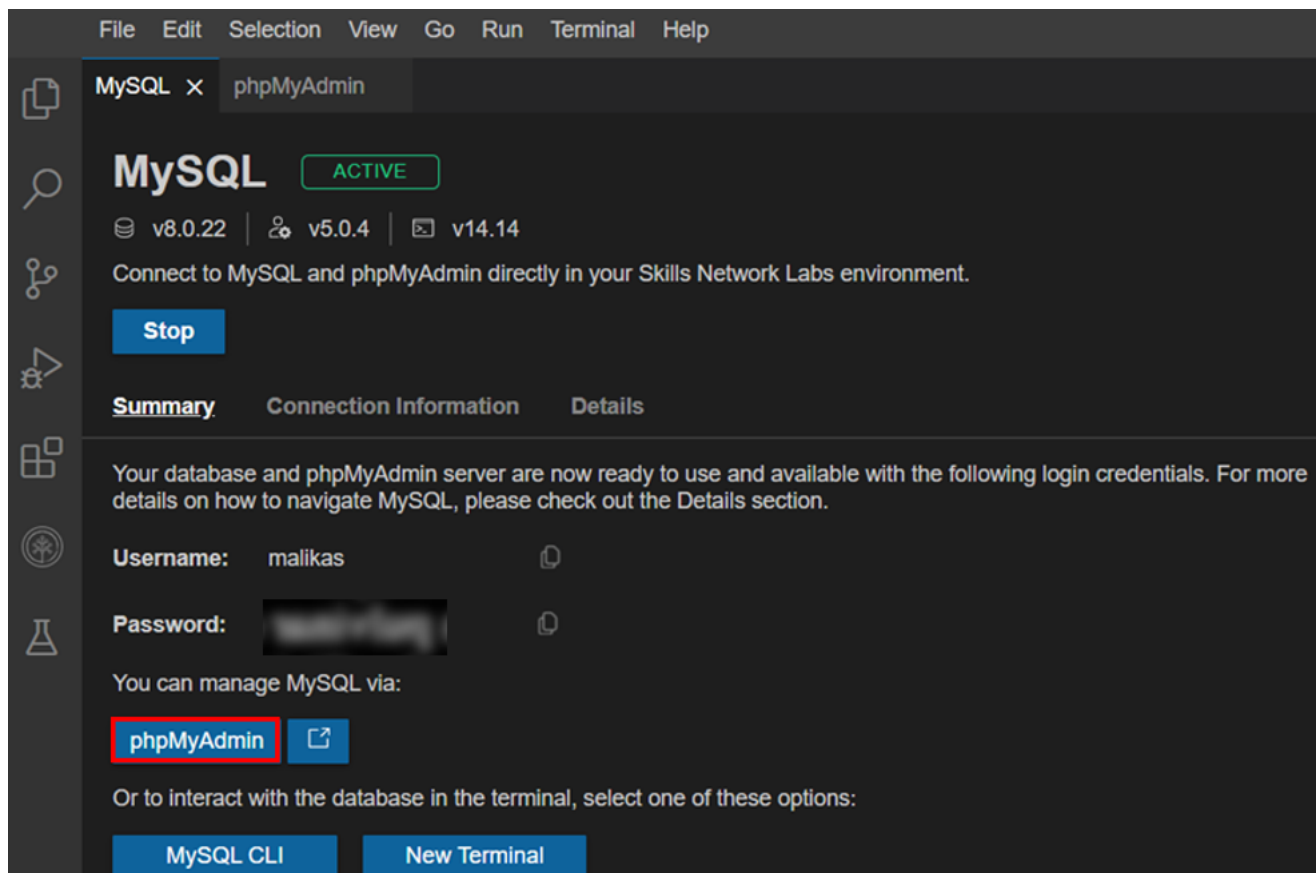
### Task A: Create a database

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

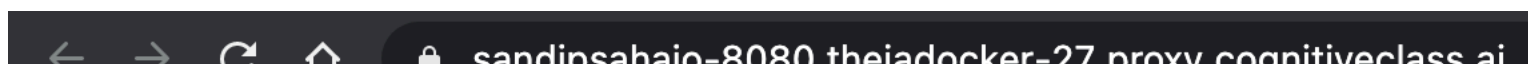
To start the MySQL click **Start**.



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



3. You will see the phpMyAdmin GUI tool.





Recent

Favorites

- New
- information\_schema
- mysql
- performance\_schema
- sakila
- sys



Server: mysql:3306



Databases



SQL



Status



Us

## General settings



Server connection collation: ?

utf8mb4\_unicc



More settings

## Appearance settings



Language ?

English



Theme:

pmahomme ▼

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 **utf8\_general\_ci** and click **Create**.

UTF-8 is the most commonly used character encoding for content or data.

Proceed to Task B.

Databases

SQL

Status

User accounts

Export

Import

# Databases

Create database

Mysql\_Learners

utf8\_general\_ci

Create

	Database	Collation	Master replication	Action
<input type="checkbox"/>	information_schema	utf8_general_ci	✓ Replicated	Check privileges
<input type="checkbox"/>	mysql	utf8mb4_0900_ai_ci	✓ Replicated	Check privileges
<input type="checkbox"/>	performance_schema	utf8mb4_0900_ai_ci	✓ Replicated	Check privileges
<input type="checkbox"/>	sys	utf8mb4_0900_ai_ci	✓ Replicated	Check privileges
Total: 4				

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?

```
1. 1
2. 2
3. 3
4. 4
5. 5
6. 6

1. CREATE TABLE table_name (
2.     column1 datatype,
3.     column2 datatype,
4.     column3 datatype,
5.     ....
6. );
```

Copied!

### How does the syntax of an ALTER statement look?

```
1. 1
2. 2
3. 3
4. 4
5. 5
6. 6
7. 7
8. 8
9. 9
10. 10
11. 11

1. ALTER TABLE table_name
2. ADD COLUMN column_name data_type column_constraint;
3.
4. ALTER TABLE table_name
5. DROP COLUMN column_name;
6.
7. ALTER TABLE table_name
8. ALTER COLUMN column_name SET DATA TYPE data_type;
9.
10. ALTER TABLE table_name
11. CHANGE current_column_name new_column_name;
```

Copied!

## How does the syntax of a TRUNCATE statement look?

- 1
1. TRUNCATE TABLE table\_name;

Copied!

## How does the syntax of a DROP statement look?

- 1
1. DROP TABLE table\_name;

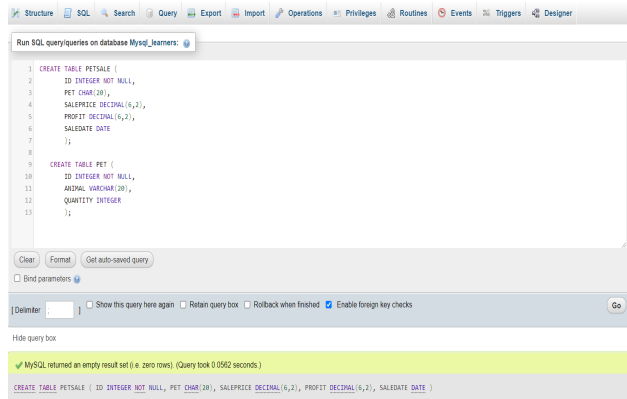
Copied!

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**.

```
1. 1
2. 2
3. 3
4. 4
5. 5
6. 6
7. 7
8. 8
9. 9
10. 10
11. 11
12. 12
13. 13
1. CREATE TABLE PETSALE (
2.     ID INTEGER NOT NULL,
3.     PET CHAR(20),
4.     SALEPRICE DECIMAL(6,2),
5.     PROFIT DECIMAL(6,2),
6.     SALEDATE DATE
7. );
8.
9. CREATE TABLE PET (
10.     ID INTEGER NOT NULL,
11.     ANIMAL VARCHAR(20),
12.     QUANTITY INTEGER
13. );
```

Copied!



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**.

```
1. 1
2. 2
3. 3
4. 4
5. 5
6. 6
7. 7
8. 8
9. 9
10. 10
11. 11
12. 12
13. 13
14. 14
1. INSERT INTO PETSALE VALUES
2.     (1, 'Cat', 450.09, 100.47, '2018-05-29'),
3.     (2, 'Dog', 666.66, 150.76, '2018-06-01'),
4.     (3, 'Parrot', 50.00, 8.9, '2018-06-04'),
5.     (4, 'Hamster', 60.60, 12, '2018-06-11'),
6.     (5, 'Goldfish', 48.48, 3.5, '2018-06-14');
7.
8. INSERT INTO PET VALUES
```

```

9.      (1, 'Cat', 3),
10.     (2, 'Dog', 4),
11.     (3, 'Hamster', 2);
12.
13. SELECT * FROM PETSale;
14. SELECT * FROM PET;

```

Copied!

1 INSERT INTO PETSale VALUES  
2 (1, 'Cat', 450.00, 100.47, '2018-05-29'),  
3 (2, 'Dog', 666.66, 150.76, '2018-06-01'),  
4 (3, 'Parrot', 50.00, 8.90, '2018-06-04'),  
5 (4, 'Hamster', 60.60, 12.00, '2018-06-11'),  
6 (5, 'Goldfish', 48.48, 3.50, '2018-06-14');  
7  
8 INSERT INTO PET VALUES  
9 (1, 'Cat', 3),  
10 (2, 'Dog', 4),  
11 (3, 'Hamster', 2);  
12  
13 SELECT \* FROM PETSale;  
14 SELECT \* FROM PET;

Showing rows 0 - 4 (5 total, Query took 0.000 seconds)

SELECT \* FROM PETSale

☐ Show all | Number of rows: 25 | Filter rows | Search this table

Options

ID	PET	SALEPRICE	PROFIT	SALEDATE
1	Cat	450.00	100.47	2018-05-29
2	Dog	666.66	150.76	2018-06-01
3	Parrot	50.00	8.90	2018-06-04
4	Hamster	60.60	12.00	2018-06-11
5	Goldfish	48.48	3.50	2018-06-14

Showing rows 0 - 2 (3 total, Query took 0.000 seconds)

SELECT \* FROM PET

☐ Show all | Number of rows: 25 | Filter rows | Search this table

Options

ID	ANIMAL	QUANTITY
1	Cat	3
2	Dog	4
3	Hamster	2

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..**

```

1. 1
2. 2
3. 3
4. 4
5. 1. ALTER TABLE PETSale
6. 2. ADD COLUMN QUANTITY INTEGER;
7. 3.
8. 4. SELECT * FROM PETSale;

```

Copied!

MySQL returned an empty result set (i.e. zero rows). (Query took 0.0481 seconds)

ALTER TABLE PETSale ADD COLUMN QUANTITY INTEGER

[\[Edit sql\]](#)
[\[Edit\]](#)
[\[Create PHP code\]](#)

Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete features are not available.

Showing rows 0 - 4 (5 total, Query took 0.0005 seconds)

SELECT \* FROM PETSale

☐ Profiling
[\[Edit sql\]](#)
[\[Edit\]](#)
[\[Explain SQL\]](#)
[\[Create PHP code\]](#)
[\[Refresh\]](#)

☐ Show all | Number of rows: 25 | Filter rows | Search this table

Options

ID	PET	SALEPRICE	PROFIT	SALEDATE	QUANTITY
1	Cat	450.00	100.47	2018-05-29	NULL
2	Dog	666.66	150.76	2018-06-01	NULL
3	Parrot	50.00	8.90	2018-06-04	NULL
4	Hamster	60.60	12.00	2018-06-11	NULL
5	Goldfish	48.48	3.50	2018-06-14	NULL

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.**

```

1. 1
2. 2
3. 3
4. 4
5. 5
6. 6
7. 7
8. 1. UPDATE PETSale SET QUANTITY = 9 WHERE ID = 1;
9. 2. UPDATE PETSale SET QUANTITY = 3 WHERE ID = 2;
10. 3. UPDATE PETSale SET QUANTITY = 2 WHERE ID = 3;
11. 4. UPDATE PETSale SET QUANTITY = 6 WHERE ID = 4;
12. 5. UPDATE PETSale SET QUANTITY = 24 WHERE ID = 5;
13. 6.
14. 7. SELECT * FROM PETSale;

```

Copied!

```

UPDATE PETALE SET QUANTITY = 9 WHERE ID = 1;
UPDATE PETALE SET QUANTITY = 3 WHERE ID = 2;
UPDATE PETALE SET QUANTITY = 2 WHERE ID = 3;
UPDATE PETALE SET QUANTITY = 6 WHERE ID = 4;
UPDATE PETALE SET QUANTITY = 24 WHERE ID = 5;

SELECT * FROM PETALE;

```

ID	PET
1	Cat
2	Dog
3	Parro
4	Hams
5	Goldf

## Task B: ALTER using DROP COLUMN

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

```

1. 1
2. 2
3. 3
4. 4
1. ALTER TABLE PETALE
2. DROP COLUMN PROFIT;
3.
4. SELECT * FROM PETALE;

```

Copied!

Browse
 Structure
 SQL
 Search
 Insert

Run SQL query/queries on table Mysql\_learners.PETALE:

```

1 ALTER TABLE PETALE
2   DROP COLUMN PROFIT;
3
4   SELECT * FROM PETALE;

```

+ Options

ID	PET	SALEPRICE	SALEDATE	QUANTIT
1	Cat	450.09	2018-05-29	
2	Dog	666.66	2018-06-01	
3	Parrot	50.00	2018-06-04	
4	Hamster	60.60	2018-06-11	
5	Goldfish	48.48	2018-06-14	2

☐ Show all
 | Number of rows: 25
 
 Fil

## Task C: ALTER using ALTER COLUMN

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

```

1. 1
2. 2
1. ALTER TABLE PETALE CHANGE PET PET VARCHAR(20);
2. SELECT * FROM PETALE;

```

Copied!

Browse
Structure
SQL
Search
Insert
Export
Import
Privileges
Operations

Run SQL query/queries on table Mysql\_learners.PETSALE:

```

1
2 ALTER TABLE PETSALE CHANGE `PET` `PET` VARCHAR(20);
3
4 SELECT * FROM PETSALE;

```

+ O
ID
1
2
3
4
5

Table structure
Relation view

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/> 1	ID	int			No	None			Change  Drop  More
<input type="checkbox"/> 2	PET	varchar(20)	utf8mb4_0900_ai_ci		Yes	NULL			Change  Drop  More
<input type="checkbox"/> 3	SALEPRICE	decimal(6,2)			Yes	NULL			Change  Drop  More
<input type="checkbox"/> 4	SALEDATE	date			Yes	NULL			Change  Drop  More
<input type="checkbox"/> 5	QUANTITY	int			Yes	NULL			Change  Drop  More

## 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**.

```

1. 1
2. 2
3. 3
1. ALTER TABLE `PETSALE` CHANGE `PET` `ANIMAL` varchar(20);
2.
3. SELECT * FROM PETSALE;

```

Copied!



Run SQL query/queries on table Mysql\_learners.PETSALE:

```
1 ALTER TABLE `PETSALE` CHANGE `PET` `ANIMAL` varchar(20);
```

✓ Showing rows 0 - 4 (5 total, Query took 0.0006 seconds.)

```
select * from `PETSALE`
```

☐ Show all | Number of rows: 25 | Filter rows: Search this table

- Options

ID	ANIMAL	SALEPRICE	SALEDATE	QUANTITY
1	Cat	450.09	2018-05-29	9
2	Dog	666.66	2018-06-01	3
3	Parrot	50.00	2018-06-04	2
4	Hamster	60.60	2018-06-11	6
5	Goldfish	48.48	2018-06-14	24

☐ Show all | Number of rows: 25 | Filter rows: Search this table

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**.

```
1. 1
2. 2
3. 3
1. TRUNCATE TABLE PET ;
2.
3. SELECT * FROM PET;
```

Copied!

[Browse](#)
[Structure](#)
[SQL](#)
[Search](#)
[Insert](#)
[Export](#)
[Import](#)
[Privileges](#)
[Operations](#)

Run SQL query/queries on table Mysql\_learners.PETSALE:

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

✓ MyS  
 SELECT  
 ID AN  
 Query i  
 Cre

