

Hands-on Lab: Built-in functions - Aggregate, Scalar, String, Date and Time Functions 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:

- Compose queries consting of built in functions and check the results.

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**. Under **Database** section, click **MySQL**.

To start the MySQL click **Create**.

The screenshot shows the Skills Network Tools interface. On the left, there's a sidebar with various icons and sections: SKILLS NETWORK TOO... (with a file icon), Welcome (with a question mark icon), DATABASES (with a dropdown arrow), MySQL INACTIVE (highlighted with a red box and circled with a red number 1), PostgreSQL INACTIVE, Cassandra INACTIVE, MongoDB INACTIVE, BIG DATA, CLOUD, EMBEDDABLE AI, OTHER, Launch Application, and a tree icon. On the right, there's a large MySQL card. It has a title 'MySQL' with a 'C' icon, version numbers 8.0.22 and 5.0.4, a 'Connect to MySQL and...' link, a prominent blue 'Create' button (circled with a red number 2), and links for 'Summary' and 'Connect'. Below the card, it says 'Get started with MySQL'.

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

The screenshot shows the Skills Network Tools interface. On the left, there's a sidebar with various icons: a file icon, a magnifying glass, a gear, a triangle, a flask, and a tree. Below these are sections for 'DATASES', 'BIG DATA', 'CLOUD', 'EMBEDDABLE AI', 'OTHER', and 'Launch Application'. The 'DATASES' section has a dropdown menu open, showing 'MySQL ACTIVE' (which is highlighted with a red box), 'PostgreSQL INACTIVE', 'Cassandra INACTIVE', and 'MongoDB INACTIVE'. To the right of the sidebar is a main panel titled 'MySQL'. It displays the MySQL logo, version information (8.0.22), and connection details (5.0.4). Below this is a 'Create' button. Further down, there are 'Summary' and 'Connect' tabs, followed by a summary of the database and PHPMyAdmin details. A large blue button labeled 'phpMyAdmin' is present, with a red box highlighting its 'Copy' icon. At the bottom, there's a 'MySQL CLI' button.

File Edit Selection View Go Run Terminal Help

← → | □

SKILLS NETWORK TOO... ⚙️ ⓘ Welcome MySQL

DATABASES

MySQL ACTIVE

PostgreSQL INACTIVE

Cassandra INACTIVE

MongoDB INACTIVE

BIG DATA

CLOUD

EMBEDDABLE AI

OTHER

Launch Application

MySQL

8.0.22 | 5.0.4

Connect to MySQL and ...

Create

Summary Connect

Your database and phpMyAdmin details on how to navigate the interface.

You can manage MySQL using the phpMyAdmin interface.

phpMyAdmin

Or to interact with the database, use the MySQL CLI.

MySQL CLI

3. You will see the phpMyAdmin GUI tool.

The screenshot shows the phpMyAdmin interface with the title bar "Server: labs-mysql-echoing-cold-eggplant:3306". The left sidebar lists databases: information_schema, mysql, performance_schema, and sys. The main area has two tabs: "General settings" and "Appearance settings". In "General settings", the "Server connection collation" dropdown is set to "utf8mb4_unicode_ci". In "Appearance settings", the "Language" dropdown is set to "English", and the "Theme" is "pmahomme". On the right, there are two panels: "Database server" listing MySQL server details, and "Web server" listing Apache and MySQLd versions.

General settings

Server connection collation: **utf8mb4_unicode_ci**

Appearance settings

Language: English

Theme: pmahomme

Database server

- Server: labs-mysql-echoing-cold-eggplant via TCP/IP
- Server type: MySQL
- Server connection: **SSL is not being used**
- Server version: 8.0.37 - MySQL Community Server - GPL
- Protocol version: 10
- User: root@172.17.25.218
- Server charset: UTF-8 Unicode (utf8mb4)

Web server

- Apache
- Database client version: libmysql - mysqlnd 8.2.23

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

The encoding will be left as **utf8mb4_0900_ai_ci**. UTF-8 is the most commonly used character encoding for content or data.

Proceed to Task B.

phpMyAdmin

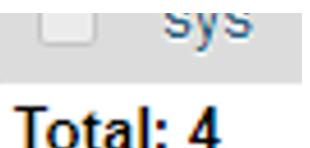
Recent Favorites

New 1

information_schema
mysql
performance_schema
sys

Serve Datab Data Create Mysql_L Check Data inform mysql perfo sys

The screenshot shows the phpMyAdmin interface. On the left, there's a tree view of databases: information_schema, mysql, performance_schema, and sys. A 'New' button is highlighted with a red box and circled with a red number '1'. On the right, a 'Create' dialog is open, also with a red box and circled with a red number '2'. The dialog contains fields for 'Name' (Mysql_L), 'Type' (InnoDB), 'Collation' (latin1_swedish_ci), and a 'Check' checkbox. Below the dialog, there are checkboxes for selecting other databases: information_schema, mysql, performance_schema, and sys.



Compose and run the following queries in the textarea of the SQL page. Click **Go** to execute the queries and observe the results.

Note: The solutions are provided at the end of this lab, but please try to compose the queries on your own before checking the solutions.

Exercise 1: Create the Pet Rescue table

Rather than create the table manually by typing the DDL commands in the SQL editor, you will execute a script containing the create table command.

1. Download the script file [PETRESCUE-CREATE.sql](#)

Note: To download, just right-click on the link above and click on **Save As..** or **Save Link As...** depending on your browser. Save the file as a .sql file and not HTML.

2. Next load the sql to your database using the Import option.

The screenshot shows the MySQL Workbench Import dialog. The left sidebar lists databases: New, HR, DEPARTMENTS, EMPLOYEES, JOBS, JOB_HISTORY, LOCATIONS, information_schema, mysql, Mysql_learners, PETSALe, performance_schema, and sys. The main area has tabs for Structure, SQL, Search, Query, Export, Import, and Operations. The Import tab is selected. The title is "Importing into the database 'Mysql_learners'".

File to import:
File may be compressed (gzip, bzip2, zip) or uncompressed.
A compressed file's name must end in **.[format].[compression]**. Example: **.sql.zip**
Browse your computer: PETRESCUE-CREATE.sql (Max: 2,048KiB)
You may also drag and drop a file on any page.
Character set of the file:

Partial import:
 Allow the interruption of an import in case the script detects it is close to the PHP timeout limit. *(This m*
Skip this number of queries (for SQL) starting from the first one:

Other options:
 Enable foreign key checks

Format:

Below the dialog, two tabs are visible: PETRESCUE-CREATE.sql and HR_Database_Crea....sql.

3. Once the table is loaded open the sql editor to start executing the functions.

The screenshot shows the phpMyAdmin interface. On the left, the database structure tree is visible, showing databases like HR, information_schema, mysql, Mysql_learners, performance_schema, and sys. Under Mysql_learners, there are tables: New, PETRESCUE, and PETSALe. The main panel shows a SQL query editor with the following details:

- Server: mysql:3306 » Database: Mysql_learners
- Structure, SQL, Search, Query, Export, Import, Operations tabs are present.
- Run SQL query/queries on database Mysql_learners: input field (empty).
- Query number: 1
- Buttons: Clear, Format, Get auto-saved query.
- Checkboxes: Bind parameters, Show this query here again, Retain query box, Rollback when finished.
- Delimiter input field set to ;

Exercise 2: Aggregate Functions

Query A1: Enter a function that calculates the total cost of all animal rescues in the PETRESCUE table.

Query A2: Enter a function that displays the total cost of all animal rescues in the PETRESCUE table in a column called SUM_OF_COST.

Query A3: Enter a function that displays the maximum quantity of animals rescued.

Query A4: Enter a function that displays the average cost of animals rescued.

Query A5: Enter a function that displays the average cost of rescuing a dog.

Exercise 3: Scalar and String Functions

Query B1: Enter a function that displays the rounded cost of each rescue.

Query B2: Enter a function that displays the length of each animal name.

Query B3: Enter a function that displays the animal name in each rescue in uppercase.

Query B4: Enter a function that displays the animal name in each rescue in uppercase without duplications.

Query B5: Enter a query that displays all the columns from the PETRESCUE table, where the animal(s) rescued are cats. Use cat in lower case in the query.

Exercise 4: Date and Time Functions

Query C1: Enter a function that displays the day of the month when cats have been rescued.

Query C2: Enter a function that displays the number of rescues on the 5th month.

Query C3: Enter a function that displays the number of rescues on the 14th day of the month.

Query C4: Animals rescued should see the vet within three days of arrivals. Enter a function that displays the third day from each rescue.

Query C5: Enter a function that displays the length of time the animals have been rescued; the difference between today's date and the rescue date.

Lab Solutions

Exercise 2: Aggregate Functions

Query A1: Enter a function that calculates the total cost of all animal rescues in the PETRESCUE table.

```
select SUM(COST) from PETRESCUE;
```

Query A2: Enter a function that displays the total cost of all animal rescues in the PETRESCUE table in a column called SUM_OF_COST.

```
select SUM(COST) AS SUM_OF_COST from PETRESCUE;
```

Query A3: Enter a function that displays the maximum quantity of animals rescued.

```
select MAX(QUANTITY) from PETRESCUE;
```

Query A4: Enter a function that displays the average cost of animals rescued.

```
select AVG(COST) from PETRESCUE;
```

Query A5: Enter a function that displays the average cost of rescuing a dog.

Hint - Bear in mind the cost of rescuing one dog on day, is different from another day. So you will have to use an average of averages.

```
select AVG(COST/QUANTITY) from PETRESCUE where ANIMAL = 'Dog' ;
```

Exercise 3: Scalar and String Functions

Query B1: Enter a function that displays the rounded cost of each rescue.

```
select ROUND(COST) from PETRESCUE;
```

Query B2: Enter a function that displays the length of each animal name.

```
select LENGTH(ANIMAL) from PETRESCUE;
```

Query B3: Enter a function that displays the animal name in each rescue in uppercase.

```
select UCASE(ANIMAL) from PETRESCUE;
```

Query B4: Enter a function that displays the animal name in each rescue in uppercase without duplications.

```
select DISTINCT(UCASE(ANIMAL)) from PETRESCUE;
```

Query B5: Enter a query that displays all the columns from the PETRESCUE table, where the animal(s) rescued are cats. Use cat in lower case in the query.

```
select * from PETRESCUE where LCASE(ANIMAL) = 'cat' ;
```

Exercise 4: Date and Time Functions

Query C1: Enter a function that displays the day of the month when cats have been rescued.

```
select DAY(RESCUEDATE) from PETRESCUE where ANIMAL = 'Cat' ;
```

Query C2: Enter a function that displays the number of rescues on the 5th month.

```
select SUM(QUANTITY) from PETRESCUE where MONTH(RESCUEDATE)='05' ;
```

Query C3: Enter a function that displays the number of rescues on the 14th day of the month.

```
select SUM(QUANTITY) from PETRESCUE where DAY(RESCUEDATE)='14' ;
```

Query C4: Animals rescued should see the vet within three days of arrivals. Enter a function that displays the third day from each rescue.

```
select DATE_ADD(RESCUEDATE, INTERVAL 3 DAY) from PETRESCUE;
```

Query C5: Enter a function that displays the length of time the animals have been rescued; the difference between today's date and the rescue date.

```
select DATEDIFF(CURRENT_TIMESTAMP, RESCUEDATE) from PETRESCUE;
```

Summary

You can now compose and run queries, check results and view the logs. You will use these skills in later labs.

Author(s)

[Lakshmi Holla](#)

[Malika Singla](#)



Skills Network