

SQL Lab



Estimated Time Needed: 60 min

In this challenge, you will create a database, import data from three sources to populate tables, and perform database operations in the MySQL database service using the phpMyAdmin graphical user interface (GUI) tool.

Objectives

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

- Create and populate a database and tables.
- Execute Structured Query Language (SQL) commands to perform basic database operations.
- Retrieve data from tables using SELECT statements.
- Filter the data output using WHERE statements.
- Aggregate data get ordered results using functions like SUM, MIN, MAX, ORDER BY.
- Use window functions to get specified output.
- Retrieve data from two or more tables using SQL JOINS.

Software used in this lab



You will use MySQL to complete this lab. MySQL is a free, open-source relational database system that offers a command line interface (MySQL) and a third-party web interface (phpMyAdmin) to efficiently store, manipulate, and retrieve data.

MySQL is a service available on Skills Network Labs (SN Labs) Cloud IDE, a virtual lab environment used in this course. SN Labs Cloud IDE is great way to do projects without downloading, installing, configuring, and integrating software on your own computer.

Two Components of the SN Labs Cloud IDE:

- The instructions that you will follow to complete this lab are displayed on the left side of the screen.
- The area on the right side of the screen is where you will use the menus, terminals, and tools to complete the lab exercises.

Dataset used in This Lab

The datasets used in this lab are three SQL files called **Salary Data**, **Sales Data** and **Employee Data**. To complete the exercises in this lab, you will be instructed to save and upload the datasets to your local machine and use a locally installed database or the tool provided in the course.

Import the given SQL input files into a database.

Prework - Create and populate database

TASK A: Create a Database

1. Start the MySQL service session using the Open MySQL Page in IDE button.

[Open MySQL Page in IDE](#)

To start the MySQL, click Start.



MySQL



MySQL

INACTIVE

8.0.22 | 5.0.4 | 2.0.2

Connect to MySQL and phpMyAdmin directly

[Start](#)[Summary](#)[Connection Information](#)

Get started with MySQL in a faster, easier way.



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

The screenshot shows the Skills Network Labs environment. At the top, there's a navigation bar with links: File, Edit, Selection, View, Go, Run, Terminal, and Help. Below the navigation bar, there are several icons: a clipboard, a magnifying glass, a wrench, a play arrow, a tree, and a flask. A tab labeled "MySQL X" is active, and another tab labeled "phpMyAdmin" is visible. The main area displays the MySQL service status: v8.0.22 | v5.0.4 | v14.14. It also says "ACTIVE". Below this, a message reads: "Connect to MySQL and phpMyAdmin directly in your Skills Network Labs environment." There's a blue "Stop" button. At the bottom of the main section, there are three tabs: "Summary" (underlined), "Connection Information", and "Details". The "Summary" tab contains information about the database and phpMyAdmin server being ready for use, along with login credentials: Username: malikas and a blurred Password. It also lists management options: "phpMyAdmin" (which is highlighted with a red box) and "MySQL CLI".

3. You will see the **phpMyAdmin GUI** tool.

phpMyAdmin



Recent Favorites

- [New](#)
- [information_schema](#)
- [mysql](#)
- [performance_schema](#)
- [sakila](#)
- [sys](#)

Server: mysql:3306



Databases



SQL



Status

General settings

Server connection collation: utf8mb4

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 or any other name you desire, 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.

The screenshot shows the phpMyAdmin interface with the 'Databases' tab selected. At the top, there are tabs for Databases, SQL, Status, User accounts, Export, and Import. Below the tabs, a 'Create database' button is visible. A form allows entering the database name 'Mysql_Learners' and the character set/collation 'utf8_general_ci'. A 'Create' button is present. The main area displays a table of existing databases:

Database	Collation	Master replication	Action
information_schema	utf8_general_ci	Replicated	Check privileges
mysql	utf8mb4_0900_ai_ci	Replicated	Check privileges
performance_schema	utf8mb4_0900_ai_ci	Replicated	Check privileges
sys	utf8mb4_0900_ai_ci	Replicated	Check privileges

Total: 4

TASK B: Create and load tables using sql files.

1. Download the 3 sql files below to your local computer:

- o [Employee Data.sql](#)
- o [Salary Data.sql](#)
- o [Sales Data.sql](#)

2. To load each sql file do the following steps.

- o Select your database, in the case of the example shown it is **Mysql_Learners** and click on **Import** tab.
- o Click on **Choose File**. Browse for the file and upload it .
- o Later scroll down and click the Go button.

The screenshot shows the phpMyAdmin interface with the 'Import' tab selected. The left sidebar shows the database tree, with 'Mysql_Learners' highlighted and a red box around it. The main area is titled '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: Sales Data.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 might be a good way to import large file)

Skip this number of queries (for SQL) starting from the first one:

Other options:
 Enable foreign key checks

Console

- Once the scripts are loaded, you will get a message that, it is imported successfully.

✓ Import has been successfully finished, 3 queries executed. (Sales Data.sql)

✓ MySQL returned an empty result set (i.e. zero rows). (Query took 0.0366 seconds.)

```
CREATE TABLE `Sales` ( `SalesID` varchar(10) DEFAULT NULL, `EmpID` varchar(10) DEFAULT NULL, `Units_Sold` double DEFAULT NULL, `Sale_
```

✓ 24 rows inserted. (Query took 0.0068 seconds.)

```
INSERT INTO `Sales` (`SalesID`, `EmpID`, `Segment`, `Product`, `Un
```

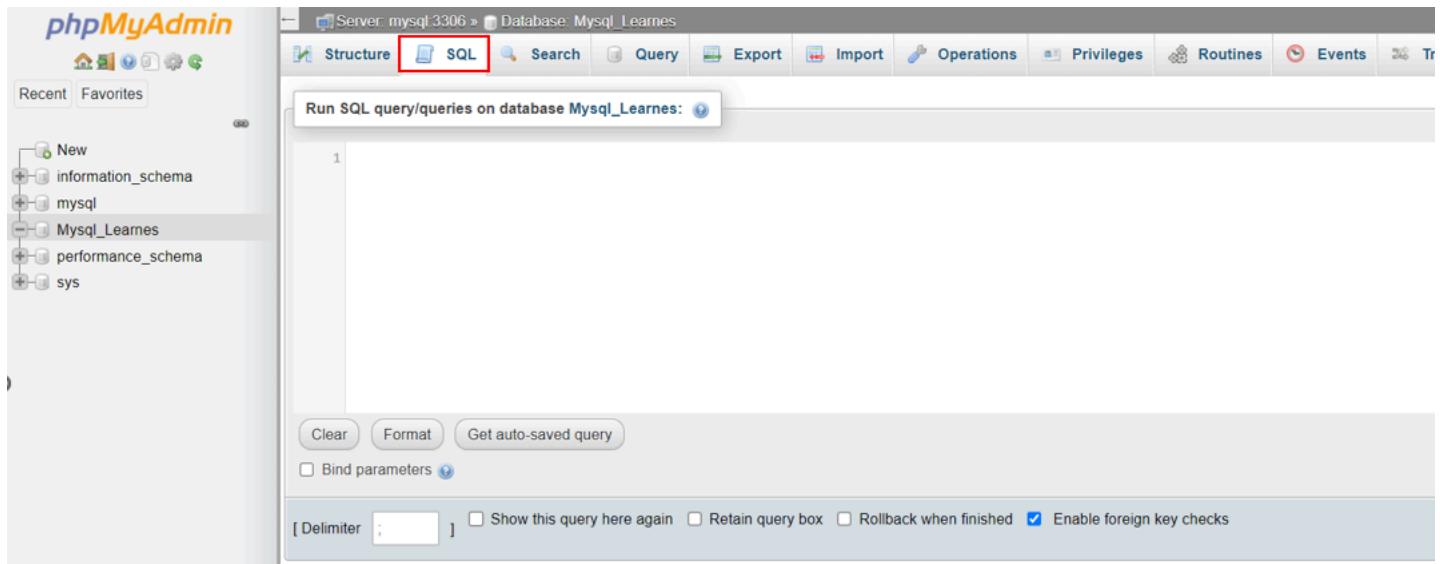
'E04732', 'Government', 'Product2', 252, 20, 5040, 2920, 2120, '04
571, '07/24/2021'), ('S2530', 'E03496', 'Midmarket', 'Product2', 2
'Product1', 2133, 7, 14931, 10730, 4201, '09/29/2022'), ('S2512',
('S2513', 'E04732', 'Channel Partners', 'Product1', 1001, 30, 3003
'Product1', 2513, 12, 30156, 7554, 22602, '06/21/2022'), ('S2514',

✓ MySQL returned an empty result set (i.e. zero rows). (Query took 0.0002 seconds.)

COMMIT

You can import the other sql files in the same way.

3. To run the SQL queries you need to copy the given codes and paste it to the text area of the SQL page and click on Go.



Data Engineering

In this section you will perform data cleansing (removing duplicates) and data transformation (change column name) operations on the data.

1. Identify the duplicate entry for employees in the employee table using GROUP BY and HAVING statements.

► Solution syntax
► Output

2. Select the duplicate entry for employees and delete the row with the higher EMPID.

► Solution syntax
► Output

3. Data Transformation – Change the column Salary in the Salary table to “Annual_Income”

► Solution syntax
► Output

SQL JOINS, Aggregations

Use SQL JOINS, Aggregations where needed, to derive metrics from the database tables.

1. Using the tables given, find out the Total number of men and women employees in the company who are aged below 50 yrs.

► Solution syntax
► Output

2. Using the tables, find the employees whose salary is greater than \$150000.

Note – Rename column Annual_Income back to Salary in the table Salary

► Solution syntax
► Output

SQL GROUP BY, HAVING

Use SQL GROUP BY and HAVING statements to get some count metrics from database tables.

1. Display products grouped by segments with total Sales greater than \$100,000.

► Solution syntax
► Output

Formatted Output

Show output result ordered in a certain way (Use window functions row_number() or rank() and order by statements).

1. Show an output table of Sales generated by employees ordered highest to lowest.

► Solution syntax
► Output

2. Show an output table of Sales(ordered highest to lowest) generated by employees in different segments and rank them for each employee.

► Solution syntax
► Output

Explain the following:

- ▶ When you would use COALESCE function
- ▶ What is the difference between Union and Union
- ▶ What is the difference between clustered and non-clustered indexes

Author

Roopa Raghavan

© IBM Corporation. All rights reserved.