

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 consisting of built-in functions and check the results.

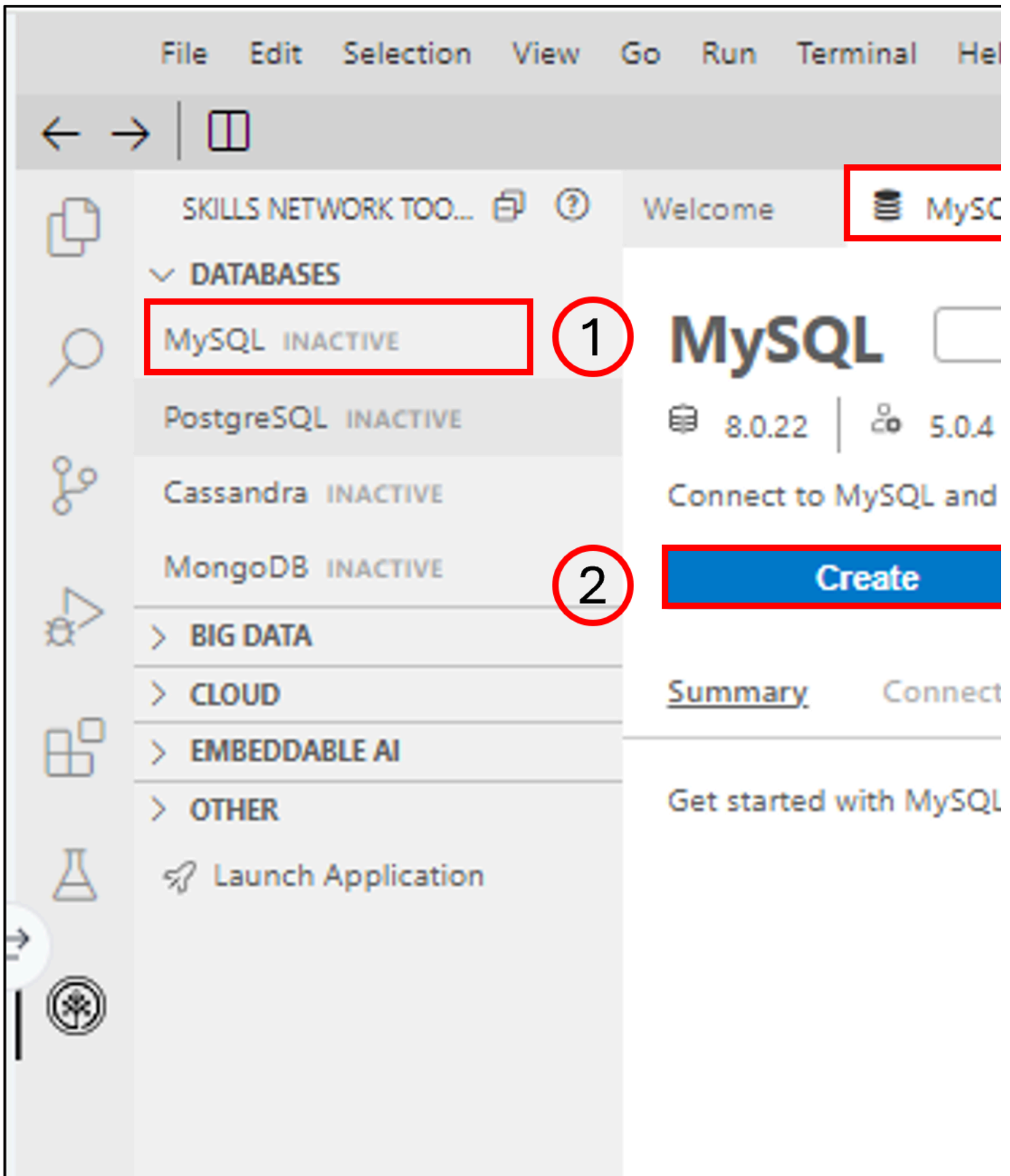
Exercise

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**.



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

The screenshot shows the Skills Network Tools interface. The top menu bar includes File, Edit, Selection, View, Go, Run, Terminal, and Help. Below the menu is a navigation bar with a back arrow, a forward arrow, and a tab labeled "SKILLS NETWORK TOO..." with a question mark icon. The main content area is divided into two panels. The left panel, titled "Databases", lists several database options: MySQL (ACTIVE, highlighted with a red box), PostgreSQL (INACTIVE), Cassandra (INACTIVE), and MongoDB (INACTIVE). Below these are sections for "BIG DATA", "CLOUD", "EMBEDDABLE AI", and "OTHER", each with a right-pointing arrow. At the bottom of the left panel is a "Launch Application" button with a rocket icon. The right panel, titled "Welcome", shows the MySQL logo and version information: 8.0.22 | 5.0.4. Below this is a "Connect to MySQL and" button. A large blue "Create" button is prominently displayed. Underneath, there are tabs for "Summary" and "Connecti". The "Summary" tab is active, showing text about the database and phpMyAdmin. Below this text is a blue button labeled "phpMyAdmin" with a red box around its icon, and another blue button labeled "MySQL CLI".

File Edit Selection View Go Run Terminal Help

SKILLS NETWORK TOO... ?

Welcome MySQL

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 Connecti

Your database and phpM details on how to naviga

You can manage MySQL

phpMyAdmin

Or to interact with the d

MySQL CLI

3. You will see the phpMyAdmin GUI tool.

The screenshot shows the phpMyAdmin web interface. The top navigation bar includes links for Databases, SQL, Status, User accounts, Export, Import, and More. The left sidebar shows a tree view of databases: New, information_schema, mysql, performance_schema, and sys. The main content area is divided into three panels: General settings, Appearance settings, and Database server. The General settings panel shows the Server connection collation set to utf8mb4_unicode_ci. The Appearance settings panel shows the Language set to English and the Theme set to pmahomme. The Database server panel lists server details: 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, and Server charset: UTF-8 Unicode (utf8mb4). The Web server panel lists Apache and 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.

The screenshot displays the phpMyAdmin web interface. The main header features the 'phpMyAdmin' logo and a row of icons for home, back, help, document, settings, and refresh. Below these are 'Recent' and 'Favorites' buttons. A sidebar on the left lists databases: 'New' (highlighted with a red box and a circled '1'), 'information_schema', 'mysql', 'performance_schema', and 'sys'. The right sidebar shows a 'Database' section with a 'Create' button (highlighted with a red box and a circled '2') and a list of databases with checkboxes: 'information_schema', 'mysql', 'performance_schema', and 'sys'.

phpMyAdmin

Recent Favorites

New 1

information_schema

mysql

performance_schema

sys

Database

Create 2

MySQL_L

Check

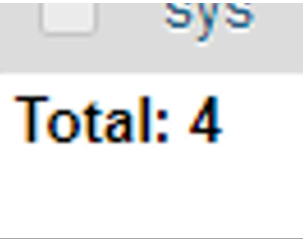
Data

inform

mysql

perfo

sys


Total: 4

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

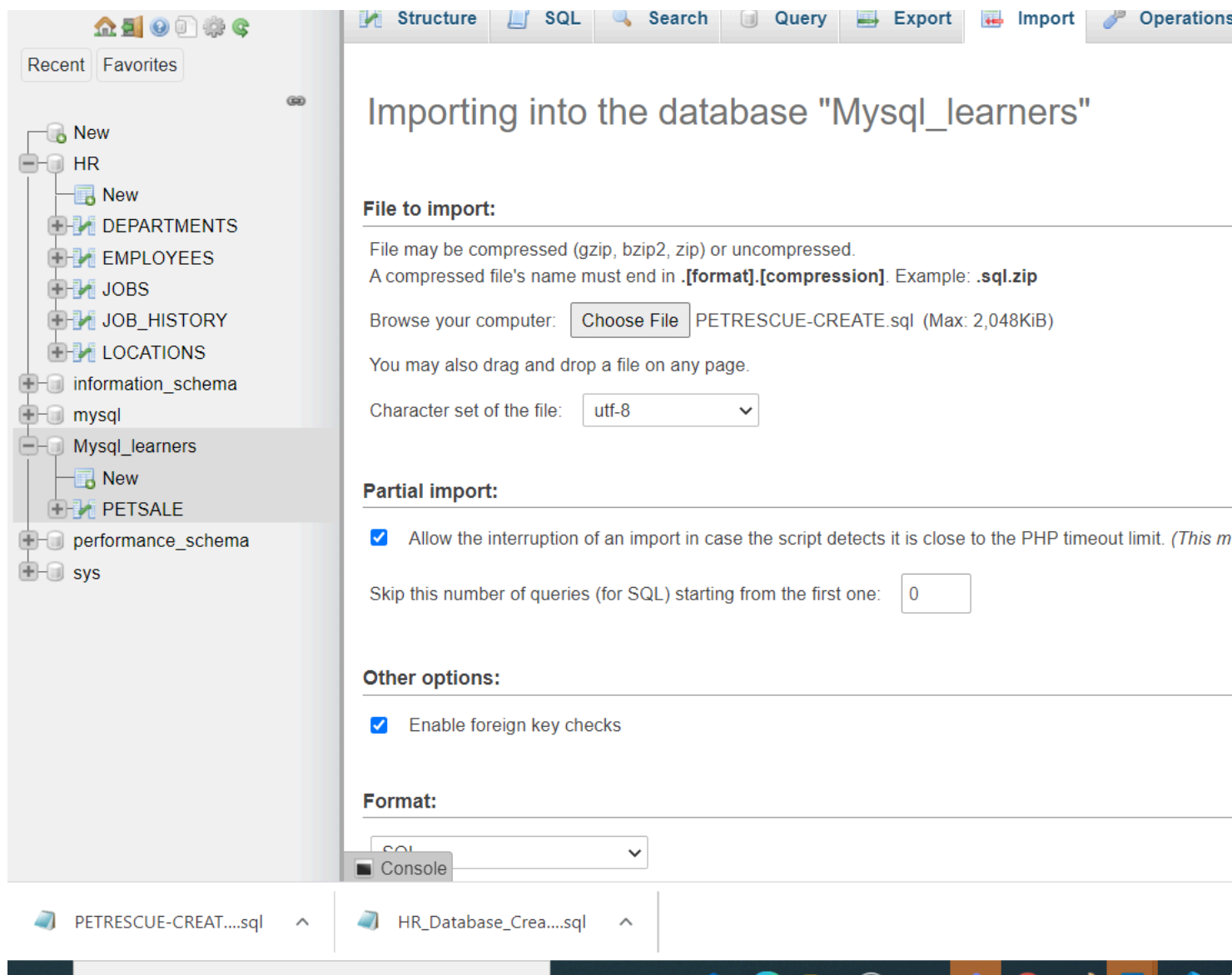
Exercise 1: Create the Pet Rescue table

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.



Recent Favorites

New
HR
New
DEPARTMENTS
EMPLOYEES
JOBS
JOB_HISTORY
LOCATIONS
information_schema
mysql
Mysql_learners
New
PETALE
performance_schema
sys

Structure SQL Search Query Export Import Operations

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: **Choose File** PETRESCUE-CREATE.sql (Max: 2,048KiB)

You may also drag and drop a file on any page.

Character set of the file: **utf-8**

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: **0**

Other options:

☒ Enable foreign key checks

Format:

SQL

Console

PETRESCUE-CREAT....sql HR_Database_Crea....sql

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

The screenshot shows the phpMyAdmin web interface. On the left is a sidebar with a tree view of databases and tables. The 'Mysql_learners' database is selected, showing tables: PETRESCUE and PETSAL. The main panel on the right is titled 'Run SQL query/queries on database Mysql_learners:'. It contains a large text area for writing SQL queries, with a '1' indicating the first query. Below the text area are buttons for 'Clear', 'Format', and 'Get auto-saved query'. There is a checkbox for 'Bind parameters' and a row of checkboxes for 'Show this query here again', 'Retain query box', and 'Rollback when finished'. A 'Delimiter' dropdown is set to ';'. At the bottom, there is a 'Console' tab.

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 my the cost of rescuing one dog on day, is different from another day. So you will have to use and 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