

# Database Table Operations

## Scenario

The database operations team for an organization has configured a relational database instance. The team has asked you to practice creating and dropping (deleting) databases and tables.

## Lab overview and objectives

This lab demonstrates how to use some common database and table operations.

After completing this lab, you should be able to:

- Use the **CREATE** statement to create databases and tables
- Use the **SHOW** statement to view available databases and tables
- Use the **ALTER** statement to alter the structure of a table
- Use the **DROP** statement to delete databases and tables

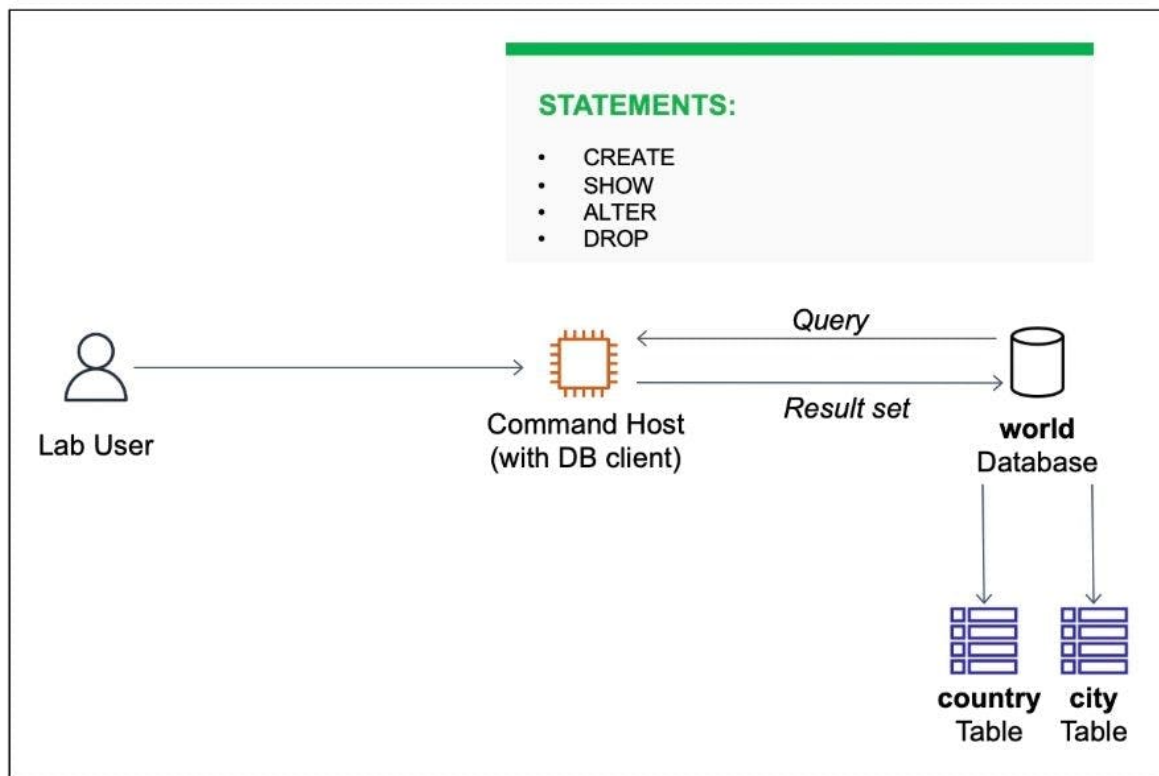
When you start the lab, the following resources are already created for you:



**Command Host**  
(with DB client)

*A database client is installed on an instance.*

At the end of this lab, you will have completed some common database and table operations:



A lab user creates a database and tables. Other displayed statements are *SHOW*, *ALTER*, and *DROP*.

Sample data in this course is taken from Statistics Finland, general regional statistics, February 4, 2022.

## Duration

This lab requires approximately **45 minutes** to complete.

## AWS service restrictions

In this lab environment, access to AWS services and service actions might be restricted to the ones that you need to complete the lab instructions. You might encounter errors if you attempt to access other services or perform actions beyond the ones that this lab describes.

## Accessing the AWS Management Console

1. At the upper-right corner of these instructions, choose ► **Start Lab**

**Troubleshooting tip:** If you get an **Access Denied** error, close the error box, and choose ► **Start Lab** again.

2. The following information indicates the lab status:

- A red circle next to **AWS** ● at the upper-left corner of this page indicates that the lab has not been started.
- A yellow circle next to **AWS** ● at the upper-left corner of this page indicates that the lab is starting.
- A green circle next to **AWS** ● at the upper-left corner of this page indicates that the lab is ready.

Wait for the lab to be ready before proceeding.

3. At the top of these instructions, choose the green circle next to **AWS** ●

This option opens the AWS Management Console in a new browser tab. The system automatically signs you in.


**Tip:** If a new browser tab does not open, a banner or icon at the top of your browser will indicate that your browser is preventing the site from opening pop-up windows. Choose the banner or icon, and choose **Allow pop-ups**.

4. Arrange the AWS Management Console tab so that it displays along side these instructions. Ideally, you should be able to see both browser tabs at the same time so that you can follow the lab steps.

**⚠ Do not change the lab Region unless specifically instructed to do so.**

## Task 1: Connect to the Command Host

In this task, you connect to an EC2 instance configured with a database client. The client is used to run structured query language (SQL) queries against a relational database. This instance is referred to as the Command Host.

5. In the AWS Management Console, choose the  **Services** menu. Choose **Compute**, and then choose **EC2**.
6. In the left navigation menu, choose **Instances**.
7. Next to the instance labelled **Command Host**, select the check box ☒ and then choose **Connect**.

**Note:** If you do not see the **Command Host**, the lab is probably still being provisioned, or you may be using another Region.

8. For **Connect to instance**, choose the **Session Manager** tab.
9. Choose **Connect** to open a terminal window.

**Note:** If the **Connect** button is not available, wait for a few minutes and try again.

10. To configure the terminal to access all required tools and resources, run the following command:

```
sudo su
cd /home/ec2-user/
```

► Recall Linux commands


### Tips:

- Copy and paste the command into the Session Manager terminal window.
  - If you are using a Windows system, press Shift+Ctrl+v to paste the command.
11. To connect to the relational database instance, run the following command in the terminal. A password was configured when the database was installed.

```
mysql -u root --password='re:St@rt!9'
```

The MySQL command-line client is an SQL shell that you can use to interact with database engines.

Switch	Description
-u or --user	The MySQL user name used to connect to a database instance
-p or --password	The MySQL password used to connect to a database instance

** Tip:** At any stage of the lab, if the Session Manager window is not responsive or if you need to reconnect to the database instance, then follow these steps:

- Close the Session Manager window, and try to reconnect using the previous steps.
- Run the following commands in the terminal.

```
sudo su
cd /home/ec2-user/
mysql -u root --password='re:St@rt!9'
```

## Task 2: Create a database and a table

In this task, you create a database named **world** and a table named **country**. You then alter the **country** table.

12. To show the existing databases, run the following query.

```
SHOW DATABASES;
```

To determine the available database and to ensure that you are working with the correct database instance, use the **SHOW DATABASES;** command.

13. To create a new database named **world**, run the following command.

```
CREATE DATABASE world;
```

14. To verify that the **world** database has been created, run the following query.

```
SHOW DATABASES;
```

15. To store data in a database, the database needs to contain one or more tables. In an SQL database, a table needs a well-defined structure, known as a table schema. To create a table named **country**, run the following command.

```
CREATE TABLE world.country (  
  `Code` CHAR(3) NOT NULL DEFAULT '',  
  `Name` CHAR(52) NOT NULL DEFAULT '',  
  `Conitinent` enum('Asia','Europe','North  
America','Africa','Oceania','Antarctica','South America') NOT NULL DEFAULT 'Asia',  
  `Region` CHAR(26) NOT NULL DEFAULT '',  
  `SurfaceArea` FLOAT(10,2) NOT NULL DEFAULT '0.00',  
  `IndepYear` SMALLINT(6) DEFAULT NULL,  
  `Population` INT(11) NOT NULL DEFAULT '0',  
  `LifeExpectancy` FLOAT(3,1) DEFAULT NULL,  
  `GNP` FLOAT(10,2) DEFAULT NULL,  
  `GNPold` FLOAT(10,2) DEFAULT NULL,  
  `LocalName` CHAR(45) NOT NULL DEFAULT '',  
  `GovernmentForm` CHAR(45) NOT NULL DEFAULT '',  
  `HeadOfState` CHAR(60) DEFAULT NULL,  
  `Capital` INT(11) DEFAULT NULL,  
  `Code2` CHAR(2) NOT NULL DEFAULT '',  
  PRIMARY KEY (`Code` )  
);
```

16. To verify that the **country** table was created, use the **SHOW TABLES;** command to list the tables in the database. You use the **USE** command to specify which database to run a query against. Run the following commands in your terminal.

```
USE world;  
SHOW TABLES;
```

17. Use the **SHOW COLUMNS** query to list all the columns on a table. Run the following query to list all columns and their properties in the **country** table.

```
SHOW COLUMNS FROM world.country;
```

**Note:** Notice that the **Continent** column is spelled incorrectly as **Conitinent**.

18. The **ALTER TABLE** command is used to alter the table's schema. To fix the incorrectly spelled **Continent** column, run the following command.

```
ALTER TABLE world.country RENAME COLUMN Conitinent TO Continent;
```

19. To verify that the **Continent** column name in the **country** table has been corrected, run the following query.

```
SHOW COLUMNS FROM world.country;
```

## Challenge 1

► Create a table named **city** and add two columns named **Name** and **Region**. Both columns should use the **CHAR** data type.

**Tip:** Expand the question to reveal the solution.



## Task 3: Delete a database and tables

In this task, you delete the **world** database and **country** table.

20. The **DROP TABLE** command is used to delete (drop) a table in a database. Once a table has been dropped, it cannot be recovered unless a backup is available. To drop the **city** table, run the following command.

```
DROP TABLE world.city;
```

## Challenge 2

- Write a query to drop the **country** table.

**Tip:** Expand the question to reveal the solution.

21. To verify that both tables have been dropped, run the following query.

```
SHOW TABLES;
```

22. To drop the **world** database, run the following command.

```
DROP DATABASE world;
```

23. To verify that the **world** database has been deleted, run the following query.

```
SHOW DATABASES;
```

## Conclusion

👏 Congratulations! You now have successfully:

- Used the **CREATE** statement to create databases and tables
- Used the **SHOW** statement to view available databases and tables
- Used the **ALTER** statement to alter the structure of a table
- Used the **DROP** statement to delete databases and tables

## Lab complete

24. Choose **■ End Lab** at the top of this page, and then select **Yes** to confirm that you want to end the lab.
25. An **Ended AWS Lab Successfully** message is briefly displayed indicating that the lab has ended.