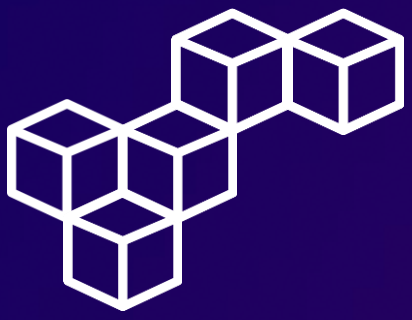
AWS re:Start LAB

Database Table Operations

WEEK 6

aws re/start



Overview

Database table operations encompass the core functionalities of managing structured data within a database system. These operations revolve around creating, viewing, altering, and deleting tables, which are essential components for storing and organizing data.

At the core of these operations is the creation of databases and tables, providing a structured framework for data storage. Viewing available databases and tables offers insights into existing data structures, aiding in data analysis and management decisions. Altering table structures allows for adjustments to data schemas, ensuring adaptability to changing data requirements. Deleting databases and tables streamlines data management by removing redundant or obsolete structures, optimizing database resources for improved performance. These operations collectively form the foundation for effective data organization and manipulation within database environments.

Topics covered

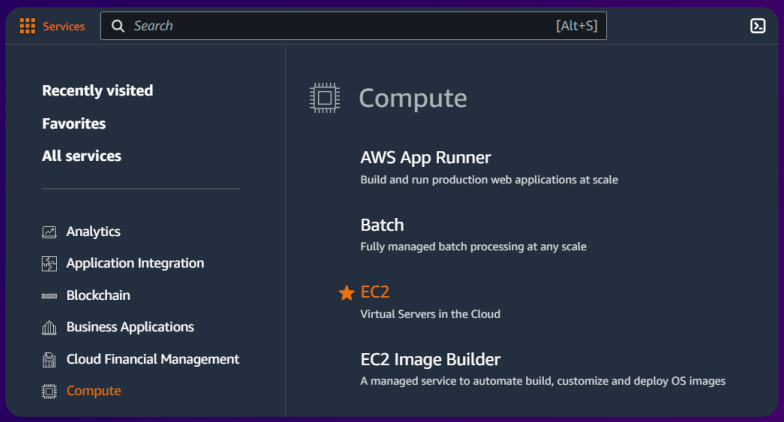
- Use the CREATE statement to create databases and tables
- Use the SHOW statement to view available databases and tables
- Use the ALTER statement to alter the structure of a table
- Use the DROP statement to delete databases and tables

Task 1

Connect to the Command Host

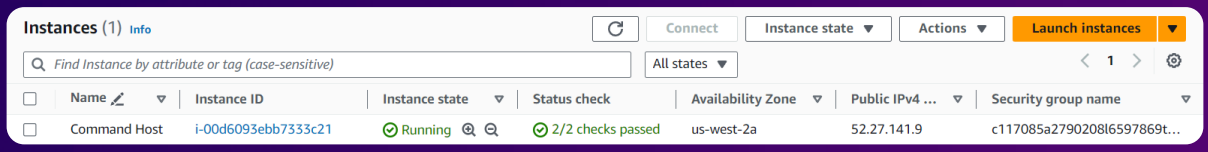
Step 1: Access the EC2 Management Console

Open the AWS Management Console, and select EC2.



Step 2: Review running instances

Navigate to the **Instances** section. The running **Command Host** instance is listed.



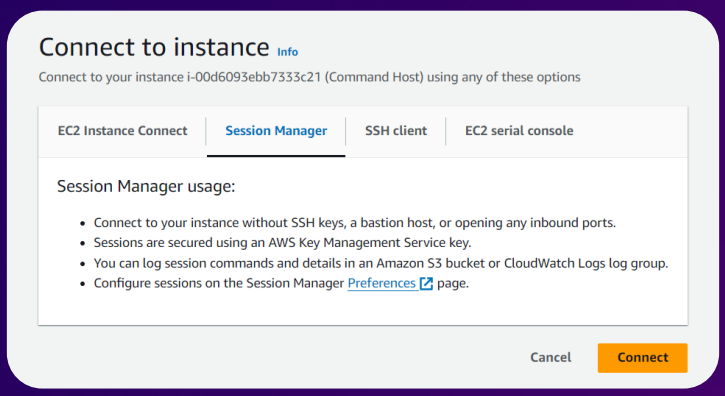
aws re/start

Task 1

Connect to the Command Host

Step 3: Connect to the instance

Connect to the **Command Host** EC2 instance, which is configured with a database client, using Session Manager.



Step 4: Connect to the relational database

To connect to the relational database instance, run the following commands in the terminal.

```
sh-4.2$ sudo su
[root@ip-10-1-11-39 bin]# cd /home/ec2-user/
[root@ip-10-1-11-39 ec2-user]# mysql -u root --password='re:St@rt!9'
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 14
Server version: 10.6.17-MariaDB MariaDB Server

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]>
```

aws re/start

Task 2

Create a database and a table

Step 1: Create a new database

Review existing databases and create a database named **world**.

```
MariaDB [(none)]> SHOW DATABASES;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| mysql              |
| performance_schema |
| sys                |
+--------------------+
4 rows in set (0.000 sec)
```

```
MariaDB [(none)]> CREATE DATABASE world;
Query OK, 1 row affected (0.000 sec)

MariaDB [(none)]> SHOW DATABASES;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| mysql              |
| performance_schema |
| sys                |
| world              |
+--------------------+
5 rows in set (0.000 sec)
```

Step 2: Create new tables

Create a table named **country** and a table named **city**.

```
MariaDB [(none)]> 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`)
    -> );
Query OK, 0 rows affected (0.010 sec)

MariaDB [(none)]> CREATE TABLE world.city (`Name` CHAR(52), `Region` CHAR(26));
Query OK, 0 rows affected (0.006 sec)
```

Task 2

Create a database and a table

Step 3: Verify table and fields

First, run the USE command to specify which database to run a query against. Then, use the SHOW TABLES query to list all the tables in the specified database.

```
MariaDB [(none)]> USE world;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
MariaDB [world]> SHOW TABLES;
+-----------------+
| Tables_in_world |
+-----------------+
| city            |
| country         |
+-----------------+
2 rows in set (0.000 sec)
```

Use the SHOW COLUMNS query to list all the columns on the **country** table.

```
MariaDB [world]> SHOW COLUMNS FROM world.country;
+---------------+-----------------------------------------------------------------------------------------+------+-----+---------+-------+
| Field         | Type                                                                                    | Null | Key | Default | Extra |
+---------------+-----------------------------------------------------------------------------------------+------+-----+---------+-------+
| Code          | char(3)                                                                                 | NO   | PRI |         |       |
| Name          | char(52)                                                                                | NO   |     |         |       |
| Conitinent    | enum('Asia','Europe','North America','Africa','Oceania','Antarctica','South  America') | NO   |     | Asia    |       |
| Region        | char(26)                                                                                | NO   |     |         |       |
| SurfaceArea   | float(10,2)                                                                             | NO   |     | 0.00    |       |
| IndepYear     | smallint(6)                                                                             | YES  |     | NULL    |       |
| Population    | int(11)                                                                                 | NO   |     | 0       |       |
| LifeExpectancy | float(3,1)                                                                             | YES  |     | NULL    |       |
| GNP           | float(10,2)                                                                             | YES  |     | NULL    |       |
| GNPOld        | float(10,2)                                                                             | YES  |     | NULL    |       |
| LocalName     | char(45)                                                                                | NO   |     |         |       |
| GovernmentForm | char(45)                                                                               | NO   |     |         |       |
| HeadOfState   | char(60)                                                                                | YES  |     | NULL    |       |
| Capital       | int(11)                                                                                 | YES  |     | NULL    |       |
| Code2         | char(2)                                                                                 | NO   |     |         |       |
+---------------+-----------------------------------------------------------------------------------------+------+-----+---------+-------+
15 rows in set (0.001 sec)
```

Task 2

Create a database and a table

Step 4: Fix field misspelling

Notice that the **Continent** column is spelled incorrectly as Conitinent. The ALTER TABLE query is used to alter the table's schema, use it to fix the incorrectly spelled **Continent** column.

```
MariaDB [world]> ALTER TABLE world.country RENAME COLUMN Conitinent TO Continent;
Query OK, 0 rows affected (0.010 sec)
Records: 0  Duplicates: 0  Warnings: 0
```

Verify that the **Continent** column name in the **country** table has been corrected.

```
MariaDB [world]> SHOW COLUMNS FROM world.country;
+---------------+-----------------------------------------------------------------------------------------+------+-----+---------+-------+
| Field         | Type                                                                                    | Null | Key | Default | Extra |
+---------------+-----------------------------------------------------------------------------------------+------+-----+---------+-------+
| Code          | char(3)                                                                                 | NO   | PRI |         |       |
| Name          | char(52)                                                                                | NO   |     |         |       |
| Continent     | enum('Asia','Europe','North America','Africa','Oceania','Antarctica','South  America') | NO   |     | Asia    |       |
| Region        | char(26)                                                                                | NO   |     |         |       |
| SurfaceArea   | float(10,2)                                                                             | NO   |     | 0.00    |       |
| IndepYear     | smallint(6)                                                                             | YES  |     | NULL    |       |
| Population    | int(11)                                                                                 | NO   |     | 0       |       |
| LifeExpectancy| float(3,1)                                                                              | YES  |     | NULL    |       |
| GNP           | float(10,2)                                                                             | YES  |     | NULL    |       |
| GNPOld        | float(10,2)                                                                             | YES  |     | NULL    |       |
| LocalName     | char(45)                                                                                | NO   |     |         |       |
| GovernmentForm| char(45)                                                                                | NO   |     |         |       |
| HeadOfState   | char(60)                                                                                | YES  |     | NULL    |       |
| Capital       | int(11)                                                                                 | YES  |     | NULL    |       |
| Code2         | char(2)                                                                                 | NO   |     |         |       |
+---------------+-----------------------------------------------------------------------------------------+------+-----+---------+-------+
15 rows in set (0.001 sec)
```

Task 3

Delete a database and tables

Step 1: Drop tables

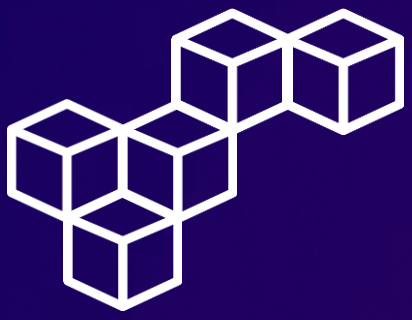
The DROP TABLE query is used to delete (drop) a table in a database. Drop the **city** table and the **country** table and verify that both tables have been dropped.

```
MariaDB [world]> DROP TABLE world.city;
Query OK, 0 rows affected (0.006 sec)

MariaDB [world]> DROP TABLE world.country;
Query OK, 0 rows affected (0.005 sec)

MariaDB [world]> SHOW TABLES;
Empty set (0.000 sec)
```

Step 2: Drop database

Drop the **world** database and verify that it has been deleted.

```
MariaDB [world]> DROP DATABASE world;
Query OK, 0 rows affected (0.003 sec)

MariaDB [(none)]> SHOW DATABASES;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| mysql              |
| performance_schema |
| sys                |
+--------------------+
4 rows in set (0.000 sec)
```



Conclusions

Database table operations

Database table operations are fundamental for managing data effectively within a database system, involving actions like creating, viewing, altering, and dropping tables.

The CREATE statement

The CREATE statement is crucial for establishing databases and tables, providing the foundational structure for organizing and storing data.

The SHOW statement

The SHOW statement is valuable for gaining visibility into available databases and tables, aiding in data analysis and management decisions.

The ALTER statement

The ALTER statement allows for flexible modifications to table structures, ensuring adaptability to changing data requirements without the need for recreating tables.

The DROP statement

The DROP statement is essential for removing obsolete or redundant databases and tables, optimizing database resources and streamlining data management processes.

aws re/start

Cristhian Becerra

cristhian-becerra-espinoza

+51 951 634 354

cristhianbecerra99@gmail.com

Lima, Peru