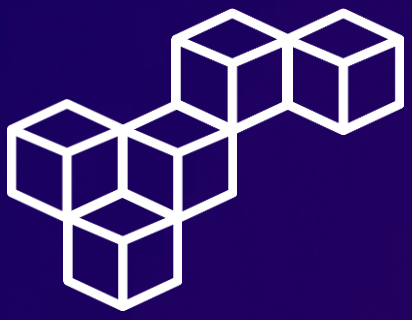
AWS re:Start LAB

Introduction to Amazon Aurora

WEEK 7

aws re/start



Overview

Aurora is a fully managed, MySQL-compatible, relational database engine that combines the performance and reliability of high-end commercial databases with the simplicity and cost-effectiveness of open-source databases. It delivers up to five times the performance of MySQL without requiring changes to most of your existing applications that use MySQL databases.

Amazon Aurora offers high-performance and scalable database solutions. Creating an Aurora instance allows efficient data management, while connecting to Amazon EC2 instances streamlines integration. Configuring EC2 to connect to Aurora ensures smooth data communication, enabling seamless querying for valuable insights and informed decision-making. Overall, Aurora's advanced features and architecture simplify database administration and enhance performance, making it a top choice for scalable and reliable data solutions.

Topics covered

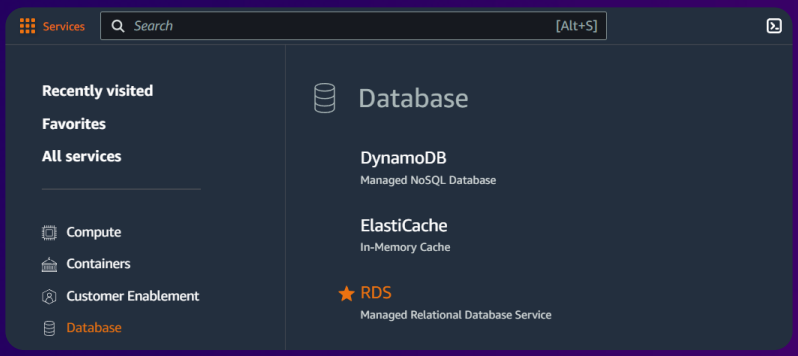
- Create an Aurora instance
- Connect to a pre-created Amazon Elastic Compute Cloud (Amazon EC2) instance
- Configure the Amazon EC2 instance to connect to Aurora
- Query the Aurora instance

Task 1

Create an Aurora instance

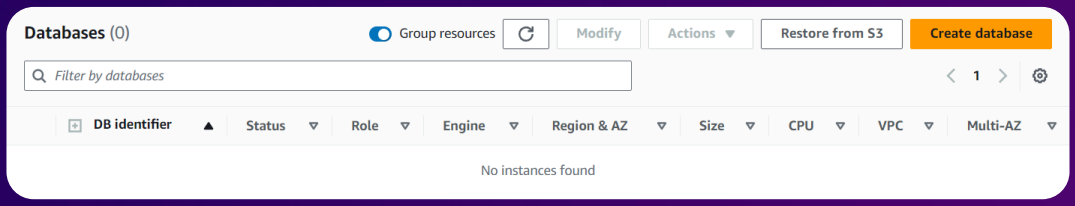
Step 1: Access the RDS database service

Open the AWS Management Console, and select RDS.



Step 2: Create database

Navigate to the Databases section, and select Create database.



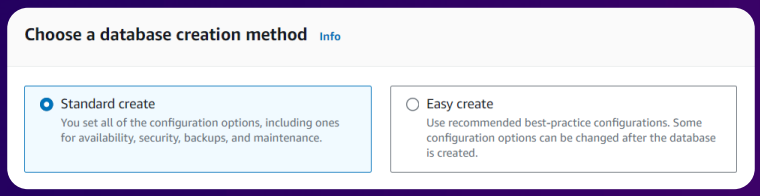
aws re/start

Task 1

Create an Aurora instance

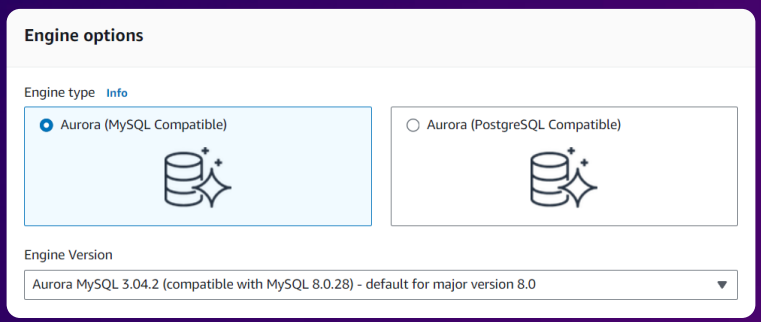
Step 3: Choose a database creation method

In the **Choose a database creation method** section, choose Standard create.



Step 4: Engine options

In the **Engine options** section, for Engine type, choose Aurora (MySQL Compatible), for Engine version, choose the default for major version 8.0.



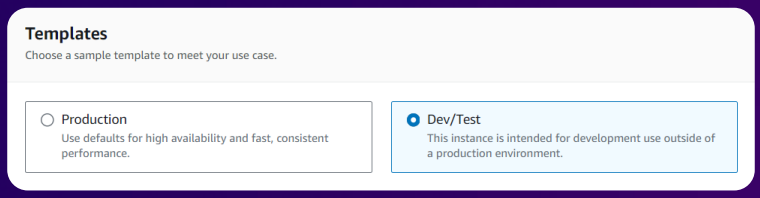
aws re/start

Task 1

Create an Aurora instance

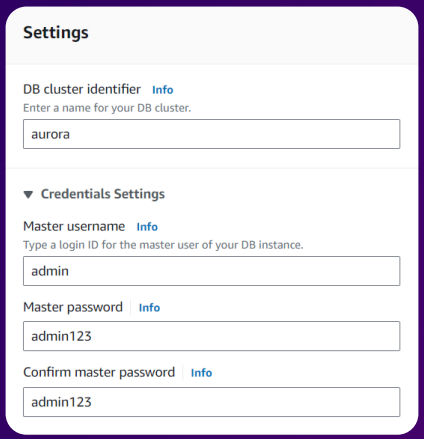
Step 5: Templates

In the **Templates** section, choose Dev/Test.



Step 6: Settings

In the **Settings** section, configure the following options.



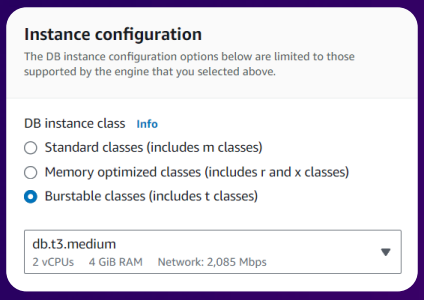
aws re/start

Task 1

Create an Aurora instance

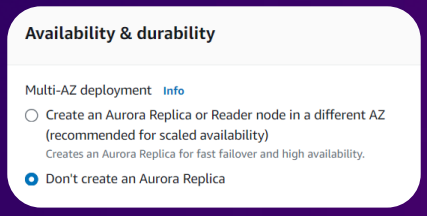
Step 7: Instance configuration

In the **Instance configuration** section, for DB instance class, configure the following settings.



Step 8: Availability & durability

In the **Availability & durability** section, for Multi-AZ deployment, choose Don't create an Aurora Replica.



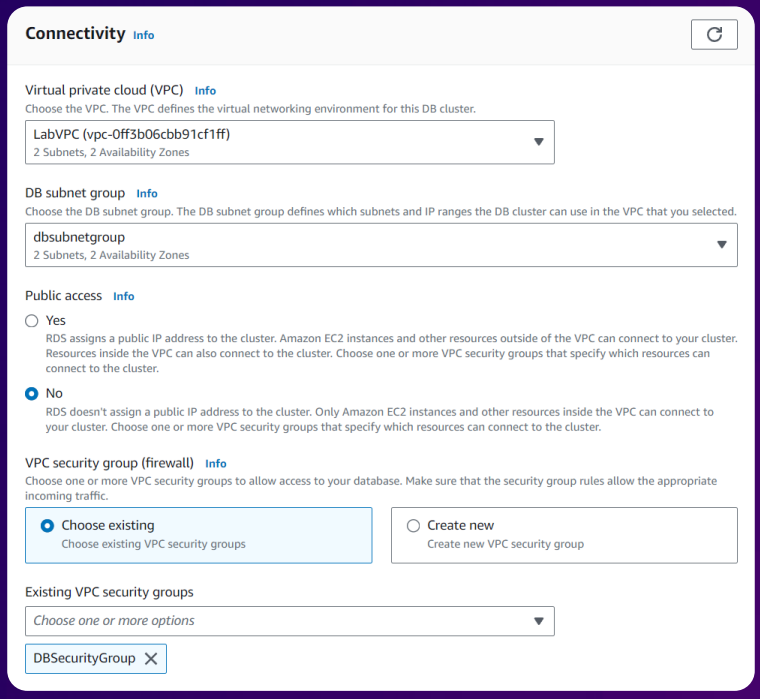
aws re/start

Task 1

Create an Aurora instance

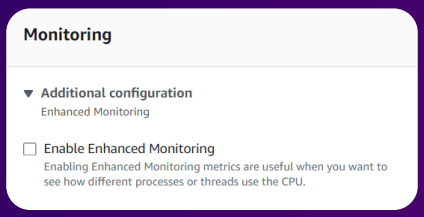
Step 9: Connectivity

In the **Connectivity** section, configure the following options.



Step 10: Monitoring

In the **Monitoring** section, for Additional configuration, clear the check box for Enable Enhanced Monitoring.



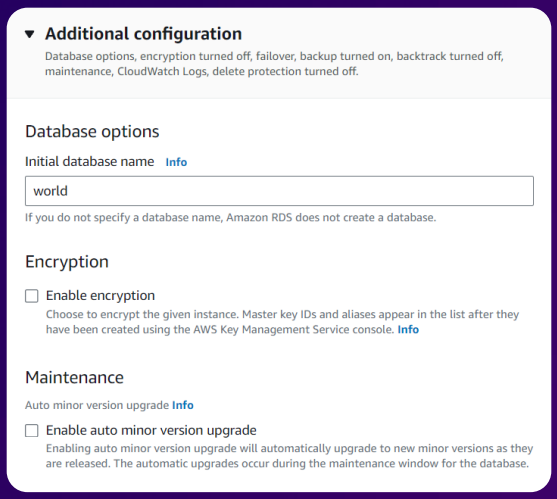
aws re/start

Task 1

Create an Aurora instance

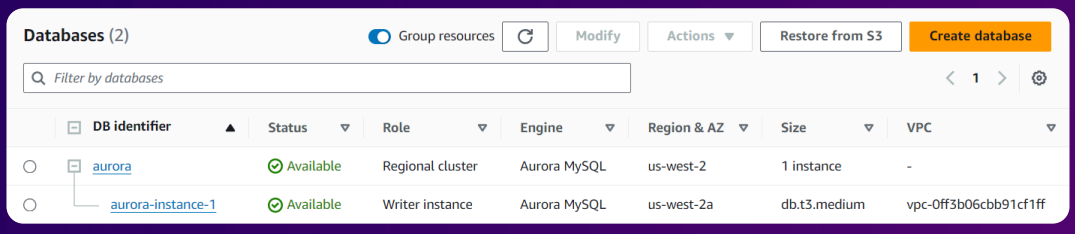
Step 11: Additional configuration

In the **Additional configuration** section, configure the following options.



Step 12: Review database instance creation

Verify the availability of the aurora-instance-1 database.



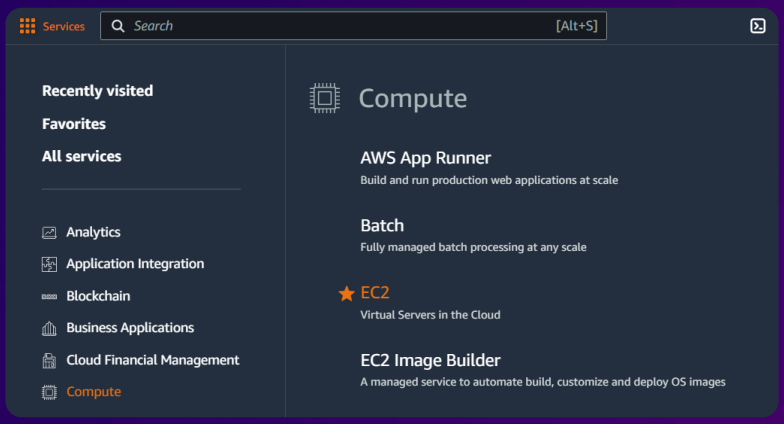
aws re/start

Task 2

Connect to an Amazon EC2 Linux instance

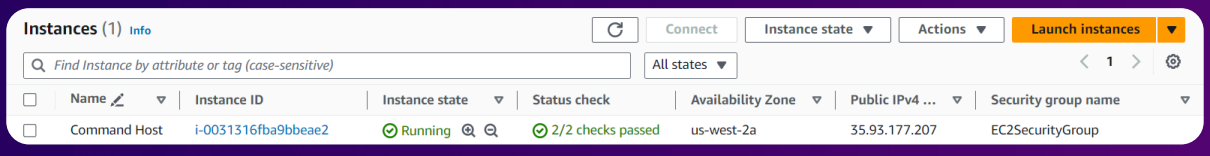
Step 1: Access the EC2 Management Console

In the AWS Management Console, select EC2.



Step 2: Review running EC2 instances

Navigate to the **Instances** section. The running **Command Host** EC2 instance is listed.



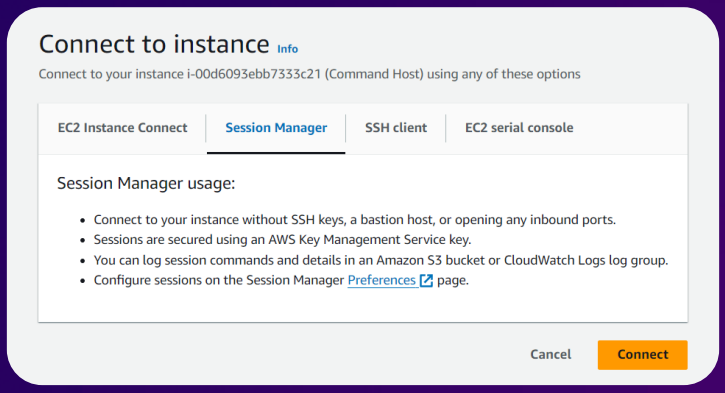
aws re/start

Task 2

Connect to an Amazon EC2 Linux instance

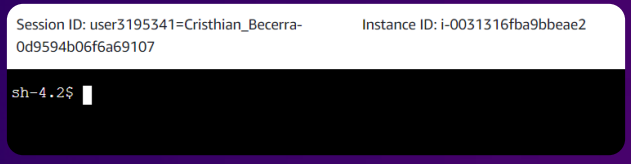
Step 3: Connect to the instance

Connect to the **Command Host** EC2 instance using Session Manager.



Step 4: Review connection

You have successfully connected to the Amazon EC2 instance named **Command Host**.



aws re/start

Task 3

Configure the Amazon EC2 Linux instance to connect to Aurora

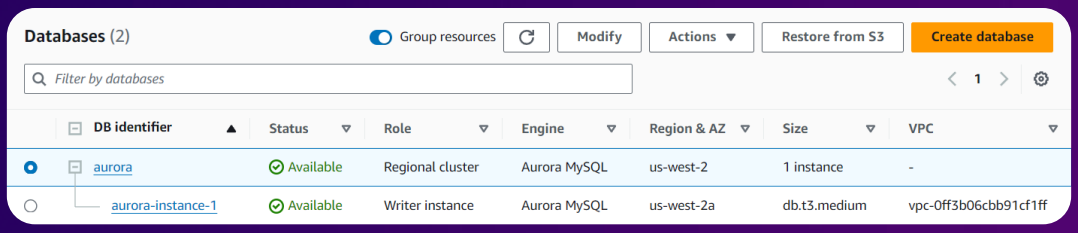
Step 1: Install the DB client

To install the MariaDB client, run the following command.

```
sh-4.2$ sudo yum install mariadb -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core
Resolving Dependencies
--> Running transaction check
---> Package mariadb.x86_64 1:5.5.68-1.amzn2.0.1 will be installed
--> Finished Dependency Resolution
```

Step 2: Review the Aurora DB cluster

Go back to the RDS Management Console, navigate to the **Databases** section, and select the aurora DB cluster.



aws re/start

Task 3

Configure the Amazon EC2 Linux instance to connect to Aurora

Step 3: Review Endpoints

Choose the **Connectivity & Security** tab, and in the **Endpoints** section, copy the Endpoint name for the Writer instance.

Step 4: Connect to Aurora

To connect to the Aurora instance, run the following command.

```
sh-4.2$ mysql -u admin --password='admin123' -h aurora.cluster-c7ewsm2ky6s6.us-west-2.rds.amazonaws.com
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MySQL connection id is 104
Server version: 8.0.28 Source distribution

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

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

MySQL [(none)]>
```

aws re/start

Task 4

Create a table and insert and query records

Step 1: Switch to the world database

To list the available databases and switch to the **world** database, run the following commands.

```
MySQL [(none)]> SHOW DATABASES;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| mysql              |
| performance_schema |
| sys                |
| world              |
+--------------------+
5 rows in set (0.00 sec)

MySQL [(none)]> USE world;
Database changed
MySQL [world]>
```

Step 2: Create a new table

To create a new table in the **world** database, run the following command.

```
MySQL [world]> CREATE TABLE `country` (
    -> `Code` CHAR(3) NOT NULL DEFAULT '',
    -> `Name` CHAR(52) NOT NULL DEFAULT '',
    -> `Continent` 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 '',
    -> `Capital` INT(11) DEFAULT NULL,
    -> `Code2` CHAR(2) NOT NULL DEFAULT '',
    -> PRIMARY KEY (`Code`)
    -> );
Query OK, 0 rows affected, 7 warnings (0.04 sec)
```

Task 4

Create a table and insert and query records

Step 3: Insert new records

To insert new records into the **country** table that you just created, run the following commands.

```
MySQL [world]> INSERT INTO `country` VALUES ('GAB','Gabon','Africa','Central Africa',267668.00,1960,1226000,
50.1,5493.00,5279.00,'Le Gabon','Republic',902,'GA');
Query OK, 1 row affected (0.00 sec)

MySQL [world]> INSERT INTO `country` VALUES ('IRL','Ireland','Europe','British Islands',70273.00,1921,377510
0,76.8,75921.00,73132.00,'Ireland/Éire','Republic',1447,'IE');
Query OK, 1 row affected (0.00 sec)

MySQL [world]> INSERT INTO `country` VALUES ('THA','Thailand','Asia','Southeast Asia',513115.00,1350,6139900
0,68.6,116416.00,153907.00,'Prathet Thai','Constitutional Monarchy',3320,'TH');
Query OK, 1 row affected (0.00 sec)

MySQL [world]> INSERT INTO `country` VALUES ('CRI','Costa Rica','North America','Central America',51100.00,1
821,4023000,75.8,10226.00,9757.00,'Costa Rica','Republic',584,'CR');
Query OK, 1 row affected (0.00 sec)

MySQL [world]> INSERT INTO `country` VALUES ('AUS','Australia','Oceania','Australia and New Zealand',7741220
.00,1901,18886000,79.8,351182.00,392911.00,'Australia','Constitutional Monarchy, Federation',135,'AU');
Query OK, 1 row affected (0.01 sec)

MySQL [world]>
```

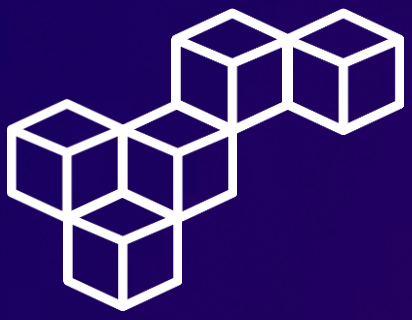
Step 4: Query the table

To query the table, run the following SELECT statement.

```
MySQL [world]> SELECT * FROM country WHERE GNP > 35000 and Population > 10000000;
+------+-----------+-----------+--------------------------+-------------+------------+------------+----------------+-----------+-----------+-----------+--------------------------------+---------+-------+
| Code | Name      | Continent | Region                   | SurfaceArea | IndepYear  | Population | LifeExpectancy | GNP       | GNPOld    | LocalName | GovernmentForm                 | Capital | Code2 |
+------+-----------+-----------+--------------------------+-------------+------------+------------+----------------+-----------+-----------+-----------+--------------------------------+---------+-------+
| AUS  | Australia | Oceania   | Australia and New Zealand |  7741220.00 |       1901 |   18886000 |           79.8 | 351182.00 | 392911.00 | Australia | Constitutional Monarchy, Federation |     135 | AU    |
| THA  | Thailand  | Asia      | Southeast Asia           |   513115.00 |       1350 |   61399000 |           68.6 | 116416.00 | 153907.00 | Prathet Thai | Constitutional Monarchy        |    3320 | TH    |
+------+-----------+-----------+--------------------------+-------------+------------+------------+----------------+-----------+-----------+-----------+--------------------------------+---------+-------+
2 rows in set (0.00 sec)

MySQL [world]>
```

aws re/start



Conclusions

Amazon Aurora
Amazon Aurora offers high performance and scalability, making it an ideal choice for managing large datasets efficiently.

Create an Aurora instance
Creating an Aurora instance provides users with a robust and reliable database solution tailored to their specific needs.

Connect to an Amazon EC2 instance
Connecting to an Amazon EC2 instance enables seamless integration and utilization of Aurora's advanced features.

Configure the EC2 instance to connect to Aurora
Configuring the EC2 instance to connect to Aurora ensures smooth communication and data accessibility between applications and databases.

Query the Aurora instance
Querying the Aurora instance allows for extracting valuable insights and performing data analysis, supporting informed decision-making processes.

aws re/start

Cristhian Becerra

cristhian-becerra-espinoza

+51 951 634 354

cristhianbecerra99@gmail.com

Lima, Peru