



nextwork.org

Connect a Web App to Amazon Aurora



Sanjana Tripathy

The screenshot shows a web browser window with the URL `ec2-13-203-229-137.ap-south-1.compute.amazonaws.com/SamplePage.php`. The page title is "Sample page". It contains two input fields: "NAME" and "ADDRESS". Below these fields is a table with columns "ID", "NAME", and "ADDRESS". The table has two rows:

ID	NAME	ADDRESS
1	sanjana	Pune,Maharashtra
2	Neha	San Francisco,Golf Road

A circular profile picture of a young woman with dark hair, wearing a pink top and blue pants, sitting on a blue chair.

Sanjana Tripathy
NextWork Student

nextwork.org

Introducing Today's Project!

What is Amazon Aurora?

Amazon Aurora is a fully managed relational database engine designed for high performance, availability, and scalability. It is useful because it automates backups, replication, and patching—reducing operational overhead while ensuring reliability.

How I used Amazon Aurora in this project

In today's project, I used Amazon Aurora to store and manage data entered through my web application. This enabled a secure, scalable, and efficient backend solution, ensuring that all user inputs were reliably captured and persisted in the database.

One thing I didn't expect in this project was...

One thing I didn't expect in this project was how enjoyable it would be to work with the command line. Navigating, configuring, and managing services through CLI turned out to be both empowering and fun to learn.



Sanjana Tripathy
NextWork Student

nextwork.org

This project took me...

This project took me around 2 hours.

Creating a Web App

```
Microsoft Windows [Version 10.0.26100.4770]
(c) Microsoft Corporation. All rights reserved.

C:\Users\tripa\OneDrive\Documents\Nextwork>dir
 Volume in drive C is OS
 Volume Serial Number is F8FC-D6B6

 Directory of C:\Users\tripa\OneDrive\Documents\Nextwork

25-07-2025 22:05    <DIR>          .
25-07-2025 22:05    <DIR>          ..
25-07-2025 19:36           1,674 NextWorkAuroraApp.pem
                           1 File(s)      1,674 bytes
                           2 Dir(s)   126,244,995,072 bytes free

C:\Users\tripa\OneDrive\Documents\Nextwork>ssh -i NextWorkAuroraApp.pem ec2-user@ec2-13-201-185-223.ap-south-1.compute.amazonaws.com
#_
`--\_\####_          Amazon Linux 2023
`--\_\####_\
`--\_\###_\
`--\_\#/   https://aws.amazon.com/linux/amazon-linux-2023
`--\_\`-->
`--\_\`-->/
`--\_\`-->/`-->
`--\_\`-->/`-->
`--\_\`-->/`-->
```

To connect to my EC2 instance, I opened the CMD from the directory containing my .pem file. Then, I used the following command: ssh -i NextWorkAuroraApp.pem ec2-user@<your-ec2-public-dns> (replacing <your-ec2-public-dns> with the actual Public DNS).

To help me create my web app, I first ran sudo dnf update -y to update packages, then installed Apache, PHP, and MySQL libraries using sudo dnf install -y httpd php php-mysqli mariadb105. Finally, I started Apache with sudo systemctl start httpd.

Sanjana Tripathy
NextWork Student

nextwork.org

Connecting my Web App to Aurora

I set up my EC2 instance's connection details to the Aurora database by editing the dbinfo.inc file using the nano text editor, adding the Writer endpoint, username, password, and database name to enable secure web app connectivity.

The screenshot shows a terminal window titled "ec2-user@ip-172-31-8-163:~" with the file "dbinfo.inc" open in the nano text editor. The file contains the following PHP code:

```
GNU nano 8.3                               dbinfo.inc                                Modified
<?php
define('DB_SERVER', 'nextwork-db-cluster-instance-1.c3uaooa4qxdb.ap-south-1.rds.amazonaws.com');
define('DB_USERNAME', 'admin');
define('DB_PASSWORD', 'Sanjana020502');
define('DB_DATABASE', 'sample');
?>
```



My Web App Upgrade

Next, I upgraded my web app by creating a `SamplePage.php` file in the `/var/www/html` directory. This PHP script stores data entered via the web app directly into the Aurora database I created earlier—enabling real-time data handling.

The screenshot shows a web browser window with the URL `ec2-13-203-229-137.ap-south-1.compute.amazonaws.com/SamplePage.php`. The page title is "Sample page". It contains two input fields: "NAME" and "ADDRESS", each with a corresponding text input box. Below these fields is a button labeled "Add Data". At the bottom of the page, there is a table with three columns: "ID", "NAME", and "ADDRESS". The table currently has one row of data: ID 1, NAME Sanjana, and ADDRESS Tripathy.

ID	NAME	ADDRESS
1	Sanjana	Tripathy

Testing my Web App

To ensure my web app was functioning correctly, I connected to the Aurora database using MySQL CLI and ran SQL queries to verify that the data entered through the web interface was successfully stored and reflected in the EMPLOYEES table.

```
ec2-user@ip-172-31-8-163:/var/lib/mysql$ MySQL [(none)]> SHOW DATABASES;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| performance_schema |
| sample |
| sys |
+-----+
5 rows in set (0.001 sec)

MySQL [(none)]> USE sample;
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
MySQL [sample]> SHOW TABLES;
+-----+
| Tables_in_sample |
+-----+
| EMPLOYEES |
+-----+
1 row in set (0.002 sec)

MySQL [sample]> DESCRIBE EMPLOYEES;
+-----+-----+-----+-----+-----+-----+
| Field | Type      | Null | Key | Default | Extra       |
+-----+-----+-----+-----+-----+-----+
| ID    | int unsigned | NO   | PRI  | NULL    | auto_increment |
| NAME  | varchar(45)  | YES  |      | NULL    |              |
| ADDRESS | varchar(98) | YES  |      | NULL    |              |
+-----+-----+-----+-----+-----+-----+
3 rows in set (0.002 sec)

MySQL [sample]> SELECT * FROM EMPLOYEES;
+-----+-----+-----+
| ID  | NAME | ADDRESS |
+-----+-----+-----+
| 1   | sanjana | Pune, Maharashtra |
| 2   | Neha | San Francisco, Golf Road |
+-----+-----+-----+
2 rows in set (0.001 sec)
```



nextwork.org

The place to learn & showcase your skills

Check out nextwork.org for more projects

