



# Connect a Web App to Amazon Aurora



Joba Amunigun

<https://www.linkedin.com/in/dvoice>

## Sample page

NAME

ADDRESS

Add Data

ID	NAME	ADDRESS
1	Mr Eddy	Lagos State, Nigeria
2	Hannah Smith	Oyo State, Nigeria
3	Patrick Jane	Sacramento, United States
4	Teresa Lisbon	Sacramento, United States
5	Kimball Cho	Sacramento, United States
6	Grace Van Pelt	Sacramento, United States
7	Wayne Rigsby	Sacramento, United States



# Introducing Today's Project!

## What is Amazon Aurora?

Amazon Aurora is a fully managed relational database engine offered by AWS that is compatible with MySQL and PostgreSQL. It combines the speed and reliability of high-end commercial databases with the simplicity and cost-effectiveness of open-source databases. Aurora is useful because it automatically handles time-consuming tasks like provisioning, patching, backup, recovery, and failover, allowing developers to focus on building applications. It's highly scalable, fault-tolerant, and offers better performance than standard MySQL—with up to 5x faster throughput—making it ideal for modern cloud-based applications.

## How I used Amazon Aurora in this project

In today's project, I used Amazon Aurora to create a highly available, MySQL-compatible relational database cluster. I connected it to an EC2 instance that acted as a web server, enabling my web app to store and retrieve employee data in real time. Aurora's scalability and built-in failover features made it a reliable choice for handling database operations securely and efficiently in the cloud.

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

One thing I didn't expect in this project was how many permissions and ownership settings I needed to adjust manually in the EC2 instance just to create and modify files. It really showed me how important user roles and security are when managing servers in AWS.

## This project took me...

This project took me approximately 3–4 hours to complete. That included setting up the Aurora database, configuring the EC2 instance, troubleshooting permissions, and verifying everything worked together seamlessly.



# Creating a Web App

```
PS C:\Users\USER\Desktop\NextWork> ssh -i NextWorkAuroraApp.pem ec2-user@ec2-18-221-61-181.us-east-2.compute.amazonaws.com
The authenticity of host 'ec2-18-221-61-181.us-east-2.compute.amazonaws.com (18.221.61.181)' can't be established.
ED25519 key fingerprint is SHA256:Bt0s2LBGCoQykae8wx1Uu6VvHTp3yAvZccBYejxUoxrk.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-18-221-61-181.us-east-2.compute.amazonaws.com' (ED25519) to the list of known hosts.

          #_
         /###
        /###\
       /##/
      /# \
     /#  \
    /#   \
   /#    \
  /#     \
 /#      \
/_#       \
          Amazon Linux 2023
          https://aws.amazon.com/linux/amazon-linux-2023

[ec2-user@ip-172-31-34-109 ~]$ client_loop: send disconnect: Connection reset
PS C:\Users\USER\Desktop\NextWork> ^S
```

To connect to my EC2 instance, I first located my ".pem" key file (NextWorkAuroraApp.pem) and moved it into a dedicated nextwork folder on my Desktop for easy access. Using the terminal, I navigated into this folder by running "cd Desktop/nextwork" and confirmed the key file was present using the ls command. Before connecting, I updated the file permissions using "chmod 400 NextWorkAuroraApp.pem" to ensure it was secure and accessible only by me, as required by AWS SSH standards. I then successfully established a secure connection to my EC2 instance using the SSH command: "ssh -i NextWorkAuroraApp.pem ec2-user@ec2-18-221-61-181.us-east-2.compute.amazonaws.com" This allowed me to remotely access the EC2 instance's terminal environment from my local machine.



**Joba Amunigun**  
NextWork Student

[nextwork.org](http://nextwork.org)

---

To help me create my web app, I ran the following commands on my WINDOWS Powershell (After connecting to my EC2 instance through SSH) "sudo dnf update -y" , "sudo dnf install -y httpd php php-mysqli mariadb105" , and "sudo systemctl start httpd " Here's what each component does: httpd: Installs and starts the Apache web server, which delivers web pages to users. php: Installs PHP, the language used to write server-side web logic. php-mysqli: Adds MySQL support to PHP so it can talk to my Aurora database. mariadb105: Installs the MariaDB client, a MySQL-compatible tool to let EC2 communicate with the database. After installation, I tested the setup by visiting my EC2 instance's public DNS in the browser:"<http://ec2-18-221-61-181.us-east-2.compute.amazonaws.com>" This confirmed that my web server was running and ready to connect to the database. (It Works!)



# Connecting my Web App to Aurora

To connect my EC2 instance to my Aurora database, I: Navigated to the EC2 web directory at /var/www and changed its ownership using sudo chown ec2-user:ec2-user /var/www to gain the necessary permissions for creating and modifying files. Created a new subfolder named inc to store configuration files, then navigated into it using cd inc. Created a configuration file named dbinfo.inc to securely store

The screenshot shows a terminal window with a dark background. At the top, the title bar displays "ec2-user@ip-172-31-26-2:~/.var/www/inc" on the left, "dbinfo.inc" in the center, and "Modified" on the right. The main area of the window contains the following PHP code:

```
<?php
define('DB_SERVER', 'nextwork-db-cluster-instance-1.c3oy8cw0egr.us-east-2.rds.amazonaws.com');
define('DB_USERNAME', 'root');
define('DB_PASSWORD', 'Mysql123');
define('DB_DATABASE', 'sample');
?>
```

At the bottom of the window, there is a menu bar with various options like Help, Exit, Read File, Replace, Cut, Paste, Execute, Justify, Location, Go To Line, Undo, Redo, Set Mark, Copy, To Bracket, Where Was, and Where Is. The "Modified" status in the title bar indicates that the file has been edited since it was last saved.



# My Web App Upgrade

To upgrade my web app and connect it to my Aurora database, I: Navigated to the EC2 web root directory at /var/www/html, where all public-facing website files are stored. Created a new PHP file called SamplePage.php using the nano text editor. Wrote a dynamic PHP script that: Connected to my Aurora database using the credentials stored in dbinfo.inc Created an EMPLOYEES table if it didn't already exist Allowed users to add employee data

## Sample page

NAME	ADDRESS	Add Data
<input type="text"/>	<input type="text"/>	<input type="button" value="Add Data"/>
ID	NAME	ADDRESS

# Testing my Web App

To make sure my web app was working correctly, I:

- Installed the MySQL CLI on my EC2 instance to directly interact with my Aurora MySQL database.
- Connected to the Aurora Writer instance using the MySQL command: mysql -h nextwork-db-cluster-instance-1.c3oy8cw0egrr.us-east-2.rds.amazonaws.com -P 3306 -u admin -p
- Logged in and ran the following SQL commands to inspect the data: SHOW DATABASES; – to confirm that the sample database

```
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(90) | YES | | NULL | |
+-----+-----+-----+-----+-----+-----+
3 rows in set (0.003 sec)

MySQL [sample]> SELECT * FROM EMPLOYEES;
+-----+-----+-----+
| ID | NAME | ADDRESS |
+-----+-----+-----+
| 1 | Mr Eddy | Lagos State, Nigeria |
| 2 | Hannah Smith | Oyo State, Nigeria |
| 3 | Patrick Jane | Sacramento, United States |
| 4 | Teresa Lisbon | Sacramento, United States |
| 5 | Kimball Cho | Sacramento, United States |
| 6 | Grace Van Pelt | Sacramento, United States |
| 7 | Wayne Rigsby | Sacramento, United States |
+-----+-----+-----+
```



[nextwork.org](https://nextwork.org)

# The place to learn & showcase your skills

Check out [nextwork.org](https://nextwork.org) for more projects

