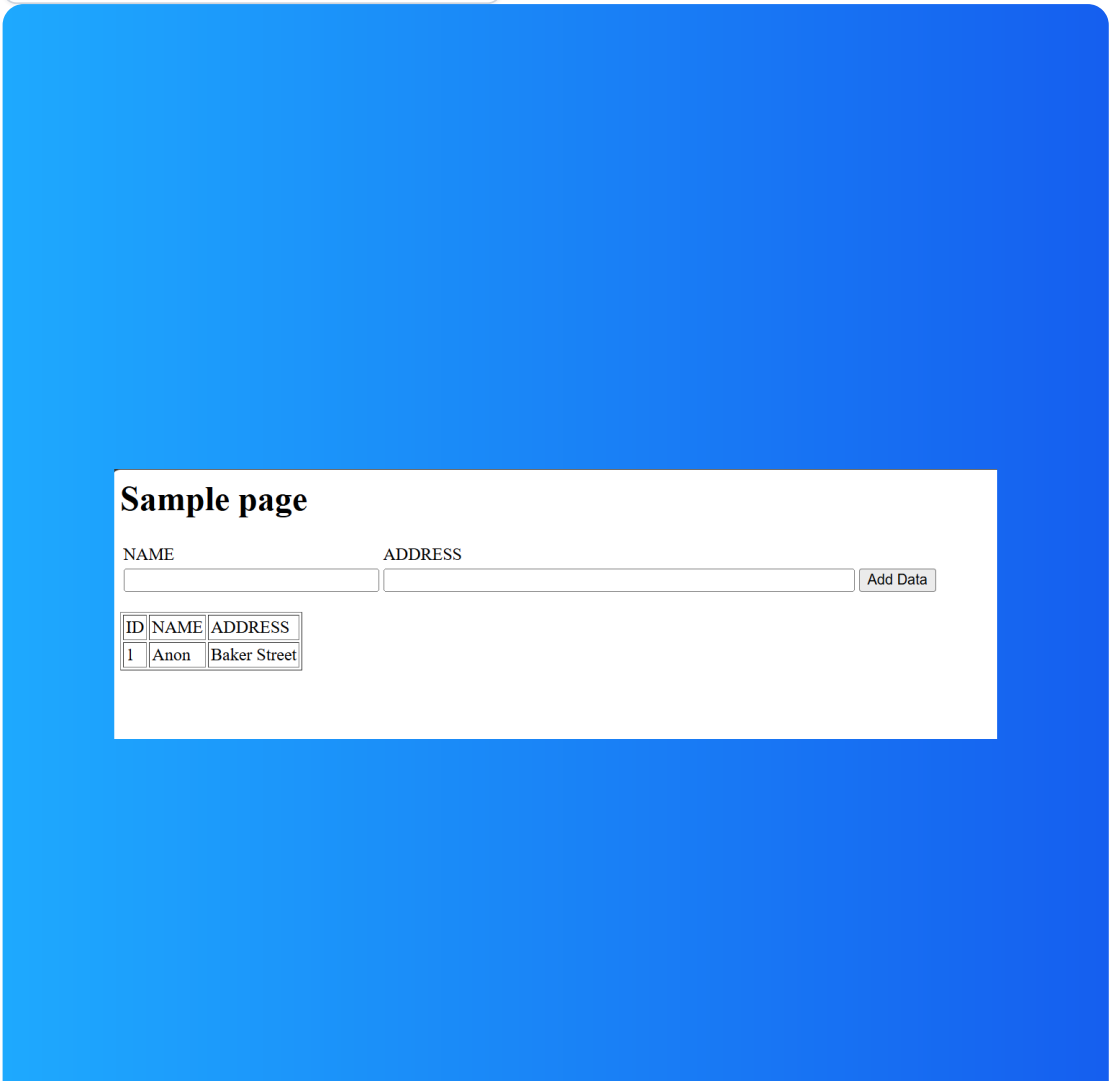
Connect a Web App to Amazon Aurora

Unmilan Mukherjee



Introducing Today's Project!

What is Amazon Aurora?

Amazon Aurora is a MySQL Database provided by AWS to handle high demand database situations.

How I used Amazon Aurora in this project

I used Aurora to store data pertaining to the web app.

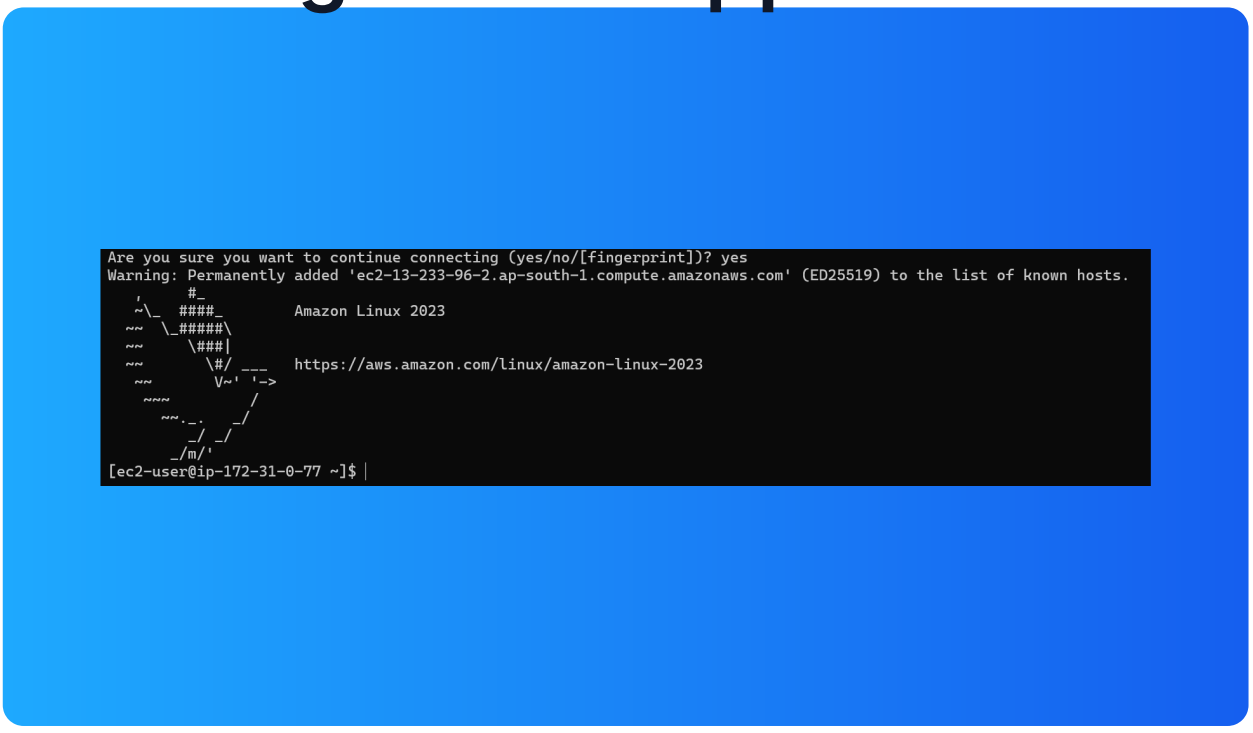
One thing I didn't expect in this project was...

I did not expect how easy it was going to be to use the CLI for both my instance and MySQL to accomplish my tasks for this project.

This project took me...

This project took me 50 minues.

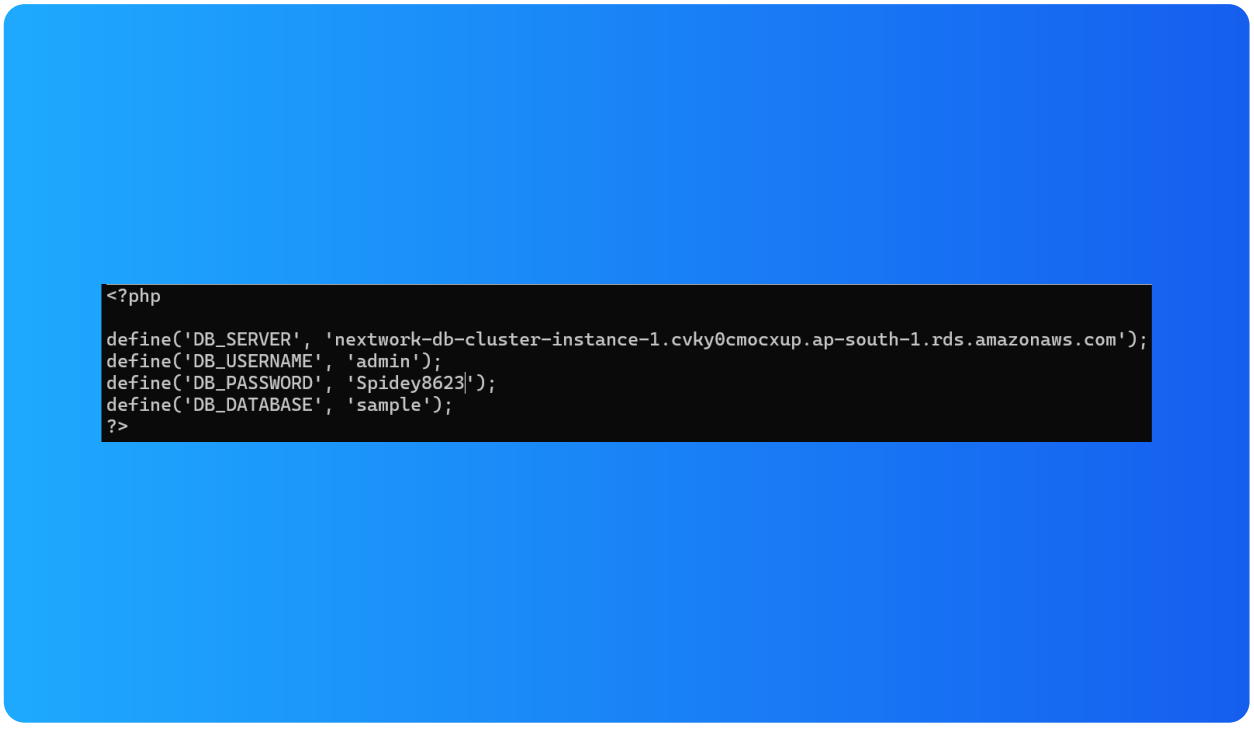
Creating a Web App



To help me create my web app, I first had to install a bunch of dependencies: PHP, MariaDB, Apache web server, php-mysqli and httpd.

To connect to my EC2 instance, I first downloaded the pem file created during key creation, then used SSH with the public IPv4 and key to login with powershell on Windows. Command: ssh -i YOUR_PEM_FILE_NAME ec2-user@YOUR_EC2_IPv4_ADDRESS

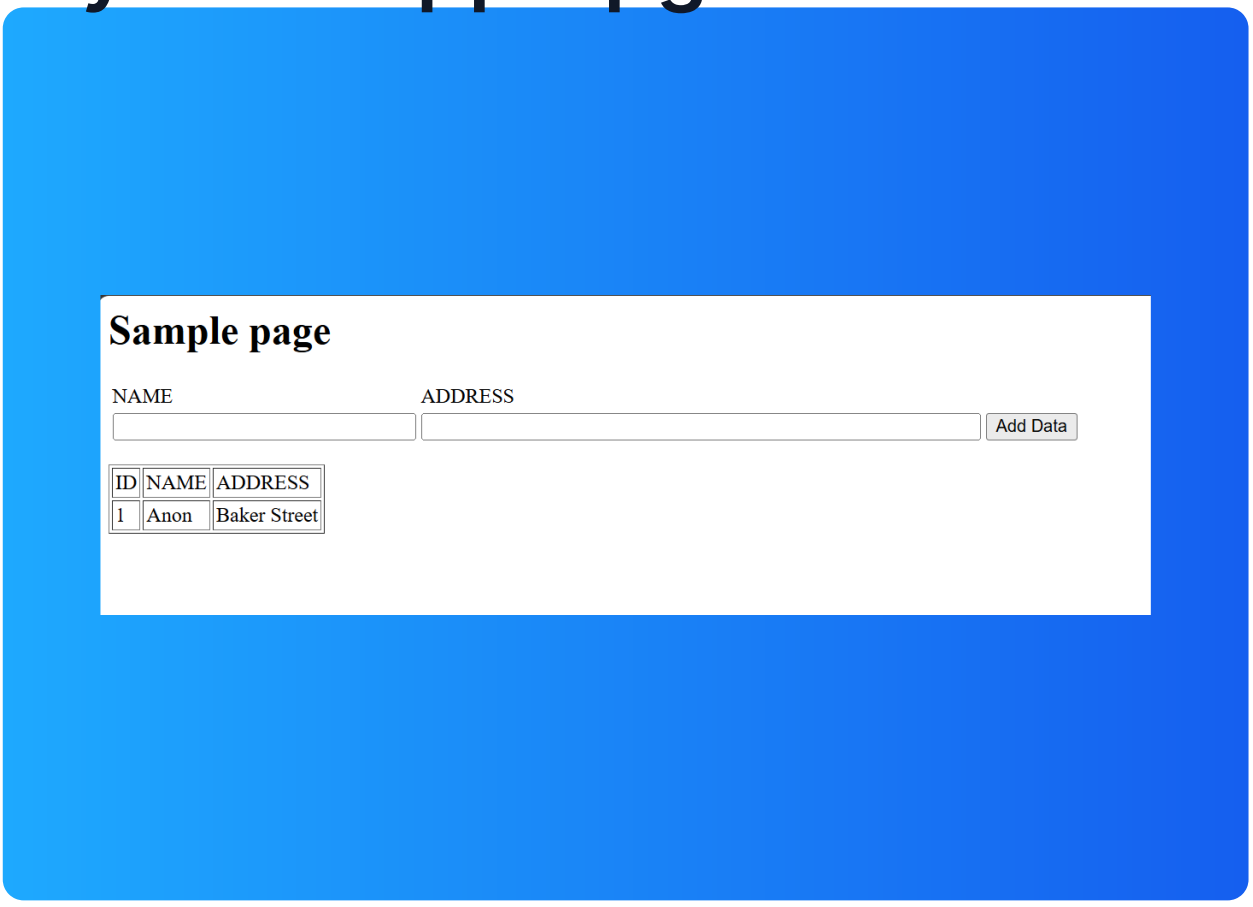
Connecting my Web App to Aurora



I set up my EC2 instance's connection details to my database by editing the dbinfo.inc file in the instance.

To connect to my EC2 instance, I first downloaded the pem file created during key creation, then used SSH with the public IPv4 and key to login with powershell on Windows. Command: ssh -i YOUR_PEM_FILE_NAME ec2-user@YOUR_EC2_IPv4_ADDRESS

My Web App Upgrade



Next, I upgraded my web app by creating a php page that pulls the data about our database connection and updates the page to look like the one in the screenshot.

Testing my Web App

To make sure my web app was working correctly, I first tested entering values on the webpage. After confirming that it was working I logged into my database using MySQL CLI to double check if the data was being udpated on Aurora.