



NextWork.org

Connect a Web App to Amazon Aurora

G Gloria

Engine options

Engine type: [Info](#)

<input checked="" type="radio"/> Aurora (MySQL Compatible) 	<input type="radio"/> Aurora (PostgreSQL Compatible) 
<input type="radio"/> MySQL 	<input type="radio"/> PostgreSQL 
<input type="radio"/> MariaDB 	<input type="radio"/> Oracle  ORACLE®
<input type="radio"/> Microsoft SQL Server 	<input type="radio"/> IBM Db2 

Engine version
Aurora MySQL 3.05.2 (compatible with MySQL 8.0.32) - default for major version 8.0



Introducing Today's Project!

What is Amazon Aurora?

Amazon Aurora is a database service that works with MySQL. It's fast, reliable, and scales easily as your needs grow. It handles backups automatically, keeps your data safe, and is great for apps that need strong performance and security.

How I used Amazon Aurora in this project

I used Amazon Aurora in today's project to set up a reliable and scalable database. I created an Aurora cluster, configured it to store and manage data securely, connected it to an EC2 instance, prepared it to support a web application in the future

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

One thing I didn't expect in this project was how easily Amazon Aurora connects to an EC2 instance. It was simple to link the database to the server, and Aurora's features like backups and scaling made setting up a reliable system much easier.

This project took me...

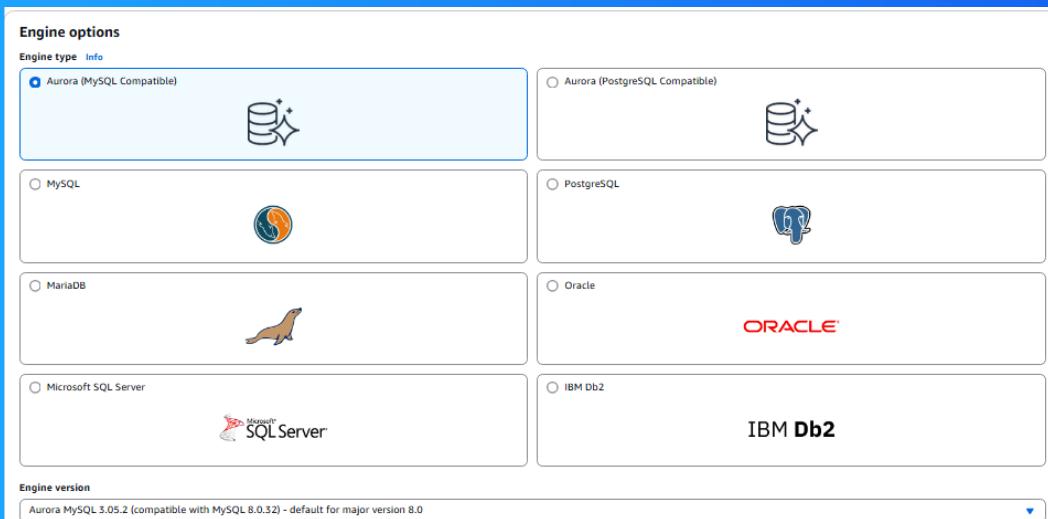
This project took me 15 minutes to complete. Setting up the Aurora database, creating the cluster, and connecting it to the EC2 instance was quick and straightforward, thanks to the tools and automation provided by AWS.

In the first part of my project...

Creating an Aurora Cluster

A relational database is a type of database that organizes data into tables with rows and columns, like a spreadsheet. Each row represents a record, and columns represent attributes. It uses relationships between tables to keep data connected.

Aurora is a good choice when you need a fast, reliable, and scalable database. It's easy to use, supports MySQL, handles backups automatically, and scales to handle more users or data as needed, making it great for growing applications.



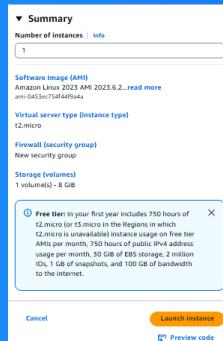
Halfway through I stopped!

I stopped creating my Aurora database because I needed to first create an EC2 instance. The EC2 instance is required to connect to the Aurora database and act as the compute resource for hosting the web application that will use the database.

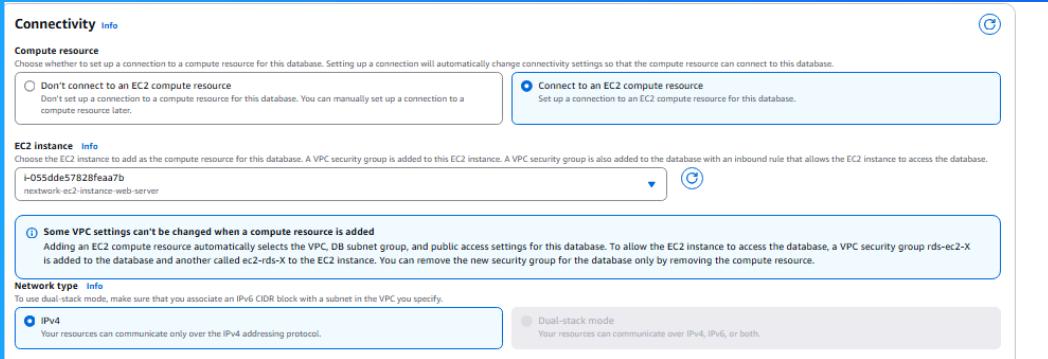
Features of my EC2 instance

I created a new key pair for my EC2 instance because it's needed to securely connect to the instance using SSH. The key pair ensures that only I, with the private key, can access the server, keeping it secure from unauthorized access.

When I created my EC2 instance, I took particular note of the Public IPv4 DNS, which is needed to locate and connect to the instance, and the Key pair name, which ensures secure access. Both are critical for accessing and managing the instance.



Then I could finish setting up my database



Aurora Database uses clusters because they make it faster, reliable, and able to handle more users. Clusters let you create copies of your database for quick reads, switch to backups if something breaks, and easily add capacity as your needs grow.



NextWork.org

Everyone should be in a job they love.

Check out nextwork.org for
more projects

