



NextWork.org

Connect a Web App to Amazon Aurora



tskhelp@gmail.com

Standard create
You set all of the configuration options, including ones for availability, security, backups, and maintenance.

Easy create
Use recommended best-practice configurations. Some configuration options can be changed after the database is created.

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
<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 0.

Enable RDS Extended Support Info
Amazon RDS Extended Support is a paid offering. By selecting this option, you consent to being charged for this offering if you are running your database major version past the RDS end of standard support date for that version. Check the end of standard support date for your major version in the Amazon Aurora documentation.

⚠ Parallel query is off by default. To enable it, use a DB instance parameter group with the aurora_parallel_query parameter enabled. [Learn more](#)

Templates
Choose a sample template to meet your use case.

Production
Use defaults for high availability and fast, consistent performance.

Dev/Test
This instance is intended for development use outside of a production environment.

Introducing Today's Project!

What is Amazon Aurora?

Amazon Aurora is a fully managed, high-performance relational database compatible with MySQL and PostgreSQL. It offers scalability, durability, and automated backups, making it ideal for high-demand applications with minimal management overhead.

How I used Amazon Aurora in this project

In today's project, I deployed an Amazon Aurora database cluster within a Virtual Private Cloud and configured Security Groups to manage access. I then connected an EC2 instance to the Aurora cluster, enabling secure and efficient data operations.

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

I didn't expect the price of Amazon Aurora to be high enough, but you can use it for free in the free tier.

This project took me...

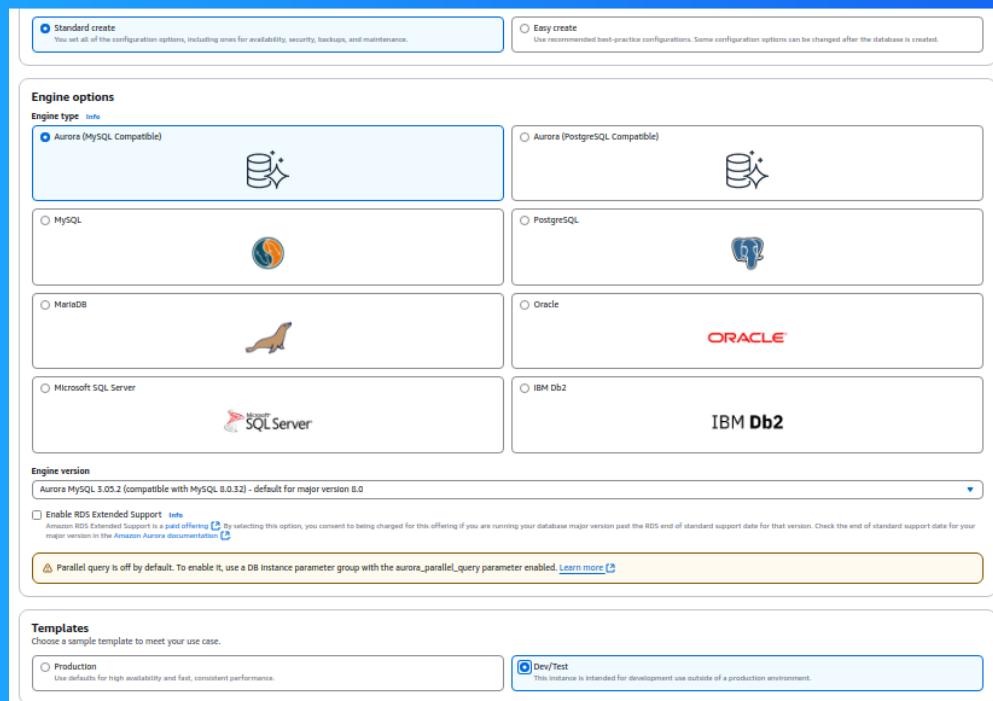
This project took me less than 1 hour to complete.

In the first part of my project...

Creating an Aurora Cluster

A relational database is a type of database that organizes data into tables, which are collections of rows and columns. Kind of like a spreadsheet! We call it "relational" because the rows relate to the columns and vice-versa.

Aurora is a good choice when you need something large-scale, with peak performance and uptime. This is because Aurora databases use clusters.



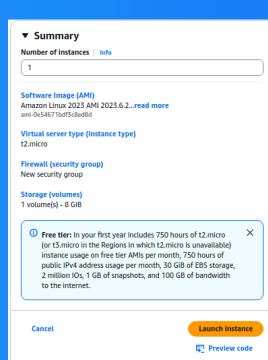
Halfway through I stopped!

I stopped creating my Aurora database because I need to select an EC2 instance to communicate with my Aurora database and I don't have one. And now I need to create an EC2 instance first.

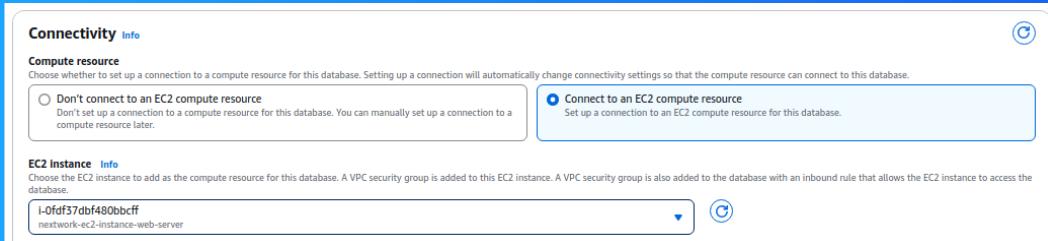
Features of my EC2 instance

I created a new key pair for my EC2 instance because I need it to access my EC2 instance. I need keys to my EC2 instance if I want to add, change, or update how my EC2 instance is running.

When I created my EC2 instance, I took particular note of Public IPv4 DNS and the Key pair name. Public IPv4 DNS is a location of my instance, and the Key pair name is a key for it.



Then I could finish setting up my database



Aurora Database uses clusters because they provide high availability and scalability by replicating data across multiple nodes, enabling fast failover, load balancing, and seamless scaling for read and write operations.



NextWork.org

Everyone should be in a job they love.

Check out nextwork.org for
more projects

