

Connect a Web App to Amazon Aurora

Erik Gonzalez

Aurora and RDS > Databases > Create database

Create database


Choose a database creation method


☒ Full configuration
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


☒ Aurora (MySQL Compatible)



☐ Aurora (PostgreSQL Compatible)



☐ MySQL


☐ PostgreSQL


☐ MariaDB


☐ Oracle


☐ Microsoft SQL Server


☐ IBM Db2


Engine version

Aurora MySQL 3.08.2 (compatible with MySQL 8.0.39) - default for major version 8.0

☐ Enable RDS Extended Support
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.

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 relational database and why it is useful because of the clusters which allow for large-scale applications.

How I used Amazon Aurora in this project

In today's project, I used Amazon Aurora to connect with an EC2 instance.

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

One thing I didn't expect in this project was how expensive it can be to run a database.

This project took me...

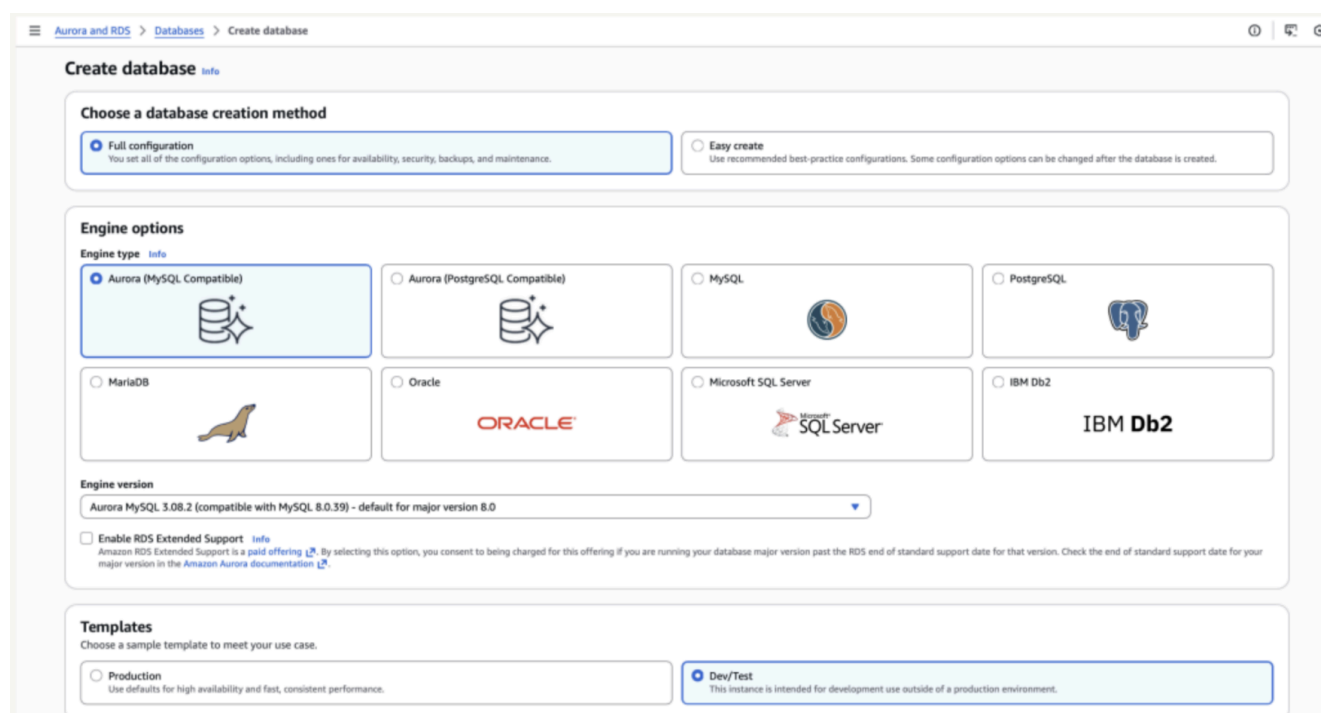
How much time did this project take you about an hour to complete.

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.

Aurora is a good choice when you have a large-scale application, perfect for relational databases. Aurora uses clusters which are great for more efficient practices.



The screenshot shows the 'Create database' page in the Amazon RDS console. The breadcrumb navigation at the top reads 'Aurora and RDS > Databases > Create database'. The page is titled 'Create database' with an 'Info' link. It is divided into several sections:

- Choose a database creation method:** Two options are shown: 'Full configuration' (selected, with a description: 'You set all of the configuration options, including ones for availability, security, backups, and maintenance.') and 'Easy create' (description: 'Use recommended best-practice configurations. Some configuration options can be changed after the database is created.').
- Engine options:** A section titled 'Engine type' with an 'Info' link. It contains eight selectable options, each with a database icon: 'Aurora (MySQL Compatible)' (selected), 'Aurora (PostgreSQL Compatible)', 'MySQL', 'PostgreSQL', 'MariaDB', 'Oracle', 'Microsoft SQL Server', and 'IBM Db2'.
- Engine version:** A dropdown menu showing 'Aurora MySQL 3.08.2 (compatible with MySQL 8.0.39) - default for major version 8.0'.
- Enable RDS Extended Support:** A checkbox that is currently unchecked. A note below it states: '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.'
- Templates:** A section titled 'Choose a sample template to meet your use case.' with two options: 'Production' (description: 'Use defaults for high availability and fast, consistent performance.') and 'Dev/Test' (selected, description: 'This instance is intended for development use outside of a production environment.').

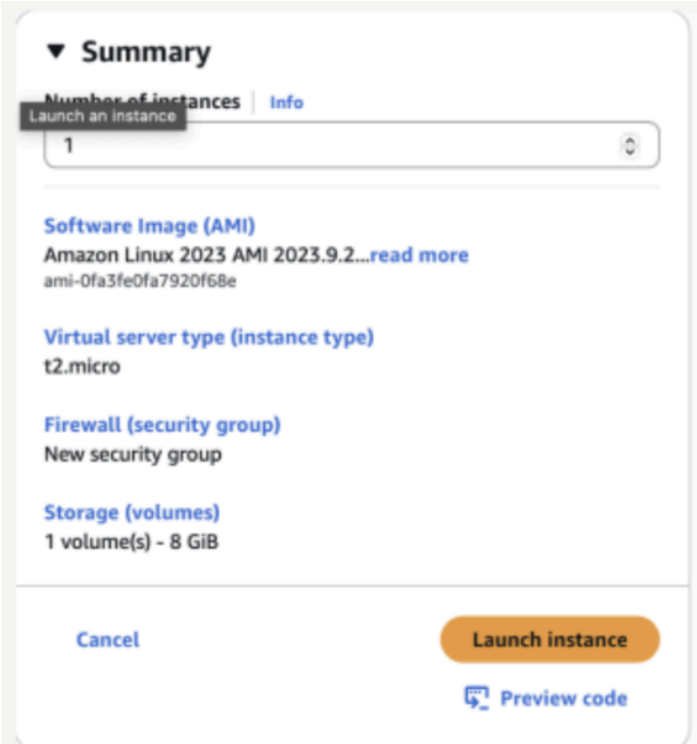
Halfway through I stopped!

I stopped creating my Aurora database because I need to create an EC2 Instance in order to connect my Aurora databases to it.

Features of my EC2 instance

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

When I created my EC2 instance, I took particular note of the public IPv4 DNS address and names as well as the key pair name so that I know what I access to with the key pair.




The screenshot shows the 'Launch Instance' summary page in the AWS Management Console. At the top, there's a 'Summary' section with a dropdown for 'Number of instances' set to 1. Below this, the configuration details are listed: 'Software Image (AMI)' is 'Amazon Linux 2023 AMI 2023.9.2...read more' with ID 'ami-0fa3fe0fa7920f68e'; 'Virtual server type (instance type)' is 't2.micro'; 'Firewall (security group)' is 'New security group'; and 'Storage (volumes)' is '1 volume(s) - 8 GiB'. At the bottom, there are 'Cancel' and 'Launch instance' buttons, along with a 'Preview code' link.

Configuration Item	Value
Number of instances	1
Software Image (AMI)	Amazon Linux 2023 AMI 2023.9.2...read more ami-0fa3fe0fa7920f68e
Virtual server type (instance type)	t2.micro
Firewall (security group)	New security group
Storage (volumes)	1 volume(s) - 8 GiB

Buttons: Cancel, Launch instance, Preview code

Then I could finish setting up my database

Connectivity [Info](#) 


Compute resource
Choose whether to set up a connection to a compute resource for this database. Setting up a connection will automatically change connectivity settings so that the compute resource can connect to this database.

☐ Don't connect to an EC2 compute resource
Don't set up a connection to a compute resource for this database. You can manually set up a connection to a compute resource later.

☒ Connect to an EC2 compute resource
Set up a connection to an EC2 compute resource for this database.

EC2 instance [Info](#)
Choose the EC2 instance to add as the compute resource for this database. A VPC security group is added to this EC2 instance. A VPC security group is also added to the database with an inbound rule that allows the EC2 instance to access the database.

i-0bd0bdd71d4b07c1a
nimbus-ec2-instance-web-server



Some VPC settings can't be changed when a compute resource is added
Adding an EC2 compute resource automatically selects the VPC, DB subnet group, and public access settings for this database. To allow the EC2 instance to access the database, a VPC security group rds-ec2-X is added to the database and another called ec2-rds-X to the EC2 instance. You can remove the new security group for the database only by removing the compute resource.

Network type [Info](#)
To use dual-stack mode, make sure that you associate an IPv6 CIDR block with a subnet in the VPC you specify.

☒ IPv4
Your resources can communicate only over the IPv4 addressing protocol.

☐ Dual-stack mode
Your resources can communicate over IPv4, IPv6, or both.

Virtual private cloud (VPC) [Info](#)

Aurora Database uses clusters because it handles large-scale applications and operations. By using clusters, the database instances are separated by the kinds of operations (ie reader instance vs writer instance.)