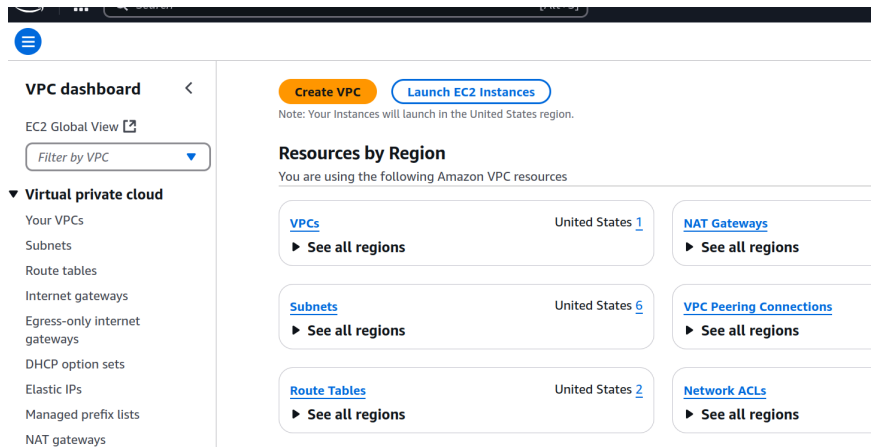


Make VPC

- 1) Go to dashboard and look up VPC in search bar. Go to VPC dashboard, and then go to your VPCS



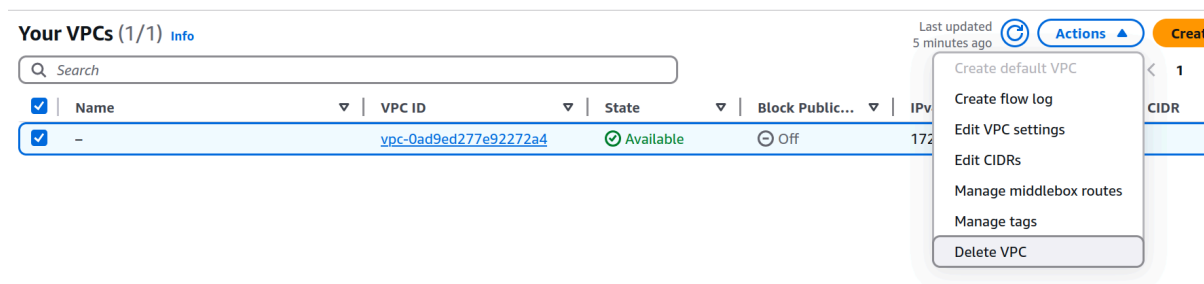
Virtual private cloud

Your VPCs

Subnets

Route tables

Delete the default VPC



Now create a new one with these settings

VPC settings

Resources to create [Info](#)
Create only the VPC resource or the VPC and other networking resources.

☐ VPC only ☒ VPC and more

Name tag auto-generation [Info](#)
Enter a value for the Name tag. This value will be used to auto-generate Name tags for all resources in the VPC.

☒ Auto-generate

IPv4 CIDR block [Info](#)
Determine the starting IP and the size of your VPC using CIDR notation.

65,536 IPs
CIDR block size must be between /16 and /28.

IPv6 CIDR block [Info](#)

☒ No IPv6 CIDR block ☐ Amazon-provided IPv6 CIDR block

Tenancy [Info](#)

Number of Availability Zones (AZs) [Info](#)
Choose the number of AZs in which to provision subnets. We recommend at least two AZs for high availability.

☐ 1 ☒ 2 ☐ 3

► Customize AZs

Number of public subnets [Info](#)
The number of public subnets to add to your VPC. Use public subnets for web applications that need to be publicly accessible over the internet.

☐ 0 ☒ 2

NAT gateways (\$) [Info](#)
Choose the number of Availability Zones (AZs) in which to create NAT gateways. Note that there is a charge for each NAT gateway.

☒ None ☐ In 1 AZ ☐ 1 per AZ

VPC endpoints [Info](#)
Endpoints can help reduce NAT gateway charges and improve security by accessing S3 directly from the VPC. By default, full access policy is used. You can customize this policy at any time.

☐ None ☒ S3 Gateway

DNS options [Info](#)

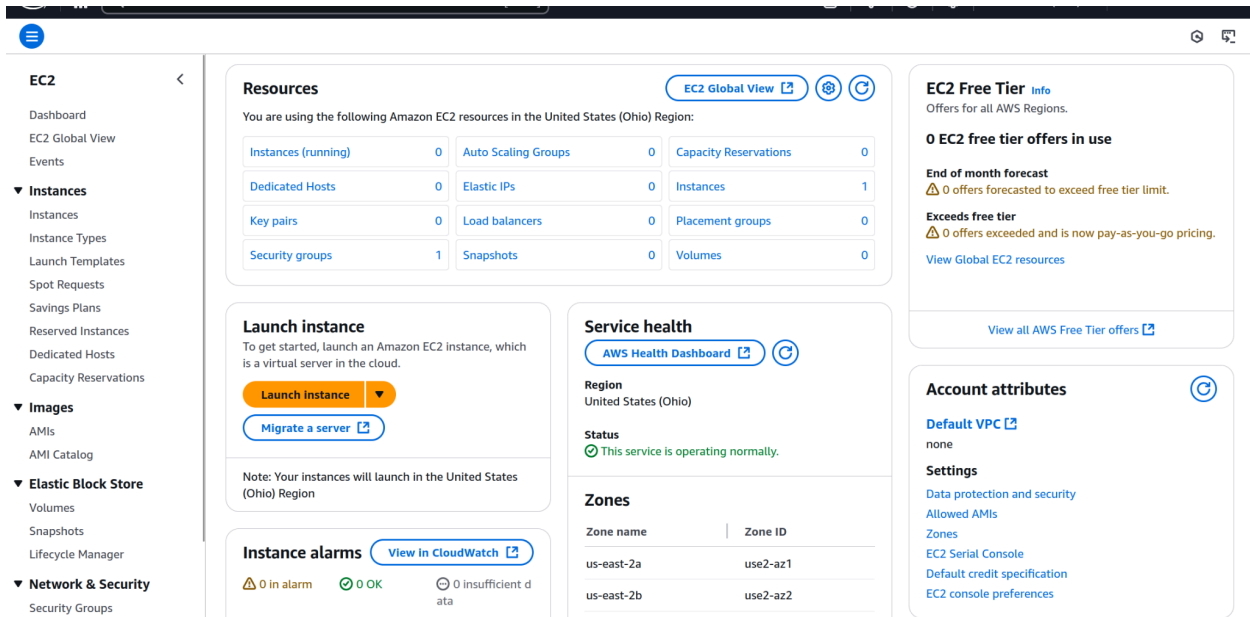
☒ Enable DNS hostnames ☒ Enable DNS resolution

▼ Additional tags
Add tags to the VPC and all resources within the VPC. Do not set the Name tag here. Set the Name tag under Name tag auto-generation above or directly in the visualizer.

You can add 49 more tags.

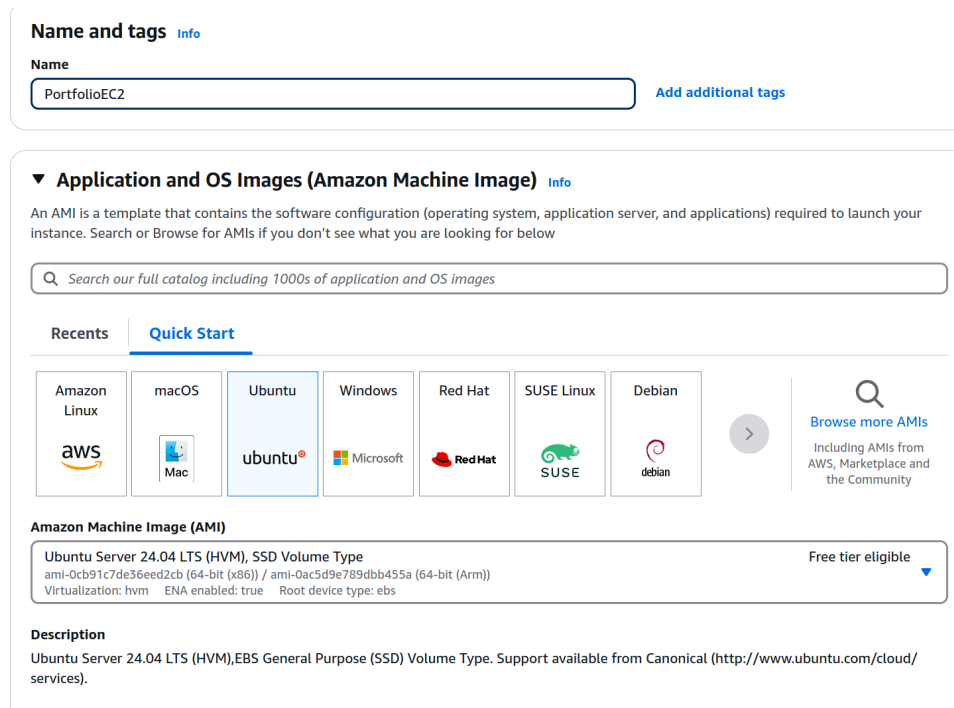
2) Making the EC2

Now go to EC2 Dashboard and launch the instance



The screenshot shows the AWS Management Console EC2 Dashboard. The left sidebar contains navigation links for EC2, including Dashboard, EC2 Global View, Events, Instances, Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity Reservations, Images, AMIs, AMI Catalog, Elastic Block Store, Volumes, Snapshots, Lifecycle Manager, and Network & Security, Security Groups. The main content area is divided into several sections: Resources (showing counts for Instances (running), Dedicated Hosts, Key pairs, Security groups, Auto Scaling Groups, Elastic IPs, Load balancers, Snapshots, Capacity Reservations, Instances, Placement groups, and Volumes), Launch instance (with buttons for Launch Instance and Migrate a server), Service health (showing Region: United States (Ohio) and Status: This service is operating normally), Instance alarms (showing 0 in alarm, 0 OK, and 0 insufficient data), EC2 Free Tier (showing 0 offers in use and end of month forecast), and Account attributes (showing Default VPC, none, and Settings). The top right corner shows the user's profile and the current region (us-east-1).

Name it and choose Ubuntu



The screenshot shows the AWS Management Console EC2 Instance Launch Wizard. The first step is "Name and tags", where the instance name is "PortfolioEC2". The second step is "Application and OS Images (Amazon Machine Image)", where the user is selecting an AMI. The "Quick Start" tab is active, showing a grid of AMIs. The selected AMI is "Ubuntu Server 24.04 LTS (HVM), SSD Volume Type" (ami-0cb91c7de36eed2cb). The description of the AMI is: "Ubuntu Server 24.04 LTS (HVM), EBS General Purpose (SSD) Volume Type. Support available from Canonical (http://www.ubuntu.com/cloud/services)." The AMI is marked as "Free tier eligible".

Make sure free tier is selected

t2.nano
Family: t2 1 vCPU 0.5 GiB Memory Current generation: true
On-Demand Ubuntu Pro base pricing: 0.0076 USD per Hour
On-Demand Windows base pricing: 0.0081 USD per Hour On-Demand Linux base pricing: 0.0058 USD per Hour
On-Demand SUSE base pricing: 0.0058 USD per Hour

t2.micro Free tier eligible
Family: t2 1 vCPU 1 GiB Memory Current generation: true
On-Demand Ubuntu Pro base pricing: 0.0134 USD per Hour
On-Demand Linux base pricing: 0.0116 USD per Hour On-Demand SUSE base pricing: 0.0116 USD per Hour
On-Demand Windows base pricing: 0.0162 USD per Hour On-Demand RHEL base pricing: 0.026 USD per Hour

t2.small
Family: t2 1 vCPU 2 GiB Memory Current generation: true
On-Demand SUSE base pricing: 0.053 USD per Hour On-Demand RHEL base pricing: 0.0376 USD per Hour
On-Demand Ubuntu Pro base pricing: 0.025 USD per Hour
On-Demand Windows base pricing: 0.032 USD per Hour On-Demand Linux base pricing: 0.023 USD per Hour

t2.medium
Family: t2 2 vCPU 4 GiB Memory Current generation: true

t2.micro Free tier eligible
Family: t2 1 vCPU 1 GiB Memory Current generation: true
On-Demand Ubuntu Pro base pricing: 0.0134 USD per Hour
On-Demand Linux base pricing: 0.0116 USD per Hour On-Demand SUSE base pricing: 0.0116 USD per Hour
On-Demand Windows base pricing: 0.0162 USD per Hour On-Demand RHEL base pricing: 0.026 USD per Hour

Additional costs apply for AMIs with pre-installed software

l (http:

C

C

Choose no key pair

We will connect to it through EC2 Connect

Edit the network settings

▼ **Network settings** [Info](#)

[Edit](#)

And Use these settings

Network settings [Info](#)

VPC - required [Info](#)

vpc-0e23050b43b76917b (Portfolio VPC-vpc)
10.0.0.0/16

Subnet [Info](#)

subnet-0cf4eadca9659721f
Portfolio VPC-subnet-public1-us-east-2a
VPC: vpc-0e23050b43b76917b Owner: 831926625893 Availability Zone: us-east-2a
Zone type: Availability Zone IP addresses available: 4091 CIDR: 10.0.0.0/20

Create new subnet

Auto-assign public IP [Info](#)

Enable

Additional charges apply when outside of free tier allowance

Firewall (security groups) [Info](#)

A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

Create security group

Select existing security group

Security group name - required

EC2 Inbound

This security group will be added to all network interfaces. The name can't be edited after the security group is created. Max length is 255 characters. Valid characters: a-z, A-Z, 0-9, spaces, and _-:/()#,@[]+=&:{}!\$*

Description - required [Info](#)

Connections inbound to EC2 Instance

Inbound Security Group Rules

Security group rule 1 (TCP, 22, 0.0.0.0/0)

Remove

Type [Info](#)

ssh

Protocol [Info](#)

TCP

Port range [Info](#)

22

Source type [Info](#)

Anywhere

Source [Info](#)

Q Add CIDR, prefix list or security group

Description - optional [Info](#)

e.g. SSH for admin desktop

Now Launch instance and wait for it to load

3) Making your RDS

Search up RDS (Should show something like Aurora and RDS) and go to the RDS Dashboard. Then click on create database

RDS

> Dashboard

Amazon RDS

<

Dashboard

Databases

Query Editor

Performance insights

Snapshots

Exports in Amazon S3

Automated backups

Reserved instances

Proxies

Subnet groups

Parameter groups

Option groups

Custom engine versions

Zero-ETL integrations [New](#)

Events

Event subscriptions

Recommendations 0

Certificate update

Resources

You are using the following Amazon RDS resources in the US East (Ohio) region (used/quota)

DB Instances (0/40)
Allocated storage (0 TB/100 TB)
Instances and storage include Neptune and DocumentDB. [Increase DB instances limit](#)

DB Clusters (0/40)

Reserved instances (0/40)

Snapshots (2)
Manual
DB Cluster (0/100)
DB Instance (1/100)
Automated
DB Cluster (0)
DB Instance (1)
Recent events (10)
Event subscriptions (0/20)

Parameter groups (1)
Default (1)
Custom (0/100)

Option groups (1)
Default (1)
Custom (0/20)

Subnet groups (1/50)

Supported platforms [VPC](#)
Default network none

Create database

Amazon Relational Database Service (RDS) makes it easy to set up, operate, and scale a relational database in the cloud.

Create database

You can use a backup from Amazon S3 to create a new Aurora MySQL database.

Restore from S3

Note: your DB instances will launch in the **US East (Ohio)** region

Use these settings


Choose a database creation method


☐ 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.


Configuration


Engine type [Info](#)


☐ Aurora (MySQL Compatible)


☐ Aurora (PostgreSQL Compatible)


☒ MySQL


☐ PostgreSQL


☐ MariaDB


☐ Oracle


☐ Microsoft SQL Server

Edition

☒ MySQL Community

DB instance size

☐ Production

db.r7g.xlarge
4 vCPUs
32 GiB RAM
500 GiB
1.114 USD/hour

☐ Dev/Test

db.r7g.large
2 vCPUs
16 GiB RAM
100 GiB
0.255 USD/hour

☒ Free tier

db.t4g.micro
2 vCPUs
1 GiB RAM
20 GiB
0.019 USD/hour

DB instance identifier

Type a name for your DB Instance. The name must be unique across all DB instances owned by your AWS account in the current AWS Region.

PortfolioRDS

The DB Instance Identifier is case-insensitive, but is stored as all lowercase (as in "mydbinstance"). Constraints: 1 to 63 alphanumeric characters or hyphens. First character must be a letter. Can't contain two consecutive hyphens. Can't end with a hyphen.

Master username [Info](#)

Type a login ID for the master user of your DB instance.

admin

1 to 16 alphanumeric characters. The first character must be a letter.

Credentials management

You can use AWS Secrets Manager or manage your master user credentials.

☐

Managed in AWS Secrets Manager - *most secure*

RDS generates a password for you and manages it throughout its lifecycle using AWS Secrets Manager.

☒

Self managed

Create your own password or have RDS create a password that you manage.

☒ Auto generate password

Amazon RDS can generate a password for you, or you can specify your own password.

[i](#) You can view your credentials after you create your database. Click the 'View credential details' in the database creation banner to view the password.

Make sure we set up an EC2 Connection to the one we just created

▼ Set up EC2 connection - *optional*

You can also set up a connection to an EC2 instance after creating the database. Go to the database list page or the database details page, choose **Actions**, and then choose **Set up to EC2 connection**.

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-Oee04a7687601b4fa

PortfolioEC2



[i](#) 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.

Then create the database. Make sure you view credentials after creating! This will disappear if you navigate away from the page. Copy password and save it somewhere (notepad or somewhere else). YOU WILL USE THIS FOR THE REST OF YOUR PROJECTS, SO STORE SOMEWHERE SAFE. Should look like “avwmKKEE7H5xtfQcEm9O”

Creating database `portfoliordsserver`

Your database might take a few minutes to launch. The only way to view your master password is to choose [View credential details](#) during database creation. You can modify your DB instance to create a new password at any time. You can use settings from `portfoliordsserver` to simplify configuration of suggested database add-ons while we finish creating your DB for you.

[View credential details](#)

Connection details to your database `portfoliordsserver`

This is the only time you can view this password. Copy and save the password for your reference. If you lose the password, you must modify your database to change it. You can use a SQL client application or utility to connect to your database.

[Learn about connecting to your database](#)

Master username
admin

Master password

[Close](#)

Wait for your RDS to finish creating. Then click on it


[portfoliordsserver](#)

Creating

Instance

Copy the endpoint somewhere (notepad) Should look like
“portfoliordsserver.cjgoi4kygis9.us-east-2.rds.amazonaws.com”

Endpoint

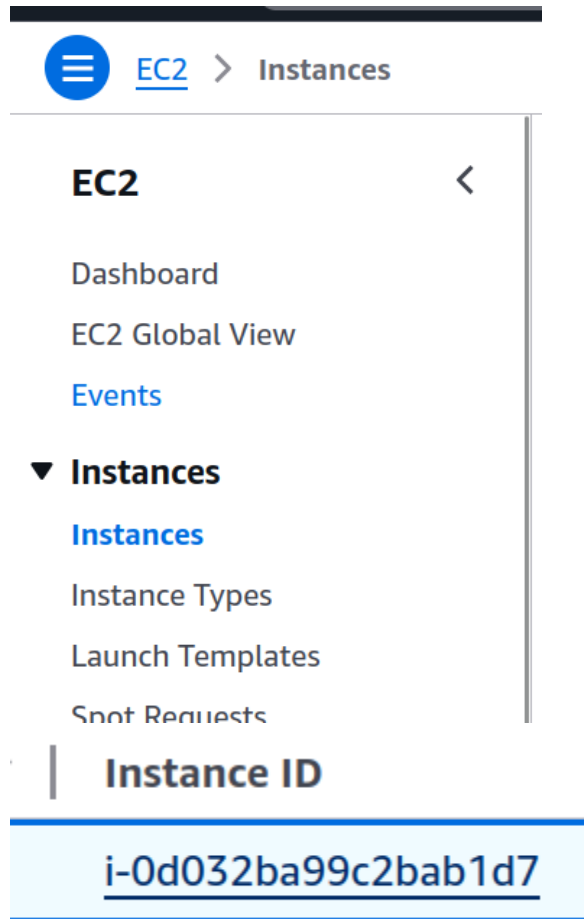
 portfoliordsserver.cjgoi4kygis9.us-east-2.rds.amazonaws.com

Port

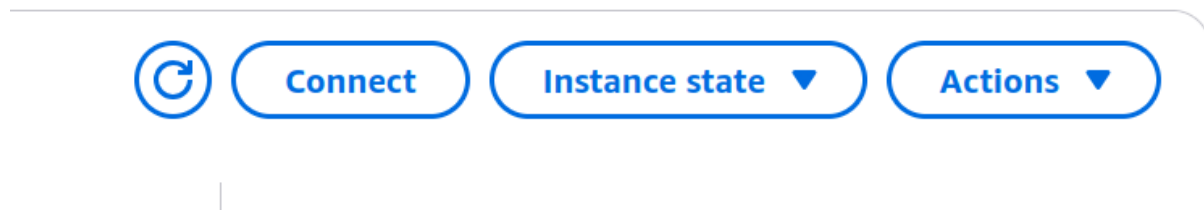
3306

4) Connect to your EC2

Find your EC2 on the dashboard by going to instances. Click on the instance id that shows up.



Click on Connect



Connect to your instance i-Oee04a7687601b4fa (PortfolioEC2) using any of these options

EC2 Instance Connect

Session Manager

SSH client

EC2 serial console

Instance ID

i-Oee04a7687601b4fa (PortfolioEC2)

Connection Type

☒ Connect using EC2 Instance Connect

Connect using the EC2 Instance Connect browser-based client, with a public IPv4 or IPv6 address.

☐ Connect using EC2 Instance Connect Endpoint

Connect using the EC2 Instance Connect browser-based client, with a private IPv4 address and a VPC endpoint.

Public IPv4 address

☒ 3.139.72.150

☐ IPv6 address

-

Username

Enter the username defined in the AMI used to launch the instance. If you didn't define a custom username, use the default username, ubuntu.

Note: In most cases, the default username, ubuntu, is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI username.

Cancel

Connect

You should see this

```
Welcome to Ubuntu 24.04.1 LTS (GNU/Linux 6.8.0-1021-aws x86_64)

* Documentation:  https://help.ubuntu.com
* Management:    https://landscape.canonical.com
* Support:        https://ubuntu.com/pro

System information as of Thu Mar  6 05:39:56 UTC 2025

System load:  0.16           Processes:            106
Usage of /:   24.9% of 6.71GB Users logged in:          0
Memory usage: 22%           IPv4 address for enX0: 10.0.12.100
Swap usage:   0%

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

The list of available updates is more than a week old.
To check for new updates run: sudo apt update

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-10-0-12-100:~$
```

Enter these commands (One at a time please):

Here we install updates to linux system. Then we install necessary python packages for the server. USE CTRL+SHIFT+V to paste in a terminal rather than CTRL+V

```
sudo apt update
```

```
sudo apt upgrade

sudo apt install python3-dotenv
sudo apt install python3-flask
sudo apt install python3-pymysql
```

Getting the code for our server

```
git clone https://github.com/K12-NYU-Center/CPath.git
cd CPath
cd "Part 2"
```

Making an .env file. This is to store important credentials

```
nano .env
```

paste this in. Remove the bracketed part and put your saved endpoint and password earlier:

```
DB_HOST={rds endpoint}
DB_USER=admin
DB_PASSWORD={password saved earlier}
DB_NAME=forms
DB_PORT=3306
```

Ctrl-x then enter to save

Test the server

Type in the command:

```
python3 rds.py
```

If it works with no errors, press Ctrl+c to quit it

5) Connect to your RDS MYSQL Server

More bash commands.

Install your mysql command line client. This let's us connect to the RDS server

```
sudo apt install mysql-client
```

Now connect to the server. Use your saved endpoint earlier

```
mysql -h {endpoint} -u admin -p
```

It should prompt you for your password. Copy paste it in.

You should see this

```
mysql>
```

MySQL Commands:

Enter these into the mysql server

```
create database forms;
use forms;
CREATE TABLE form_submissions (
  id INT AUTO_INCREMENT PRIMARY KEY,
  name VARCHAR(255),
  subject VARCHAR(255),
  email VARCHAR(255),
  message TEXT,
  created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);
```

Now type quit to quit the mysql server

Setting up access from the web

Try connecting your EC2 Instance from web

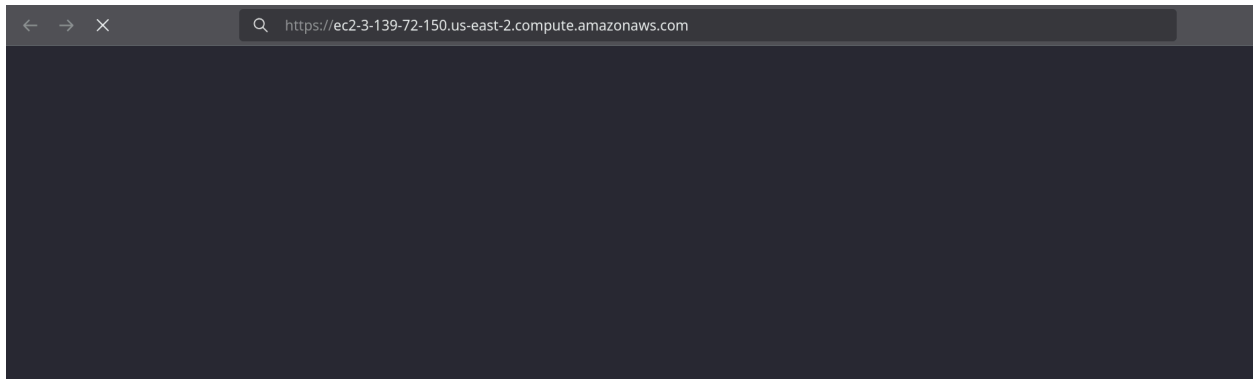
Try connecting. Remember to add port :5000 to end of it, because we are using port :5000 for the server. Example: <https://ec2-3-139-82-150.us-east-2.compute.amazonaws.com:5000>

Public IPv4 DNS

 ec2-3-139-72-150.us-east-2.compute.amazonaws.com

| [open address](#) 

It doesn't work (ignore that the picture doesn't have port 5000)




Editing the security group

We need to allow access from the web for our EC2 with security groups. It's not enough to just have it in a public subnet.

Click on your subnet from earlier (EC2 Inbound)


| <input type="checkbox"/> | Name | Security group ID | Security group name | VPC ID |
|--------------------------|------|--------------------------------------|----------------------------------|---------------------------|
| <input type="checkbox"/> | - | sg-0705d8768a17e0bc5 | rds-ec2-2 | vpc-0e230 |
| <input type="checkbox"/> | - | sg-083aec5b3e1066806 | rds-ec2-1 | vpc-0e230 |
| <input type="checkbox"/> | - | sg-009d5f362c01f1791 | default | vpc-0e230 |
| <input type="checkbox"/> | - | sg-03957583aa4e2c437 | ec2-rds-1 | vpc-0e230 |
| <input type="checkbox"/> | - | sg-0679625a044c41849 | Application Load Balancer to EC2 | vpc-0e230 |
| <input type="checkbox"/> | - | sg-0c6e8dca006960dff | EC2 Inbound | vpc-0e230 |
| <input type="checkbox"/> | - | sg-0ca25c20595f8ab15 | ec2-rds-2 | vpc-0e230 |

Edit the Inbound rules



Manage tags

Edit inbound rules

< 1 > 

| ▼ | Protocol | ▼ | Port range |
|---|----------|---|------------|
| | TCP | | 22 |

Add this rule

-

Custom TCP ▼

TCP

5000

Anyw... ▼

0.0.0.0/0 X


Q

Connecting to server from web

Delete

0.0.0.0/0 X

Add rule

 Rules with source of 0.0.0.0/0 or ::/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only. X

Cancel Preview changes Save rules

Putting it all together

Make sure your server is running

Go to your EC2 and start up the server again.

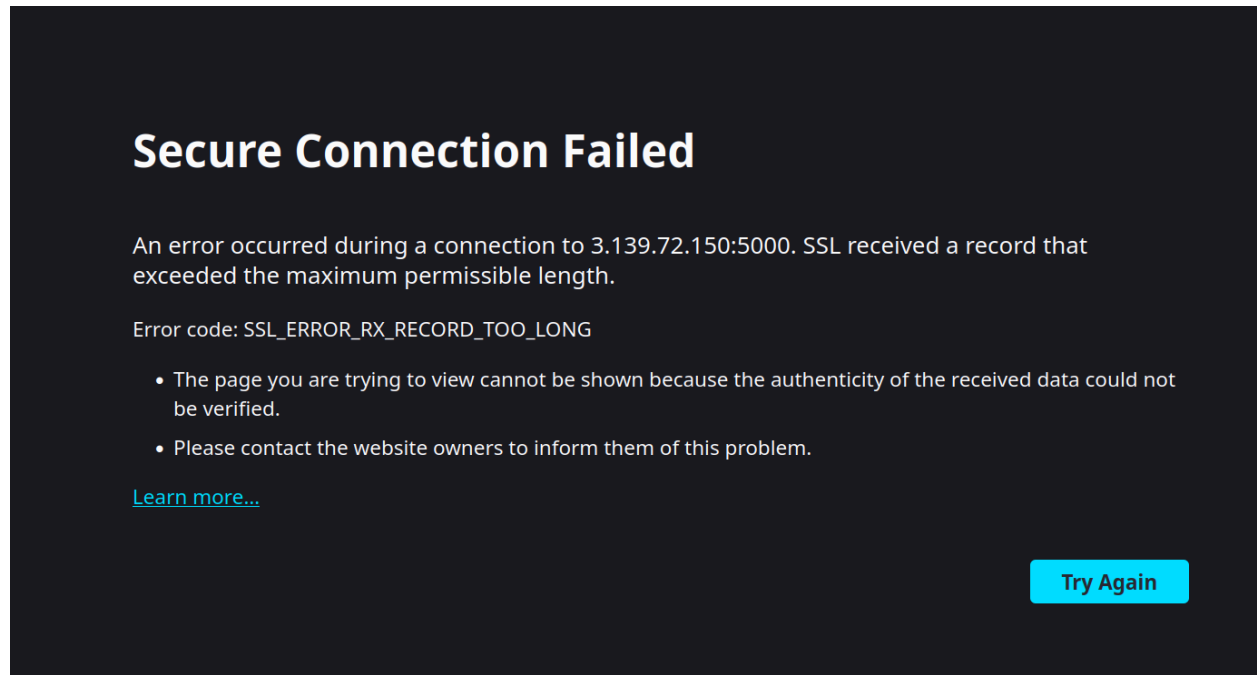
If you closed the tab, here's the command To navigate to the folder (condensed from earlier section):

```
cd /home/ubuntu/CPath/"Part 2"
```

```
^Cubuntu@ip-10-0-12-100:~/CPath/Part 2$ python3 rds.py
* Serving Flask app 'rds'
* Debug mode: off
INFO:werkzeug:WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
* Running on all addresses (0.0.0.0)
* Running on http://127.0.0.1:5000
* Running on http://10.0.12.100:5000
INFO:werkzeug:Press CTRL+C to quit
```

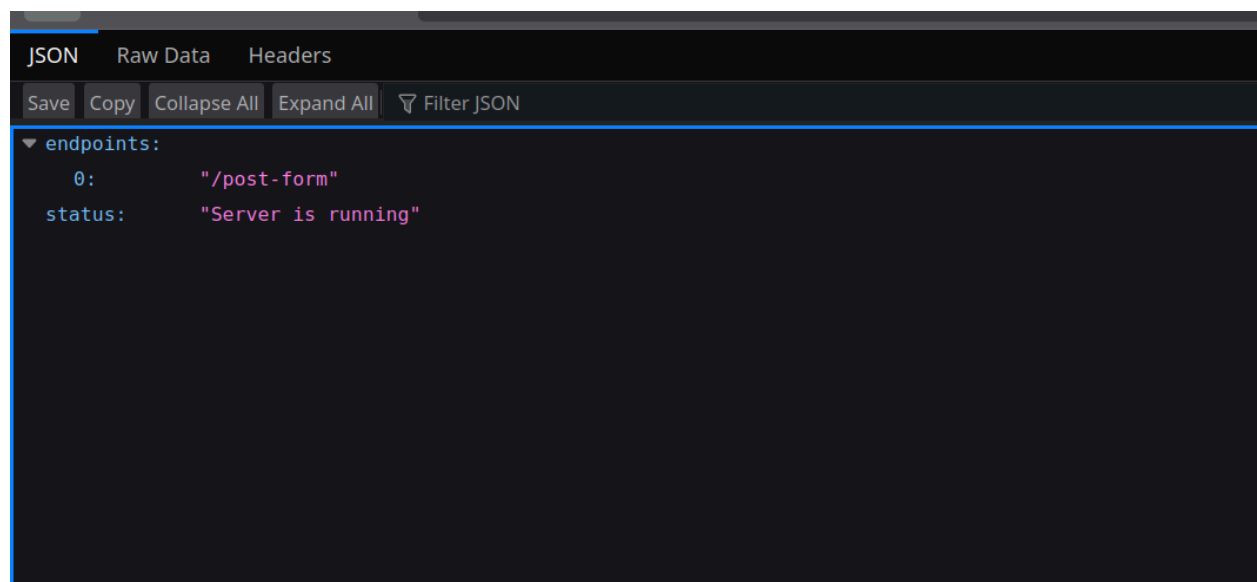
Now connect again

If you see this



Go to the url and replace https with http. We don't have SSL yet.

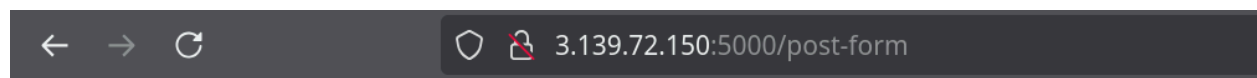
Success!



If you see something like this in your logs. Don't worry. It's just when you try to initiate https connection with http server

```
ERROR:werkzeug:71.247.202.11 - - [06/Mar/2025 07:42:45] code 400, message Bad request version ('\x10@\x16)\x98\x9ez\x16\x88ya=I0'8E\x9c\t0s\x8e\x8c@\x9e|v\x92\x12;\x00"\x13\x01\x13\x03\x13\x02A+Ã/I0I"Ã,Ã0Ã')
INFO:werkzeug:71.247.202.11 - - [06/Mar/2025 07:42:45] "\x16\x03\x01\x07\x88\x01\x00\x07\x84\x03\x03t0+g\x1b\x060\x9crrn0\x0c0Ã\x85\x8e0aw\x80\x7f\x1b0p_'p000\x80) \x10e\x16)\x98\x9ez\x16\x88ya=I0'8E\x9c\t0s\x8e\x8c@\x9e|v\x92\x12;\x00"\x13\x01\x13\x03\x13\x02A+Ã/I0I"Ã,Ã0Ã" 400 -
ERROR:werkzeug:71.247.202.11 - - [06/Mar/2025 07:42:45] code 400, message Bad request version ('h50p"\x00"\x13\x01\x13\x03\x13\x02A+Ã/I0I"Ã,Ã0Ã')
INFO:werkzeug:71.247.202.11 - - [06/Mar/2025 07:42:45] "\x16\x03\x01\x02.\x01\x00\x02'\x03\x03\x06\x98i\x88i\x9f\x8e\x1a00({\x137iyi\x90K4\x0f0c+0_[\x13L4i0 'pae*v\x04\x1e+0EX\x8f(\x0A5\x9e0p() \x88AV\x1ch00p"\x00"\x13\x01\x13\x03\x13\x02A+Ã/I0I"Ã,Ã0Ã" 400 -
```

Try going to /post-form



Method Not Allowed

The method is not allowed for the requested URL.

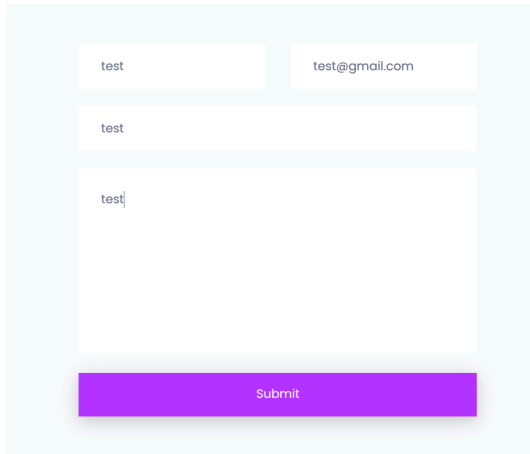
Not allowed, because we are making a GET request from url. We need to use it on our website instead.

Send a form from our website

Add the attached form code to your website somewhere. Replace the action with the post-form url.

Example: <http://ec2-3-139-72-150.us-east-2.compute.amazonaws.com:5000/post-form>

Send the form (Remember not to send anything sensitive. HTTP is unencrypted)



test test@gmail.com

test

test

Submit

(If you're using the brownie website, click on the submit letters. The button is slightly broken)

You should see something like this

```
← → ↻ ⚠ Not secure ec2-3-139-72-150.us-east-2.compute.amazonaws.com:5000/post-form
Pretty-print
{"data":{"email":"test@gmail.com","message":"test","name":"test","subject":"test"},"message":"Data stored successfully","status":"success"}
```

Let's see our data in the database.

Press ctrl+C on your EC2 to exit the flask server. Connect to your RDS again

```
* Running on http://10.0.12.100:5000
INFO:werkzeug:Press CTRL+C to quit
^Cubuntu@ip-10-0-12-100:~/CPath/Part 2mysql -h portfolioordsserver.cjgoi4kygis9.us-east-2.rds.amazonaws.com -u admin -p
```

Now lets see our table

```
Use forms;  
select * from form_submissions;
```

```
mysql> use forms;  
Database changed  
mysql> select * from form_submissions;  
+----+-----+-----+-----+-----+-----+  
| id | name | subject | email          | message | created_at          |  
+----+-----+-----+-----+-----+-----+  
|  1 | test | test    | test@gmail.com | test    | 2025-03-06 07:45:38 |  
+----+-----+-----+-----+-----+-----+  
1 row in set (0.00 sec)  
  
mysql> 
```

Success!