# Placement Empowerment Program

Poc 14: Deploy a Web Application on the CloudWrite a Python Flask application and deploy it on your cloud VM.

Name: Subiksha J Department: Ads

# 

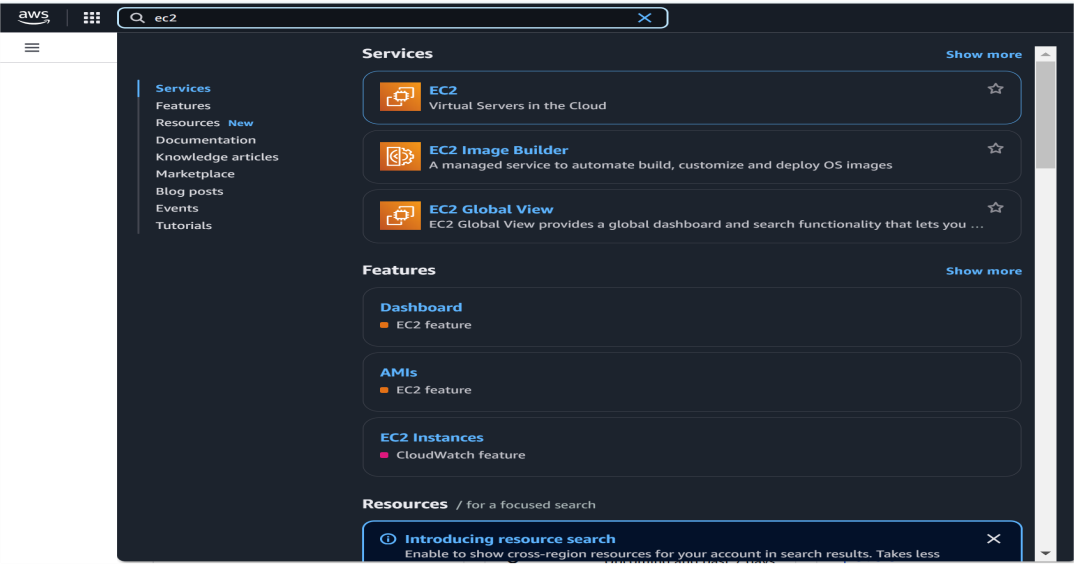
# Step-by-Step Overview

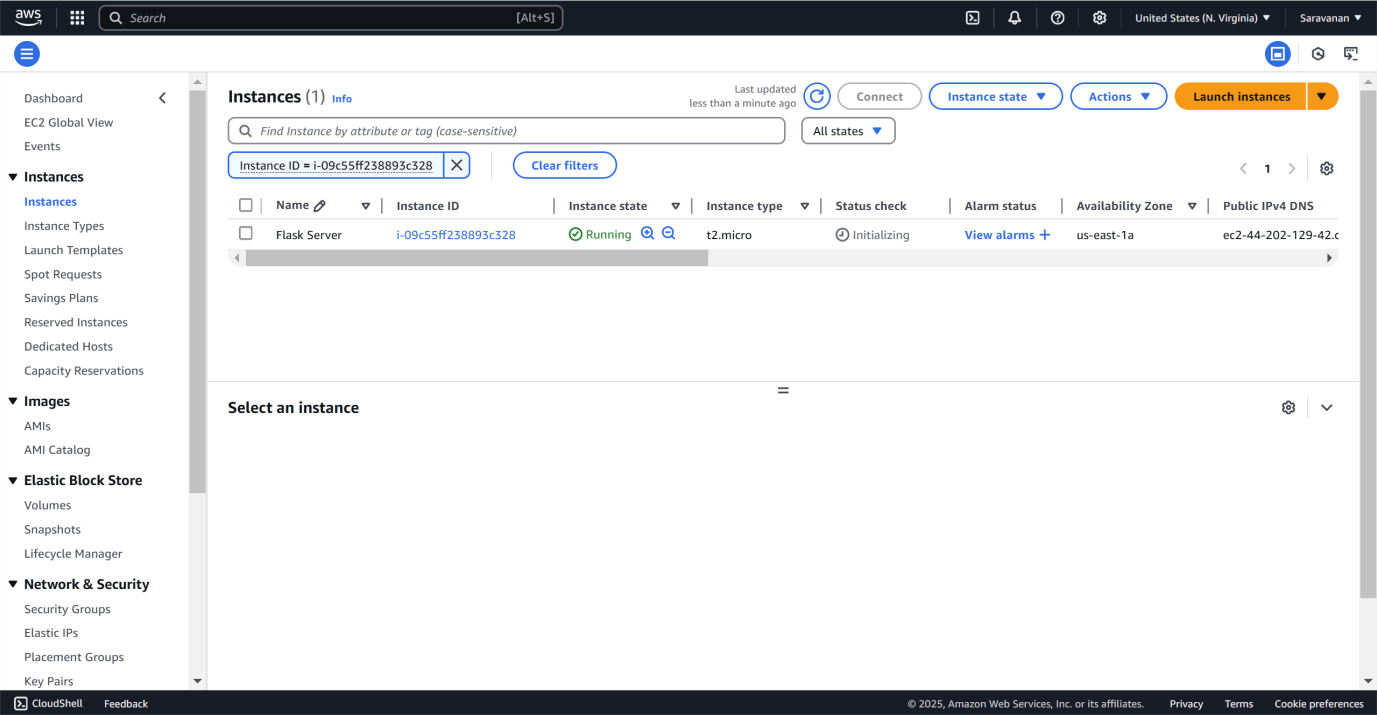
## Step 1:

* 1. Go to [AWS Management Console](https://aws.amazon.com/console/).
  2. Enter your username and password to log in.

## Step 2:

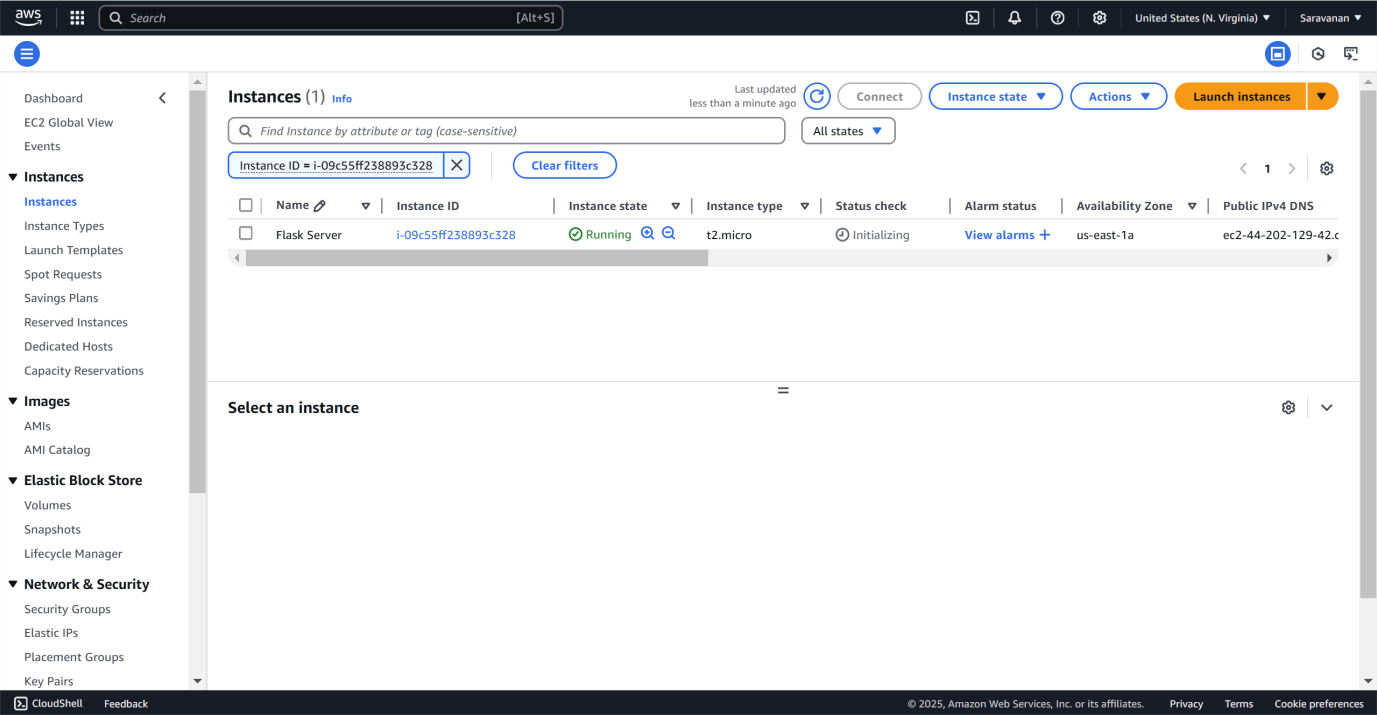
On the EC2 Dashboard, click on **Launch Instances** and enter a name for your instance (e.g., "Flask Server") and select Ubuntu as OS and create a key pair. Leave other settings as default and Click **Launch Instance**.





## Step 3:

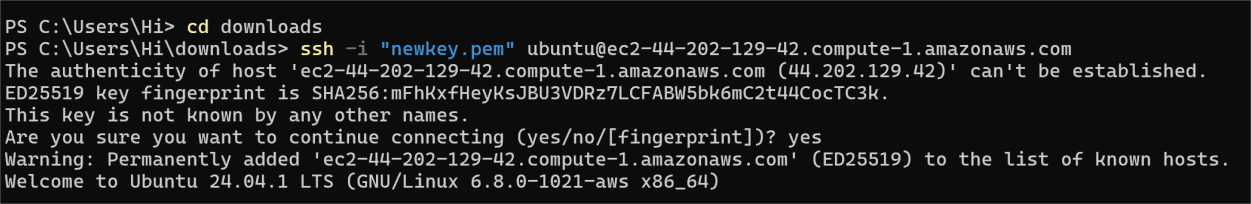
Click the 'Connect' option on your launched instance, go to the SSH client section, and copy the command provided under the 'Example' section.



## Step 4:

Open PowerShell, navigate to the 'Downloads' directory where the downloaded key pair is located using the **cd Downloads** command

Paste the command copied from the EC2 Connect's SSH client section, replace the key pair name with your downloaded key (e.g., new.pem), press Enter, and type 'yes' when prompted.



## Step 5:

Update the Package List :

Screenshot 2025-02-01 123237.png

## Step 6:

Install Python3 and pip

Screenshot 2025-02-01 123256.png

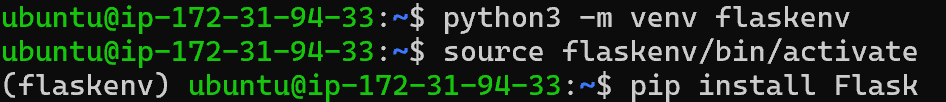
## Step 7:

Install Virtual Environment Tools : This helps keep your app’s dependencies separate.

Screenshot 2025-02-01 123435.png

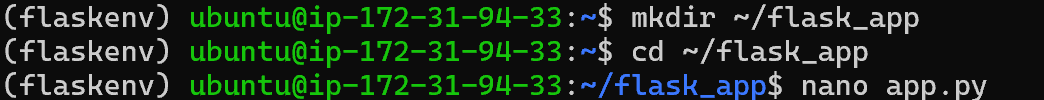
## Step 8:

Create and Activate a Virtual Environment and install Flask



## Step 9:

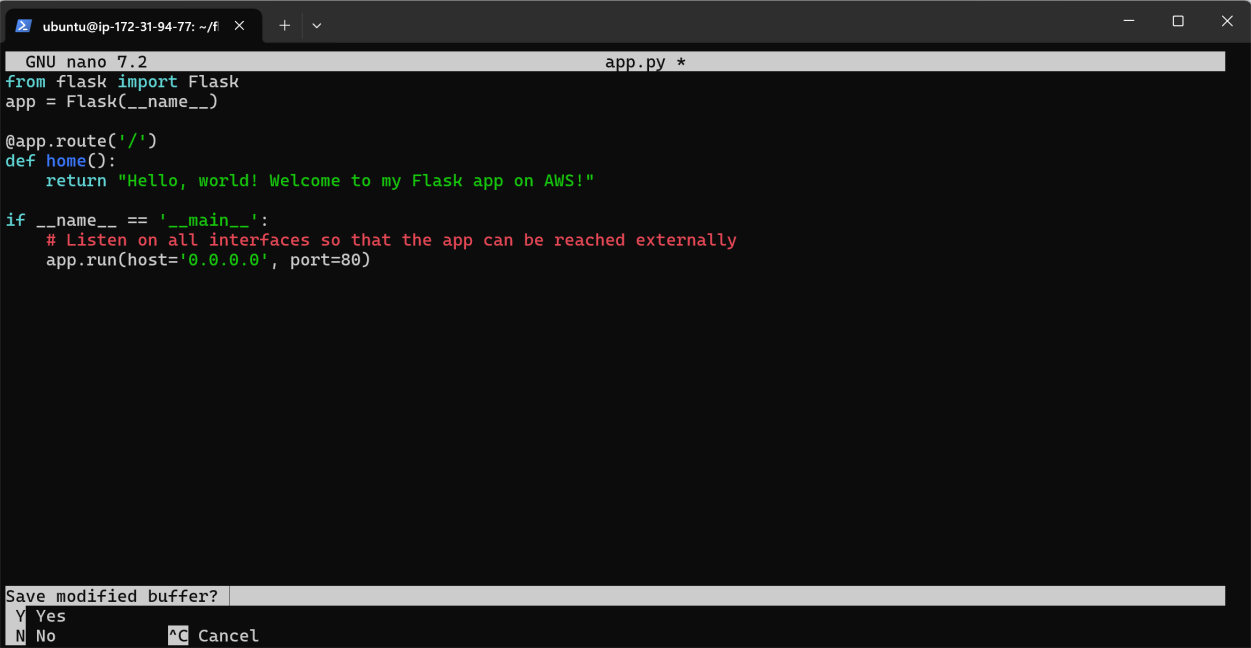
Create a Directory for Your App and Create a file called app.py using a text editor (like nano).



## Step 10:

Write this code into the editor and press **Ctrl + O** (to write out) and then

**Enter**, then **Ctrl + X** to exit.



## Step 11:

Exit the virtual environment:

Screenshot 2025-02-01 123540.png

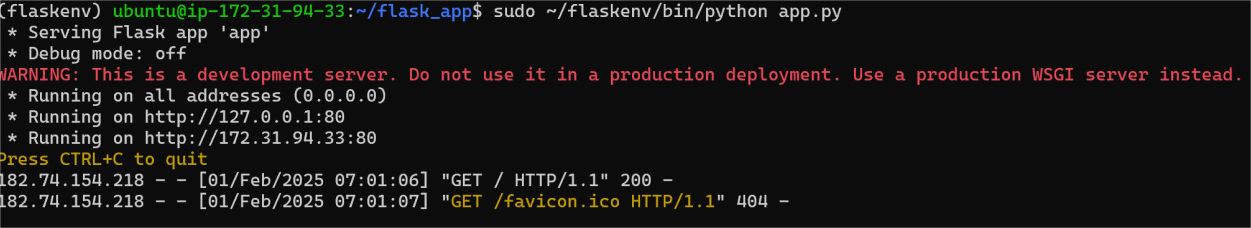
## Step 12:

Add your virtual environment’s Python path to the sudo command and Run the application using the virtual environment's Python:

Screenshot 2025-02-01 123607.png

## Step 13:

Your Flask app is now running!



## Step 14:

Go to the **EC2 Dashboard** > **Instances**.

Find your instance and note the **Security Group** attached to it. Navigate to **Security Groups** under the **Network & Security**

section.

Select the Security Group associated with your EC2 instance.

Under the **Inbound Rules** tab, ensure there is a rule for **HTTP (port 80)**:

**Type:** HTTP **Protocol:** TCP **Port Range:** 80

**Source:** Anywhere (0.0.0.0/0, ::/0)

If there isn't an HTTP rule, click **Edit inbound rules** and add it.

## Step 15:

Open your browser and navigate to:

http://<Your-Instance-Public-IP>/

Replace <Your-Instance-Public-IP> with the Public IPv4 address of your EC2 instance (e.g., <http://54.123.45.67/>).

Public IPv4 address can be found in your Ec2 instance dashboard.

