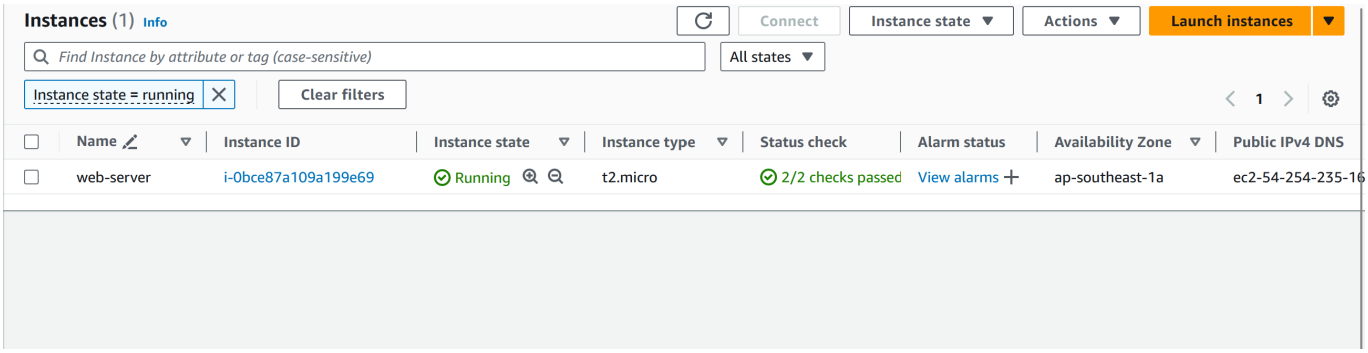
Class 2 report

Step 1: Deploy a EC2 instance allowing SSH access on AWS using terraform.

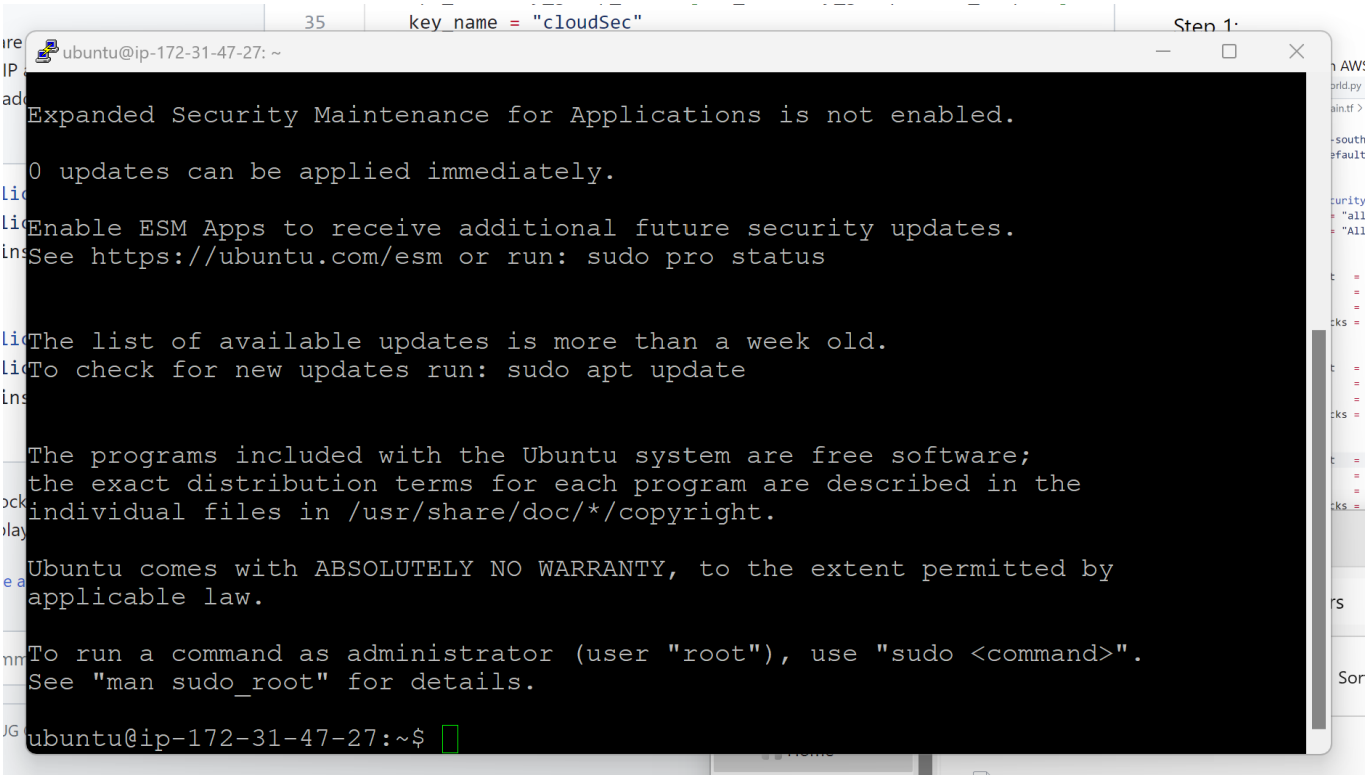
Welcome | helloWorld.py | main.tf | readme.md

lab1 > terraformSetup > main.tf > resource "aws_security_group" "allow_http" > egress > from_port

```
provider "aws" {
    region = "ap-southeast-1"  // Replace with your AWS region
    profile = "default"
}

resource "aws_security_group" "allow_http" {
    name        = "allow_http"
    description = "Allow HTTP inbound traffic"

    ingress {
        from_port   = 80
        to_port     = 80
        protocol    = "tcp"
        cidr_blocks = ["0.0.0.0/0"]
    }
    ingress {
        from_port   = 22
        to_port     = 22
        protocol    = "tcp"
        cidr_blocks = ["0.0.0.0/0"]
    }
    egress {
        from_port   = 0
        to_port     = 0
        protocol    = "-1"
        cidr_blocks = ["0.0.0.0/0"]
    }
}

resource "aws_instance" "web" {
```

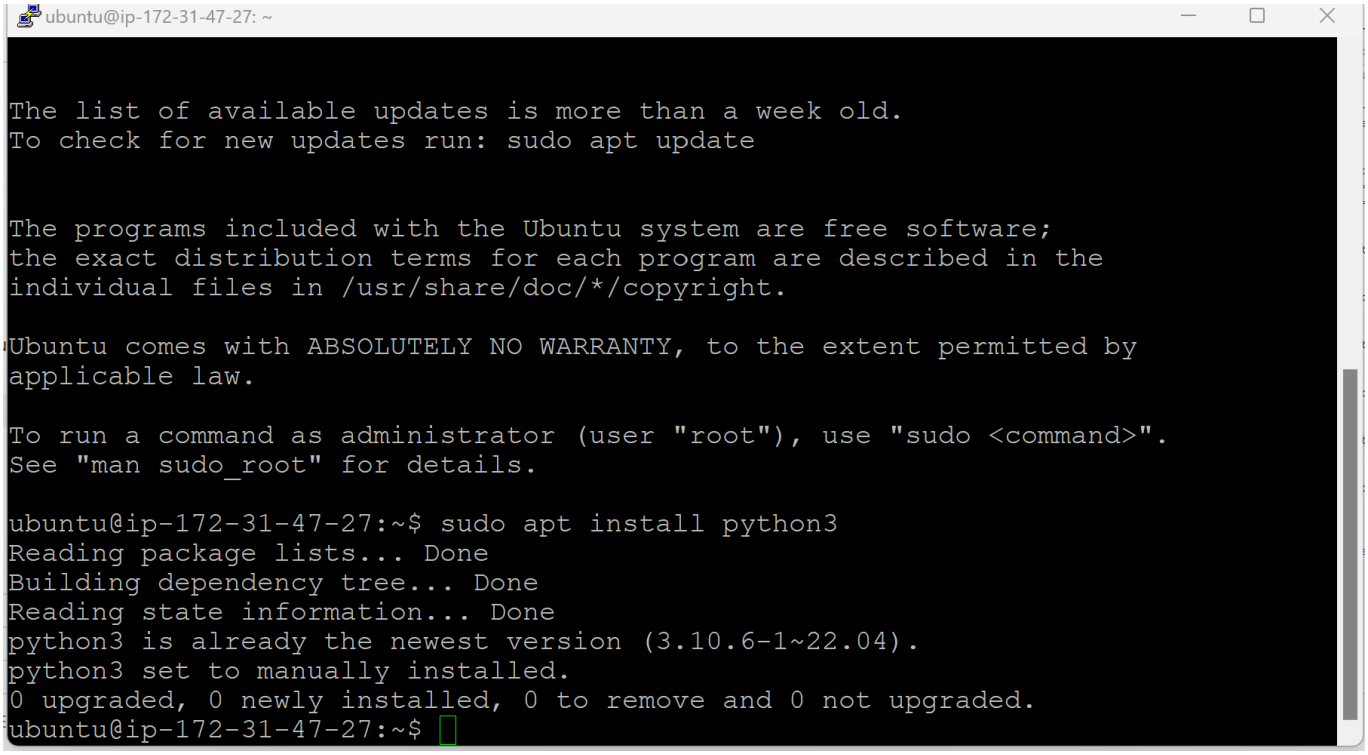


Step 2: Connect to instance using putty.



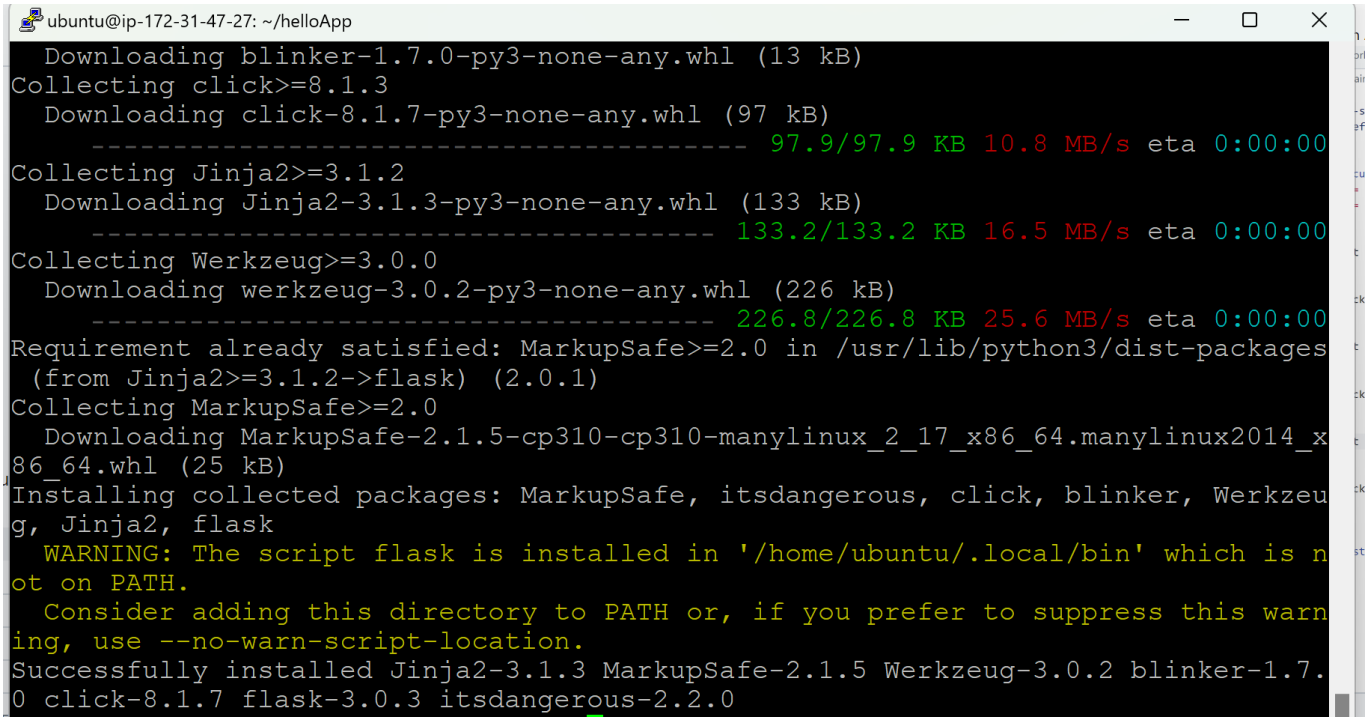
Step 3: Install Python Setup the environment.

```
sudo apt install python 3
```

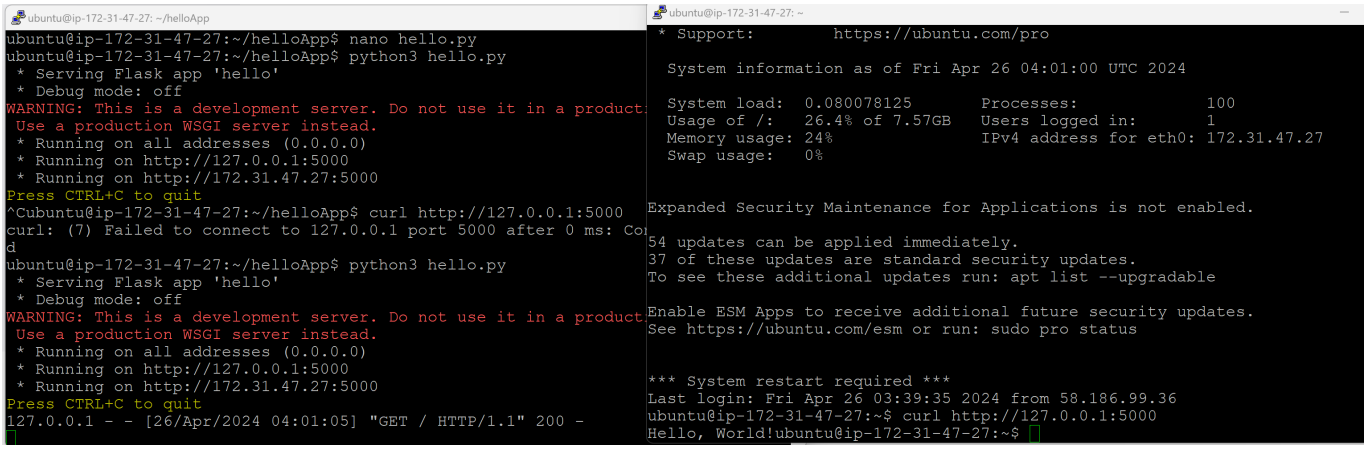


Step 4: Install Flask.

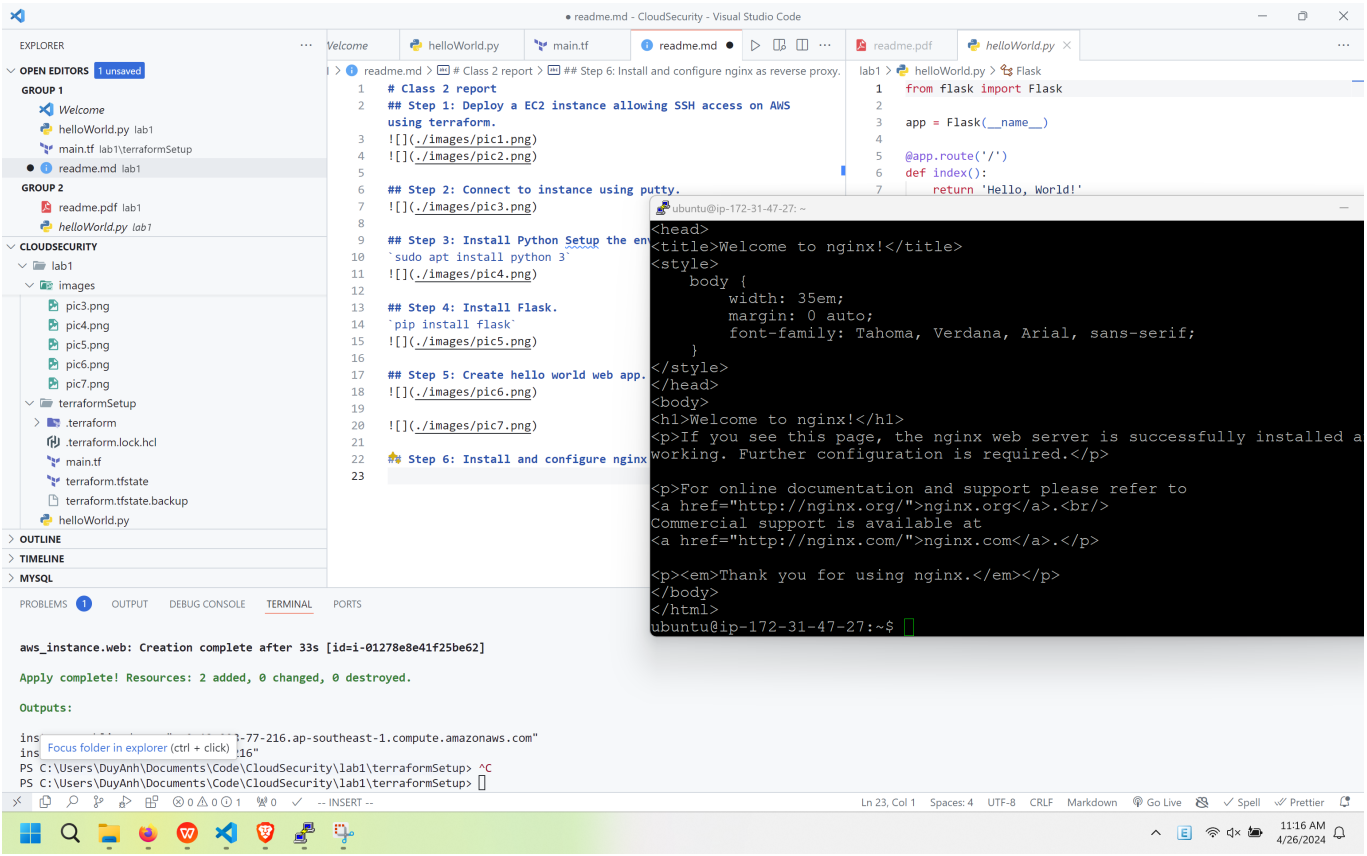
```
pip install flask
```



Step 5: Create hello world web app.



Step 6: Install and configure nginx as reverse proxy.



Create a file with the name of the EC2 instance's IP address:

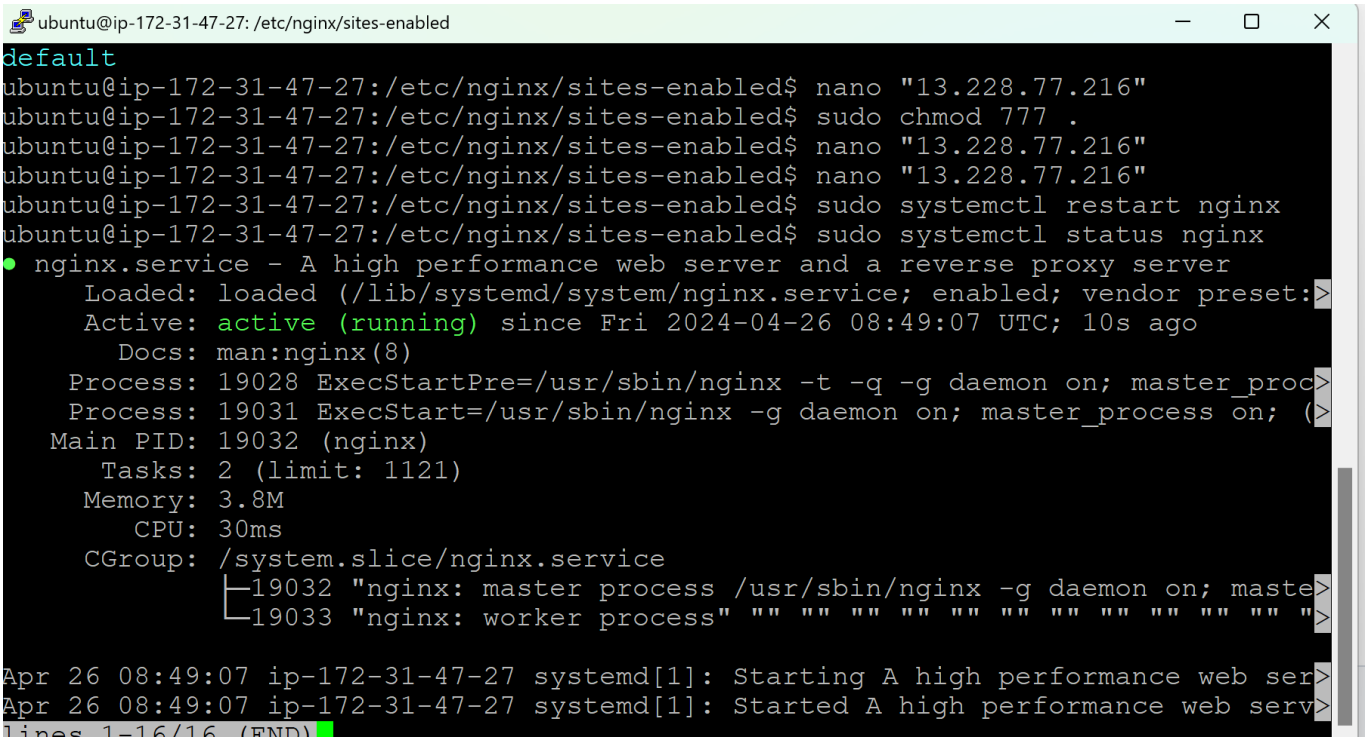
```
cd /etc/nginx/sites-enabled
nano "13.228.77.216"
```

The content inside the file:

```
server {
    listen 80;
    listen [::]:80;
    server_name 13.228.77.216;

    location / {
        proxy_pass http://127.0.0.1:5000;
        include proxy_params;
    }
}
```

Restart Nginx server and check the status of server.

```
sudo systemctl restart nginx
sudo systemctl status nginx
```



Step 7: Install Python virtual environment and gnuicorn.

```
sudo apt-get install python3-venv -y
pip install wheel
```

```
pip install gunicorn
```

Step 9: Create WSGI Entry Point for Gunicorn.

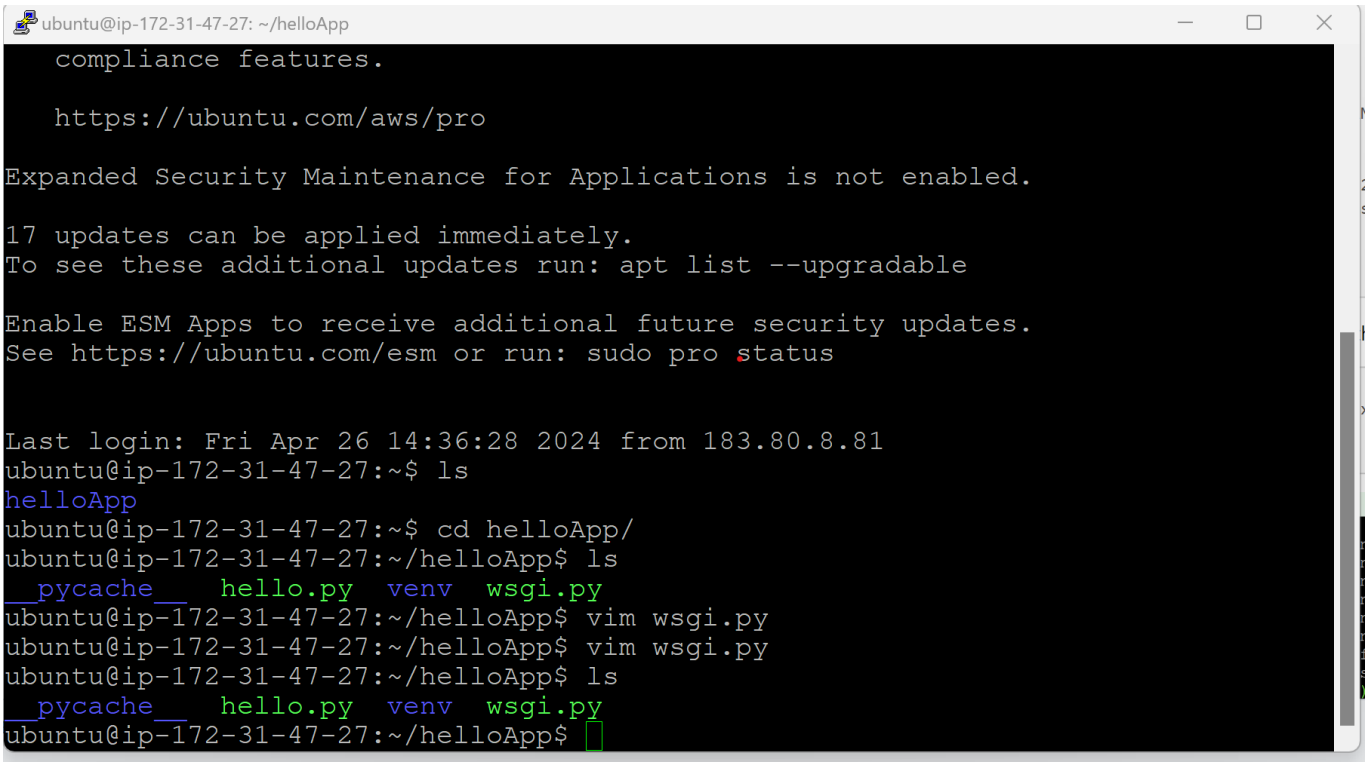
`~/helloApp$ vim wsgi.py`

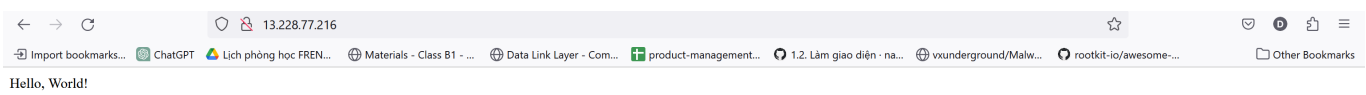
```
from hello import app
if __name__ == "__main__":
    app.run()
```

Activate virtual environment and create entry point for the app:

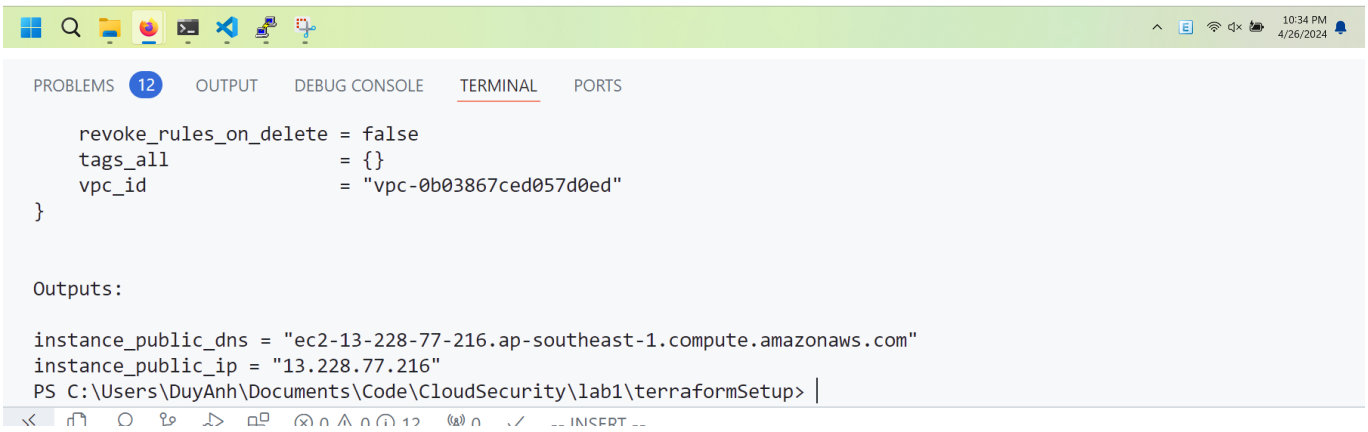
`source venv/bin/activate ~/helloApp$ gunicorn --bind 0.0.0.0:5000 wsgi:app`

Step 10: Check the result:


ubuntu@ip-172-31-47-27: ~/helloApp
compliance features.
https://ubuntu.com/aws/pro
Expanded Security Maintenance for Applications is not enabled.
17 updates can be applied immediately.
To see these additional updates run: apt list --upgradable
Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status
Last login: Fri Apr 26 14:36:28 2024 from 183.80.8.81
ubuntu@ip-172-31-47-27:~$ ls
helloApp
ubuntu@ip-172-31-47-27:~$ cd helloApp/
ubuntu@ip-172-31-47-27:~/helloApp$ ls
__pycache__ hello.py venv wsgi.py
ubuntu@ip-172-31-47-27:~/helloApp$ vim wsgi.py
ubuntu@ip-172-31-47-27:~/helloApp$ vim wsgi.py
ubuntu@ip-172-31-47-27:~/helloApp$ ls
__pycache__ hello.py venv wsgi.py
ubuntu@ip-172-31-47-27:~/helloApp$


13.228.77.216
Import bookmarks...
ChatGPT
Lịch phòng học FREN...
Materials - Class B1 - ...
Data Link Layer - Com...
product-management...
1.2. Làm giao diện - na...
vxunderground/Malw...
rootkit-io/awesome-...
Other Bookmarks

Hello, World!


10:34 PM
4/26/2024
PROBLEMS 12
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
revoke_rules_on_delete = false
tags_all = {}
vpc_id = "vpc-0b03867ced057d0ed"
}
Outputs:
instance_public_dns = "ec2-13-228-77-216.ap-southeast-1.compute.amazonaws.com"
instance_public_ip = "13.228.77.216"
PS C:\Users\DuyAnh\Documents\Code\CloudSecurity\lab1\terraformSetup>