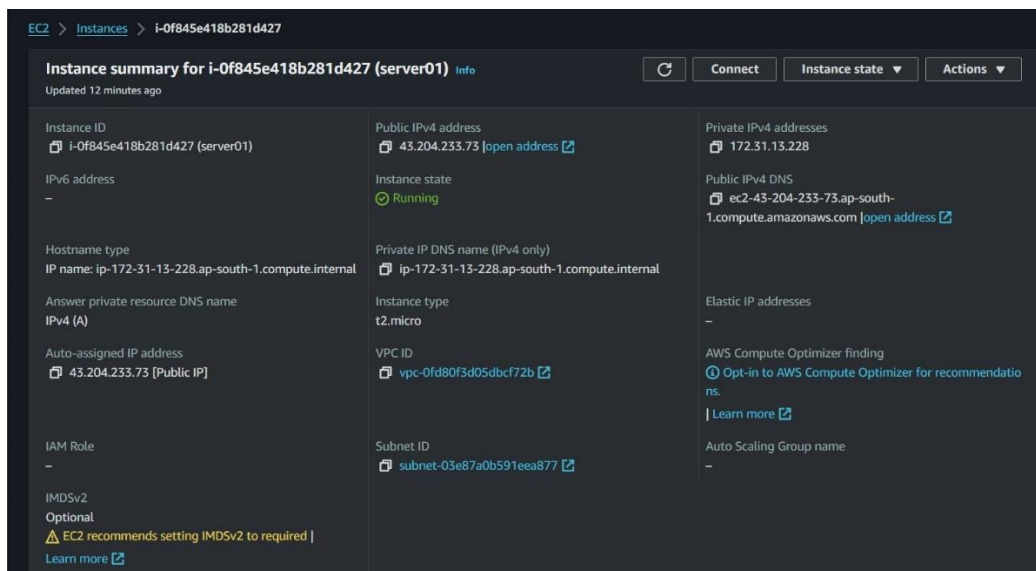


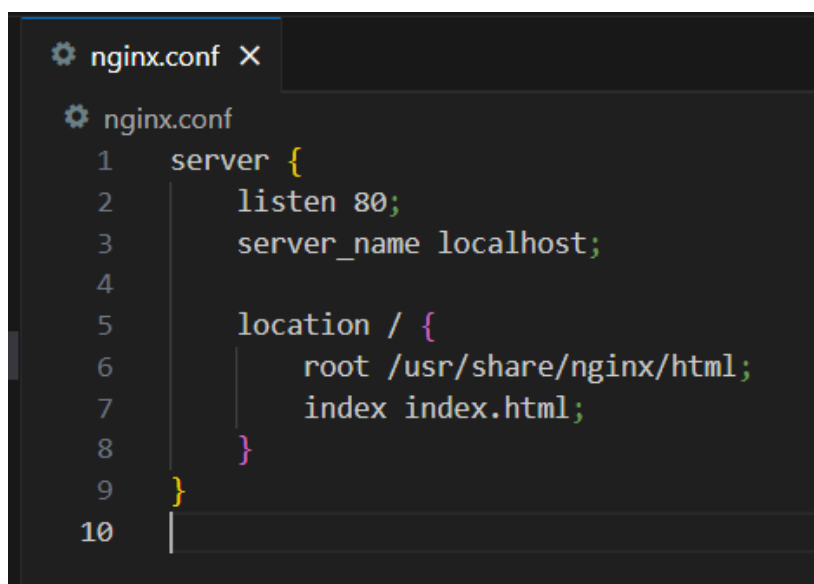
Objective: In this assignment, you will create a simple GitHub Actions workflow that builds a custom Docker image with a basic NGINX configuration, deploys it to an Amazon EC2 instance, and ensures that the container remains running even after an EC2 instance restart.

For the assignment, I am using this EC2 instance:



Steps:

1. Create a file named "nginx.conf"



2. Create a file index.html

```
nginx.conf  index.html X
index.html > html > body > h1
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4    <meta charset="UTF-8">
5    <meta name="viewport" content="width=device-width, initial-scale=1.0">
6    <title>Document</title>
7  </head>
8  <body>
9    <h1>HELLO, I AM ADITYA</h1>
10 </body>
11 </html>
```

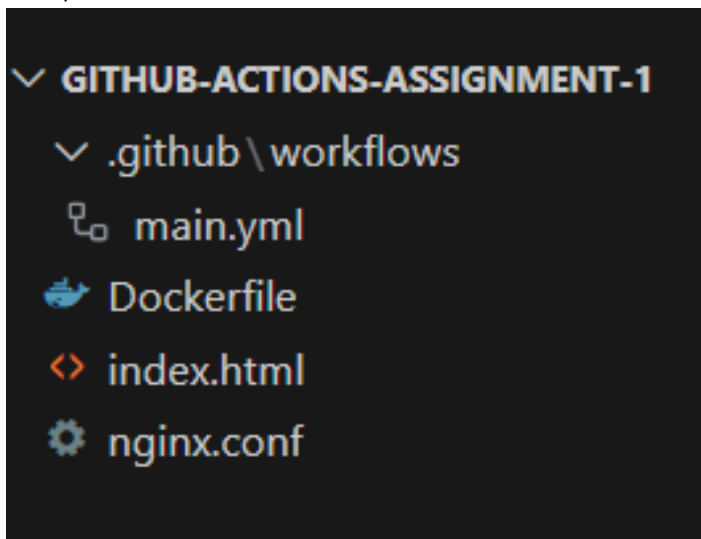
3. Then, a Dockerfile

```
nginx.conf  Dockerfile X
Dockerfile > ...
1  FROM nginx:latest
2  COPY nginx.conf /etc/nginx/conf.d/default.conf
3  COPY ./index.html /usr/share/nginx/html
4
```

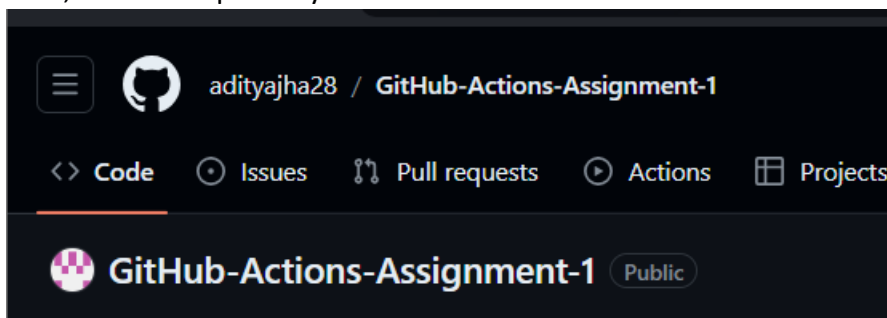
4. Create a folder .github , inside it create a folder workflows and in it create a file main.yml

```
main.yml X
.github > workflows > main.yml
1  name: Docker Image CI
2
3  on:
4    push:
5      branches: [ "dev" ]
6    pull_request:
7      branches: [ "dev" ]
8
9  jobs:
10   build:
11     runs-on: ubuntu-latest
12
13     steps:
14       - name: Github checkout
15         uses: actions/checkout@v3
16       - name: docker login
17         uses: docker/login-action@v3
18         with:
19           username: ${ secrets.DOCKER_USERNAME }
20           password: ${ secrets.DOCKER_PASSWORD }
21
22       - name: Docker build
23         run: docker build -t ${ secrets.DOCKER_USERNAME }/mynginx:latest .
24       - name: Docker push
25         run: docker push ${ secrets.DOCKER_USERNAME }/mynginx:latest
26
27       - name: SSH into EC2 Instance and Pull Docker Image
28         uses: appleboy/ssh-action@master
29         with:
30           host: ${ secrets.EC2_INSTANCE_IP }
31           username: ec2-user
32           key: ${ secrets.EC2_SSH_KEY }
33           script: |
34             sudo docker stop nginx-container
35             sudo docker rm nginx-container
36             sudo docker run -d -p 80:80 --restart always --name nginx-container ${ secrets.DOCKER_USERNAME }/mynginx:latest
```

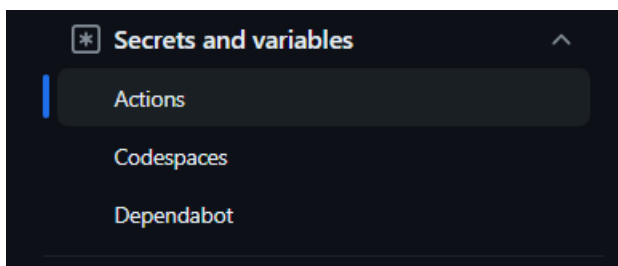
Now, the folder structure will look like:



5. Now, create a repository



6. Add secrets in the repository according to the main.yml file requirement
In order to add secrets go to settings then, secrets and variables after that click on "Actions".



Repository secrets

New repository secret

Name	Last updated
DOCKER_PASSWORD	13 hours ago
DOCKER_USERNAME	13 hours ago
EC2_INSTANCE_IP	13 hours ago
EC2_SSH_KEY	13 hours ago

- In the repository create a dev branch
Push these files to dev branch
dev branch:

adityajha28 / GitHub-Actions-Assignment-1

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

GitHub-Actions-Assignment-1 (Public) Pin Unwatch 1

dev 2 Branches 0 Tags

Go to file Add file <> Code

This branch is 9 commits ahead of, 1 commit behind main.

adityajha28 Update main.yml 776e210 · 12 hours ago 11 Commits

.github/workflows	Update main.yml	12 hours ago
Dockerfile	Final commit	2 days ago
index.html	Added index.html	2 days ago
nginx.conf	Initial commit	2 days ago

README

After pushing go to “Actions” and wait for the workflows to be completed.

- After the completion of workflows:

adityajha28 / GitHub-Actions-Assignment-1

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

Actions New workflow

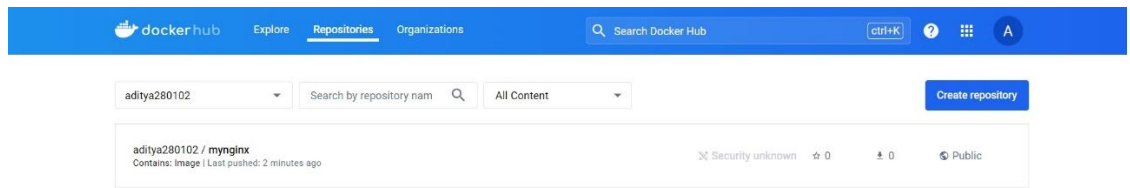
All workflows Showing runs from all workflows Filter workflow runs

Workflow run	Event	Status	Branch	Actor
Update main.yml Docker Image CI #2, Commit 776e210 pushed by adityajha28	push	Completed	dev	adityajha28

8 minutes ago 25s

Checkout the results:

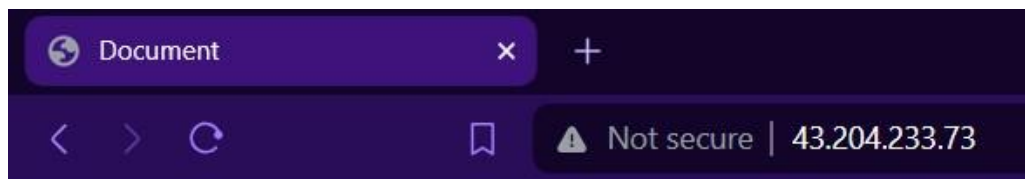
a. dockerhub:



b. EC2 instance:

```
[ec2-user@ip-172-31-13-228 ~]$ sudo docker ps
CONTAINER ID   IMAGE                  COMMAND                  CREATED        STATUS        PORTS                    NAMES
2f5b74a16313   aditya280102/mynginx:latest   "/docker-entrypoint..."  49 seconds ago   Up 48 seconds   0.0.0.0:80->80/tcp, :::80->80/tcp   nginx-container
[ec2-user@ip-172-31-13-228 ~]$
```

c. Access this public IP <http://43.204.233.73/>



HELLO, I AM ADITYA