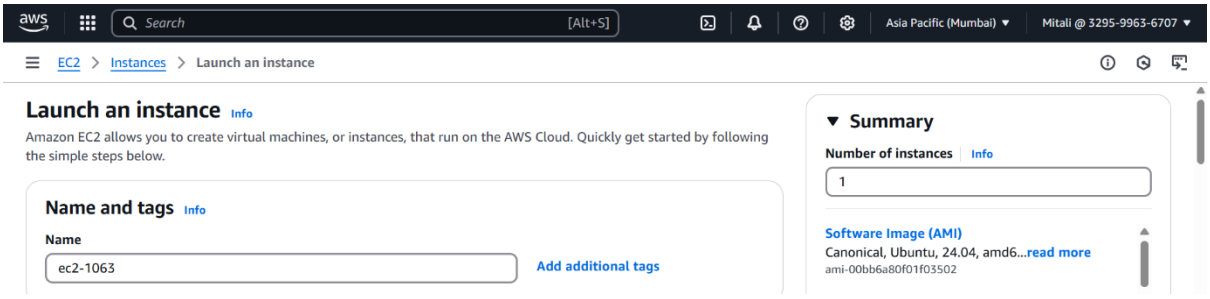
Name of the Student: **Mitali Bhattad** **PRN: 20220801063**

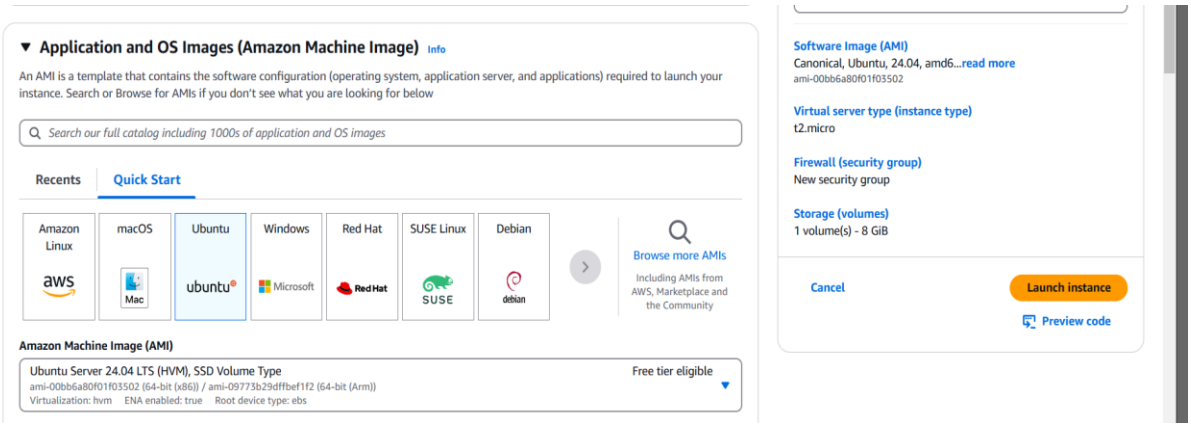
Title of Practical: **Implementing Live Changes in a Running Docker Container**

Step1: Launch an EC2 Instance

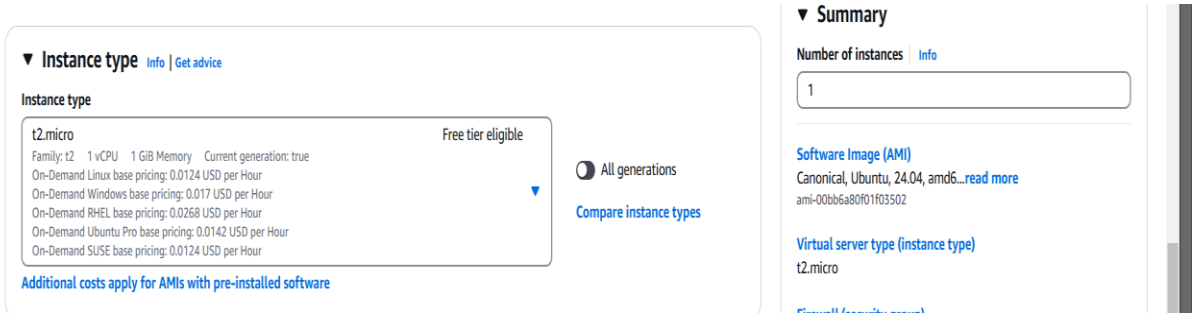
- Name the Instance



- Choose AMI: Ubuntu



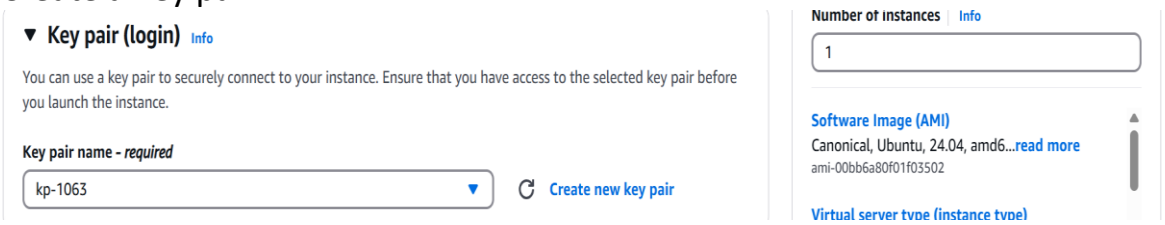
- Select the instance type: t2 micro



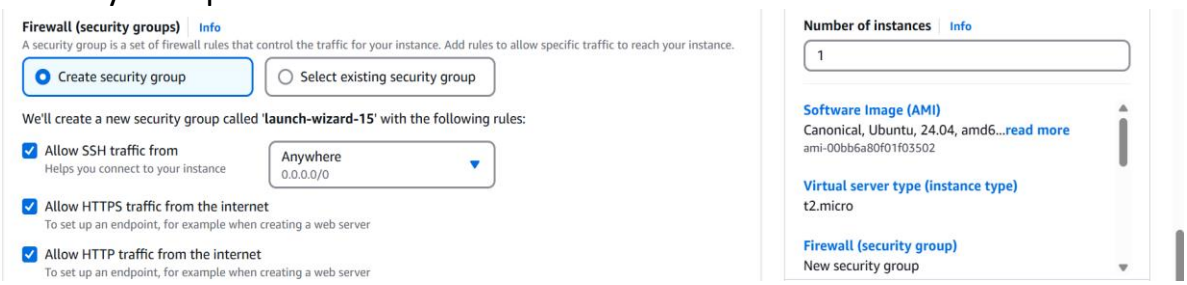
Name of the Student: Mitali Bhattad **PRN:** 20220801063

Title of Practical: Implementing Live Changes in a Running Docker Container

- Create a Key pair



- Security Group



- Launch the instance

Step 2: Connect the EC2 Instance and run the following commands:

1. Switch to root user and update and upgrade system packages
- sudo -i
- sudo apt update -y

```
ubuntu@ip-172-31-39-97:~$ sudo -i
root@ip-172-31-39-97:~# sudo apt-get update -y
```

2. Install Docker:
- sudo apt install docker.io -y

```
root@ip-172-31-39-97:~# sudo apt install docker.io -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
```

Name of the Student: Mitali Bhattad **PRN:** 20220801063

Title of Practical: Implementing Live Changes in a Running Docker Container

3. Start and enable Docker and verify the version:
- sudo systemctl start docker
- sudo systemctl enable docker
- docker –version

```
root@ip-172-31-39-97:~# sudo systemctl enable docker
root@ip-172-31-39-97:~# sudo systemctl start docker
root@ip-172-31-39-97:~# docker --version
Docker version 26.1.3, build 26.1.3-0ubuntu1~24.04.1
```

4. Create a Directory and move to that directory.
Create an index.html file in that directory and write in that file:
- mkdir ~/my-website
- cd ~/my-website
- echo "<h1>Hello from Mitali</h1>" > index.html

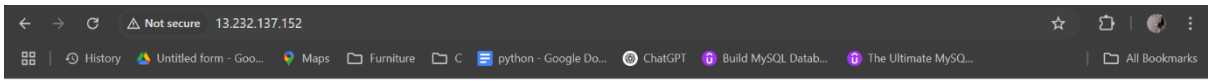
```
root@ip-172-31-39-97:~# mkdir ~/my-website
root@ip-172-31-39-97:~# cd ~/my-website
root@ip-172-31-39-97:~/my-website# echo "<h1>Hello from Mitali</h1>" > index.html
```

5. Pull and Run the Nginx Container
docker run -d \
--name my-nginx \
-p 80:80 \
-v ~/my-website:/usr/share/nginx/html \
nginx:latest

```
root@ip-172-31-39-97:~/my-website# docker run -d \
  --name my-nginx \
  -p 80:80 \
  -v ~/my-website:/usr/share/nginx/html \
  nginx:latest
Unable to find image 'nginx:latest' locally
latest: Pulling from library/nginx
7cf63256a31a: Pull complete
bf9acace214a: Pull complete
513c3649bb14: Pull complete
d014f92d532d: Pull complete
9dd21ad5a4a6: Pull complete
943ea0f0c2e4: Pull complete
103f50cb3e9f: Pull complete
Digest: sha256:9d6b58feebd2dbd3c56ab5853333d627cc6e281011cfd6050fa4bcf2072c9496
Status: Downloaded newer image for nginx:latest
ba2ef934472e63cc2551c9f8c4d9ad2915cb0c7a473a7aadc53b414d60241506
```

Name of the Student:	**Mitali Bhattad**	**PRN:**	**20220801063**
Title of Practical:	**Implementing Live Changes in a Running Docker Container**		

6. Now, open your browser and paste the public ipv4 of your instance
- http://<your-public-ip>
You should see the text **"Hello from Mitali"**.

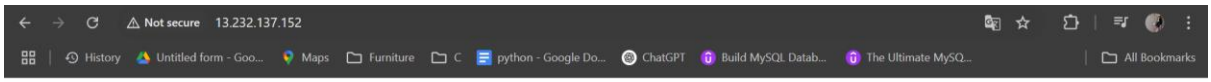


Hello from Mitali

7. Modify the content of index.html to check whether we are able to see the updated content or not:
- echo "<h1>Updated content</h1>" > ~/my-website/index.html

```
root@ip-172-31-39-97:~/my-website# echo "<h1>Updated content</h1>" > ~/my-website/index.html
```

8. Refresh the Web Page
You should now see the updated text **"Updated content"** without restarting the container.



Updated content