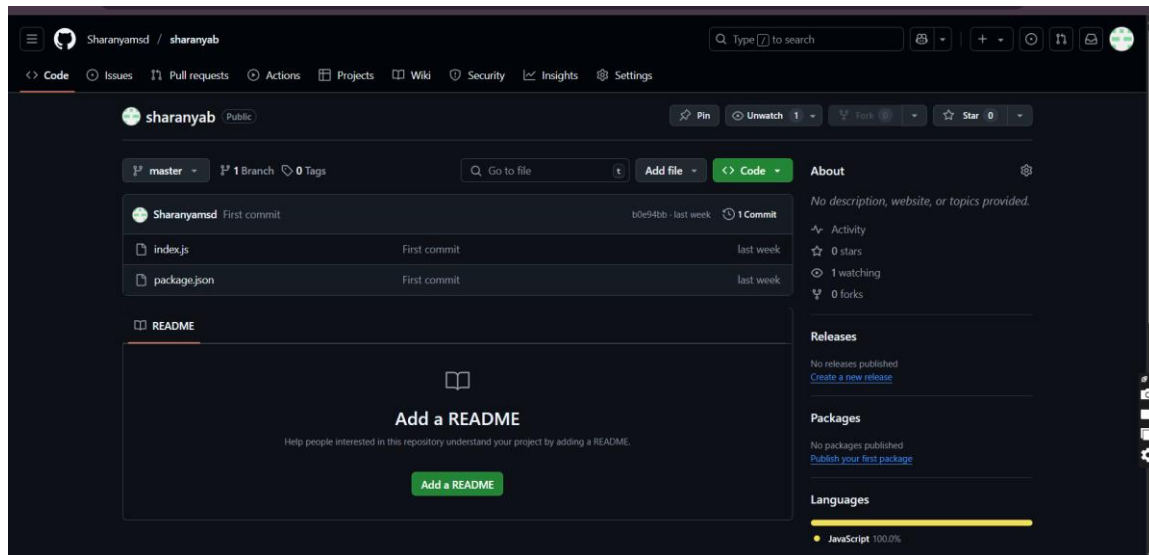


Assignment No:12

Title: Deploy and run the project in AWS without using the port.

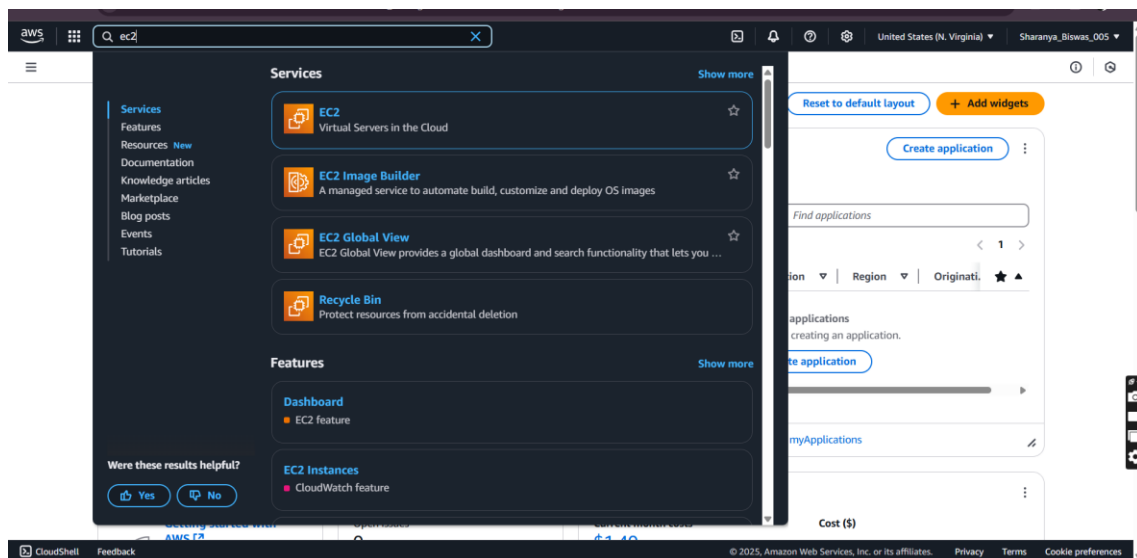
Step-1:

Upload required files to github



Step-2:

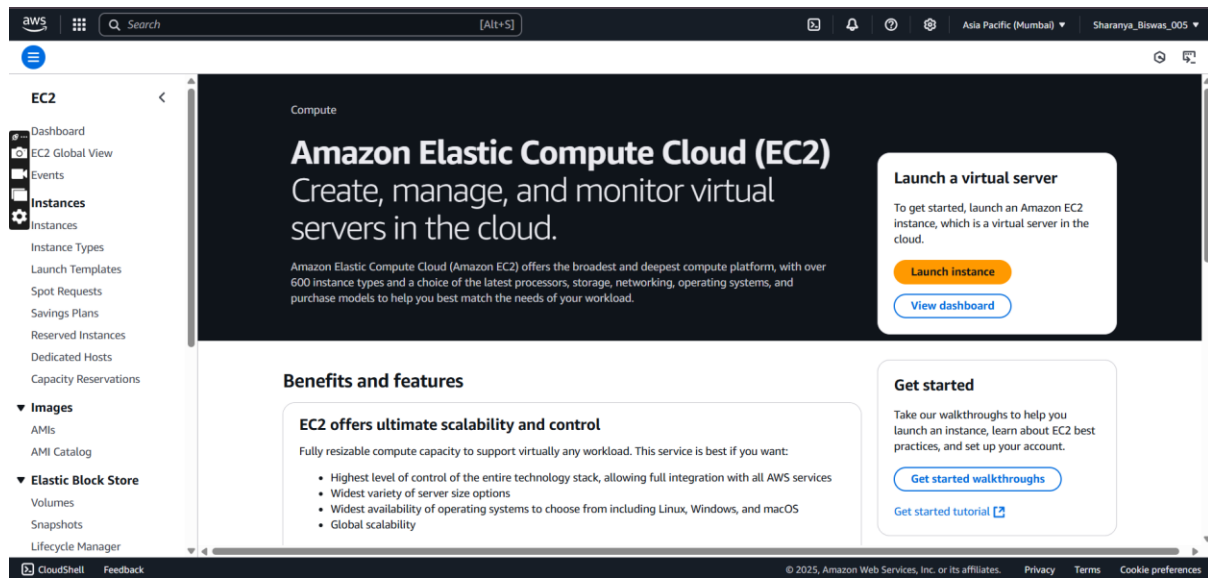
Log into AWS and open EC2.



Step-3:

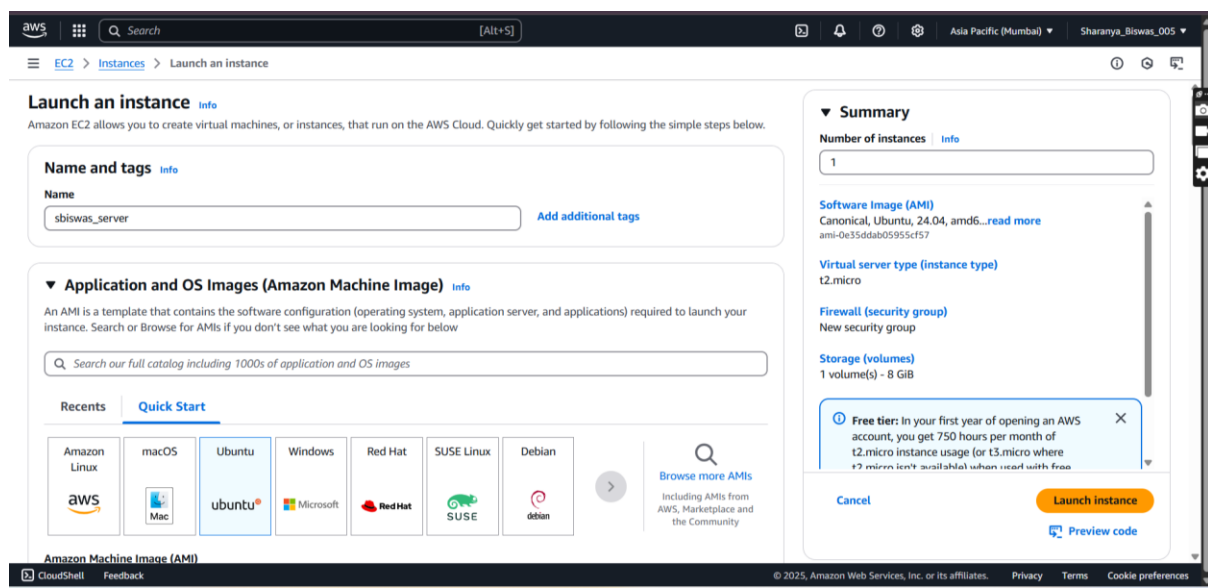
Open the EC2

Name:Sharanya Biswas
Class:CSE(DS)/22/005

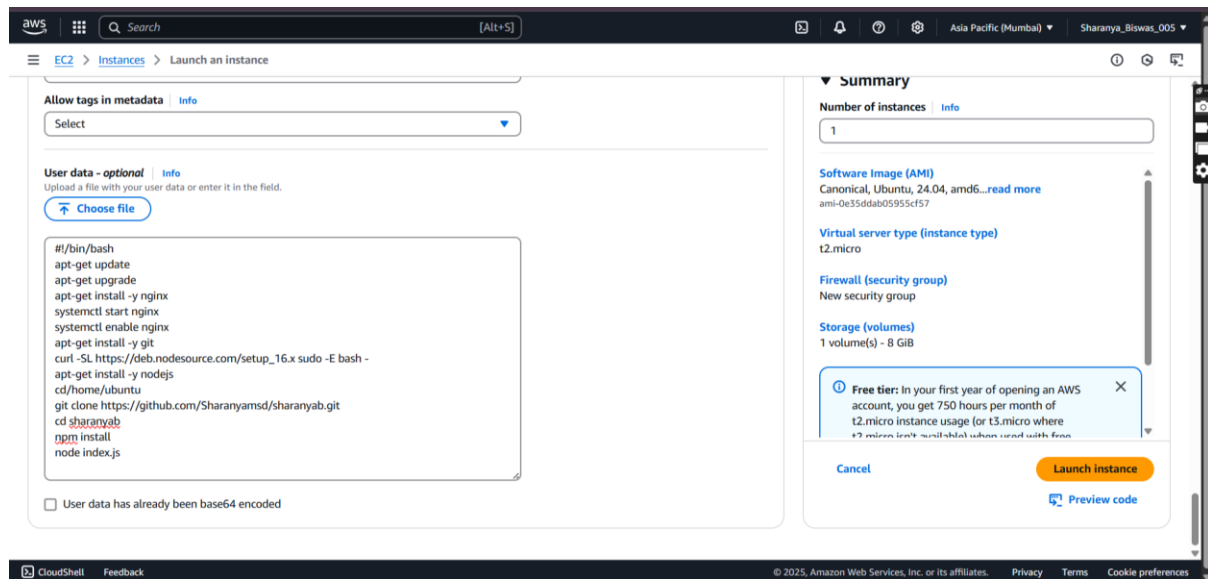


Step-4:

Create ec2 instance and add the below bash commands in user data section.

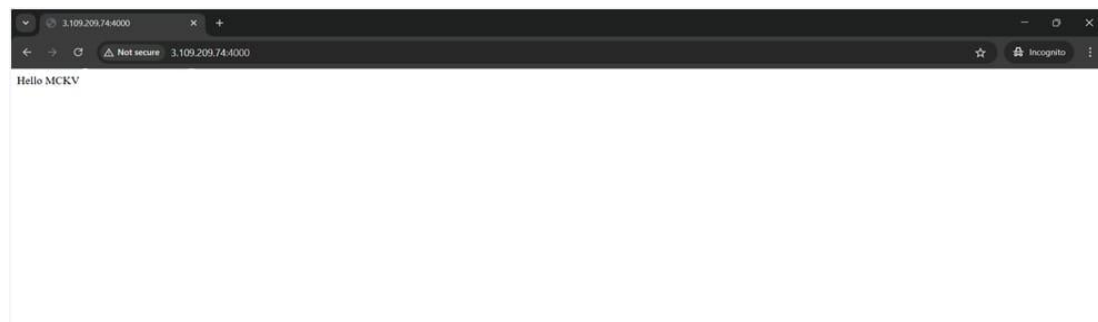


Name: Sharanya Biswas
Class: CSE(DS)/22/005



Step-5:

After creating instance the site should successfully run on the port 4000.



Step-6:

Now log into bitwise and cd ../.. to go to root directory then open etc with cd etc and nginx with cd nginx and sites-available with cd sites-available then change the permissions of the default file using sudo chmod 777 default. Then edit the default file with nano default and change the code to what is given below and close the editor then restart the nginx with sudo systemctl restart nginx.

```
ubuntu@3.109.209.7422 - Bitvise xterm - ubuntu@ip-172-31-8-237:/etc/nginx/sites-available
System load:  0.0      Processes:    111
Usage of /:   34.0% of 6.71GB
Memory usage: 30%
Swap usage:   0%

* Ubuntu Pro delivers the most comprehensive open source security and
  compliance features.

  https://ubuntu.com/aws/pro

Expanded Security Maintenance for Applications is not enabled.

9 updates can be applied immediately.
3 of these updates are standard security updates.
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: Mon Apr 21 17:55:00 2025 from 116.206.203.191
ubuntu@ip-172-31-8-237:~$ cd ../..
ubuntu@ip-172-31-8-237:/$ cd etc
ubuntu@ip-172-31-8-237:/etc$ cd nginx
ubuntu@ip-172-31-8-237:/etc/nginx$ ls
conf.d      fastcgi_params  koi-utf        modules-enabled  nginx.conf      scgi_params    sites-enabled  uwsgi_params
fastcgi.conf  koi-utf        mime.types     modules-enabled  proxy_params    sites-available  snippets      win-utf
ubuntu@ip-172-31-8-237:/etc/nginx$ cd sites-available
-bash: cd: too many arguments
ubuntu@ip-172-31-8-237:/etc/nginx$ cd sites-available
ubuntu@ip-172-31-8-237:/etc/nginx/sites-available$ ls
default
ubuntu@ip-172-31-8-237:/etc/nginx/sites-available$ sudo chmod 777 default
ubuntu@ip-172-31-8-237:/etc/nginx/sites-available$ nano default
ubuntu@ip-172-31-8-237:/etc/nginx/sites-available$ sudo chmod 777 default
ubuntu@ip-172-31-8-237:/etc/nginx/sites-available$ sudo systemctl restart nginx
Warning: The unit file, source configuration file or drop-ins of nginx.service changed on disk. Run 'systemctl daemon-reload' to reload units.
Job for nginx.service failed because the control process exited with error code.
See "systemctl status nginx.service" and "journalctl -xeu nginx.service" for details.
ubuntu@ip-172-31-8-237:/etc/nginx/sites-available$ nano default
ubuntu@ip-172-31-8-237:/etc/nginx/sites-available$ sudo systemctl restart nginx
Warning: The unit file, source configuration file or drop-ins of nginx.service changed on disk. Run 'systemctl daemon-reload' to reload units.
ubuntu@ip-172-31-8-237:/etc/nginx/sites-available$
```

```
GNU nano 7.2      default
#
# Self signed certs generated by the ssl-cert package
# Don't use them in a production server!
#
# include snippets/snakeoil.conf;

root /var/www/html;

# Add index.php to the list if you are using PHP
index index.html index.htm index.nginx-debian.html;

server_name _;

location / {
    # First attempt to serve request as file, then
    # as directory, then fall back to displaying a 404.
    # try_files $uri $uri/ =404;
    proxy_pass http://localhost:4000;
    proxy_http_version 1.1;
    proxy_set_header Upgrade $http_upgrade;
    proxy_set_header Connection 'Upgrade';
    proxy_set_header Host $host;
    proxy_cache_bypass $http_upgrade;
}

# pass PHP scripts to FastCGI server
#
#location ~ \.php$ {
#    include snippets/fastcgi-php.conf;
#
#    # With php-fpm (or other unix sockets):
#    fastcgi_pass unix:/run/php/php7.4-fpm.sock;
#    # With php-cgi (or other tcp sockets):
#    fastcgi_pass 127.0.0.1:9000;
#}

# deny access to .htaccess files, if Apache's document root
# concurs with nginx's one
#
```

Step-7:

Now the site runs on IP without giving the port.



Name:Sharanya Biswas
Class:CSE(DS)/22/005