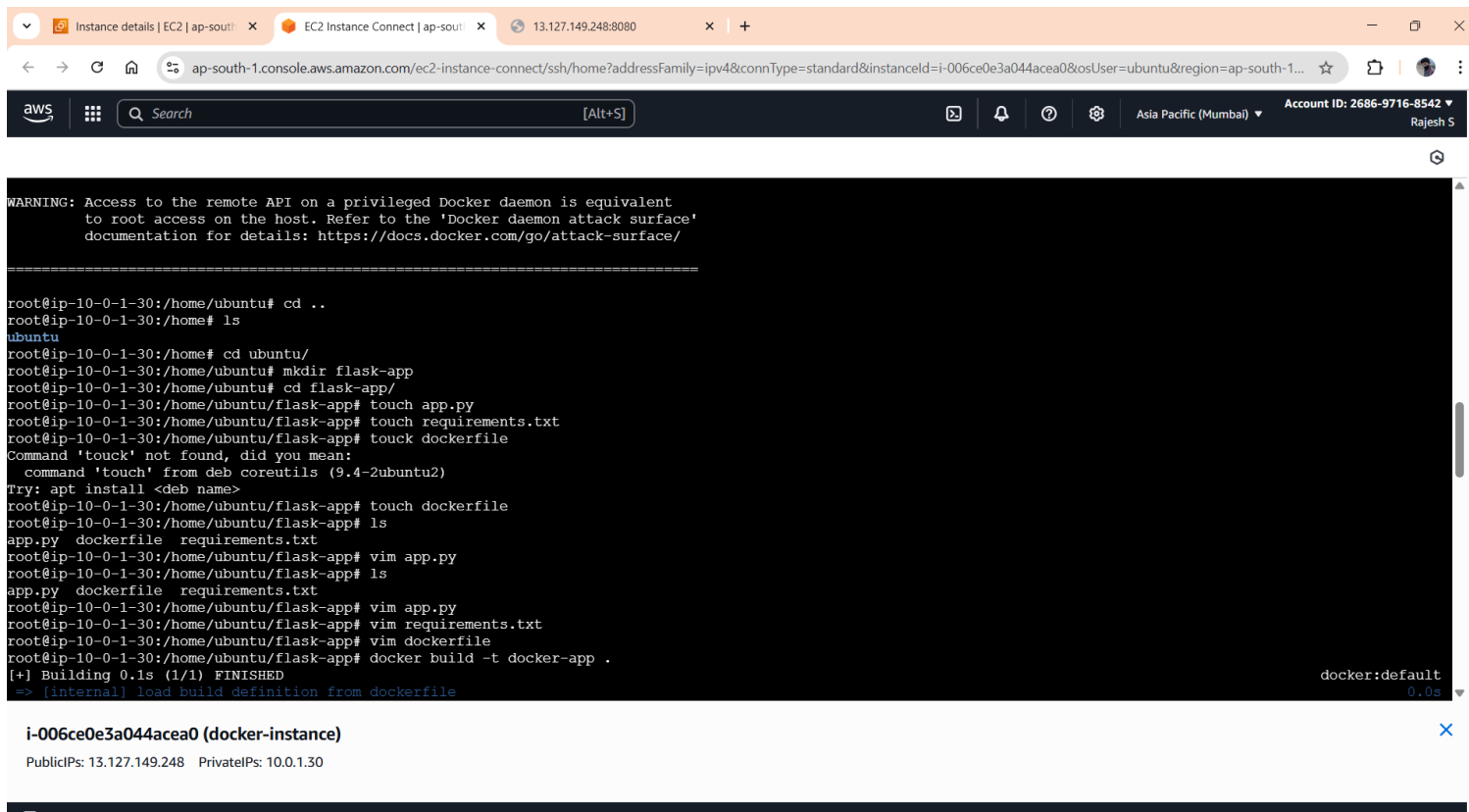


# MINI PROJECT 2

## TASKS:

1. Create a docker image with simple python web app you can use (flask)

## RESULTS:



The screenshot shows the AWS Management Console interface with a terminal window open for an EC2 instance. The terminal displays the following commands and output:

```
WARNING: Access to the remote API on a privileged Docker daemon is equivalent
to root access on the host. Refer to the 'Docker daemon attack surface'
documentation for details: https://docs.docker.com/go/attack-surface/

=====

root@ip-10-0-1-30:/home/ubuntu# cd ..
root@ip-10-0-1-30:/home# ls
ubuntu
root@ip-10-0-1-30:/home# cd ubuntu/
root@ip-10-0-1-30:/home/ubuntu# mkdir flask-app
root@ip-10-0-1-30:/home/ubuntu# cd flask-app/
root@ip-10-0-1-30:/home/ubuntu/flask-app# touch app.py
root@ip-10-0-1-30:/home/ubuntu/flask-app# touch requirements.txt
root@ip-10-0-1-30:/home/ubuntu/flask-app# touch dockerfile
Command 'touch' not found, did you mean:
  command 'touch' from deb coreutils (9.4-2ubuntu2)
Try: apt install <deb name>
root@ip-10-0-1-30:/home/ubuntu/flask-app# touch dockerfile
root@ip-10-0-1-30:/home/ubuntu/flask-app# ls
app.py  dockerfile  requirements.txt
root@ip-10-0-1-30:/home/ubuntu/flask-app# vim app.py
root@ip-10-0-1-30:/home/ubuntu/flask-app# ls
app.py  dockerfile  requirements.txt
root@ip-10-0-1-30:/home/ubuntu/flask-app# vim app.py
root@ip-10-0-1-30:/home/ubuntu/flask-app# vim requirements.txt
root@ip-10-0-1-30:/home/ubuntu/flask-app# vim dockerfile
root@ip-10-0-1-30:/home/ubuntu/flask-app# docker build -t docker-app .
[+] Building 0.1s (1/1) FINISHED
=> [internal] load build definition from dockerfile
docker:default 0.0s
```

Below the terminal output, the instance details are shown:

**i-006ce0e3a044acea0 (docker-instance)**

PublicIPs: 13.127.149.248 PrivateIPs: 10.0.1.30

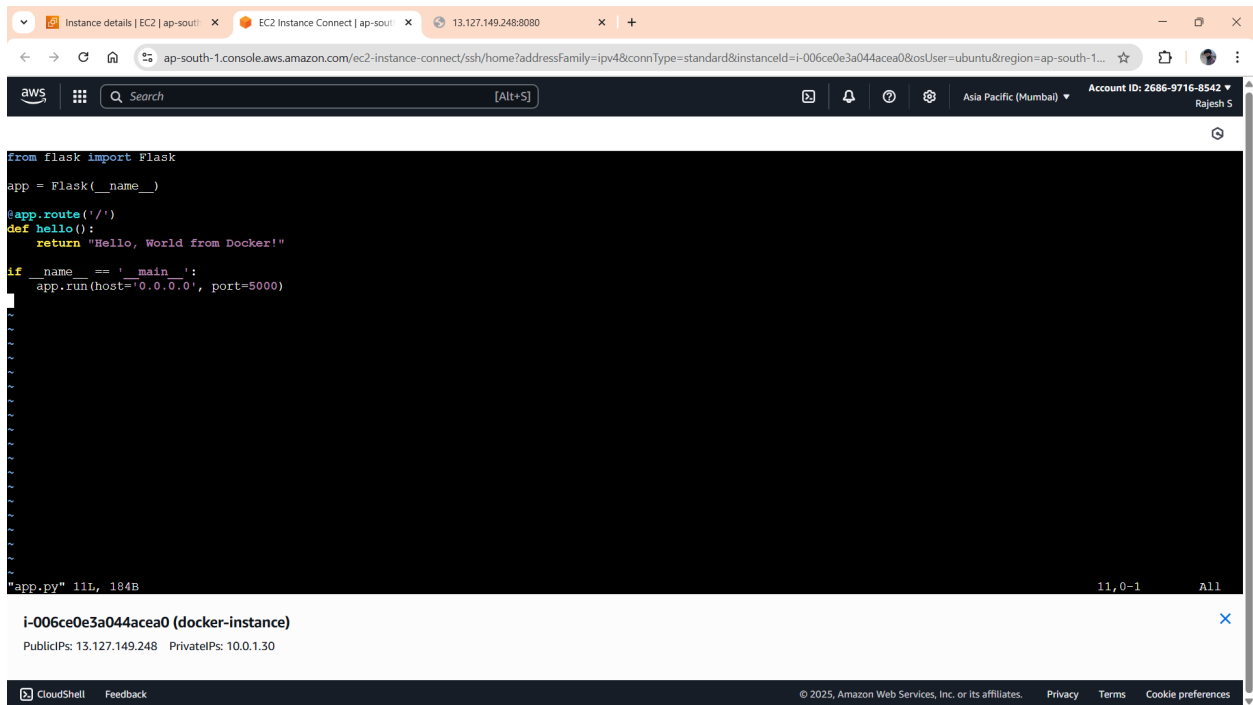


IMAGE CREATED:

```
[+] Building 0.1s (1/1) FINISHED                                docker:default
=> [internal] load build definition from dockerfile              0.0s
=> => transferring dockerfile: 322B                             0.0s
dockerfile:12
-----
10 |
11 | # Copy application code
12 | >>> COPY app.py
13 |
14 | # Run the app
-----
ERROR: failed to build: failed to solve: dockerfile parse error on line 12: COPY requires at least two arguments, but only one was provided. Destination could not be determined

root@ip-10-0-1-30:/home/ubuntu/flask-app# docker images
REPOSITORY TAG IMAGE ID CREATED SIZE
docker-app latest 840fe0bd065f 24 seconds ago 145MB
root@ip-10-0-1-30:/home/ubuntu/flask-app# docker run -d name flask-app -p 80:8080 docker-app
Unable to find image 'name:latest' locally
docker: Error response from daemon: pull access denied for name, repository does not exist or may require 'docker login': denied: requested access to the resource is denied

Run 'docker run --help' for more information
root@ip-10-0-1-30:/home/ubuntu/flask-app# docker run --name docker-app -p 80:8080
docker: 'docker run' requires at least 1 argument

Usage: docker run [OPTIONS] IMAGE [COMMAND] [ARG...]

See 'docker run --help' for more information
root@ip-10-0-1-30:/home/ubuntu/flask-app#
```

OUTPUT:

