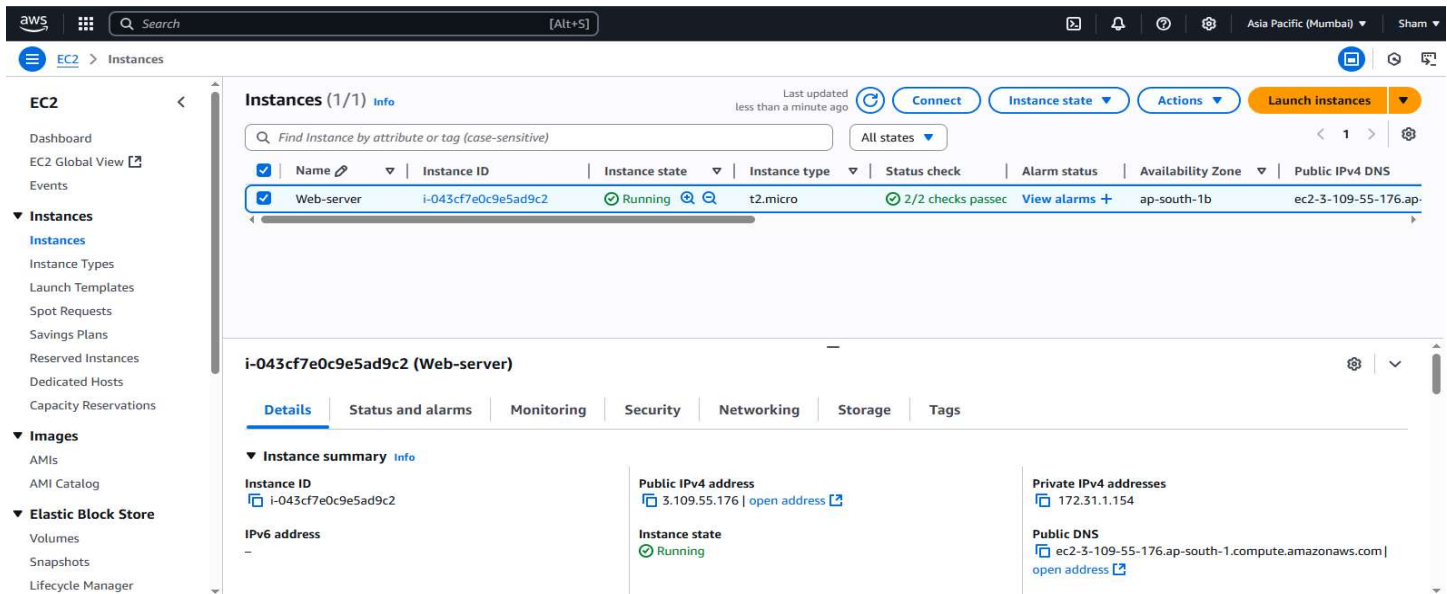


Build, and Scale Node.js & Flask Applications Using Docker Compose.

1. Create Instance with Docker and Docker Compose Socket.



```
root@ip-172-31-1-154:~# docker --version
Docker version 28.3.0, build 38b7060
root@ip-172-31-1-154:~# docker compose version
Docker Compose version v2.37.3
root@ip-172-31-1-154:~# ls
flask-app  node-app  snap
root@ip-172-31-1-154:~#
```

2. Writing Docker Compose file with Build Services and Network.

```
root@ip-172-31-1-154:~# ls
flask-app  node-app  snap
root@ip-172-31-1-154:~# vi docker-compose.yml
root@ip-172-31-1-154:~# cat docker-compose.yml
```

```
networks:
  my-network:
    driver: bridge

services:
  node-app:
    build: ./node-app
    container_name: my-node-app
    ports:
      - "3000:3000"
    networks:
      - my-network
    environment:
      Dept: DEVOPMENT

  flask-app:
    build: ./flask-app
    container_name: my-flask-app
    ports:
      - "5000:5000"
    networks:
      - my-network
    environment:
      Dept: QA
```

3. Build Image and Running Container With docker-compose.yml file.

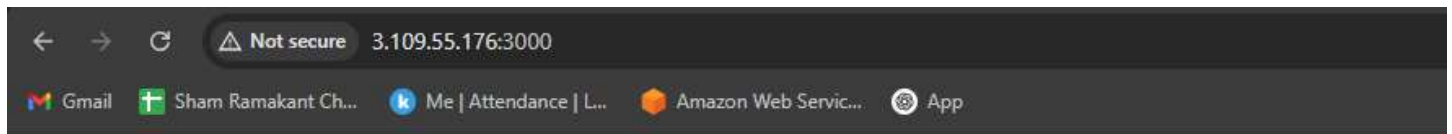
```
root@ip-172-31-1-154:~# ls
docker-compose.yml flask-app node-app snap
root@ip-172-31-1-154:~# docker compose up --build -d
[+] Building 29.8s (22/22) FINISHED
=> [internal] load local bake definitions
=> => reading from stdin 586B
=> [node-app internal] load build definition from Dockerfile
=> => transferring dockerfile: 154B
=> [flask-app internal] load build definition from Dockerfile
=> => transferring dockerfile: 368B
=> [flask-app internal] load metadata for docker.io/library/python:3.11-slim
=> [node-app internal] load metadata for docker.io/library/node:18-alpine
=> [flask-app internal] load .dockerignore
=> => transferring context: 2B
=> [flask-app 1/4] FROM docker.io/library/python:3.11-slim@sha256:139020233cc412efe4c8135b0efe1c7569dc8b28ddd88bddb109b764f8977e30
=> => resolve docker.io/library/python:3.11-slim@sha256:139020233cc412efe4c8135b0efe1c7569dc8b28ddd88bddb109b764f8977e30
=> => sha256:139020233cc412efe4c8135b0efe1c7569dc8b28ddd88bddb109b764f8977e30 9.13kB / 9.13kB
=> => sha256:153bae509ec02c9ac789e2e35f3cbe94be446b59c3afc7b9c88c96a344d2eb73 1.75kB / 1.75kB
=> => sha256:0b14a859cd8a15104c5f194ef813fccc72749d74bc7be4550c06a0fc0d482e6 5.37kB / 5.37kB
=> => sha256:3da95a905ed546f99c4564407923a681757d89651a388ec3f1f5e9bf5ed0b39d 28.23MB / 28.23MB
=> => sha256:483d0dd375188d7d3b2d66116d5974d2b67e6988c6146eb2c6a3e2bc33a92121 3.51MB / 3.51MB
=> => sha256:02a5d22e0d6f85a6ac1c7fb356e9eed39a981decdd1fac1205a31f31f4cb010f1 16.21MB / 16.21MB
=> => sha256:471797cdda8c4cd4a9795c8cb56078e627b3fc7486fd8574804ec7d5f1676b9b 249B / 249B
=> => extracting sha256:3da95a905ed546f99c4564407923a681757d89651a388ec3f1f5e9bf5ed0b39d
=> => extracting sha256:483d0dd375188d7d3b2d66116d5974d2b67e6988c6146eb2c6a3e2bc33a92121
=> => extracting sha256:02a5d22e0d6f85a6ac1c7fb356e9eed39a981decdd1fac1205a31f31f4cb010f1
=> => extracting sha256:471797cdda8c4cd4a9795c8cb56078e627b3fc7486fd8574804ec7d5f1676b9b
=> [flask-app internal] load build context
=> => transferring context: 34.44kB
=> [node-app internal] load .dockerignore
```

```
=> => sha256:25ff2da83641908f65c3a74d80409d6b1b62ccfaab220b9ea70b80df5a2e0549 446B / 446B
=> => extracting sha256:dd71dde834b5c203d162902e6b8994cb2309ae049a0eabc4efea161b2b5a3d0e
=> => extracting sha256:1e5a4c89cee5c0826c540ab06d4b6b491c96eda01837f430bd47f0d26702d6e3
=> => extracting sha256:25ff2da83641908f65c3a74d80409d6b1b62ccfaab220b9ea70b80df5a2e0549
=> [node-app internal] load build context
=> => transferring context: 34.53kB
=> [flask-app 2/4] WORKDIR /app
=> [flask-app 3/4] COPY . .
=> [flask-app 4/4] RUN pip install -r requirements.txt
=> [node-app 2/5] WORKDIR /app
=> [node-app 3/5] COPY package.json ./
=> [node-app 4/5] RUN npm install
=> [node-app 5/5] COPY . .
=> [node-app] exporting to image
=> => exporting layers
=> => writing image sha256:f3e4d1b80dd5c75b2961fc2f8c8efc894598ead953ec6abbd3055c06df6459a4
=> => naming to docker.io/library/root-node-app
=> [node-app] resolving provenance for metadata file
=> [flask-app] exporting to image
=> => exporting layers
=> => writing image sha256:2b215c3143dd86a66343e124398694cb8a897bc34fde8b6503efa322726f5d78
=> => naming to docker.io/library/root-flask-app
=> [flask-app] resolving provenance for metadata file
[+] Running 5/5
✔ flask-app Built
✔ node-app Built
✔ Network root_my-network Created
✔ Container my-node-app Started
✔ Container my-flask-app Started
root@ip-172-31-1-154:~#
```

4. Verify Logs and Application using Server IP.

```
root@ip-172-31-1-154:~# docker images
REPOSITORY TAG IMAGE ID CREATED SIZE
root-flask-app latest 2b215c3143dd 3 minutes ago 330MB
root-node-app latest f3e4d1b80dd5 3 minutes ago 138MB
root@ip-172-31-1-154:~# docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
f30efcd04538 root-flask-app "python app.py" 3 minutes ago Up 3 minutes 0.0.0.0:5000->5000/tcp, [::]:5000->5000/tcp my-flask-app
a8066a688c2e root-node-app "docker-entrypoint.s..." 3 minutes ago Up 3 minutes 0.0.0.0:3000->3000/tcp, [::]:3000->3000/tcp my-node-app
root@ip-172-31-1-154:~# docker network ls
NETWORK ID NAME DRIVER SCOPE
d1080da20ab1 bridge bridge local
9af51bbd2fd1 host host local
b20f90550bc0 none null local
28333997284b root_my-network bridge local
root@ip-172-31-1-154:~#
```

Node-App:

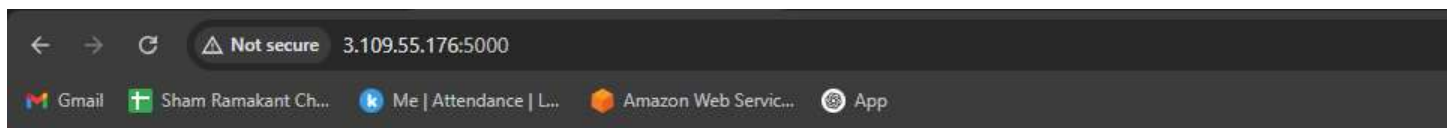


Welcome to My Node.js App!

Click below to get a random programming quote:

[Get Quote](#)

Flask-App:

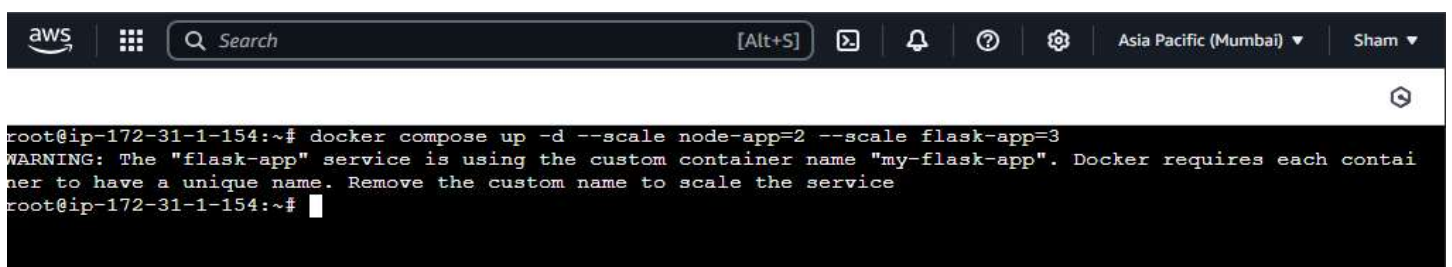


Welcome to Flask App on EC2!

This app is running directly on EC2 without Docker.

- [/fetch - External API Call](#)
- [/health - Health Check](#)

5. Scale up Container



Bcos of Custom name we cannot run the container so removing Container name in yaml file and scaling.

```
root@ip-172-31-1-154:~# vi docker-compose.yml
root@ip-172-31-1-154:~# cat docker-compose.yml
```

```
networks:
  my-network:
    driver: bridge

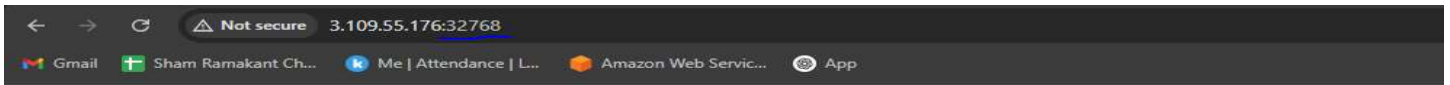
services:
  node-app:
    build: ./node-app
    ports:
      - "3000"
    networks:
      - my-network
    environment:
      Dept: DEVOPMENT

  flask-app:
    build: ./flask-app
    ports:
      - "5000"
    networks:
      - my-network
    environment:
      Dept: QA
```

```
=> => transferring context: 2.08kB
=> [node-app 1/5] FROM docker.io/library/node:18-alpine@sha256:8d6421d663b4c28fd3ebc498332f249011d118945588d0a35cb9bc4b8ca09d9e
=> [node-app internal] load build context
=> => transferring context: 2.07kB
=> CACHED [flask-app 2/4] WORKDIR /app
=> CACHED [flask-app 3/4] COPY . .
=> CACHED [flask-app 4/4] RUN pip install -r requirements.txt
=> CACHED [node-app 2/5] WORKDIR /app
=> CACHED [node-app 3/5] COPY package.json ./
=> CACHED [node-app 4/5] RUN npm install
=> CACHED [node-app 5/5] COPY . .
=> [flask-app] exporting to image
=> => exporting layers
=> => writing image sha256:2b215c3143dd86a66343e124398694cb8a897bc34fde8b6503efa322726f5d78
=> => naming to docker.io/library/root-flask-app
=> [node-app] exporting to image
=> => exporting layers
=> => writing image sha256:f3e4d1b80dd5c75b2961fc2f8c8efc894598ead953ec6abbd3055c06df6459a4
=> => naming to docker.io/library/root-node-app
=> [flask-app] resolving provenance for metadata file
=> [node-app] resolving provenance for metadata file
[+] Running 5/5
✓ flask-app           Built
✓ node-app           Built
✓ Network root_my-network Created
✓ Container root-flask-app-1 Started
✓ Container root-node-app-1 Started
root@ip-172-31-1-154:~# vi docker-compose.yml
```

Scale Container Manually.

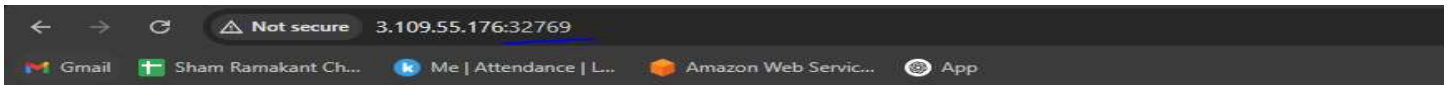
```
root@ip-172-31-1-154:~# docker compose up -d --scale node-app=2 --scale flask-app=3
[+] Running 5/5
✓ Container root-flask-app-3 Started
✓ Container root-node-app-1 Started
✓ Container root-flask-app-2 Started
✓ Container root-flask-app-1 Started
✓ Container root-node-app-2 Started
root@ip-172-31-1-154:~# docker ps
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS                               NAMES
9412cddcdd81   root-flask-app "python app.py"         19 seconds ago Up 8 seconds  0.0.0.0:32768->5000/tcp, [::]:32768->5000/tcp  root-flask-app-1
2287e61e35d3   root-node-app  "docker-entrypoint.s..." 19 seconds ago Up 8 seconds  0.0.0.0:32769->3000/tcp, [::]:32769->3000/tcp  root-node-app-2
6fa6294768e9   root-flask-app "python app.py"         19 seconds ago Up 8 seconds  0.0.0.0:32770->5000/tcp, [::]:32770->5000/tcp  root-flask-app-2
2aad8a67ef70   root-flask-app "python app.py"         19 seconds ago Up 7 seconds  0.0.0.0:32772->5000/tcp, [::]:32772->5000/tcp  root-flask-app-3
7aef416608d5   root-node-app  "docker-entrypoint.s..." 19 seconds ago Up 8 seconds  0.0.0.0:32771->3000/tcp, [::]:32771->3000/tcp  root-node-app-1
root@ip-172-31-1-154:~#
```



Welcome to Flask App on EC2!

This app is running directly on EC2 without Docker.

- [/fetch - External API Call](#)
- [/health - Health Check](#)



Welcome to My Node.js App!

Click below to get a random programming quote:

[Get Quote](#)

Scale Container with docker-compose.yml

```
root@ip-172-31-1-154:~# vi docker-compose.yml
root@ip-172-31-1-154:~# cat docker-compose.yml

networks:
  my-network:
    driver: bridge

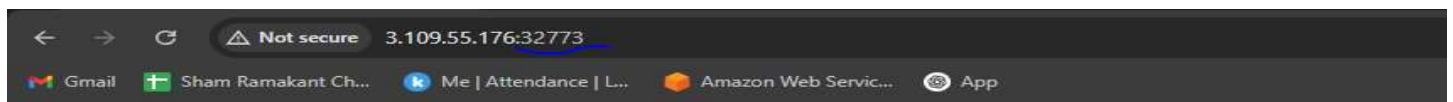
services:
  node-app:
    build: ./node-app
    ports:
      - "3000"
    networks:
      - my-network
    environment:
      Dept: DEVOPMENT
    deploy:
      replicas: 4

  flask-app:
    build: ./flask-app
    ports:
      - "5000"
    networks:
      - my-network
    environment:
      Dept: QA
    deploy:
      replicas: 4
```

```

root@ip-172-31-1-154:~# docker compose up --build -d
[+] Building 2.3s (22/22) FINISHED
=> [internal] load local bake definitions
=> => reading from stdin 586B
=> [node-app internal] load build definition from Dockerfile
=> => transferring dockerfile: 154B
=> [flask-app internal] load build definition from Dockerfile
=> => transferring dockerfile: 368B
=> [flask-app internal] load metadata for docker.io/library/python:3.11-slim
=> [node-app internal] load metadata for docker.io/library/node:18-alpine
=> [node-app internal] load .dockerignore
=> => transferring context: 2B
=> [flask-app internal] load .dockerignore
=> => transferring context: 2B
=> [node-app 1/5] FROM docker.io/library/node:18-alpine4c8a256:8d6431d663b4c385d3b6c498322f6249011d118845588d0e35-
=> CACHED [flask-app 3/4] COPY . .
=> CACHED [flask-app 4/4] RUN pip install -r requirements.txt
=> [flask-app] exporting to image
=> => exporting layers
=> => writing image sha256:2b215c3143dd86a66343e124398694cb8a897bc34fde8b6503efa322726f5d78
=> => naming to docker.io/library/root-flask-app
=> [node-app] resolving provenance for metadata file
=> [flask-app] resolving provenance for metadata file
[+] Running 10/10
✓ node-app Built
✓ flask-app Built
✓ Container root-flask-app-4 Started
✓ Container root-node-app-4 Started
✓ Container root-flask-app-1 Started
✓ Container root-flask-app-2 Started
✓ Container root-flask-app-3 Started
✓ Container root-node-app-2 Started
✓ Container root-node-app-1 Started
✓ Container root-node-app-3 Started
root@ip-172-31-1-154:~# docker ps
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS                               NAMES
ce64e48fdc02   root-node-app  "docker-entrypoint.s..." About a minute ago Up About a minute 0.0.0.0:32773->3000/tcp, [::]:32773->3000/tcp root-node-app-1
5d61854f9a9a   root-node-app  "docker-entrypoint.s..." About a minute ago Up About a minute 0.0.0.0:32775->3000/tcp, [::]:32775->3000/tcp root-node-app-2
9d0441ec909e   root-flask-app "python app.py"         About a minute ago Up About a minute 0.0.0.0:32774->5000/tcp, [::]:32774->5000/tcp root-flask-app-3
7f30e288edf1   root-flask-app "python app.py"         About a minute ago Up About a minute 0.0.0.0:32776->5000/tcp, [::]:32776->5000/tcp root-flask-app-2
bce0c1084ac8   root-flask-app "python app.py"         About a minute ago Up About a minute 0.0.0.0:32778->5000/tcp, [::]:32778->5000/tcp root-flask-app-1
a233c1de4877   root-node-app  "docker-entrypoint.s..." About a minute ago Up About a minute 0.0.0.0:32777->3000/tcp, [::]:32777->3000/tcp root-node-app-4
5a00dc5274f1   root-node-app  "docker-entrypoint.s..." About a minute ago Up About a minute 0.0.0.0:32779->3000/tcp, [::]:32779->3000/tcp root-node-app-3
83f14027155e   root-flask-app "python app.py"         About a minute ago Up About a minute 0.0.0.0:32780->5000/tcp, [::]:32780->5000/tcp root-flask-app-4
root@ip-172-31-1-154:~#

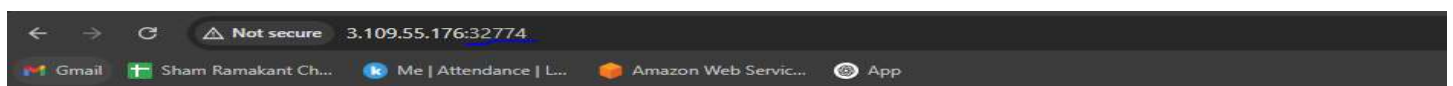
```



Welcome to My Node.js App!

Click below to get a random programming quote:

[Get Quote](#)



Welcome to Flask App on EC2!

This app is running directly on EC2 without Docker.

- [/fetch - External API Call](#)
- [/health - Health Check](#)