

School of Computer Science, Engineering and Applications(SCSEA)
B.Tech FIY (CCSA)
Subject : Cloud Automation & Devops (P)

Name of the Student: Pratik.M.Rebari

PRN: 20220802183

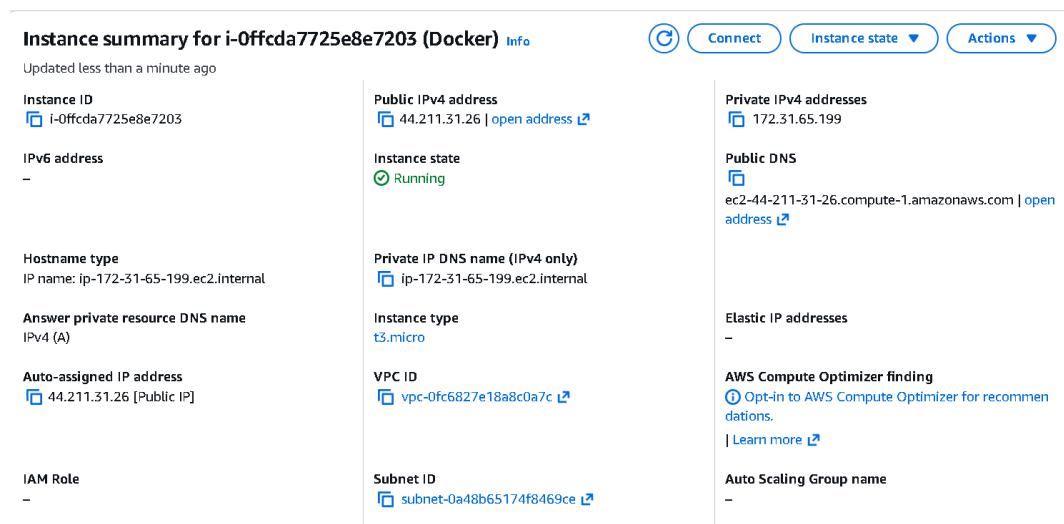
Title of Practicle : 3. Containerizing applications with Docker and Docker Compose

1. Launch the instance.

Name: Docker

AMI: Amazon Linux2023

Instance Type: t3.micro



Instance summary for i-0ffcda7725e8e7203 (Docker) Info

Updated less than a minute ago

Instance ID i-0ffcda7725e8e7203	Public IPv4 address 44.211.31.26 open address	Private IPv4 addresses 172.31.65.199
IPv6 address -	Instance state Running	Public DNS ec2-44-211-31-26.compute-1.amazonaws.com open address
Hostname type IP name: ip-172-31-65-199.ec2.internal	Private IP DNS name (IPv4 only) ip-172-31-65-199.ec2.internal	Elastic IP addresses -
Answer private resource DNS name IPv4 (A)	Instance type t3.micro	AWS Compute Optimizer finding Opt-in to AWS Compute Optimizer for recommendations. Learn more
Auto-assigned IP address 44.211.31.26 [Public IP]	VPC ID vpc-0fc6827e18a8c0a7c	Auto Scaling Group name -
IAM Role -	Subnet ID subnet-0a48b65174f8469ce	

1. Install the Docker & Docker Compose.

```
sudo yum update -y
```

```
sudo dnf install docker -y
```

```
sudo systemctl start docker
```

```
sudo systemctl enable docker
```

```
sudo usermod -aG docker ec2-user
```

```
sudo usermod -aG docker $USER
```

```
sudo curl -L
```

```
"https://github.com/docker/compose/releases/download/v2.29.2/docker-compose-  
linux-x86_64" -o /usr/local/bin/docker-compose
```

School of Computer Science, Engineering and Applications(SCSEA)
B.Tech FIY (CCSA)
Subject : Cloud Automation & Devops (P)

Name of the Student: Pratik.M.Rebari

PRN: 20220802183

Title of Practicle : 3. Containerizing applications with Docker and Docker Compose

```
sudo chmod +x /usr/local/bin/docker-compose
```

```
docker --version
```

```
docker compose version
```

```
docker-compose version
```

```
ec2-user@ip-172-31-65-199:~$ sudo yum update -y
sudo dnf install docker -y
sudo systemctl start docker
sudo systemctl enable docker
sudo usermod -aG docker ec2-user
sudo curl -L "https://github.com/docker/compose/releases/download/v2.29.2/docker-compose-linux-x86_64" -o /usr/local/bin/docker-compose
sudo chmod +x /usr/local/bin/docker-compose
docker --version
docker compose version
docker-compose version
```

```
ec2-user@ip-172-31-65-199:~$ Running scriptlet: container-selinux-4:2.242.0-1.amzn2023.noarch 11/11
Running scriptlet: docker-25.0.13-1.amzn2023.0.2.x86_64 11/11
Verifying : container-selinux-4:2.242.0-1.amzn2023.noarch 1/11
Verifying : containerd-2.1.4-1.amzn2023.0.2.x86_64 2/11
Verifying : docker-25.0.13-1.amzn2023.0.2.x86_64 3/11
Verifying : iptables-libs-1.8.8-3.amzn2023.0.2.x86_64 4/11
Verifying : iptables-nft-1.8.8-3.amzn2023.0.2.x86_64 5/11
Verifying : libcgrou-3.0-1.amzn2023.0.1.x86_64 6/11
Verifying : libnetfilter_conntrack-1.0.8-2.amzn2023.0.2.x86_64 7/11
Verifying : libnftnl-1.0.1-19.amzn2023.0.2.x86_64 8/11
Verifying : libnftnl-1.2.2-2.amzn2023.0.2.x86_64 9/11
Verifying : pigz-2.5-1.amzn2023.0.3.x86_64 10/11
Verifying : runc-1.3.3-2.amzn2023.0.1.x86_64 11/11

Installed:
  container-selinux-4:2.242.0-1.amzn2023.noarch          containerd-2.1.4-1.amzn2023.0.2.x86_64
  docker-25.0.13-1.amzn2023.0.2.x86_64                iptables-libs-1.8.8-3.amzn2023.0.2.x86_64
  iptables-nft-1.8.8-3.amzn2023.0.2.x86_64             libcgrou-3.0-1.amzn2023.0.1.x86_64
  libnetfilter_conntrack-1.0.8-2.amzn2023.0.2.x86_64  libnftnl-1.0.1-19.amzn2023.0.2.x86_64
  libnftnl-1.2.2-2.amzn2023.0.2.x86_64                 libnftnl-1.2.2-2.amzn2023.0.2.x86_64
  runc-1.3.3-2.amzn2023.0.1.x86_64                    pigz-2.5-1.amzn2023.0.3.x86_64

Complete!
Created symlink /etc/systemd/system/multi-user.target.wants/docker.service → /usr/lib/systemd/system/docker.service.
% Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
           Dload  Upload    Total   Spent    Left   Speed
  0     0    0     0    0     0      0      0  0:00:00  0:00:00  0:00:00     0
100 60.2M 100 60.2M    0     0 166M      0  0:00:03  0:00:03  0:00:00 166M
Docker version 25.0.13, build 0bab007
docker: 'compose' is not a docker command.
See 'docker --help'
Docker Compose version v2.29.2
[ec2-user@ip-172-31-65-199 ~]$
```



School of Computer Science, Engineering and Applications(SCSEA)

B.Tech FIY (CCSA)

Subject : Cloud Automation & Devops (P)

Name of the Student: Pratik.M.Rebari

PRN: 20220802183

Title of Practicle : 3. Containerizing applications with Docker and Docker Compose

2. Create Your Project Folder

mkdir myapp

cd myapp

```
ec2-user@ip-172-31-65-199:~$ mkdir myapp
ec2-user@ip-172-31-65-199:~$ ls
myapp
ec2-user@ip-172-31-65-199:~$ cd myapp/
ec2-user@ip-172-31-65-199:myapp$
```

3. Add Your Application Code

Example: Node.js Express app

Create app.js

\$ vi app.js

```
const express = require("express");
```

```
const app = express();
```

```
app.get("/", (req, res) => res.send("Hello from Docker!"));
```

```
app.listen(3000, () => console.log("Server running on port 3000"));
```

School of Computer Science, Engineering and Applications(SCSEA)

B.Tech FIY (CCSA)

Subject : Cloud Automation & Devops (P)

Name of the Student: Pratik.M.Rebari

PRN: 20220802183

Title of Practicle : 3. Containerizing applications with Docker and Docker Compose

```
ec2-user@ip-172-31-65-199:~$ cat index.js
const express = require("express");
const app = express();

app.get("/", (req, res) => res.send("Hello from Docker!"));

app.listen(3000, "0.0.0.0", () => {
  console.log("Server running on port 3000");
});
```

Create package.json

\$ package.json

```
{
  "name": "docker-app",
  "version": "1.0.0",
  "dependencies": {
    "express": "^4.17.1"
  }
}
```

```
ec2-user@ip-172-31-65-199:~$ cat package.json
{
  "name": "docker-app",
  "version": "1.0.0",
  "dependencies": {
    "express": "^4.17.1"
  }
}
```



School of Computer Science, Engineering and Applications(SCSEA)

B.Tech FIY (CCSA)

Subject : Cloud Automation & Devops (P)

Name of the Student: Pratik.M.Rebari

PRN: 20220802183

Title of Practicle : 3. Containerizing applications with Docker and Docker Compose

4. Write the Dockerfile

Create Dockerfile

\$ vi dockerfile

FROM node:18

WORKDIR /app

COPY package*.json ./

RUN npm install

COPY ..

EXPOSE 3000

CMD ["node", "app.js"]

```
ec2-user@ip-172-31-65-199:~$ cat dockerfile
FROM node:18
WORKDIR /app
COPY package*.json ./
RUN npm install
COPY . .
EXPOSE 3000
CMD ["node", "app.js"]
```

School of Computer Science, Engineering and Applications(SCSEA)

B.Tech FIY (CCSA)

Subject : Cloud Automation & Devops (P)

Name of the Student: Pratik.M.Rebari

PRN: 20220802183

Title of Practicle : 3. Containerizing applications with Docker and Docker Compose

5. Build Docker Image

\$ docker build -t myapp .

```
ec2-user@ip-172-31-65-199: ~$ docker build -t myapp .
[+] Building 26.8s (10/10) FINISHED
=> [internal] load build definition from dockerfile
=> => transferring dockerfile: 210B
=> [internal] load metadata for docker.io/library/node:18
=> [internal] load .dockerignore
=> => transferring context: 2B
=> [1/5] FROM docker.io/library/node:18@sha256:c6ae79e38498325db67193d391e6ec1d224d96c693a8a4d9434985 19.9s
=> => resolve docker.io/library/node:18@sha256:c6ae79e38498325db67193d391e6ec1d224d96c693a8a4d9434985 0.0s
=> => sha256:eb29363371ee2859fad6a3c5af88d4abc6ff7d399addb13b7de3c1f11bdee6b9 2.49kB / 2.49kB
=> => sha256:37927ed901b1b2608b72796c6881bf645480268eca4ac9a37b9219e050bb4d84 24.02MB / 24.02MB
=> => sha256:c6ae79e38498325db67193d391e6ec1d224d96c693a8a4d943498556716d3783 6.41kB / 6.41kB
=> => sha256:b50082bc3670d0396b2d90e4b0e5bb10265ba5d0ee16bf40f9a505f7045ee563 6.39kB / 6.39kB
=> => sha256:3e6b9d1a95114e19f12262a4e8a59ad1d1a10ca7b82108adc0605a200294964 48.49MB / 48.49MB
=> => sha256:79b2f47ad4443652b9b5cc81a95ede249fd976310efdbee159f29638783778c0 64.40MB / 64.40MB
=> => extracting sha256:3e6b9d1a95114e19f12262a4e8a59ad1d1a10ca7b82108adc0605a200294964 3.2s
=> => sha256:e23f099911d692f62b851cf49a1e93294288a115f5cd2d014180e4d3684d34ab 211.36MB / 211.36MB
=> => sha256:cda7f44f2bddcc4bb7514474024b3f3705de00ddb6355a33be5ac7808e5b7125 3.32kB / 3.32kB
=> => sha256:c6b30c3f16966552af10ac00521f60355b1fcfd46ac1c20b1038587e28583ce7 45.68MB / 45.68MB
=> => sha256:3697be50c98b9d071df4637e1d3491d00e7b9f3a732768c876d82309b3c5a145 1.25MB / 1.25MB
=> => sha256:461077a72fb7fe40d34a37d6a1958c4d16772d0dd77f572ec50a1fdc41a3754d 446B / 446B
=> => extracting sha256:37927ed901b1b2608b72796c6881bf645480268eca4ac9a37b9219e050bb4d84 0.7s
=> => extracting sha256:79b2f47ad4443652b9b5cc81a95ede249fd976310efdbee159f29638783778c0 3.1s
=> => extracting sha256:e23f099911d692f62b851cf49a1e93294288a115f5cd2d014180e4d3684d34ab 8.5s
=> => extracting sha256:cda7f44f2bddcc4bb7514474024b3f3705de00ddb6355a33be5ac7808e5b7125 0.0s
=> => extracting sha256:c6b30c3f16966552af10ac00521f60355b1fcfd46ac1c20b1038587e28583ce7 2.5s
=> => extracting sha256:3697be50c98b9d071df4637e1d3491d00e7b9f3a732768c876d82309b3c5a145 0.0s
=> => extracting sha256:461077a72fb7fe40d34a37d6a1958c4d16772d0dd77f572ec50a1fdc41a3754d 0.0s
=> [internal] load build context
```

Check image:

\$ docker images

```
ec2-user@ip-172-31-65-199: ~$ docker images
REPOSITORY TAG IMAGE ID CREATED SIZE
myapp latest 11ddef4c9302 45 seconds ago 1.1GB
ec2-user@ip-172-31-65-199: ~$
```



School of Computer Science, Engineering and Applications(SCSEA)

B.Tech FIY (CCSA)

Subject : Cloud Automation & Devops (P)

Name of the Student: Pratik.M.Rebari

PRN: 20220802183

Title of Practicle : 3. Containerizing applications with Docker and Docker Compose

6. Run Container (Without Compose)

```
$ docker run -d -p 3000:3000 --name myapp_container myapp
```

```
ec2-user@ip-172-31-65-199:~$  
[ec2-user@ip-172-31-65-199 myapp]$  
[ec2-user@ip-172-31-65-199 myapp]$ sudo docker run -d -p 3000:3000 myapp  
7db5339680ce1e6f5eb3f0751623323c16ea36073c973c99cbce56992394bc34  
[ec2-user@ip-172-31-65-199 myapp]$ sudo docker ps  
CONTAINER ID   IMAGE      COMMAND                  CREATED        STATUS        PORTS  
7db5339680ce   myapp     "docker-entrypoint.s..." 17 seconds ago Up 16 seconds 0.0.0.0:3000->3000/tcp, :::3000->3000/tcp  
[ec2-user@ip-172-31-65-199 myapp]$
```

Test:

http://<ip>:3000



Hello from Docker!

7. Create docker-compose.yml

```
$ docker-compose.yml
```

```
version: "3.8"
```

```
services:
```

```
  web:
```

```
    build: .
```

```
    container_name: myapp_container
```

```
    ports:
```

```
      - "3000:3000"
```

```
    restart: unless-stopped
```

School of Computer Science, Engineering and Applications(SCSEA)

B.Tech FIY (CCSA)

Subject : Cloud Automation & Devops (P)

Name of the Student: Pratik.M.Rebari

PRN: 20220802183

Title of Practicle : 3. Containerizing applications with Docker and Docker Compose

```
ec2-user@ip-172-31-65-199:~, × + v
version: "3.8"

services:
  web:
    build: .
    container_name: myapp_container
    ports:
      - "3000:3000"
    restart: unless-stopped
~
```

8. Run Using Docker Compose

Start:

\$ docker-compose up -d

```
ec2-user@ip-172-31-65-199:~, × + v
[ec2-user@ip-172-31-65-199 myapp]$ docker-compose up -d
WARN[0000] /home/ec2-user/myapp/docker-compose.yml: the attribute 'version' is obsolete, it will be ignored, please r
emove it to avoid potential confusion
[+] Running 1/1
✔ Container myapp_container Started 0.4s
[ec2-user@ip-172-31-65-199 myapp]$
```

Check running:

\$ docker ps

```
ec2-user@ip-172-31-65-199:~, × + v
[ec2-user@ip-172-31-65-199 myapp]$ docker ps
CONTAINER ID   IMAGE      NAMES                  CREATED          STATUS              PORTS
ca119894f1c0   myapp-web  "docker-entrypoint.s..." 38 seconds ago   Up 37 seconds      3000/tcp, 0.0.0.0:8080->8080/tcp
, ::8080->8080/tcp  myapp_container
7db5339680ce   myapp      "docker-entrypoint.s..." 19 minutes ago   Up 19 minutes      0.0.0.0:3000->3000/tcp, ::3000-
>3000/tcp      busy_edison
[ec2-user@ip-172-31-65-199 myapp]$
```