# Docker Task -2

**Task Description:**

Create a dockerfile, docker-compose file which when executed must display your basic details in the website.

➔ Step 1: Make Project Directory

```
docker-compose version 1.29.2, build unknown
ubuntu@ip-172-31-42-197:~$ mkdir my-website
ubuntu@ip-172-31-42-197:~$ cd my-website/
```

➔ Step 2: Create HTML file and write or paste a code.

```
ubuntu@ip-172-31-42-197:~/my-website$ nano index.html

<!DOCTYPE html>
<html>
<head>
    <title>My Details</title>
</head>
<body style="font-family: Arial; text-align: center; margin-top: 50px;">
    <h1>Hello! 👋</h1>
    <p><b>Name:</b> Pranit Pisal</p>
    <p><b>Age:</b> 26</p>
    <p><b>Location:</b> Kolhapur, India</p>
    <p><b>Interests:</b> Cloud, DevOps, Fitness</p>
</body>
</html>
```

➔ Step 3: Create DockerFile

```
ubuntu@ip-172-31-42-197:~/my-website$ nano Dockerfile

# Use official Nginx image as base
FROM nginx:alpine

# Copy your HTML file into nginx default folder
COPY index.html /usr/share/nginx/html/index.html
```
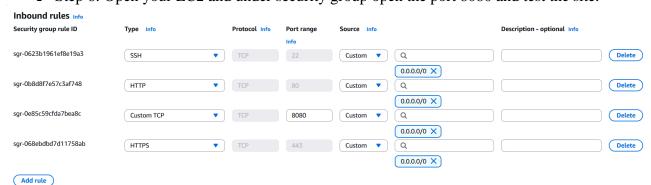
➔ Step 4: Create Docker-Compose.yml

```
ubuntu@ip-172-31-42-197:~/my-website$ nano docker-compose.yml
```

```
version: '3'
services:
  mywebsite:
    build: .
    ports:
      - "8080:80"
```

➔ Step 5: Build and run with Docker Compose.

```
ubuntu@ip-172-31-42-197:~/my-website$ sudo docker-compose up --build -d
Creating network "my-website_default" with the default driver
Building mywebsite
DEPRECATED: The legacy builder is deprecated and will be removed in a future release.
            Install the buildx component to build images with BuildKit:
            https://docs.docker.com/go/buildx/

Sending build context to Docker daemon  4.096kB
Step 1/2 : FROM nginx:alpine
alpine: Pulling from library/nginx
9824c27679d3: Pull complete
6bc572a340ec: Pull complete
403e3f251637: Pull complete
9adfbae99cb7: Pull complete
7a8a46741e18: Pull complete
c9ebe2ff2d2c: Pull complete
a992fbc61ecc: Pull complete
cb1ff4086f82: Pull complete
Digest: sha256:42a516af16b852e33b7682d5ef8acbd5d13fe08fecadc7ed98605ba5e3b26ab8
Status: Downloaded newer image for nginx:alpine
```

➔ Step 6: Open your EC2 and under security group open the port 8080 and test the site.

**Inbound rules** Info

| Security group rule ID | Type | Info | Protocol Info | Port range Info | Source | Info | | Description - optional Info | |
|---|---|---|---|---|---|---|---|---|---|
| sgr-0623b1961ef8e19a3 | SSH | ▼ | TCP | 22 | Custom ▼ | Q 0.0.0.0/0 ✕ | | | Delete |
| sgr-0b8d8f7e57c3af748 | HTTP | ▼ | TCP | 80 | Custom ▼ | Q 0.0.0.0/0 ✕ | | | Delete |
| sgr-0e85c59cfda7bea8c | Custom TCP | ▼ | TCP | 8080 | Custom ▼ | Q 0.0.0.0/0 ✕ | | | Delete |
| sgr-068ebdbd7d11758ab | HTTPS | ▼ | TCP | 443 | Custom ▼ | Q 0.0.0.0/0 ✕ | | | Delete |

Add rule

# Hello! ðŸ'‹

**Name:** Pranit Pisal

**Age:** 26

**Location:** Kolhapur, India

**Interests:** Cloud, DevOps, Fitness