

## Experiment No. 2.2

**Student Name:** Parvinder Singh

**UID:** 22MCC20043

**Branch:** MCA - CCD

**Section/Group:** 22MCD-1/ Grp B

**Semester:** III

**Subject Name:** Containerization With Docker

**Subject Code:** 22CAH-742

1. **Aim/Overview of the practical:** Building a Custom Docker Image for a Web Application

2. **Code for experiment/practical:**

To build a custom Docker image for a web application, you will need to create a Dockerfile. The Dockerfile will contain the instructions that Docker needs to build the image.

The first step is to choose a base image. The base image is the image that your custom image will be built on. For a web application, you will typically want to choose a base image that contains a web server, such as Nginx or Apache.

Here, we are creating web application image using nginx base image,.

Step1: Create an index.html and Dockerfile.

```
index.html X
index.html > html > body > p
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>Web Application Using Docker</title>
7 </head>
8 <body>
9   <h1>This is an Web Application Docker Image using Dockerfile</h1>
10  <p>
11    If you are using a more complex web application,
12    you may need to add additional instructions to your Dockerfile,
13    such as installing additional software packages or configuring the web server.<br>
14    For more information on writing Dockerfiles,
15    please see the Docker documentation: https://docs.docker.com/engine/reference/builder/
16  </p>
17 </body>
18 </html>
```

```
Dockerfile X
Dockerfile > FROM
1 FROM nginx:alpine
2 COPY ./ /usr/share/nginx/html
3 EXPOSE 8080
```

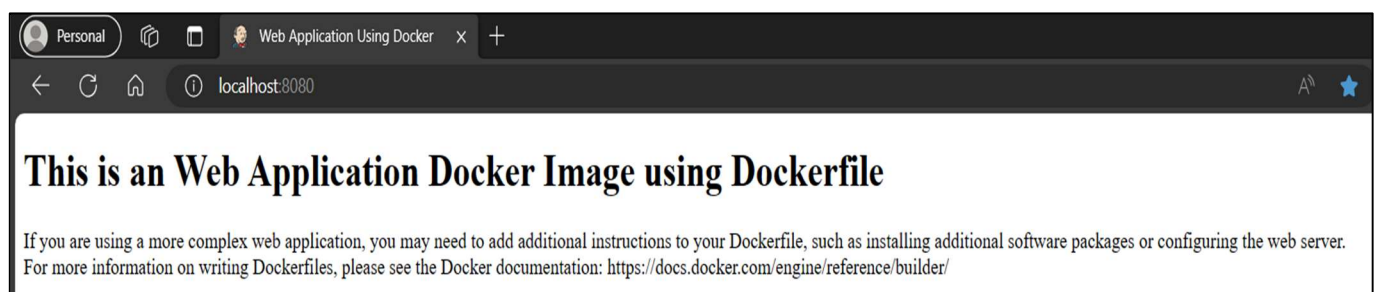
**Commands for building an Image:** \$ docker build -t web-image .

```
PS C:\Users\Pinda\Desktop\Docker\nginx> docker build -t web-image .
[+] Building 7.5s (8/8) FINISHED                                docker:default
=> [internal] load .dockerignore                                0.1s
=> => transferring context: 2B                                    0.0s
=> [internal] load build definition from Dockerfile              0.1s
=> => transferring dockerfile: 98B                                0.0s
=> [internal] load metadata for docker.io/library/nginx:alpine  7.0s
=> [auth] library/nginx:pull token for registry-1.docker.io     0.0s
=> [internal] load build context                                0.1s
=> => transferring context: 764B                                    0.0s
=> CACHED [1/2] FROM docker.io/library/nginx:alpine@sha256:4c93a3bd8bf95412889dd84213570102176b6052d88bb828eaf44 0.0s
=> [2/2] COPY ./ /usr/share/nginx/html                          0.1s
=> exporting to image                                           0.1s
=> => exporting layers                                           0.0s
=> => writing image sha256:e6c1cb774667995deb55efaa2629420180a77b9fcf8f409ea5e4f75c00fe03d9 0.0s
=> => naming to docker.io/library/web-image                      0.0s
```

**To run Image:** \$ docker run -it -p 8080:80 web-image

Note: Here we have to specify the port number for running web application.

```
PS C:\Users\Pinda\Desktop\Docker\nginx> docker run -it -p 8080:80 web-image
```



### 3. Learning outcomes (What I have learned):

- To understand Dockerfile.
- To build Docker Image from Dockerfile.
- To run an image that was created using Dockerfile.