UNIVERSITY INSTITUTE of COMPUTING
Asia's Fastest Growing University

Experiment No. 2.1

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:** Understanding the Docker file for Customizing Images

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

A Dockerfile is a text document that contains instructions for building a Docker image. It is a simple way to automate the image building process and ensure that your images are built consistently.

Dockerfiles are made up of a series of instructions, each of which is executed in order. The most common Dockerfile instructions include:

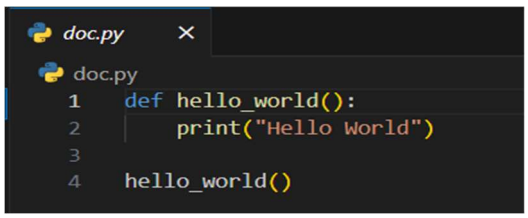
- FROM: Specifies the base image to use for building the new image.
- RUN: Executes a command in the image.
- COPY: Copies files from the host machine to the image.
- CMD: Specifies the command to run when the container is started.

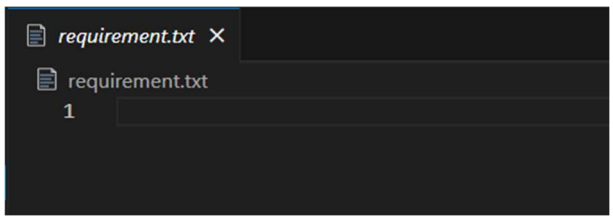
Here is an example of a simple Dockerfile for customizing an image:

Creating a simple python image

Create Three file:

- doc.py
- requirement.txt
- Dockerfile





Dockerfile

```
Dockerfile > FROM
1   FROM python:3
2
3   WORKDIR /app
4   COPY . /app
5   RUN pip install -r requirement.txt
6   EXPOSE 3000
7
8   CMD [ "python", "./doc.py" ]
```

This Dockerfile builds a new image based on the python:3 base image. It then copies the application's . /app file and source code to the image. It then installs the application's dependencies and sets the CMD instruction to start the application when the container is started.

Commands for building an Image: $ docker build -t python-image.

```
PS C:\Users\Pinda\Desktop\Docker\py> docker build -t python-image .
[+] Building 11.2s (10/10) FINISHED                                                    docker:default
 => [internal] load build definition from Dockerfile                                          0.1s
 => => transferring dockerfile: 160B                                                          0.0s
 => [internal] load .dockerignore                                                             0.0s
 => => transferring context: 2B                                                               0.0s
 => [internal] load metadata for docker.io/library/python:3                                   5.9s
 => [auth] library/python:pull token for registry-1.docker.io                                 0.0s
 => [1/4] FROM docker.io/library/python:3@sha256:1615c71b5f3d48844b8d20cac4838f34267d96c3b061dcb6e4fda61a71599a9d  0.1s
 => [internal] load build context                                                             0.0s
 => => transferring context: 88B                                                              0.0s
 => [2/4] WORKDIR /app                                                                        0.0s
 => [3/4] COPY . /app                                                                         0.0s
 => [4/4] RUN pip install -r requirement.txt                                                  4.8s
 => exporting to image                                                                        0.2s
 => => exporting layers                                                                       0.2s
 => => writing image sha256:3e72ade15438484a412480ffc72466651d3e2ba7371f6257e68037642bebc7ae  0.0s
 => => naming to docker.io/library/python-image                                               0.0s
```

To run Image: $ docker run python-image

```
PS C:\Users\Pinda\Desktop\Docker\py> docker run python-image
Hello World
```

3. Learning outcomes (What I have learned):

a) To understand Dockerfile.
b) To build Docker Image from Dockerfile.
c) To run an image that was created using Dockerfile.