Software Engineering and Project Management Lab Experiment No: - 10
Aim: To Study and Implement Dockerfile instructions

Aim: To learn Dockerfile instructions, build an image for a sample web application using DOCKERFILE.

Theory:

A container is a standard unit of software that packages up code and all its dependencies so the application runs quickly and reliably across different computing environments. Docker provides a platform for developing, shipping, and running containers easily.

Key Features of Containers:

- Isolation: Each container runs in its own isolated environment.
- Portability: Containers can run consistently across any system.
- Lightweight: Share OS kernel, making them more efficient than virtual machines.
- Scalable: Easily deploy and replicate containers across clusters.
- Fast Deployment: Start in seconds, ideal for CI/CD and agile development.
- Reproducible: Same environment from development to production.

Docker as a Container Platform

Docker enables you to create and manage containers using simple commands. It allows you to containerize applications and services with ease and deploy them in any environment.

Key Docker Container Components:

Component	Purpose
Dockerfile	Blueprint to build a Docker image
Docker Image	Snapshot of the app and its dependencies
Docker Container	Runnable instance of a Docker image
Docker CLI	Command-line tool to interact with Docker
Docker Daemon	Background process managing Docker containers
Docker Hub	Online repository to store and share images

Demonstration of Running a Container using Docker (Theoretical Steps)

1. Install Docker
   i. Download and install Docker Desktop from https://www.docker.com.
   ii. Start Docker on your local machine.
   iii. Verify installation by running:

```
docker --version
```

2. Pull a Base Image
   i. Use Docker Hub to pull a popular image (e.g., Ubuntu, Nginx, Python):

```
docker pull ubuntu
```

3. <u>Run a Container from an Image</u>
   i. Use the docker run command:

   docker run -it ubuntu

   ii. This opens an interactive terminal session in the container.
   iii. Run Linux commands inside the container (e.g., ls, pwd, apt update).

4. <u>Exit the Container</u>
   i. Type exit to close the session and stop the container.

5. <u>List Running and Stopped Containers</u>
   i. View active containers:
   docker ps

   ii. View all containers (including stopped ones):
   docker ps -a

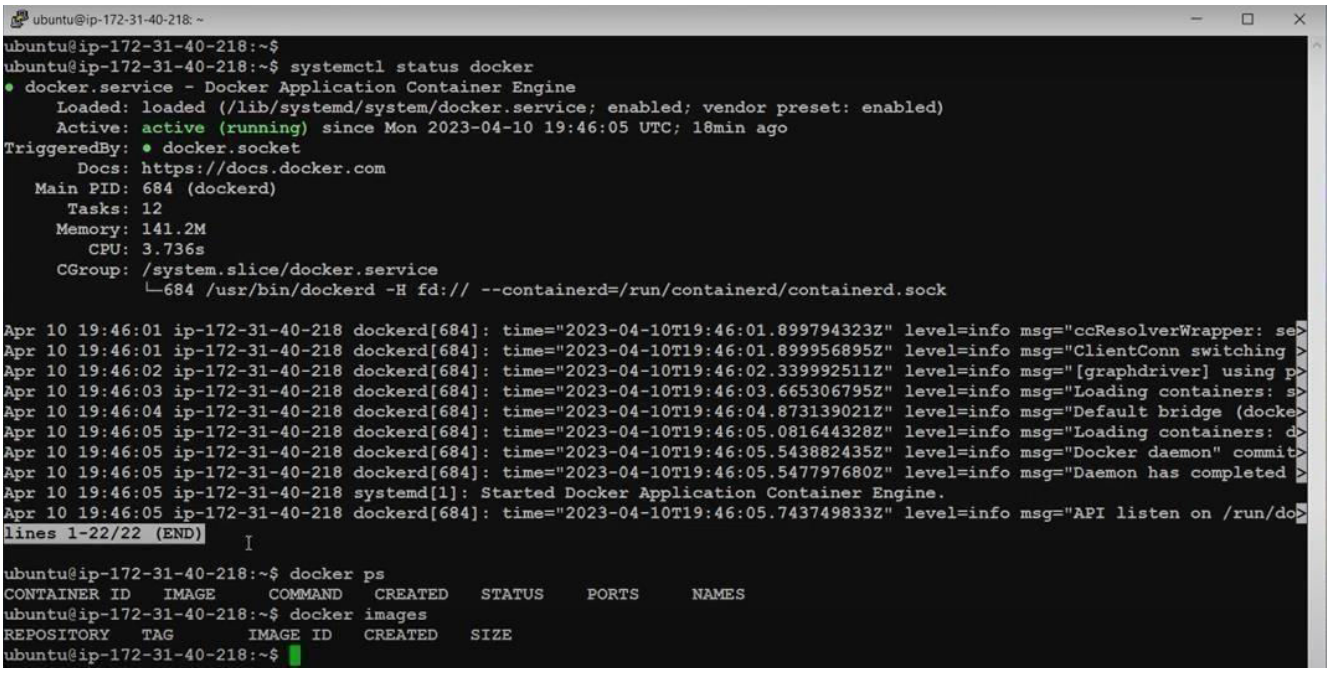
6. <u>Remove Containers</u>
   i. Stop the container (if still running):
   docker stop <container_id>
   ii. Remove the container:
   docker rm <container_id>

7. <u>Run a Web Server Container (Optional)</u>
   i. Run a web server (e.g., Nginx):
   docker run -d -p 8080:80 nginx
   ii. Open http://localhost:8080 in browser to see the Nginx welcome page.

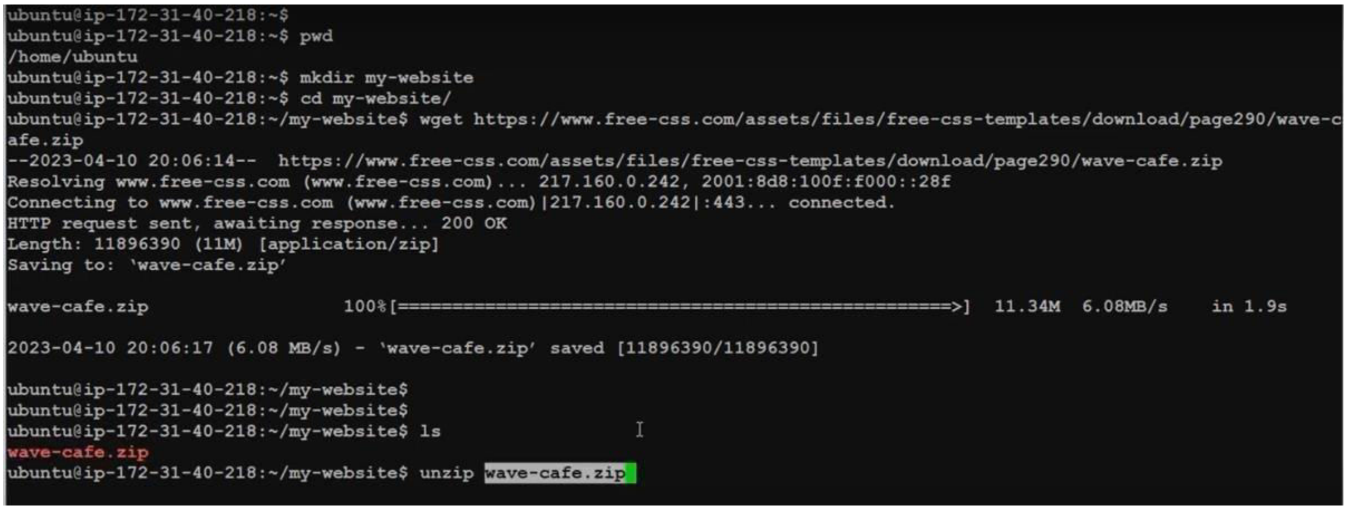
Use Case Example:

- Running a Linux environment on any OS for testing
- Hosting a web server like Apache or Nginx inside a container
- Developing and testing Python, Node.js, or Java apps
- Containerizing databases like MySQL or MongoDB for quick use
- Experimenting with new tools and languages without affecting host OS
- Education and training in DevOps or system admin practices
- Deploying a standalone app for development or demonstration

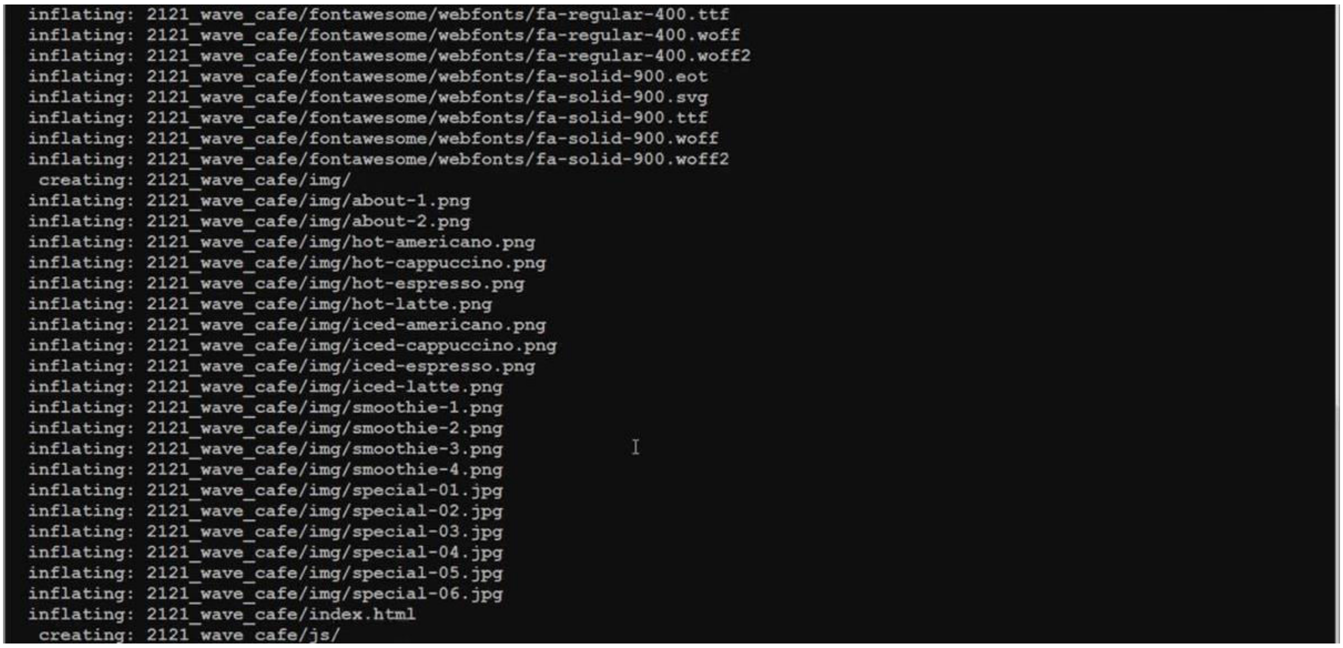
Implementation:



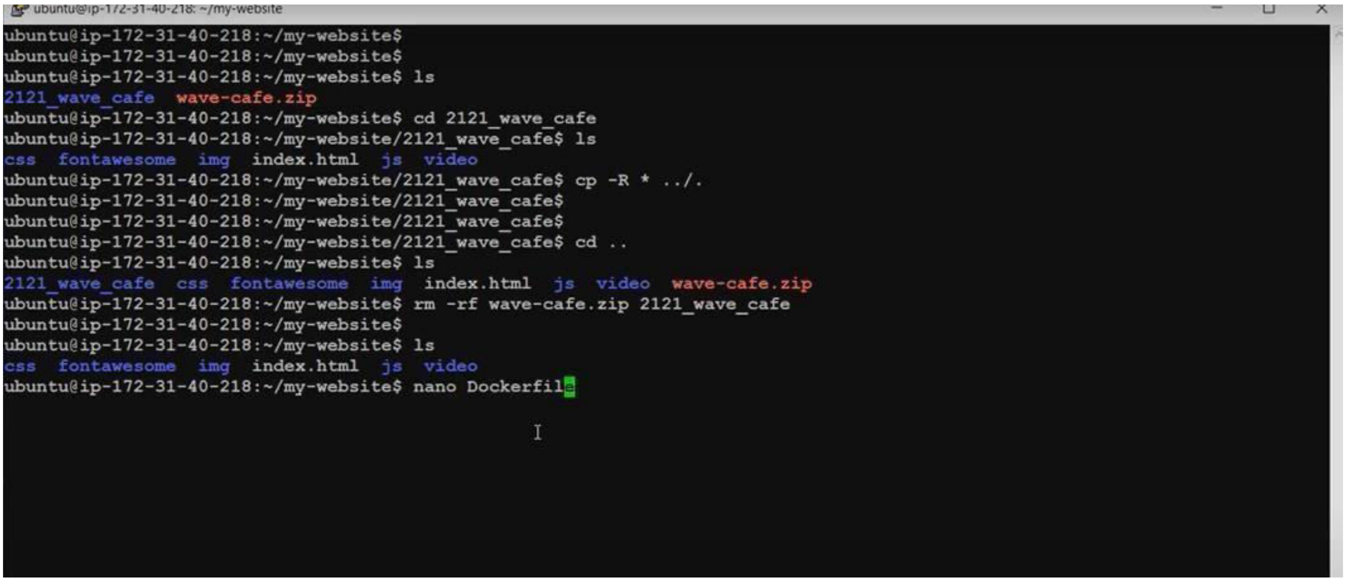
```
ubuntu@ip-172-31-40-218:~$
ubuntu@ip-172-31-40-218:~$ systemctl status docker
● docker.service - Docker Application Container Engine
     Loaded: loaded (/lib/systemd/system/docker.service; enabled; vendor preset: enabled)
     Active: active (running) since Mon 2023-04-10 19:46:05 UTC; 18min ago
TriggeredBy: ● docker.socket
       Docs: https://docs.docker.com
   Main PID: 684 (dockerd)
      Tasks: 12
     Memory: 141.2M
        CPU: 3.736s
     CGroup: /system.slice/docker.service
             └─684 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/containerd.sock

Apr 10 19:46:01 ip-172-31-40-218 dockerd[684]: time="2023-04-10T19:46:01.899794323Z" level=info msg="ccResolverWrapper: se>
Apr 10 19:46:01 ip-172-31-40-218 dockerd[684]: time="2023-04-10T19:46:01.899956895Z" level=info msg="ClientConn switching >
Apr 10 19:46:02 ip-172-31-40-218 dockerd[684]: time="2023-04-10T19:46:02.339992511Z" level=info msg="[graphdriver] using p>
Apr 10 19:46:03 ip-172-31-40-218 dockerd[684]: time="2023-04-10T19:46:03.665306795Z" level=info msg="Loading containers: s>
Apr 10 19:46:04 ip-172-31-40-218 dockerd[684]: time="2023-04-10T19:46:04.873139021Z" level=info msg="Default bridge (docke>
Apr 10 19:46:05 ip-172-31-40-218 dockerd[684]: time="2023-04-10T19:46:05.081644328Z" level=info msg="Loading containers: d>
Apr 10 19:46:05 ip-172-31-40-218 dockerd[684]: time="2023-04-10T19:46:05.543882435Z" level=info msg="Docker daemon" commit>
Apr 10 19:46:05 ip-172-31-40-218 dockerd[684]: time="2023-04-10T19:46:05.547797680Z" level=info msg="Daemon has completed >
Apr 10 19:46:05 ip-172-31-40-218 systemd[1]: Started Docker Application Container Engine.
Apr 10 19:46:05 ip-172-31-40-218 dockerd[684]: time="2023-04-10T19:46:05.743749833Z" level=info msg="API listen on /run/do>
lines 1-22/22 (END)

ubuntu@ip-172-31-40-218:~$ docker ps
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS     NAMES
ubuntu@ip-172-31-40-218:~$ docker images
REPOSITORY   TAG       IMAGE ID   CREATED   SIZE
ubuntu@ip-172-31-40-218:~$
```



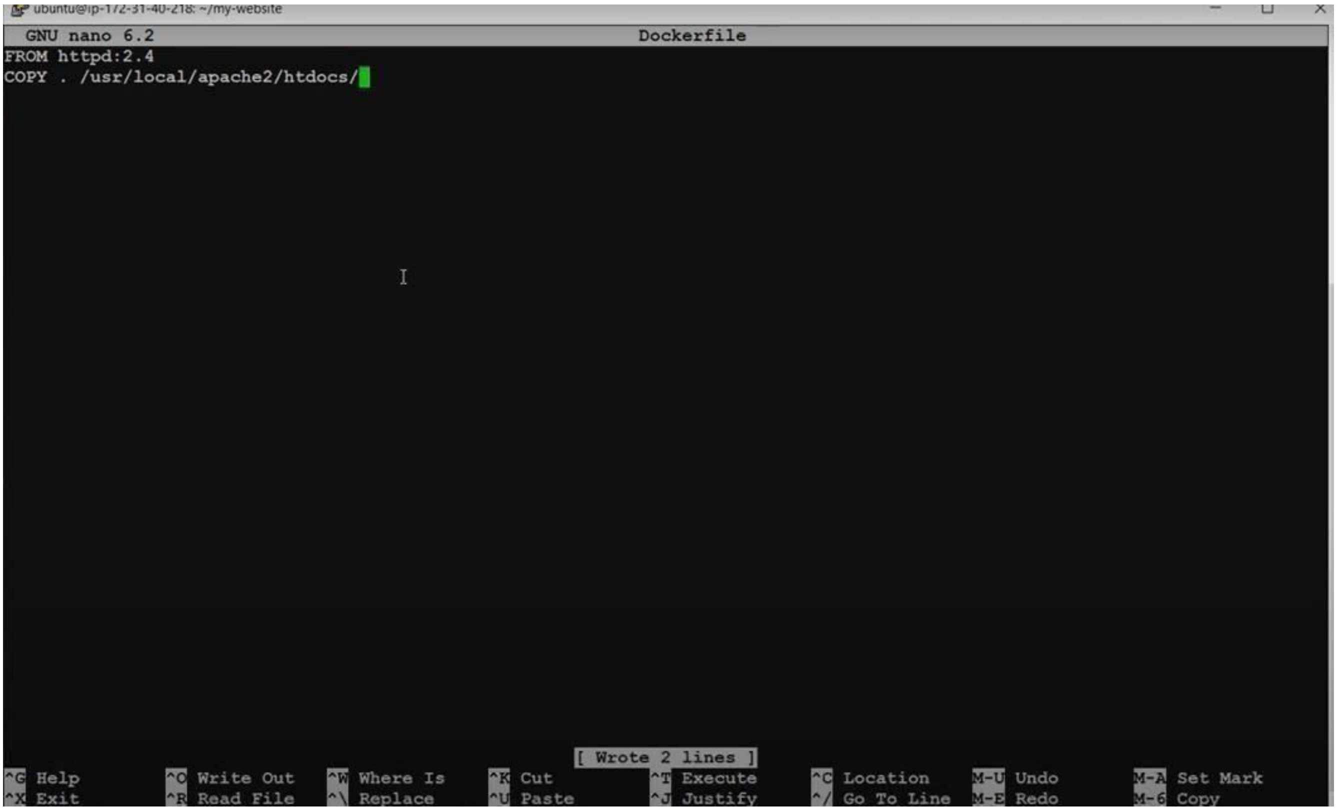
```
ubuntu@ip-172-31-40-218:~$
ubuntu@ip-172-31-40-218:~$ pwd
/home/ubuntu
ubuntu@ip-172-31-40-218:~$ mkdir my-website
ubuntu@ip-172-31-40-218:~$ cd my-website/
ubuntu@ip-172-31-40-218:~/my-website$ wget https://www.free-css.com/assets/files/free-css-templates/download/page290/wave-c
afe.zip
--2023-04-10 20:06:14--  https://www.free-css.com/assets/files/free-css-templates/download/page290/wave-cafe.zip
Resolving www.free-css.com (www.free-css.com)... 217.160.0.242, 2001:8d8:100f:f000::28f
Connecting to www.free-css.com (www.free-css.com)|217.160.0.242|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 11896390 (11M) [application/zip]
Saving to: 'wave-cafe.zip'

wave-cafe.zip                 100%[=====================================================>]  11.34M  6.08MB/s    in 1.9s

2023-04-10 20:06:17 (6.08 MB/s) - 'wave-cafe.zip' saved [11896390/11896390]

ubuntu@ip-172-31-40-218:~/my-website$
ubuntu@ip-172-31-40-218:~/my-website$
ubuntu@ip-172-31-40-218:~/my-website$ ls
wave-cafe.zip
ubuntu@ip-172-31-40-218:~/my-website$ unzip wave-cafe.zip
```



```
  inflating: 2121_wave_cafe/fontawesome/webfonts/fa-regular-400.ttf
  inflating: 2121_wave_cafe/fontawesome/webfonts/fa-regular-400.woff
  inflating: 2121_wave_cafe/fontawesome/webfonts/fa-regular-400.woff2
  inflating: 2121_wave_cafe/fontawesome/webfonts/fa-solid-900.eot
  inflating: 2121_wave_cafe/fontawesome/webfonts/fa-solid-900.svg
  inflating: 2121_wave_cafe/fontawesome/webfonts/fa-solid-900.ttf
  inflating: 2121_wave_cafe/fontawesome/webfonts/fa-solid-900.woff
  inflating: 2121_wave_cafe/fontawesome/webfonts/fa-solid-900.woff2
   creating: 2121_wave_cafe/img/
  inflating: 2121_wave_cafe/img/about-1.png
  inflating: 2121_wave_cafe/img/about-2.png
  inflating: 2121_wave_cafe/img/hot-americano.png
  inflating: 2121_wave_cafe/img/hot-cappuccino.png
  inflating: 2121_wave_cafe/img/hot-espresso.png
  inflating: 2121_wave_cafe/img/hot-latte.png
  inflating: 2121_wave_cafe/img/iced-americano.png
  inflating: 2121_wave_cafe/img/iced-cappuccino.png
  inflating: 2121_wave_cafe/img/iced-espresso.png
  inflating: 2121_wave_cafe/img/iced-latte.png
  inflating: 2121_wave_cafe/img/smoothie-1.png
  inflating: 2121_wave_cafe/img/smoothie-2.png
  inflating: 2121_wave_cafe/img/smoothie-3.png
  inflating: 2121_wave_cafe/img/smoothie-4.png
  inflating: 2121_wave_cafe/img/special-01.jpg
  inflating: 2121_wave_cafe/img/special-02.jpg
  inflating: 2121_wave_cafe/img/special-03.jpg
  inflating: 2121_wave_cafe/img/special-04.jpg
  inflating: 2121_wave_cafe/img/special-05.jpg
  inflating: 2121_wave_cafe/img/special-06.jpg
  inflating: 2121_wave_cafe/index.html
   creating: 2121_wave_cafe/js/
```

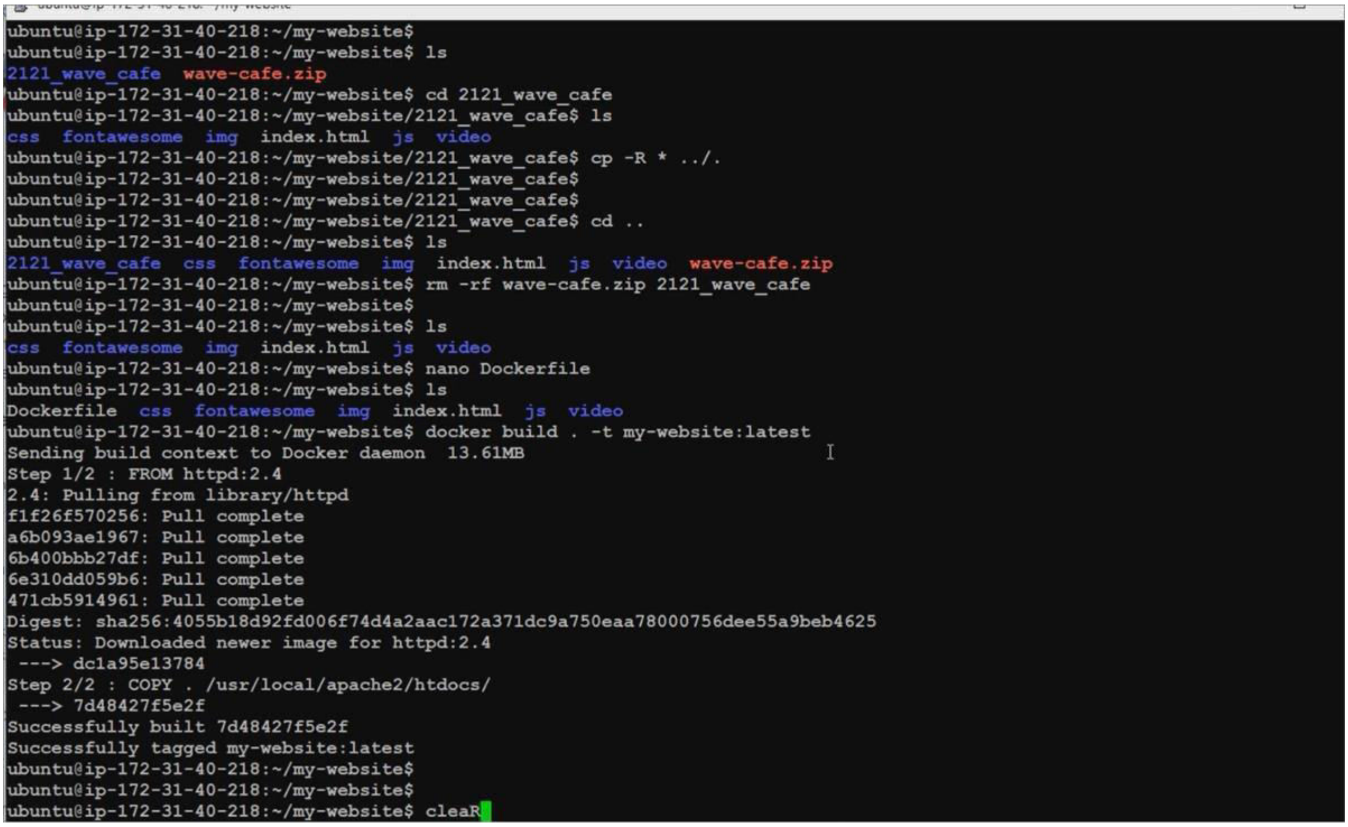


```
ubuntu@ip-172-31-40-218:~/my-website$
ubuntu@ip-172-31-40-218:~/my-website$
ubuntu@ip-172-31-40-218:~/my-website$ ls
2121_wave_cafe  wave-cafe.zip
ubuntu@ip-172-31-40-218:~/my-website$ cd 2121_wave_cafe
ubuntu@ip-172-31-40-218:~/my-website/2121_wave_cafe$ ls
css  fontawesome  img  index.html  js  video
ubuntu@ip-172-31-40-218:~/my-website/2121_wave_cafe$ cp -R * ../.
ubuntu@ip-172-31-40-218:~/my-website/2121_wave_cafe$
ubuntu@ip-172-31-40-218:~/my-website/2121_wave_cafe$
ubuntu@ip-172-31-40-218:~/my-website/2121_wave_cafe$ cd ..
ubuntu@ip-172-31-40-218:~/my-website$ ls
2121_wave_cafe  css  fontawesome  img  index.html  js  video  wave-cafe.zip
ubuntu@ip-172-31-40-218:~/my-website$ rm -rf wave-cafe.zip 2121_wave_cafe
ubuntu@ip-172-31-40-218:~/my-website$
ubuntu@ip-172-31-40-218:~/my-website$ ls
css  fontawesome  img  index.html  js  video
ubuntu@ip-172-31-40-218:~/my-website$ nano Dockerfile
```

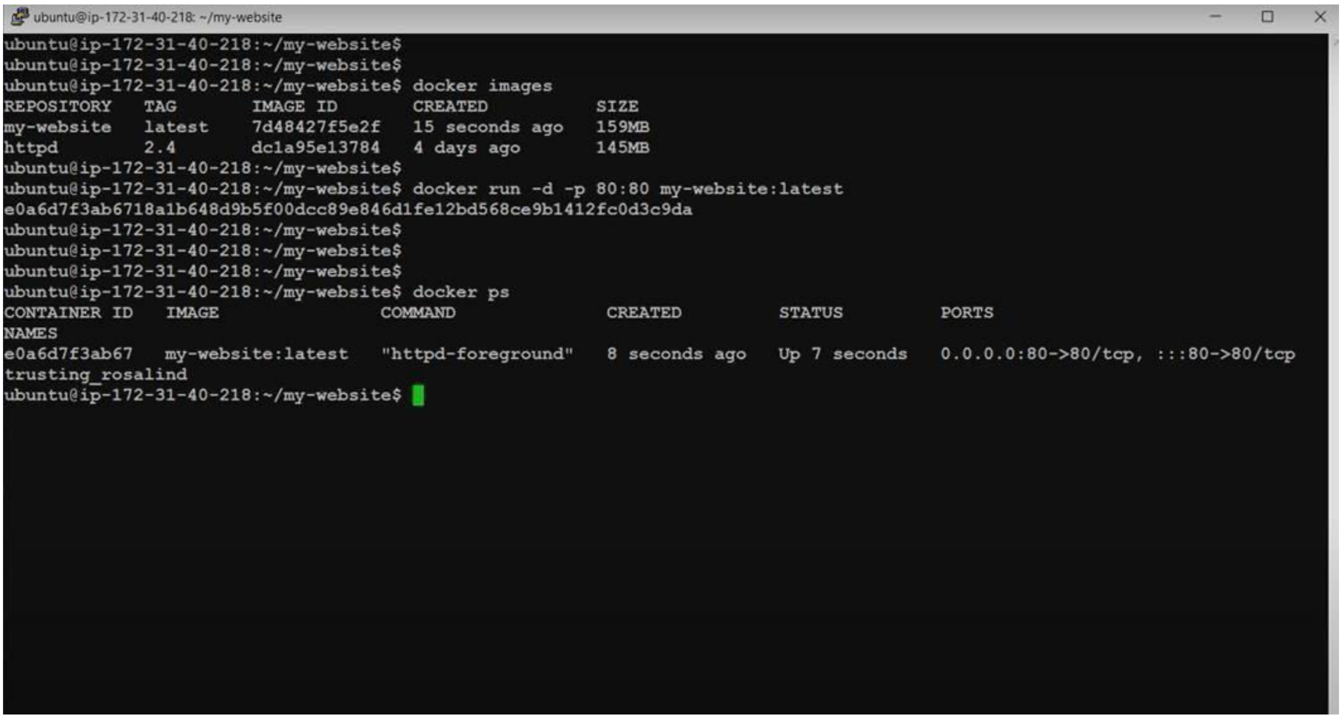


```
FROM httpd:2.4
COPY . /usr/local/apache2/htdocs/
```

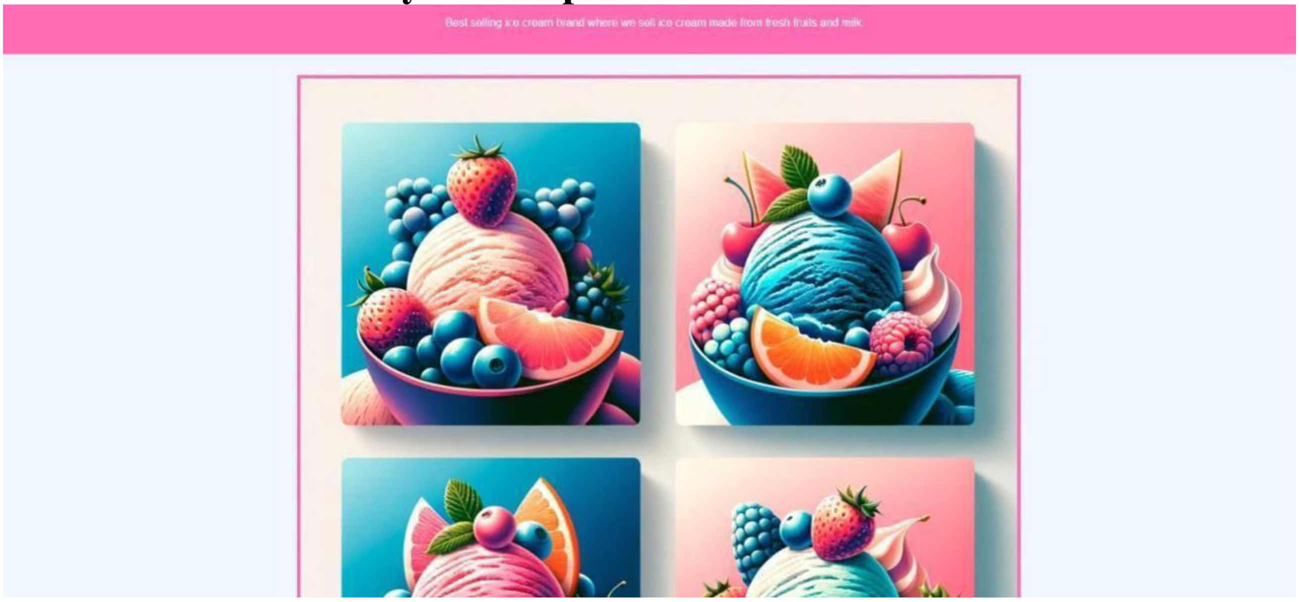
```
[ Wrote 2 lines ]
```



```
ubuntu@ip-172-31-40-218:~/my-website$
ubuntu@ip-172-31-40-218:~/my-website$ ls
2121_wave_cafe  wave-cafe.zip
ubuntu@ip-172-31-40-218:~/my-website$ cd 2121_wave_cafe
ubuntu@ip-172-31-40-218:~/my-website/2121_wave_cafe$ ls
css  fontawesome  img  index.html  js  video
ubuntu@ip-172-31-40-218:~/my-website/2121_wave_cafe$ cp -R * ../.
ubuntu@ip-172-31-40-218:~/my-website/2121_wave_cafe$
ubuntu@ip-172-31-40-218:~/my-website/2121_wave_cafe$
ubuntu@ip-172-31-40-218:~/my-website/2121_wave_cafe$ cd ..
ubuntu@ip-172-31-40-218:~/my-website$ ls
2121_wave_cafe  css  fontawesome  img  index.html  js  video  wave-cafe.zip
ubuntu@ip-172-31-40-218:~/my-website$ rm -rf wave-cafe.zip 2121_wave_cafe
ubuntu@ip-172-31-40-218:~/my-website$
ubuntu@ip-172-31-40-218:~/my-website$ ls
css  fontawesome  img  index.html  js  video
ubuntu@ip-172-31-40-218:~/my-website$ nano Dockerfile
ubuntu@ip-172-31-40-218:~/my-website$ ls
Dockerfile  css  fontawesome  img  index.html  js  video
ubuntu@ip-172-31-40-218:~/my-website$ docker build . -t my-website:latest
Sending build context to Docker daemon  13.61MB
Step 1/2 : FROM httpd:2.4
2.4: Pulling from library/httpd
f1f26f570256: Pull complete
a6b093ae1967: Pull complete
6b400bbb27df: Pull complete
6e310dd059b6: Pull complete
471cb5914961: Pull complete
Digest: sha256:4055b18d92fd006f74d4a2aac172a371dc9a750eaa78000756dee55a9beb4625
Status: Downloaded newer image for httpd:2.4
 ---> dc1a95e13784
Step 2/2 : COPY . /usr/local/apache2/htdocs/
 ---> 7d48427f5e2f
Successfully built 7d48427f5e2f
Successfully tagged my-website:latest
ubuntu@ip-172-31-40-218:~/my-website$
ubuntu@ip-172-31-40-218:~/my-website$
ubuntu@ip-172-31-40-218:~/my-website$ cleaR
```



```
ubuntu@ip-172-31-40-218:~/my-website$
ubuntu@ip-172-31-40-218:~/my-website$
ubuntu@ip-172-31-40-218:~/my-website$ docker images
REPOSITORY   TAG       IMAGE ID       CREATED          SIZE
my-website   latest    7d48427f5e2f   15 seconds ago   159MB
httpd        2.4       dc1a95e13784   4 days ago       145MB
ubuntu@ip-172-31-40-218:~/my-website$
ubuntu@ip-172-31-40-218:~/my-website$ docker run -d -p 80:80 my-website:latest
e0a6d7f3ab6718a1b648d9b5f00dcc89e846d1fe12bd568ce9b1412fc0d3c9da
ubuntu@ip-172-31-40-218:~/my-website$
ubuntu@ip-172-31-40-218:~/my-website$
ubuntu@ip-172-31-40-218:~/my-website$
ubuntu@ip-172-31-40-218:~/my-website$ docker ps
CONTAINER ID   IMAGE                COMMAND              CREATED         STATUS         PORTS
NAMES
e0a6d7f3ab67   my-website:latest   "httpd-foreground"   8 seconds ago   Up 7 seconds   0.0.0.0:80->80/tcp, :::80->80/tcp
trusting_rosalind
ubuntu@ip-172-31-40-218:~/my-website$
```



Best selling ice cream brand where we sell ice cream made from fresh fruits and milk.

Conclusion: We have successfully understood Dockerfile instructions, build an image for a sample web application using DOCKERFILE.

LO Mapping: *LO is mapped*