**Experiment No. 9 – Docker Image Creation Using Dockerfile**

**🎯 Aim:**

To build an image for a sample web application from a Dockerfile using various Dockerfile instructions.

**⚙️ PART I — Containerize an Application Using Docker CLI Commands**

**Step 1: Pull the nginx Image**

docker images

docker pull nginx

**Step 2: Run the Container from nginx Image**

docker run --name webserver1 nginx

Check running containers:

docker ps -a

To exit container:

Ctrl + C

Recheck:

docker ps -a

**Step 3: Run nginx with Port Mapping**

docker run -it -p 3031:80 --name server1 nginx:latest

Check containers again:

docker ps -a

**Step 4: Create a Static Website inside the Container**

Go inside nginx html folder:

cd /usr/share/nginx/html/

ls

Rename default file:

mv index.html index.html\_backup

Install nano (if not available):

apt update

apt install nano

Create new file:

nano index.html

Paste the sample HTML login page content inside and save.

**Step 5: Check nginx Service and Browser Output**

Start nginx service:

service nginx start

Go to browser → [http://localhost:3031](http://localhost:3031/)  
Your static website should be displayed.

**Step 6: Pause, Unpause, and Remove Container**

docker pause <container\_id>

docker unpause <container\_id>

docker rm <container\_id>

Verify removal:

docker ps -a

**⚙️ PART II — Building Custom Image Using Dockerfile**

**Step 1: Create a Dockerfile**

Create a file named Dockerfile in your project directory.  
Add basic instructions:

FROM nginx:latest

COPY . /usr/share/nginx/html

**Step 2: Build Docker Image**

docker build -t yourdockerhubusername/websitetest .

Example:

docker build -t sujatadocker2024/websitetest .

**Step 3: Run the Container**

docker run -d -p 3032:80 sujatadocker2024/websitetest

**Step 4: Verify in Browser**

Open: [http://localhost:3032](http://localhost:3032/)

You should see your web page running from the new Docker image.

**☁️ Step 5: Push Image to DockerHub**

1. Login to DockerHub:
2. docker login
3. Tag the image:
4. docker tag sujatadocker2024/websitetest yourdockerhubusername/websitetest
5. Push the image:
6. docker push yourdockerhubusername/websitetest

Check it on your DockerHub repository page.