Experiment 8.1 – Creating a Dockerfile

Aim:
To create a Dockerfile in the root of a project to build a custom Docker image and understand the process of containerizing an application using Docker.

Tools / Technologies Used:

- Docker Engine
- Docker CLI (Command Line Interface)
- VS Code / Any Code Editor
- Base Operating System: Windows / Linux / macOS
- Programming language in project (example: Node.js / Python / Java)

Theory (Short Explanation):
A Dockerfile is a text file containing a set of instructions used to create a Docker image. It defines everything required to run an application, including base image, dependencies, commands, and environment settings. When we run *docker build*, Docker reads this file and creates an image.

Procedure:

1. Install Docker Desktop on your system.
2. Create or open an existing project folder.
3. In the root directory of the project, create a new file named **Dockerfile**.
4. Add the required Dockerfile instructions (FROM, WORKDIR, COPY, RUN, CMD, etc.).
5. Open the terminal and navigate to the project folder.
6. Build the Docker image using the command:
   docker build -t my-app-image .
7. After successful build, run the container using:
   docker run -p 3000:3000 my-app-image
8. Verify that your application is running in the container.

Sample Dockerfile Code (Node.js Example):

```
# Step 1: Use an official Node.js base image
FROM node:18-alpine

# Step 2: Set working directory inside container
WORKDIR /app

# Step 3: Copy package.json and install dependencies
COPY package*.json ./
RUN npm install

# Step 4: Copy the rest of the source code
COPY . .

# Step 5: Expose the application port
EXPOSE 3000

# Step 6: Start the application
CMD ["npm", "start"]
```

Output:

- Successful creation of Docker image.
- Container runs the application successfully.
- Application accessible via browser at: *http://localhost:3000*

Sample Terminal Output:

```
Successfully built 2d3a4fe91cc8
Successfully tagged my-app-image:latest
```

Learning Outcomes:

After performing this experiment, the student will be able to:

- Understand what a Dockerfile is and how it works
- Write Dockerfile instructions to containerize an application
- Build a Docker image using Dockerfile
- Run and test application inside a Docker container
- Understand containerization workflow for deployments