Steps to Create a Docker Image

1. Create Your Application Files

Prepare your application source code and related files in a project folder. For example:

```
/my-app
  ├── app.py
  ├── requirements.txt
  └── Dockerfile
```

2. Write a Dockerfile

The `Dockerfile` is a script that defines how the Docker image is built.

Example Dockerfile for a Python App:

- # Use an official Python base image
  FROM python:3.10-slim
- # Set the working directory
  WORKDIR /app
- # Copy project files into the container
  COPY . /app
- # Install dependencies
  RUN pip install -r requirements.txt
- # Expose port (optional, depends on your app)
  EXPOSE 5000
- # Command to run the app
  CMD ["python", "app.py"]

3. Build the Docker Image

Open terminal, navigate to the project folder, and run:

```
docker build -t my-app-image .
```

`-t` is used to name the image (`my-app-image`)

`.` refers to the current directory where the Dockerfile is located

4. Verify the Image

After the build is complete, verify the image is created:

```
docker images
```

5. Run a Container from the Image

To test the image by running it in a container:

```
docker run -d -p 5000:5000 --name my-app-container my-app-image
```

`-d` runs in detached mode

`-p` maps container port to host port

`--name` gives a name to the running container

6. Stop and Remove Container

- To stop:
  `docker stop my-app-container`
- To remove:
  `docker rm my-app-container`

7. Push Image to Docker Hub (Optional)

If you want to share your image:

- `docker tag my-app-image your-dockerhub-username/my-app-image`
- `docker push your-dockerhub-username/my-app-image`