**Experiment No : 6**

Title: Running php application over docker.

Aim : To run php application over docker.

Theory:

## What is Docker?

Docker is a software platform. It enables software developers to develop, ship and run applications within its containers. Containers are lightweight software applications. We are going to build a Docker image in this experiment.

## What is a docker file, image and container?

A docker file is a text file that contains the set of instructions for the Docker platform. Therefore, it can be versioned and committed to a code repository.

An image includes everything needed to run an application — the code or binary, runtime, dependencies, and any other file system objects required.

Docker containers run the application code.

we will create a simple php program, then we will build a Docker image for that application, and lastly we will instantiate a container from that image.

Docker allows you to package an application with its environment and all of its dependencies into a "box", called a container. Usually, a container consists of an application running in a stripped-to-basics version of a Linux operating system. An image is the blueprint for a container, a container is a running instance of an image.

**What is PHP ?**

PHP, which stands for "Hypertext Preprocessor," is a popular and widely used server-side scripting language primarily designed for web development. It is an open-source, general-purpose scripting language that is especially well-suited for creating dynamic web pages and web applications.

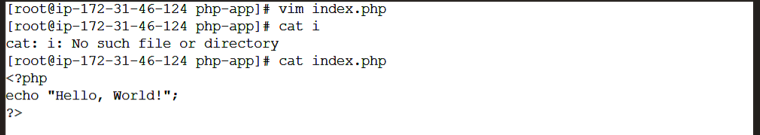
**Running PHP applications :**

Running PHP applications in Docker is a convenient way to develop and deploy PHP applications in a consistent and isolated environment. Here are the key steps for running PHP in Docker:

Create a PHP Application:

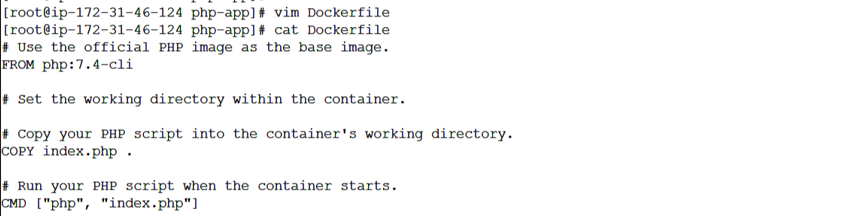
First, create a PHP application or script that you want to run in a Docker container. You can start with a simple "Hello, World!" PHP script or a more complex PHP application.





Dockerfile:

Create a Dockerfile in the same directory as your PHP application. This file defines the image, environment, and commands to execute. Here's an example of a basic Dockerfile:



Build the Docker Image:

Use the docker build command to build the Docker image from your Dockerfile. Navigate to the directory with your Dockerfile and run:

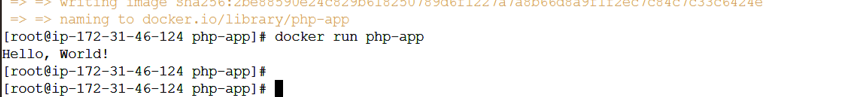
docker build -t my-php-app .



Run a Docker Container:

After building the image, you can create and run a Docker container based on the image:

docker run my-php-app



This command will start a container, execute your PHP script, and display the output in the terminal.

Conclusion: Thus we have run the node.js application on docker