**Name: Rohit Harawade**

**Roll No.: A09**

**Id: TU4F2122009**

EXPERIMENT NO: 8

# Aim: To perform Docker commands with push and pull commands.

# Theory:

# Docker is a popular platform for developing, shipping, and running applications inside containers. Containers are lightweight and portable, making it easy to move applications between different environments. Docker uses images as the building blocks for containers. Images are a snapshot of a file system that contains everything needed to run a container, including the application code, libraries, and dependencies.

# Docker is a powerful tool for containerization and offers a wide range of commands to help you manage containers and images. Here are some different Docker commands and a brief description of each:

# docker run: This command is used to create and run a Docker container based on a specified Docker image. You can specify various options, environment variables, and other configurations when using this command.

# Syntax:

# docker run -it --name my-container ubuntu:latest

# docker ps: Lists the currently running containers. By default, it shows only running containers, but you can use options like -a to display all containers, including stopped ones.

# Syntax:

# docker ps

# docker images: Displays a list of locally available Docker images on your machine. It provides information about image names, tags, sizes, and more.

# Syntax:

# docker images

# docker build: This command is used to build a Docker image from a Dockerfile. You typically navigate to the directory containing the Dockerfile to execute this command.

# Syntax:

# docker build -t my-custom-image .

# docker stop: Stops a running container. You can specify either the container's ID or its name.

# Syntax:

# docker stop my-container

# docker exec: Executes a command in a running container. This command is useful for running commands within the context of a container.

# Syntax:

# docker exec -it my-container bash

# docker network: Manages Docker networks, allowing containers to communicate with each other. You can create, list, and manage Docker networks using this command.

# Syntax:

# docker network create my-network

# Docker Push:

# The docker push command allows you to upload Docker images from your local machine to a container registry, such as Docker Hub, Google Container Registry, or a private registry. It's an essential step if you want to share your Docker images with others or deploy them on remote servers. The basic syntax for pushing an image is as follows:

# docker push <image\_name>

# 

# Docker Pull:

# The docker pull command is used to download Docker images from a container registry to your local machine. It's the counterpart to docker push. You can use this command to fetch images from public or private registries and use them for local development or deployment. The basic syntax for pulling an image is as follows:

# docker pull <image\_name>

# 

# Conclusion: In this experiment, we performed docker commands with push and pull commands.