DEVOPS FOR AI

JENKINS INSTALLATION AND SETUP

**NAME : PRISCA.S**

**USN : 23BTRCL068**

**BRANCH : AIML -C**

**1. Install Docker**

**For Windows:**

* **Download Docker Desktop**: Go to the Docker website and download Docker Desktop for Windows.
* **Run the Installer**: Once downloaded, open the installer file. Follow the prompts to complete the installation. You may need to restart your computer.
* **Enable WSL 2**: If prompted, Docker may require you to enable Windows Subsystem for Linux (WSL) 2. Follow the instructions provided to do so.

**2. Start Docker**

* **Open Docker**: After installation, start Docker by finding it in your applications (Windows and macOS) or starting the service (Linux).
* **Wait for Initialization**: Give Docker a moment to initialize. You’ll usually see the Docker icon in your system tray indicating that it's running.

**3. Explore Docker**

* **Open Docker Dashboard**: Once Docker is running, you can access the Docker Dashboard (for Windows and macOS) by clicking the Docker icon in your system tray. This dashboard provides a user-friendly interface to manage your containers and images.
* **Understand Docker Components**:
  + **Images**: These are templates for creating containers. Think of them like a blueprint or recipe.
  + **Containers**: These are instances of images running in isolation. You can think of them like rooms built from the blueprint.
  + **Volumes**: These are used for persistent storage, allowing you to keep data even if the container is removed.
* **Search for Images**: Use the search feature in the Docker Dashboard to find official images for popular applications like databases, web servers, and programming languages.
* **Pull an Image**: From the dashboard, you can pull an image to your local machine. For example, search for "nginx" to get a web server image.
* **Create a Container**: Once you have an image, you can create a container from it. The dashboard will guide you through selecting an image and configuring settings like ports and volume mappings.
* **Run and Manage Containers**: After creating a container, you can start it, stop it, or remove it directly from the dashboard. You can also view logs and statistics related to each container.
* **Example**: commands

dockerfile

# Use an official Python runtime as a base image

FROM python:3.8-slim

# Set the working directory

WORKDIR/app

# Copy the requirements file and install dependencies

COPY requirements.txt.

RUN pip install --no-cache-dir-r requirements.txt

# Copy the project files into the container COPY..

# Run the application CMD ["python", "main.py"]

3. Building and Running the Container

Build the image:

docker build-t your-app-name.

Run the container:

docker run -d-p 8000:8000 your-app-name

This will start the container, exposing it on port 8000.

**Summary**

You have now installed Docker and explored its basic components, including images and containers. You can use the Docker Dashboard to manage your containers easily.