**Docker-compose**

**DOCKER COMPOSE**

Docker Compose is a tool that allows you to define and manage multi-container Docker applications. It simplifies the process of running multiple containers, their configurations, and their interdependencies. Compose uses a YAML file to define the services, networks, and volumes required for your application.

* Docker Compose is a tool which is used to manage multi container-based applications.
* Using Docker Compose we can easily setup & deploy multi container-based applications.
* We will give containers information to Docker Compose using YML file (docker-compose.yml)
* Docker Compose YML should have all the information related to containers creation.
* Docker Compose YML File Looks Like:

sudo apt install docker-compose

docker-compose images

docker-compose up -d

docker exec -it arunesh\_db\_1 /bin/bash

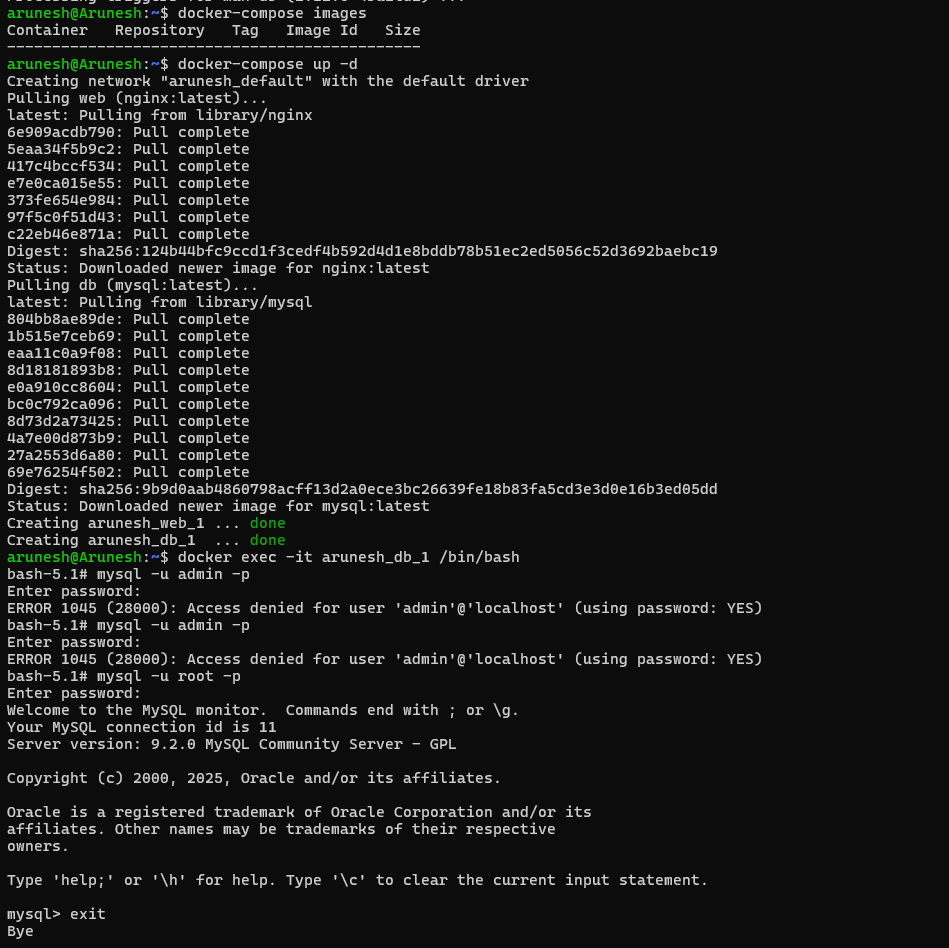
sudo nano docker-compose.yaml

mysql -u root -p

exit

docker-compose ps

docker-compose down





**DOCKER COMPOSE:**

sudo nano docker-compose.yaml

version: '3'

services:

web:

image: nginx:latest

ports:

- 80:80

db:

image: mysql:latest

environment:

- MYSQL\_ROOT\_PASSWORD=secret

sudo apt install docker-compose -y

docker-compose up -d

docker exec -it arunesh-db-1 /bin/bash

**Jenkins**

Jenkins is an open-source automation tool written in Java programming language that allows continuous integration. Jenkins offers a straightforward way to set up a continuous integration or continuous delivery environment for almost any combination of languages and source code repositories using pipelines, as well as automating other routine development tasks.

The following are the main or most popular Jenkins use cases:

● Continuous Integration: With Jenkins pipelines, we can achieve CI for both applications and infrastructure as code.

● Continuous Delivery: You can set up well-defined and automated application delivery workflows with Jenkins pipelines.

Jenkins achieves CI (Continuous Integration) and CD (Continuous Deployment) with the help of plugins. Plugins are used to allow the integration of various DevOps stages. If you want to integrate a particular tool, you must install the plugins for that tool.

**ADVANTAGES OF JENKINS:**

● It is an open-source tool.

● It is free of cost.

● It does not require additional installations or components. Means it is easy to install.

● Easily configurable.

● It supports 1000 or more plugins to ease your work. If a plugin does not exist, you can write the script for it and share with community.

● It is built in java and hence it is portable.

● It is platform independent. It is available for all platforms and different operating systems. Like OS X, Windows, or Linux.

● Easy support since its open source and widely used.

● Jenkins also supports cloud-based architecture so that we can deploy Jenkins in cloud-based platforms.