Docker Compose:

It’s a tool for defining & running multi-container docker applications.

Docker Compose working steps:

1. Install docker compose (by default installed on windows and mac with docker)

docker-compose -v

There are two ways to install docker-compose

i. https://github.com/docker/compose/releases

ii. Using PIP => pip install docker-compose

2. Create docker compose file at any location on your system - docker-compose.yml

3. Check the validaty of file by command - docker-compose config

4. Run docker-compose.yml file by command - docker-compose up -d

5. Bring down application by command - docker compose down

6. You can remove the container with – docker compose rm

Advantages:

1. Used yaml files to configure application’s services

**Docker-compose.yml**

2. Start, stop and rebuild services

3. Can start all services with a single command

**Docker compose up**

2. View the status of running services

3. Stream the log output of running services

4. Run a one-off command on a service

5. Can stop all services with a single command

**docker compose down**

6. Can scale up selected services when required

Implementation:

To implement compose, it consists the following steps.

1. Put Application environment variables inside the Dockerfile to access publicly.

2. Provide services name in the docker-compose.yml file so that they can be run together in an isolated environment.

3. run docker-compose up and compose will start and run your entire app.

A typical docker-compose.yml file has the following format and arguments.

// docker-compose.yml

version: '3'

services:

web1:

build: .

ports:

- "5000:5000"

volumes:

- .:/code

- ./myproject:/var/www/html

- logvolume01:/var/log

links:

- redis

- php

redis:

image: redis

volumes:

logvolume01: {}

php:

image: php

volumes:

myproject: {}