Course 3 – Sprint 4: Docker Compose

Docker Compose

Compose is a tool for defining and running multi-container Docker applications. With compose, you use a YAML file to configure your application's services. Then, with a single command, you create and start all the services from your configuration.

- It is a utility tool for defining & running multi-container docker applications.
- We use YAML files to configure application services (docker-compose.yml).
- We can start all services with a single command: docker compose up.
- We can stop all services with a single command: docker compose down.
- We can scale up selected services when required.

Features

The features of Compose that make it effective are:

- Multiple isolated environments on a single host
- Preserve volume data when containers are created
- Only recreate containers that have changed
- Variables and moving a composition between environments

Please click on each link to know more about them.

Using Compose is basically a three-step process:

1. Define your app's environment with a Dockerfile so it can be reproduced anywhere.
2. Define the services that make up your app in docker-compose.yml so they can be run together in an isolated environment.
3. Run docker-compose up and the Docker compose command starts and runs your entire app. You can alternatively run docker-compose up using the docker-compose binary.

Docker Compose Demo:

Step 1 : Install docker compose.

Step 2 : Create docker compose file at any location on your system.

docker-compose.yml

```
version: '3'
services:
  web:
    image: nginx
  database:
    image: redis
```

Step 3: Check the validity of file by giving the command:

docker-compose config

docker network ls

If there is no "root_default" network listed, execute the below step to create this network

step 4: Run docker-compose.yml file by giving the command:

docker-compose up -d

$docker ps

$docker inspect network root_default

so docker-compose internally creates a custom bridge network.

Steps 5: Bring down application by giving the command:

$docker-compose down

$docker inspect network root-default