**DOCKER**

1. **Initial GCP Setup –** 
   1. Create one free trial account on GCP and Create a project in GCP.
   2. Now create a VM instance and SSH into that VM.
   3. Upload jar file in VM

**2. Setup Docker VM –**

2.1 Before installing docker, we need to add couple of dependencies.

sudo apt-get -y install apt-transport-https ca-certificates curl software-properties-common

2.2 Before installing Docker repo, we need to add a gpg key from docker.com for authentication reasons

curl -fsSL https://download.docker.com/linux/debian/gpg | sudo apt-key add -

2.3 Now, we can finally add Docker repository

sudo add-apt-repository "deb [arch=amd64] https://download.docker.com/linux/debian $(lsb\_release -cs) stable"

* 1. Now, just to run a simple update and to install docker-ce

sudo apt-get update

sudo apt-get install docker-ce

* 1. To eliminate the need to write sudo at every command, add your current user to docker group

sudo usermod -aG docker $USER

* 1. Docker is successfully installed and now to install docker-compose follow the below commands

sudo curl -L "https://github.com/docker/compose/releases/download/1.22.0/docker-compose-$(uname -s)-$(uname -m)" -o /usr/local/bin/docker-compose

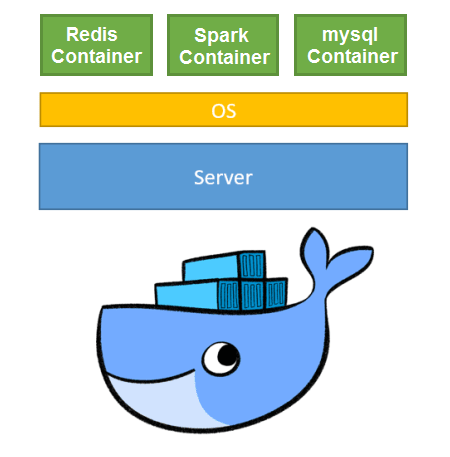
sudo chmod +x /usr/local/bin/docker-compose

2.7 Now, docker and docker-compose is installed on your VM and to check if they are successfully installed check the version with docker –v for docker version and docker-compose –v for docker-compose version.

**3. Creating container** **–**

3.1 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.

Following is the structure that is implemented :



3.2 To create container upload docker-compose.yml file to your VM and to run the docker compose execute the following command

docker-compose up –d

3.3 Now all the containers mentioned in the docker compose files are up and running and to run the spark job on spark container open the terminal of spark container

docker exec –it <sparkContainer> bash

cd ~/../sparkJob

spark-submit –class test <jarFile>