\*\*\*Docker Creation Commands\*\*\*

------------------------------

1. yum install docker -y ---> Install docker in Linux

2. service docker status🡪 To check the docker status

3. service docker start🡪 To start the docker

4. docker –version --. To check the docker version

5. docker images 🡪 To check the docker images

6. vi Dockerfile🡪 To write the docker script (should be in root)

Dockerfile🡪 it’s a default file

Sample script:

FROM ubuntu

ENV TZ 'Europe/Tallinn'

RUN echo $TZ > /etc/timezone && \

apt-get update && apt-get install -y tzdata && \

rm /etc/localtime && \

ln -snf /usr/share/zoneinfo/$TZ /etc/localtime && \

dpkg-reconfigure -f noninteractive tzdata && \

apt-get clean

RUN apt-get update && apt install apache2 -y

COPY index.html /var/www/html/index.html

7. After Docker file, create any file in it (should be in root)

Eg: vi index.html

<h1> Hello welcome to Docker world</h1>

8. docker build -t demo1 . -->To build the docker image

9. needs to login into docker hub

Eg: <https://hub.docker.com/>

10. To create tag in the docker

docker tag demo1 anilmikkili/test-docker:1

11. Needs to login docker hub

Command: Docker login and enter username and password.

12. docker push anilmikkili/test-docker:1🡪 to push docker image into docker hub

## container creation:

13. docker run -dti --name cntr1 demo1

cntr1🡪 container name

demo1🡪 image name

14. docker run -dti --name cntr2 -p 80:80 demo1🡪 adding port to the container

15. docker ps🡪 container details

16. docker attach cntr1🡪 to move into container

17. service apache2 status 🡪 To check the status else start it

18. open container file with public ip of instance with port 80

Eg:- 54.175.156.41:80

19. finally stop the service apache2 in the container.

\*\*\*\*\*\*\*Tomcat Server-Installation\*\*\*\*\*\*\*\*\*\*

1. Launch new Ec2 instance
2. Install java

yum install java-1.8\* -y

1. download Apache-tomcat

sudo wget <https://dlcdn.apache.org/tomcat/tomcat-8/v8.5.77/bin/apache-tomcat-8.5.77.zip>

1. Unzip tomcat

unzip apache-tomcat-8.5.77.zip

# tar -xvf apache-tomcat-8.5.77.zip

* 1. Give access read write n execute

Chmod 777 webapps

1. Change dir to conf and go to edit the tomcat-users.xml (vi tomcat-users.xml)

And change values like below

<role rolename="manager-gui"/>

<role rolename="manager-script"/>

<user username="tomcat" password="tomcat" roles="manager-gui, manager-script"/>

1. Change dir to /webapps/manager/META-INF/ and open context.xml, delete the

Value tag n allow tag.

1. Change dir to /bin and provide permissions to /shutdown.sh, /startup.sh

Eg: chmod 700 startup.sh

1. Start the Apache service

Eg: service Apache start

**To give Access to Jenkins and Docker:**

Firstly, both the services should be available, in Active mode in the same instance.

1. Needs to give an access to Jenkins and Docker

Command:

1. Needs to add group

**Pipeline Script**

pipeline{

agent any

stages{

stage("Git Checkout"){

steps{

git credentialsId: 'git', url: 'https://github.com/amikkili/myweb.git'

}

}

stage("Maven Build"){

steps{

sh "mvn clean package"

sh "mv target/\*.war target/myweb.war"

}

}

stage("Docker-Image"){

steps{

sh "docker build -t anilmikkili/test-docker:1234 ."

}

}

stage("DockerImage-upload"){

steps{

sh "docker login -u anilmikkili -p Jerush@2018"

sh "docker push anilmikkili/test-docker:1234"

sh "docker run -dti --name cntr9 -p 8083:8080 anilmikkili/test-docker:1234"

}

}

stage("deploy-dev"){

steps{

sshagent(['deploy-user']) {

sh "scp -o StrictHostKeyChecking=no target/myweb.war ec2-user@34.238.42.26:/home/ec2-user/apache-tomcat-8.5.77/webapps"

//sh "ssh ec2-user@34.238.42.26 /home/ec2-user/apache-tomcat-8.5.77/bin/shutdown.sh"

//sh "ssh ec2-user@34.238.42.26 /home/ec2-user/apache-tomcat-8.5.77/bin/startup.sh"

}

}

}

}

}

}

}

}

}

Few commands:

To start the Docker in other way

systemctl start docker