Container Operations

>> Download IMAGE

>> Create DOCKER

>> Docker commands

Dockerfile formatting

Install java21, maven, terraform, kubectl

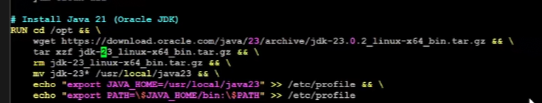
Create IMAGE from Dockerfile docker build -t <imagename> . ……displays error

Create image as dockerhub repo name alongwith <reponame>/<imagename>

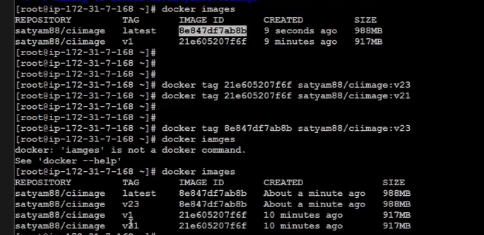
* Docker build -t <reponame>/<imagename> . ……eligible for hub upload

Create tagged version of image from earlier image which looks like v1 for versioning in dockerhub

* Docker tag <imageid> <repositoryname>:v1
* Docker images ……displays two images having only tagname separate



* Another update to dockerfile as per abv for java23 alongwith wget
* Create tagged images as v23, v21 ……creates separate images total 4 in all



* Create CONTAINER using v21, v23 ……2 docker created
* Docker exec -it <containername\_21> bash
* Java –version
* Exit
* Docker exec -it <containername\_23> bash
* Java –version
* exit

Goto Dockerhub >> get docker push commands from repo

* Docker push <reponame>/<imagename>:<versiontag> ……displays error
* Docker login
* Docker push <reponame>/<imagename>:<versiontag>

Goto AWS ECR >>

>> Create private repository=ciimage >> Immutability=No >> Create >> View push commands

Goto IAM >> Create Role >> AWS service >> use case=EC2 >> Next >> AdministratorAccess >> Next >> Rolename=dockerimage-upload >> Create Role

Goto EC2 >> select EC2 >> Modify IAM role >> Attach role

Bookmark EC2 & IAM

* Aws configure list ……displays type appropriately as per iam\_role
* Docker push commands from AWS
* Follow and verify the image in AWS
* Terminate latest EC2 work done & move for automation

Goto Jenkins >> update public-ip