Commands to push image on dockerhub

1. Cd (path of project)
2. Ls :to see list and dockerfile there
3. docker build -t docker-demo1 . : to start creating image
4. docker images : to see docker image which is local
5. docker tag docker-demo niteshnw/docker-demo : this will create repo on dockerhub
6. docker push niteshnw/docker-demo : to push image on dockerhub
7. docker images : to see docker image which is local and also the one which is pushed
8. docker rmi docker-demo1 niteshnw/docker-demo1 : to delete both local and pushed image
9. docker images : to see whether images deleted or not

Commands to pull image:

1. docker run -p 8080:8080 niteshnw/docker-demo1 : to pull image from dockerhub and run on local
2. check on browser using <http://localhost:8080/test-mydocker>

Steps involved in deploying service on cloud on Kubernetes engine:

1. create yaml file in project
2. refer yaml file created in docker-demo1 project
3. open google cloud
4. open kubernetes engine
5. create cluster
6. click on connect and then “run in cloud shell”
7. hit enter to connect to the cluster
8. click on three dots, upload yaml file which you created in project
9. type ls to see whether file loaded or not
10. command to deploy and create container which will run replica/pods : kubectl apply -f docker-k8s-demo1.yaml(name of yaml file)
11. goto workloads
12. click on deployment
13. click on expose
14. target port 8080 and Service type Load balancer which will have external ip
15. Service created with external endpoint
16. Click on external endpoint, append mapping url created in controller: <http://34.67.249.96/test-mydocker>
17. Hence finally service deployed via cloud and can be accessed through endpoint ip