**To delete all images in powershell**

$images = docker images -a -q  
foreach ($image in $images) { docker image rm $image -f }

**Docker**

**For Flask Application in K8**

1. **Ensure to keep the target port in service same same on which flask application is running on the pod**

**Building Images**

**docker build ./ -t <tag>** - run this from the location where Dockerfile is present

*(not mandatory)* **docker run –it <tag**> - to check if docker image is running fine or not

Example: docker build ./ -t hcecosmosapi

**Building Images using File Name**

docker build -f HCEMachineLearning-Authentication-Dockerfile . (Run this command where dockerfile exist and place the dot at end)

**Push images to Azure Container Registery**

1. az acr login --name hcecontainersregistery
2. docker build ./ -t airim
3. docker tag optrendapi hcecontainersregistery.azurecr.io/optrendapi:v1
4. docker push hcecontainersregistery.azurecr.io/optrendapi:v1
5. **docker ps -a (To check running containers)**

**Delete all containers**

(use git bash ,coz cmd doesn’t support nested commands)

**docker rm -vf $(docker ps -a -q**) – remove all container

**docker rmi -f $(docker images -a -q)** –remove all images(delete all container before this)

For any mount related error

**docker volume ls –** lists  all volumes

**docker volume prune –** remove unused volumes

if you know what mount is creating that error use:

**docker volume rm <id/name>**

**Connect with Azure Container Registry from cli**

az aks get-credentials --resource-group Celena-eastUS --name hcecontainersregistery

**Kubectl Command**

1. kubectl apply -f Deployment.yaml
2. kubectl logs -f airim-5c9b5dd76d-25l5w