Question 1

1.Open docker desktop

2. Set up Jenkins Command prompt:

Docker version

(docker pull jenkinsci/blueocean

docker run -d -p 8080:8080 -v /var/run/docker.sock:/var/run/docker.sock jenkinsci/blueocean)

docker run -d -v /var/run/docker.sock:/var/run/docker.sock -p 8080:8080 jenkins/jenkins

3. access localhost:8080

4.To get the password:

Docker ps

docker exec 7bdcac350c45 cat /var/jenkins\_home/secrets/initialAdminPassword

* If jenkins not started after restart,

A black screen with white text

Description automatically generated

Update mailer plugin

Git

Git init

Git add

Git commit -m “ “

git remote add origin <remote\_repository\_url>

git push -u origin master

Question 4

Installing Minikube

curl -Lo minikube https://storage.googleapis.com/minikube/releases/latest/minikube-linux-amd64&& chmod +x minikube && sudo mv minikube /usr/local/bin/

Run

sudo minikube version

sudo apt install conntrack

sudo minikube start --kubernetes-version=v1.23.3 --vm-driver=none

sudo minikube status

alias k="sudo kubectl"

**Creating a pod that goes to the pending mode after creation is something you should design (5 mark**

conditions that cause a pod to become "Pending."

1. Resource Requests Exceed Node Capacity: If a pod's resource requests (CPU and memory) are higher than the available capacity on any node in the cluster, the pod will remain pending until sufficient resources become available.
2. Node Selector and Affinity Rules: If a pod has node selector or affinity rules defined that cannot be fulfilled by any node in the cluster, the pod will remain pending.
3. Resource Constraints: If there are other constraints in the cluster, such as node conditions or node labels, that prevent the pod from being scheduled on any node, it will remain pending.
4. Pod Priority: If the pod has a priority class set, and there are not enough resources to schedule it, it will stay in the pending state until resources are available or lower priority pods are terminated.

Pod1.yaml

apiVersion: v1

kind: Pod

metadata:

name: pending-pod

spec:

containers:

- name: nginx

image: nginx

resources:

requests:

cpu: "10"

k apply -f pod1.yaml

**create a Kubernetes Cron job1 that runs every 10 seconds and if it finds a pod that is in the pending mode, it sends an email to you.**

**-** Kubernetes CronJobs enable the scheduling and automation of recurring tasks based on a specified cron-like schedule.

Commands:

K get cronjobs

K get pods

K get pods -w

K logs <pod\_name>