1. Create the the ckad-multi-containers namespace, create a pod named  
tres-containers-pod, which has 3 containers matching the below requirements:  
  
● The first container named primero runs busybox:1.28 image and has  
ORDER=FIRST environment variable.  
● The second container named segundo runs nginx:1.17 image and is exposed  
at port 8080.  
● The last container named tercero runs busybox:1.31.1 image and has  
ORDER=THIRD environment variable.

**Answer**: Created a file “multi-container.yaml” with the below content

apiVersion: v1

kind: Pod

metadata:

creationTimestamp: null

labels:

run: tres-containers-pod

name: tres-containers-pod

namespace: ckad-multi-containers

spec:

containers:

- image: busybox:1.28

name: primero

command: ["sleep","4600"]

env:

- name: ORDER

value: FIRST

- image: nginx:1.17

name: segundo

ports:

- containerPort: 8081

- image: busybox:1.31.1

name: tercero

command: ["sleep","4600"]

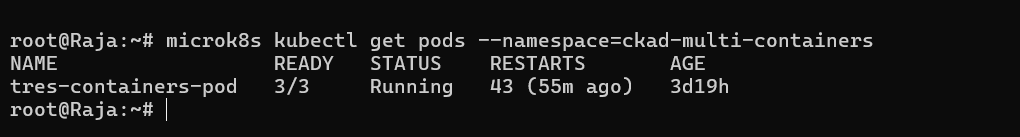
env:

- name: ORDER

value: THIRD

Executed kubectl apply -f multi-container.yaml

**Output**:



Name: tres-containers-pod

Namespace: ckad-multi-containers

Priority: 0

Service Account: default

Node: raja/172.17.92.44

Start Time: Thu, 19 Sep 2024 19:02:33 +0000

Labels: run=tres-containers-pod

Annotations: cni.projectcalico.org/containerID: 029fb2cbefd6ace74aa706944e8b8b868f726f25a1bc43d072e2c7f7c582c4c0

cni.projectcalico.org/podIP: 10.1.64.226/32

cni.projectcalico.org/podIPs: 10.1.64.226/32

Status: Running

IP: 10.1.64.226

IPs:

IP: 10.1.64.226

Containers:

primero:

Container ID: containerd://249261ff5799bb873e28ad4e044cbbb4729471ec30dc2cad2ee319e7545e8fd8

Image: busybox:1.28

Image ID: docker.io/library/busybox@sha256:141c253bc4c3fd0a201d32dc1f493bcf3fff003b6df416dea4f41046e0f37d47

Port: <none>

Host Port: <none>

Command:

sleep

4600

State: Running

Started: Mon, 23 Sep 2024 13:13:54 +0000

Last State: Terminated

Reason: Completed

Exit Code: 0

Started: Mon, 23 Sep 2024 11:55:18 +0000

Finished: Mon, 23 Sep 2024 13:13:53 +0000

Ready: True

Restart Count: 21

Environment:

ORDER: FIRST

Mounts:

/var/run/secrets/kubernetes.io/serviceaccount from kube-api-access-d5l8j (ro)

segundo:

Container ID: containerd://ef5734044aab9a92c26a53291385477c092a1bd55658a0fa13bf78f72568544c

Image: nginx:1.17

Image ID: docker.io/library/nginx@sha256:6fff55753e3b34e36e24e37039ee9eae1fe38a6420d8ae16ef37c92d1eb26699

Port: 8081/TCP

Host Port: 0/TCP

State: Running

Started: Sun, 22 Sep 2024 18:06:41 +0000

Last State: Terminated

Reason: Unknown

Exit Code: 255

Started: Thu, 19 Sep 2024 19:02:35 +0000

Finished: Sun, 22 Sep 2024 18:01:12 +0000

Ready: True

Restart Count: 1

Environment: <none>

Mounts:

/var/run/secrets/kubernetes.io/serviceaccount from kube-api-access-d5l8j (ro)

tercero:

Container ID: containerd://f551fe382b4b0f84746890945144c8eaa62c3b5eb6fefd85145b64bb8e25e075

Image: busybox:1.31.1

Image ID: docker.io/library/busybox@sha256:95cf004f559831017cdf4628aaf1bb30133677be8702a8c5f2994629f637a209

Port: <none>

Host Port: <none>

Command:

sleep

4600

State: Running

Started: Mon, 23 Sep 2024 13:13:55 +0000

Last State: Terminated

Reason: Completed

Exit Code: 0

Started: Mon, 23 Sep 2024 11:55:19 +0000

Finished: Mon, 23 Sep 2024 13:13:54 +0000

Ready: True

Restart Count: 21

Environment:

ORDER: THIRD

Mounts:

/var/run/secrets/kubernetes.io/serviceaccount from kube-api-access-d5l8j (ro)

Conditions:

Type Status

PodReadyToStartContainers True

Initialized True

Ready True

ContainersReady True

PodScheduled True

Volumes:

kube-api-access-d5l8j:

Type: Projected (a volume that contains injected data from multiple sources)

TokenExpirationSeconds: 3607

ConfigMapName: kube-root-ca.crt

ConfigMapOptional: <nil>

DownwardAPI: true

QoS Class: BestEffort

Node-Selectors: <none>

Tolerations: node.kubernetes.io/not-ready:NoExecute op=Exists for 300s

node.kubernetes.io/unreachable:NoExecute op=Exists for 300s

Events: <none>

Question2

Create a storage class with the name banana-sc-ckad08-str as per the properties given

below:

- Provisioner should be kubernetes.io/no-provisioner,

- Volume binding mode should be WaitForFirstConsumer.

- Volume expansion should be enabled.

Answer:

Created a file “**storageclass.yaml**” with the below content

apiVersion: storage.k8s.io/v1

kind: StorageClass

metadata:

name: low-latency

annotations:

storageclass.kubernetes.io/is-default-class: "false"

provisioner: kubernetes.io/no-provisioner

reclaimPolicy: Retain # default value is Delete

allowVolumeExpansion: true

mountOptions:

- discard # this might enable UNMAP / TRIM at the block storage layer

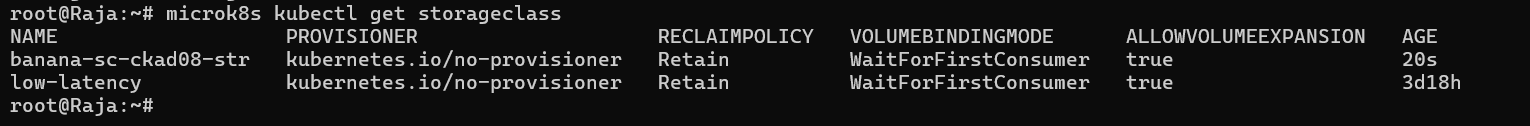
volumeBindingMode: WaitForFirstConsumer

parameters:

guaranteedReadWriteLatency: "true" # provider-specific

Executed kubectl apply -f storageclass.yaml

**Output**:



Question 3.

Create the ckad-job namespace, create a cronjob named simple-node-job to run every

30 minutes to list all the running processes inside a container that used node image (the

command needs to be run in a shell).

In Unix-based operating systems, ps -eaf can be use to list all the running processes.

Answer:

Created a file “cronjob.yaml” with the below content

apiVersion: batch/v1

kind: CronJob

metadata:

name: simple-node-job

namespace: ckad-job

spec:

schedule: "30 \* \* \* \*"

jobTemplate:

spec:

template:

spec:

containers:

- name: node

image: node

imagePullPolicy: IfNotPresent

command:

- /bin/sh

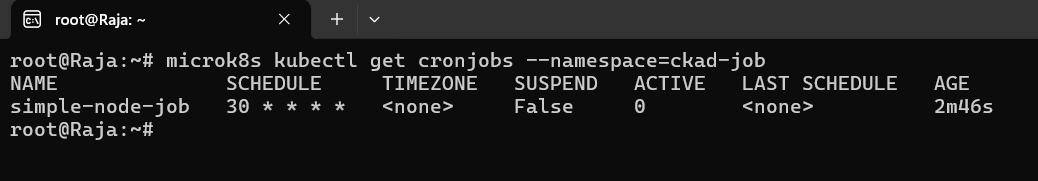
- -c

- ps -eaf

restartPolicy: OnFailure

Executed “kubectl apply -f cronjob.yaml”

Output:



Question 4

You are requested to create a network policy named deny-all-svcn that denies all

incoming and outgoing traffic to ckad12-svcn namespace.

Answer:

Created a file “networkpolicy.yaml” with the below content

kind: NetworkPolicy

apiVersion: networking.k8s.io/v1

metadata:

name: web-deny-all

namespace: ckad12-svcn

spec:

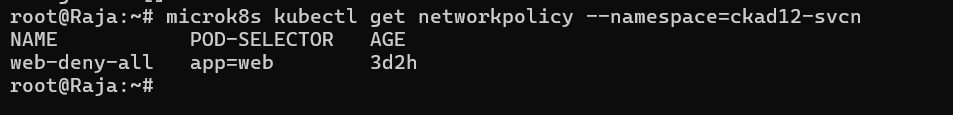
podSelector:

matchLabels:

app: web

ingress: []

**Output**:



Question 5

Deploy a pod with name webapp-svcn using the kodekloud/webapp-color image with the

label tier=msg.

Now, create a service webapp-service-svcn to expose the pod webapp-svcn application

within the cluster on port 6379.

Answer:

Created a file “web-app.yaml” with the below content

apiVersion: v1

kind: Pod

metadata:

labels:

tier: msg

name: webapp-svcn

namespace: default

spec:

containers:

- image: kodekloud/webapp-color

name: webapp

---

apiVersion: v1

kind: Service

metadata:

name: webapp-service-svcn

spec:

selector:

app.kubernetes.io/name: msg

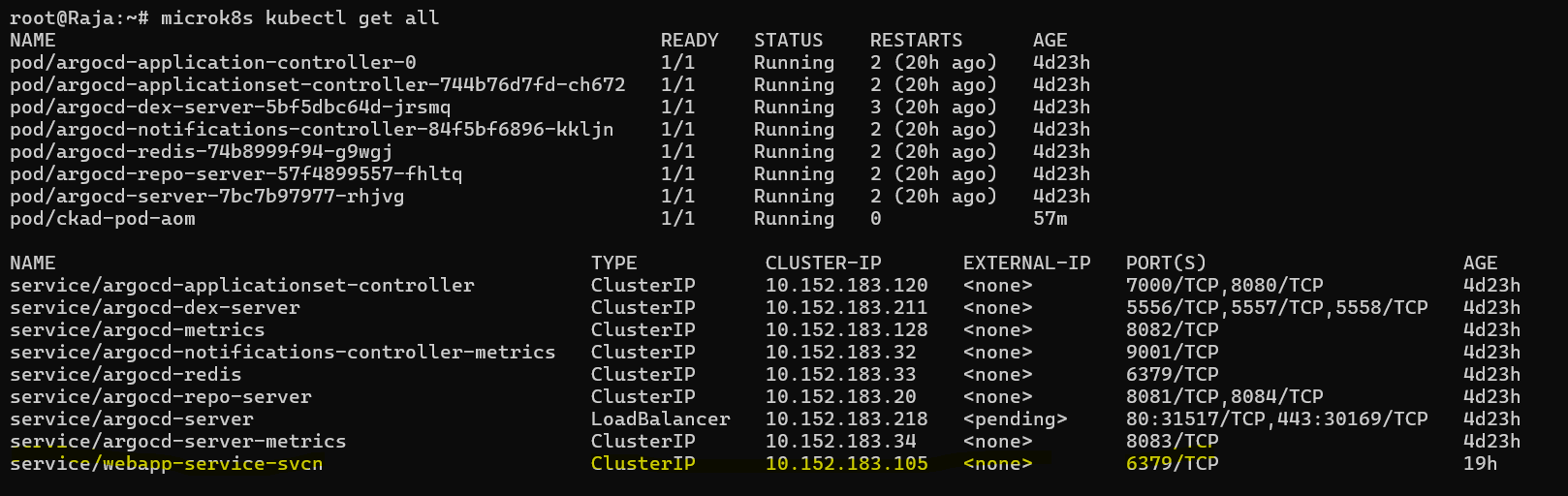
ports:

- name: service-port

protocol: TCP

port: 6379

**Output**:



Question 6

Create a ConfigMap named ckad04-config-multi-env-files-aecs in the default

namespace from the environment(env) files provided

file1.properties

—----------------------------

exam=ckad

retries=2

allowed=true

file2.properties

—------------------------------

practice=must

modetype=openbook

difficulty=fairlyEasy

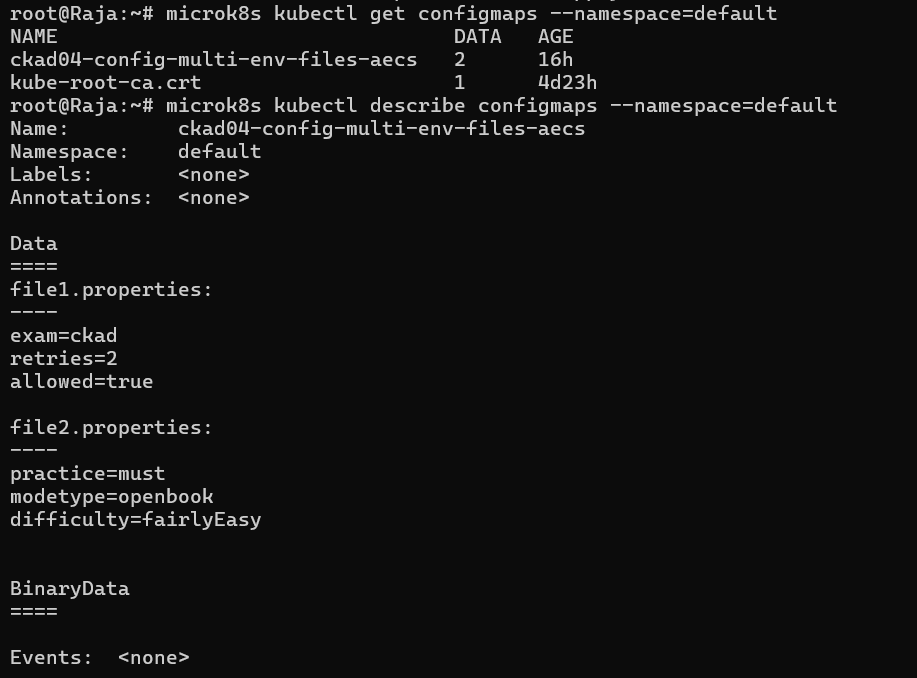
**Answer**:

Executed below command

Created a folder “configure-pod-container/configmap/” & Copied properties files file1.properites & file2.properties under it.

Executed “kubectl create configmap ckad04-config-multi-env-files-aecs --namespace=default --from-file=configure-pod-container/configmap/ “

**Output**:



Question 7

Create a new secret named ckad01-db-scrt-aecs with the data given below.

Secret Name: ckad01-db-scrt-aecs

Secret 1: DB\_Host=sql01

Secret 2: DB\_User=root

Secret 3: DB\_Password=password123

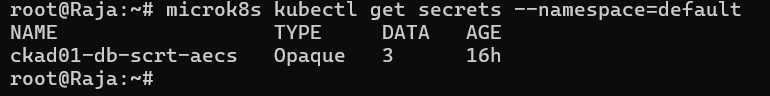
Answer:

Created 3 files secret1, secret2 and secret3 with the content provided in the question and copied these files under a new folder secrets/

Executed below command

kubectl create secret generic ckad01-db-scrt-aecs --from-file=secrets/secret1.txt --from-file=secrets/secret2.txt --from-file=secrets/secret3.txt --namespace=default

**Output**:



Question 8

Create a ResourceQuota called ckad16-rqc in the namespace ckad16-rqc-ns and

enforce a limit of one ResourceQuota for the namespace.

Answer:

Created a file resourcequota.yaml with below content

apiVersion: v1

kind: ResourceQuota

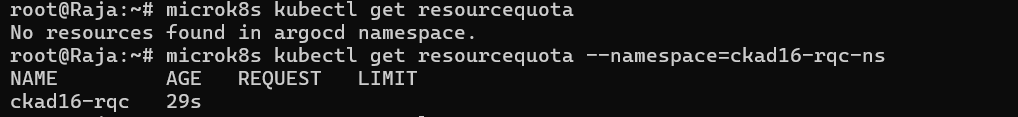
metadata:

name: ckad16-rqc

namespace: ckad16-rqc-ns

What’s Missing? Couldn’t enforce the limit for ResourceQuota.

**Output**:



Question 9

Create a web application pod simple-webapp-aom with a readinessProbe using the given

specifications.

Configure an HTTP readiness probe to the existing pod simple-webapp with path value

set to /ready and port number to access container is 8080.

**Answer**:

Created a file “readiness-probe.yaml” with below content

apiVersion: v1

kind: Pod

metadata:

name: simple-webapp-aom

namespace: default

spec:

containers:

- name: nginx-container

image: nginx

ports:

- containerPort: 8080

readinessProbe:

httpGet:

path: /ready

port: 8080

initialDelaySeconds: 15

periodSeconds: 10

**Output**:

Name: simple-webapp-aom

Namespace: default

Priority: 0

Service Account: default

Node: raja/172.17.92.44

Start Time: Mon, 23 Sep 2024 13:02:42 +0000

Labels: <none>

Annotations: cni.projectcalico.org/containerID: b7c8f4665de1f464427ecd02d9497a19e48cff7811cdd398e2b161c4ad44469f

cni.projectcalico.org/podIP: 10.1.64.199/32

cni.projectcalico.org/podIPs: 10.1.64.199/32

Status: Running

IP: 10.1.64.199

IPs:

IP: 10.1.64.199

Containers:

nginx-container:

Container ID: containerd://d0464c24842788f562b7f7a236f04b0ad20ffa4640647d174c4be1b348974caa

Image: nginx

Image ID: docker.io/library/nginx@sha256:04ba374043ccd2fc5c593885c0eacddebabd5ca375f9323666f28dfd5a9710e3

Port: 8080/TCP

Host Port: 0/TCP

State: Running

Started: Mon, 23 Sep 2024 13:02:46 +0000

Ready: False

Restart Count: 0

Readiness: http-get http://:8080/ready delay=15s timeout=1s period=10s #success=1 #failure=3

Environment: <none>

Mounts:

/var/run/secrets/kubernetes.io/serviceaccount from kube-api-access-bbmq9 (ro)

Conditions:

Type Status

PodReadyToStartContainers True

Initialized True

Ready False

ContainersReady False

PodScheduled True

Volumes:

kube-api-access-bbmq9:

Type: Projected (a volume that contains injected data from multiple sources)

TokenExpirationSeconds: 3607

ConfigMapName: kube-root-ca.crt

ConfigMapOptional: <nil>

DownwardAPI: true

QoS Class: BestEffort

Node-Selectors: <none>

Tolerations: node.kubernetes.io/not-ready:NoExecute op=Exists for 300s

node.kubernetes.io/unreachable:NoExecute op=Exists for 300s

Events:

Type Reason Age From Message

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

Warning Unhealthy 3m44s (x1292 over 3h16m) kubelet Readiness probe failed: Get "http://10.1.64.199:8080/ready": dial tcp 10.1.64.199:8080: connect: connection refused

Question 10

Fix the yaml

apiVersion: v1/beta1

kind: Pod

metadata:

name: ckad-pod-aom

spec:

containers:

- command:

- sleep

- 3600

image: busybox

name: pods-simple-container

**Answer:**

**Corrected the yaml as below:**

**apiVersion: v1**

**kind: Pod**

**metadata:**

**name: ckad-pod-aom**

**spec:**

**containers:**

**- name: pods-simple-container**

**image: busybox**

**command: ["sleep","3600"]**