## Ex1

## 

use consul:1.15.4 as container image for consul container

## Ex2

Creating Deployment using Recreate Strategy (Blue/Green)

## Ex3

Do - Wordpress-App, Gogs-App and Voting\_app deployments

**Note**: Deployment manifests are available in NOTES folder

## Ex4

Create a pod called a multi-cont-pod with two containers.

Container 1: name: alpha, image: nginx

Container 2: name: beta, image: busybox, command sleep 4800.

Environment Variables:

container 1:

name: alpha

Container 2:

name: beta

## Ex5

Create deployment using ‘game-demo’ configMap with 5 replicas

## 

## Ex6

Create MySQL Pod using

image - mysql:5.7

Env:

MYSQL\_ROOT\_PASSWORD: "rootroot"

MYSQL\_PASSWORD: "rootroot"

MYSQL\_DATABASE: test

Provide the kubectl command to use mysql using password and database name.

## Ex7

Storage - PV and PVC using NFS

# CKAD Nov 2024

1. Create a namespace called 'mynamespace' and a pod with image nginx called nginx on this namespace
2. Create a busybox pod (using YAML) that runs the command "env". Run it and see the output

kubectl run nginx **--image**=nginx --command -- env

1. Get the YAML for a new namespace called ‘act-clad’ without creating it
2. Create the YAML for a new ResourceQuota called 'act-clad' with hard limits of 1 CPU, 1G memory

kubectl create quota act-clad --hard=cpu=1,memory=1Gi --dry-run=client -o yaml

1. Show all labels of the pods

kubectl get pods --show-labels

1. Add labels “app=v2” for pod 'nginx' in namespace 'mynamespace'
2. Create a pod that will be placed on a node **controlplane**.
3. Create a configmap named “config” with values foo=lala,foo2=lolo
4. Create a configMap 'cmvolume' with values 'var8=val8', 'var9=val9'.   
   Load this as a volume inside an nginx pod on path '/etc/lala'.   
   Create the pod and 'ls' into the '/etc/lala' directory.

K Create cm **--from-literal**=

K exec -it po1 – ls /etc/lala > file1.txt

### Create an nginx pod that uses 'myuser' as a service account

1. Create a pod with an nginx container exposed on port 80. Add a busybox init container which downloads a page using "wget -O /work-dir/index.html<http://neverssl.com/online>". Make a volume of type emptyDir and mount it in both containers. For the nginx container, mount it on "/usr/share/nginx/html" and for the initcontainer, mount it on "/work-dir". When done, get the IP of the created pod and create a busybox pod and run "wget -O- IP"

YAML

apiVersion: v1

kind: Pod

metadata:

labels:

run: box

name: box

spec:

initContainers:

- args:

- /bin/sh

- -c

- "wget -O /work-dir/index.html http://neverssl.com/online"

image: busybox

name: box

volumeMounts:

- name: vol

mountPath: /work-dir

containers:

- image: nginx

name: nginx

ports:

- containerPort: 80

volumeMounts:

- name: vol

mountPath: /usr/share/nginx/html

volumes:

- name: vol

emptyDir: {}

kubectl run box-test --image=busybox --restart=Never -it --rm -- /bin/sh -c "wget -O- $(kubectl get pod box -o jsonpath='{.status.podIP}')"