Run a test pods:

kubectl run my-nginx --image=nginx --replicas=2 --port=80

kubectl run webserver --image=nginx:alpine

Pod:

List all pods

#kubectl get pods

List pods for all namespace

# kubectl get pods -all-namespaces

List all critical pods

#kubectl get -n kube-system pods -a

List pods with more info

#kubectl get pod -o wide, kubectl get pod/<pod-name> -o yaml

Get pod info

#kubectl describe pod/srv-mysql-server

List all pods with labels

#kubectl get pods --show-labels

List all unhealthy pods

#kubectl get pods –field-selector=status.phase!=Running –all-namespaces

List running pods

#kubectl get pods –field-selector=status.phase=Running

Get Pod initContainer status

#kubectl get pod --template ’{{.status.initContainerStatuses}}’ <pod-name>

kubectl run command

#kubectl exec -it -n "$ns" "$podname" – sh -c "echo $msg »/dev/err.log"

Watch pods

#kubectl get pods -n wordpress --watch

Get pod by selector

#kubectl get pods –selector="app=syslog" -o jsonpath=’{.items[\*].metadata.name}’

List pods and images

#kubectl get pods -o=’custom-columns=PODS:.metadata.name,Images:.spec.containers[\*].image’

List pods and containers

# kubectl get pods -o=’custom-columns=PODS:.metadata.name,CONTAINERS:.spec.containers[\*].name’

Check your pods’ IPs:

# kubectl get pods -l run=my-nginx -o yaml | grep podIP

show status:

#kubectl get deploy deploymentname -o yaml --export

enter into a pod from kub cli:

kubectl exec -it <podname> -- /bin/bash

kubectl run webserver --image=nginx:alpine

kubectl get pods

kubectl delete pod webserver-67bd7c7fc7-scs6j

kubectl get services

minikube service webserver --url

kubectl expose deployment webserver --type=LoadBalancer --port=80

kubectl get pvc

minikube service wordpress --url

How to enter a running pods

kubectl exec -it shell-demo -- /bin/bash

Opening a shell when a Pod has more than one Container

#kubectl exec -it my-pod --container main-app -- /bin/bash

#kubectl apply -k ./

#kubectl delete -k ./

Run images:

Run curl test temporarily kubectl run --generator=run-pod/v1 --rm mytest --image=yauritux/busybox-curl

Run wget test temporarily kubectl run --generator=run-pod/v1 --rm mytest --image=busybox -it wget

Run nginx deployment with 2 replicas

# kubectl run my-nginx --image=nginx --replicas=2 --port=80

Run nginx pod and expose it

#kubectl run my-nginx --restart=Never --image=nginx --port=80 --expose

Run nginx deployment and expose it

#kubectl run my-nginx --image=nginx --port=80 --expose

List authenticated contexts

#kubectl config get-contexts, ~/.kube/config

Set namespace preference

#kubectl config set-context <context\_name> --namespace=<ns\_name>

List pods with nodes info

#kubectl get pod -o wide

List everything

#kubectl get all --all-namespaces

Get all services

#kubectl get service --all-namespaces

Get all deployments

#kubectl get deployments --all-namespaces

Show nodes with labels

#kubectl get nodes --show-labels

Get resources with json output

#kubectl get pods --all-namespaces -o json

Validate yaml file with dry run

#kubectl create --dry-run --validate -f pod-dummy.yaml

Start a temporary pod for testing

#kubectl run --rm -i -t --image=alpine test-$RANDOM -- sh

kubectl run shell command

#kubectl exec -it mytest -- ls -l /etc/hosts

Get system conf via configmap

#kubectl -n kube-system get cm kubeadm-config -o yaml

Get deployment yaml

#kubectl -n denny-websites get deployment mysql -o yaml

Explain resource kubectl explain pods, kubectl explain svc

Watch pods

#kubectl get pods -n wordpress --watch

Open a bash terminal in a pod

#kubectl exec -it storage sh

Check pod environment variables

#kubectl exec redis env

Enable kubectl shell autocompletion echo "source <(kubectl completion bash)" »~/.bashrc, and reload

kubectl edit service my-nginx1

Check Performance:

Get node resource usage

#kubectl top node

Get pod resource usage

#kubectl top pod

Get resource usage for a given pod k

#ubectl top <podname> --containers

List resource utilization for all containers

#kubectl top pod --all-namespaces --containers=true

Resources Deletion:

Delete pod

#kubectl delete pod/<pod-name> -n <my-namespace>

Delete pod by force

#kubectl delete pod/<pod-name> --grace-period=0 --force

Delete pods by labels

#kubectl delete pod -l env=test

Delete deployments by labels

#kubectl delete deployment -l app=wordpress

Delete all resources filtered by labels

#kubectl delete pods,services -l name=myLabel

Delete resources under a namespace

#kubectl -n my-ns delete po,svc --all

Delete persist volumes by labels

#kubectl delete pvc -l app=wordpress

Delete state fulset only (not pods)

#kubectl delete sts/<stateful\_set\_name> --cascade=false

Log & Conf Files

Config folder --> /etc/kubernetes/

Certificate files ---> /etc/kubernetes/pki/

Credentials to API server ---->/etc/kubernetes/kubelet.conf

Superuser credentials ----> /etc/kubernetes/admin.conf

kubectl config file---> ~/.kube/config

Kubernets working dir---> /var/lib/kubelet/

Docker working dir --->/var/lib/docker/, /var/log/containers/

Etcd working dir ----> /var/lib/etcd/

Network cni ---> /etc/cni/net.d/

Log files ---> /var/log/pods/

log in worker node ---> /var/log/kubelet.log, /var/log/kube-proxy.log

log in master node ----> kube-apiserver.log, kube-scheduler.log, kube-controller-manager.log

Env ---> /etc/systemd/system/kubelet.service.d/10-kubeadm.conf

Env ---> export KUBECONFIG=/etc/kubernetes/admin.conf

<https://kubernetes.io/docs/tasks/run-application/run-stateless-application-deployment/>