Popular commands

```
# kubeadm init
# kubeadm reset
# kubectl delete node {{node name}}
# kubeadm token create --print-join-command
```

```
# kubectl version
# kubectl get componentstatuses
# kubectl cluster-info
# kubectl config view
```

```
# kubectl get nodes
# kubectl describe node {nodeName}
```

```
# kubectl get events
```

```
# kubectl get service
# kubectl get services --namespace=kube-system kube-dns
```

```
# kube-down.sh
```

```
# kubectl get pods
# kubectl get pods --namespaces=kube-system
# kubectl describe pods
# kubectl describe pods {{pod name}}
# kubectl describe pods/{{pod name}}
# kubectl exec {{pod name}} -- curl <<IP>>
```

,,

yaml . ( )

```
# kubectl run kuard --image=gcr.io/kuar-demo/kuard-amd64:1
# kubectl apply -f name.yaml

# kubectl delete -f name.yaml
# kubectl delete deployments/kuard
# kubectl delete pods/kuard
# kubectl delete {} {}


# kubectl run user-service --image={} --replicas={} --label={}
# kubectl expose deployment user-service
# kubectl get service -o wide
# kubectl edit deployment/user-service

# kubectl get endpoints user-service --watch
# kubectl edit service user-service
# kubectl describe service user-sercie


# kubectl port-forward kuard 8080:8080
```

```
# kubectl ger daemonSets --namespace=kube-system kube-proxy
```

```
# kubectl get deployments --namespace=kube-system kube-dns
```

```
# kubectl config set-context myContext --namespace=context //
# kubectl config use-context myContext
```

(name pod color=service)

label, annotate .

--overwrite .

```
# kubectl label pods name color=service
```

▪

```
# kubectl label pods name -color
```

pod , -c .

```
# kubectl logs -f {pod name}
# kubectl logs kuard
```

.

```
# kubectl exec -it {pod name} --bash
# kubectl exec kuard date
# kubectl exec -it kuard ash
# kubectl exec -it {pod name} bash
```

```
# kubectl cp {pod name}:/path/file /local/path/file
# kubectl cp {pod name}:/captures/captrue1.txt ./capture1.txt
# kubetctl cp $HOME/config.txt {pod name}:/config.txt
```