minikube

Description	Instruction (Sample)

Starts (and installs if not available) the Minikube node using the driver (optional)
Driver: `docker`, `virtualbox`, `podman`, `vmwarefusion`, `kvm2`, `hyperkit`, `hyperv`, `vmware`, `parallels`, `none`
	`minikube start \`
	`--kubernetes-version=v1.18.0 \`
	`--driver=docker`

Stops the Minikube node	`minikube stop`
Deletes the Minikube node	`minikube delete`
Opens Minikube Kubernetes dashboard inside browser	`minikube dashboard`
Returns status of the Minikube node	`minikube status`
Shows the available addons	`minikube addons list`
Activates an addon (Here: Metrics-Server)	`minikube addons enable metrics-server`
Deactivates an Addon (Here: Metrics-Server)	`minikube addons disable metrics-server`
Access to a service via proxy	`minikube service <service_name>`
Deploy local image to Minikube	`minikube image load <image:tag>`
Show images	`minikube image ls`

kubectl

Description	Instruction (Sample)
Resource Overview	
Generates a list of all objects, e.g. `pods, services, deamonset, replicationcontroller, namespace`	`kubectl get <object1>,<object2>`
List becomes more detailed	`kubectl get <object> -o wide`
List of all pods running on a specific node	`kubectl get pods --field-selector=spec.nodeName=myserver`
Returns a specific object by its name	`kubectl get <object> myobject`
Create or update resources	
Creates a new object (throws errors if any)	`kubectl create <object> demo-object`
Creates a new object based on manifest	`kubectl create -f filename`
Creates or modifies an object based on the manifest	`kubectl apply -f ./obj1.yaml`
Creates or modifies objects corresponding to all manifest files in a folder	`kubectl apply -f ./dir`
Change Resources	
Editing a service, here with a “non-standard” editor (nano)	`KUBE_EDITOR="nano" kubectl edit svc/my-service`
Converts a service to the “NodePort” type	`kubectl patch svc sample-service -p '{"spec": {"type": "NodePort"}}'`
Show state of resources	
Verbose output on all pods	`kubectl describe pods`
Verbose output about a pod	`kubectl describe pods my-pod`
Delete Resources	
Deletes all pods and services named my1 and my2 (also possible with files, then use -f)	`kubectl delete pod,service my1 my2`
Deletes pods and services with the label “`myLabel=labelValue`"	`kubectl delete pods,services -l myLabel=labelValue`
Deletes all pods and services in the “demo-ns” namespace	`kubectl -n demo-ns delete po,svc --all`
Interactions with running pods	
Execute command (bash) in a container of a pod, if -c is omitted, the first container is taken	`kubectl exec my-pod -c my-container -- /bin/bash`
Output a pod's log file (-f optional for streaming)	`kubectl logs -f my-pod`
Read/change KubeConfig	
Read out KubeConfig, -o optional for e.g. JSON-Path	`kubectl config view -o jsonpath='{.u.id}'`
Sets a cluster entry in the KubeConfig	`kubectl config set-cluster mycluster --server=myServer`

Service Manifest

YAML

```yaml
apiVersion: v1
kind: Service
metadata:
  labels:
    app: hello-tutorial
  name: hello-tutorial
spec:
  ports:
  - name: http
    port: 80
    protocol: TCP
    targetPort: 8080
  selector:
    run: hello-tutorial
  type: LoadBalancer
```

Pod Manifest

YAML

```yaml
apiVersion: v1
kind: Pod
metadata:
  name: hello-yaml
spec:
  containers:
  - name: nginx
    image: nginx
```

Config Manifest

YAML

```yaml
apiVersion: v1
kind: ConfigMap
metadata:
  name: configdemo
data:
  config.ini: |
    Content of the config
    file with many lines
```

Ingress Manifest

YAML

```yaml
apiVersion: networking.k8s.io/v1
kind: Ingress
metadata:
  name: test-ingress
  annotations:
    nginx.ingress.kubernetes.io/rewrite-target: /
spec:
  rules:
  - host: a.example.com
    http:
      paths:
      - path: /optional
        backend:
          service:
            name: red
            port:
              number: 80
```

Kubernetes terminology

Resource	Description	Context
Namespace (ns)	Grouping of resources, e.g. clients, environments	Cluster
Pod (po)	Deployment unit of one or more containers	Deployment
ReplicaSet (rs)	Ensures availability of pods	Deployment
ReplicationController (rc)	Predecessor of ReplicaSet	Deployment
Job	Processes a task	Deployment
CronJob	Time-controlled job	Deployment
DaemonSet (ds)	Node related pod	Deployment
StatefulSet (sts)	Stateful pod with fixed identity	Deployment
Deployment (deploy)	Declarative pod deployment and update	Deployment
Taint	Marking of nodes, influences scheduling	Deployment
Toleration	Tag pods, accepts taints	Deployment
Service (svc)	Deploys one/multiple pods with a single, stable IP (port).	Service
Endpoint (ep)	Indicates which pods/external servers are provided as a service	Service
NodePort	Provides a service under one port on all nodes	Service
HostPort	Makes a port of a pod available on the respective host	Service
Ingress (ing)	Provides services to external users through external IP	Service
ConfigMap (cm)	Key-value map for non-secret configuration options	Config
Secret	Similar to the ConfigMap for sensitive data	Config
PersistentVolume (pv)	Reference of persistent storage mountable in a pod through a PersistentVolumeClaim	Storage
PersistentVolumeClaim (pvc)	Request and ownership of a PersistentVolume	Storage
StorageClass (sc)	Defines a type of storage that can be claimed in a PersistentVolumeClaim	Storage
Role	Defines a role in a namespace for mapping rights	Security
ClusterRole	Defines a global role in a cluster	Security
ServiceAccount	Account without reference to a user	Security
RoleBinding	Binds Role or ClusterRole to a Subject (User, Group, ServiceAccount) in a namespace	Security
ClusterRoleBinding	Binds a global role globally to a user/group	Security