

# KUBERNETES CHEATSHEET

Die wichtigsten Kubernetes-Befehle im Überblick:

## SETUP

Befehl	Beschreibung
export KUBECONFIG=PFAD	Setzt die Umgebungsvariable KUBECONFIG auf den Ort der „.kubeconfig“ Datei um eine Cluster-Verbindung einzurichten
kubectl create namespace NAME	Erstellt einen Namespace mit dem Namen „NAME“
kubectl get ns	Zeigt alle Namespaces an (ns = Kurzform für Namespace)
kubectl config set-context --current --namespace NAME	Setzt den Namespace NAME als Default für alle Kommandos
kubectl completion bash	Erstellt Script zur Auto-Completion

## STATUS

Befehl	Item	Kurzform	Beschreibung
kubectl get	nodes	no	Zeigt Kubernetes Cluster-Nodes an
	namespaces	ns	Zeigt alle namespaces
	deployment	deploy	Zeigt alle Deployments im aktuellen Namespace
	ingress	ing	Zeigt alle Ingress im Namespace
	service	svc	Zeigt alle Services im NS
	replicaset	rs	Zeigt alle Replicasets
	Pods	po	Zeigt alle Pods
	horizontalpodautoscalers	hpa	Zeigt alle Autoscaler
	persistentvolume	pv	Zeigt alle PersistentVolumes (Cluster-Weit)
	persistentvolumeclaim	pvc	Zeigt alle PersistentVolumeClaims (im Namespace)
	statefulset	-	Zeigt alle StatefulSets
kubectl describe	s.o.	s.o.	Zeigt mehr Details zur Resource

Beispiel: „kubectl describe service nginx-service“

Mit „kubectl -n beispiel“ oder „kubectl --namespace beispiel“ wird das Kommando auf den Namespace „beispiel“ bezogen.



## ERSTELLEN UND LÖSCHEN

Befehl	Beschreibung
kubectl create	Erstellt Ressource
kubectl apply	Erstellt Ressource oder passt diese an (Update)
kubectl delete	Löscht Ressource
kubectl edit	Editieren der Live-Ressource (VORSICHT)

## Beispiele:

Befehl	Beschreibung
kubectl create service clusterip test-service --clusterip=None --tcp=8080	Erstellt interaktiv einen Service mit Namen test-service und Typ ClusterIP für Port 8080
kubectl create -f service.yaml	Erstellt Service wie in service.yaml beschrieben
kubectl apply -f service.yaml	Erstellt oder aktualisiert Service wie in service.yaml beschrieben
kubectl edit ing test-ingress	Editiert den Live-Stand des Ingress „test-ingress“
kubectl delete pod nginx-vt756-a4g	Löscht den Pod „nginx-vt756-a4g“
kubectl delete pod -l app=nginx	Löscht alle Pods mit dem Label „app“ und dem Labelwert „nginx“
kubectl delete pod --all	Löscht alle Pods

## ENTWICKLUNG, DEBUGGING, WEITERE

Befehl	Beschreibung
kubectl logs nginx-deployment-575895866f-dm88s	Zeigt Logs des Containers „nginx-deployment-575895866f-dm88s“ Mit „-f“ können die Logs gestreamt werden
kubectl port-forward service/nginx-service 8888:8080	Leitet den lokalen Port 8888 auf Service „nginx-service“ auf Port 8080
kubectl scale statefulset cassandra --replicas=4	Skaliert das StatefulSet „cassandra“ auf 4 Replikas
kubectl get service nginx-service -o yaml	Gibt die YAML-Definition des Service nginx-service aus

