

Kubernetes Cheatsheet

Creating Objects:-

Name	Command
Create resource	kubectl apply -f ./<file_name>.yaml
Create from multiple files	kubectl apply -f ./<file_name_1>.yaml -f ./<file_name_2>.yaml
Create all files in directory	kubectl apply -f ./<directory_name>
Create from url	kubectl apply -f https://<url>
Create pod	kubectl run <pod_name> --image <image_name>
Create pod, then expose it as service	kubectl run <pod_name> --image <image_name> --port <port> --expose
Create pod yaml file	kubectl run <pod_name> --image image_name --dry-run=client -o yaml > <file_name>.yaml
Create deployment	kubectl create deployment <deployment_name> --image <image_name>
Create deployment yaml file	kubectl create deployment <deployment_name> --image <image_name> --dry-run=client -o yaml > <file_name>.yaml
Create service	kubectl create service <service-type> <service_name> --tcp=<port:target_port>
Create service yaml file	kubectl create service <service-type> <service_name> --tcp=<port:target_port> --dry-run=client -o yaml > <file_name>.yaml
Expose service from pod/deployment	kubectl expose deployment <pod/deployment_name> --type=<service-type> --port <port> --target-port <target_port>
Create config map from key-value	kubectl create configmap <configmap_name> --from-literal=<key>:<value> --from-literal=<key>:<value>
Create config map from file	kubectl create configmap <configmap_name> --from-file=<file_name>
Create config map from env file	kubectl create configmap <configmap_name> --from-env-file=<file_name>
Create secret from key-value	kubectl create secret generic <secret_name> --from-literal=<key>:<value> --from-literal=<key>:<value>
Create secret from file	kubectl create secret generic <secret_name> --from-file=<file_name>

Name	Command
Create job	kubectl create job <job_name> --image=<image_name>
Create job from cronjob	kubectl create job <job_name> --from=cronjob/<cronjob-name>
Create cronjob	kubectl create cronjob --image=<image_name> --schedule='<cron-syntax>' -- <command> <args>

Monitoring Usage Commands:-

Name	Command
Get node cpu and memory utilization	kubectl top node <node_name>
Get pod cpu and memory utilization	kubectl top pods <pod_name>

Node Commands:-

Name	Command
Describe node	kubectl describe node <node_name>
Get node in yaml	kubectl get node <node_name> -o yaml
Get node	kubectl get node <node_name>
Drain node	kubectl drain node <node_name>
Cordon node	kubectl cordon node <node_name>
Uncordon node	kubectl uncordon node <node_name>

Pod Commands:-

Name	Command
Get pod	kubectl get pod <pod_name>
Get pod in yaml	kubectl get pod <pod_name> -o yaml
Get pod wide information	kubectl get pod <pod_name> -o wide
Get pod with watch	kubectl get pod <pod_name> -w
Edit pod	kubectl edit pod <pod_name>
Describe pod	kubectl describe pod <pod_name>
Delete pod	kubectl delete pod <pod_name>
Log pod	kubectl logs pod <pod_name>
Tail -f pod	kubectl logs pod -f <pod_name>

Name	Command
Execute into pod	kubectrl exec -it pod <pod_name> /bin/bash
Running Temporary Image	kubectrl run <pod_name> --image=curlimages/curl --rm -it --restart=Never -- curl <destination>

Deployment Commands:-

Name	Command
Get deployment	kubectrl get deployment <deployment_name>
Get deployment in yaml	kubectrl get deployment <deployment_name> -o yaml
Get deployment wide information	kubectrl get deployment <deployment_name> -o wide
Edit deployment	kubectrl edit deployment <deployment_name>
Describe deployment	kubectrl describe deployment <deployment_name>
Delete deployment	kubectrl delete deployment <deployment_name>
Log deployment	kubectrl logs deployment/deployment_name -f
Update image	kubectrl set image deployment <deployment_name> <container_name>=<new_image_name>
Scale deployment with replicas	kubectrl scale deployment <deployment_name> --replicas <replicas>

Service Commands:-

Name	Command
Get service	kubectrl get service <service>
Get service in yaml	kubectrl get service <service> -o yaml
Get service wide information	kubectrl get service <service> -o wide
Edit service	kubectrl edit service <service>
Describe service	kubectrl describe service <service>
Delete service	kubectrl delete service <service>

Endpoints Commands:-

Name	Command
Get endpoints	kubectrl get endpoints <endpoints_name>

Ingress Commands:-

Name	Command
Get ingress	kubectl get ingress
Get ingress in yaml	kubectl get ingress -o yaml
Get ingress wide information	kubectl get ingress -o wide
Edit ingress	kubectl edit ingress <ingress_name>
Describe ingress	kubectl describe ingress <ingress_name>
Delete ingress	kubectl delete ingress <ingress_name>

DaemonSet Commands:-

Name	Command
Get daemonset	kubectl get daemonset <daemonset_name>
Get daemonset in yaml	kubectl get daemonset <daemonset_name> -o yaml
Edit daemonset	kubectl edit daemonset <daemonset_name>
Describe daemonset	kubectl describe daemonset <daemonset_name>
Delete daemonset	kubectl delete deployment <daemonset_name>

StatefulSet Commands:-

Name	Command
Get statefulset	kubectl get statefulset <statefulset_name>
Get statefulset in yaml	kubectl get statefulset <statefulset_name> -o yaml
Edit statefulset	kubectl edit statefulset <statefulset_name>
Describe statefulset	kubectl describe statefulset <statefulset_name>
Delete statefuleset	kubectl delete statefulset <statefulset_name>

ConfigMaps Commands:-

Name	Command
Get configmap	kubectl get configmap <configmap_name>
Get configmap in yaml	kubectl get configmap <configmap_name> -o yaml
Edit configmap	kubectl edit configmap <configmap_name>
Describe configmap	kubectl describe configmap <configmap_name>
Delete configmap	kubectl delete configmap <configmap_name>

Secret Commands:-

Name	Command
Get secret	kubectl get secret <secret_name>
Get secret in yaml	kubectl get secret <secret_name> -o yaml
Edit secret	kubectl edit secret <secret_name>
Describe secret	kubectl describe secret <secret_name>
Delete secret	kubectl delete secret <secret_name>

Rollout Commands:-

Name	Command
Restart deployment	kubectl rollout restart deployment <deployment_name>
Undo deployment with the latest revision	kubectl rollout undo deployment <deployment_name>
Undo deployment with specified revision	kubectl rollout undo deployment <deployment_name> --to-revision <revision_number>
Get all revisions of deployment	kubectl rollout history deployment <deployment_name>
Get specified revision of deployment	kubectl rollout history deployment <deployment_name> --revision=<revision_number>

Job Commands:-

Name	Command
Get job	kubectl get job <job_name>
Get job in yaml	kubectl get job <job_name> -o yaml
Edit job in yaml	kubectl edit job <job_name>
Describe job	kubectl describe job <job_name>
Delete job	kubectl delete job <job_name>

Cronjob Commands:-

Name	Command
Get cronjob	kubectl get cronjob cronjob_name
Get cronjob in yaml	kubectl get cronjob <cronjob_name> -o yaml

Name	Command
Edit cronjob	kubectrl edit cronjob <cronjob_name>
Describe cronjob	kubectrl describe cronjob <cronjob_name>
Delete cronjob	kubectrl delete cronjob <cronjob_name>

Network Policy Commands:-

Name	Command
Get networkpolicy	kubectrl get networkpolicy <networkpolicy_name>
Get networkpolicy in yaml	kubectrl get networkpolicy <networkpolicy_name> -o yaml
Get networkpolicy wide information	kubectrl get networkpolicy <networkpolicy_name> -o wide
Edit networkpolicy	kubectrl edit networkpolicy <networkpolicy_name>
Describe networkpolicy	kubectrl describe networkpolicy <networkpolicy_name>
Delete networkpolicy	kubectrl delete networkpolicy <networkpolicy_name>

Labels and Selectors Commands:-

Name	Command
Show labels of node,pod and deployment	kubectrl get <node/pod/deployment> --show-labels
Attach labels to <node/pod/deployment>	kubectrl label <node/pod/deployment> <pod_name> <key>=<value>
Remove labels from <node/pod/deployment>	kubectrl label <node/pod/deployment> <pod_name> <key>-
Select node,pod and deployment by using labels	kubectrl get <node/pod/deployment> -l <key>=<value>