Kubernetes Cheatsheet

Creating Objects
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 and expose 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>

Monitoring Usage Commands
Get node CPU and memory utilization: kubectl top node <node_name>
Get pod CPU and memory utilization: kubectl top pods <pod_name>

Node Commands

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

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

Deployment Commands

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

Service Commands

Get service: kubectl get service <service>
Get service in yaml: kubectl get service <service> -o yaml
Get service wide information: kubectl get service <service> -o wide
Edit service: kubectl edit service <service>
Describe service: kubectl describe service <service>
Delete service: kubectl delete service <service>

Endpoints Commands

Get endpoints: kubectl get endpoints <endpoints_name>

Ingress Commands

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

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

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 statefulset: kubectl delete statefulset <statefulset_name>

ConfigMaps Commands

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

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>