Kubernetes Cheatsheet

Pod Commands:-  
kubectl get pod : Get pod  
kubectl get pod -o wide : Get pod wide information  
kubectl get pod -w : Get pod with watch  
kubectl get pod -o yaml : Get pod in yaml  
kubectl edit pod <pod\_name> : Edit pod  
kubectl describe pod <pod\_name> : Describe pod  
kubectl delete pod <pod\_name> : Delete pod  
kubectl logs pod <pod\_name> : Logs of the pod  
kubectl exec -it pod <pod\_name> /bin/bash :  
Execute into pod

Node Commands:-  
kubectl describe node <node\_name> : Describe node  
kubectl get node <node\_name> -o yaml: Get node in  
yaml  
kubectl get node <node\_name> : Get node  
kubectl drain node <node\_name> : Drain node  
kubectl cordon node <node\_name> : Cordon node  
kubectl uncordon node <node\_name> : Uncordon node

Creating Objects:-  
kubectl apply -f <file\_name> yaml : Create resource  
kubectl apply -f <file\_name>.yaml -f  
<file\_name>.yaml : Create from multiple files  
kubectl apply -f ./ <directory\_name> : Create all files in  
directory  
kubectl apply -f https:// <url> : Create from url  
kubectl run <pod\_name> --  
image<image\_name> : Create pod  
kubectl run <pod\_name>-image  
<image\_name> --port <port> --expose :

Create pod, then expose it as service  
kubectl run <pod\_name> --image=<image\_name> -  
-dry-run=client -o yaml > <file\_name>.yaml :  
Create Pod YAML File  
kubectl create deployment <deployment\_name> -  
-image=<image\_name> : Create Deployment  
kubectl create deployment <deployment\_name> -  
-image=<image\_name> --dry-run=client -o yaml >  
<file\_name>.yaml : Create Deployment YAML File  
kubectl create service <service-type>  
<service\_name> --tcp=<port:target\_port> :

Create Service

kubectl create service <service-type> <service\_name> --  
tcp=<port:target\_port> --dry-run=client -o yaml >  
<file\_name>.yaml : Create Service YAML File  
kubectl expose deployment  
<pod/deployment\_name> --type=<service-type> --  
port=<port> --target-port=<target\_port> :

Expose Service from Pod/Deployment  
kubectl create configmap <configmap\_name> --  
from-literal=<key>=<value> --from-literal=<key>=  
<value> : Create ConfigMap from Key-Value Pairs  
kubectl create configmap <configmap\_name> --  
from-file=<file\_name> : Create ConfigMap from File  
kubectl create configmap <configmap\_name> --from-  
env-file=<file\_name> :

Create ConfigMap from Environment File  
kubectl create secret generic <secret\_name> --  
from-literal=<key>=<value> --from-literal=<key>=  
<value> : Create Secret from Key-Value Pairs  
kubectl create secret generic <secret\_name> --from-  
file=<file\_name> : Create Secret from File

Monitoring Usage Commands:-  
kubectl top node <node\_name> :  
Get node cpu and memory utilization  
kubectl top pods <pod\_name> :  
Get pod cpu and memory utilization  
Deployment Commands:-  
kubectl get deployment <deployment\_name> :

Get Deployment  
kubectl get deployment  
<deployment\_name> -o yaml :  
Get Deployment in YAML Format  
kubectl get deployment  
<deployment\_name> -o wide :  
Get Deployment Wide Information  
kubectl edit deployment  
<deployment\_name> : Edit Deployment  
kubectl describe deployment  
<deployment\_name> : Describe Deployment  
kubectl delete deployment  
<deployment\_name> : Delete Deployment  
kubectl scale deployment <deployment\_name> --  
replicas=<replicas> : Scale Deployment with Replicas

Service Commands:-  
kubectl get service <service> : Get Service  
kubectl get service <service> -o yaml :  
Get Service in YAML Format  
kubectl get service <service> -o wide :  
Get Service Wide Information  
kubectl edit service <service> : Edit Service  
kubectl describe service <service> : Describe Service  
kubectl delete service <service> : Delete Service  
Ingress Commands:-  
kubectl get ingress ; Get Ingress  
kubectl get ingress -o yaml : Get Ingress in YAML Format  
kubectl get ingress -o wide :Get Ingress Wide Information  
kubectl edit ingress <ingress\_name> :  
Describe Ingress

Edit Ingress  
kubectl describe ingress <ingress\_name> :  
kubectl delete ingress <ingress\_name> : Delete Ingress  
Endpoints Commands:-  
kubectl get endpoints <endpoints \_name> :  
Get endpoints

DaemonSet Commands:-  
kubectl get daemonset  
<daemonset\_name> : Get DaemonSet  
kubectl get daemonset  
<daemonset\_name> -o yaml :  
Get DaemonSet in YAML Format  
kubectl edit daemonset  
<daemonset\_name> : Edit DaemonSet  
kubectl describe daemonset  
<daemonset\_name> : Describe DaemonSet  
kubectl delete daemonset  
<daemonset\_name> : Delete DaemonSet  
Job Commands:-  
kubectl get job <job\_name> : Get Job  
kubectl get job <job\_name> -o yaml : Get Job in YAML  
Format  
kubectl edit job <job\_name> : Edit Job  
kubectl describe job <job\_name> : Describe Job  
kubectl delete job <job\_name> : Delete Job

Rollout Commands:-  
kubectl rollout restart deployment  
<deployment\_name> : Restart Deployment  
kubectl rollout undo deployment  
<deployment\_name> : Undo Deployment with the  
Latest Revision  
kubectl rollout undo deployment  
<deployment\_name> --to-revision=  
<revision\_number> : Undo Deployment with Specified  
Revision  
kubectl rollout history deployment  
<deployment\_name> : Get All Revisions of Deployment  
kubectl rollout history deployment  
<deployment\_name> --revision=  
<revision\_number> : Get Specified Revision of  
Deployment

Secret Commands:-  
kubectl get secret <secret\_name> : Get Secret  
kubectl describe secret <secret\_name> : Describe Secret  
kubectl delete secret <secret\_name> : Delete Secret  
kubectl edit secret <secret\_name> : Edit Secret