#KubernetesIn30Days Challenge :-

#Day14 :-

Kubectl cheatsheet

① kubectl apply

#kubectl apply -f ./manifest.yaml

To create any resource(s) of k8 using a single file

#kubectl apply -f ./m1.yaml -f ./m2.yaml

To create any resource(s) of k8 using two definition files.

#kubectl apply -f ./dir

To create any resource(s) of k8 using list of Manifest files.

#kubectl apply -f <URL>

To create any resource(s) of k8 using a URL to downloadble YAML file

#kubectl create deployment nginx --image=nginx

To create deployment object directly by pulling image using imperative way of crating k8s resources.

This is about kubectl apply/create cmd.

② kubectl get :-

```
# kubectl get svc
```

To list the services in default Namespace

```
# kubectl get pods -A
```

To list the all pods across all Namespaces.

```
# kubectl get pods -o wide
```

To get more details about pods.

```
# kubectl get deploy -A
```

To list all the deployment including K8s cluster

```
# kubectl get pod my-pod -o yaml
```

To get the details of pod in YAML format.

```
# kubectl get nodes
```

To list the Nodes in cluster

```
# kubectl describe pod my-pod
```

To get the data about pod in normal text format

```
# kubectl describe svc service1
```

To get the data about service1 in normal text format

```
# kubectl describe deploy deploy1
```

To get the data about deployment in text format

Few Generic Kubectl commands :-

#kubectl cluster-info

Displays the info about K8s master and services

#kubectl describe nod <Node1>

Provides more details about the specified Node

kubectl logs <Pod_name>

To list the logs of a Pod

kubectl rollout status deploy <deployment -name>

To Monitor the status of a deployment rollout.

kubectl get ns

To list the namespaces in the cluster

kubectl get pod --namespace = <ns1>

To list pods from particular ns like ns1

kubectl config view

To Display Kubeconfig file (It'll hide sensitive data)

#kubectl config use-context <context-name>

sets the current context.

kubectl delete pod <pod_name>

kubectl delete deploy <deploy1>

kubectl delete svc <svc1>

kubectl exec -it <pod1> -- /bin/bash

↳ It opens running pod through shell to troubleshoot.

kubectl port-forward <pod1> <local port>: <pod port>

Forwards the incoming request to localport to pod's port.

Imp

kubectl rollout undo deployment <deploy1>

This command is used to Rollback to previous change.

If Incase any issue with the updated version then we can rollback to previous version. Most useful in production Realtime scenarios.

this is the Glance of kubectl command. you can go anytime to k8s documentation to get your required command(s).