Kubernetes commands you should definitely know!

https://youtu.be/wS277TdV3f8?si=XAdQ3hCRO7WOwUbw

Prerequisite to Practice Along

1. Kubernetes Cluster
2. kubectl CLI tool

Installation Link: https://kubernetes.io/docs/tasks/tools/

https://labs.play-with-k8s.com/

Get started with Minikube

```
minikube start
```

Explanation: This command sets up Minikube, a tool for running Kubernetes clusters locally for development purposes.

Check Kubectl

```
kubectl version --client
```

Explanation: This command checks the version of the installed Kubectl client.

Set Context for Cluster in different cloud providers

```
# Connect to EKS
aws eks --region <region> update-kubeconfig --name <cluster-name

# Connect to AKS
az aks get-credentials --resource-group <resource-group> --name

# Connect to GKE
gcloud container clusters get-credentials <cluster-name> --regi
```

Explanation: These commands set the context for the specified cloud-managed Kubernetes clusters, allowing subsequent commands to interact with the chosen cluster.

kubeconfig file

The `kubeconfig` file is a crucial part of managing Kubernetes configurations, stored at `~/.kube/config` by default. It defines clusters, users, and contexts. To check the current active context (cluster and user), use:

```
# Check the current Kubernetes context
kubectl config current-context
```

Explanation: This command displays the name of the current Kubernetes context, which represents the cluster that kubectl commands will interact with.

Manifest File - Creation, Deletion, and Editing Commands

- A Kubernetes manifest file is a short YAML or JSON document that defines how a specific cluster resource should be configured and managed. It declares the desired state, allowing Kubernetes to maintain that state in the cluster.

Kubernetes Deployment:
https://kubernetes.io/docs/concepts/workloads/controllers/deployment/

> 🗃️ **Deployment** is a blueprint for running and managing application instances, ensuring the specified number of **pods** are running. It also manages aspects like the **container image**, configurations, and enables seamless updates, scalability, and self-healing.

```
# Create a Deployment using Manifest file
kubectl apply -f deployment.yaml

# Delete an object using Manifest file
kubectl delete -f deployment.yaml

# Edit a Deployment using Manifest file
kubectl apply -f deployment.yaml
```

Explanation: These commands demonstrate how to create, delete, and edit Kubernetes resources using YAML manifest files.

GET Commands (Status)

```
# Get commands to check the status
kubectl get pods
kubectl get services
kubectl get deployments
```

Explanation: These commands retrieve information about different Kubernetes resources, such as pods, services, and deployments.

Different Options with GET Commands

```
# Get information in YAML format
kubectl get pods -o yaml

# Get information in JSON format
kubectl get services -o json
```

Explanation: These commands show how to retrieve information in different formats for more detailed view with better readability or automation.

Kubectl Describe

```
# Describe a resource
kubectl describe pod <pod-name>
```

Explanation: This command provides detailed information about a specific Kubernetes resource, aiding in troubleshooting and understanding its configuration.

Imperative Commands

Explanation: Imperative commands allow quick actions without the need for manifest files.

https://kubernetes.io/docs/tasks/manage-kubernetes-objects/imperative-command/

```
# Create a Deployment imperatively
kubectl create deployment redis-deploy --image=redis --replicas=
```

💡 Manifest files in Kubernetes are best practice as they offer a clear, version-controlled, and reproducible way to declare the desired state of your application.

Kubectl Create Commands

```
# Create a Deployment with Redis image and 2 replicas
kubectl create deployment redis-deploy --image=redis --replicas=

# Create an Nginx Deployment
kubectl create deployment nginx-deploy --image=nginx
```

Explanation: These commands demonstrate creating deployments with specific images and replica counts.

Create a Pod

```
# Create a Pod
kubectl run <pod-name> --image=<image-name>
```

Explanation: The `kubectl run` command creates a pod imperatively, allowing quick pod deployment without the need for a manifest file.

Edit Commands

```
# Edit a Deployment imperatively
kubectl edit deployment <deployment-name>

# Scale a Deployment imperatively
kubectl scale deployment <deployment-name> --replicas=3
```

Explanation: Edit commands are used to modify existing deployments, either imperatively or by scaling the number of replicas.

> ⚠️ Certain fields of a resource, like its name or type, cannot be edited using `kubectl edit` as they are immutable. For such cases, the `kubectl replace --force` command is used to enforce an update by deleting and recreating the resource with the new configuration.

Kubectl Replace - Force

```
# Replace and force update an object
kubectl replace --force -f <manifest_file>
```

Explanation: This command forcefully updates an object by replacing it with a new configuration.

Debugging Commands

```
# Create a Deployment for debugging
kubectl create deployment debug-deploy --image=alpine

# View logs of a pod
kubectl logs <pod-name>

# Execute a command inside a container
kubectl exec -it <pod-name> -- /bin/sh
```

Explanation: These commands demonstrate debugging techniques, including viewing pod logs and executing commands inside a container.

Logging and Monitoring Commands

```
# View logs of a pod
kubectl logs <pod-name>

# Display Resource Usage (CPU and Memory) of Pods
kubectl top pods
```

Explanation: These commands showcase how to view logs of a pod and display resource usage information for pods.

kubectl Cheat Sheet by Kubernetes

kubectl Cheat Sheet

This page contains a list of commonly used kubectl commands and flags. Note: These instructions are for Kubernetes v1.29. To check the version, use the kubectl version command. Kubectl

https://kubernetes.io/docs/reference/kubectl/cheatsheet/

Kubectl Reference Docs

This section contains the most basic commands for getting a workload running on your cluster.

https://kubernetes.io/docs/reference/generated/kubectl/kubectl-commands#-em-deployment-em-