**Kuberenetes manifest file**

* manifest files are written in yaml
* yaml files are mostly constructed in form of maps (key : pair) and lists

**objective**

* using this file we are creating pods with one container loaded

**Required fields in manifest file**

* apiVersion - Which version of the Kubernetes API you're using to create this object
* kind - What kind of object you want to create
* metadata - Data that helps to identify the object, including a name string, UID, and optional - namespace
* spec - What state you desire for the object

**How to apply manifest file**

* kubectl apply -f pod.yaml
* kubectl create -f pod.yaml

**Apply in specific namespace**

* kubectl apply -f pod.yaml -n <namespace>

**View pods**

* kubectl get pods
* kubect describe pods

**Delete pod through manifest file**

* kubectl delete -f pod.yaml

# Kuberenetes manifest file

* manifest files are written in yaml
* yaml files are mostly constructed in form of maps (key : pair) and lists

# objective-MultiContainerPod

* using this file we are creating pods with two container loaded

# Required fields in manifest file

* apiVersion : Which version of the Kubernetes API you're using to create this object
* kind : What kind of object you want to create
* metadata : Data that helps to identify object, including a name string, UID, and optional - namespace
* spec : What state you desire for the object

# How to apply manifest file

* kubectl apply -f pod.yaml
* kubectl create -f pod.yaml

# Apply in specific namespace

* kubectl apply -f pod.yaml -n <namespace>

# View pods

* kubectl get pods
* kubect describe pods

# Delete pod through manifest file

* kubectl delete -f pod.yaml

# Login into container

* kubectl exec -it fullstack -c apache -- bash
* kubectl exec -it fullstack -c database -- bash

# Logs of each container

* kubectl logs fullstack -c apache
* kubectl logs fullstack -c database