**here are some commonly asked Kubernetes interview questions along with their answers: -**

1. What is Kubernetes?

Answer: Kubernetes is an open-source container orchestration platform that automates the deployment, scaling, and management of containerized applications. It was originally developed by Google and is now maintained by the Cloud Native Computing Foundation (CNCF).

2. What are the key components of Kubernetes architecture?

Answer: The key components of Kubernetes architecture are:

Master Node: Manages the Kubernetes cluster.

Node/Minion/Worker Node: Where the containers are deployed.

etcd: Distributed key-value store used to store cluster data.

kube-apiserver: Exposes the Kubernetes API.

kube-scheduler: Schedules pods to nodes.

kube-controller-manager: Manages various controller processes.

kubelet: Agent running on each node in the cluster.

kube-proxy: Network proxy that runs on each node in the cluster.

3. What is a Pod in Kubernetes?

Answer: A Pod is the smallest and simplest Kubernetes object. It represents a single instance of a running process in the cluster. Pods contain one or more containers that share network and storage resources and are scheduled together on the same node.

4. What is a Kubernetes Deployment?

Answer: A Kubernetes Deployment is a resource object used to define and manage the lifecycle of a replicated application. It ensures that a specified number of pod replicas are running at any given time and handles updates to the application by managing rolling updates and rollbacks.

5. How do you scale a Kubernetes deployment?

Answer: You can scale a Kubernetes deployment using the kubectl scale command. For example, to scale a deployment named my-deployment to three replicas, you would use the following command:

6. What is a Kubernetes Service?

Answer: A Kubernetes Service is an abstraction that defines a logical set of Pods and a policy by which to access them. It provides a stable endpoint for accessing the pods, regardless of changes in the cluster.

7. How do you expose a Kubernetes Deployment externally?

Answer: You can expose a Kubernetes Deployment externally using a Service of type LoadBalancer. For example:

yaml

apiVersion: v1

kind: Service

metadata:

name: my-service

spec: type:

LoadBalancer selector:

app: my-app

ports: - protocol: TCP

port: 80 targetPort: 8080

8. What is the difference between a Deployment and a StatefulSet in Kubernetes?

Answer:

Deployment: Used for stateless applications, manages the deployment and scaling of replica pods.

StatefulSet: Used for stateful applications, maintains a sticky identity for each pod. It is useful for applications that require stable, unique network identifiers, persistent storage, and ordered, graceful deployment and scaling.

9. How do you debug a Kubernetes Pod?

Answer: You can debug a Kubernetes Pod using the following commands:

To view Pod logs: kubectl logs <pod-name>

To execute a command in a Pod: kubectl exec -it <pod-name> -- <command>

10. What is a Kubernetes Ingress?

Answer: A Kubernetes Ingress is an API object that manages external access to services within a Kubernetes cluster. It provides HTTP and HTTPS routing to services based on hostnames or paths.

**point-to-point Kubernetes interview questions along with their answers :-**

1. What is Kubernetes?

Kubernetes is an open-source container orchestration platform for automating the deployment, scaling, and management of containerized applications.

2. What are the key components of Kubernetes architecture?

Master Node: Manages the Kubernetes cluster.

Node/Minion/Worker Node: Where the containers are deployed.

etcd: Distributed key-value store.

kube-apiserver: Exposes the Kubernetes API.

kube-scheduler: Schedules pods to nodes.

kube-controller-manager: Manages various controller processes.

kubelet: Agent running on each node.

kube-proxy: Network proxy running on each node.

3. What is a Pod in Kubernetes?

A Pod is the smallest and simplest Kubernetes object representing a single instance of a running process in the cluster.

4. What is a Kubernetes Deployment?

A Kubernetes Deployment manages the lifecycle of replicated applications, ensuring a specified number of pod replicas are running.

5. How do you scale a Kubernetes deployment?

Scale a deployment using

kubectl scale --replicas=<number of replicas> deployment/<deployment-name>.

6. What is a Kubernetes Service?

A Kubernetes Service provides a stable endpoint for accessing pods, regardless of changes in the cluster.

7. How do you expose a Kubernetes Deployment externally?

Expose a deployment using a Service of type LoadBalancer.

apiVersion: v1

kind: Service

metadata:

name: my-service

spec:

type: LoadBalancer

selector:

app: my-app

ports:

- protocol: TCP

port: 80

targetPort: 8080

8. What is the difference between a Deployment and a StatefulSet in Kubernetes?

Deployment: Stateless applications, manages deployment and scaling of replica pods.

StatefulSet: Stateful applications, maintains a unique identity for each pod, useful for applications that require stable network identifiers and persistent storage.

9. How do you debug a Kubernetes Pod?

View Pod logs: kubectl logs <pod-name>

Execute a command in a Pod: kubectl exec -it <pod-name> -- <command>

10. What is a Kubernetes Ingress?

A Kubernetes Ingress manages external access to services within a Kubernetes cluster, providing HTTP and HTTPS routing based on hostnames or paths.