1. **[What is the difference between config map and secret? (Differentiate the answers as with examples)](https://www.knowledgehut.com/interview-questions/kubernetes" \l "collapse-beginner-2405)**

Config maps ideally stores application configuration in a plain text format whereas Secrets store sensitive data like password in an encrypted format. Both config maps and secrets can be used as volume and mounted inside a pod through a pod definition file.

1. **[If a node is tainted, is there a way to still schedule the pods to that node?](https://www.knowledgehut.com/interview-questions/kubernetes" \l "collapse-beginner-2406)**

When a node is tainted, the pods don't get scheduled by default, however, if we have to still schedule a pod to a tainted node we can start applying tolerations to the pod spec.

1. **[Can we use many claims out of a persistent volume? Explain?](https://www.knowledgehut.com/interview-questions/kubernetes" \l "collapse-beginner-2407)**

The mapping between persistentVolume and persistentVolumeClaim is always one to one. Even When you delete the claim, PersistentVolume still remains as we set persistentVolumeReclaimPolicy is set to Retain and It will not be reused by any other claims.

1. **[How do you deploy a feature with zero downtime in Kubernetes?](https://www.knowledgehut.com/interview-questions/kubernetes" \l "collapse-beginner-2410)**

By default, Deployment in Kubernetes using RollingUpdate as a strategy. Let's say we have an example that creates a deployment in Kubernetes

1. **[How to monitor that a Pod is always running?](https://www.knowledgehut.com/interview-questions/kubernetes" \l "collapse-beginner-2411)**

We can introduce probes. A liveness probe with a Pod is ideal in this scenario.

A liveness probe always checks if an application in a pod is running, if this check fails the container gets restarted. This is ideal in many scenarios where the container is running but somehow the application inside a container crashes.

## [How do you tie service to a pod or to a set of pods?](https://www.knowledgehut.com/interview-questions/kubernetes" \l "collapse-beginner-2414)

By declaring pods with the label(s) and by having a selector in the service which acts as a glue to stick the service to the pods.

1. **[Having a Pod with two containers, can I ping each other? like using the container name?](https://www.knowledgehut.com/interview-questions/kubernetes" \l "collapse-beginner-3506)**

Containers on same pod act as if they are on the same machine. You can ping them using localhost:port itself. Every container in a pod shares the same IP. You can `ping localhost` inside a pod. Two containers in the same pod share an IP and a network namespace and They are both localhost to each other.

1. **[Does the rolling update with state full set replicas =1 makes sense?](https://www.knowledgehut.com/interview-questions/kubernetes" \l "collapse-beginner-3507)**

No, because there is only 1 replica, any changes to state full set would result in an outage. So rolling update of a StatefulSet would need to tear down one (or more) old pods before replacing them.

1. **[what does kube-proxy do?](https://www.knowledgehut.com/interview-questions/kubernetes" \l "collapse-beginner-3509)**

kube-proxy does 2 things

* for every Service, open a random port on the node and proxy that port to the Service.
* install and maintain iptables rules which capture accesses to a virtual ip:port and redirect those to the port in (1)

The kube-proxy is a component that manages host sub-netting and makes services available to other components.Kubeproxy handles network communication and shutting down master does not stop a node from serving the traffic and kubeproxy works, in the same way, using a service. The iptables will route the connection to kubeproxy, which will then proxy to one of the pods in the service.kube-proxy translate the destination address to whatever is in the endpoints.

1. **[Do rolling updates declared with a deployment take effect if I manually delete pods of the replica set with kubectl delete pods or with the dashboard? Will the minimum required a number of pods be maintained?](https://www.knowledgehut.com/interview-questions/kubernetes" \l "collapse-beginner-3513)**

Yes, the scheduler will make sure (as long as you have the correct resources) that the number of desired pods are met. If you delete a pod, it will recreate it. Also deleting a service won't delete the Replica set. if you remove Service or deployment you want to remove all resources which Service created. Also having a single replica for a deployment is usually not recommended because you cannot scale out and are treating in a specific way.

1. **[what is the ingress, is it something that runs as a pod or on a pod?](https://www.knowledgehut.com/interview-questions/kubernetes" \l "collapse-beginner-3516)**

An ingress is an object that holds a set of rules for an ingress controller, which is essentially a reverse proxy and is used to (in the case of nginx-ingress for example) render a configuration file. It allows access to your Kubernetes services from outside the Kubernetes cluster. It holds a set of rules. An Ingress Controller is a controller. Typically deployed as a Kubernetes Deployment. That deployment runs a reverse proxy, the ingress part, and a reconciler, the controller part. the reconciler configures the reverse proxy according to the rules in the ingress object. Ingress controllers watch the k8s api and update their config on changes. The rules help to pass to a controller that is listening for them. You can deploy a bunch of ingress rules, but nothing will happen unless you have a controller that can process them.

LoadBalancer service -> Ingress controller pods -> App service (via ingress) -> App pods

1. **[In  Kubernetes - A Pod is running 2 containers, when One container stops - another Container is still running, on this event, I want to terminate this Pod?](https://www.knowledgehut.com/interview-questions/kubernetes" \l "collapse-beginner-3515)**

You need to add a liveness and readiness probe to query each container,  if the probe fails, the entire pod will be restarted .add **liveness object** that calls any api that returns 200 to you from another container and both liveness and readiness probes run in infinite loops for example, If X depended to Y So add liveness  in X that check the health of Y.Both readiness/liveness probes always have to run after the container has been started .kubelet component performs the liveness/readiness checks and set initialDelaySeconds and it can be anything from a few seconds to a few minutes depending on app start time.

1. **[Suppose you have to use database with your application but well, if you make a database container-based deployment. how would the data persist?](https://www.knowledgehut.com/interview-questions/kubernetes" \l "collapse-beginner-3521)**

Deployments are for stateless services, you want to use a StatefulSet or just define 3+ pods without a replication controller at all. If you care about stable pod names and volumes, you should go for StatefulSet.Using statefulsets you can maintain which pod is attached to which disk.StatefulSets make vanilla k8s capable of keeping Pod state (things like IPs, etc) which makes it easy to run clustered databases. A stateful set is a controller that orchestrates pods for the desired state. StatefulSets formerly known as PetSets will help for the database if hosting your own. Essentially StatefulSet is for dealing with applications that inherently don't care about what node they run on, but need unique storage/state.

## [.](https://www.knowledgehut.com/interview-questions/kubernetes" \l "collapse-beginner-2415)

1. **[Let’s say a Kubernetes job should finish in 40 seconds, however on a rare occasion it takes 5 minutes, How can I make sure to stop the application if it exceeds more than 40 seconds?](https://www.knowledgehut.com/interview-questions/kubernetes" \l "collapse-beginner-2415)**

When we create a job spec, we can give --activeDeadlineSeconds flag to the command, this flag relates to the duration of the job, once the job reaches the threshold specified by the flag, the job will be terminated.

1. **[How do you initiate a rollback for an application?](https://www.knowledgehut.com/interview-questions/kubernetes" \l "collapse-beginner-2417)**

Rollback and rolling updates are a feature of Deployment object in the Kubernetes. We do the Rollback to an earlier Deployment revision if the current state of the Deployment is not stable due to the application code or the configuration. Each rollback updates the revision of the Deployment

1. **[How do you package Kubernetes applications?](https://www.knowledgehut.com/interview-questions/kubernetes" \l "collapse-beginner-2418)**

Helm is a package manager which allows users to package, configure, and deploy applications and services to the Kubernetes cluster.

1. **[What is node affinity and pod affinity?](https://www.knowledgehut.com/interview-questions/kubernetes" \l "collapse-beginner-2420)**
2. Node Affinity ensures that pods are hosted on particular nodes.

Pod Affinity ensures two pods to be co-located in a single node.

1. **[How do you drain the traffic from a Pod during maintenance?](https://www.knowledgehut.com/interview-questions/kubernetes" \l "collapse-beginner-2421)**

When we take the node for maintenance, pods inside the nodes also take a hit. However, we can avoid it by using the below command

kubectl drain <nodename>

When we run the above command it marks the node unschedulable for newer pods then the existing pods are evicted if the API Server supports eviction else it deletes the pods

Once the node is up and running and you want to add it in rotation we can run the below command

kubectl uncordon <nodename>

1. **[What is the impact of upgrading kubelet if we leave the pods on the worker node - will it break running pods? why?](https://www.knowledgehut.com/interview-questions/kubernetes" \l "collapse-beginner-3499)**

Restarting kubelet, which has to happen for an upgrade will cause all the Pods on the node to stop and be started again. It’s generally better to drain a node because that way Pods can be gracefully migrated, and things like Disruption Budgets can be honored. The problem is that `kubectl` keeps up with the state of all running pods, so when it goes away the containers don’t necessarily die, but as soon as it comes back up, they are all killed so `kubectl` can create a clean slate.

1. **[If I have multiple containers running inside a pod, and I want to wait for a specific container to start before starting another one.](https://www.knowledgehut.com/interview-questions/kubernetes" \l "collapse-beginner-3498)**

One way is  Init Containers are for one-shot tasks that start, run, end; all before the next init container or the main container start,

1. **[Does the container restart When applying/updating the secret object (kubectl apply -f mysecret.yml)?  If not, how is the new password applied to the database?](https://www.knowledgehut.com/interview-questions/kubernetes" \l "collapse-beginner-3501)**

If you are mounting the secret as a volume into your pod, when the secret is updated the content will be updated in your pod, without the pod restarting. It's up to your application to detect that change and reload

1. **[If you have a pod that is using a ConfigMap which you updated, and you want the container to be updated with those changes, what should you do?](https://www.knowledgehut.com/interview-questions/kubernetes" \l "collapse-beginner-3505)**

if the config map is mounted into the pod as a volume, it will automatically update not instantly and the files will change inside the container. If it is an environment variable it stays as the old value until the container is restarted

1. [**What happens when a master fails? What happens when a worker fails?**](https://www.zeolearn.com/interview-questions/kubernetes#collapse-beginner-596)

Whenever master node under kubernetes fails, the cluster still remain in an operational mode. It doesn’t affect pod creation or service member changes. If worker node fails, master stop receiving updates from worker node.

1. [**How do I debug a Pending pod?**](https://www.zeolearn.com/interview-questions/kubernetes#collapse-beginner-626)

Pending pod cannot be scheduled onto a node.

Performing command  kubectl describe pod <pod\_name> will help you undestand the problem.

1. [**How do I expose a service to a host outside the cluster?**](https://www.zeolearn.com/interview-questions/kubernetes#collapse-beginner-631)

There are two ways by which we can expose a service to host outside the cluster:

1) Service type should be set to NodePort. Every node in the cluster will be made to listen on the specified NodePort, then all the traffic from any node will be forwarded from that NodePort to a random pod in the service.

2) Service type should be set to Load Balancer mode. Nodeport will be provisioned as mentioned in the above step, and then additional step which will automatically provision the load balancer in the cloud infrastructure.

1. [**Is it possible to route traffic from outside the Kubernetes cluster directly to pods?**](https://www.zeolearn.com/interview-questions/kubernetes#collapse-beginner-636)

Yes, it is possible to route traffic to any or all of the K8 minions as well as Pods. Though there are multiple ways like Ingress, Load-Balancer & NodePort, but the Ingress seems to be a recommended and convenient solution. If you choose the NodePort service, you might want to implement a load balancer in front of your cluster as well.

1. [**Is it possible for containers within a pod to communicate with each other?**](https://www.zeolearn.com/interview-questions/kubernetes#collapse-beginner-639)

Yes, it is possible for containers to communicate within a pod. They reach other on localhost network. For E.g, if you have two containers within a pod, a MySQL container running on port 3306, and a PHP container running on port 80, the PHP container could access the MySQL one through localhost:3306.

1. [**Can you brief about Taints and Tolerations**](https://www.zeolearn.com/interview-questions/kubernetes#collapse-beginner-641)

This is an interesting feature of Kubernetes. This feature help users to mark a node (its like tainting the node) so that no pods can be scheduled to it, unless a pod explicitly tolerates the taint. With this feature, one can create nodes that are reserved for specific pods.

1. [**Explain the role of Secrets in Kubernetes**](https://www.zeolearn.com/interview-questions/kubernetes#collapse-beginner-642)

A Secret is an object that contains a small amount of sensitive data such as a password, a token, or a key. Such information might otherwise be put in a Pod specification or in an image; putting it in a Secret object allows for more control over how it is used, and reduces the risk of accidental exposure.

Objects of type secret are intended to hold sensitive information, such as passwords, OAuth tokens, and ssh keys. Putting this information in a secret is safer and more flexible than putting it verbatim in a pod definition or in a docker image.

1. **What is Kubernetes Load Balancing?**

Load Balancing is one of the most common and standard ways of exposing the services. There are two types of load balancing in K8s and they are:

**Internal load balancer –** This type of balancer automatically balances loads and allocates the pods with the required incoming load.

**External Load Balancer –** This type of balancer directs the traffic from the external loads to backend pods

1. **How to troubleshoot if the POD is not getting scheduled?**

In K8’s scheduler is responsible to spawn pods into nodes. There are many factors that can lead to unstartable POD. The most common one is running out of resources, use the commands like kubectl describe <POD> -n <Namespace> to see the reason why POD is not started. Also, keep an eye on kubectl to get events to see all events coming from the cluster.

1. **What are the different ways to provide external network connectivity to K8?**

By default, POD should be able to reach the external network but vice-versa we need to make some changes. Following options are available to connect with POD from the outer world.

* Nodeport (it will expose one port on each node to communicate with it)
* Load balancers (L4 layer of TCP/IP protocol)
* Ingress (L7 layer of TCP/IP Protocol)