

**2Q) what is the namespace in kubernetes?**

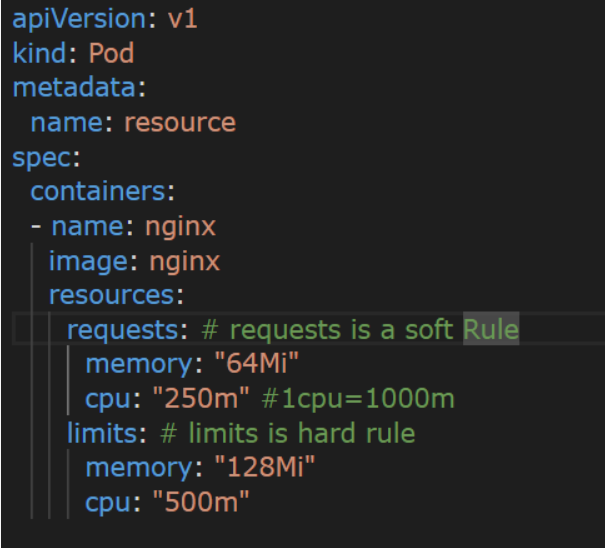
**ANS)** Namespace is logical isolation in kubernetes.Ideally we will create one namespace for one project. So resources will be to the isolated namspace level.

**3Q) what is the deference between pod and containers?**

**Ans)** pod is a smallest deployble unit in kubernetes. It can have multiple containers. Containers in a podshare the same ip address , volumes. This feature is useful when you want to keep multiple apps together with the same Ip,volume, side car containers,proxies etc.

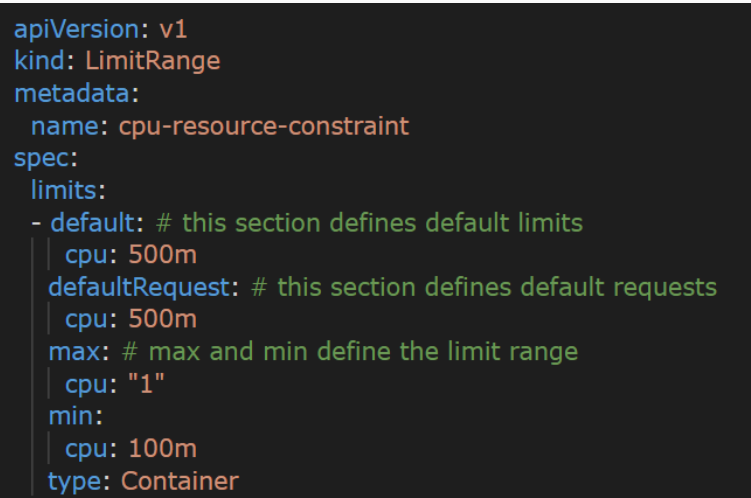
**4Q) how do you restrict pod and containers to use limited host resources?**

Ans) we can limit the resources to the pod by mentioning limits and requests YAML that can restrict containers not to use more CPU and Memory.



**5Q) how can you restrict resources to pod if the engineer forgets to mention them in pod definition?**

**Ans)** we have a resource called LimitRange in kubernetes. As administrators we can restrict the resoures at the namespace level. So if the enginners to forgets to mention limit range will be applied autometically.

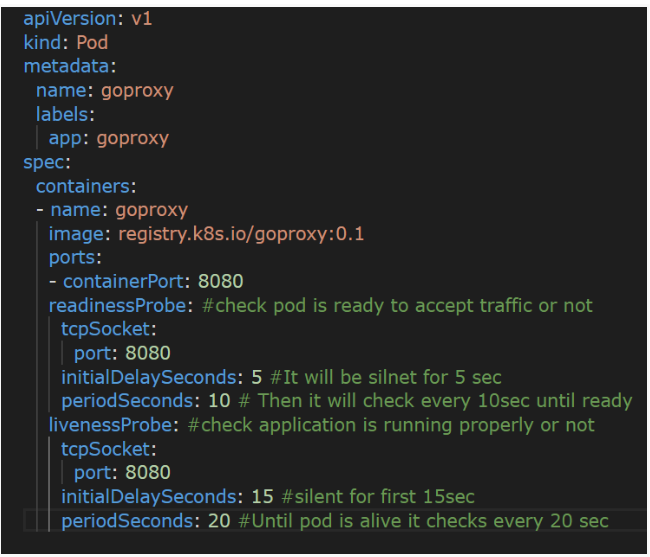


**6Q) how to configure health checks in pod?**

Ans) we can configure using readiness probe and liveness probe.

Readiness is to check at the time of container creation whether pod is ready to take traffic or not.

Liveness probe will kick in when readiness is done. It will be checked at a particular interval to check if the application is working properly or not.



**7Q) what is the image pull policy in pod?**

Ans) we have 3 types of image pull policy are available.

* IfNotPresent – if it is not present then it will pull, if available already it will not pull.
* Always – whether image is available or not it pulls the image.
* Never – if present in host it can run the pod, otherwise error.

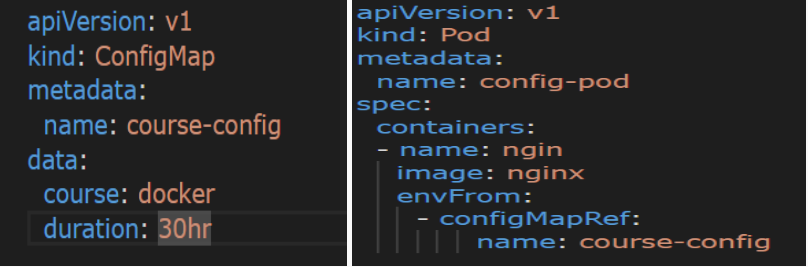
Keep always is the best way.

**8Q) how can you provide env variables to the pod in a better way?**

Ans) we can create configMap resource. We can refer to the configMap in the pod so that the entire config will be loaded at the at the time of container creation.

In future if we need to add more configurations, do the changes in configMap restart the pod so that there is no need to make changes in

Pod definition and deployment.



**9Q) what are the secrets in Kubernetes?**

Ans) secrets are confidential information like DB username, password, etc. we can store this info in secret and refer it to a Pod definition like configMap.

Secrets are base64 encoded in Kubernetes. By default it is not secure, so we use third party solutions like AWS secret manager and Hashicorp Vault, etc.



**10Q) what is service in Kubernetes? How many types are there?**

Ans) Pods are normally ephemeral in Kubernetes.so pod IP will be changed Every time it is created. To achieve pod to pod communication we can’t rely on IP addresses. Solution is service.

service can have name and port , when a pod is created it can go attached to service itself. It acts as a service mesh.

Multiple pods can be attached to service based on labels selectors, service acts as Loadbalancer between pods. 3 types of services available.

**Cluster IP**: Default type. It can get one IP address. We can configure cluster IP service to achieve pod to pod communication. But it can’t be accessed over the internet.

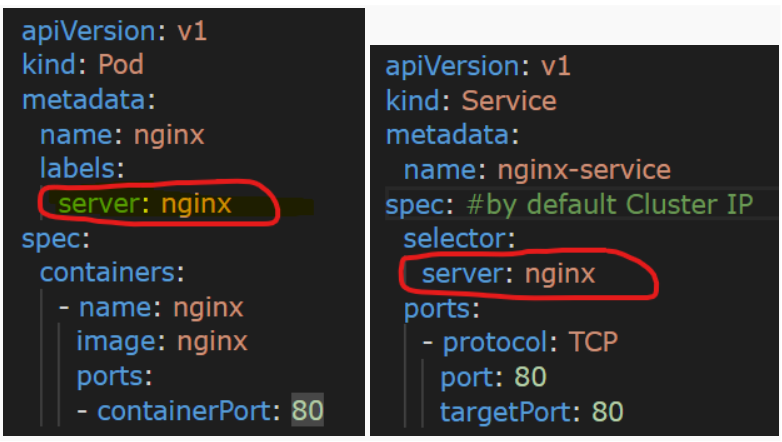
**NodePort:** Nodeport by default creates cluster IP in background. When we say Nodeport a port will be opened on each and every node. This port will be redirected to the cluster IP. NodePort can be accessed over the internet.

**Loadbalancer:** we can create Loadbalancer through cloud providers like

Aws, gcp, azure, etc.

LoadBalancer By default creates NodePort and Cluster IP in the background.

LoadBalancer 🡪NodePort 🡪 Cluster IP

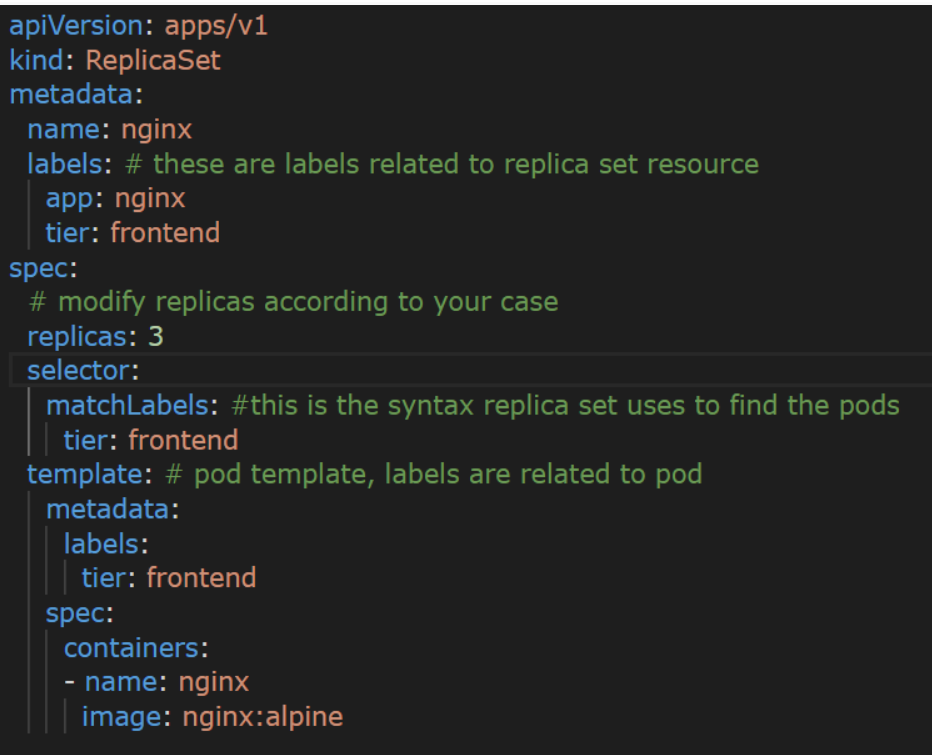


**11Q) What is replicaset in Kubernetes?**

**Ans)** by default it can increase the number of pods, if you want multiple

Instances of pods to serve high traffic we can use replicaset. It guarantees the declared no of pods will run always. It is most important feature of high availability and autoscale.

When using ReplicaSet we no need to have separate pod, it is included Under template section.



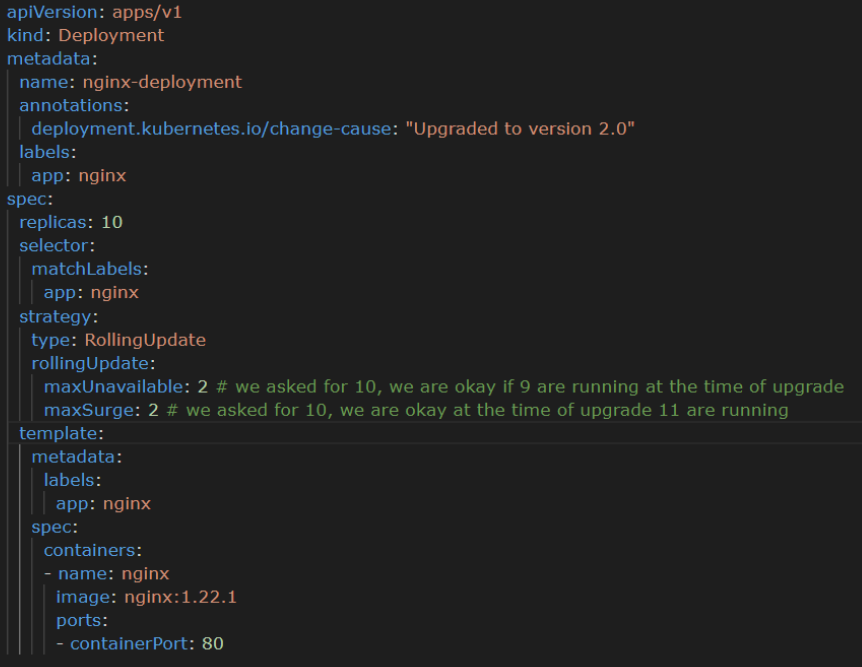
**12Q) what is the deployment in kubernetes?**

Ans) Deployment is most important resource in kubernetes maintain the applications. It ensures zero downtime of application. Deployment

Creates ReplicaSet in background to maintain number of replicas.

We can use deployment for stateless applications. Finally, deployment is the highest resource in kubernetes to maintain

* Availability – Make sure desired number of replicas are always available.
* Scalability – can be used to scale at runtime based on traffic
* Maintainability – we can do the updates with new image versions



**13Q) what is Daemon set in kubernetes?**

Ans) Daemon set is similar to deployment. But there is only one difference.

Deployment makes pods available in any node.

But Daemon set makes sure one replica of pod runs in each and every node in kubernetes cluster.

This feature is helpful for Cluster level administration like collecting the metrics, logs of all nodes, etc.

