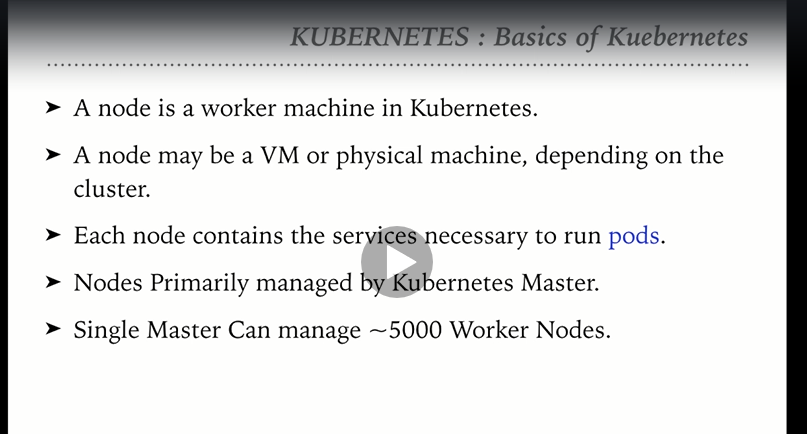
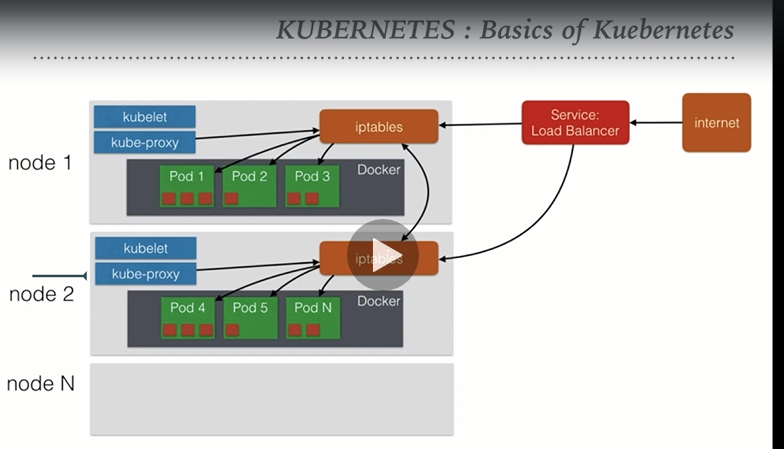
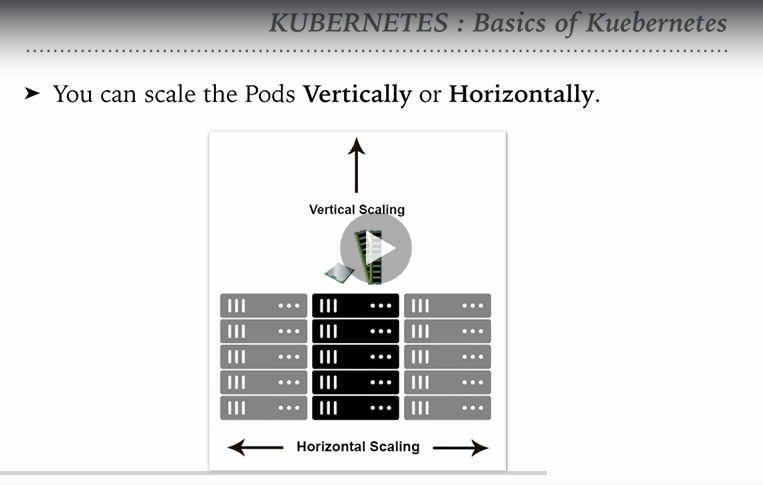
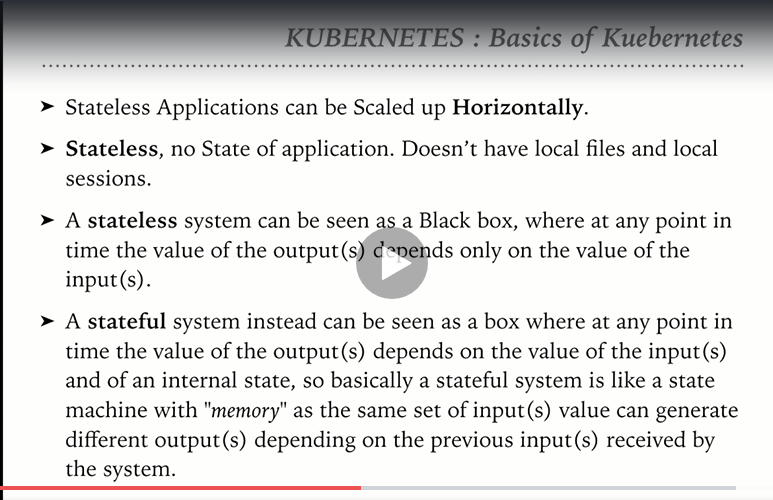
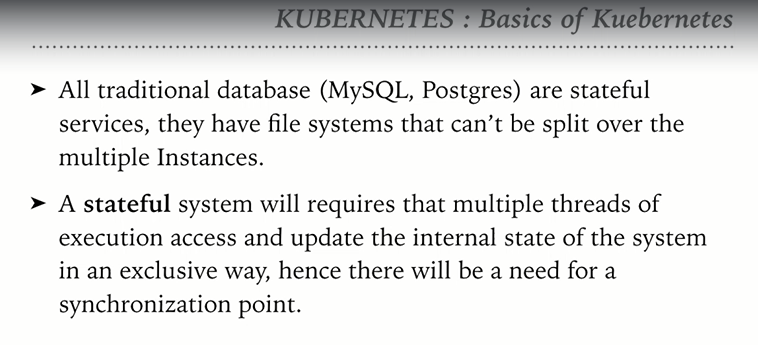
**Topic-1: Kubernetes NodeWorkflow**

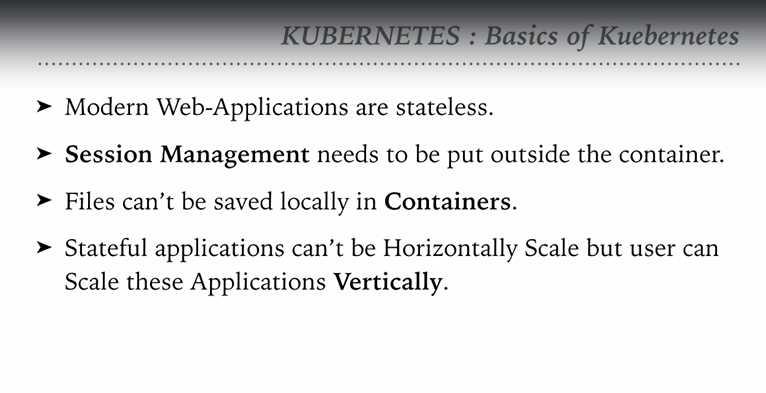


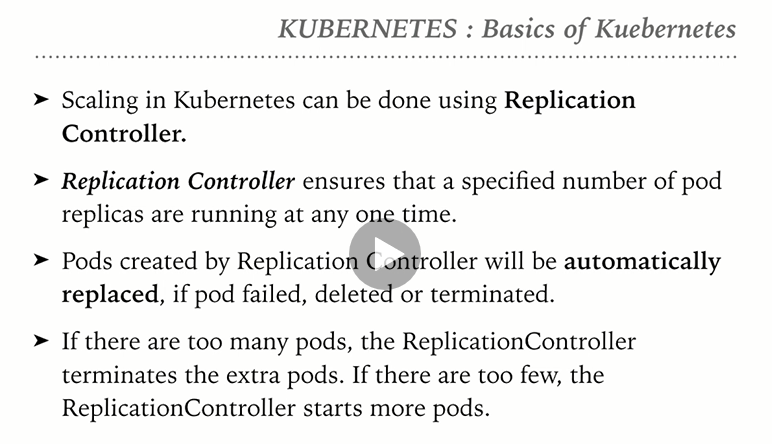
**Topic-2: Scaling PODs in Kubernetes**

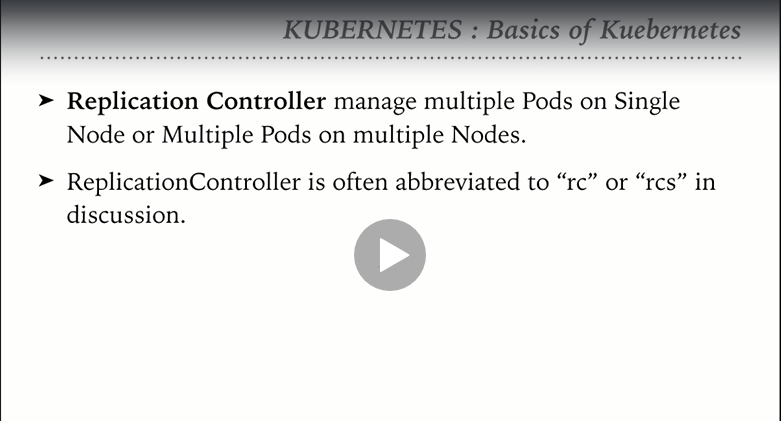




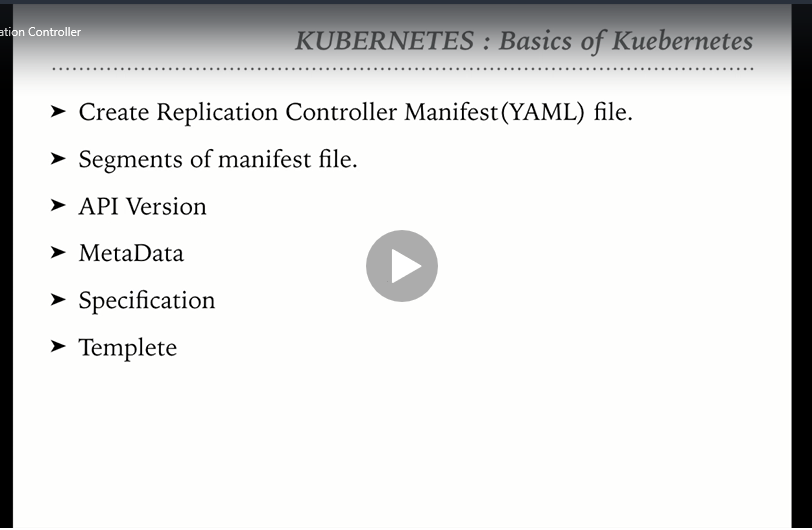


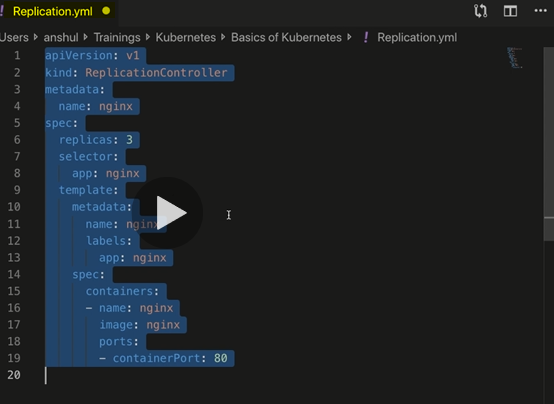




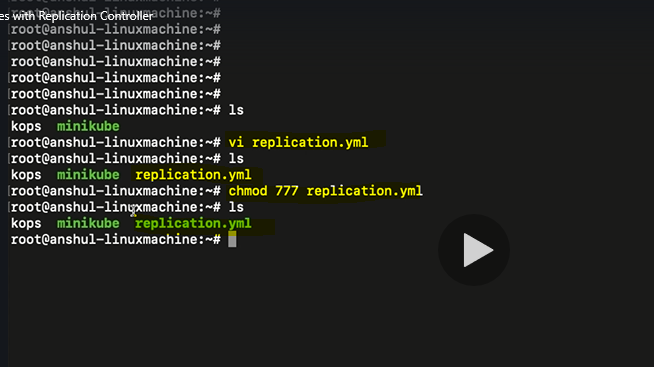


**Topic-3: Scaling pods with replication controller**

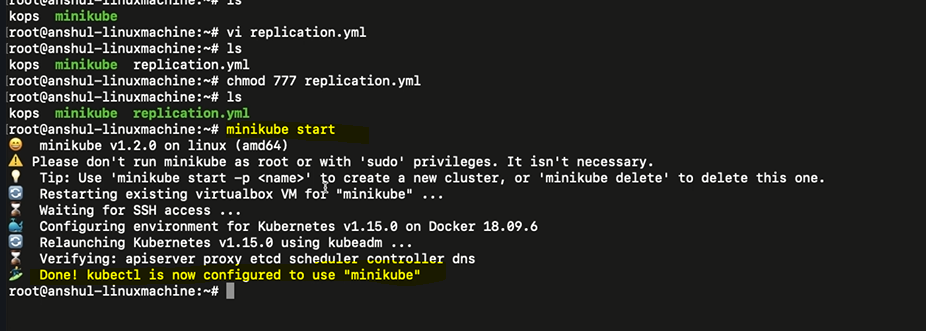


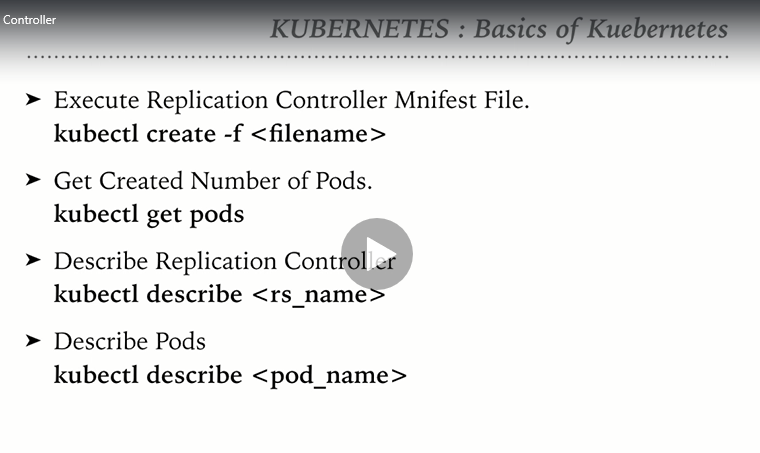


Create a Replication.yml file



Start minikube

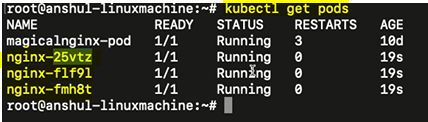




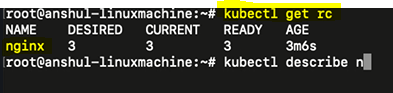
Create a replication controller



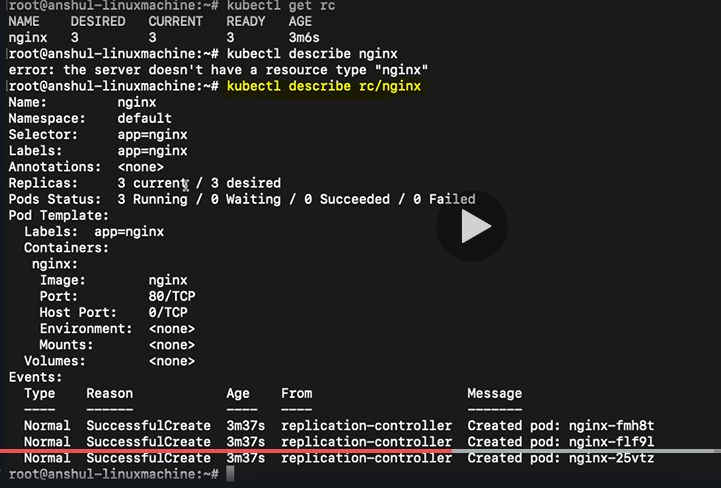
Get Pods : 3 pods were mentioned in the replication controller



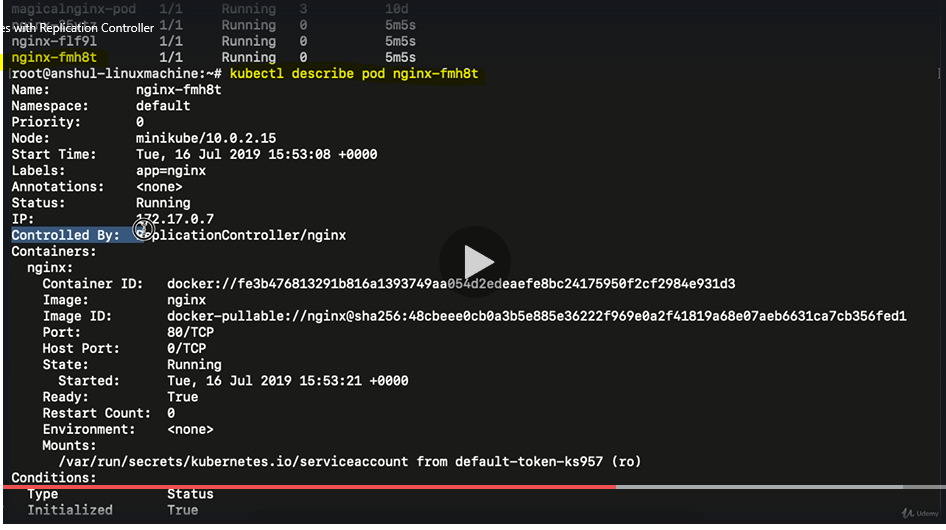
Get replication controller



Describe replication controller

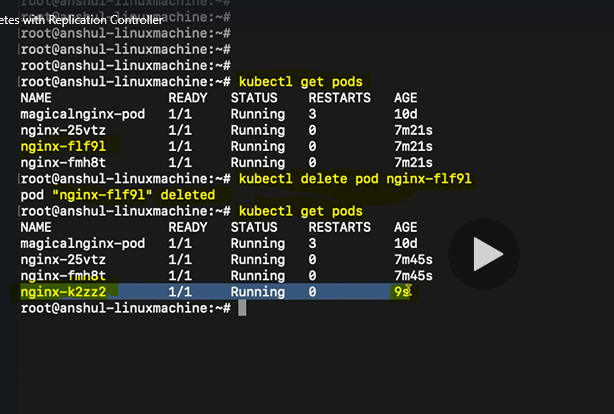


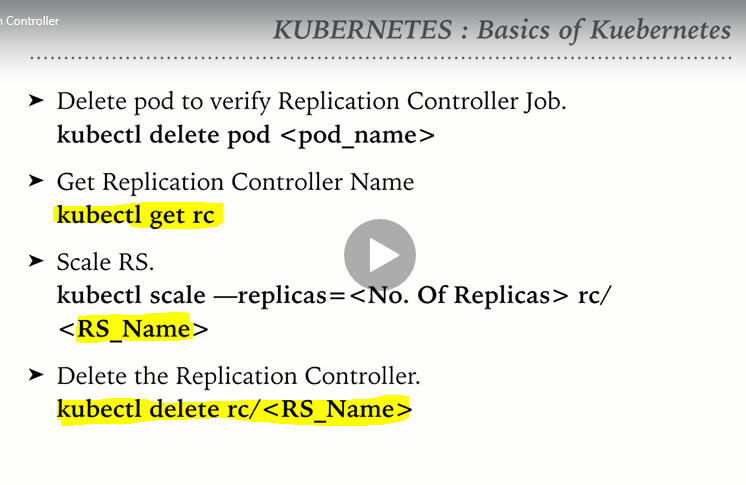
Describe pods



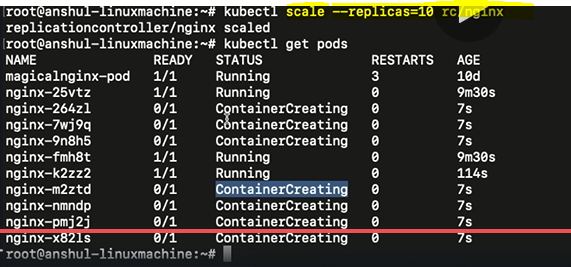


Delete a pod manually and verify if the Replication controller spin up a new pod

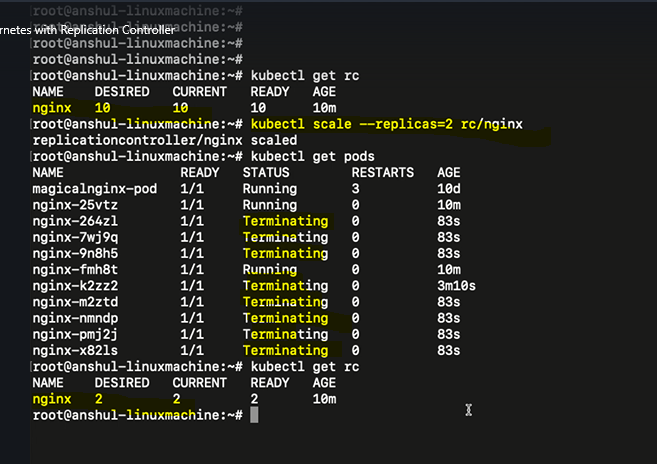




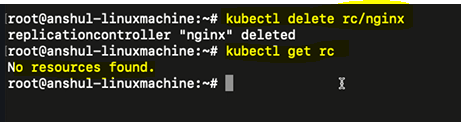
How to scale up the number of replicas in an existing RC:



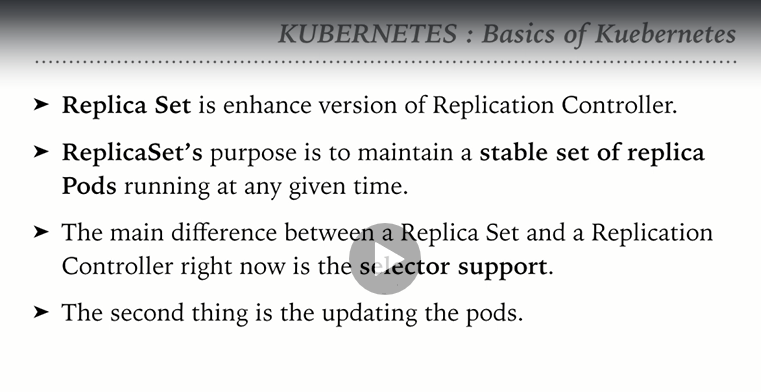
Scale down the replicas

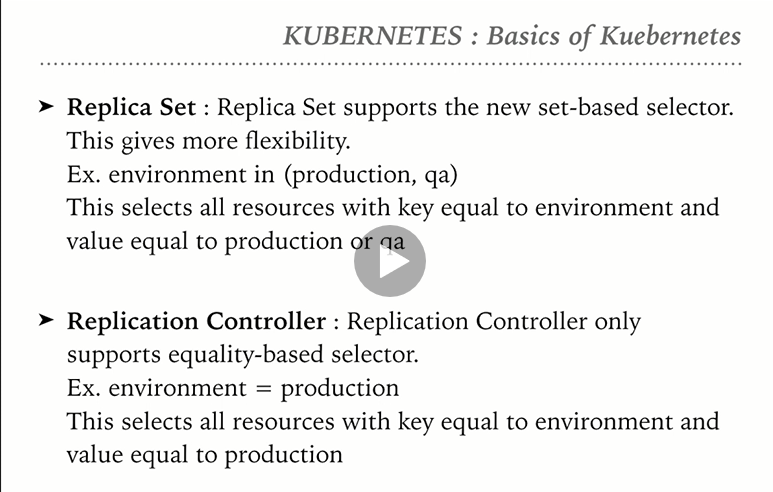


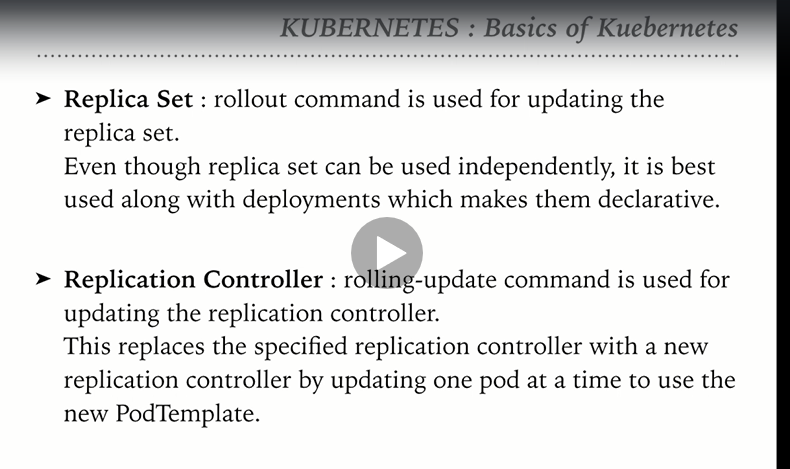
How to delete a replication controller



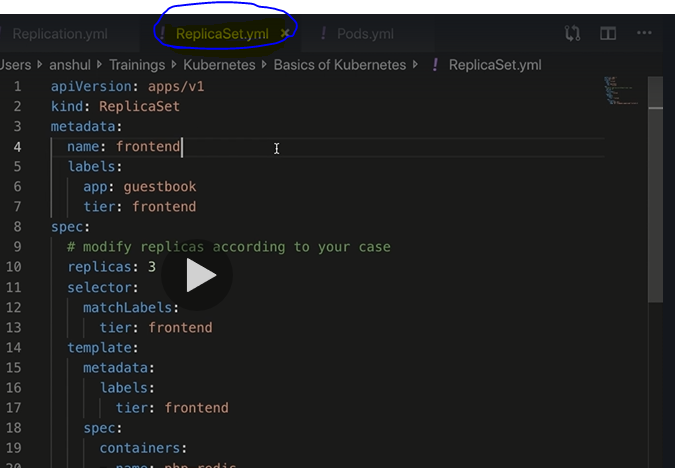
**Topic-4: Replica set in Kubernetes**



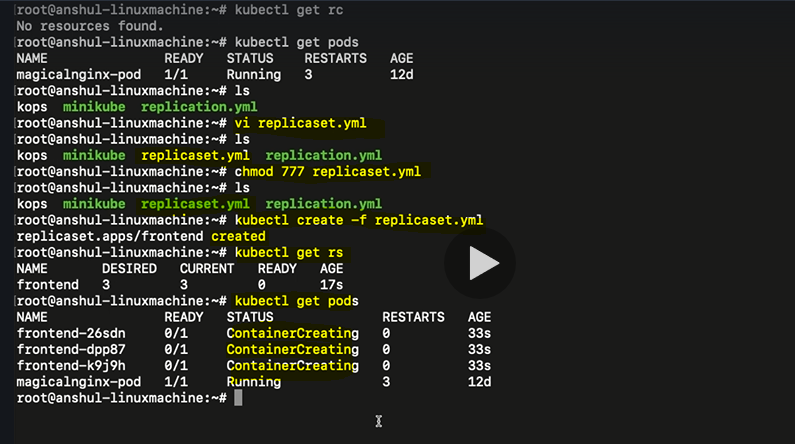




Create a Replicaset.yml file

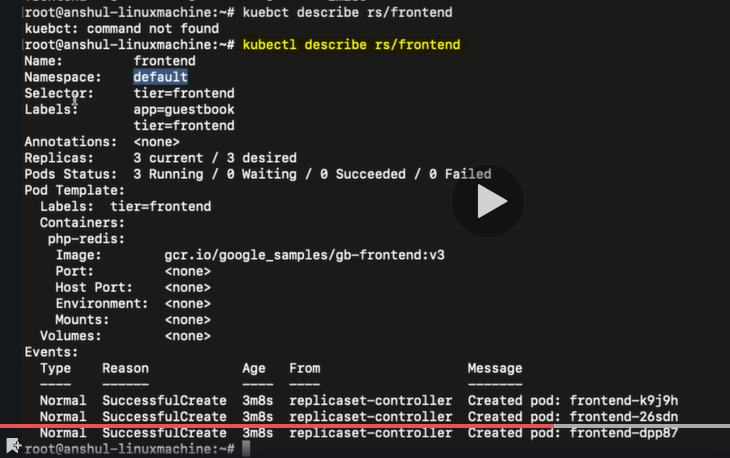


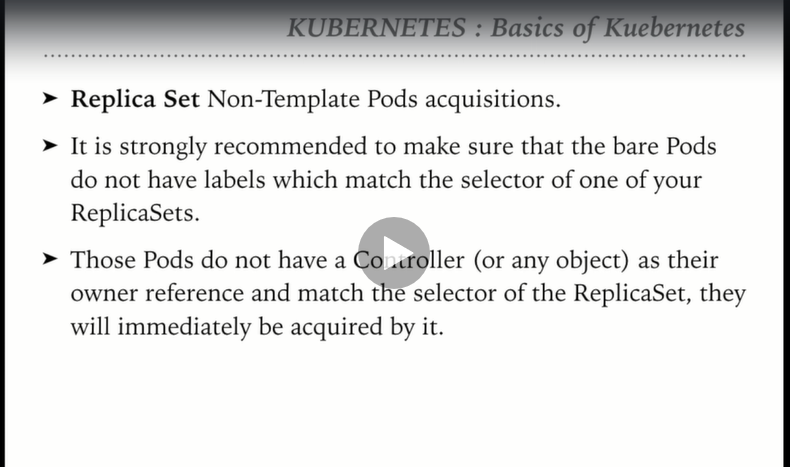
Create a replicaset and view the status:



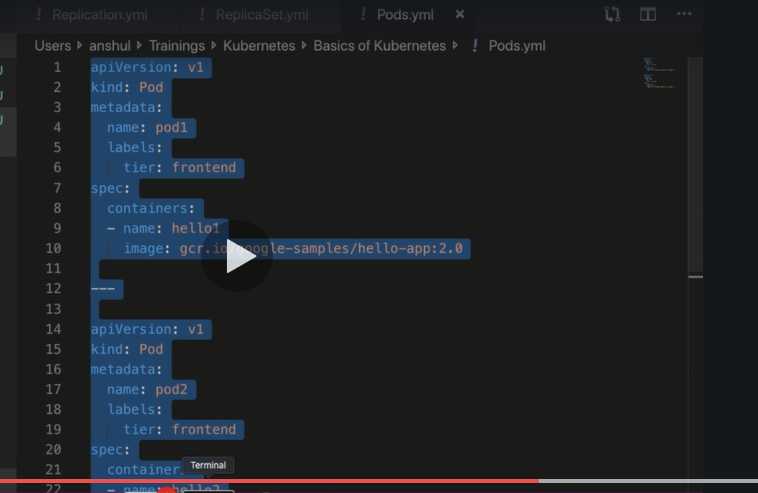
Note: Replicas refer to pods

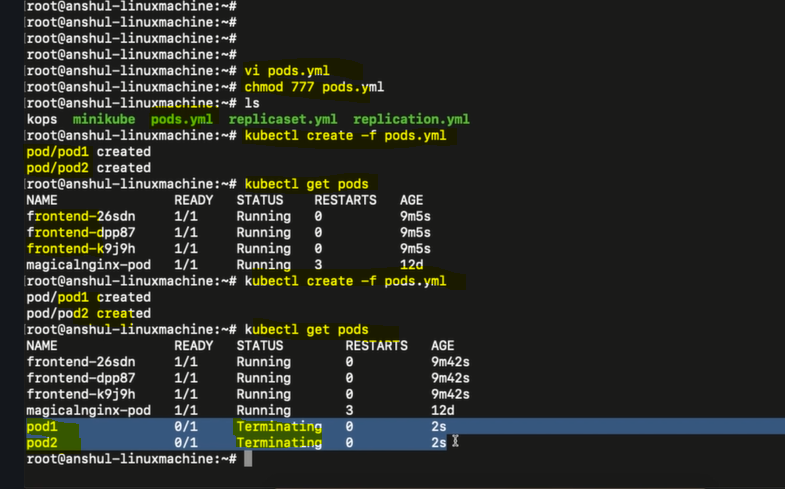
To describe replicaset





Create pod



As the bare pods had the same label as that of the replica set then the bare pods are acquired by th e replica set. That is the reason, even though bear pods are cretaed, they are getting terminated

Create the bare pods first and then create the replicaset with the same label. We can onserve that only one pod of the replicas set is created as there are already 2 bare pods of the same specification are running.

