**DEPLOYMENTS**

* A **ReplicaSet** is a declarative way to manage Pods. Think o fit as kind of the boss of the pods that sits behind the scenes to make sure that they are working efficiently and if one of them gets sick, we can get that replaced.
* **Deployment** is a declarative way to manage Pods using a ReplicaSet. Sits up at a higher level and kind of wraps a ReplicaSet. It is also a declarative way to manage Pods but it uses ReplicaSets under the covers.
* **Pods** can be created and destroyed but never re-created. What happens if a Pod is destroyed? Deployments and ReplicaSets ensure Pods stay running and can be used to scale Pods.
* A screenshot of a cell phone

  Description automatically generated
* A screenshot of a cell phone

  Description automatically generated
* A screenshot of a cell phone

  Description automatically generated
* **kubectl create -f file.deployment.yml** to create a deployment
* **kubectl apply -f file.deployment.yml** to create/update a deployment
* **kubectl apply -f file.deployment.yml –save-config**
* **kubectl get deployments** for listing all deployments
* **kubectl get deployments –show-labels** list all labels
* **kubectl get deployments -l app=nginx** get all deployments with a specific label
* **kubectl delete deployment deployment\_name** will delete all associated pods/containers
* **kubectl scale deployment deployment\_name –replicas=5 –** this tells kubernetes that you want five replicas, in other words, five pods
* **Zero downtime deployments** allow software updates to be deployed to production without impacting end users
* it can bring up new pods and once those are running, kill of the old pods and re-route the network traffic
* several other options:
  + **rolling updates**
  + **blue-green deployments –** is where you are going to have multiple environments running exactly at the same time and then once you have proven that the new one is good, you will switch all the traffic over to the new one
  + **canary deployments –** will be where a very small amount of traffic goes to a new deployment and then once thats proven out by the users hitting it, you can switch all the traffic over to that one
  + **rollbacks –** would be, we have tried it, it did not work, lets go back to the previous version