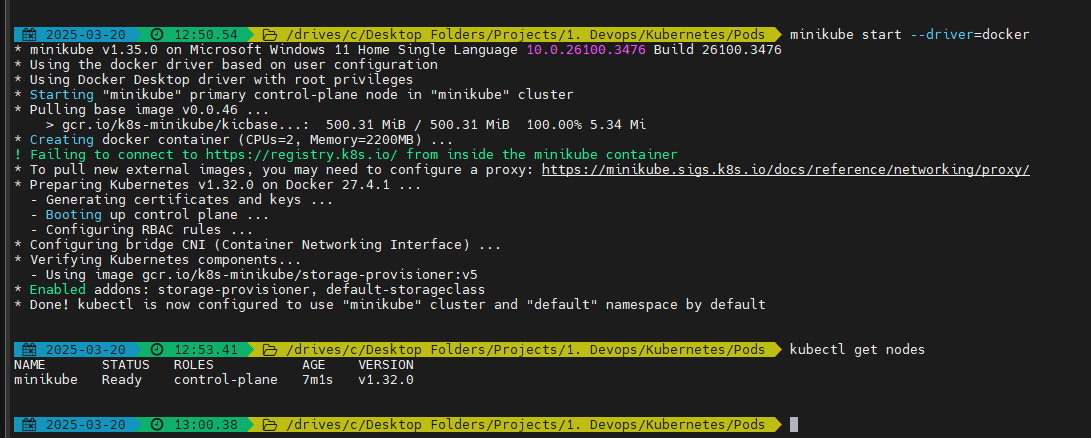
Make sure you’re starting docker desktop to start docker engine to start the minikube.

If we use mac or windows, the minikube will start by creating a VM and then create single node k8s clusters.

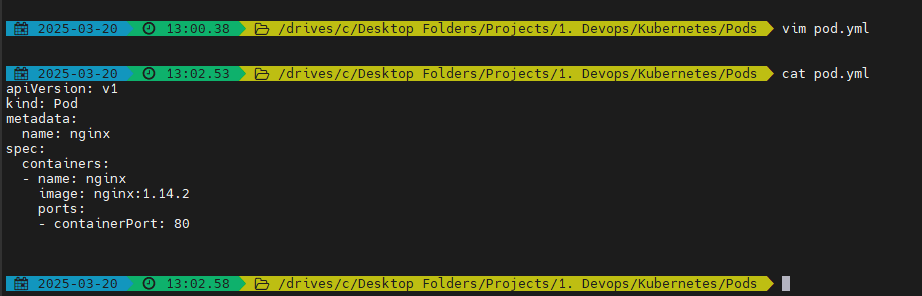
* Use:
  + minikube start –driver=docker
  + kubectl get nodes



* We better not use the default docker driver when we move to advance k8s concepts

Now you can go to: <https://kubernetes.io/docs/concepts/workloads/pods/>

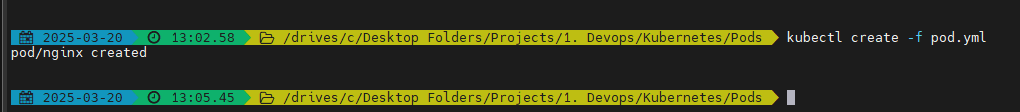
And create a pod by creating pod.yml



* Docker equivalent: docker run -d nginx:1.14.2 –name nginx -p 80:80

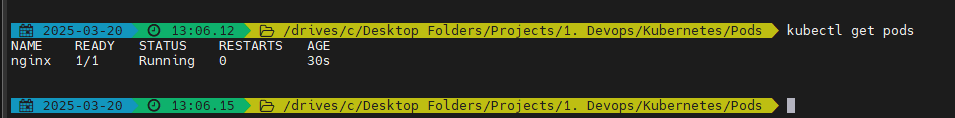
Now create the pod by using:

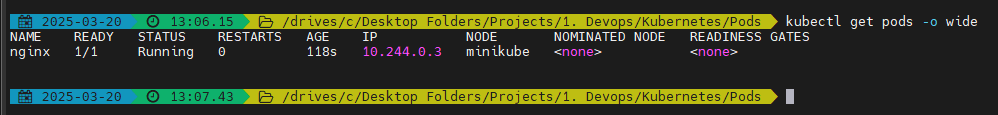
* kubectl create -f pod.yml



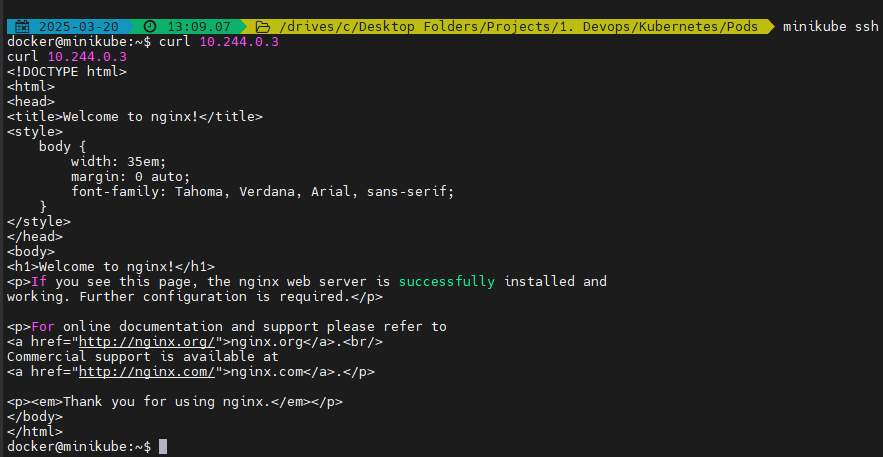
Get the pod details by using:

* kubectl get pods
* kubectl get pods -o wide





Now we can curl to the application to check if the application is running:

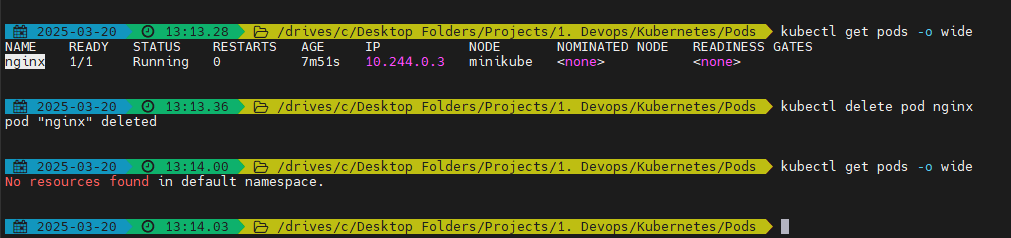


Note: if we are using real time cluster, we can ssh to master or any worker node IP address.

You can always refer: <https://kubernetes.io/pt-br/docs/reference/kubectl/cheatsheet/>

You can delete the pod using:

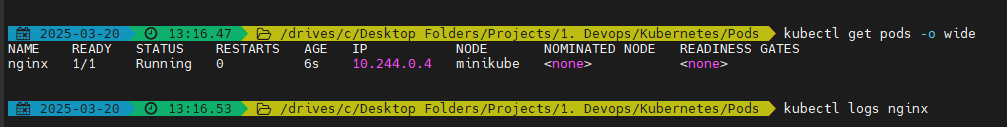
* kubectl delete pod nginx



We can add deployment to add capabilities like autoscaling and autohealing to kubernetes

We can debug the pod using:

* Currently no logs
* kubectl logs nginx



* kubectl describe pod nginx

