1. Make sure you have a virtualbox with Ubuntu installed.

The machine must have at least 20GB of storage.

If it does not have that, please reinstall the VM.

**Done**

2. Install Docker

**Done**

3. Install MiniKube:

https://minikube.sigs.k8s.io/docs/start/

**Done**

4. Create a POD with Nginx using kubectl command

**kubectl run my-nginx --image=nginx --port=80**

5. Create a POD with Nginx using a yaml file.

**a. nginx-deployment.yaml:**

תמונה שמכילה טקסט

התיאור נוצר באופן אוטומטי

**b. apply the deployment:**

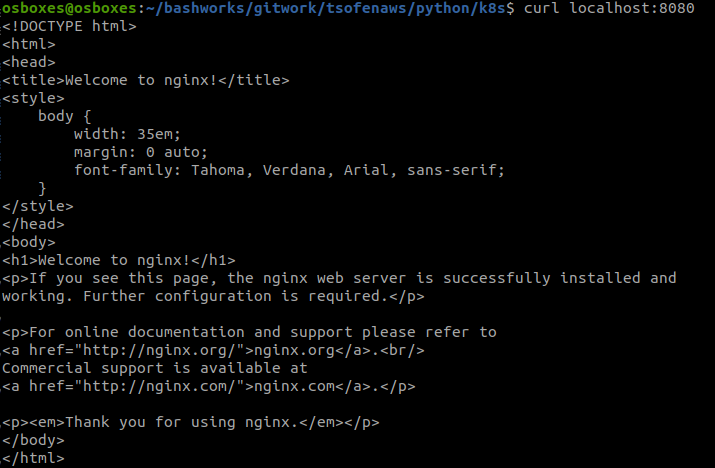


6. Using kubectl port-forward access the nginx web page.

**a. port-forward:**



**b. able to access the page:**



7. Edit the image of the container in the pod and deploy it.

**a. pod name:**



**b. current image:**



**c. update image:**



**d. image is update:**



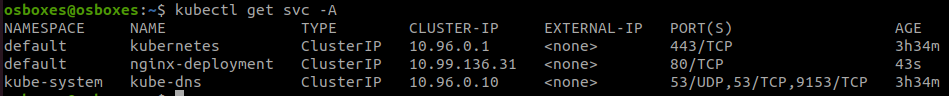
**e. other option is to update the yaml and then to apply it again.**

8. Create a ClusterIP service and connect it to the pod.

**a. expose:**



**b. get services:**



**c. port-forward:**



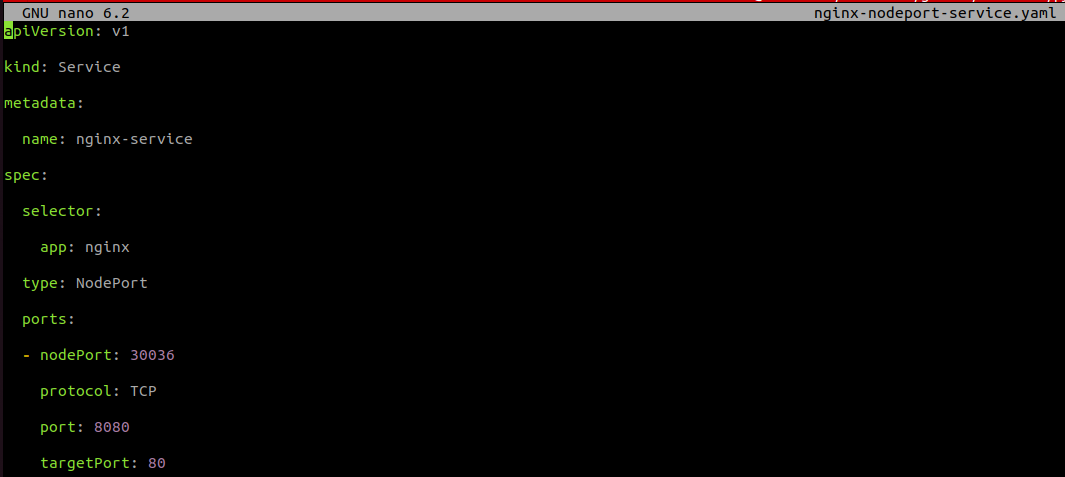
**d. able to access the page (from browser also):**

תמונה שמכילה טקסט

התיאור נוצר באופן אוטומטי

9. Create an external service (nodeport or loadbalancer) and connect it to the POD.

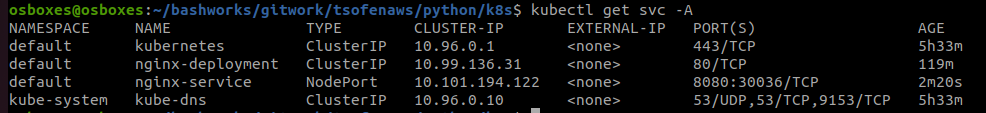
**a. nginx-nodeport-service.yaml:**



**b. apply the service:**



**c. get services:**



10. Create a deployment instead of the pod.

- Why should we use deployment and not ReplicaSet or POD objects?

**Deployment done in ex #5.**

**Deployment is better than ReplicaSet because of the following:**

**In Deployment when we change the yaml file and updating the image for example and doing apply again, we can see the pods are working in the updated image.**

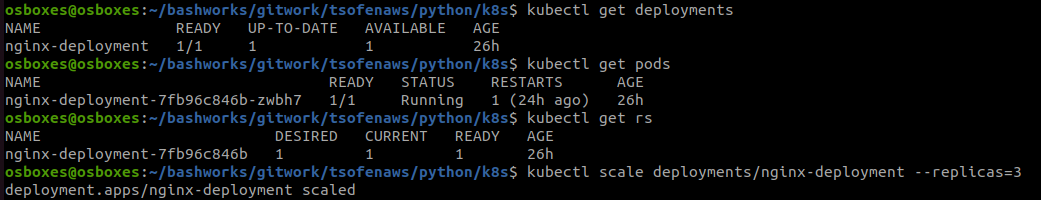
**In ReplicaSet we don't see any changes in Pod and they are still using old image.**

11. Connect your external services to the deployment.

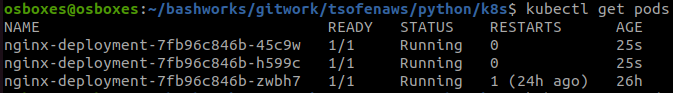
**Done in ex #9.**

12. Scale your application to 3 instances.

**a. scale to 3 instances:**



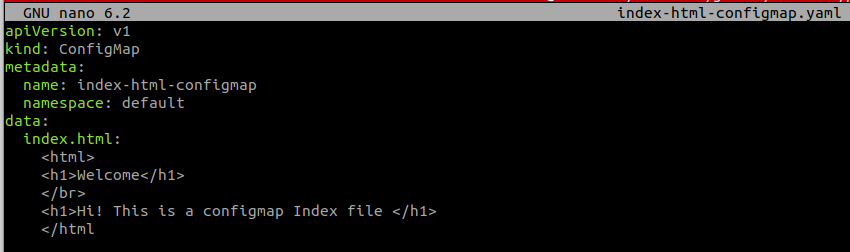
**b. 3 pods are running:**



13. Go inside each pod and edit the index.html file and try to access the nginx web site.

- Did it change?

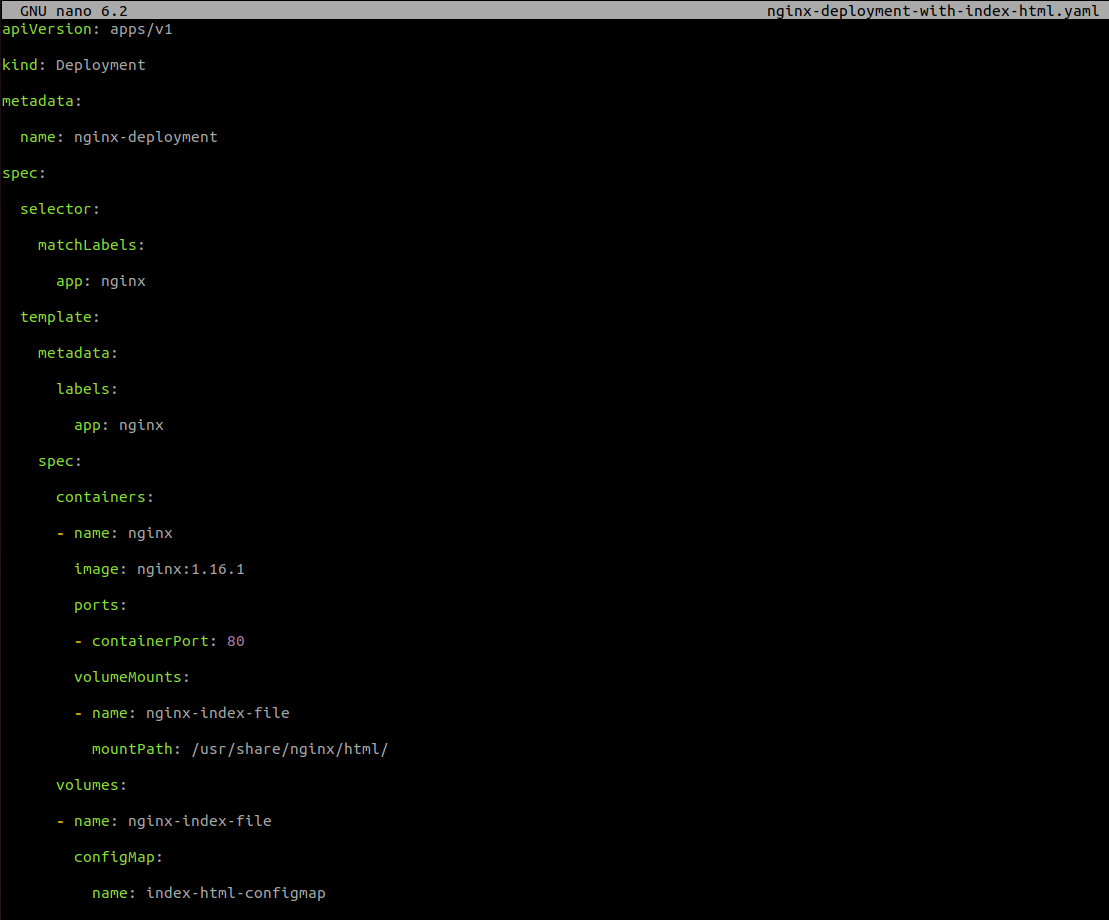
**a. index-html-configmap.yaml:**

****

**b. apply the config map (index-html-configmap.yaml) with the index.html:**

**kubectl apply -f index-html-configmap.yaml**

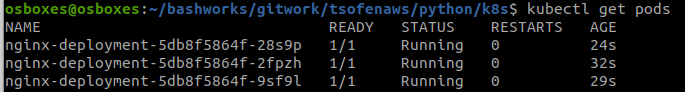
**c. nginx-deployment-with-index-html.yaml:**

****

**d. apply the deployment (nginx-deployment-with-index-html.yaml):**

****

**e. new pods are running:**

****

**g. port-forward:**

****

**g. yes working:**

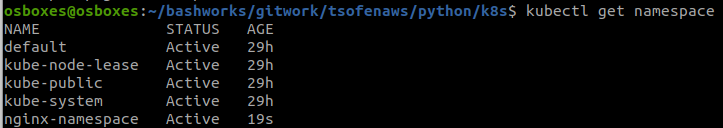
****

14. Create a namespace for the new deployment and move the deployment created to that namespace.

**a. namespace apply:**



**b. get namespace to see that namespace is created:**



**c. move the deployment to the new namespace:**



15. Can you use the same service to connect to the deployment now?

**Yes, but we need to update the namespace inside the service yaml.**

16. Create a 3 yaml files that hold all the configuration created:

**- Namespace**

<https://github.com/salmanmak1/tsofenaws/blob/06cf9cb4450a27580185201b4cfb64af123dc2bc/python/k8s/nginx-namespace.yaml>

**- Deployment**

<https://github.com/salmanmak1/tsofenaws/blob/06cf9cb4450a27580185201b4cfb64af123dc2bc/python/k8s/nginx-deployment-with-index-html.yaml>

**- Service**

<https://github.com/salmanmak1/tsofenaws/blob/06cf9cb4450a27580185201b4cfb64af123dc2bc/python/k8s/nginx-nodeport-service.yaml>

Good luck!