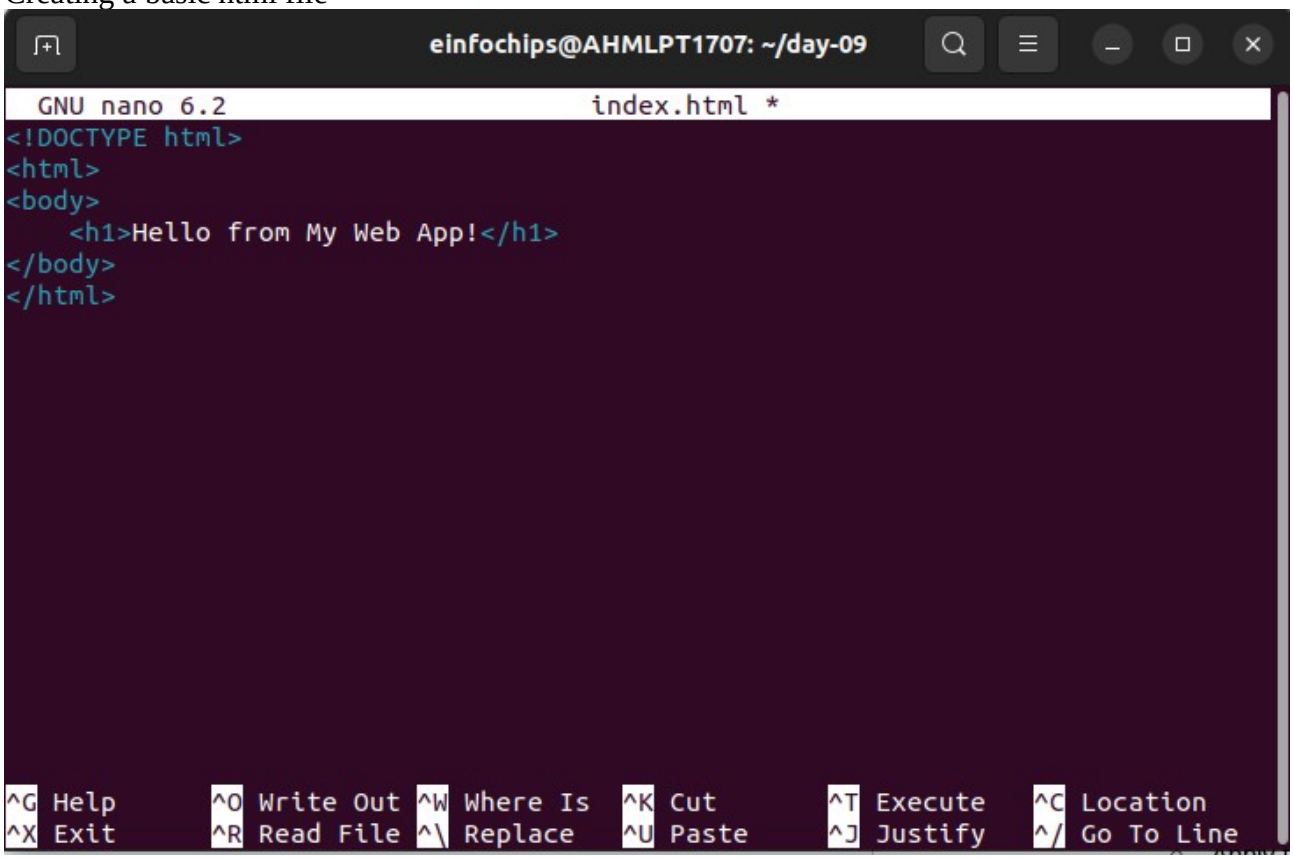
Day 09- Kubernetes ingress, URL rewriting, sticky sessions, and autoscaling.

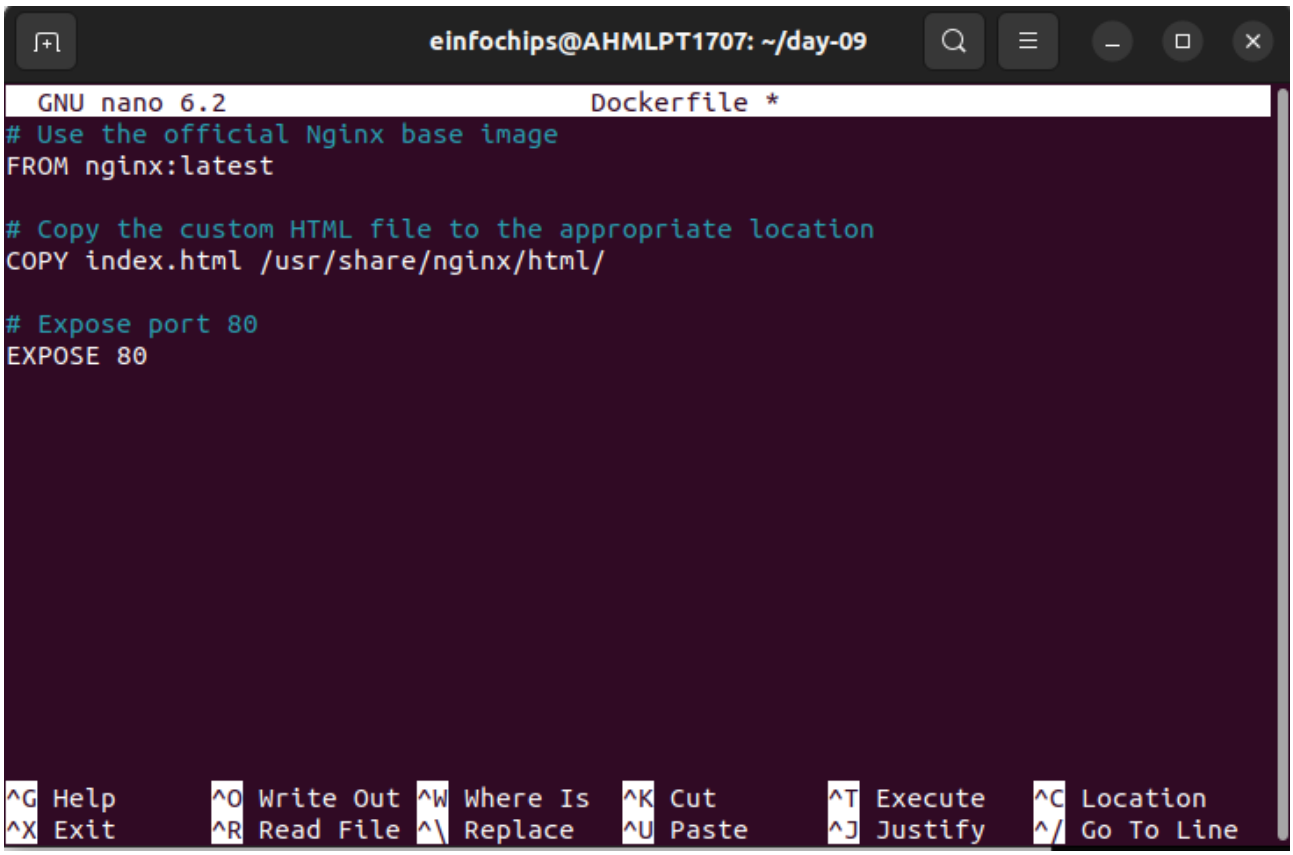
1: Setting Up the Kubernetes Cluster and Static Web App

```
einfochips@AHMLPT1707:~$ mkdir day-09
einfochips@AHMLPT1707:~$ minikube start
😄  minikube v1.33.1 on Ubuntu 22.04
✨  Using the docker driver based on existing profile
👍  Starting "minikube" primary control-plane node in "minikube" cluster
🚜  Pulling base image v0.0.44 ...
🏃  Updating the running docker "minikube" container ...
🐳  Preparing Kubernetes v1.30.0 on Docker 26.1.1 ...
🔎  Verifying Kubernetes components...
    ▪ Using image gcr.io/k8s-minikube/storage-provisioner:v5
🌟  Enabled addons: storage-provisioner, default-storageclass
🏄  Done! kubectl is now configured to use "minikube" cluster and "default" name
space by default
einfochips@AHMLPT1707:~$
```

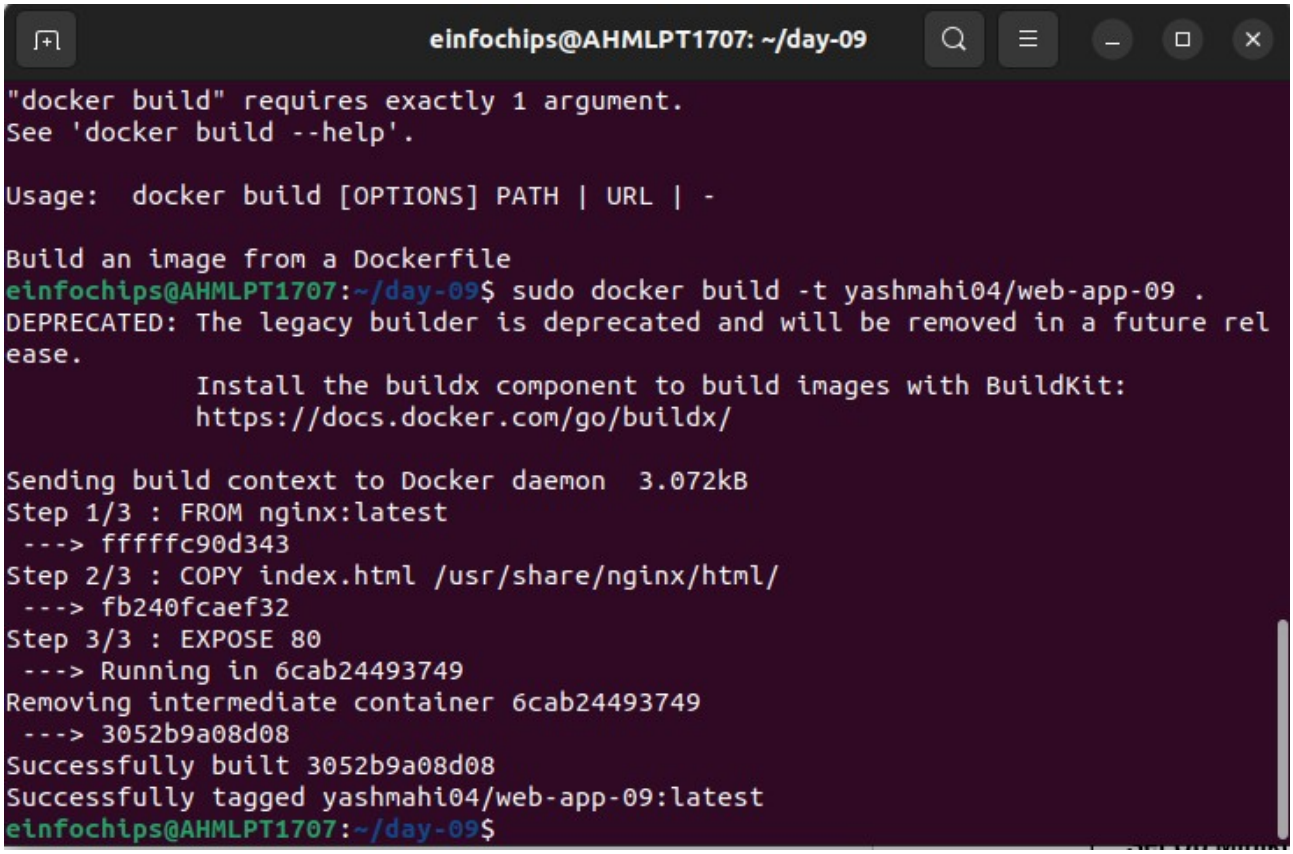
Creating a basic html file



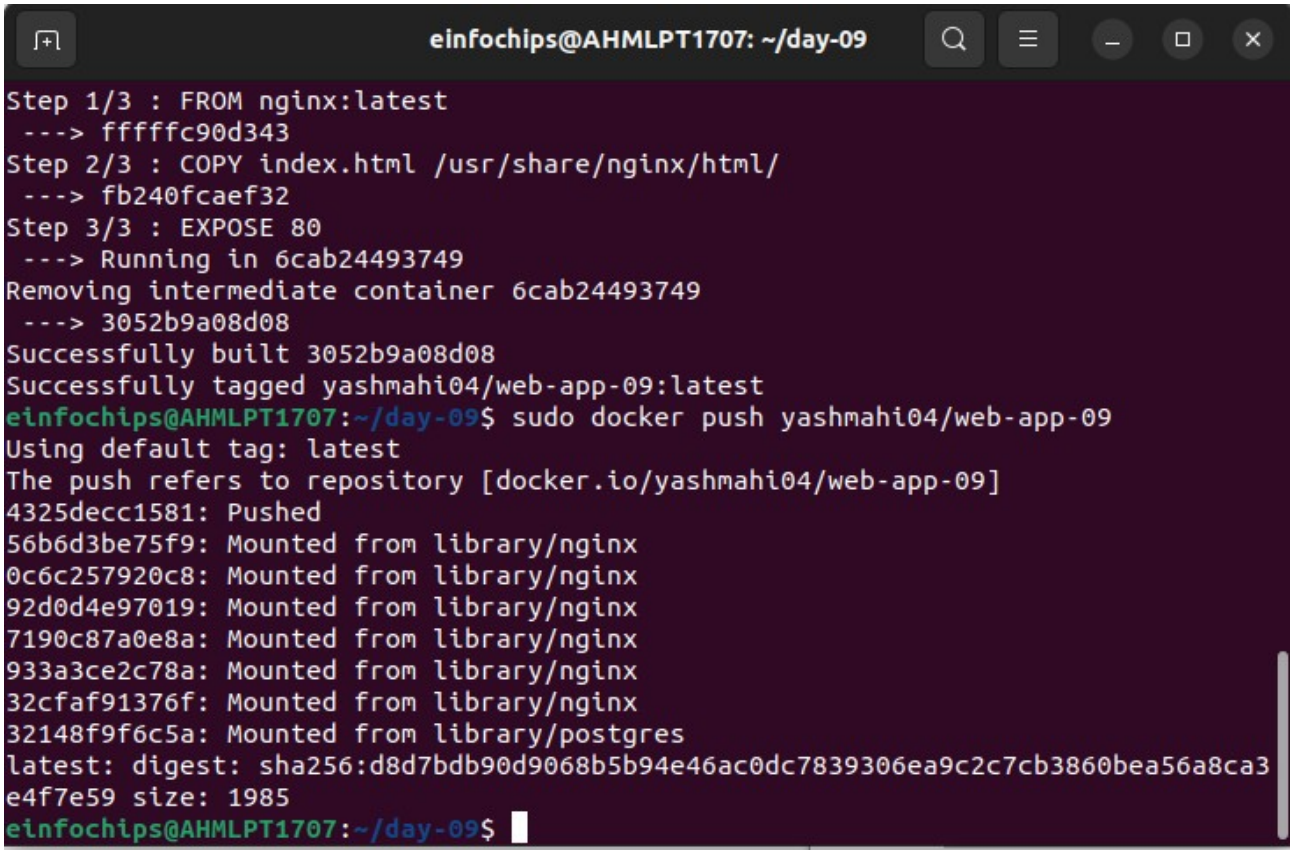
creating docker file



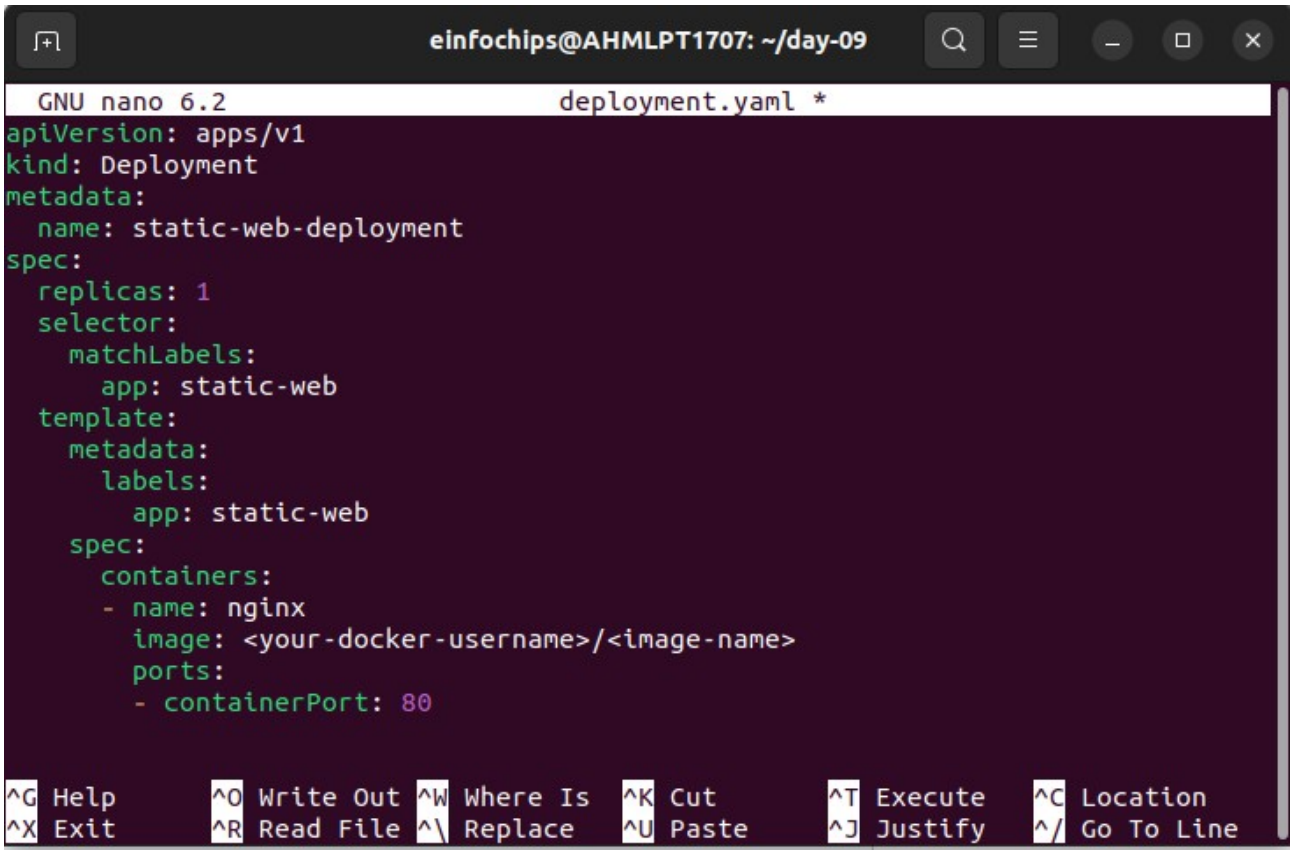
Building the dockerhub image



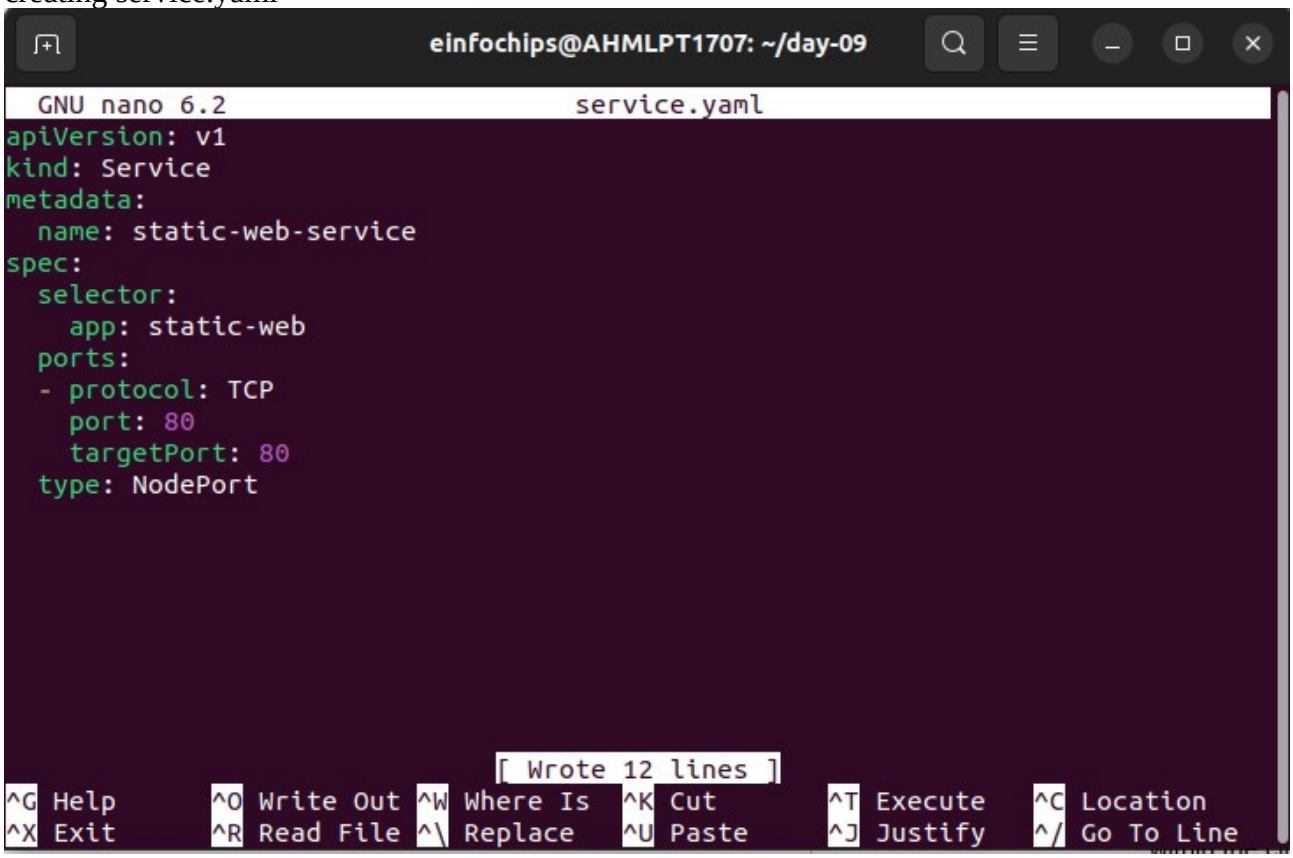
Pushing the image to the docker hub



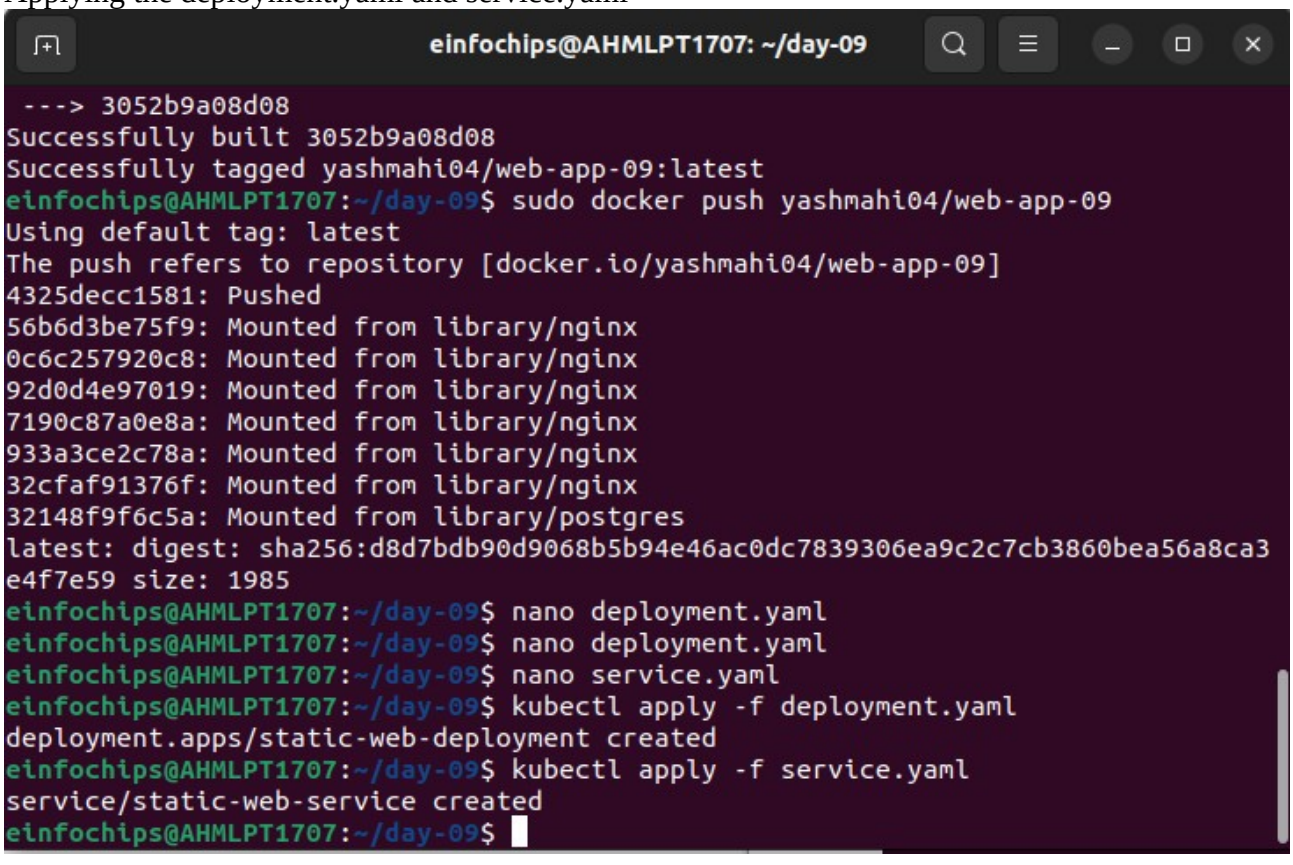
Kubernetes Deployment:
creating deployment.yaml



creating service.yaml

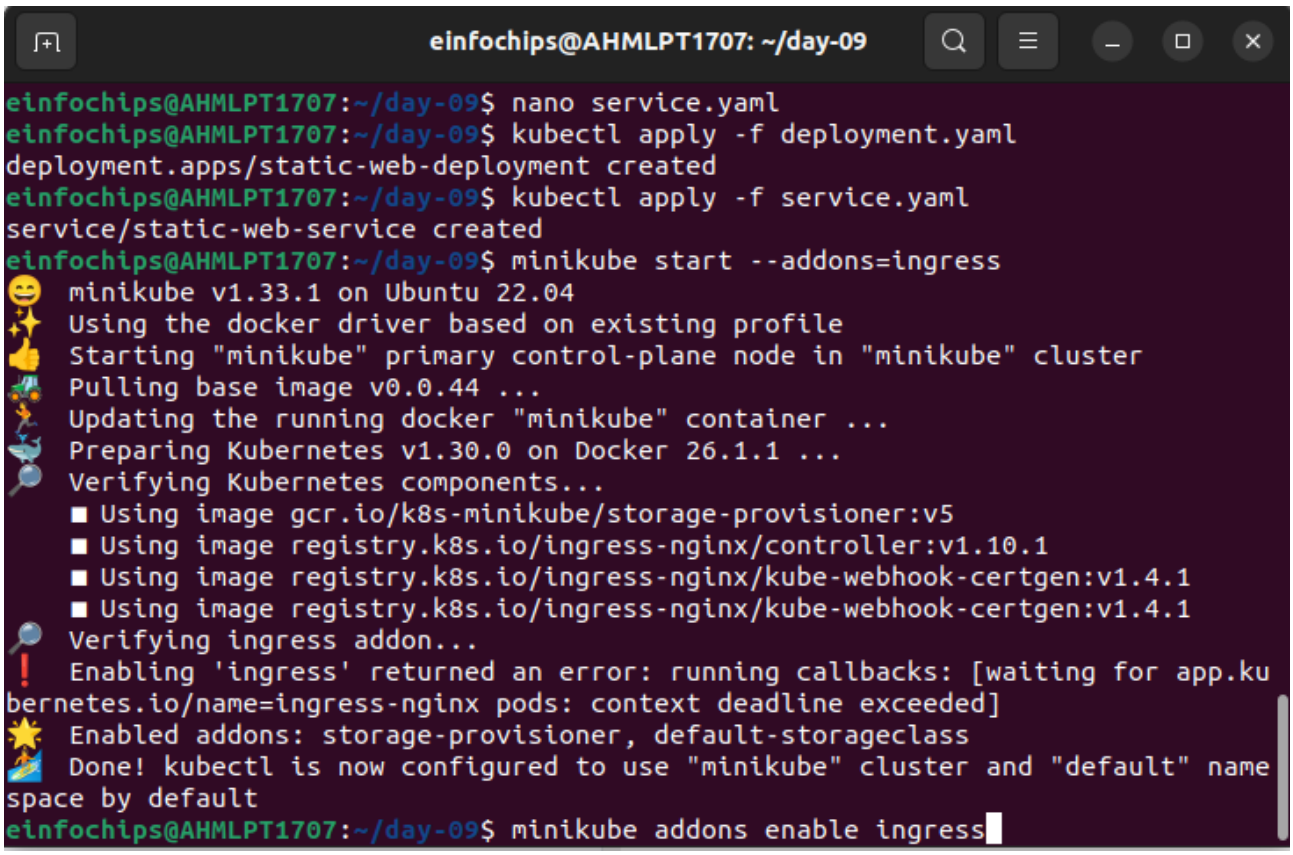


Applying the deployment.yaml and service.yaml

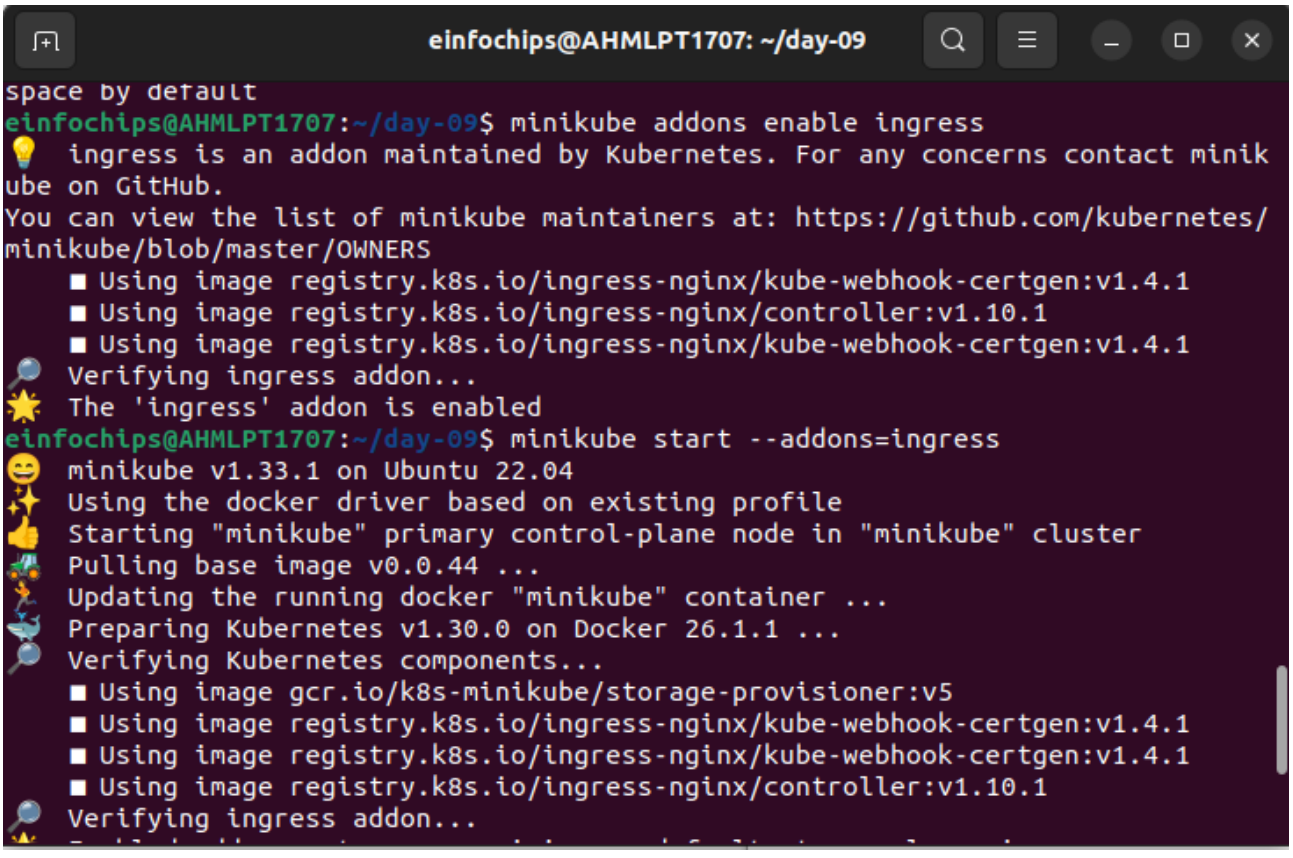


2: Configuring Ingress Networking

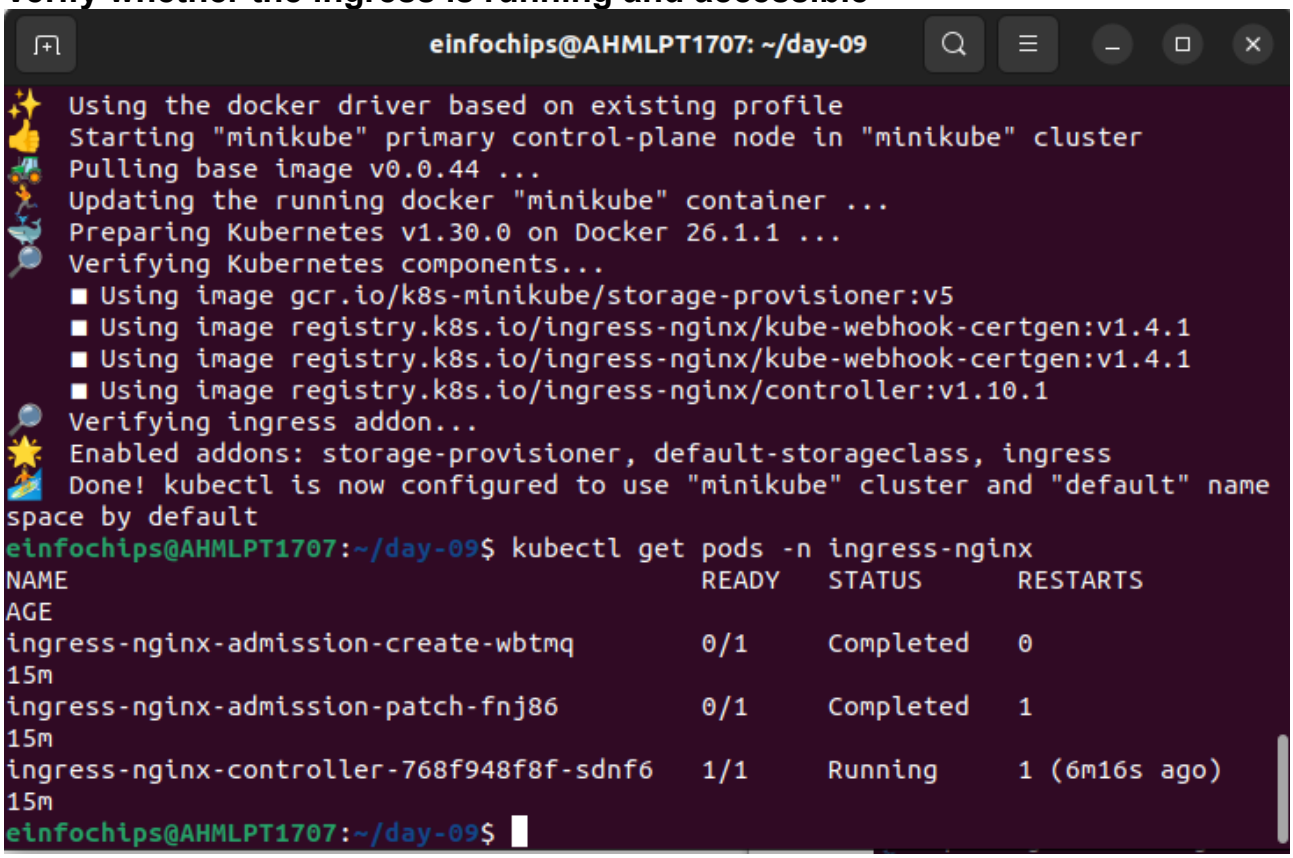
Tried to install the ingress using the command
minikube start –addons=ingress
Error occurred



In order to solve this we have to enable the ingress
minikube addons enable ingress

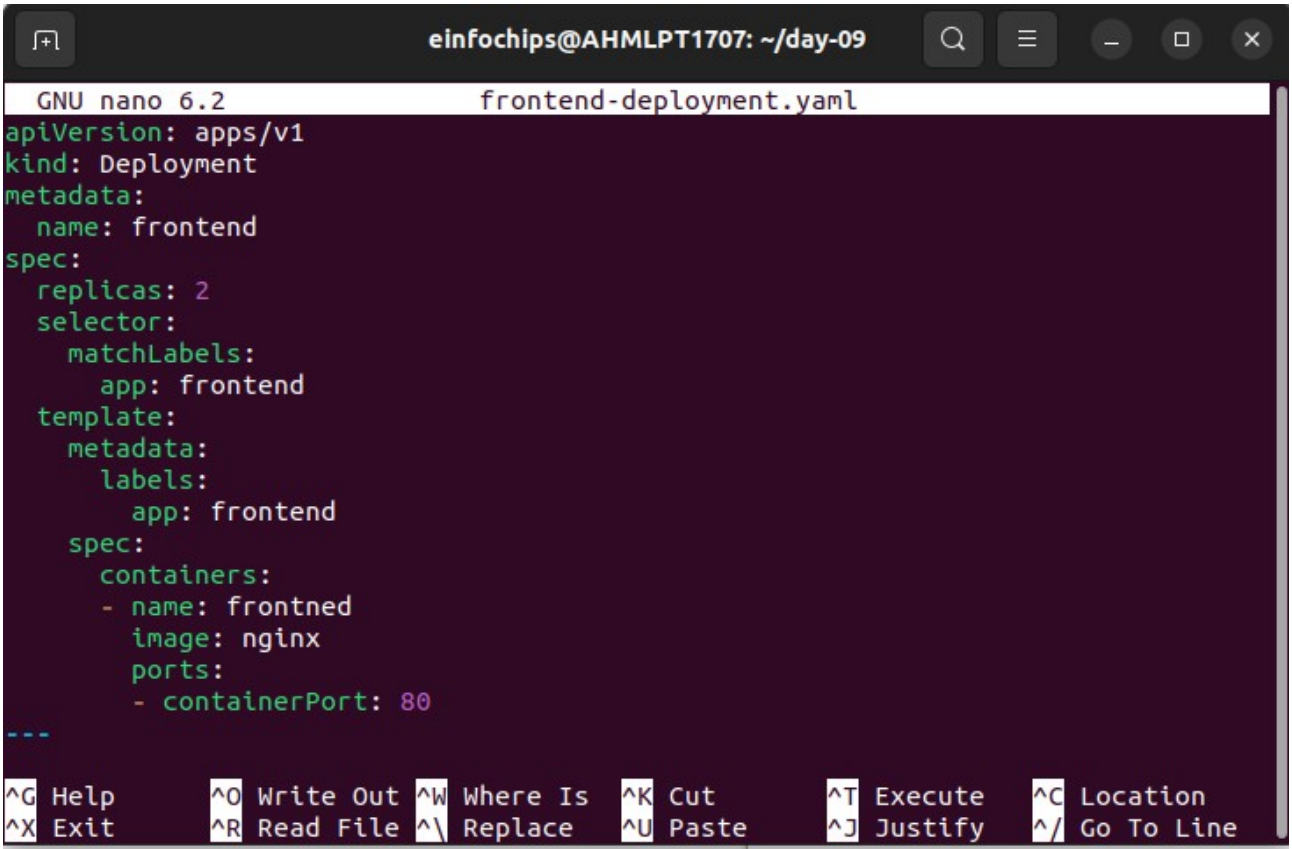


Verify whether the ingress is running and accessible

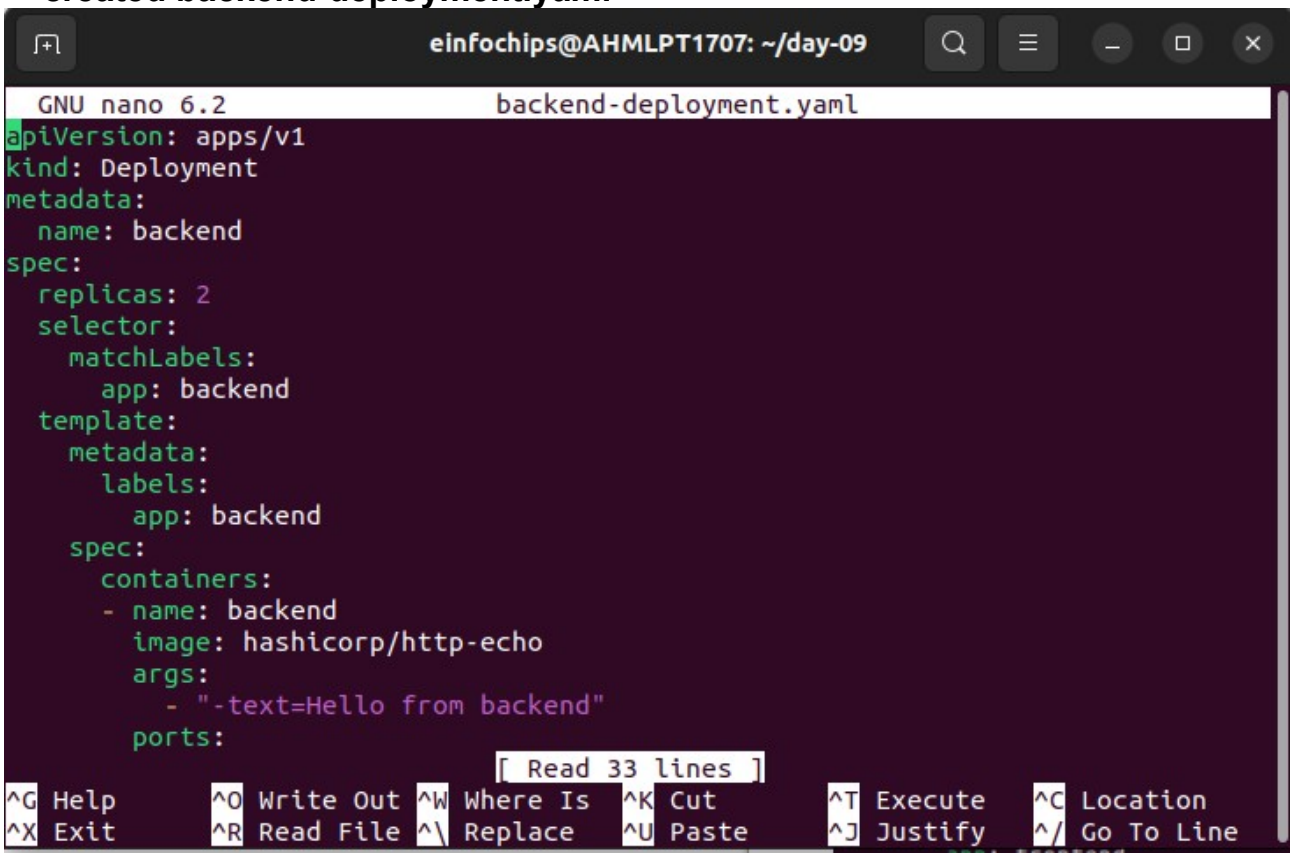


Create Ingress Resource:

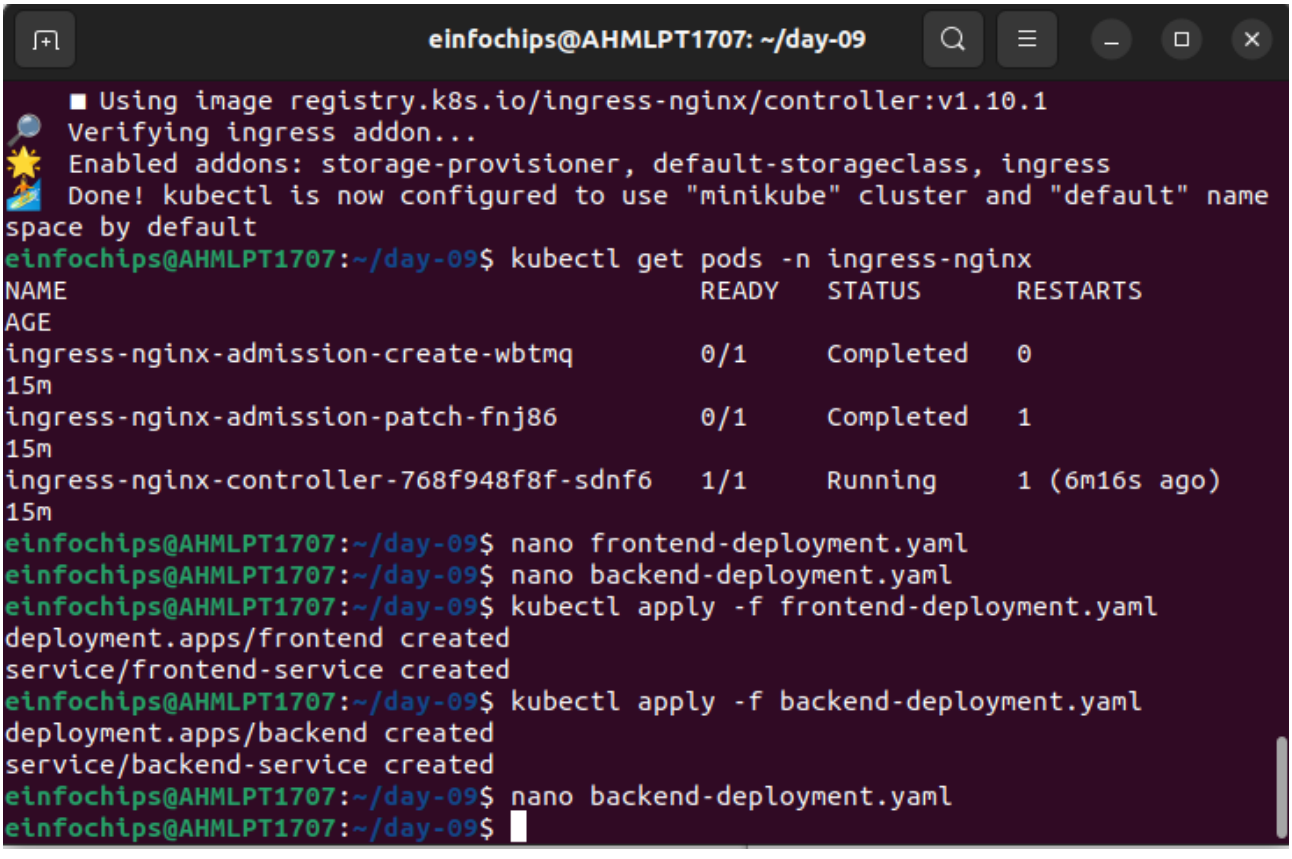
Created frontend-deployment.yaml



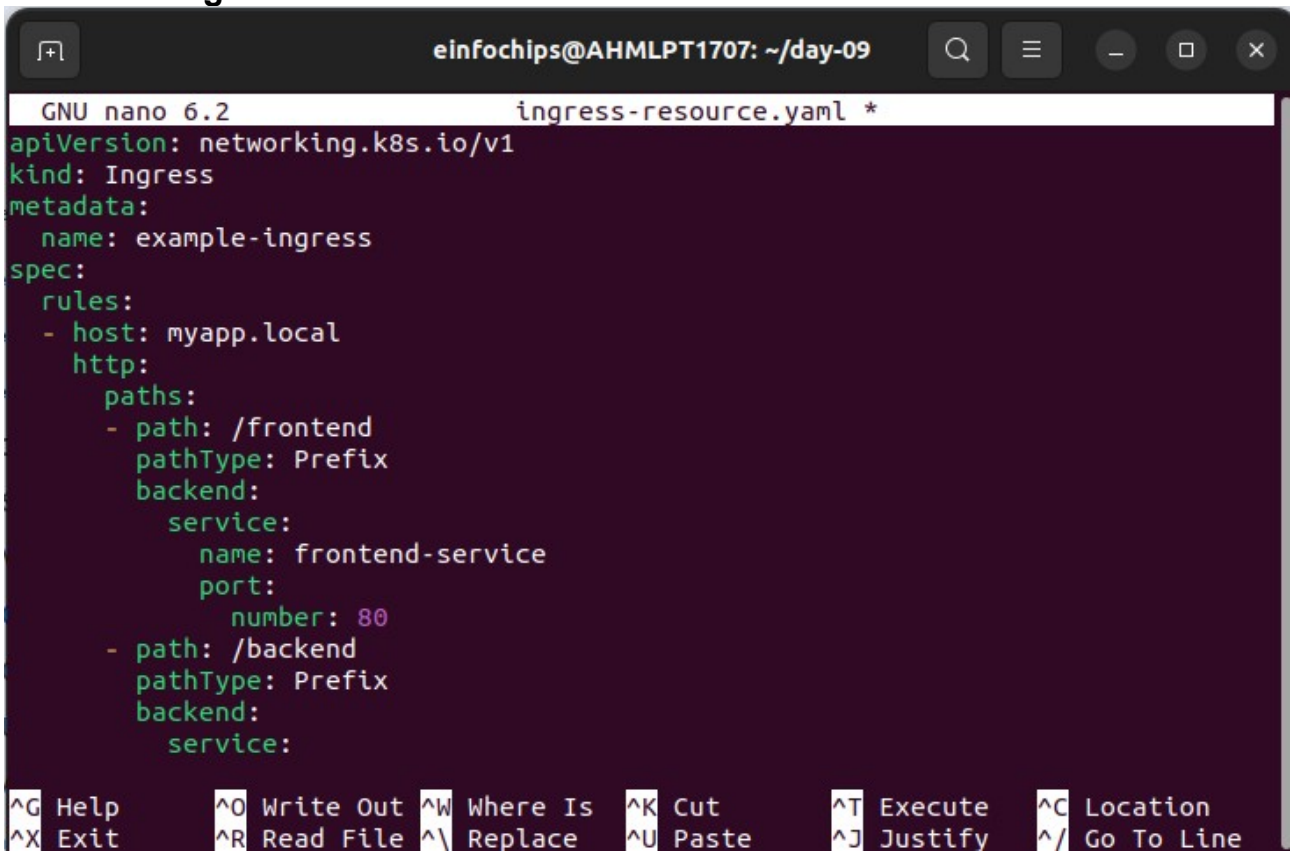
created backend-deployment.yaml



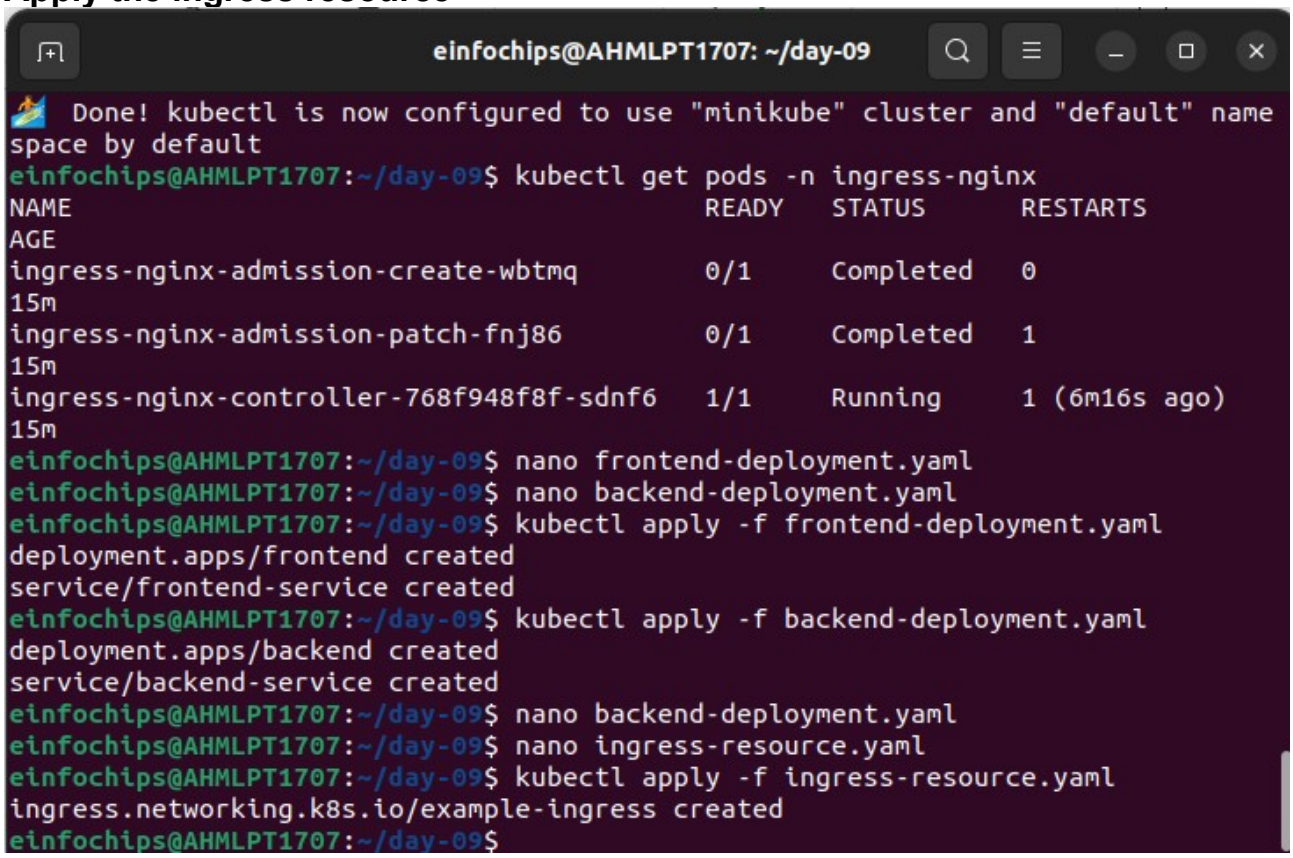
Applying the changes



Create an ingress resource

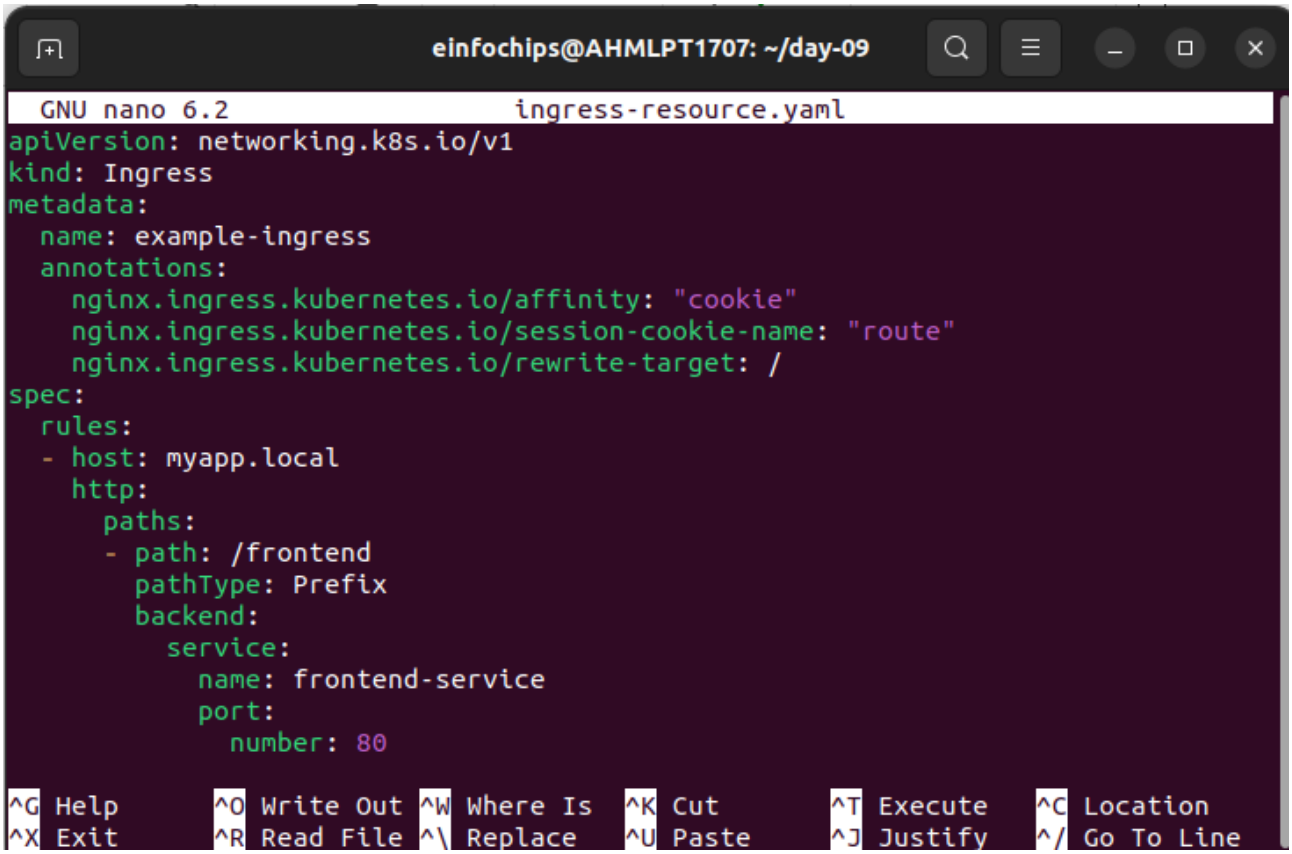


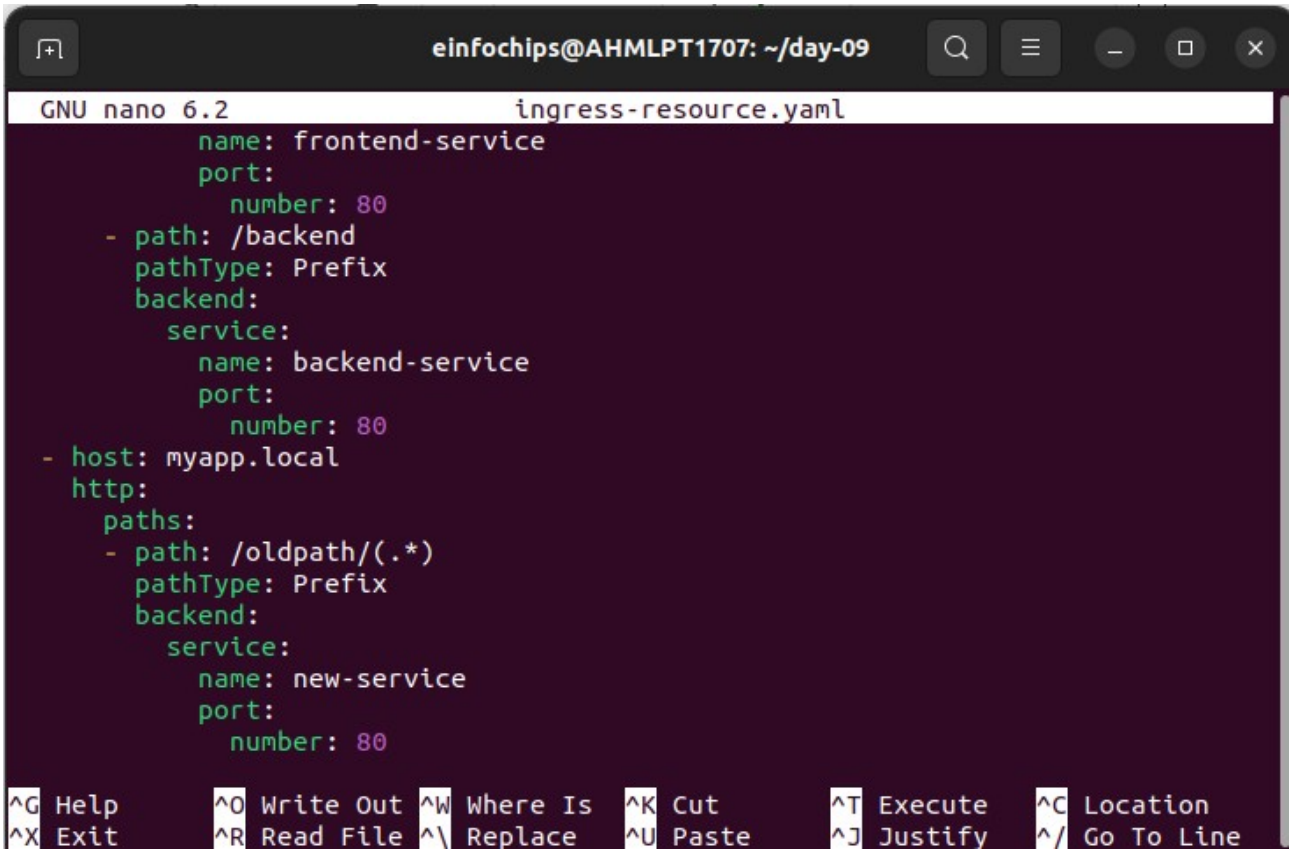
Apply the ingress resource



Configure URL rewriting in the ingress resource to modify incoming URLs before they reach the backend services.

Enable sticky sessions to ensure that requests from the same client are directed to the same backend pod.

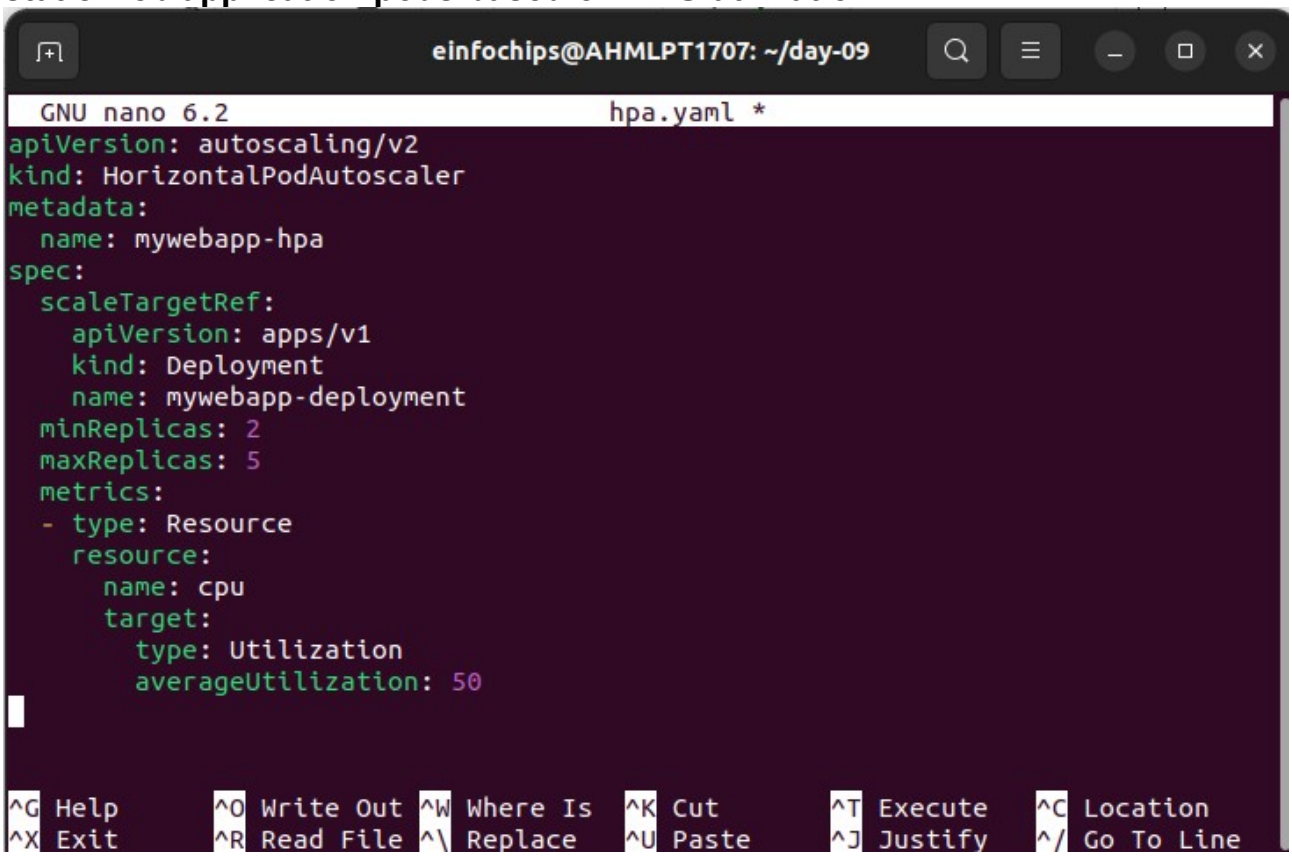




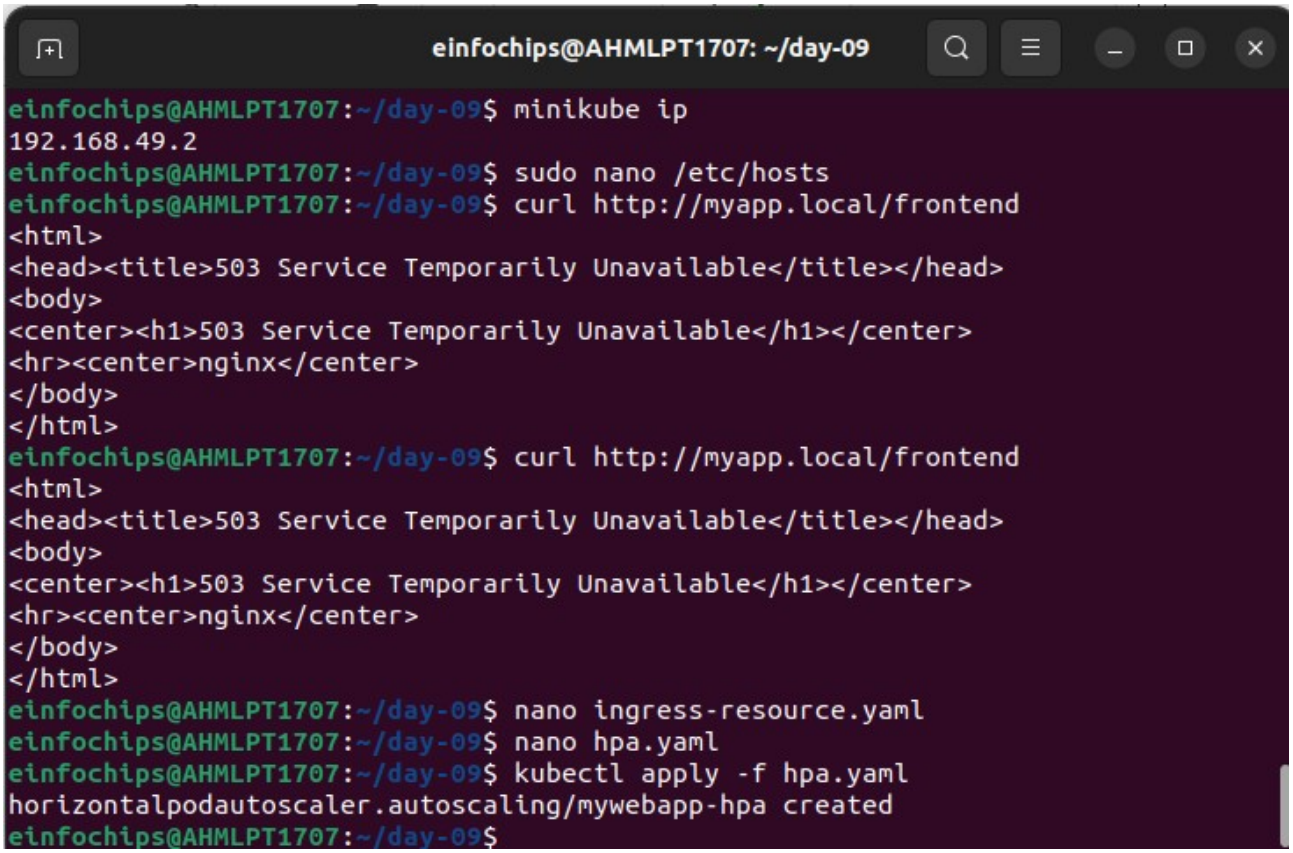
3: Implementing Horizontal Pod Autoscaling

Configure Horizontal Pod Autoscaler:

Write a horizontal pod autoscaler (HPA) manifest to automatically scale the static web application pods based on CPU utilization.

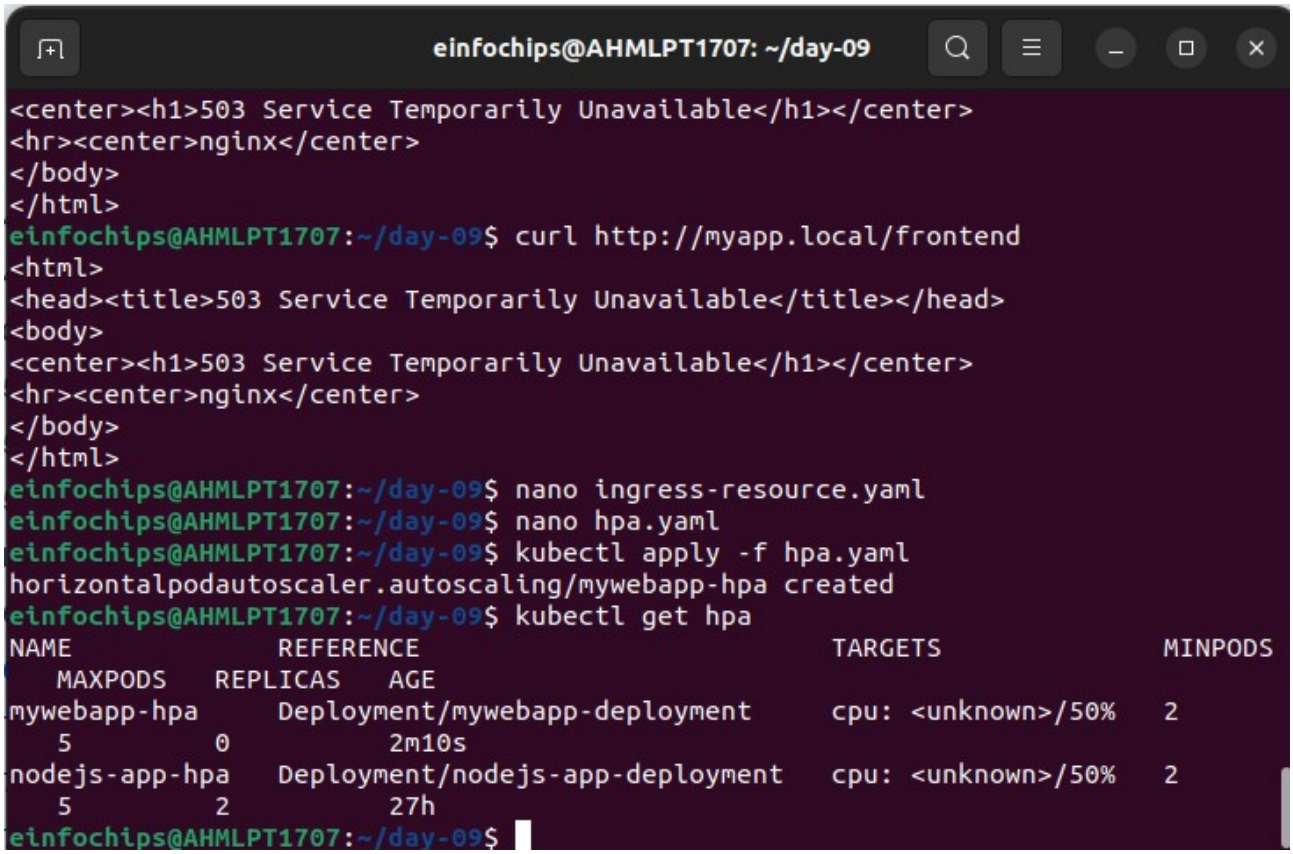


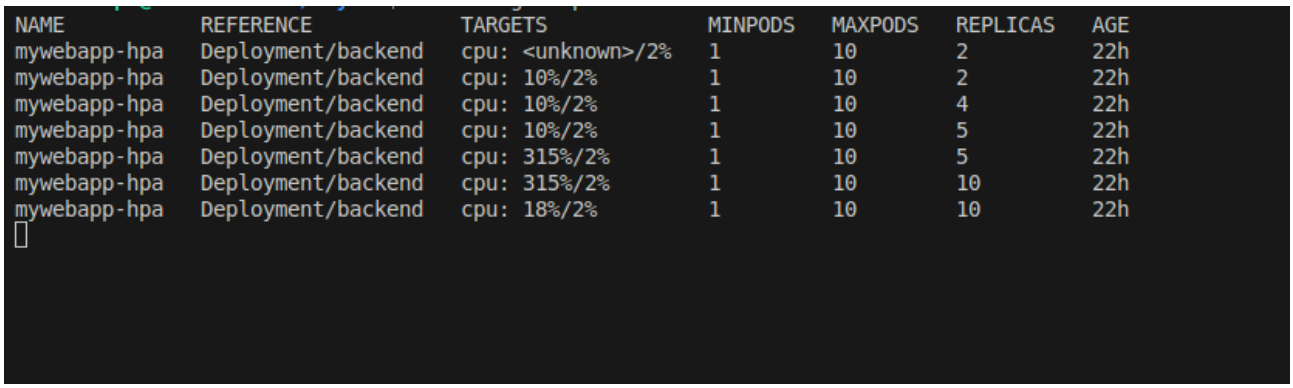
Applying hpa.yaml

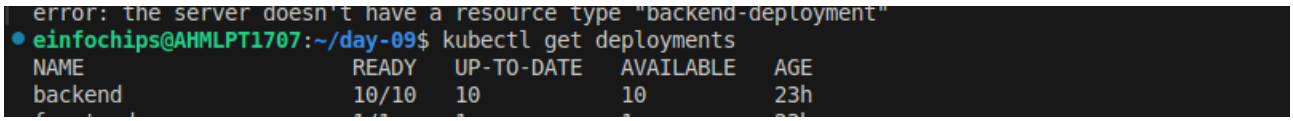


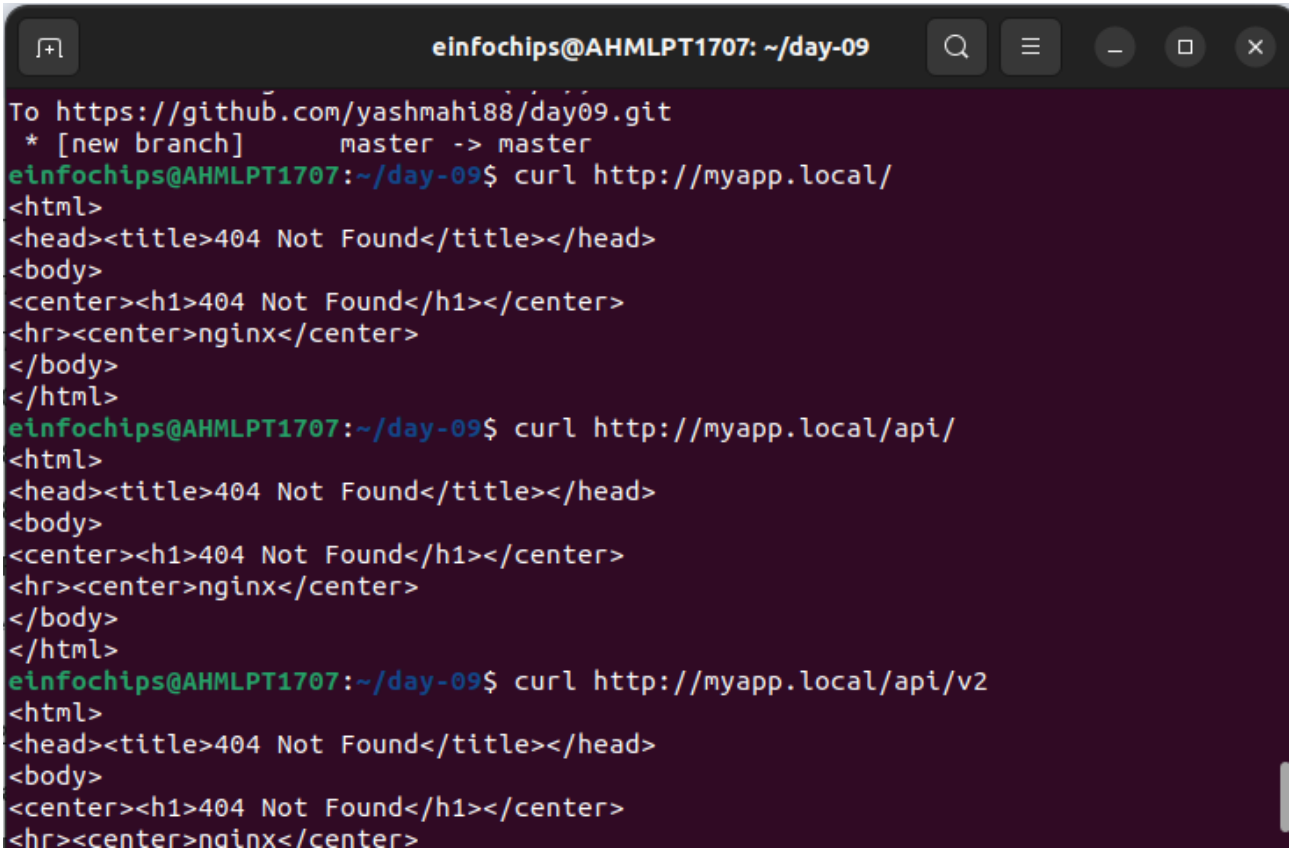
Stres testing

Perform stress testing to simulate traffic and validate the HPA configuration.
Monitor the scaling behavior and ensure the application scales up and down based on the load.

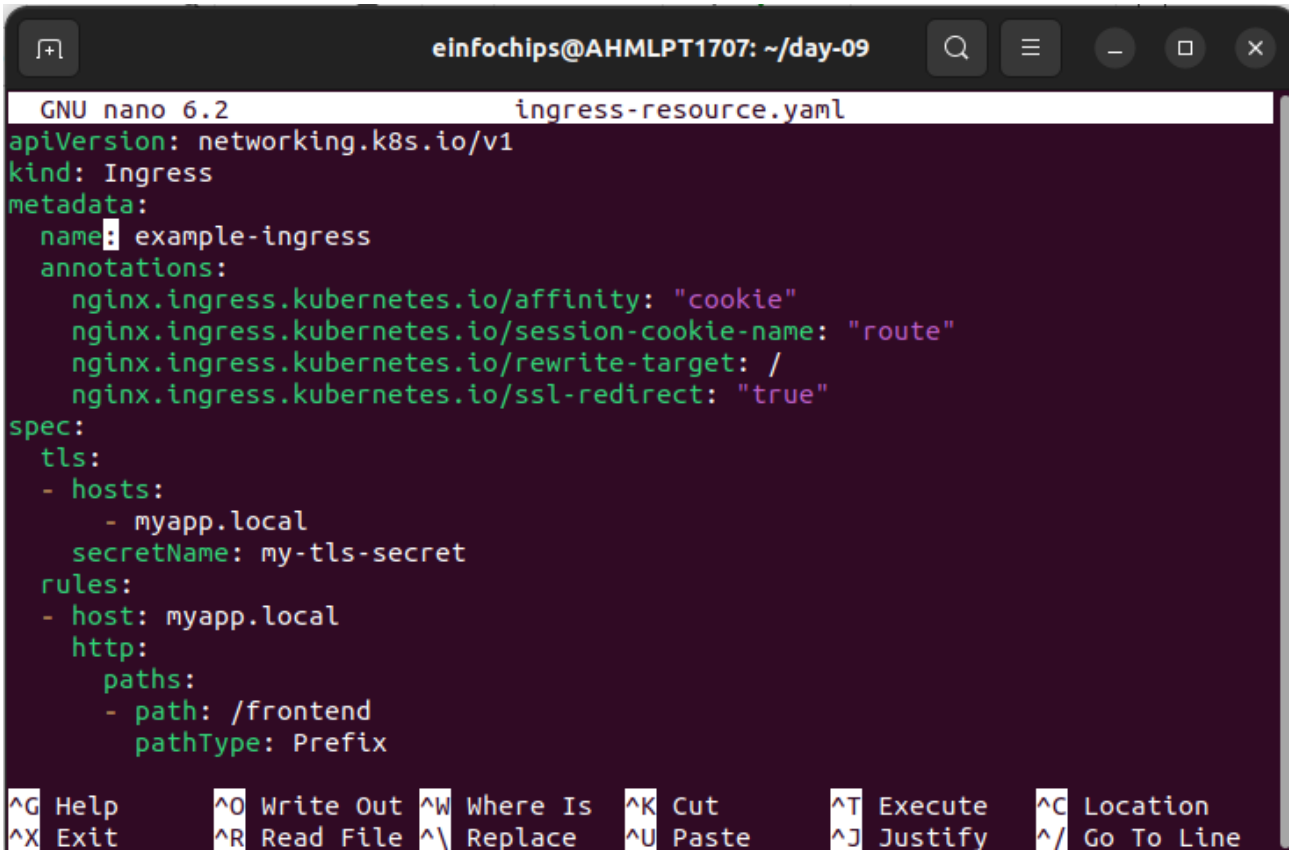








Adding TLS Secret to the ingress-resource.yaml

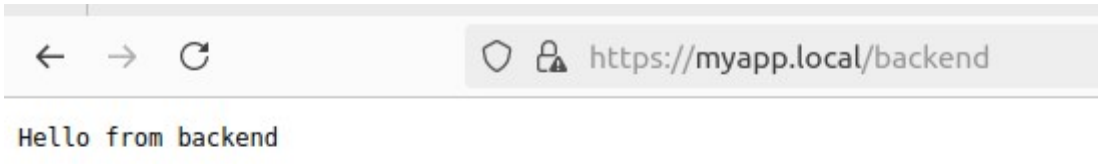


Creating SSL certificate
openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout tls.key -out tls.crt -subj "/CN=myapp.ingress/O=ingressapp-tls"

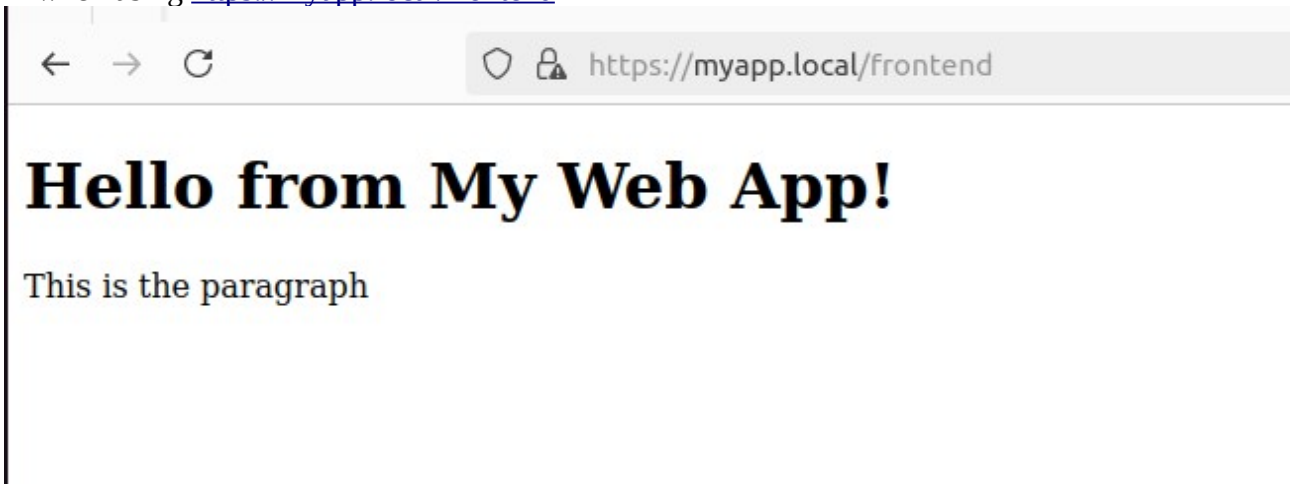
kubectl create secret tls my-tls-secret --cert=tls.crt –key=tls.key

```
einfochips@AHMLPT1707:~/day-09$ kubectl create secret tls my-tls-secret --cert=tls.crt --key=tls.key
secret/my-tls-secret created
einfochips@AHMLPT1707:~/day-09$ nano ingress-resource.yaml
einfochips@AHMLPT1707:~/day-09$ curl https://myapp.local/frontend
curl: (60) SSL certificate problem: self-signed certificate
More details here: https://curl.se/docs/sslcerts.html

curl failed to verify the legitimacy of the server and therefore could not
establish a secure connection to it. To learn more about this situation and
how to fix it, please visit the web page mentioned above.
einfochips@AHMLPT1707:~/day-09$ curl https://myapp.local/backend
curl: (60) SSL certificate problem: self-signed certificate
More details here: https://curl.se/docs/sslcerts.html

curl failed to verify the legitimacy of the server and therefore could not
establish a secure connection to it. To learn more about this situation and
how to fix it, please visit the web page mentioned above.
einfochips@AHMLPT1707:~/day-09$
```

When using https://myapp.local/backend



when using https://myapp.local/frontend



```
NAME           REFERENCE             TARGETS              MINPODS   MAXPODS   REPLICAS   AGE
mywebapp-hpa   Deployment/backend    cpu: <unknown>/2%    1         10        2          22h
mywebapp-hpa   Deployment/backend    cpu: 10%/2%          1         10        2          22h
mywebapp-hpa   Deployment/backend    cpu: 10%/2%          1         10        4          22h
mywebapp-hpa   Deployment/backend    cpu: 10%/2%          1         10        5          22h
mywebapp-hpa   Deployment/backend    cpu: 315%/2%         1         10        5          22h
mywebapp-hpa   Deployment/backend    cpu: 315%/2%         1         10        10         22h
mywebapp-hpa   Deployment/backend    cpu: 18%/2%          1         10        10         22h
```

```
error: the server doesn't have a resource type "backend-deployment"
einfochips@AHMLPT1707:~/day-09$ kubectl get deployments
NAME                    READY   UP-TO-DATE   AVAILABLE   AGE
backend                 10/10   10           10          23h
```