

```

einfochips@AHMLPT3361:~/bay7/nodejs-k8s-project$ 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! kubectrl is now configured to use "minikube" cluster and "default" namespace by default
einfochips@AHMLPT3361:~/bay7/nodejs-k8s-project$ kubectrl apply -f https://github.com/kubernetes-sigs/pttrics-server/releases/latest/download/components.yaml

```

```

temp 3/3 3 3 1/n
einfochips@AHMLPT3361:~/static-web-app$ kubectrl get nodes
NAME STATUS ROLES AGE VERSION
minikube Ready control-plane 3d21h v1.30.0
einfochips@AHMLPT3361:~/static-web-app$

```

```

einfochips@AHMLPT3361:~/static-web-app$ docker build -t yashparmar04/temp:400 .
[+] Building 1.7s (8/8) FINISHED
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 220B
=> [internal] load .dockerignore
=> => transferring context: 2B
=> [internal] load metadata for docker.io/library/nginx:alpine
=> [auth] library/nginx:pull token for registry-1.docker.io
=> [internal] load build context
=> => transferring context: 32B
=> [1/2] FROM docker.io/library/nginx:alpine@sha256:a45ee5d042aaa9e81e013f97ae40c3dda26fbe98f22b
=> CACHED [2/2] COPY index.html /usr/share/nginx/html
=> exporting to image
=> => exporting layers
=> => writing image sha256:2b269ce8458cd8423f88e4e4d3428e242bacd3cb34457c339209e700a24934e0
=> => naming to docker.io/yashparmar04/temp:400
einfochips@AHMLPT3361:~/static-web-app$ docker push yashparmar04/temp:400

```

```

mysql 5.7 510/3350080 7 MONTHS ago 501MB
einfochips@AHMLPT3361:~/static-web-app$ docker build -t temp .
[+] Building 2.0s (7/7) FINISHED
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 220B
=> [internal] load .dockerignore
=> => transferring context: 2B
=> [internal] load metadata for docker.io/library/nginx:alpine
=> [internal] load build context
=> => transferring context: 32B
=> [1/2] FROM docker.io/library/nginx:alpine@sha256:a45ee5d042aaa9e81e013f97ae40c3dda26fbe
=> CACHED [2/2] COPY index.html /usr/share/nginx/html
=> exporting to image
=> => exporting layers
=> => writing image sha256:2b269ce8458cd8423f88e4e4d3428e242bacd3cb34457c339209e700a24934e
=> => naming to docker.io/library/temp
einfochips@AHMLPT3361:~/static-web-app$ docker images

```

```

gcr.io/k8s-minikube/kicbase      v0.0.44      5a6e59a9bdc0      2 months ago      1.26GB
mysql                            5.7          5107333e08a8      7 months ago      501MB
einfochips@AHMLPT3361:~/static-web-app$ docker push yashparmar04/temp:400
The push refers to repository [docker.io/yashparmar04/temp]
c6238bd5240a: Pushed
a51b172d7184: Mounted from library/nginx
b7486fe26981: Mounted from library/nginx
320c8baef084: Mounted from library/nginx
d2cef4a1b224: Mounted from library/nginx
4275164ce225: Mounted from library/nginx
6e92270dbfe6: Mounted from library/nginx
b5d2e1fcf1ad: Mounted from library/nginx
af9a70194aa4: Mounted from library/nginx

```

```

einfochips@AHMLPT3361:~/static-web-app$ ls
Dockerfile index.html
einfochips@AHMLPT3361:~/static-web-app$ nano deployment.yaml
einfochips@AHMLPT3361:~/static-web-app$ nano service.yaml
einfochips@AHMLPT3361:~/static-web-app$ ls
deployment.yaml Dockerfile index.html service.yaml
einfochips@AHMLPT3361:~/static-web-app$ mv 'service.yaml' 'service.yaml'
einfochips@AHMLPT3361:~/static-web-app$ ls
deployment.yaml Dockerfile index.html service.yaml
einfochips@AHMLPT3361:~/static-web-app$ kubectl apply -f deployment.yaml
deployment.apps/temp created
einfochips@AHMLPT3361:~/static-web-app$ kubectl apply -f service.yaml
service/temp-service created

```

```

temp-service      ClusterIP      10.98.77.26      <none>      80/TCP      4m34s
einfochips@AHMLPT3361:~/static-web-app$ kubectl get pods
NAME                                READY    STATUS    RESTARTS    AGE
backend-64d8b65b8f-bb6kw            1/1      Running   1 (53m ago)  177m
backend-64d8b65b8f-hsk5f            1/1      Running   1 (53m ago)  177m
frontend-f4584dffc-4v5vj            1/1      Running   1 (53m ago)  177m
frontend-f4584dffc-j7k8m            1/1      Running   1 (53m ago)  177m
nodejs-app-67b6cb9fcf-bb24w          1/1      Running   9 (53m ago)  32h
nodejs-app-67b6cb9fcf-v4t8w          1/1      Running   9 (53m ago)  32h
nodejs-app-deployment-fd98f79bf-9cxxv 0/1      ImagePullBackOff 0             26h
nodejs-app-deployment-fd98f79bf-z2p2h 0/1      ImagePullBackOff 0             26h
temp-7b84d987f5-c8hkv                1/1      Running   0             5m9s
temp-7b84d987f5-mf9pr                1/1      Running   0             90s
temp-7b84d987f5-r5r74                1/1      Running   0             5m9s
temp-7c64657f9c-gbh4m                0/1      ImagePullBackOff 0             90s

```

```

einfochips@AHMLPT3361:~/static-web-app$ minikube start --addons=ingress
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 registry.k8s.io/ingress-nginx/kube-webhook-certgen:v1.4.1
  ■ Using image registry.k8s.io/ingress-nginx/controller:v1.10.1
  ■ Using image gcr.io/k8s-minikube/storage-provisioner:v5
  ■ Using image registry.k8s.io/ingress-nginx/kube-webhook-certgen:v1.4.1
Verifying ingress addon...
Enabled addons: storage-provisioner, default-storageclass, ingress
Done! kubectrl is now configured to use "minikube" cluster and "default" namespace by default
einfochips@AHMLPT3361:~/static-web-app$ nano ingress-resource.yaml
einfochips@AHMLPT3361:~/static-web-app$ nano frontend-deployment.yaml
einfochips@AHMLPT3361:~/static-web-app$ nano backend-deployment.yaml

```

```

service "frontend-service" deleted
einfochips@AHMLPT3361:~/static-web-app$ kubectl apply -f frontend-deployment.yaml
deployment.apps/frontend unchanged
service/frontend-service created
einfochips@AHMLPT3361:~/static-web-app$ kubectl apply -f backend-deployment.yaml
deployment.apps/backend unchanged
service/backend-service created

```

```

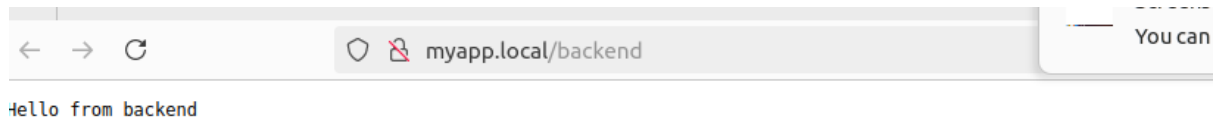
minikube   Ready   control-plane   3d21h   v1.30.0
einfochips@AHMLPT3361:~/static-web-app$ kubectl get deployments
NAME                READY   UP-TO-DATE   AVAILABLE   AGE
backend              2/2     2             2           20h
frontend             2/2     2             2           20h
nodejs-app           2/2     2             2           2d2h
nodejs-app-deployment 0/2     2             0           44h
temp                 3/3     3             3           17h

```

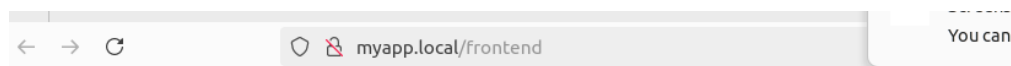
```

</html>
einfochips@AHMLPT3361:~/static-web-app$ curl http://myapp.local/backend
Hello from backend
einfochips@AHMLPT3361:~/static-web-app$ nano ingress-resource.yaml

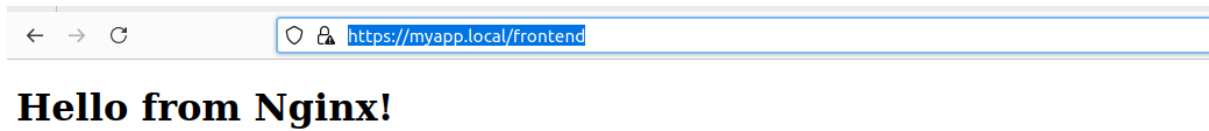
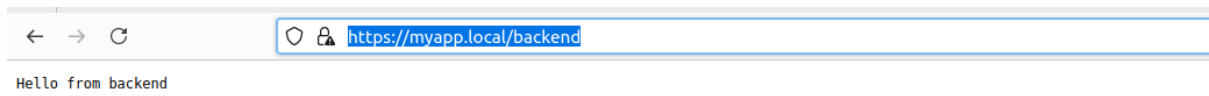
```



```
targetPort: 80
einfochips@AHMLPT3361:~/static-web-app$ curl http://myapp.local/frontend
<!DOCTYPE html>
<html>
<head>
  <title>Welcome to My Static Site</title>
</head>
<body>
  <h1>Hello from Nginx!</h1>
</body>
</html>
```



**Hello from Nginx!**



```
einfochips@AHMLPT2581:~/day9_task$ kubectl get hpa
NAME                REFERENCE                TARGETS          MINPODS   MAXPODS   REPLICAS   AGE
day9-image-hpa      Deployment/day9-image-deployment  cpu: <unknown>/2%  1         10        1          100m
einfochips@AHMLPT2581:~/day9_task$ kubectl get hpa
NAME                REFERENCE                TARGETS          MINPODS   MAXPODS   REPLICAS   AGE
day9-image-hpa      Deployment/day9-image-deployment  cpu: <unknown>/2%  1         10        1          101m
```