

Source Links:

- **Github** - <https://github.com/dirghayu101/clo835-assignment2>
- **Docker** - <https://hub.docker.com/r/dirghayu101/clo835-assignment-2>

Screenshots:

```
dirghayujoshi@Dirghayus-MacBook-Air assignment-2 % kubectl apply -f service.yaml
service/clo835-assignment-2-service created
dirghayujoshi@Dirghayus-MacBook-Air assignment-2 % kubectl get service
NAME                                TYPE        CLUSTER-IP      EXTERNAL-IP  PORT(S)          AGE
clo835-assignment-2-service        NodePort    10.102.252.49   <none>       3000:30002/TCP   11s
kubernetes                         ClusterIP   10.96.0.1       <none>       443/TCP          21h
dirghayujoshi@Dirghayus-MacBook-Air assignment-2 % kubectl apply -f service.yaml
service/clo835-assignment-2-service configured
dirghayujoshi@Dirghayus-MacBook-Air assignment-2 % kubectl get service -o wide
NAME                                TYPE        CLUSTER-IP      EXTERNAL-IP  PORT(S)          AGE      SELECTOR
clo835-assignment-2-service        NodePort    10.102.252.49   <none>       3030:30002/TCP   3m38s   app=clo835-assignment-2
kubernetes                         ClusterIP   10.96.0.1       <none>       443/TCP          21h      <none>
```

```
dirghayujoshi@Dirghayus-MacBook-Air assignment-2 % kubectl get pods -o wide
NAME                                READY  STATUS   RESTARTS  AGE  IP            NODE       NOMINATED NODE  READINESS GATES
clo835-assignment-2-deployment-657548df86-7psxt  1/1    Running  0          4m56s  10.244.0.25   minikube   <none>          <none>
clo835-assignment-2-deployment-657548df86-tvlfh  1/1    Running  0          4m56s  10.244.0.24   minikube   <none>          <none>
clo835-assignment-2-deployment-657548df86-zskds  1/1    Running  0          4m56s  10.244.0.26   minikube   <none>          <none>
```

```
dirghayujoshi@Dirghayus-MacBook-Air assignment-2 % minikube ssh
docker@minikube:~$ echo $shell

docker@minikube:~$ echo $SHELL
/bin/bash
```

```
This is not the full help, this menu is stripped into categories.
Use "--help category" to get an overview of all categories.
For all options use the manual or "--help all".
docker@minikube:~$ curl http://10.102.252.49:3030 #This is the service created inside the cluster being accessed using the private port assigned.
<html><head><title>Time Service</title></head><body><h1>Current time in Toronto: 2024-07-02 12:11:38</h1></body></html>docker@minikube:~$
docker@minikube:~$
```

```
docker@minikube:~$ curl http://10.244.0.25:3030 # This the first of the three pods created by k8s based on specification in deployment.yaml
<html><head><title>Time Service</title></head><body><h1>Current time in Toronto: 2024-07-02 12:14:23</h1></body></html>docker@minikube:~$
docker@minikube:~$ curl http://10.244.0.24:3030 # This the second of the three pods created by k8s based on specification in deployment.yaml
<html><head><title>Time Service</title></head><body><h1>Current time in Toronto: 2024-07-02 12:14:47</h1></body></html>docker@minikube:~$
docker@minikube:~$ curl http://10.244.0.26:3030 # This is the last of the three pods created by k8s based on specification in deployment.yaml
<html><head><title>Time Service</title></head><body><h1>Current time in Toronto: 2024-07-02 12:15:28</h1></body></html>docker@minikube:~$
docker@minikube:~$
```

```
dirghayujoshi@Dirghayus-MacBook-Air assignment-2 % minikube service clo835-assignment-2-service --url http://127.0.0.1:58052 # This will allow inbound traffic to the service we created
! Because you are using a Docker driver on darwin, the terminal needs to be open to run it.
```

```
dirghayujoshi@Dirghayus-MacBook-Air assignments % curl http://127.0.0.1:58052
<html><head><title>Time Service</title></head><body><h1>Current time in Toronto: 2024-07-02 12:19:38</h1></body></html>
dirghayujoshi@Dirghayus-MacBook-Air assignments % curl http://127.0.0.1:58052 # Running in my local system.
<html><head><title>Time Service</title></head><body><h1>Current time in Toronto: 2024-07-02 12:20:01</h1></body></html>curl: (3) URL using bad/illegal format or missing URL
```



Current time in Toronto: 2024-07-02 12:20:46

Documentation:

Step 1: Getting files required:

- `wget https://raw.githubusercontent.com/sojoudian/clo835_s24/master/project-2/app.py`
- `git init && git add . && git commit -m "setup." && git remote add origin git@github.com:dirghayu101/clo835-assignment2.git && git push -u origin main`

Step 2: Setting up docker

- `docker system prune -a` —> Delete all the unused resources.
- `docker build -t clo835-assignment-2 .` —> Build image.
- Write the docker file.
- `docker run --name clo835-assignment-2 -d -p 127.0.0.1:3030:3030 clo835-assignment-2` <- Just for testing.
- `docker tag clo835-assignment-2 dirghayu101/clo835-assignment-2`
- `docker push dirghayu101/clo835-assignment-2:latest`

Step 3: Kubernetes Setup

- `minikube start`
- `kubectl version --client` —> Check if k8s has been installed.
- `kubectl cluster-info`

Step 4: Kubernetes Manifest file:

- Write k8s file for deployment and services.

Step 5: Kubernetes Deployment:

- `kubectl apply -f deployment.yaml`
- `kubectl apply -f service.yaml`

Step 6: Kubernetes Deployment Verification and getting metadata:

- `kubectl get pods -o wide`
- `kubectl get services -o wide`
- `kubectl get nodes`

Step 7: Kubernetes testing:

- `minikube ssh` -> For testing from inside the cluster as the services won't be exposed.
- `curl <serviceIP>:<servicePort>` —> You can test the pod and the service.

Step 8: Exposing service locally

- `minikube service clo835-assignment-2-service --url` <- expose the services created in minikube cluster.

Other helpful

- `docker kill $(docker ps -q) && docker rm -f $(docker ps -a -q)` —> can remove f flag, or can use just the second half.
- `kubectl run -i --tty --rm debug --image=busybox --restart=Never -- sh\n.` —> This creates a pod which you can also use for testing.
- `kubectl get service`

- kubectl get pods
- kubectl delete deployment <name>
- kubectl delete service <name>

Challenges:

Exposing service on my local system was a major challenge. Apparently to expose a service running in your cluster, you have to create a tunnel.

[1] helped me in setting it up.

References:

[1]: <https://minikube.sigs.k8s.io/docs/handbook/accessing/> <- Accessing application running within minikube.

[2]: <https://docs.oracle.com/en/operating-systems/olcne/1.1/orchestration/kubectl-delete.html>