Task A2

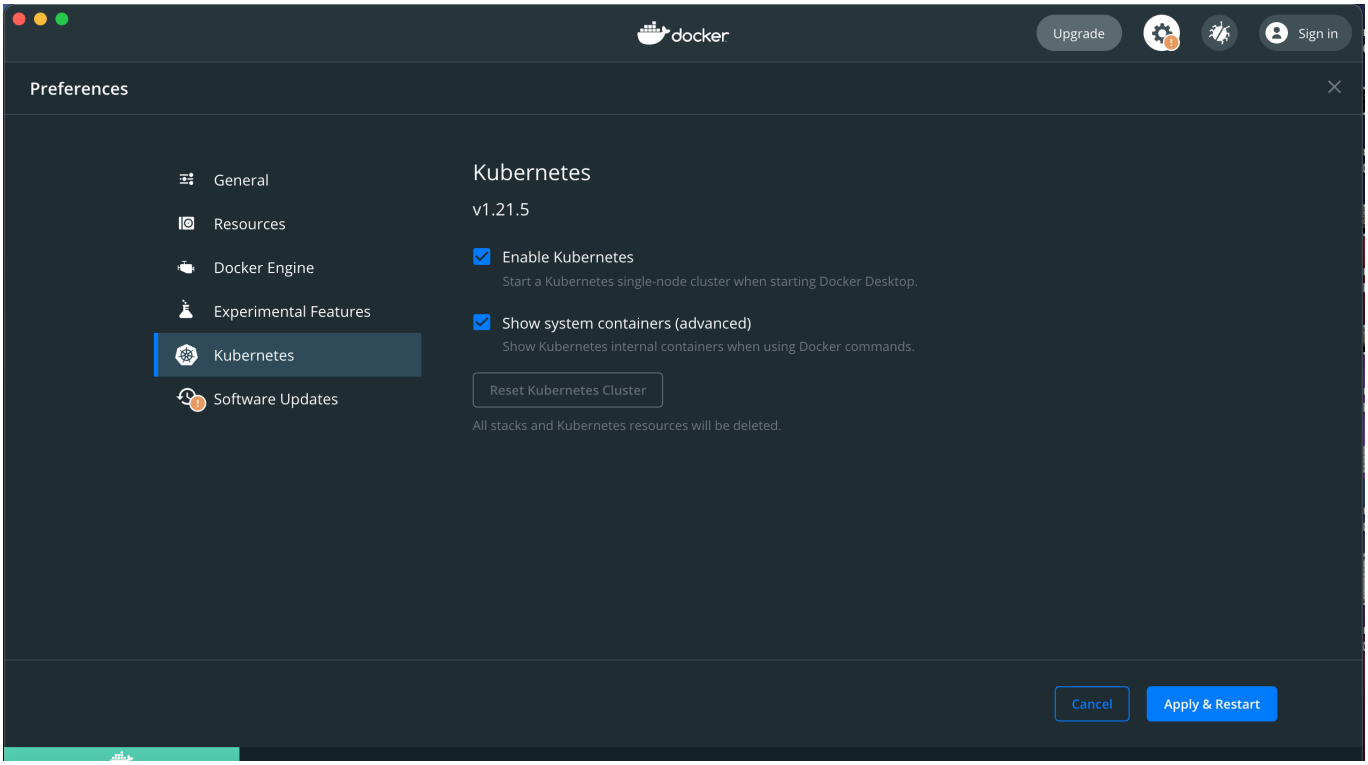
Student Name: Chow Jia Ying

Matriculation Number: A0187457M

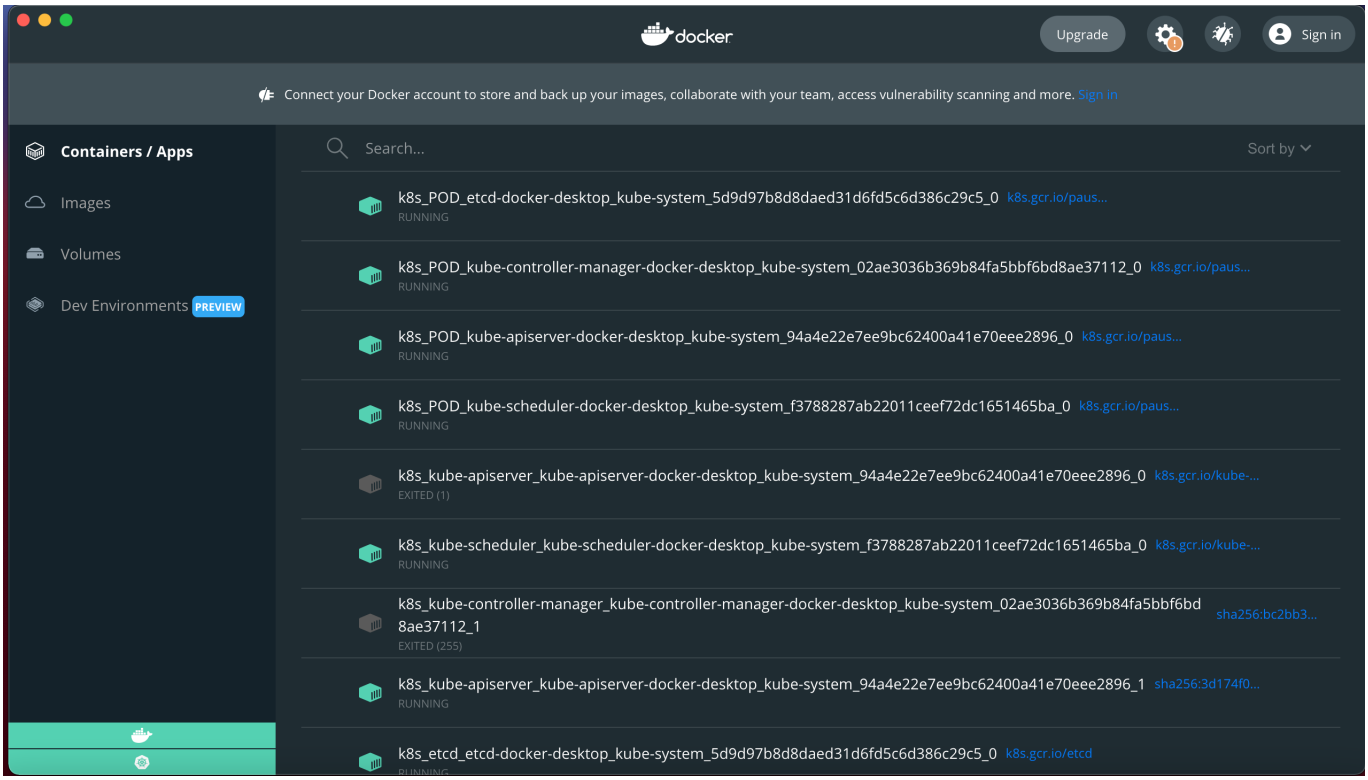
Github link: https://github.com/C-likethis123/CS3219/tree/master/Task_A2

Setting up Kubernetes through Docker Desktop

To set up Kubernetes, I went to Docker Desktop -> Preference -> Kubernetes and enabled Kubernetes.



This installs the relevant images and runs a Kubernetes cluster.



I have also installed `kubectl` in my environment.

Creating a sample image

To demonstrate, I have a simple Express server written in `index.js`. The `Dockerfile` packages `index.js` in a Docker image.

To build the Docker image: `docker build . -t clikethis123/task-a2:latest`

To run the Docker image locally: `docker run --name task-a2 --rm -d -p 3000:3000 clikethis123/task-a2`

To stop the Docker image locally: `docker stop task-a2`

Pushing the Docker image to Docker Hub: `docker push clikethis123/task-a2:latest`

Deploying the app in Kubernetes

A Deployment has been set up with the relevant configurations in `express-deployment.yaml` (link here: https://github.com/C-likethis123/CS3219/blob/master/Task_A2/express-deployment.yaml).

The deployment deploys 2 pods, each pod with the image `clikethis123/task-a2`. And it is labelled with `app: task-a2`.

To deploy the app, apply the configuration: `kubectl apply -f express-deployment.yaml`

The deployment will create two pods, one for each replica, as seen in the command `kubectl get pods`.

```
CS3219/Task_A2 on ?master [!?] is 📦 v1.0.0 via  v15.5.0 on ☁ (ap-southe
east1)
❯ kubectl apply -f express-deployment.yaml
deployment.apps/task-a2 created

CS3219/Task_A2 on ?master [!?] is 📦 v1.0.0 via  v15.5.0 on ☁ (ap-southe
east1)
❯ kubectl get deployments
NAME               READY   UP-TO-DATE   AVAILABLE   AGE
nginx-deployment   3/3     3            3           7m39s
task-a2            1/2     2            1           10s

CS3219/Task_A2 on ?master [!?] is 📦 v1.0.0 via  v15.5.0 on ☁ (ap-southe
east1) took 2s
```

To create a service for the deployment: `kubectl apply -f express-service.yaml` (link to file here: https://github.com/C-likethis123/CS3219/blob/master/Task_A2/express-service.yaml). This creates a service named `task-a2` for the pods that are currently deployed.

`kubectl get svc task-a2` shows the following:

```
CS3219/Task_A2 on ?master [!?] is 📦 v1.0.0 via  v15.5.0 on ☁ (ap-southeast-1) on
east1)
❯ kubectl get svc task-a2
NAME      TYPE        CLUSTER-IP      EXTERNAL-IP   PORT(S)    AGE
task-a2   ClusterIP   10.96.156.64    <none>        3000/TCP   7m14s
```

By accessing one of the pods and typing `curl <CLUSTER-IP>:<PORT>`, we can access the application.

```
task-a2-5f849fb96b-np7z6   1/1     Running   0          25m

CS3219/Task_A2 on ?master [!?] is 📦 v1.0.0 via  v15.5.0 on ☁ (ap-southeast-1) o
n ☁ (asia-southeast1)
❯ kubectl exec -it task-a2-5f849fb96b-hshlg --sh
Error: unknown flag: --sh
See 'kubectl exec --help' for usage.

CS3219/Task_A2 on ?master [!?] is 📦 v1.0.0 via  v15.5.0 on ☁ (ap-southeast-1) o
n ☁ (asia-southeast1)
❯ kubectl exec -it task-a2-5f849fb96b-hshlg -- sh
/app # apk add --no-cache curl
fetch http://dl-cdn.alpinelinux.org/alpine/v3.11/main/x86_64/APKINDEX.tar.gz
fetch http://dl-cdn.alpinelinux.org/alpine/v3.11/community/x86_64/APKINDEX.tar.gz
(1/4) Installing ca-certificates (20191127-r2)
(2/4) Installing nghttp2-libs (1.40.0-r1)
(3/4) Installing libcurl (7.79.1-r0)
(4/4) Installing curl (7.79.1-r0)
Executing busybox-1.31.1-r9.trigger
Executing ca-certificates-20191127-r2.trigger
OK: 9 MiB in 20 packages
/app # curl 10.103.102.106:3000
Hello World!/app #
```

To access the application in localhost, we can forward the port 3000 from the service to localhost like this:

```
kubectl port-forward svc/task-a2 3000:3000
```

And accessing it in `localhost:3000` should work.