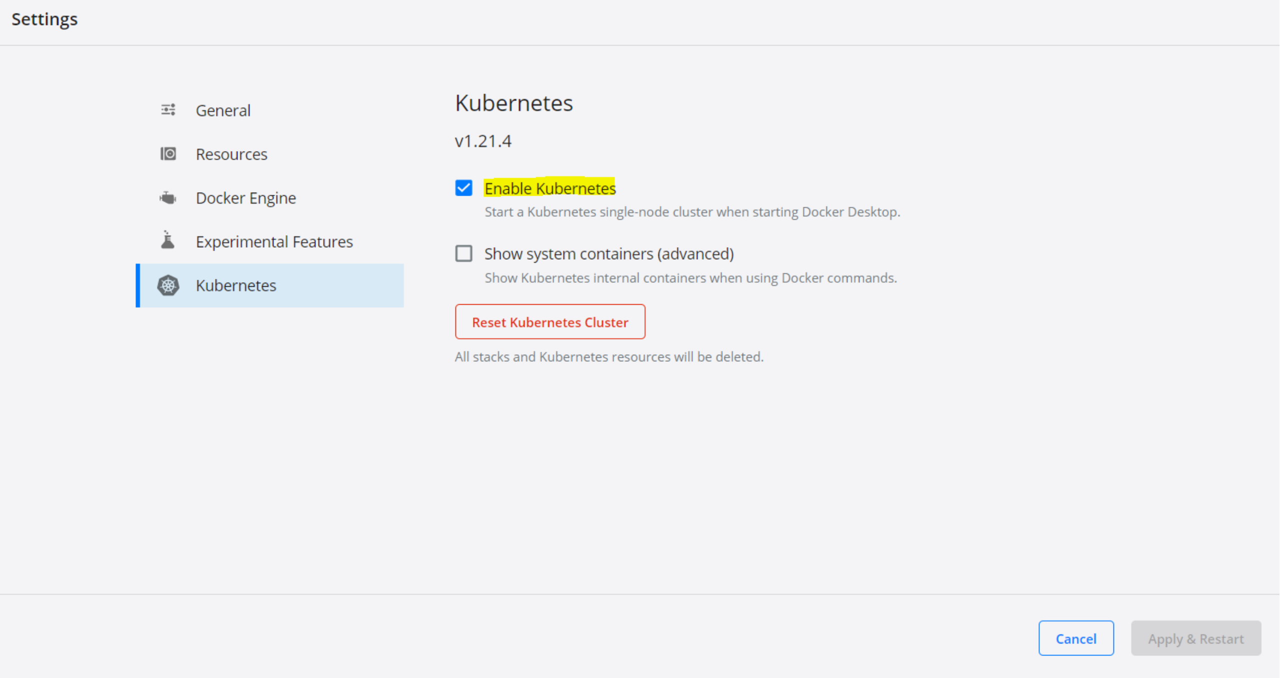
Project Details

Task: A2
Done by: Tan Wei Jie (A0202017B)
Repo Link: https://github.com/tanweijie123/CS3219_Sandbox/tree/main/Task_A/A2

Instructions on how to run

1. For my project, I will be using Docker Desktop with Kubernetes. As a start, ensure that Kubernetes is enabled on your Docker Desktop. You can do that by going to Docker Desktop > Settings, check on "Enable Kubernetes".



Kubernetes

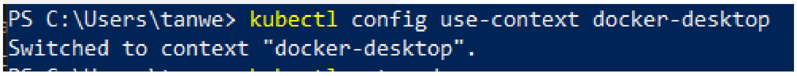
v1.21.4

☑ Enable Kubernetes
Start a Kubernetes single-node cluster when starting Docker Desktop.

□ Show system containers (advanced)
Show Kubernetes internal containers when using Docker commands.

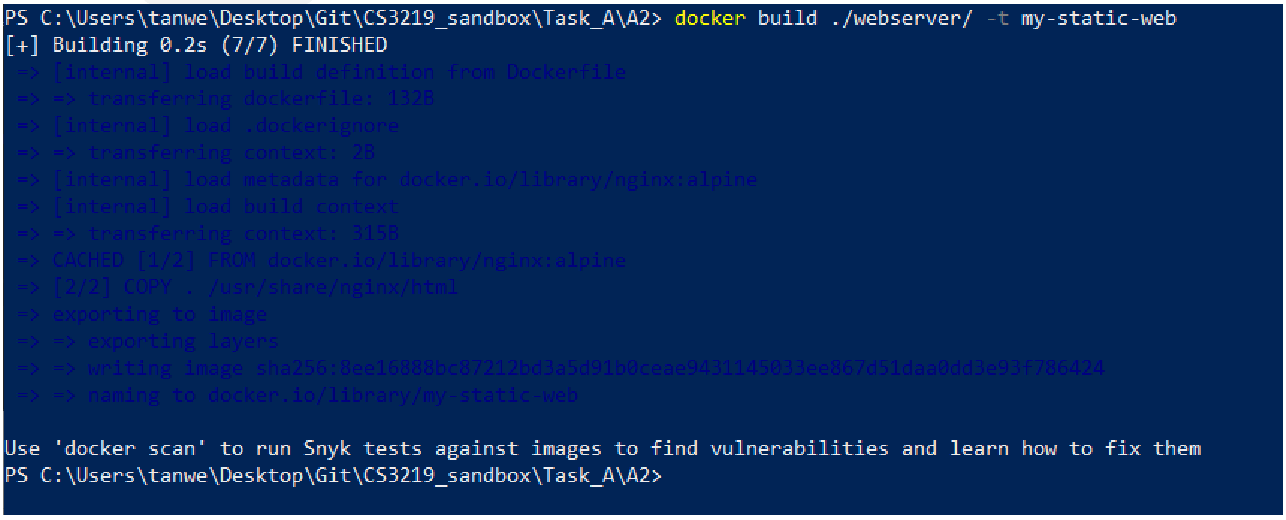
All stacks and Kubernetes resources will be deleted.

2. Next, check that your kubernetes is using the `docker-desktop` environment. You can do this by entering `kubectl config use-context docker-desktop`.



```
PS C:\Users\tanwe> kubectl config use-context docker-desktop
Switched to context "docker-desktop".
```

3. Build the docker image that you want to use for kubernetes. In this example, I will be using a simple webserver found in `./webserver/` directory. Run `docker build ./webserver/ -t my-static-web` to build the docker image, and name it `my-static-web`.

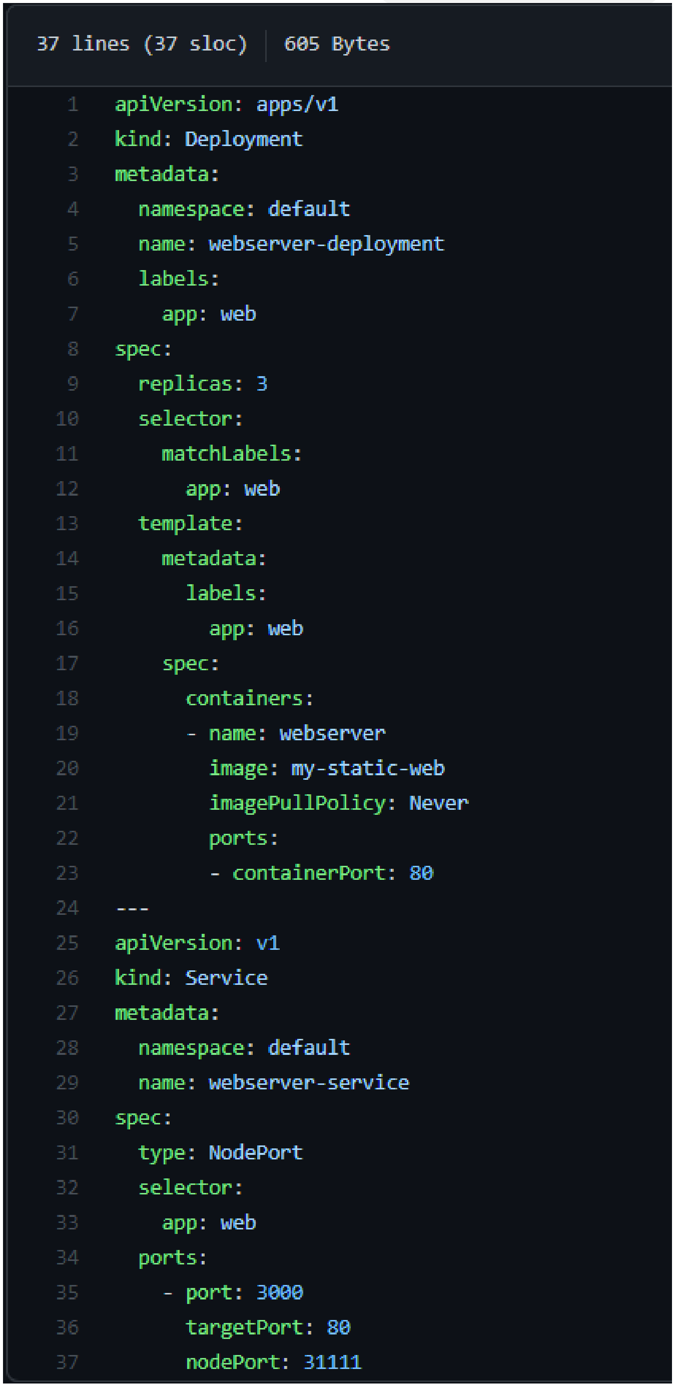


```
PS C:\Users\tanwe\Desktop\Git\CS3219_sandbox\Task_A\A2> docker build ./webserver/ -t my-static-web
[+] Building 0.2s (7/7) FINISHED
 => [internal] load build definition from Dockerfile
 => => transferring dockerfile: 132B
 => [internal] load .dockerignore
 => => transferring context: 2B
 => [internal] load metadata for docker.io/library/nginx:alpine
 => [internal] load build context
 => => transferring context: 315B
 => CACHED [1/2] FROM docker.io/library/nginx:alpine
 => [2/2] COPY . /usr/share/nginx/html
 => exporting to image
 => => exporting layers
 => => writing image sha256:8ee16888bc87212bd3a5d91b0ceae9431145033ee867d51daa0dd3e93f786424
 => => naming to docker.io/library/my-static-web

Use 'docker scan' to run Snyk tests against images to find vulnerabilities and learn how to fix them
PS C:\Users\tanwe\Desktop\Git\CS3219_sandbox\Task_A\A2>
```

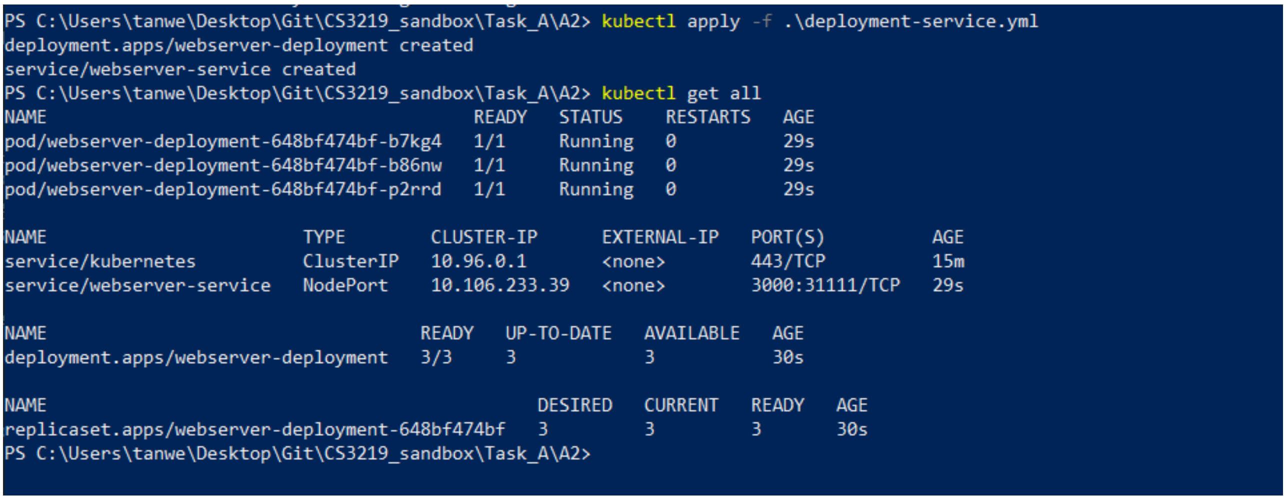
4. Once the build is complete, execute `kubectl apply -f ./deployment-service.yml` to setup kubernetes configuration. Verify that you have 3 running pods for the `webserver-service` service. You should also note that

port 31111 is exposed due to the `deployment-service.yml` configuration.



37 lines (37 sloc) | 605 Bytes

```yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  namespace: default
  name: webserver-deployment
  labels:
    app: web
spec:
  replicas: 3
  selector:
    matchLabels:
      app: web
  template:
    metadata:
      labels:
        app: web
    spec:
      containers:
      - name: webserver
        image: my-static-web
        imagePullPolicy: Never
        ports:
        - containerPort: 80
---
apiVersion: v1
kind: Service
metadata:
  namespace: default
  name: webserver-service
spec:
  type: NodePort
  selector:
    app: web
  ports:
    - port: 3000
      targetPort: 80
      nodePort: 31111
```



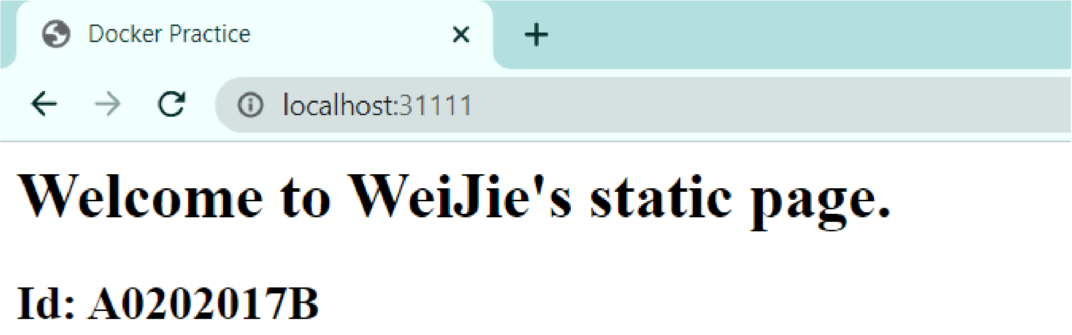
```
PS C:\Users\tanwe\Desktop\Git\CS3219_sandbox\Task_A\A2> kubectl apply -f .\deployment-service.yml
deployment.apps/webserver-deployment created
service/webserver-service created
PS C:\Users\tanwe\Desktop\Git\CS3219_sandbox\Task_A\A2> kubectl get all
NAME                                       READY   STATUS    RESTARTS   AGE
pod/webserver-deployment-648bf474bf-b7kg4   1/1     Running   0          29s
pod/webserver-deployment-648bf474bf-b86nw   1/1     Running   0          29s
pod/webserver-deployment-648bf474bf-p2rrd   1/1     Running   0          29s

NAME                        TYPE        CLUSTER-IP      EXTERNAL-IP   PORT(S)          AGE
service/kubernetes          ClusterIP   10.96.0.1       <none>        443/TCP          15m
service/webserver-service   NodePort    10.106.233.39   <none>        3000:31111/TCP   29s

NAME                                   READY   UP-TO-DATE   AVAILABLE   AGE
deployment.apps/webserver-deployment   3/3     3            3           30s

NAME                                              DESIRED   CURRENT   READY   AGE
replicaset.apps/webserver-deployment-648bf474bf   3         3         3       30s
PS C:\Users\tanwe\Desktop\Git\CS3219_sandbox\Task_A\A2>
```

5. After running the above command, a browser will pop up with the assigned port. In this case, you can access the service at `http://localhost:31111/`.



Welcome to WeiJie's static page.

Id: A0202017B

6. To close kubernetes deployment and services, run `kubectl delete -f ./deployment-service.yml`.

Learning Points

- In order to use locally created images, I need to set `imagePullPolicy: Never` and enable Docker Desktop's Kubernetes settings.
- Kubernetes are usually run on cloud, it is rarely run locally.

Resources

Resources that are used and referred to during the creation of this project.

Desc	Link
How to Run Locally Built Docker Images in Kubernetes	https://medium.com/swlh/how-to-run-locally-built-docker-images-in-kubernetes-b28fbc32cc1d
To expose port of service in Minikube	https://stackoverflow.com/questions/40767164/expose-port-in-minikube
Deploy Docker Desktop on Kubernetes	https://docs.docker.com/desktop/kubernetes/