###Please Explain all the instructions and commands you used to complete each Point in MD file

EX:

Q1:

explanation

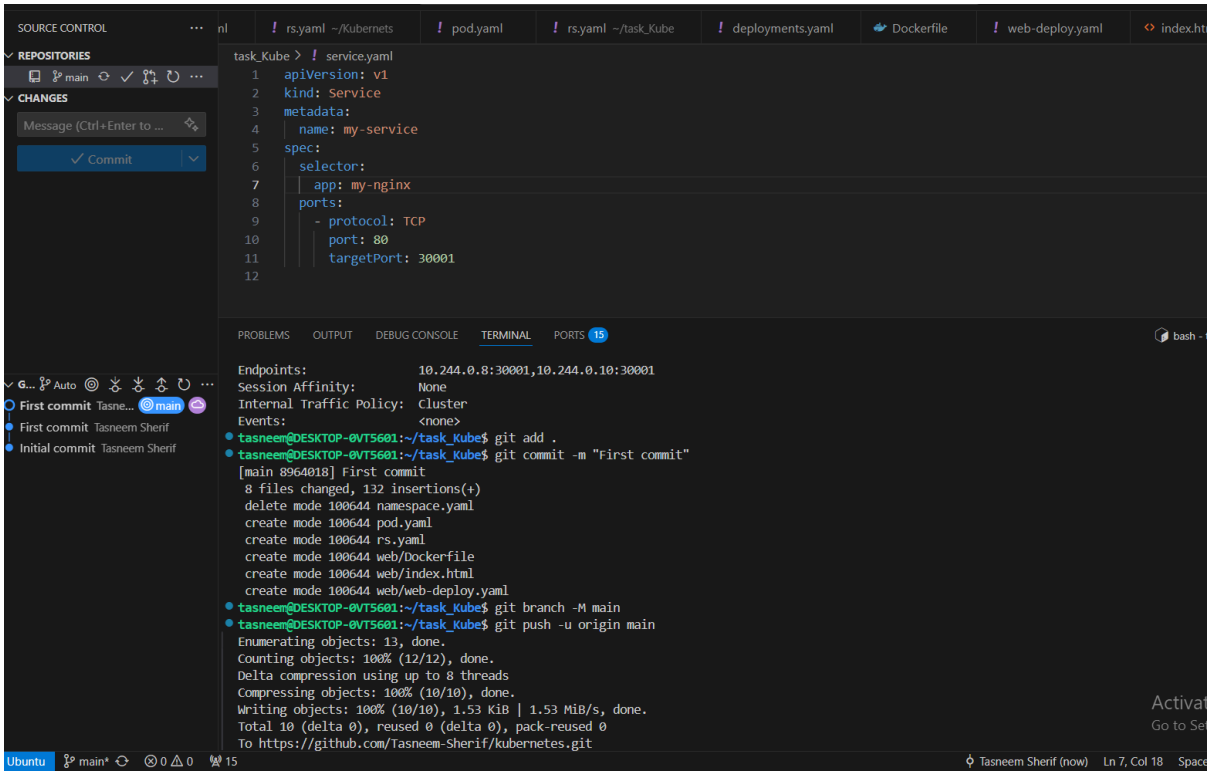
Command used

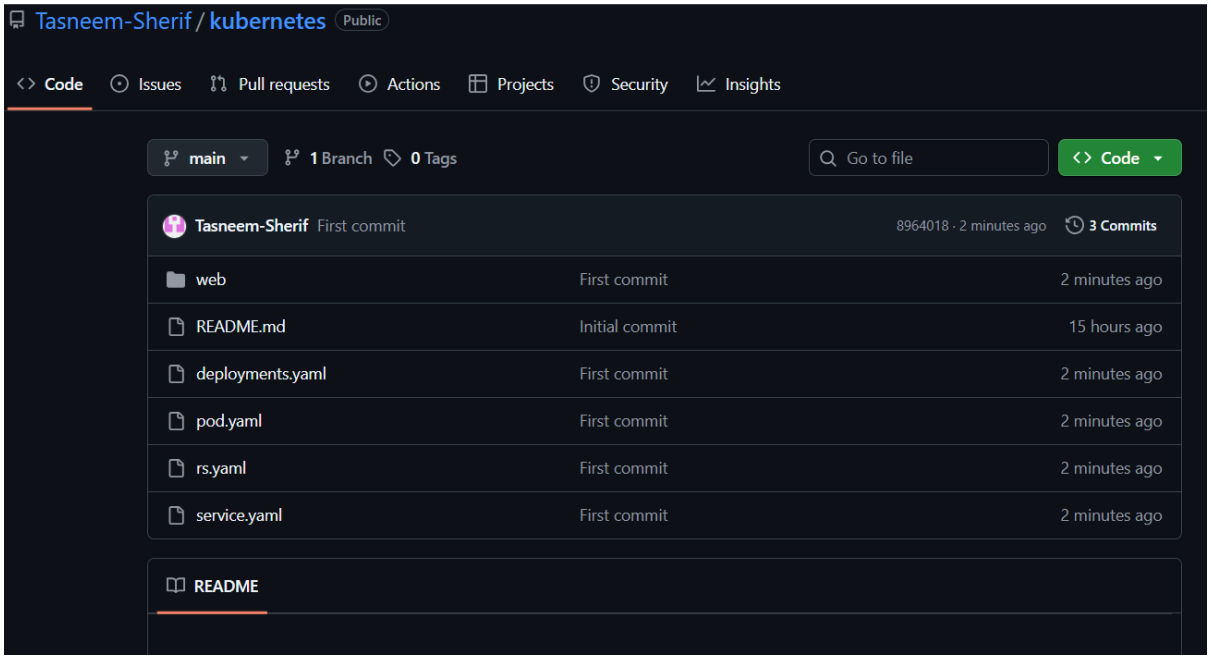
Output of the command

Screenshot for terminal and browser in case exposing service

- Provide YAML files for all Kubernetes objects created (namespace.yaml, deployments.yaml, service.yaml, etc.). Include the output of relevant kubectl commands that verify the successful completion of each step. Provide screenshots showing the services running and accessible

- Push all of them into your GitHub account.



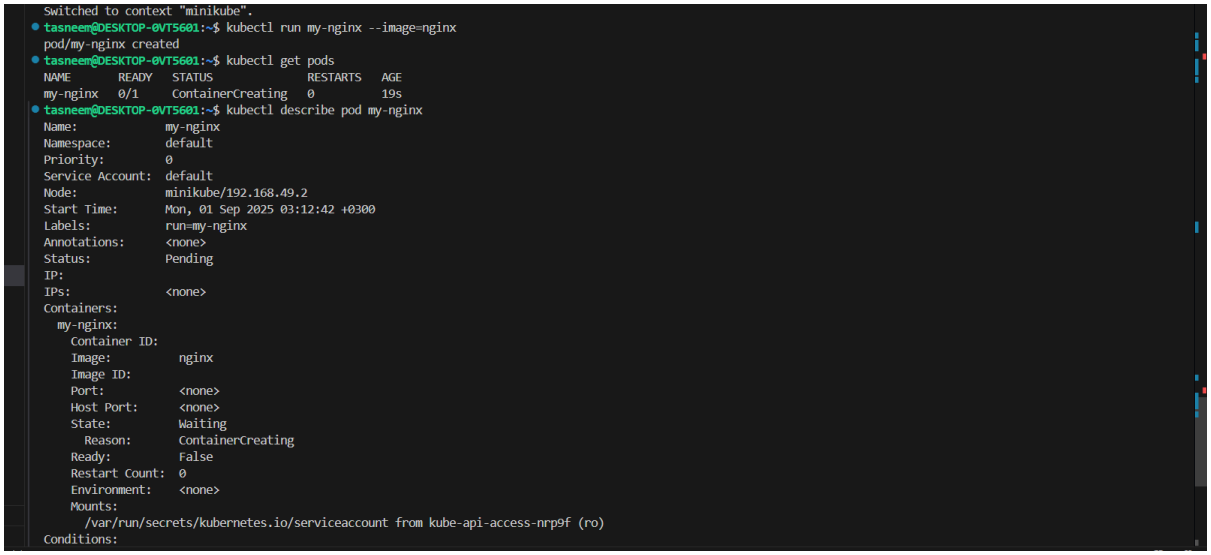


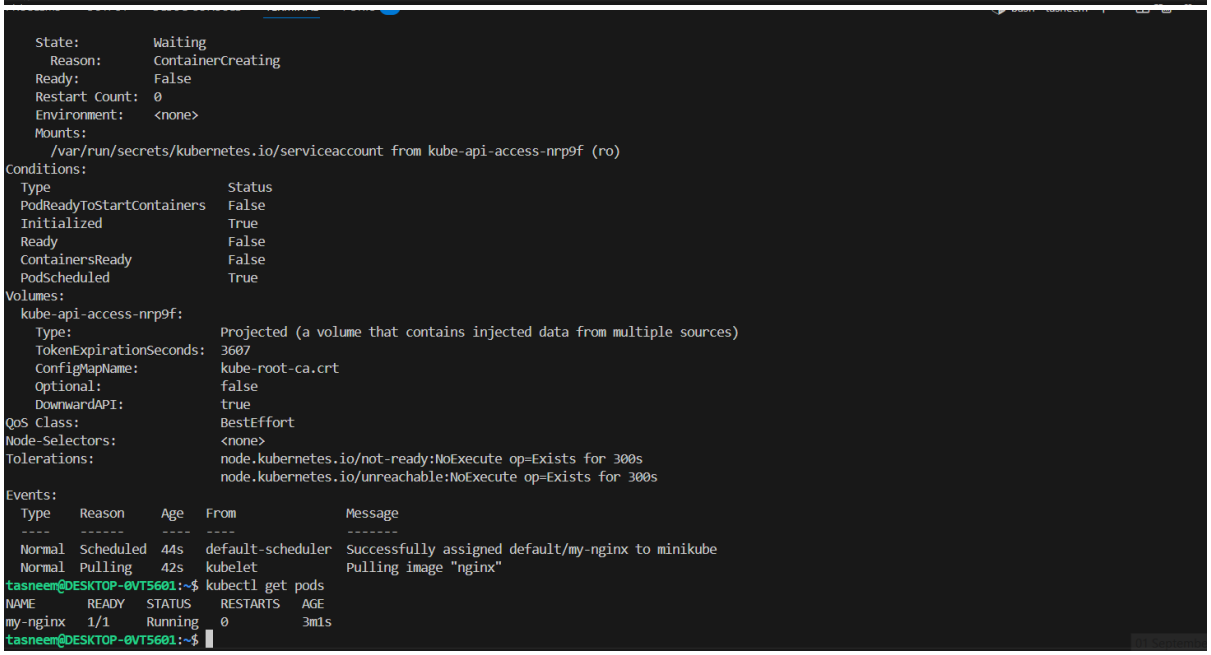
=========

Part1:

1- create pod nginx with name my nginx direct from command don't use yaml file

kubectl run my-nginx --image=nginx
kubectl get pods



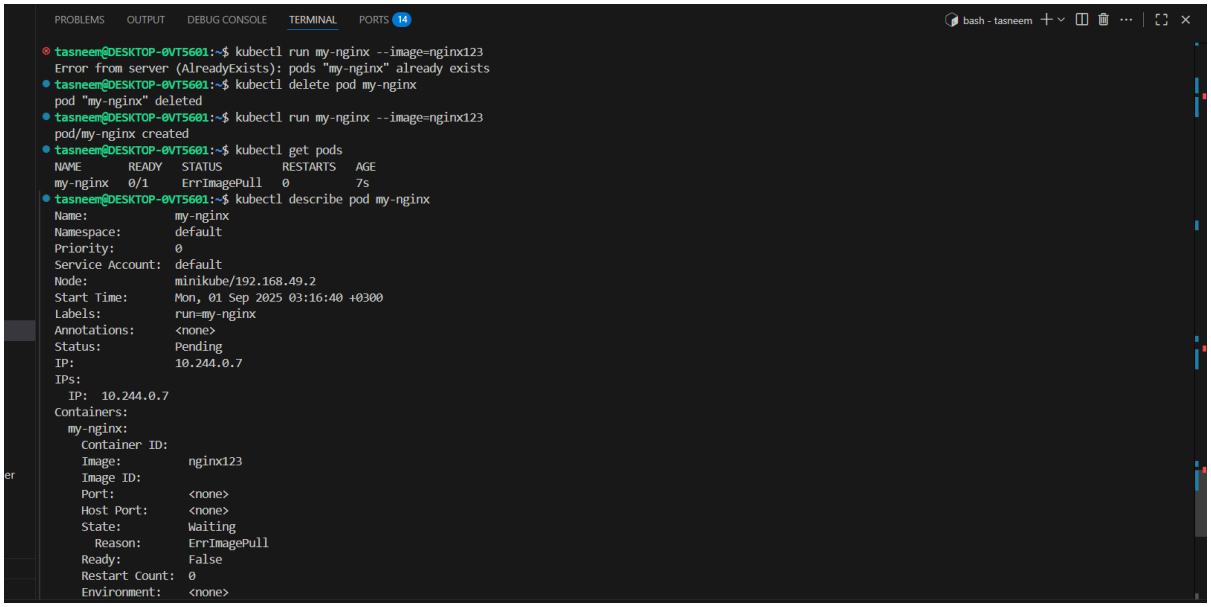


2- create pod nginx with name my nginx command and use Image nginx123 direct from command don't use yaml file

kubectl run my-nginx --image=nginx123

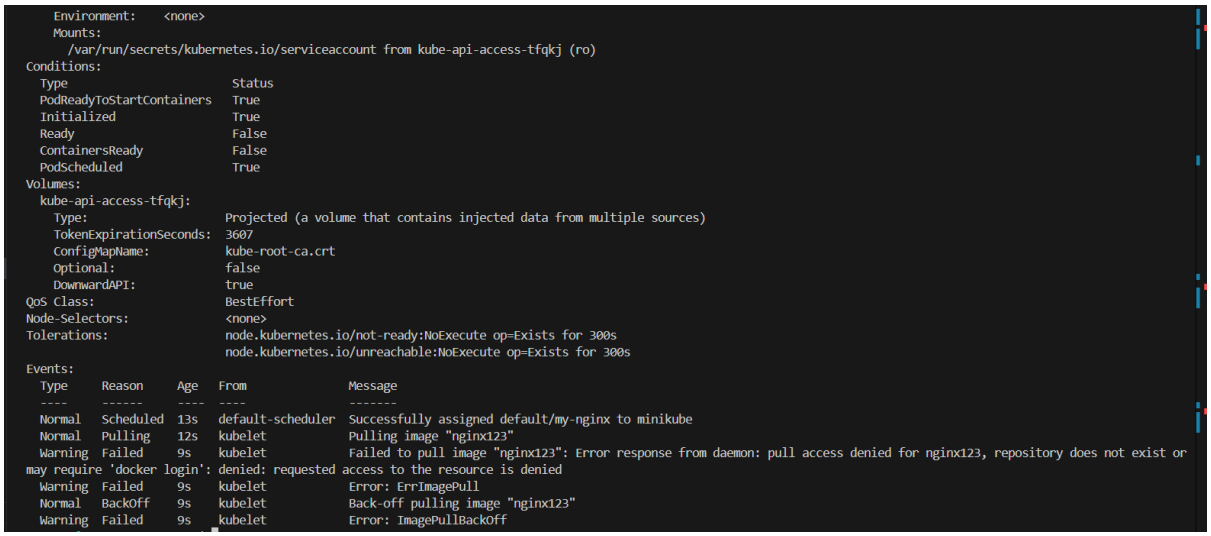
kubectl get pods

kubectl describe pod my-nginx



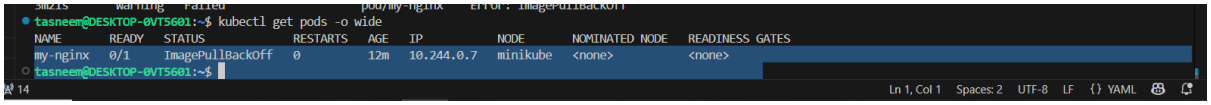
3- check the status and why it doesn't work

kubectl describe pod my-nginx



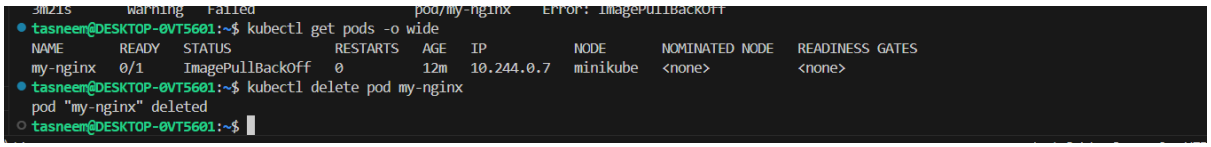
4- I need to know node name - IP - Image Of the POD

kubectl get pods -o wide



5- delete the pod

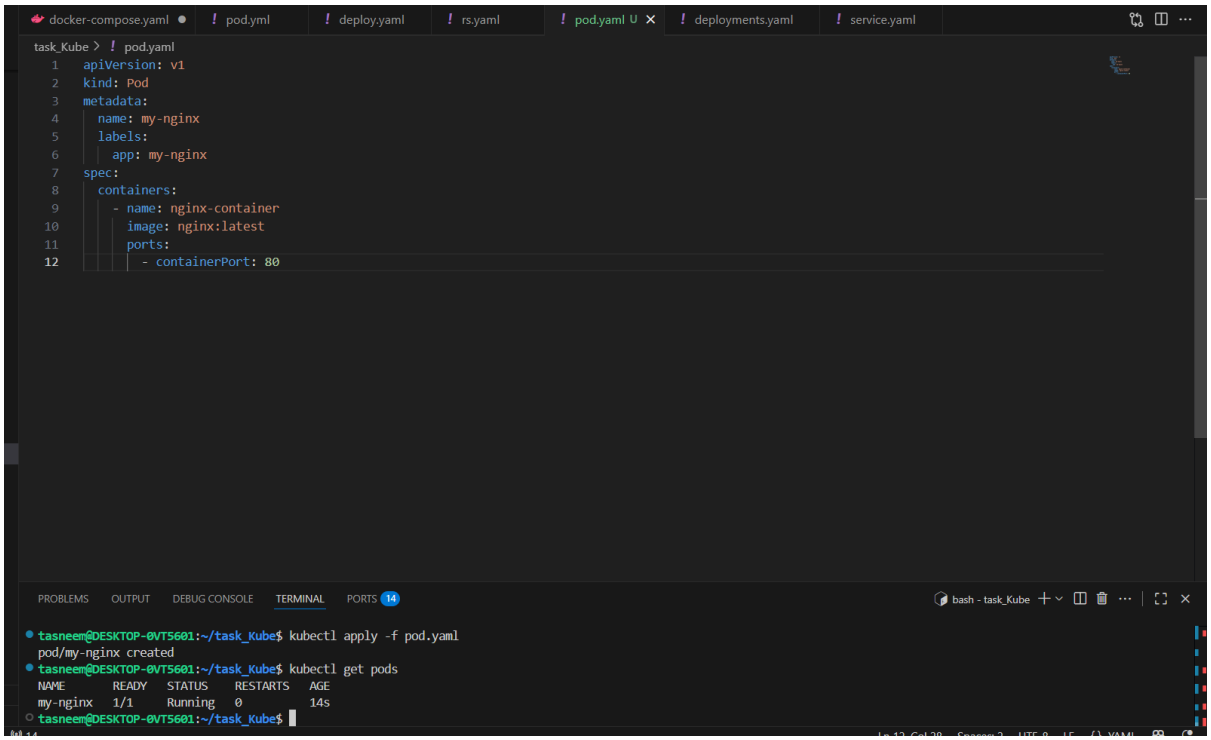
kubectl delete pod my-nginx



6- create another one with yaml file and use label

kubectl apply -f pod.yaml

kubectl get pods

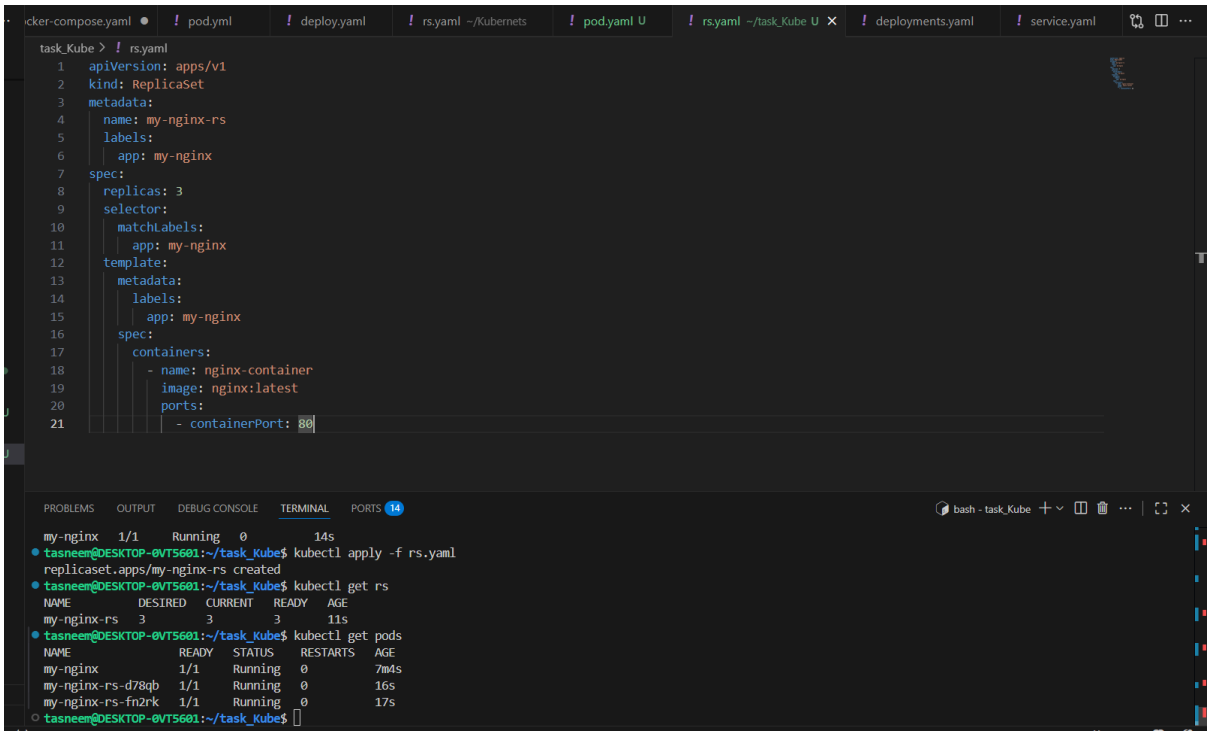


7-create Riplicaset with 3 replicas using nginx Image

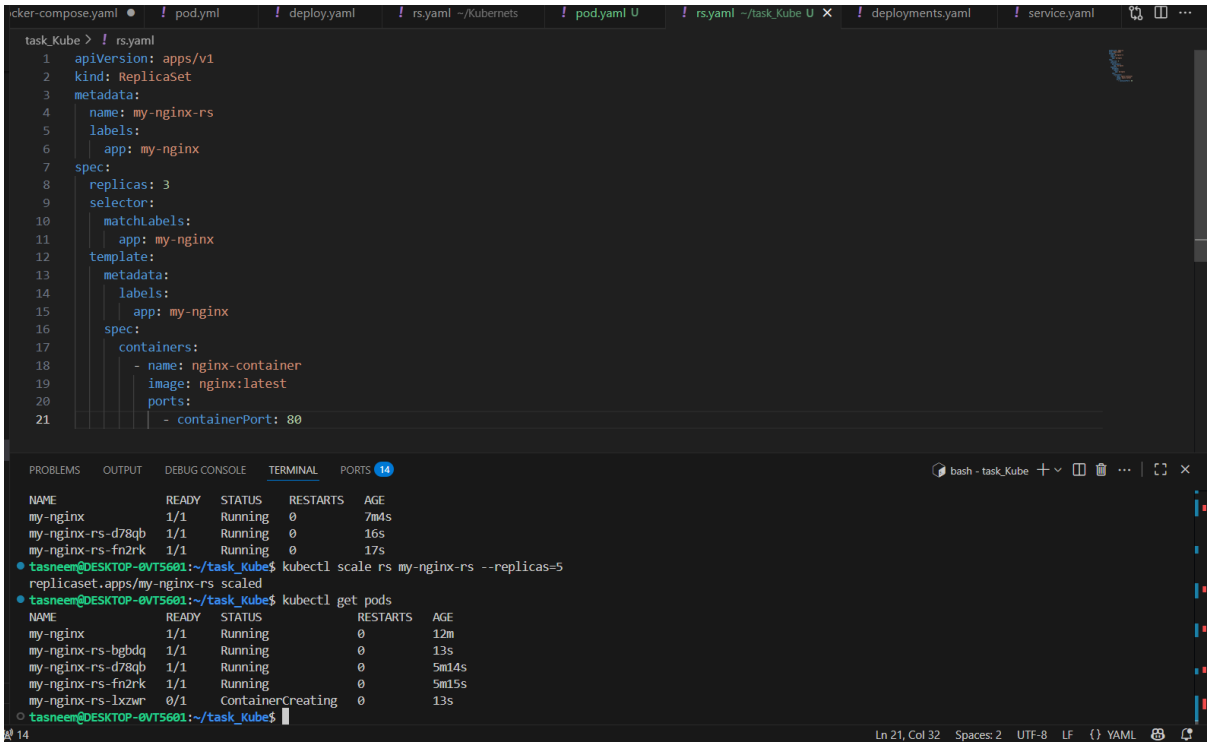
kubectl apply -f rs.yaml

kubectl get rs

kubectl get pods



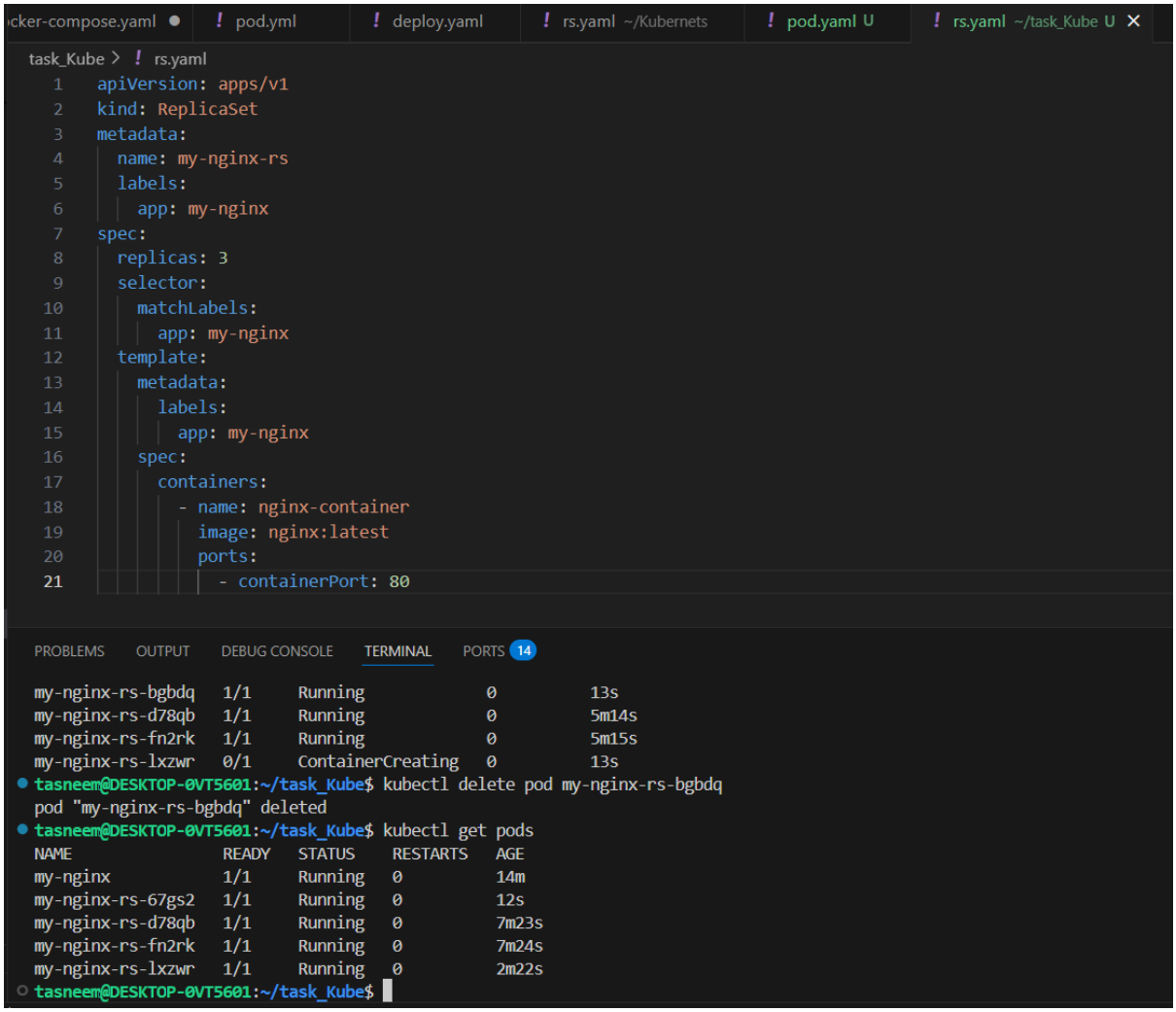
8-scale the replicas to 5 without edit in the Yaml file
kubectl scale rs my-nginx-rs --replicas=5
kubectl get pods



9-Delete any one of the 5 pods and check what happen and explain

If you delete a Pod that was created by a ReplicaSet or a Deployment, Kubernetes will immediately create a new Pod to replace it.
This happens because the Deployment (or ReplicaSet) always tries to maintain the desired number of replicas.

kubectl delete pod my-nginx-rs-bgbdq
kubectl get pods



10-Scale down the pods aging to 2 without scale command use terminal

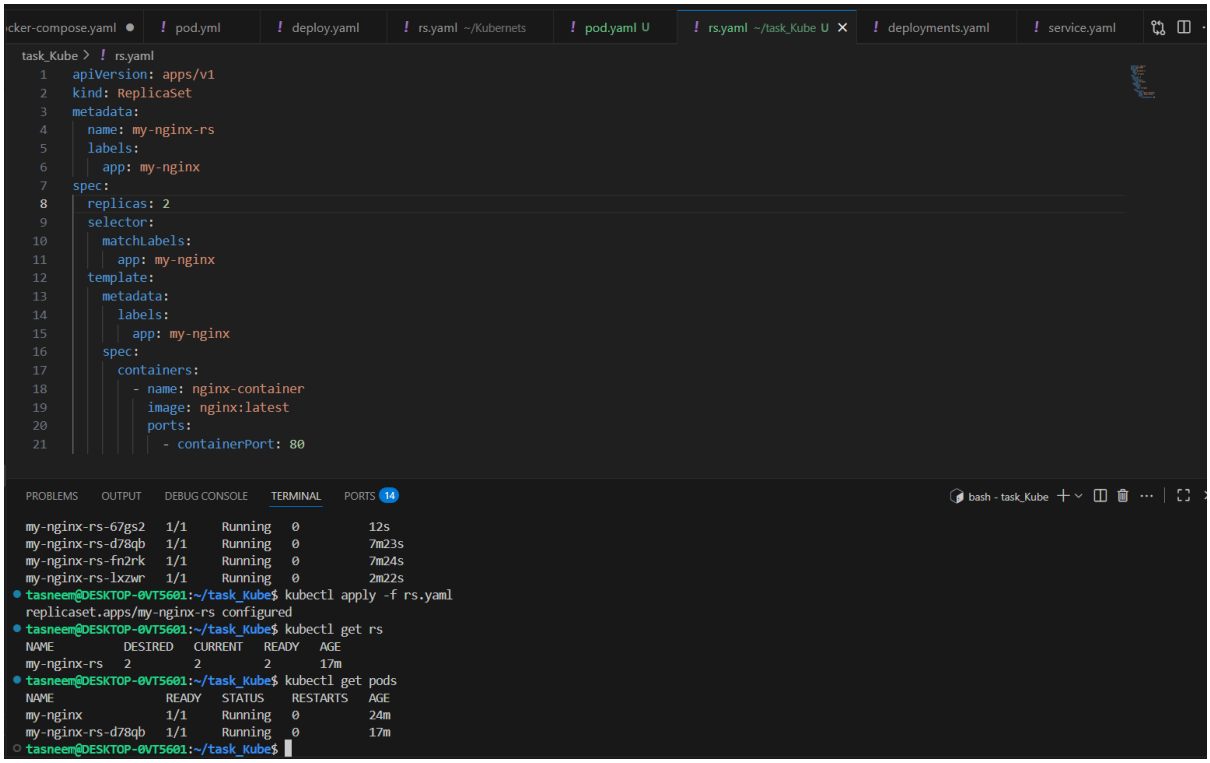
in Yaml file:

spec:

replicas: 2

kubectl apply -f rs.yaml

kubectl get pods



11- find out the issue in the below Yaml (don't use AI)

apiVersion: apps/v1

kind: ReplicaSet

metadata:

 name: replicaset-2

spec:

 replicas: 2

 selector:

  matchLabels:

   tier: frontend

 template:

  metadata:

   labels:

    tier: nginx

  spec:

```
    containers:
    - name: nginx
      image: nginx
```

The issue is that the selector labels do not match the template labels. A ReplicaSet only manages Pods that match its selector. Since tier: frontend don't match tier: nginx, the ReplicaSet won't manage the created Pods. To fix it, make both labels the same

Sol:

```
apiVersion: apps/v1
kind: ReplicaSet
metadata:
  name: replicaset-2
spec:
  replicas: 2
  selector:
    matchLabels:
      tier: nginx
  template:
    metadata:
      labels:
        tier: nginx
    spec:
      containers:
      - name: nginx
        image: nginx
```

12- find out the issue in the below Yaml (don't use AI)

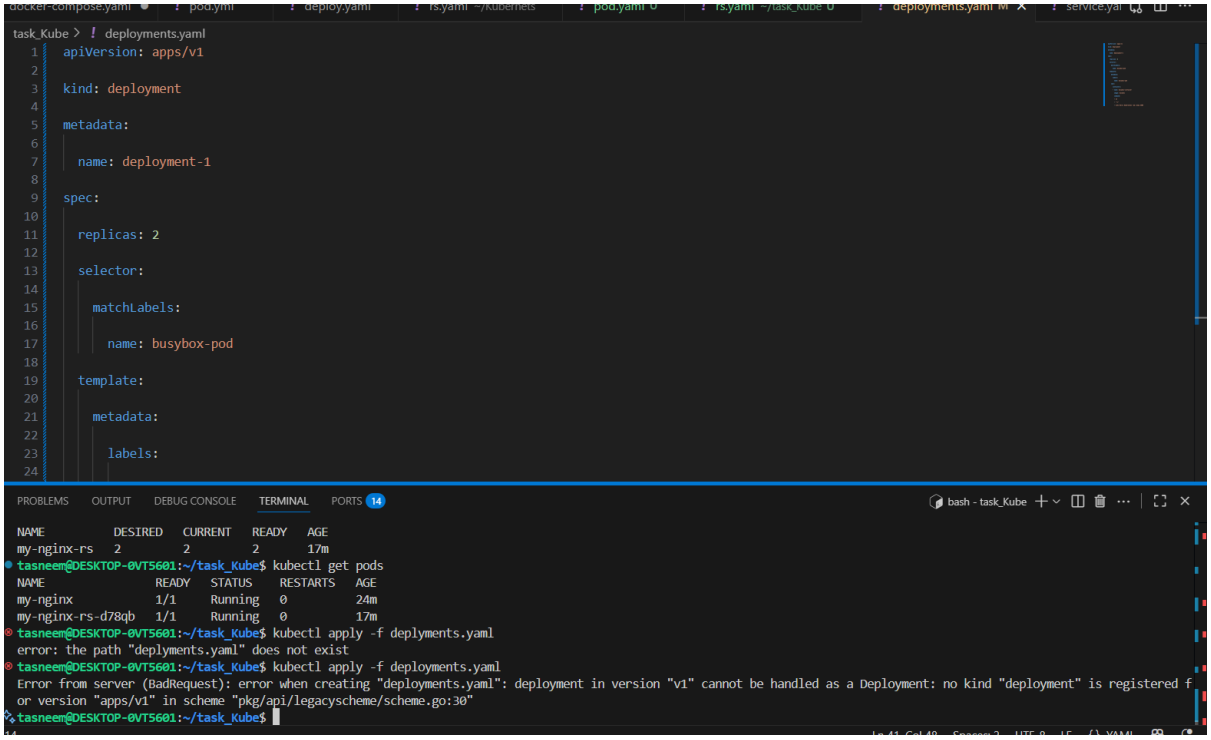
```
apiVersion: apps/v1
```

```
kind: deployment
metadata:
  name: deployment-1
spec:
  replicas: 2
  selector:
    matchLabels:
      name: busybox-pod
  template:
    metadata:
      labels:
        name: busybox-pod
    spec:
      containers:
      - name: busybox-container
        image: busybox
        command:
        - sh
        - "-c"
        - echo Hello Kubernetes! && sleep 3600
```

The issue is that kind is written as deployment with a small 'd'. Kubernetes is case-sensitive, so it should be Deployment
sol:

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: deployment-1
spec:
```

```
replicas: 2
selector:
  matchLabels:
    name: busybox-pod
template:
  metadata:
    labels:
      name: busybox-pod
  spec:
    containers:
    - name: busybox-container
      image: busybox
      command:
      - sh
      - "-c"
      - echo Hello Kubernetes! && sleep 3600
```



13- find out the issue in the below Yaml (don't use AI)

apiVersion: v1

kind: Deployment

metadata:

name: deployment-1

spec:

replicas: 2

selector:

matchLabels:

name: busybox-pod

template:

metadata:

labels:

name: busybox-pod

spec:

```
    containers:
    - name: busybox-container
      image: busybox
      command:
      - sh
      - "-c"
      - echo Hello Kubernetes! && sleep 3600
```

The issue is that the apiVersion is set to v1. Deployments belong to the apps/v1 API group, not v1. Changing it to apps/v1 fixes the error
sol:

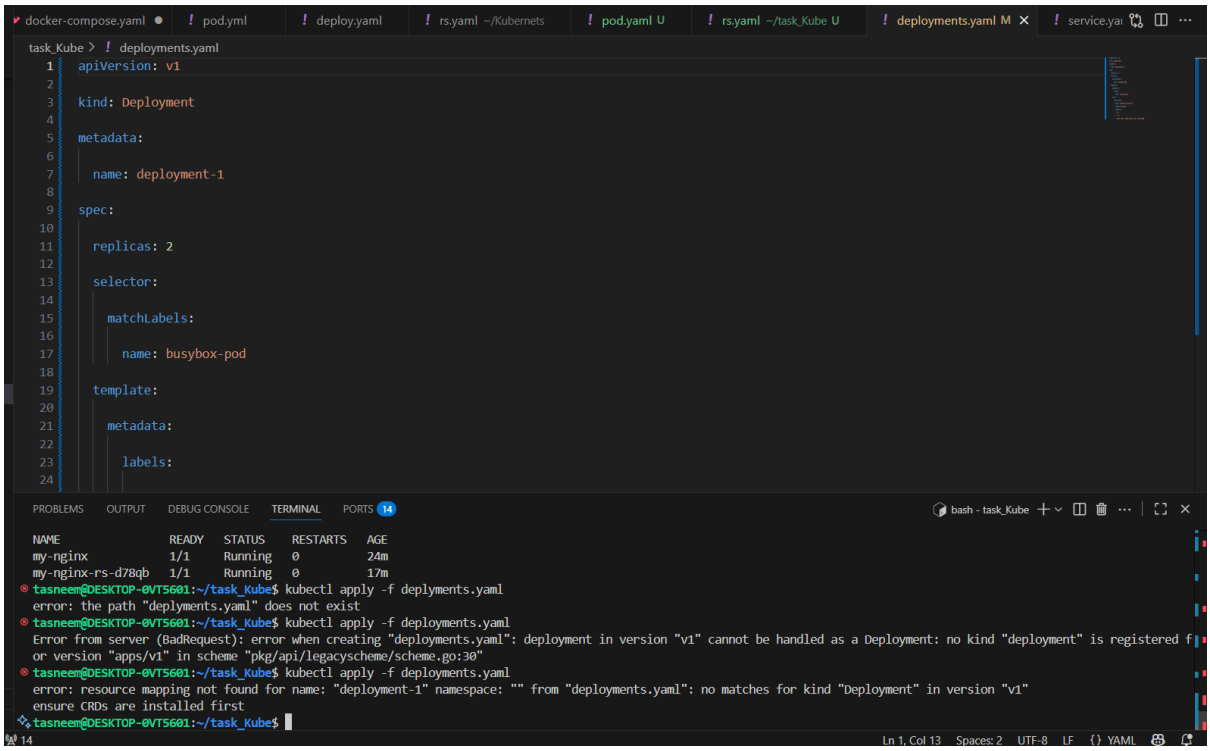
```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: deployment-1
spec:
  replicas: 2
  selector:
    matchLabels:
      name: busybox-pod
  template:
    metadata:
      labels:
        name: busybox-pod
    spec:
      containers:
      - name: busybox-container
        image: busybox
```

command:

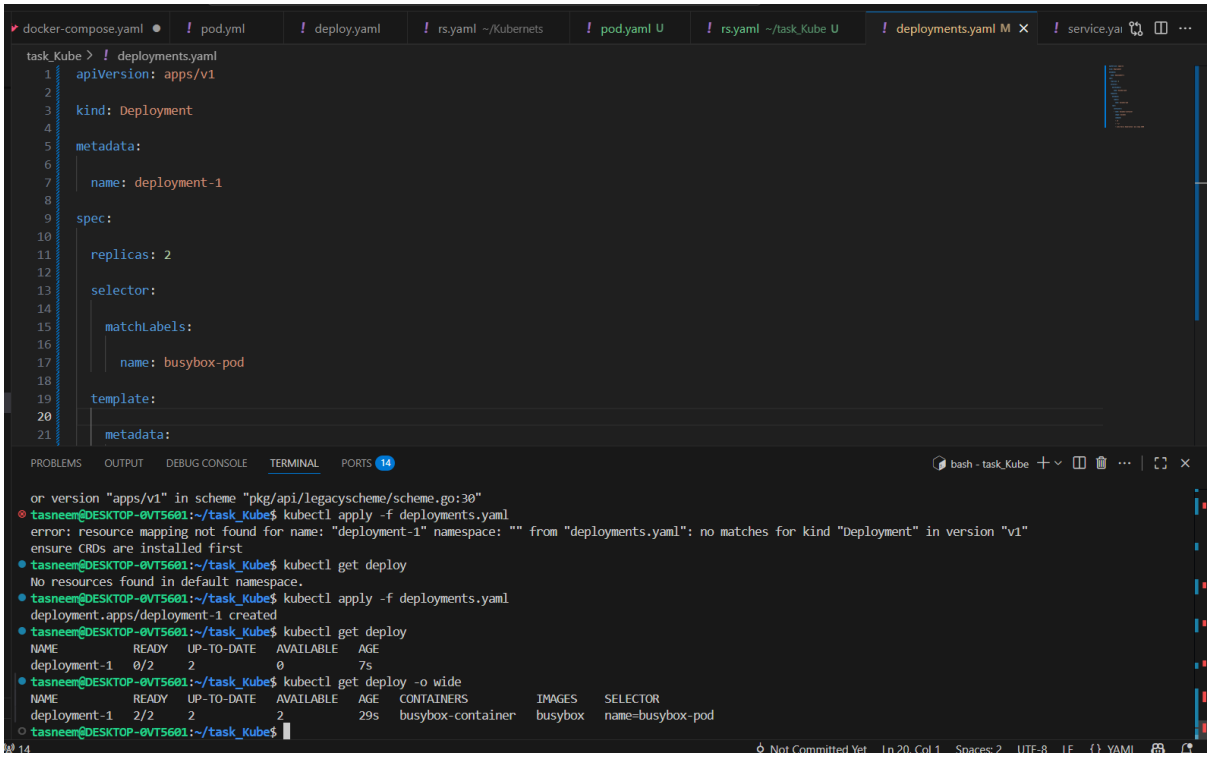
- sh

- "-c"

- echo Hello Kubernetes! && sleep 3600



14- what's command you use to know what Image name that running the deployment
kubectl get deploy -o wide



15- create deployment using following data :

Name: httpd-frontend;

Replicas: 3;

Image: httpd:2.4-alpine
in Yaml file:
apiVersion: apps/v1

kind: Deployment

metadata:

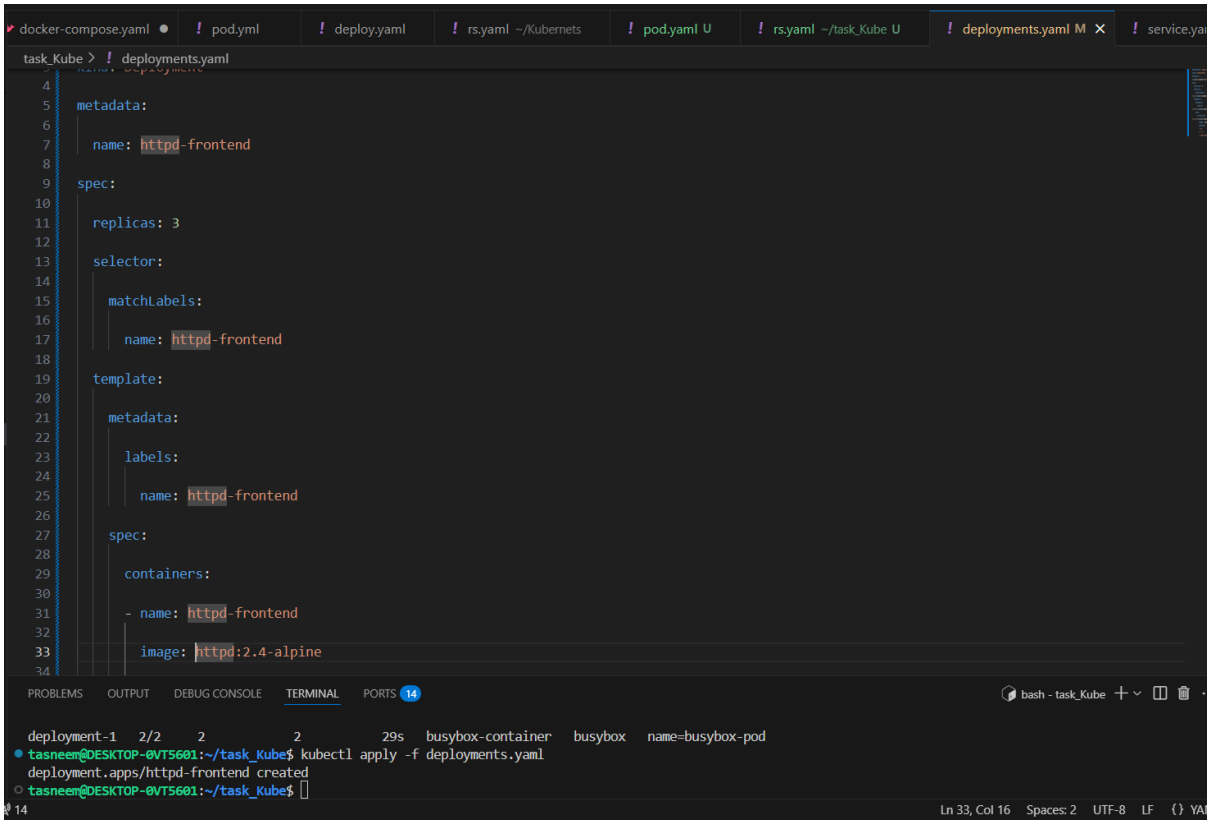
name: httpd-frontend

spec:

```
replicas: 3

selector:

 matchLabels:

  name: httpd-frontend

template:

 metadata:

  labels:

   name: httpd-frontend

 spec:

  containers:

  - name: httpd-frontend

   image: httpd:2.4-alpine

   command:

   - sh
```

- "-c"

- echo Hello Kubernetes! && sleep 3600



16- replace the image to nginx777 with command directly

kubectl set image deploy httpd-frontend httpd-frontend=nginx777
kubectl describe pod httpd-frontend

```
tasneem@DESKTOP-0VT5601:~/task_Kube$ kubectl get deploy -o wide
NAME           READY   UP-TO-DATE   AVAILABLE   AGE     CONTAINERS          IMAGES            SELECTOR
deployment-1   2/2     2            2           9m31s   busybox-container   busybox           name=busybox-pod
httpd-frontend   3/3     3            3           2m37s   httpd-frontend      httpd:2.4-alpine  name=httpd-frontend
tasneem@DESKTOP-0VT5601:~/task_Kube$ kubectl get pods
NAME                              READY   STATUS    RESTARTS   AGE
deployment-1-858c547685-2p5vx     1/1     Running   0          9m46s
deployment-1-858c547685-ckgxz     1/1     Running   0          9m46s
httpd-frontend-65c89bbdc5-5r6jb   1/1     Running   0          2m53s
httpd-frontend-65c89bbdc5-k4sxw   1/1     Running   0          2m53s
httpd-frontend-65c89bbdc5-ms85p   1/1     Running   0          2m53s
my-nginx                          1/1     Running   0          63m
my-nginx-rs-d78qb                 1/1     Running   0          56m
tasneem@DESKTOP-0VT5601:~/task_Kube$ kubectl set image deploy httpd-frontend httpd-frontend=nginx777
deployment.apps/httpd-frontend image updated
tasneem@DESKTOP-0VT5601:~/task_Kube$ kubectl get deploy -o wide
NAME           READY   UP-TO-DATE   AVAILABLE   AGE     CONTAINERS          IMAGES      SELECTOR
deployment-1   2/2     2            2           14m     busybox-container   busybox     name=busybox-pod
httpd-frontend   3/3     1            3           7m16s   httpd-frontend      nginx777    name=httpd-frontend
tasneem@DESKTOP-0VT5601:~/task_Kube$ kubectl get pods
NAME                              READY   STATUS             RESTARTS   AGE
deployment-1-858c547685-2p5vx     1/1     Running            0          14m
deployment-1-858c547685-ckgxz     1/1     Running            0          14m
httpd-frontend-65c89bbdc5-5r6jb   1/1     Running            0          7m27s
httpd-frontend-65c89bbdc5-k4sxw   1/1     Running            0          7m27s
httpd-frontend-65c89bbdc5-ms85p   1/1     Running            0          7m27s
httpd-frontend-759779c458-6lr2s   0/1     ImagePullBackOff   0          21s
my-nginx                          1/1     Running            0          68m
my-nginx-rs-d78qb                 1/1     Running            0          61m
```

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS 14                                  bash - task_Kube

Conditions:
  Type                        Status
  PodReadyToStartContainers   True
  Initialized                 True
  Ready                       False
  ContainersReady             False
  PodScheduled                True
Volumes:
  kube-api-access-64cxs:
    Type:                     Projected (a volume that contains injected data from multiple sources)
    TokenExpirationSeconds:   3607
    ConfigMapName:            kube-root-ca.crt
    Optional:                 false
    DownwardAPI:              true
QoS Class:                    BestEffort
Node-Selectors:               <none>
Tolerations:                  node.kubernetes.io/not-ready:NoExecute op=Exists for 300s
                              node.kubernetes.io/unreachable:NoExecute op=Exists for 300s

Events:
  Type     Reason     Age                  From               Message
  ----     ------     ----                 ----               -------
  Normal   Scheduled  7m11s                default-scheduler  Successfully assigned default/httpd-frontend-759779c458-6lr2s to minikube
  Normal   Pulling    3m49s (x5 over 7m9s)  kubelet            Pulling image "nginx777"
  Warning  Failed     3m44s (x5 over 7m7s)  kubelet            Failed to pull image "nginx777": Error response from daemon: pull access denied for nginx777, repository do
es not exist or may require 'docker login': denied: requested access to the resource is denied
  Warning  Failed     3m44s (x5 over 7m7s)  kubelet            Error: ErrImagePull
  Warning  Failed     103s (x20 over 7m7s)  kubelet            Error: ImagePullBackOff
  Normal   BackOff    90s (x21 over 7m7s)   kubelet            Back-off pulling image "nginx777"

tasneem@DESKTOP-0VT5601:~/task_Kube$
```

17- rollback to pervious version

kubectl rollout undo deploy httpd-frontend

```
tasneem@DESKTOP-0VT5601:~/task_Kube$ kubectl rollout undo deploy httpd-frontend
deployment.apps/httpd-frontend rolled back
tasneem@DESKTOP-0VT5601:~/task_Kube$ kubectl get pods
NAME                              READY   STATUS    RESTARTS   AGE
deployment-1-858c547685-2p5vx     1/1     Running   0          27m
deployment-1-858c547685-ckgxz     1/1     Running   0          27m
httpd-frontend-65c89bbdc5-5r6jb   1/1     Running   0          20m
httpd-frontend-65c89bbdc5-k4sxw   1/1     Running   0          20m
httpd-frontend-65c89bbdc5-ms85p   1/1     Running   0          20m
my-nginx                          1/1     Running   0          81m
my-nginx-rs-d78qb                 1/1     Running   0          74m
tasneem@DESKTOP-0VT5601:~/task_Kube$
```

18- Create a Simple Web Application:

* Use a Dockerfile to create a simple web application (e.g., an Nginx server serving an HTML page).

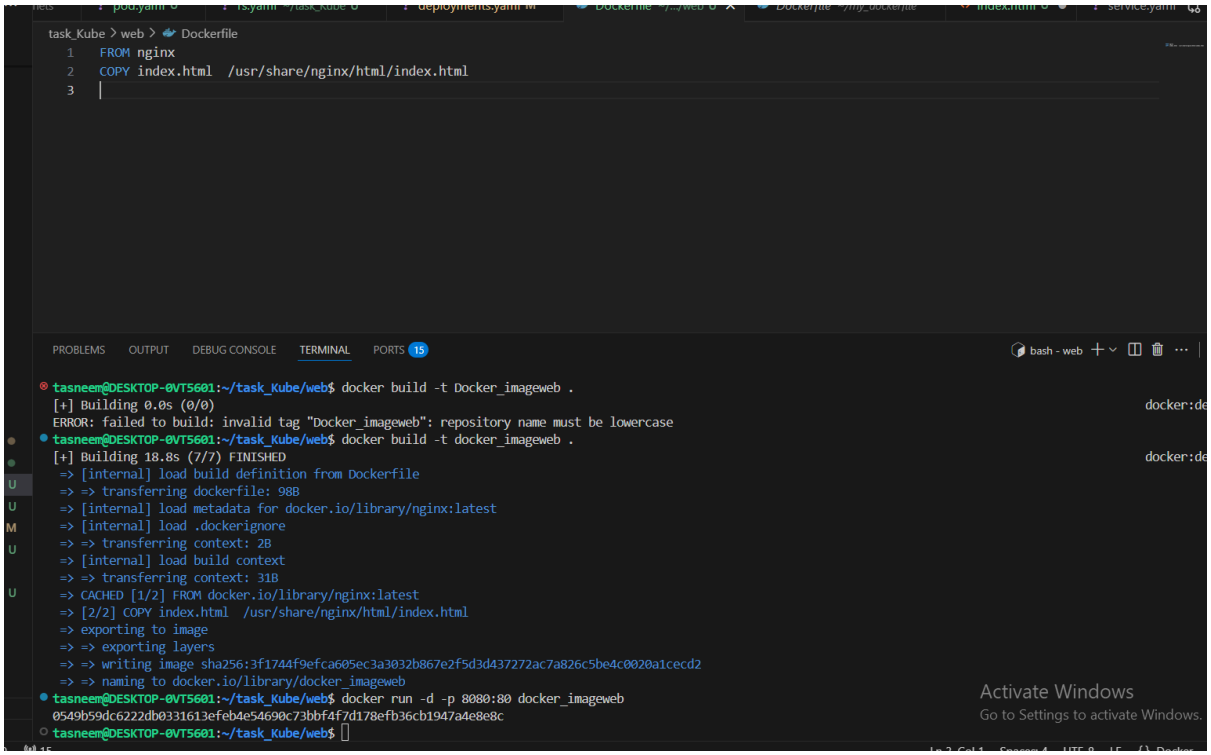
* Build the Docker image and push it to DockerHub your private Account.

docker build -t docker_imageweb .

docker run -d -p 8080:80 docker_imageweb

docker login

docker tag docker_imageweb tasneemsherif/docker_imageweb:latest
docker push tasneemsherif/docker_imageweb:latest




```
tasneem@DESKTOP-0VT5601:~/task_Kube/web$ docker login

USING WEB-BASED LOGIN

i Info → To sign in with credentials on the command line, use 'docker login -u <username>'


Your one-time device confirmation code is: NVQQ-VBFN
Press ENTER to open your browser or submit your device code here: https://login.docker.com/activate

Waiting for authentication in the browser…

WARNING! Your credentials are stored unencrypted in '/home/tasneem/.docker/config.json'.
Configure a credential helper to remove this warning. See
https://docs.docker.com/go/credential-store/

Login Succeeded
```



```
tasneem@DESKTOP-0VT5601:~/task_Kube/web$ docker tag docker_imageweb tasneemsherif/docker_imageweb:latest
tasneem@DESKTOP-0VT5601:~/task_Kube/web$ docker push tasneemsherif/docker_imageweb:latest
The push refers to repository [docker.io/tasneemsherif/docker_imageweb]
5045ed8967f3: Pushed
f17478b6e8f3: Pushed
0662742b23b2: Pushed
5c91a024d899: Pushed
6b1b97dc9285: Pushed
a6b19c3d00b1: Pushed
30837a0774b9: Pushed
7cc7fe68eff6: Pushed
latest: digest: sha256:4d0bf59a522023b33ff4f5120251bebdfb680698444f06ff7d43351fea50dbcd size: 1985
```

19- Create a Deployment Using This Image:

* Deploy the Docker image from DockerHub to Kubernetes with a Deployment that has 3 replicas.

In Yaml file:
apiVersion: apps/v1

kind: Deployment

metadata:

name: web-deploy

```
spec:
 replicas: 3
 selector:
  matchLabels:
   name: web-pod
 template:
  metadata:
   labels:
    name: web-pod
  spec:
   containers:
   - name: web-container
     image: tasneemsherif/docker_imageweb
     ports:
      - containerPort: 80
```

Cmd:

kubectl apply -f web-deploy.yaml

kubectl get pods

```
task_Kube > web > ! web-deploy.yaml
  9   spec:
 19     template:
 21       metadata:
 23         labels:
 26
 27       spec:
 28
 29         containers:
 30
 31         - name: web-container
 32
 33           image: tasneemsherif/docker_imageweb
 34           ports:
 35           - containerPort: 80
```

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS 15

tasneem@DESKTOP-0VT5601:~/task_Kube/web$ kubectl apply -f web-deploy.yaml
deployment.apps/web-deploy created
tasneem@DESKTOP-0VT5601:~/task_Kube/web$ kubectl get deploy

NAME            READY   UP-TO-DATE   AVAILABLE   AGE
deployment-1    2/2     2            2           12h
httpd-frontend  3/3     3            3           12h
web-deploy      0/3     3            0           3m32s
tasneem@DESKTOP-0VT5601:~/task_Kube/web$
tasneem@DESKTOP-0VT5601:~/task_Kube/web$ kubectl get pods
NAME                              READY   STATUS              RESTARTS      AGE
deployment-1-858c547685-2p5vx     1/1     Running             3 (50m ago)   12h
deployment-1-858c547685-ckgxz     1/1     Running             3 (50m ago)   12h
httpd-frontend-65c89bbdc5-5r6jb   1/1     Running             3 (46m ago)   12h
httpd-frontend-65c89bbdc5-k4sxw   1/1     Running             3 (46m ago)   12h
httpd-frontend-65c89bbdc5-ms85p   1/1     Running             3 (46m ago)   12h
my-nginx                          1/1     Running             0             13h
my-nginx-rs-d78qb                 1/1     Running             0             13h
web-deploy-7dcdb65bff-2xhmw       0/1     ContainerCreating   0             5m52s
web-deploy-7dcdb65bff-dqfct       0/1     ContainerCreating   0             5m53s
web-deploy-7dcdb65bff-k6nm4       0/1     ContainerCreating   0             5m52s
tasneem@DESKTOP-0VT5601:~/task_Kube/web$
```

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS 15

tasneem@DESKTOP-0VT5601:~/task_Kube/web$ kubectl get pods
NAME                              READY   STATUS    RESTARTS      AGE
deployment-1-858c547685-2p5vx     1/1     Running   3 (58m ago)   12h
deployment-1-858c547685-ckgxz     1/1     Running   3 (58m ago)   12h
httpd-frontend-65c89bbdc5-5r6jb   1/1     Running   3 (54m ago)   12h
httpd-frontend-65c89bbdc5-k4sxw   1/1     Running   3 (53m ago)   12h
httpd-frontend-65c89bbdc5-ms85p   1/1     Running   3 (53m ago)   12h
my-nginx                          1/1     Running   0             13h
my-nginx-rs-d78qb                 1/1     Running   0             13h
web-deploy-7dcdb65bff-2xhmw       1/1     Running   0             13m
web-deploy-7dcdb65bff-dqfct       1/1     Running   0             13m
web-deploy-7dcdb65bff-k6nm4       1/1     Running   0             13m
```