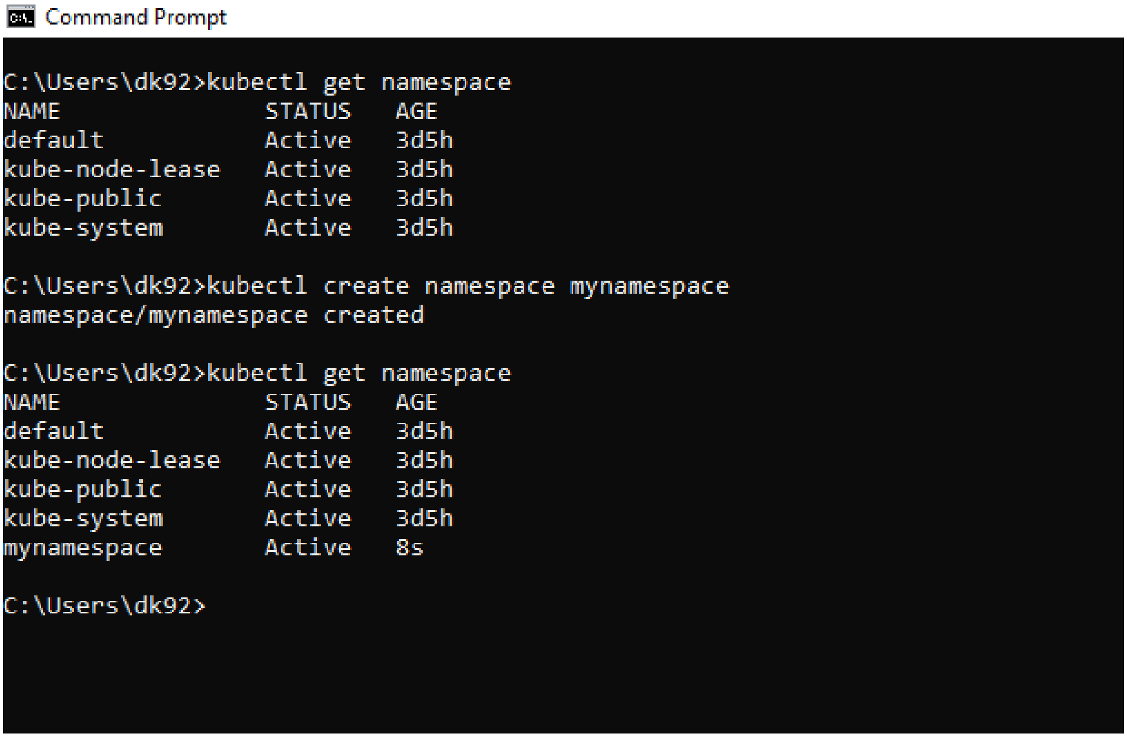
Devops Assignment- Module 7 – Kubernetes

Assignment 1 - Working with Namespaces and Resource Quotas.txt

In this Assignment, we will be working with Namespaces and Resource Quotas

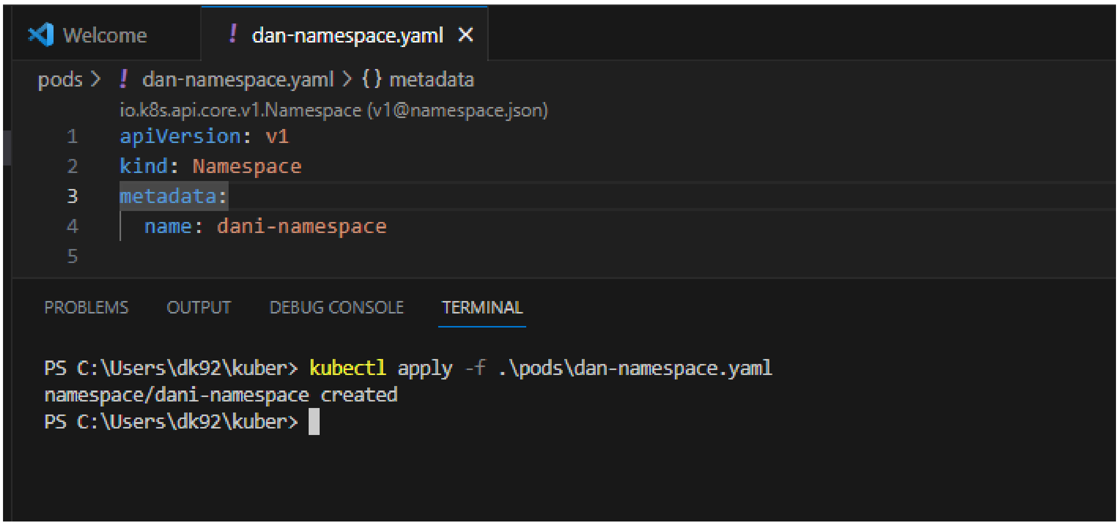
1. Create namespaces using imperative and declarative method

- command to shows the available predefined namespaces
  Kubectl get namespace
- For creating imperative namespace use command
  Kubectl create namespace name.



```
C:\Users\dk92>kubectl get namespace
NAME              STATUS   AGE
default           Active   3d5h
kube-node-lease   Active   3d5h
kube-public       Active   3d5h
kube-system       Active   3d5h

C:\Users\dk92>kubectl create namespace mynamespace
namespace/mynamespace created

C:\Users\dk92>kubectl get namespace
NAME              STATUS   AGE
default           Active   3d5h
kube-node-lease   Active   3d5h
kube-public       Active   3d5h
kube-system       Active   3d5h
mynamespace       Active   8s

C:\Users\dk92>
```

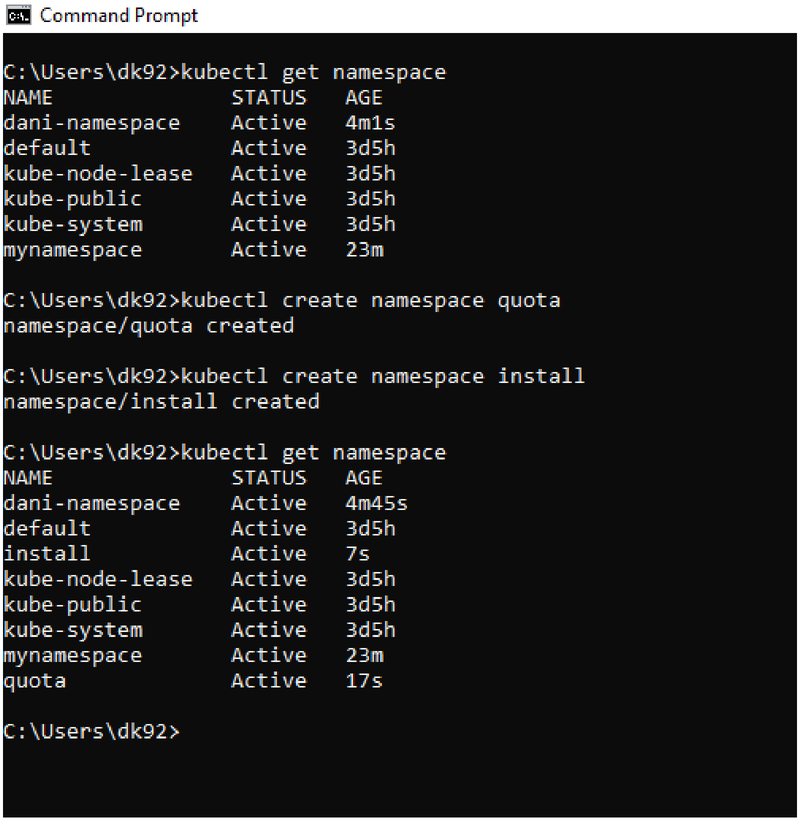
- For declarative namespace we want to write yaml for namespace as given below.
- After creating yaml file for namespace apply it using the below command
  Kubectl apply –f "filename/path"


Welcome
! dan-namespace.yaml ×
pods > ! dan-namespace.yaml > {} metadata
io.k8s.api.core.v1.Namespace (v1@namespace.json)
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL

```yaml
apiVersion: v1
kind: Namespace
metadata:
  name: dani-namespace

```

```
PS C:\Users\dk92\kuber> kubectl apply -f .\pods\dan-namespace.yaml
namespace/dani-namespace created
PS C:\Users\dk92\kuber>
```

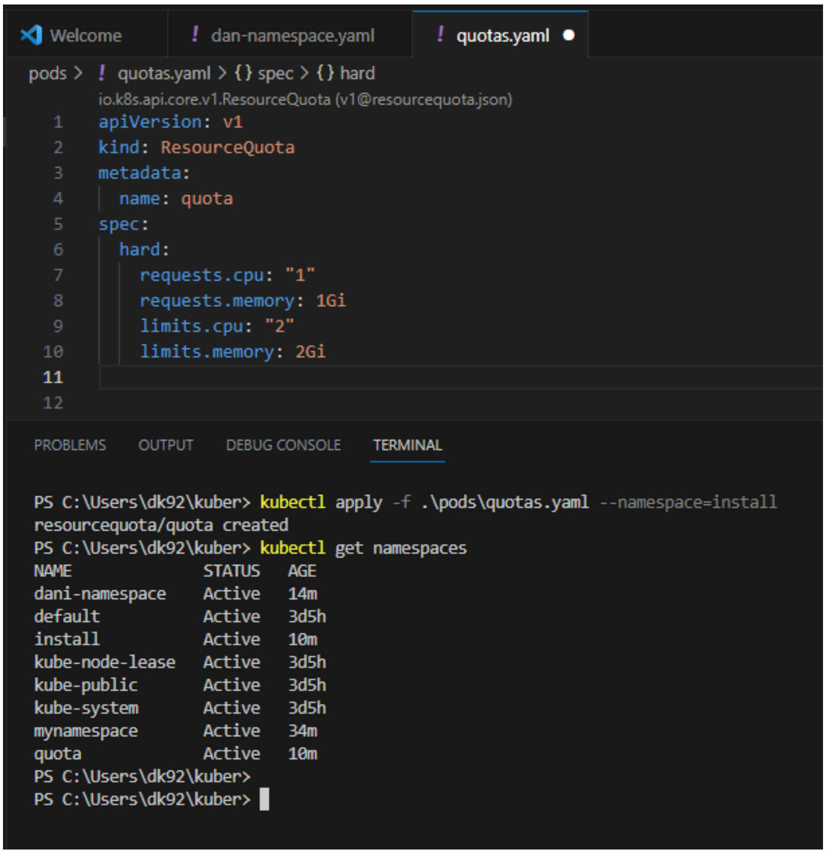
2. Create a new namespace named quota and setup cpu and memory quotas in the namespace Install and configure kubectl



Command Prompt

```
C:\Users\dk92>kubectl get namespace
NAME               STATUS   AGE
dani-namespace     Active   4m1s
default            Active   3d5h
kube-node-lease    Active   3d5h
kube-public        Active   3d5h
kube-system        Active   3d5h
mynamespace        Active   23m

C:\Users\dk92>kubectl create namespace quota
namespace/quota created

C:\Users\dk92>kubectl create namespace install
namespace/install created

C:\Users\dk92>kubectl get namespace
NAME               STATUS   AGE
dani-namespace     Active   4m45s
default            Active   3d5h
install            Active   7s
kube-node-lease    Active   3d5h
kube-public        Active   3d5h
kube-system        Active   3d5h
mynamespace        Active   23m
quota              Active   17s

C:\Users\dk92>
```

- Creating a resource quota provided cpu and memory using declarative method.

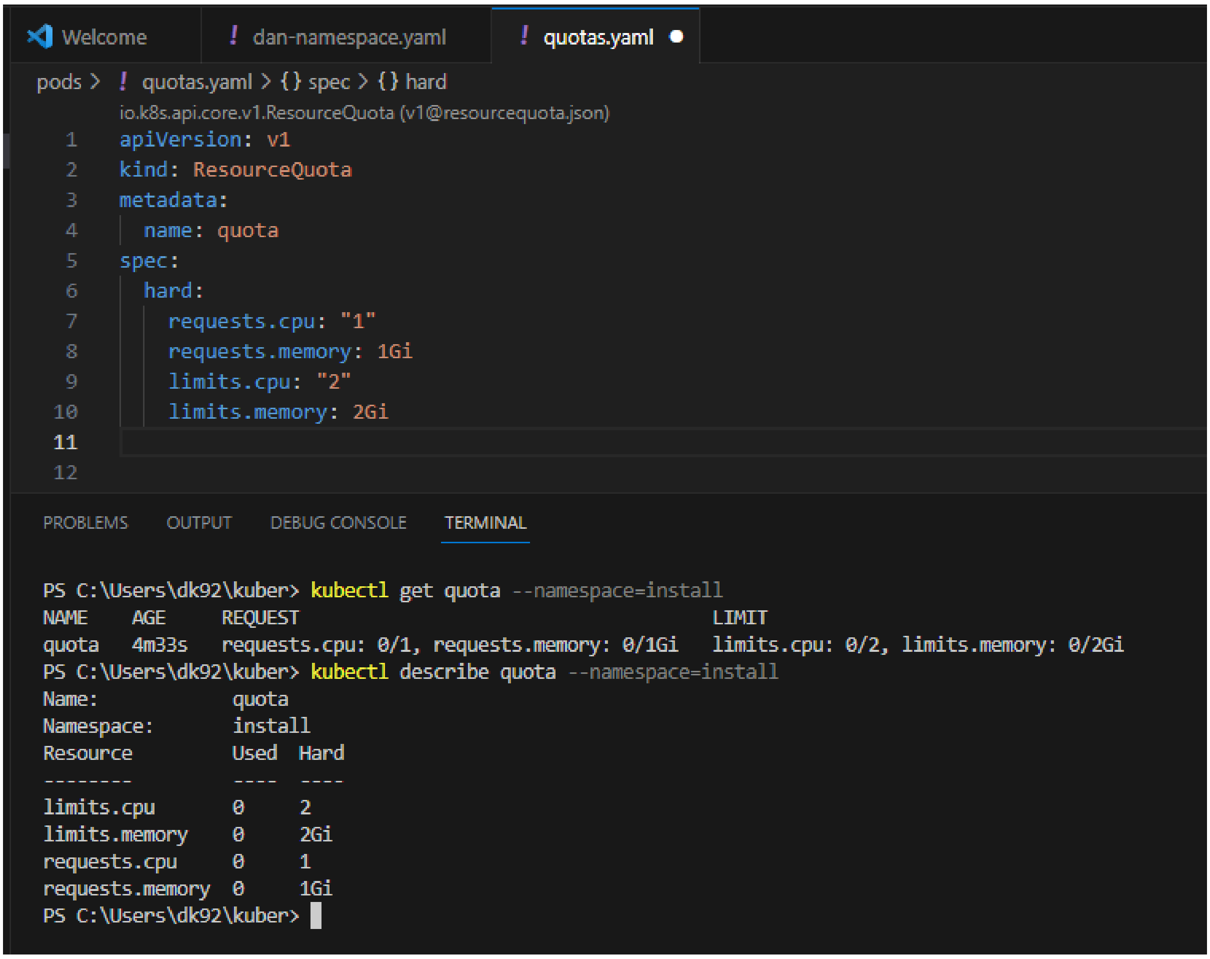


Welcome
! dan-namespace.yaml
! quotas.yaml ●
pods > ! quotas.yaml > {} spec > {} hard
io.k8s.api.core.v1.ResourceQuota (v1@resourcequota.json)
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL

```
apiVersion: v1
kind: ResourceQuota
metadata:
  name: quota
spec:
  hard:
    requests.cpu: "1"
    requests.memory: 1Gi
    limits.cpu: "2"
    limits.memory: 2Gi

```

```
PS C:\Users\dk92\kuber> kubectl apply -f .\pods\quotas.yaml --namespace=install
resourcequota/quota created
PS C:\Users\dk92\kuber> kubectl get namespaces
NAME              STATUS   AGE
dani-namespace    Active   14m
default           Active   3d5h
install           Active   10m
kube-node-lease   Active   3d5h
kube-public       Active   3d5h
kube-system       Active   3d5h
mynamespace       Active   34m
quota             Active   10m
PS C:\Users\dk92\kuber>
PS C:\Users\dk92\kuber> 
```

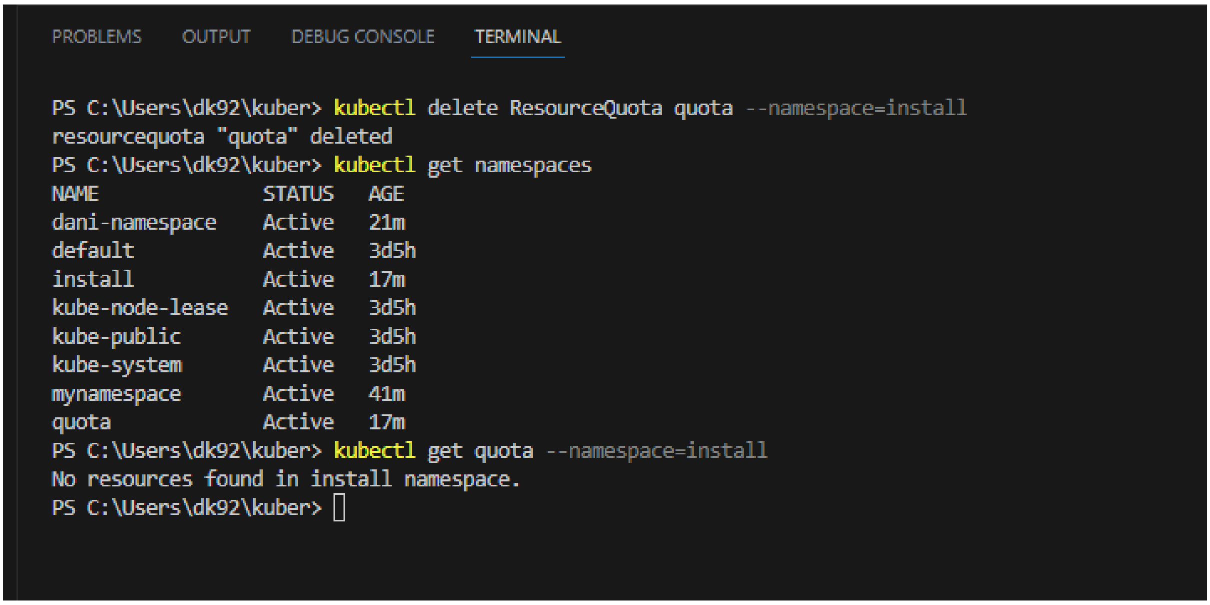
- Command to Configure quota
  kubectl describe quota.


Welcome
dan-namespace.yaml
quotas.yaml
pods > quotas.yaml > {} spec > {} hard
io.k8s.api.core.v1.ResourceQuota (v1@resourcequota.json)
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL

```yaml
apiVersion: v1
kind: ResourceQuota
metadata:
  name: quota
spec:
  hard:
    requests.cpu: "1"
    requests.memory: 1Gi
    limits.cpu: "2"
    limits.memory: 2Gi

```

```
PS C:\Users\dk92\kuber> kubectl get quota --namespace=install
NAME    AGE     REQUEST                                     LIMIT
quota   4m33s   requests.cpu: 0/1, requests.memory: 0/1Gi   limits.cpu: 0/2, limits.memory: 0/2Gi
PS C:\Users\dk92\kuber> kubectl describe quota --namespace=install
Name:            quota
Namespace:       install
Resource         Used  Hard
--------         ----  ----
limits.cpu       0     2
limits.memory    0     2Gi
requests.cpu     0     1
requests.memory  0     1Gi
PS C:\Users\dk92\kuber>
```

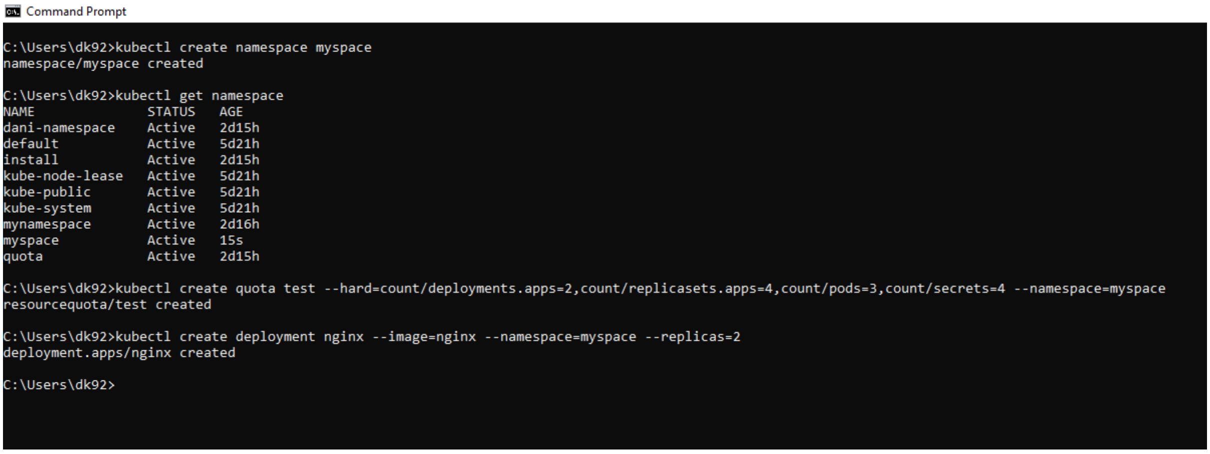
- After deleting quota namespace it will be deleted permanently.



```
PS C:\Users\dk92\kuber> kubectl delete ResourceQuota quota --namespace=install
resourcequota "quota" deleted
PS C:\Users\dk92\kuber> kubectl get namespaces
NAME              STATUS   AGE
dani-namespace    Active   21m
default           Active   3d5h
install           Active   17m
kube-node-lease   Active   3d5h
kube-public       Active   3d5h
kube-system       Active   3d5h
mynamespace       Active   41m
quota             Active   17m
PS C:\Users\dk92\kuber> kubectl get quota --namespace=install
No resources found in install namespace.
PS C:\Users\dk92\kuber>
```

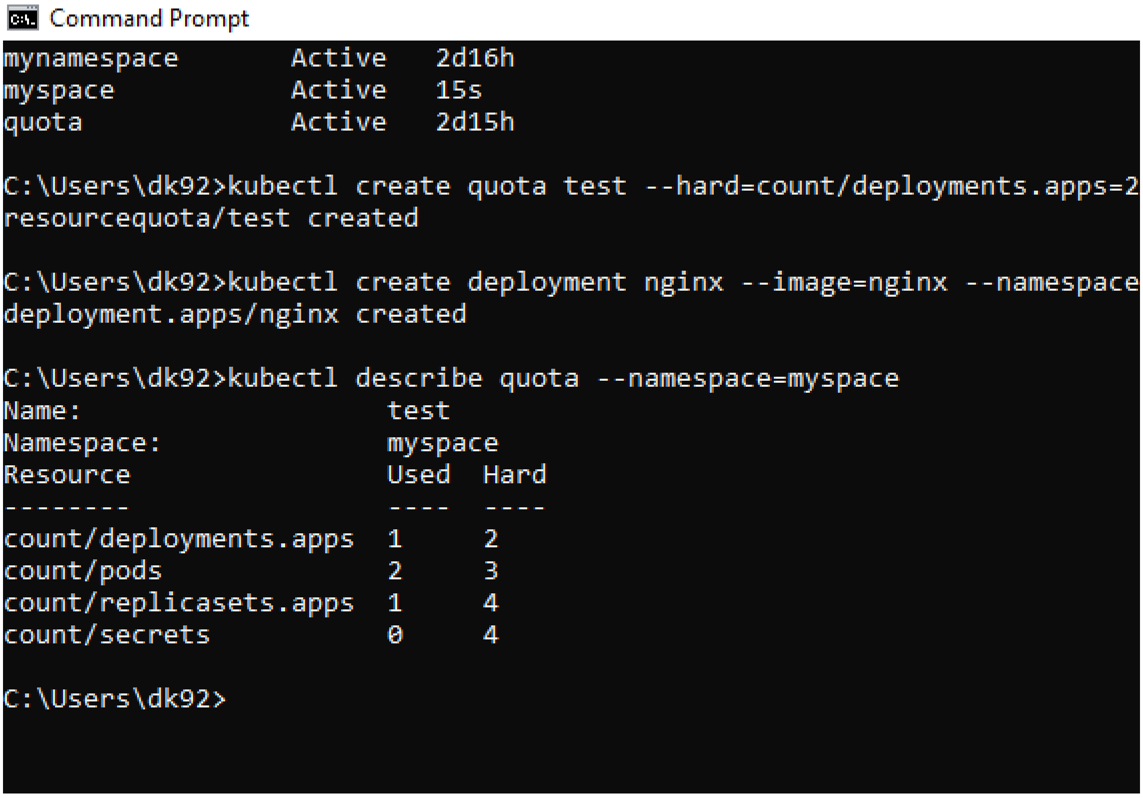
3. Check by creating multiple pods and observe the resource quotas forbidden message.

- Creating multiple pods using deployment with method imperative.



```
C:\Users\dk92>kubectl create namespace myspace
namespace/myspace created

C:\Users\dk92>kubectl get namespace
NAME              STATUS   AGE
dani-namespace    Active   2d15h
default           Active   5d21h
install           Active   2d15h
kube-node-lease   Active   5d21h
kube-public       Active   5d21h
kube-system       Active   5d21h
mynamespace       Active   2d16h
myspace           Active   15s
quota             Active   2d15h

C:\Users\dk92>kubectl create quota test --hard=count/deployments.apps=2,count/replicasets.apps=4,count/pods=3,count/secrets=4 --namespace=myspace
resourcequota/test created

C:\Users\dk92>kubectl create deployment nginx --image=nginx --namespace=myspace --replicas=2
deployment.apps/nginx created

C:\Users\dk92>
```

- Command for checking quota details
  Kubectl describe quota --namespace=myspace



```
mynamespace       Active   2d16h
myspace           Active   15s
quota             Active   2d15h

C:\Users\dk92>kubectl create quota test --hard=count/deployments.apps=2
resourcequota/test created

C:\Users\dk92>kubectl create deployment nginx --image=nginx --namespace
deployment.apps/nginx created

C:\Users\dk92>kubectl describe quota --namespace=myspace
Name:                   test
Namespace:              myspace
Resource                Used  Hard
--------                ----  ----
count/deployments.apps  1     2
count/pods              2     3
count/replicasets.apps  1     4
count/secrets           0     4

C:\Users\dk92>
```

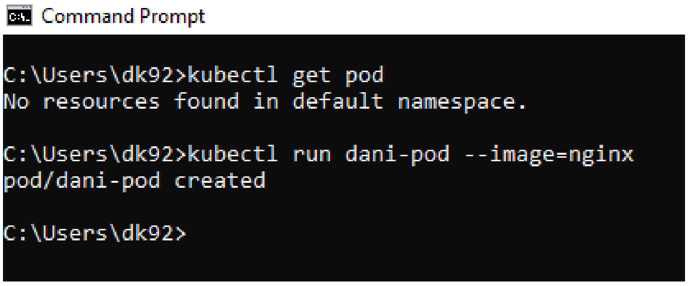
Assignment 2 - Working with Pods.txt

Imperative Method

1. Create pod (nginx image) using imperative method

- Command to create pod as imperative method

  kubectl run dani-pod - -image=nginx.



```
C:\Users\dk92>kubectl get pod
No resources found in default namespace.

C:\Users\dk92>kubectl run dani-pod --image=nginx
pod/dani-pod created

C:\Users\dk92>
```

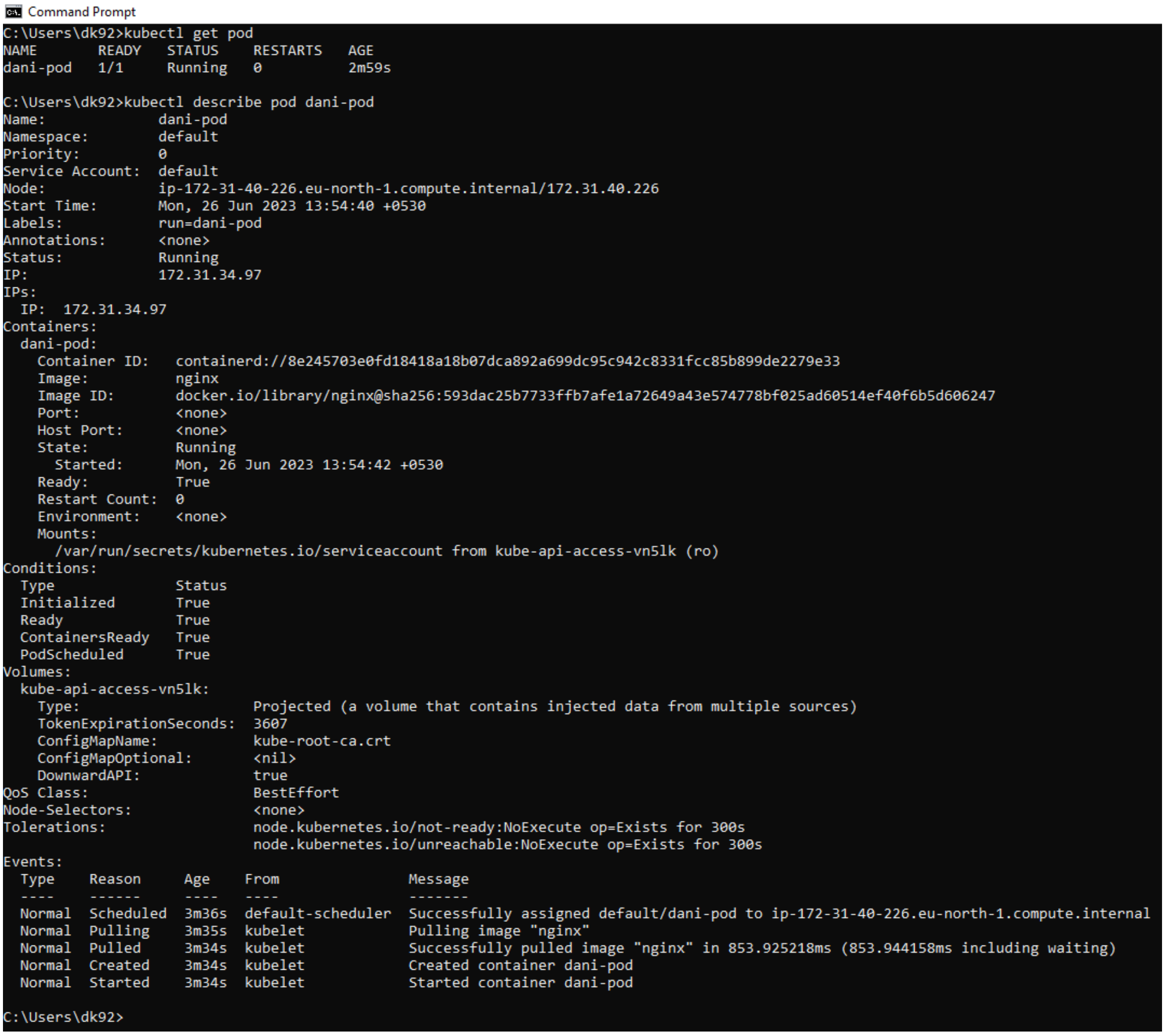
2. Check if the pod is created and describe the pod using kubectl commands

- Command to view pods

  kubectl get pod

- Command to describe the details of pod

  kubectl describe pod dani-pod

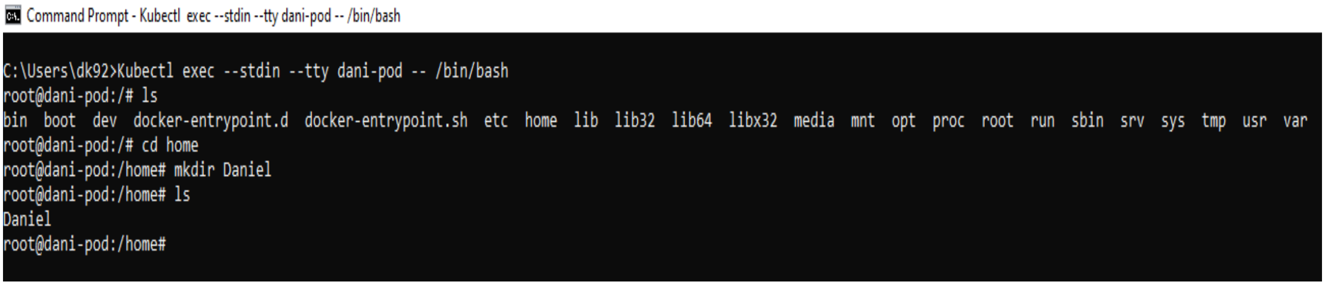


```
C:\Users\dk92>kubectl get pod
NAME       READY   STATUS    RESTARTS   AGE
dani-pod   1/1     Running   0          2m59s

C:\Users\dk92>kubectl describe pod dani-pod
Name:             dani-pod
Namespace:        default
Priority:         0
Service Account:  default
Node:             ip-172-31-40-226.eu-north-1.compute.internal/172.31.40.226
Start Time:       Mon, 26 Jun 2023 13:54:40 +0530
Labels:           run=dani-pod
Annotations:      <none>
Status:           Running
IP:               172.31.34.97
IPs:
  IP:  172.31.34.97
Containers:
  dani-pod:
    Container ID:   containerd://8e245703e0fd18418a18b07dca892a699dc95c942c8331fcc85b899de2279e33
    Image:          nginx
    Image ID:       docker.io/library/nginx@sha256:593dac25b7733ffb7afe1a72649a43e574778bf025ad60514ef40f6b5d606247
    Port:           <none>
    Host Port:      <none>
    State:          Running
      Started:      Mon, 26 Jun 2023 13:54:42 +0530
    Ready:          True
    Restart Count:  0
    Environment:    <none>
    Mounts:
      /var/run/secrets/kubernetes.io/serviceaccount from kube-api-access-vn5lk (ro)
Conditions:
  Type              Status
  Initialized       True
  Ready             True
  ContainersReady   True
  PodScheduled      True
Volumes:
  kube-api-access-vn5lk:
    Type:                    Projected (a volume that contains injected data from multiple sources)
    TokenExpirationSeconds:  3607
    ConfigMapName:           kube-root-ca.crt
    ConfigMapOptional:       <nil>
    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  3m36s  default-scheduler  Successfully assigned default/dani-pod to ip-172-31-40-226.eu-north-1.compute.internal
  Normal  Pulling    3m35s  kubelet            Pulling image "nginx"
  Normal  Pulled     3m34s  kubelet            Successfully pulled image "nginx" in 853.925218ms (853.944158ms including waiting)
  Normal  Created    3m34s  kubelet            Created container dani-pod
  Normal  Started    3m34s  kubelet            Started container dani-pod

C:\Users\dk92>
```

3. Login to the pod using kubectl commands

- Command to login into pod

  Kubectl exec - -stdin - -tty dani-pod - - /bin/bash
- We are creating one folder name as Daniel.

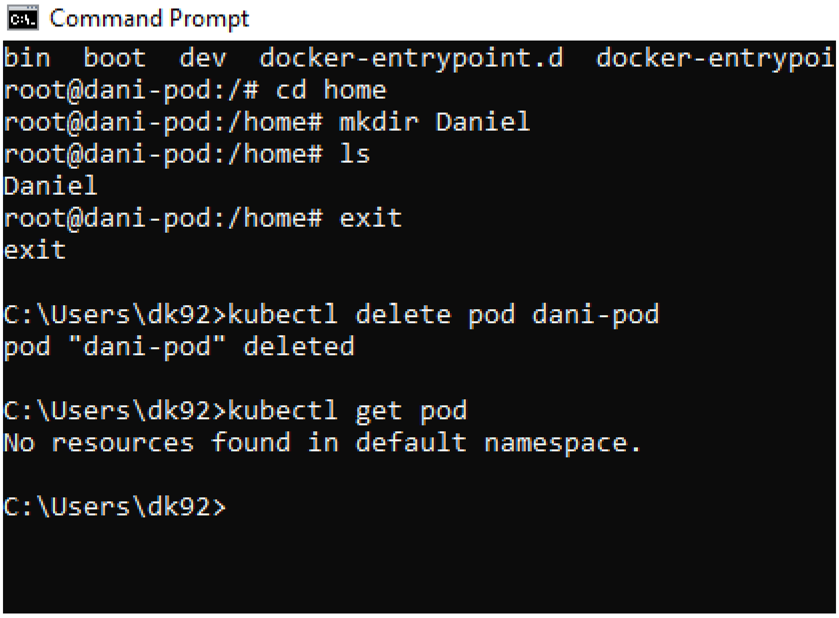

Command Prompt - Kubectl exec --stdin --tty dani-pod -- /bin/bash

```
C:\Users\dk92>Kubectl exec --stdin --tty dani-pod -- /bin/bash
root@dani-pod:/# ls
bin  boot  dev  docker-entrypoint.d  docker-entrypoint.sh  etc  home  lib  lib32  lib64  libx32  media  mnt  opt  proc  root  run  sbin  srv  sys  tmp  usr  var
root@dani-pod:/# cd home
root@dani-pod:/home# mkdir Daniel
root@dani-pod:/home# ls
Daniel
root@dani-pod:/home#
```

4. Delete the pod using kubectl commands

- Use command to delete the pod

  kubectl delete pod dani-pod
- After deleting pod, it is deleted permanently



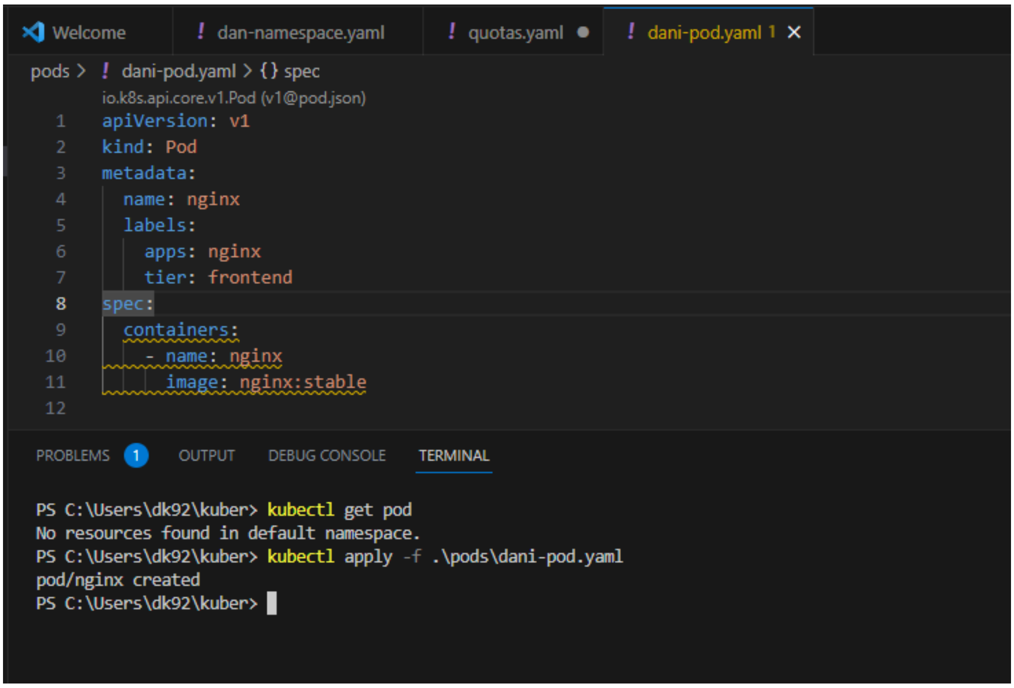
```
bin  boot  dev  docker-entrypoint.d  docker-entrypoi
root@dani-pod:/# cd home
root@dani-pod:/home# mkdir Daniel
root@dani-pod:/home# ls
Daniel
root@dani-pod:/home# exit
exit

C:\Users\dk92>kubectl delete pod dani-pod
pod "dani-pod" deleted

C:\Users\dk92>kubectl get pod
No resources found in default namespace.

C:\Users\dk92>
```

Declarative Method

1. Create pod (nginx image) using declarative method

- Create yaml file which will contain image of nginx.
- Command to create pod as declarative method

  kubectl apply –f “file name/file path”.



```yaml
apiVersion: v1
kind: Pod
metadata:
  name: nginx
  labels:
    apps: nginx
    tier: frontend
spec:
  containers:
    - name: nginx
      image: nginx:stable
```

```
PS C:\Users\dk92\kuber> kubectl get pod
No resources found in default namespace.
PS C:\Users\dk92\kuber> kubectl apply -f .\pods\dani-pod.yaml
pod/nginx created
PS C:\Users\dk92\kuber>
```

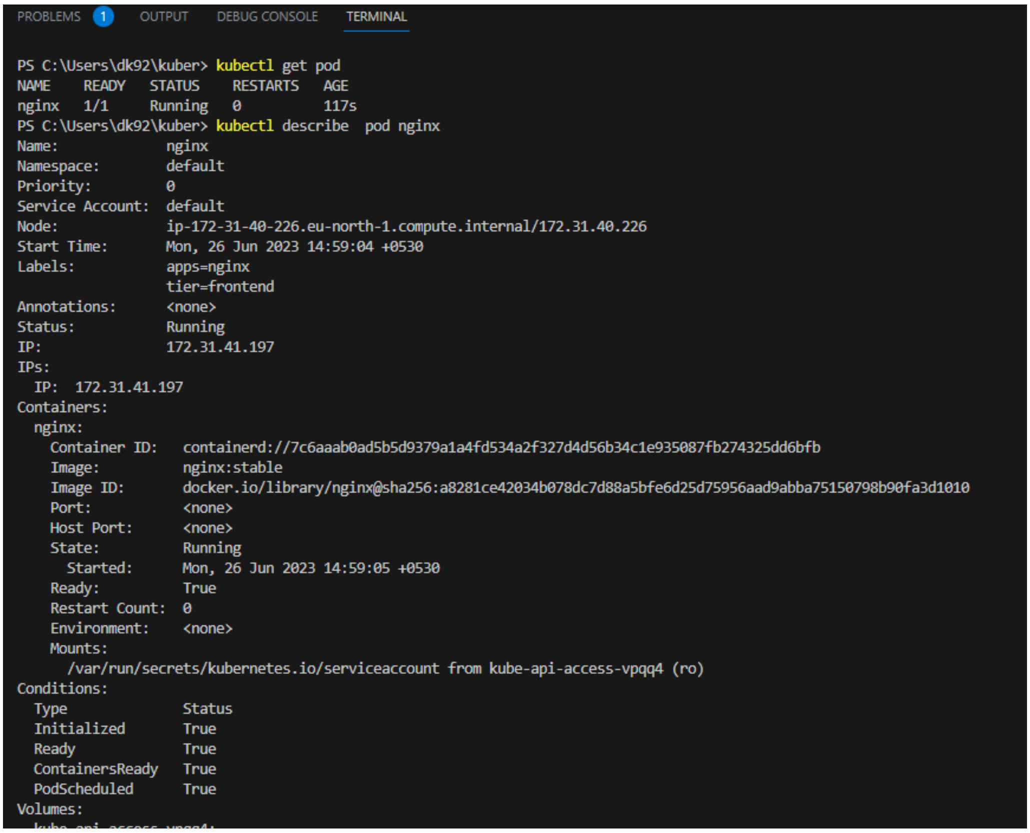
2. Check if the pod is created and describe the pod using kubectl commands

- Command to view pods

  kubectl get pod

- Command to describe the details of pod

  kubectl describe pod nginx



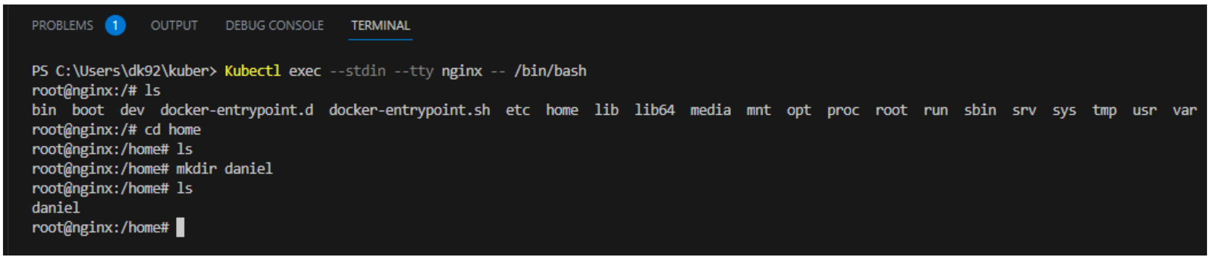
```
PS C:\Users\dk92\kuber> kubectl get pod
NAME    READY   STATUS    RESTARTS   AGE
nginx   1/1     Running   0          117s
PS C:\Users\dk92\kuber> kubectl describe  pod nginx
Name:             nginx
Namespace:        default
Priority:         0
Service Account:  default
Node:             ip-172-31-40-226.eu-north-1.compute.internal/172.31.40.226
Start Time:       Mon, 26 Jun 2023 14:59:04 +0530
Labels:           apps=nginx
                  tier=frontend
Annotations:      <none>
Status:           Running
IP:               172.31.41.197
IPs:
  IP:  172.31.41.197
Containers:
  nginx:
    Container ID:   containerd://7c6aaab0ad5b5d9379a1a4fd534a2f327d4d56b34c1e935087fb274325dd6bfb
    Image:          nginx:stable
    Image ID:       docker.io/library/nginx@sha256:a8281ce42034b078dc7d88a5bfe6d25d75956aad9abba75150798b90fa3d1010
    Port:           <none>
    Host Port:      <none>
    State:          Running
      Started:      Mon, 26 Jun 2023 14:59:05 +0530
    Ready:          True
    Restart Count:  0
    Environment:    <none>
    Mounts:
      /var/run/secrets/kubernetes.io/serviceaccount from kube-api-access-vpqq4 (ro)
Conditions:
  Type              Status
  Initialized       True
  Ready             True
  ContainersReady   True
  PodScheduled      True
Volumes:
  kube-api-access-vpqq4:
```

3. Login to the pod using kubectl commands

- Command to login into pod

  Kubectl exec - -stdin - -tty nginx - - /bin/bash

- We are creating one folder name as Daniel



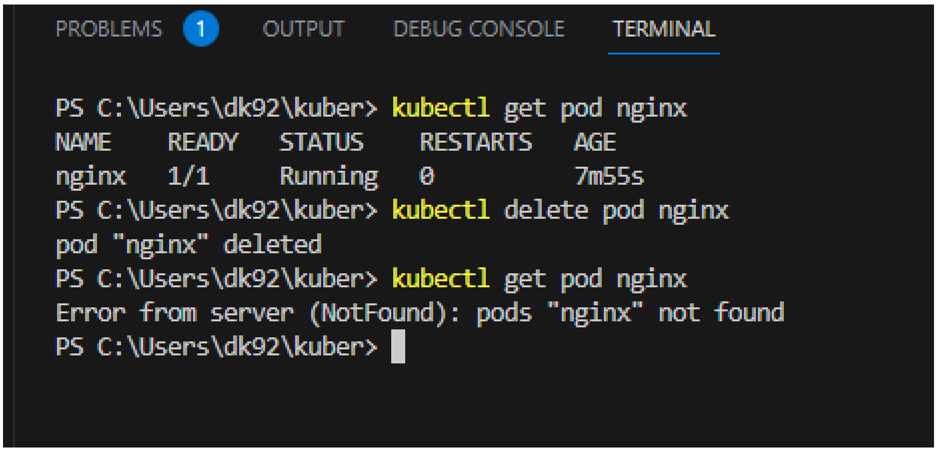
```
PS C:\Users\dk92\kuber> Kubectl exec --stdin --tty nginx -- /bin/bash
root@nginx:/# ls
bin  boot  dev  docker-entrypoint.d  docker-entrypoint.sh  etc  home  lib  lib64  media  mnt  opt  proc  root  run  sbin  srv  sys  tmp  usr  var
root@nginx:/# cd home
root@nginx:/home# ls
root@nginx:/home# mkdir daniel
root@nginx:/home# ls
daniel
root@nginx:/home# 
```

4. Delete the pod using kubectl commands

- Use command to delete the pod

  kubectl delete pod nginx

- After deleting pod, it is deleted permanently

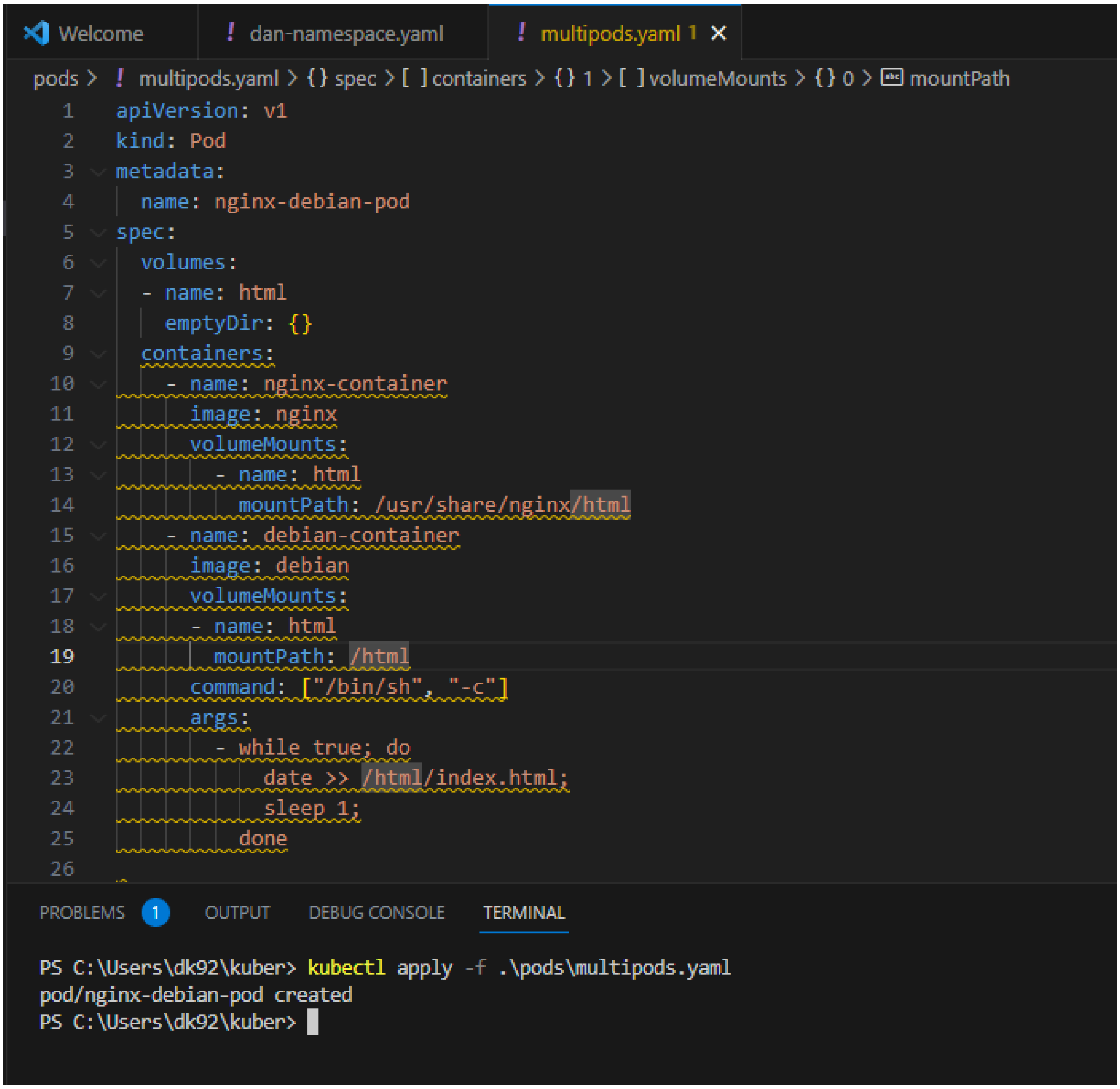


```
PS C:\Users\dk92\kuber> kubectl get pod nginx
NAME    READY   STATUS    RESTARTS   AGE
nginx   1/1     Running   0          7m55s
PS C:\Users\dk92\kuber> kubectl delete pod nginx
pod "nginx" deleted
PS C:\Users\dk92\kuber> kubectl get pod nginx
Error from server (NotFound): pods "nginx" not found
PS C:\Users\dk92\kuber> 
```

Assignment 3 - Working with Multipods with shared volume.txt

1. Create pod (nginx image with 2 containers (1st container - nginx image , 2nd container - debian image) , common volume (emptydir) mounted on both the volumes using declarative method

- Write a YAML code for creating pod with 2 containers one is nginx image and second is debain image.

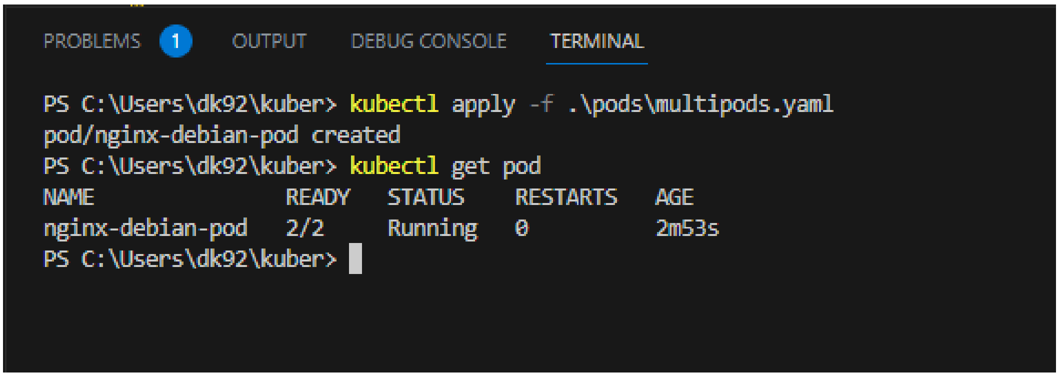

Welcome
! dan-namespace.yaml
! multipods.yaml 1 ×
pods > ! multipods.yaml > {} spec > [ ] containers > {} 1 > [ ] volumeMounts > {} 0 > mountPath
PROBLEMS 1
OUTPUT
DEBUG CONSOLE
TERMINAL

```yaml
apiVersion: v1
kind: Pod
metadata:
  name: nginx-debian-pod
spec:
  volumes:
  - name: html
    emptyDir: {}
  containers:
    - name: nginx-container
      image: nginx
      volumeMounts:
        - name: html
          mountPath: /usr/share/nginx/html
    - name: debian-container
      image: debian
      volumeMounts:
      - name: html
        mountPath: /html
      command: ["/bin/sh", "-c"]
      args:
        - while true; do
            date >> /html/index.html;
            sleep 1;
          done

```

```
PS C:\Users\dk92\kuber> kubectl apply -f .\pods\multipods.yaml
pod/nginx-debian-pod created
PS C:\Users\dk92\kuber>
```

2. Check if the pod has got created with 2 containers

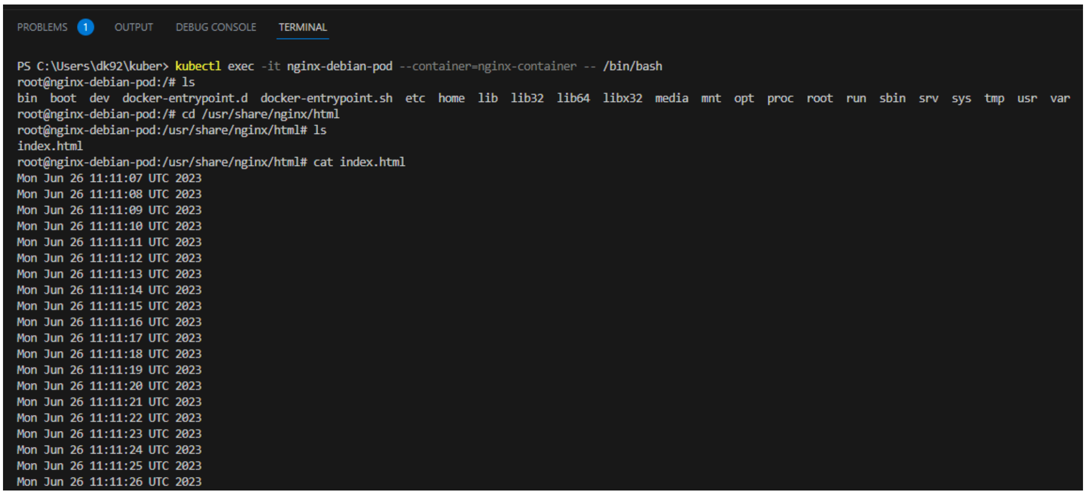
- Command to view the pod
  Kubectl get pods



```
PS C:\Users\dk92\kuber> kubectl apply -f .\pods\multipods.yaml
pod/nginx-debian-pod created
PS C:\Users\dk92\kuber> kubectl get pod
NAME               READY   STATUS    RESTARTS   AGE
nginx-debian-pod   2/2     Running   0          2m53s
PS C:\Users\dk92\kuber> 
```

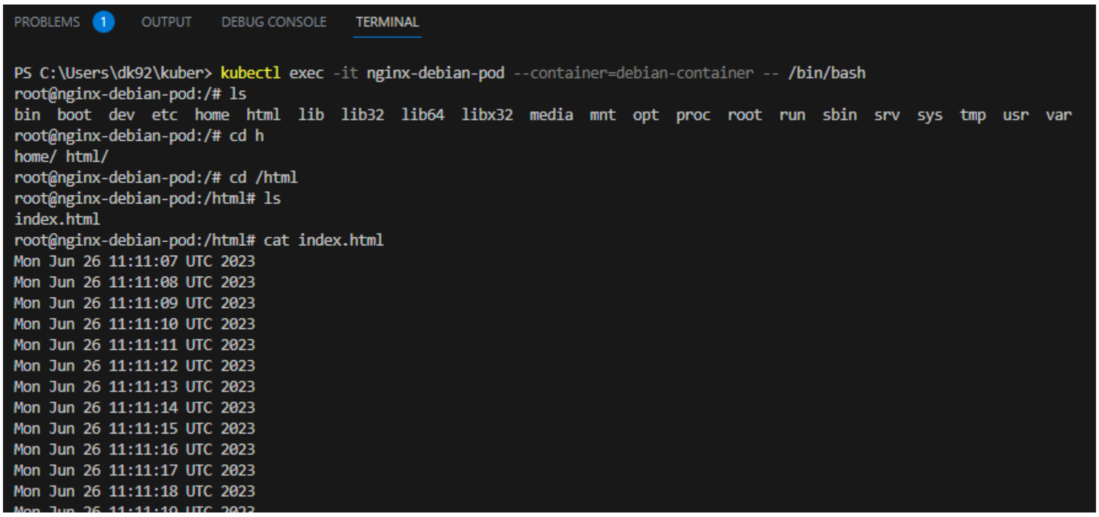
3. Check if the containers are able to access the shared volume and data mapping is working

- First log into nginx container
- Check the given data is mapping (date and time will updates every second) in index.html
- Check data using command **cat index.html**



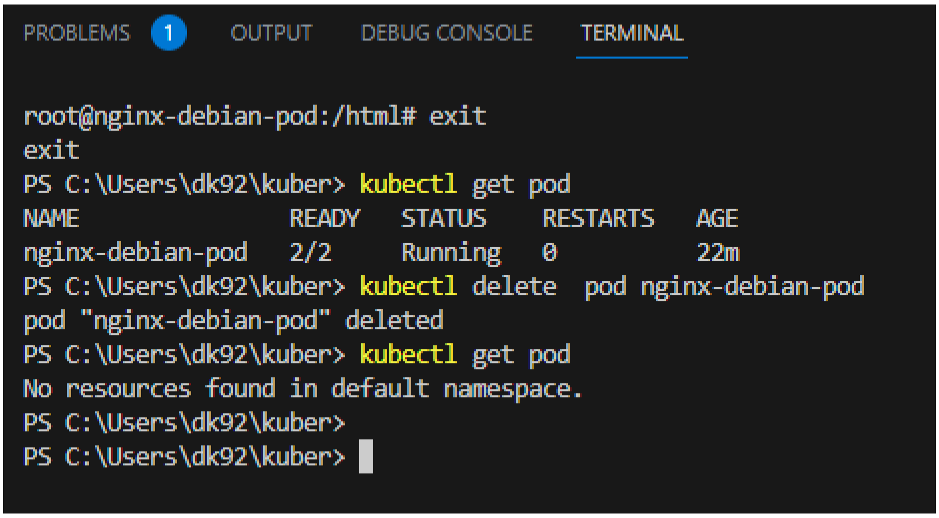
```
PS C:\Users\dk92\kuber> kubectl exec -it nginx-debian-pod --container=nginx-container -- /bin/bash
root@nginx-debian-pod:/# ls
bin  boot  dev  docker-entrypoint.d  docker-entrypoint.sh  etc  home  lib  lib32  lib64  libx32  media  mnt  opt  proc  root  run  sbin  srv  sys  tmp  usr  var
root@nginx-debian-pod:/# cd /usr/share/nginx/html
root@nginx-debian-pod:/usr/share/nginx/html# ls
index.html
root@nginx-debian-pod:/usr/share/nginx/html# cat index.html
Mon Jun 26 11:11:07 UTC 2023
Mon Jun 26 11:11:08 UTC 2023
Mon Jun 26 11:11:09 UTC 2023
Mon Jun 26 11:11:10 UTC 2023
Mon Jun 26 11:11:11 UTC 2023
Mon Jun 26 11:11:12 UTC 2023
Mon Jun 26 11:11:13 UTC 2023
Mon Jun 26 11:11:14 UTC 2023
Mon Jun 26 11:11:15 UTC 2023
Mon Jun 26 11:11:16 UTC 2023
Mon Jun 26 11:11:17 UTC 2023
Mon Jun 26 11:11:18 UTC 2023
Mon Jun 26 11:11:19 UTC 2023
Mon Jun 26 11:11:20 UTC 2023
Mon Jun 26 11:11:21 UTC 2023
Mon Jun 26 11:11:22 UTC 2023
Mon Jun 26 11:11:23 UTC 2023
Mon Jun 26 11:11:24 UTC 2023
Mon Jun 26 11:11:25 UTC 2023
Mon Jun 26 11:11:26 UTC 2023
```

- Next log into debian container
- Check the given data is mapping (date and time will updates every seconds) in index.html
- Check data using command **cat index.html**



```
PS C:\Users\dk92\kuber> kubectl exec -it nginx-debian-pod --container=debian-container -- /bin/bash
root@nginx-debian-pod:/# ls
bin  boot  dev  etc  home  html  lib  lib32  lib64  libx32  media  mnt  opt  proc  root  run  sbin  srv  sys  tmp  usr  var
root@nginx-debian-pod:/# cd h
home/ html/
root@nginx-debian-pod:/# cd /html
root@nginx-debian-pod:/html# ls
index.html
root@nginx-debian-pod:/html# cat index.html
Mon Jun 26 11:11:07 UTC 2023
Mon Jun 26 11:11:08 UTC 2023
Mon Jun 26 11:11:09 UTC 2023
Mon Jun 26 11:11:10 UTC 2023
Mon Jun 26 11:11:11 UTC 2023
Mon Jun 26 11:11:12 UTC 2023
Mon Jun 26 11:11:13 UTC 2023
Mon Jun 26 11:11:14 UTC 2023
Mon Jun 26 11:11:15 UTC 2023
Mon Jun 26 11:11:16 UTC 2023
Mon Jun 26 11:11:17 UTC 2023
Mon Jun 26 11:11:18 UTC 2023
Mon Jun 26 11:11:19 UTC 2023
```

4. Delete the pod and observe if the data is still available under empty dir folder in worker node
    - After deleting the pod, it is deleted permanently

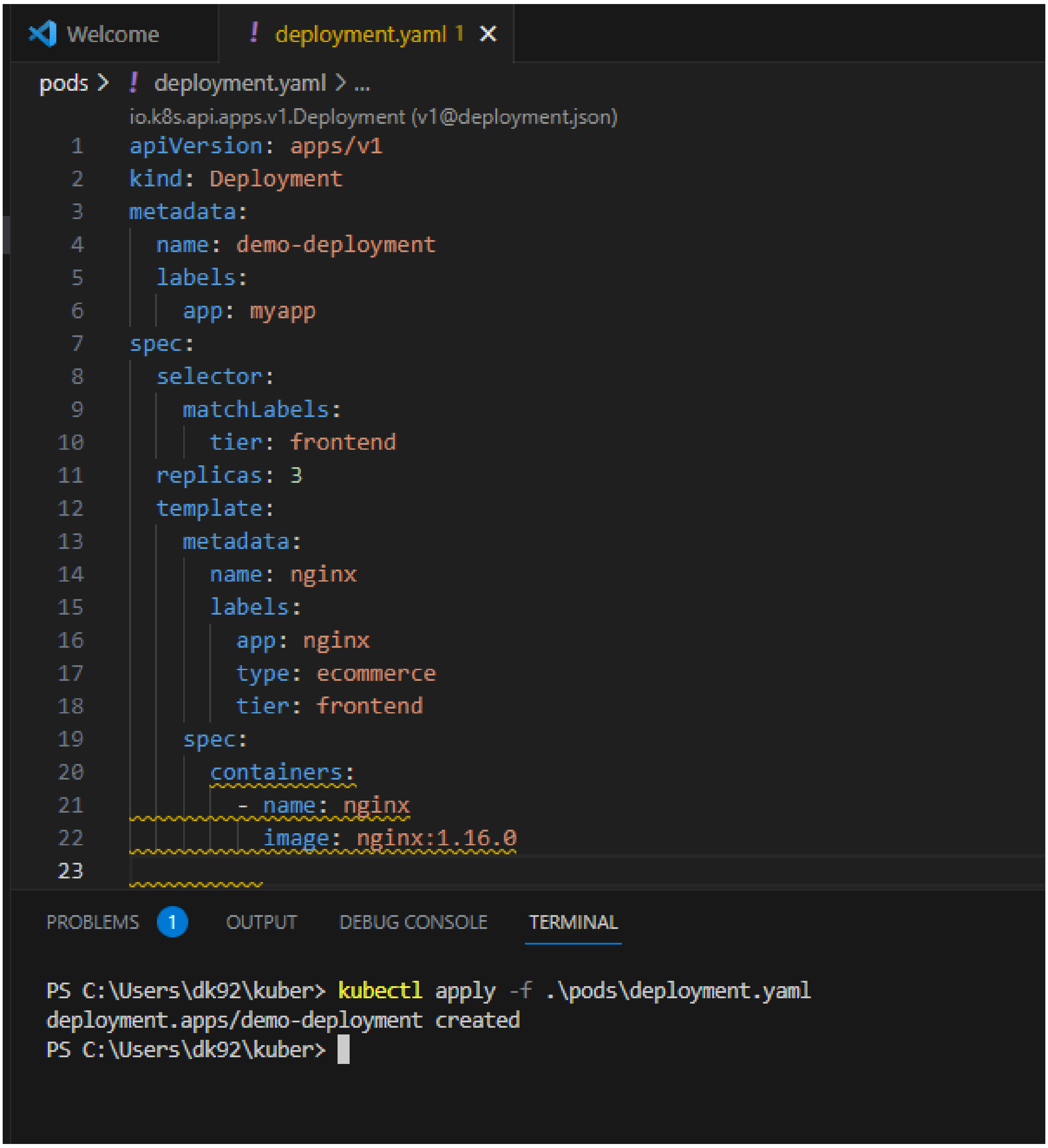


```
root@nginx-debian-pod:/html# exit
exit
PS C:\Users\dk92\kuber> kubectl get pod
NAME               READY   STATUS    RESTARTS   AGE
nginx-debian-pod   2/2     Running   0          22m
PS C:\Users\dk92\kuber> kubectl delete  pod nginx-debian-pod
pod "nginx-debian-pod" deleted
PS C:\Users\dk92\kuber> kubectl get pod
No resources found in default namespace.
PS C:\Users\dk92\kuber>
PS C:\Users\dk92\kuber>
```

Assignment 4 - Working with Deployments.txt

1. Create 3 copies of nginx Pods image with 1container (nginx image) using declarative method

- Creating a deployment yaml file using declarative method and add replicas and image



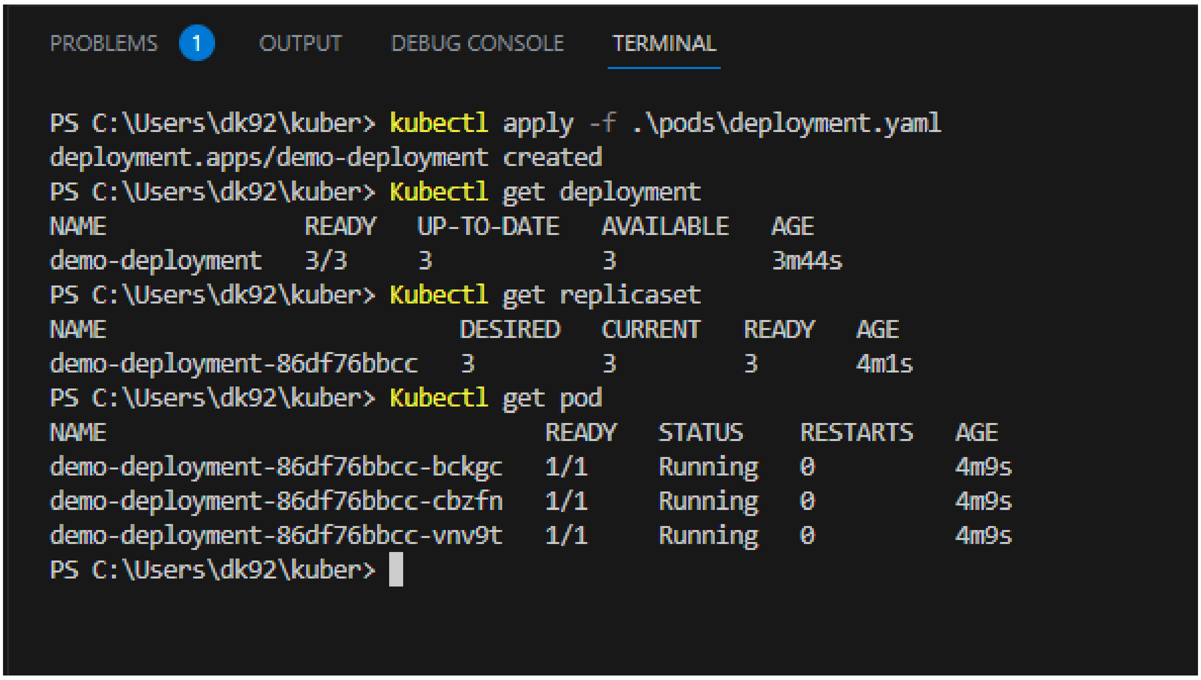
Welcome
! deployment.yaml 1 ×
pods > ! deployment.yaml > ...
io.k8s.api.apps.v1.Deployment (v1@deployment.json)
PROBLEMS 1
OUTPUT
DEBUG CONSOLE
TERMINAL

```yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: demo-deployment
  labels:
    app: myapp
spec:
  selector:
    matchLabels:
      tier: frontend
  replicas: 3
  template:
    metadata:
      name: nginx
      labels:
        app: nginx
        type: ecommerce
        tier: frontend
    spec:
      containers:
        - name: nginx
          image: nginx:1.16.0

```

```
PS C:\Users\dk92\kuber> kubectl apply -f .\pods\deployment.yaml
deployment.apps/demo-deployment created
PS C:\Users\dk92\kuber>
```

2. Check if the deployment, replicaset and pods have been created using kubectl commands

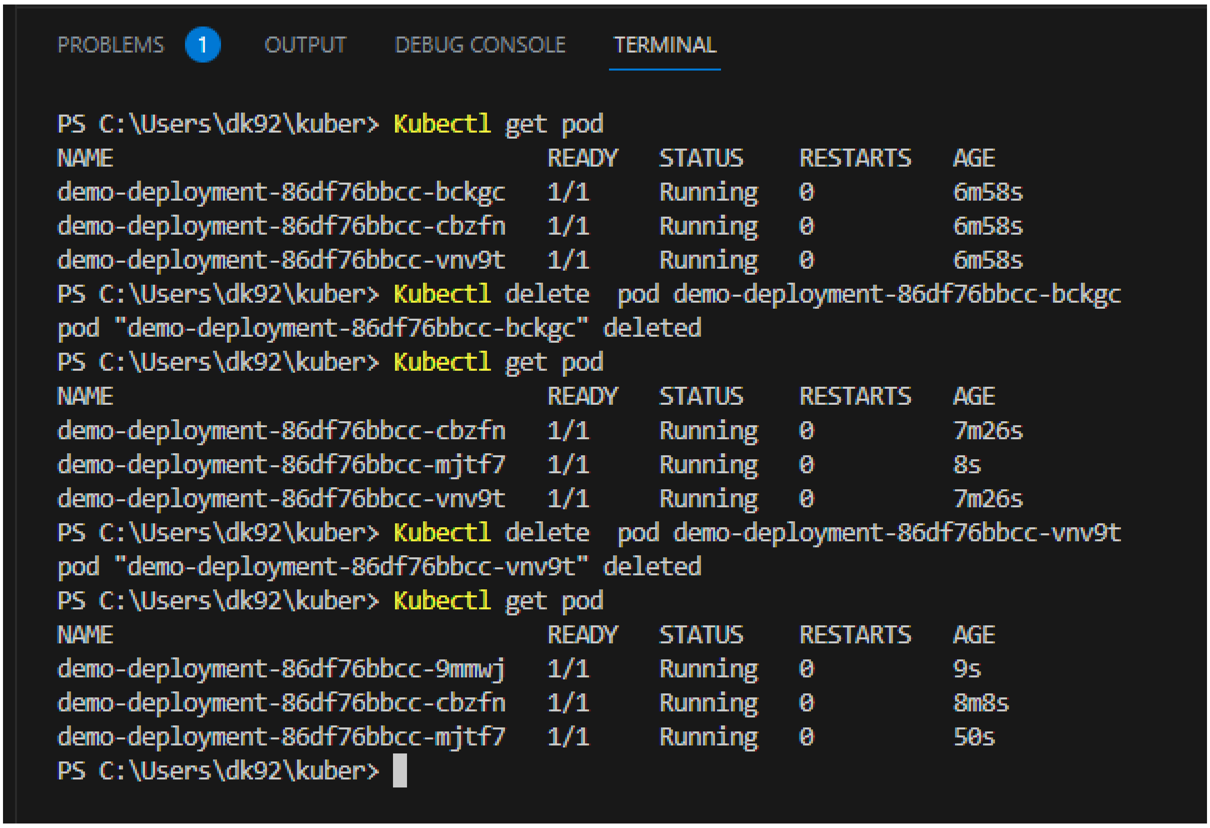
- Command to view Deployment
  Kubectl get deployment
- Command to view Replicaset
  Kubectl get replicaset
- Command to view pod
  Kubectl get pod



```
PS C:\Users\dk92\kuber> kubectl apply -f .\pods\deployment.yaml
deployment.apps/demo-deployment created
PS C:\Users\dk92\kuber> Kubectl get deployment
NAME              READY   UP-TO-DATE   AVAILABLE   AGE
demo-deployment   3/3     3            3           3m44s
PS C:\Users\dk92\kuber> Kubectl get replicaset
NAME                         DESIRED   CURRENT   READY   AGE
demo-deployment-86df76bbcc   3         3         3       4m1s
PS C:\Users\dk92\kuber> Kubectl get pod
NAME                               READY   STATUS    RESTARTS   AGE
demo-deployment-86df76bbcc-bckgc   1/1     Running   0          4m9s
demo-deployment-86df76bbcc-cbzfn   1/1     Running   0          4m9s
demo-deployment-86df76bbcc-vnv9t   1/1     Running   0          4m9s
PS C:\Users\dk92\kuber>
```

3. Delete a pod and observe and how new pod gets created automatically

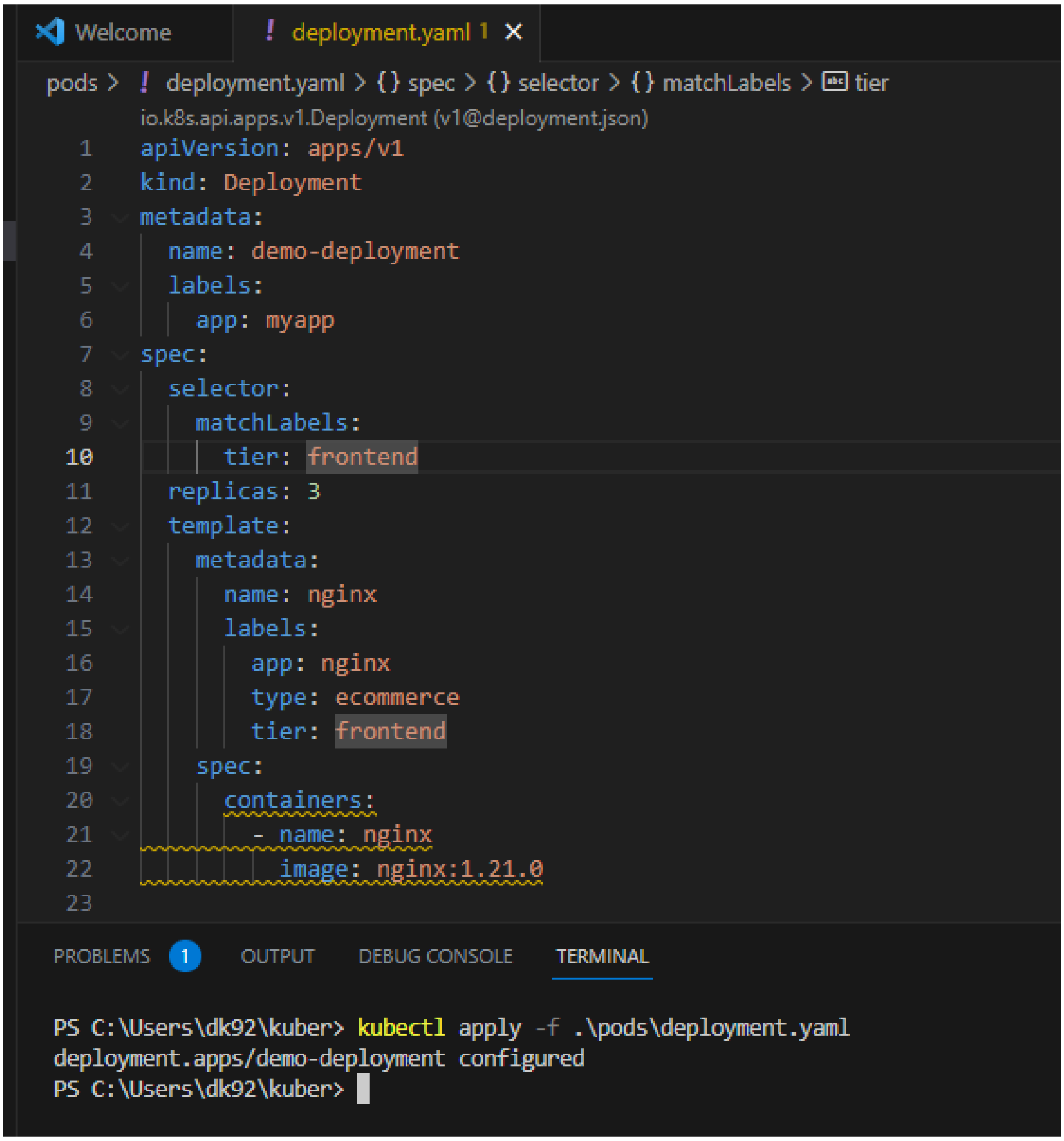
- We deleted one pod after deleting pod it will create a new pod automatically.



```
PS C:\Users\dk92\kuber> Kubectl get pod
NAME                               READY   STATUS    RESTARTS   AGE
demo-deployment-86df76bbcc-bckgc   1/1     Running   0          6m58s
demo-deployment-86df76bbcc-cbzfn   1/1     Running   0          6m58s
demo-deployment-86df76bbcc-vnv9t   1/1     Running   0          6m58s
PS C:\Users\dk92\kuber> Kubectl delete  pod demo-deployment-86df76bbcc-bckgc
pod "demo-deployment-86df76bbcc-bckgc" deleted
PS C:\Users\dk92\kuber> Kubectl get pod
NAME                               READY   STATUS    RESTARTS   AGE
demo-deployment-86df76bbcc-cbzfn   1/1     Running   0          7m26s
demo-deployment-86df76bbcc-mjtf7   1/1     Running   0          8s
demo-deployment-86df76bbcc-vnv9t   1/1     Running   0          7m26s
PS C:\Users\dk92\kuber> Kubectl delete  pod demo-deployment-86df76bbcc-vnv9t
pod "demo-deployment-86df76bbcc-vnv9t" deleted
PS C:\Users\dk92\kuber> Kubectl get pod
NAME                               READY   STATUS    RESTARTS   AGE
demo-deployment-86df76bbcc-9mmwj   1/1     Running   0          9s
demo-deployment-86df76bbcc-cbzfn   1/1     Running   0          8m8s
demo-deployment-86df76bbcc-mjtf7   1/1     Running   0          50s
PS C:\Users\dk92\kuber>
```

4. Update the image version of the deployment and record the update

- Changing image version of nginx from 1.16.0 to 1.21.0

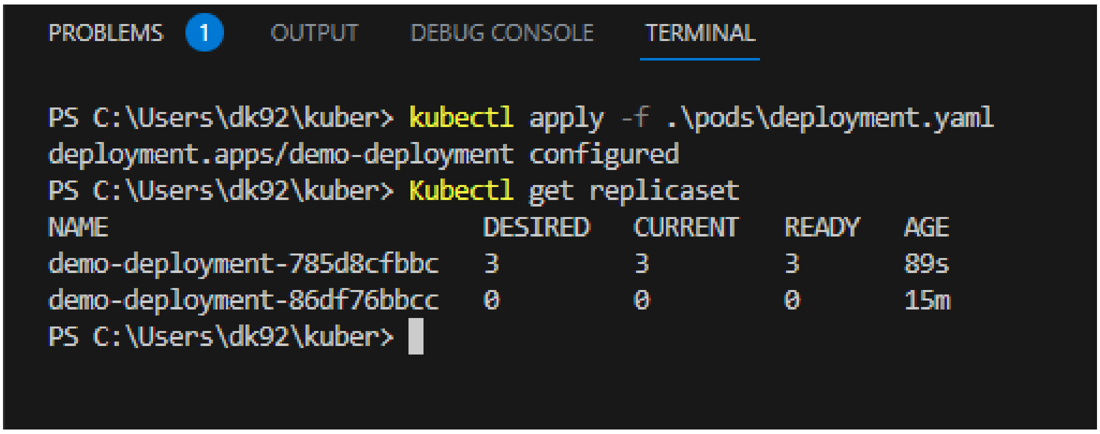


Welcome
deployment.yaml 1
pods > deployment.yaml > {} spec > {} selector > {} matchLabels > tier
io.k8s.api.apps.v1.Deployment (v1@deployment.json)
PROBLEMS 1
OUTPUT
DEBUG CONSOLE
TERMINAL

```yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: demo-deployment
  labels:
    app: myapp
spec:
  selector:
    matchLabels:
      tier: frontend
  replicas: 3
  template:
    metadata:
      name: nginx
      labels:
        app: nginx
        type: ecommerce
        tier: frontend
    spec:
      containers:
      - name: nginx
        image: nginx:1.21.0
```

```
PS C:\Users\dk92\kuber> kubectl apply -f .\pods\deployment.yaml
deployment.apps/demo-deployment configured
PS C:\Users\dk92\kuber>
```

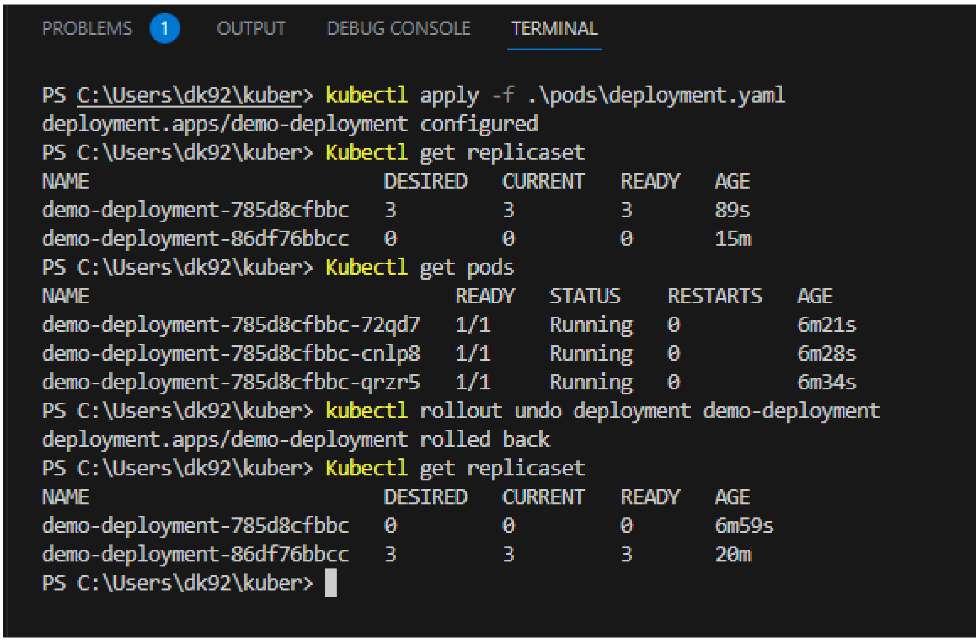
- When we are viewing replicaset it will showing newer version age and older version is removed.



```
PS C:\Users\dk92\kuber> kubectl apply -f .\pods\deployment.yaml
deployment.apps/demo-deployment configured
PS C:\Users\dk92\kuber> Kubectl get replicaset
NAME                         DESIRED   CURRENT   READY   AGE
demo-deployment-785d8cfbbc   3         3         3       89s
demo-deployment-86df76bbcc   0         0         0       15m
PS C:\Users\dk92\kuber>
```

5. Run commands kubectl rollout status / history, kubectl undo etc..

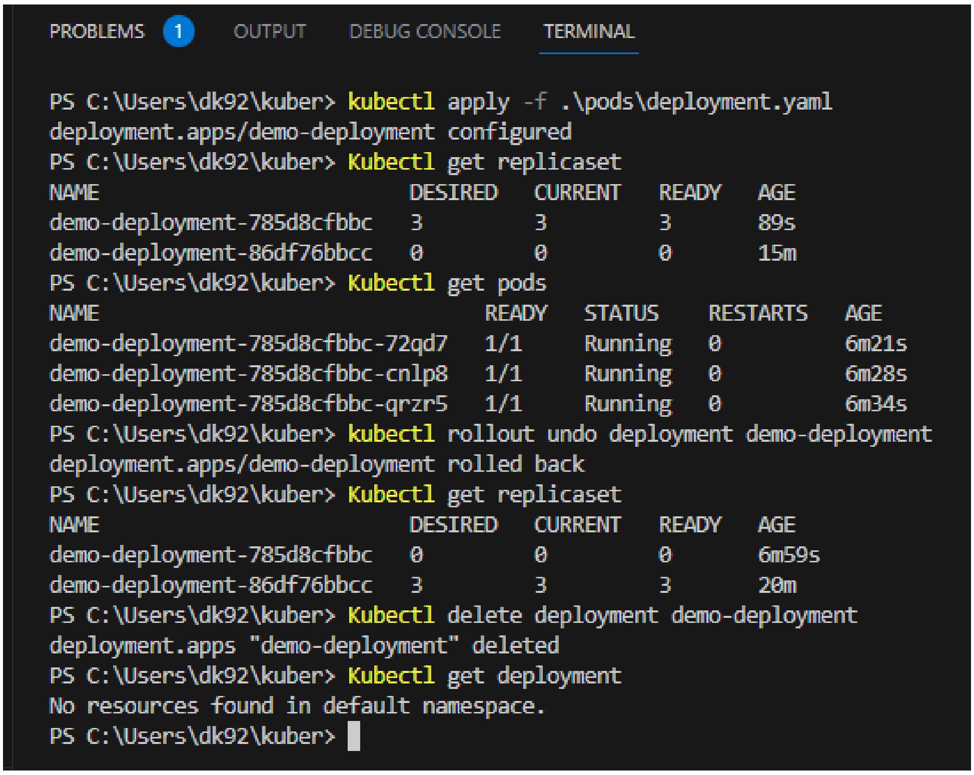
- Using rollout command
- When we rollout the deployment it removes updated version and put it on older version.



```
PS C:\Users\dk92\kuber> kubectl apply -f .\pods\deployment.yaml
deployment.apps/demo-deployment configured
PS C:\Users\dk92\kuber> Kubectl get replicaset
NAME                         DESIRED   CURRENT   READY   AGE
demo-deployment-785d8cfbbc   3         3         3       89s
demo-deployment-86df76bbcc   0         0         0       15m
PS C:\Users\dk92\kuber> Kubectl get pods
NAME                               READY   STATUS    RESTARTS   AGE
demo-deployment-785d8cfbbc-72qd7   1/1     Running   0          6m21s
demo-deployment-785d8cfbbc-cnlp8   1/1     Running   0          6m28s
demo-deployment-785d8cfbbc-qrzr5   1/1     Running   0          6m34s
PS C:\Users\dk92\kuber> kubectl rollout undo deployment demo-deployment
deployment.apps/demo-deployment rolled back
PS C:\Users\dk92\kuber> Kubectl get replicaset
NAME                         DESIRED   CURRENT   READY   AGE
demo-deployment-785d8cfbbc   0         0         0       6m59s
demo-deployment-86df76bbcc   3         3         3       20m
PS C:\Users\dk92\kuber> 
```

6. Delete the deployment and observe what happened

- Deleting the deployment using command
  Kubectl delete deployment demo-deployment.
- After deleting deployment, it will be deleted permanently.

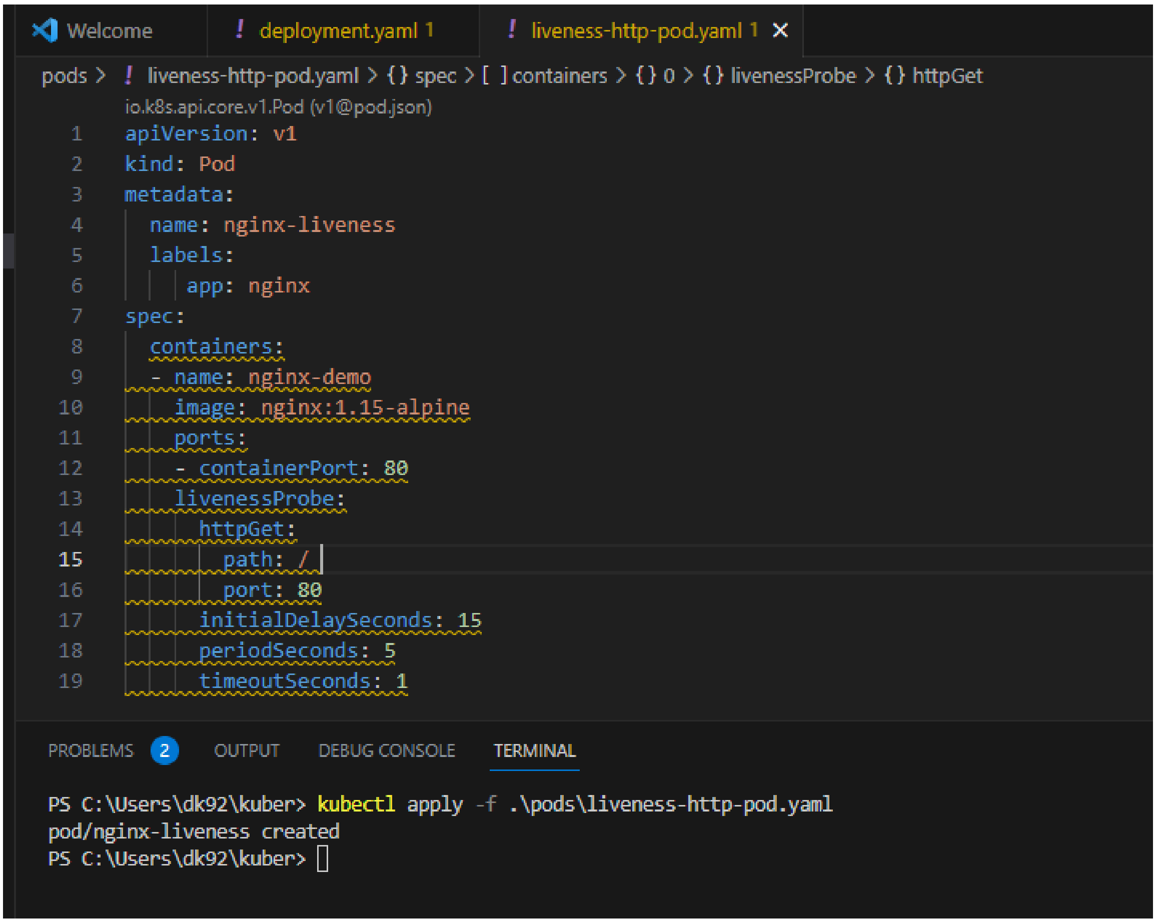


```
PS C:\Users\dk92\kuber> kubectl apply -f .\pods\deployment.yaml
deployment.apps/demo-deployment configured
PS C:\Users\dk92\kuber> Kubectl get replicaset
NAME                         DESIRED   CURRENT   READY   AGE
demo-deployment-785d8cfbbc   3         3         3       89s
demo-deployment-86df76bbcc   0         0         0       15m
PS C:\Users\dk92\kuber> Kubectl get pods
NAME                               READY   STATUS    RESTARTS   AGE
demo-deployment-785d8cfbbc-72qd7   1/1     Running   0          6m21s
demo-deployment-785d8cfbbc-cnlp8   1/1     Running   0          6m28s
demo-deployment-785d8cfbbc-qrzr5   1/1     Running   0          6m34s
PS C:\Users\dk92\kuber> kubectl rollout undo deployment demo-deployment
deployment.apps/demo-deployment rolled back
PS C:\Users\dk92\kuber> Kubectl get replicaset
NAME                         DESIRED   CURRENT   READY   AGE
demo-deployment-785d8cfbbc   0         0         0       6m59s
demo-deployment-86df76bbcc   3         3         3       20m
PS C:\Users\dk92\kuber> Kubectl delete deployment demo-deployment
deployment.apps "demo-deployment" deleted
PS C:\Users\dk92\kuber> Kubectl get deployment
No resources found in default namespace.
PS C:\Users\dk92\kuber> 
```

Assignment 5 - Working with HTTP Liveness Healthchecks.txt

1. Creating a nginx pod with liveness probe with given configuration as below.

- Creating a pod as given configuration.

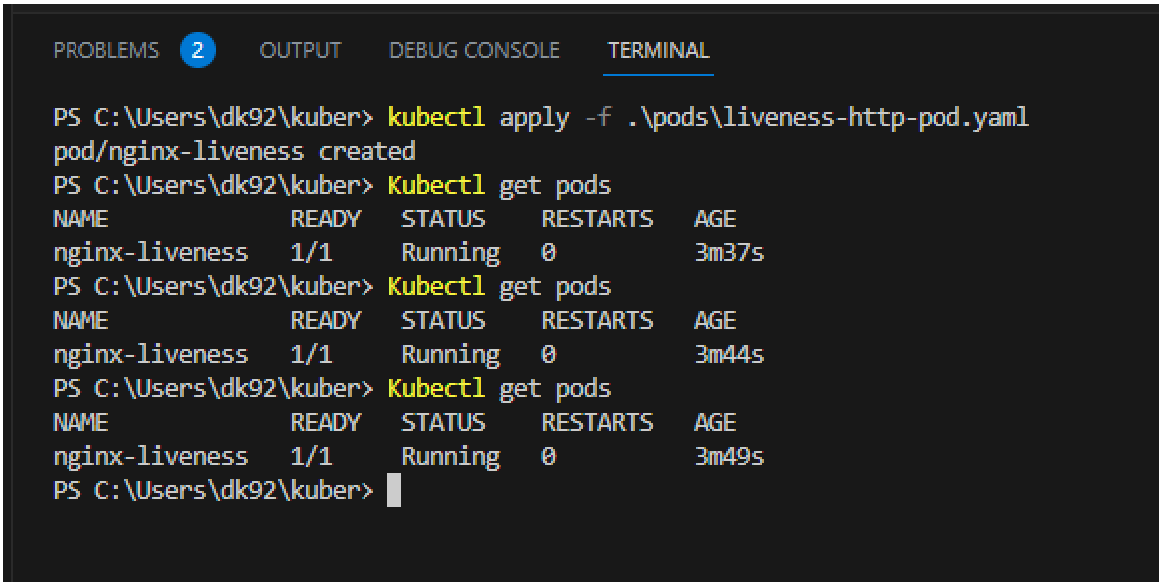

Welcome
! deployment.yaml 1
! liveness-http-pod.yaml 1
pods > ! liveness-http-pod.yaml > {} spec > [ ] containers > {} 0 > {} livenessProbe > {} httpGet
io.k8s.api.core.v1.Pod (v1@pod.json)
PROBLEMS 2
OUTPUT
DEBUG CONSOLE
TERMINAL

```yaml
apiVersion: v1
kind: Pod
metadata:
  name: nginx-liveness
  labels:
    app: nginx
spec:
  containers:
  - name: nginx-demo
    image: nginx:1.15-alpine
    ports:
    - containerPort: 80
    livenessProbe:
      httpGet:
        path: /
        port: 80
      initialDelaySeconds: 15
      periodSeconds: 5
      timeoutSeconds: 1
```

```
PS C:\Users\dk92\kuber> kubectl apply -f .\pods\liveness-http-pod.yaml
pod/nginx-liveness created
PS C:\Users\dk92\kuber>
```

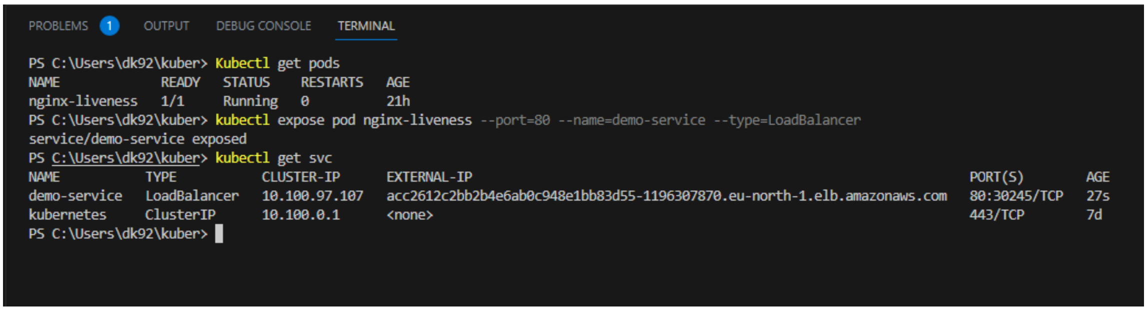
2. Check if the Pods have been created and not restarting

- Command to view pods **kubectl get pods**.
- Pod is created and it is not restarting.



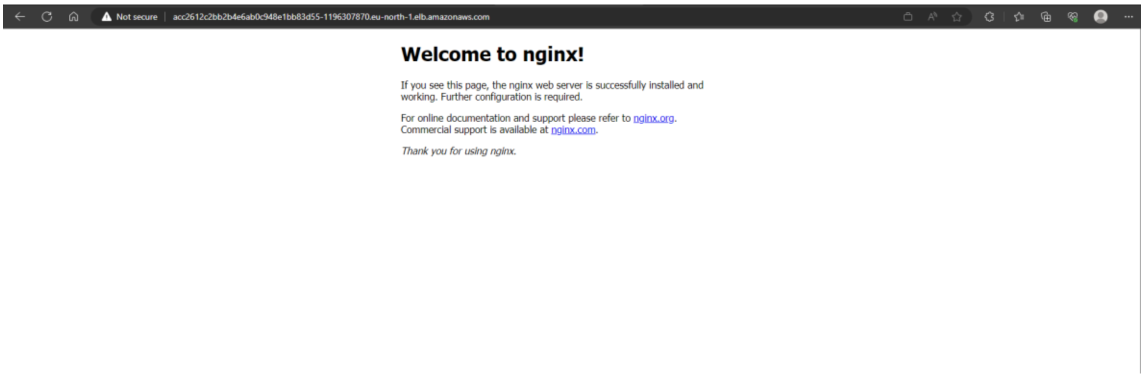
```
PS C:\Users\dk92\kuber> kubectl apply -f .\pods\liveness-http-pod.yaml
pod/nginx-liveness created
PS C:\Users\dk92\kuber> Kubectl get pods
NAME             READY   STATUS    RESTARTS   AGE
nginx-liveness   1/1     Running   0          3m37s
PS C:\Users\dk92\kuber> Kubectl get pods
NAME             READY   STATUS    RESTARTS   AGE
nginx-liveness   1/1     Running   0          3m44s
PS C:\Users\dk92\kuber> Kubectl get pods
NAME             READY   STATUS    RESTARTS   AGE
nginx-liveness   1/1     Running   0          3m49s
PS C:\Users\dk92\kuber>
```

- Expose the nginx liveness container at port 80 using service LoadBalancer.



```
PS C:\Users\dk92\kuber> Kubectl get pods
NAME             READY   STATUS    RESTARTS   AGE
nginx-liveness   1/1     Running   0          21h
PS C:\Users\dk92\kuber> kubectl expose pod nginx-liveness --port=80 --name=demo-service --type=LoadBalancer
service/demo-service exposed
PS C:\Users\dk92\kuber> kubectl get svc
NAME           TYPE           CLUSTER-IP      EXTERNAL-IP                                                                PORT(S)        AGE
demo-service   LoadBalancer   10.100.97.107   acc2612c2bb2b4e6ab0c948e1bb83d55-1196307870.eu-north-1.elb.amazonaws.com   80:30245/TCP   27s
kubernetes     ClusterIP      10.100.0.1      <none>                                                                     443/TCP        7d
PS C:\Users\dk92\kuber> 
```

- Copy the external ip of LoadBalancer form service and paste it into browser
- It is showing nginx welcome page and working fine.


Not secure
acc2612c2bb2b4e6ab0c948e1bb83d55-1196307870.eu-north-1.elb.amazonaws.com

Welcome to nginx!

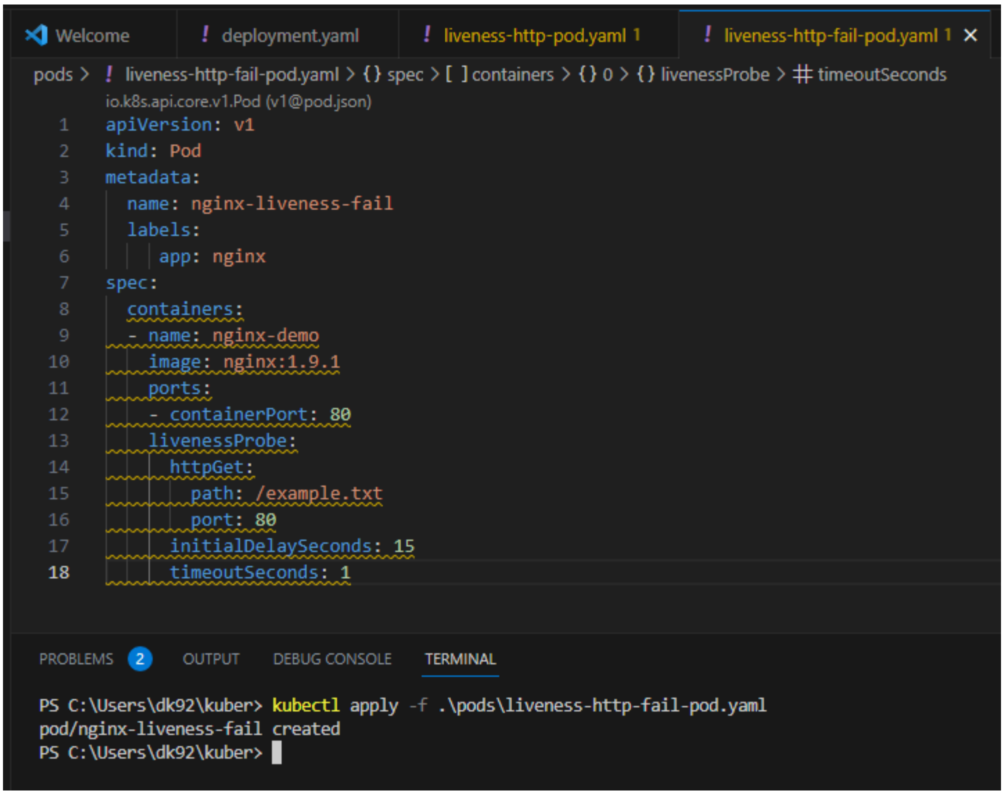
If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org. Commercial support is available at nginx.com.

Thank you for using nginx.

3. Let us now demonstrate failing liveness probe in which we will update path of container as /example.txt which is not present in the container

- Creating a pod as given configuration.


Welcome
deployment.yaml
liveness-http-pod.yaml 1
liveness-http-fail-pod.yaml 1
pods > liveness-http-fail-pod.yaml > {} spec > [ ] containers > {} 0 > {} livenessProbe > timeoutSeconds
io.k8s.api.core.v1.Pod (v1@pod.json)

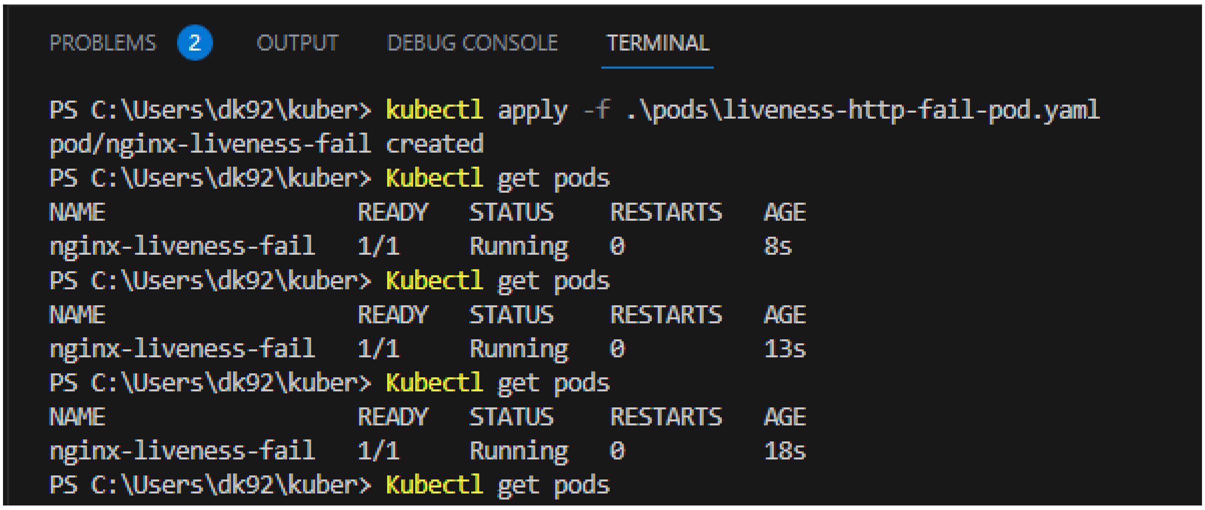
```yaml
apiVersion: v1
kind: Pod
metadata:
  name: nginx-liveness-fail
  labels:
    app: nginx
spec:
  containers:
  - name: nginx-demo
    image: nginx:1.9.1
    ports:
    - containerPort: 80
    livenessProbe:
      httpGet:
        path: /example.txt
        port: 80
      initialDelaySeconds: 15
      timeoutSeconds: 1
```

PROBLEMS 2
OUTPUT
DEBUG CONSOLE
TERMINAL

```
PS C:\Users\dk92\kuber> kubectl apply -f .\pods\liveness-http-fail-pod.yaml
pod/nginx-liveness-fail created
PS C:\Users\dk92\kuber> 
```

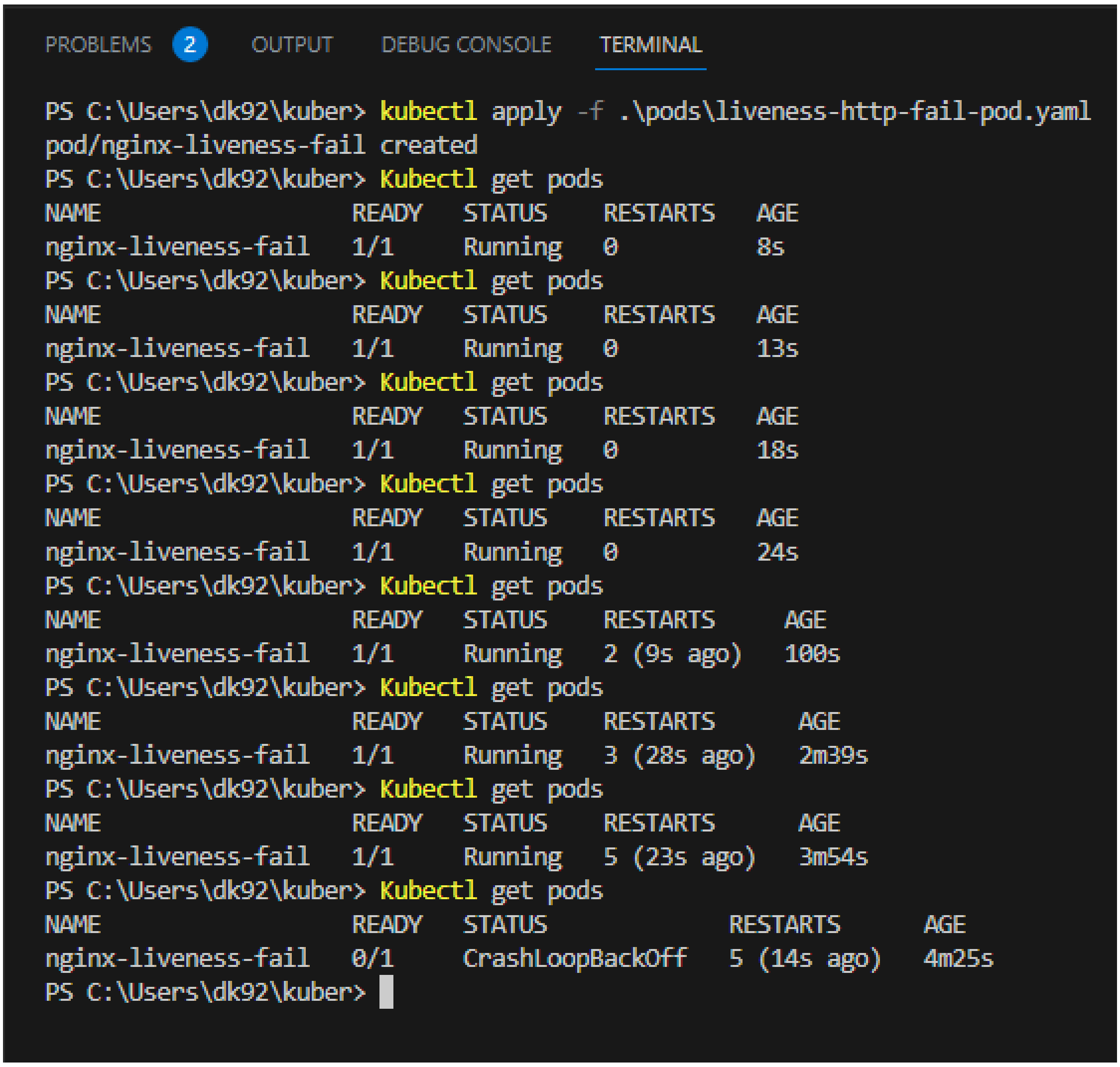
4. Check if the Pods have been created and not restarting

- Command to view pods **kubectl get pods**.
- Pod is created and it is not restarting.



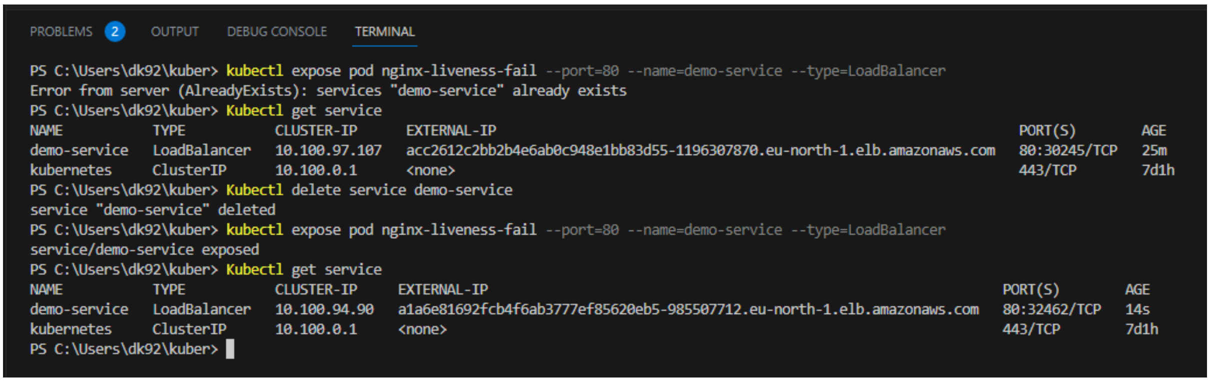
```
PS C:\Users\dk92\kuber> kubectl apply -f .\pods\liveness-http-fail-pod.yaml
pod/nginx-liveness-fail created
PS C:\Users\dk92\kuber> Kubectl get pods
NAME                  READY   STATUS    RESTARTS   AGE
nginx-liveness-fail   1/1     Running   0          8s
PS C:\Users\dk92\kuber> Kubectl get pods
NAME                  READY   STATUS    RESTARTS   AGE
nginx-liveness-fail   1/1     Running   0          13s
PS C:\Users\dk92\kuber> Kubectl get pods
NAME                  READY   STATUS    RESTARTS   AGE
nginx-liveness-fail   1/1     Running   0          18s
PS C:\Users\dk92\kuber> Kubectl get pods
```

Regularly check output of kubectl get pods ..It would be failing and Pod would get restarted automatically . After around 5 Restarts , Pod would get crashed.



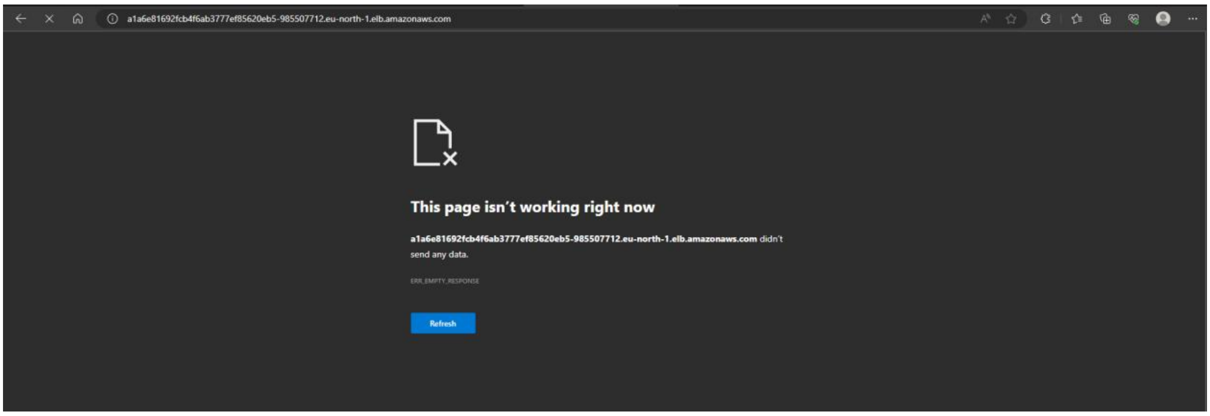
```
PS C:\Users\dk92\kuber> kubectl apply -f .\pods\liveness-http-fail-pod.yaml
pod/nginx-liveness-fail created
PS C:\Users\dk92\kuber> Kubectl get pods
NAME                  READY   STATUS    RESTARTS   AGE
nginx-liveness-fail   1/1     Running   0          8s
PS C:\Users\dk92\kuber> Kubectl get pods
NAME                  READY   STATUS    RESTARTS   AGE
nginx-liveness-fail   1/1     Running   0          13s
PS C:\Users\dk92\kuber> Kubectl get pods
NAME                  READY   STATUS    RESTARTS   AGE
nginx-liveness-fail   1/1     Running   0          18s
PS C:\Users\dk92\kuber> Kubectl get pods
NAME                  READY   STATUS    RESTARTS   AGE
nginx-liveness-fail   1/1     Running   0          24s
PS C:\Users\dk92\kuber> Kubectl get pods
NAME                  READY   STATUS    RESTARTS     AGE
nginx-liveness-fail   1/1     Running   2 (9s ago)   100s
PS C:\Users\dk92\kuber> Kubectl get pods
NAME                  READY   STATUS    RESTARTS      AGE
nginx-liveness-fail   1/1     Running   3 (28s ago)   2m39s
PS C:\Users\dk92\kuber> Kubectl get pods
NAME                  READY   STATUS    RESTARTS      AGE
nginx-liveness-fail   1/1     Running   5 (23s ago)   3m54s
PS C:\Users\dk92\kuber> Kubectl get pods
NAME                  READY   STATUS             RESTARTS      AGE
nginx-liveness-fail   0/1     CrashLoopBackOff   5 (14s ago)   4m25s
PS C:\Users\dk92\kuber> 
```

- Expose the nginx liveness fail container at port 80 using service LoadBalancer.



```
PS C:\Users\dk92\kuber> kubectl expose pod nginx-liveness-fail --port=80 --name=demo-service --type=LoadBalancer
Error from server (AlreadyExists): services "demo-service" already exists
PS C:\Users\dk92\kuber> Kubectl get service
NAME           TYPE           CLUSTER-IP      EXTERNAL-IP                                                               PORT(S)        AGE
demo-service   LoadBalancer   10.100.97.107   acc2612c2bb2b4e6ab0c948e1bb83d55-1196307870.eu-north-1.elb.amazonaws.com   80:30245/TCP   25m
kubernetes     ClusterIP      10.100.0.1      <none>                                                                    443/TCP        7d1h
PS C:\Users\dk92\kuber> Kubectl delete service demo-service
service "demo-service" deleted
PS C:\Users\dk92\kuber> kubectl expose pod nginx-liveness-fail --port=80 --name=demo-service --type=LoadBalancer
service/demo-service exposed
PS C:\Users\dk92\kuber> Kubectl get service
NAME           TYPE           CLUSTER-IP     EXTERNAL-IP                                                              PORT(S)        AGE
demo-service   LoadBalancer   10.100.94.90   a1a6e81692fcb4f6ab3777ef85620eb5-985507712.eu-north-1.elb.amazonaws.com   80:32462/TCP   14s
kubernetes     ClusterIP      10.100.0.1     <none>                                                                   443/TCP        7d1h
PS C:\Users\dk92\kuber> 
```

- Copy the external ip of LoadBalancer form service and paste it into browser
- The pod has been crashed so it is not working



This page isn't working right now

a1a6e81692fcb4f6ab3777ef85620eb5-985507712.eu-north-1.elb.amazonaws.com didn't send any data.

ERR_EMPTY_RESPONSE