Week 5 Assignment 1

1. Create an nginx deployment with 1 replica:

```
user@Dell-G3-3500:~$ kubectl create deployment nginx-deployment --image=nginx --replicas=1
deployment.apps/nginx-deployment created
user@Dell-G3-3500:~$ kubectl get deployments.apps
NAME               READY   UP-TO-DATE   AVAILABLE   AGE
kubia              1/1     1            1           43h
nginx-deployment   0/1     1            0           14s
user@Dell-G3-3500:~$
```

2. Scale the deployment to 3 replicas:

```
user@Dell-G3-3500: ~
user@Dell-G3-3500:~$ kubectl scale deployment nginx-deployment --replicas=3
deployment.apps/nginx-deployment scaled
user@Dell-G3-3500:~$ kubectl get deployments.apps
NAME               READY   UP-TO-DATE   AVAILABLE   AGE
kubia              1/1     1            1           43h
nginx-deployment   0/3     3            0           113s
user@Dell-G3-3500:~$
```

3. Expose the deployment with a ClusterIP service:

```
user@Dell-G3-3500: ~                                                                                          –  □  ×
user@Dell-G3-3500:~$ kubectl expose deployment nginx-deployment --port=80 --type=ClusterIP
service/nginx-deployment exposed
user@Dell-G3-3500:~$ kubectl get pods -o wide
NAME                                READY   STATUS             RESTARTS        AGE     IP           NODE           NOMINATED NODE   RE
ADINESS GATES
kubia-587b7cd6b7-pkvcd              1/1     Running            1 (7m12s ago)   43h     10.42.0.26   dell-g3-3500   <none>           <n
one>
nginx-deployment-6d6565499c-6pbvz   0/1     ContainerCreating  0               2m42s   <none>       dell-g3-3500   <none>           <n
one>
nginx-deployment-6d6565499c-9bpm5   0/1     ImagePullBackOff   0               4m29s   10.42.0.27   dell-g3-3500   <none>           <n
one>
nginx-deployment-6d6565499c-ln9g7   0/1     ContainerCreating  0               2m42s   <none>       dell-g3-3500   <none>           <n
one>
user@Dell-G3-3500:~$ kubectl get deployments.apps
NAME               READY   UP-TO-DATE   AVAILABLE   AGE
kubia              1/1     1            1           43h
nginx-deployment   3/3     3            3           6m16s
user@Dell-G3-3500:~$ kubectl get pods -o wide
NAME                                READY   STATUS    RESTARTS       AGE     IP           NODE           NOMINATED NODE   READINESS GAT
ES
kubia-587b7cd6b7-pkvcd              1/1     Running   1 (9m1s ago)   43h     10.42.0.26   dell-g3-3500   <none>           <none>
nginx-deployment-6d6565499c-6pbvz   1/1     Running   0              4m31s   10.42.0.28   dell-g3-3500   <none>           <none>
nginx-deployment-6d6565499c-9bpm5   1/1     Running   0              6m18s   10.42.0.27   dell-g3-3500   <none>           <none>
nginx-deployment-6d6565499c-ln9g7   1/1     Running   0              4m31s   10.42.0.29   dell-g3-3500   <none>           <none>
user@Dell-G3-3500:~$
```

4. Testing connectivity:

4.1 First container

```
user@Dell-G3-3500:~$ kubectl get pods -o wide
NAME                                      READY   STATUS    RESTARTS      AGE    IP           NODE          NOMINATED NODE   READINESS GATES
kubia-587b7cd6b7-pkvcd                    1/1     Running   1 (19m ago)   43h    10.42.0.26   dell-g3-3500  <none>           <none>
nginx-deployment-6d6565499c-6pbvz         1/1     Running   0             15m    10.42.0.28   dell-g3-3500  <none>           <none>
nginx-deployment-6d6565499c-9bpm5         1/1     Running   0             17m    10.42.0.27   dell-g3-3500  <none>           <none>
nginx-deployment-6d6565499c-ln9g7         1/1     Running   0             15m    10.42.0.29   dell-g3-3500  <none>           <none>
user@Dell-G3-3500:~$ kubectl exec -it nginx-deployment-6d6565499c-6pbvz -- curl 10.42.0.28
<!DOCTYPE html>
<html>
<head>
<title>Welcome to nginx!</title>
<style>
html { color-scheme: light dark; }
body { width: 35em; margin: 0 auto;
font-family: Tahoma, Verdana, Arial, sans-serif; }
</style>
</head>
<body>
<h1>Welcome to nginx!</h1>
<p>If you see this page, the nginx web server is successfully installed and
working. Further configuration is required.</p>

<p>For online documentation and support please refer to
<a href="http://nginx.org/">nginx.org</a>.<br/>
Commercial support is available at
<a href="http://nginx.com/">nginx.com</a>.</p>

<p><em>Thank you for using nginx.</em></p>
</body>
</html>
user@Dell-G3-3500:~$ _
```

4.2 Second container

```
<p>For online documentation and support please refer to
<a href="http://nginx.org/">nginx.org</a>.<br/>
Commercial support is available at
<a href="http://nginx.com/">nginx.com</a>.</p>

<p><em>Thank you for using nginx.</em></p>
</body>
</html>
user@Dell-G3-3500:~$ kubectl exec -it nginx-deployment-6d6565499c-9bpm5 -- curl 10.42.0.27
<!DOCTYPE html>
<html>
<head>
<title>Welcome to nginx!</title>
<style>
html { color-scheme: light dark; }
body { width: 35em; margin: 0 auto;
font-family: Tahoma, Verdana, Arial, sans-serif; }
</style>
</head>
<body>
<h1>Welcome to nginx!</h1>
<p>If you see this page, the nginx web server is successfully installed and
working. Further configuration is required.</p>

<p>For online documentation and support please refer to
<a href="http://nginx.org/">nginx.org</a>.<br/>
Commercial support is available at
<a href="http://nginx.com/">nginx.com</a>.</p>

<p><em>Thank you for using nginx.</em></p>
</body>
</html>
user@Dell-G3-3500:~$
```

4.3 Third container


```
<p>For online documentation and support please refer to
<a href="http://nginx.org/">nginx.org</a>.<br/>
Commercial support is available at
<a href="http://nginx.com/">nginx.com</a>.</p>

<p><em>Thank you for using nginx.</em></p>
</body>
</html>
user@Dell-G3-3500:~$ kubectl exec -it nginx-deployment-6d6565499c-ln9g7 -- curl 10.42.0.29
<!DOCTYPE html>
<html>
<head>
<title>Welcome to nginx!</title>
<style>
html { color-scheme: light dark; }
body { width: 35em; margin: 0 auto;
font-family: Tahoma, Verdana, Arial, sans-serif; }
</style>
</head>
<body>
<h1>Welcome to nginx!</h1>
<p>If you see this page, the nginx web server is successfully installed and
working. Further configuration is required.</p>

<p>For online documentation and support please refer to
<a href="http://nginx.org/">nginx.org</a>.<br/>
Commercial support is available at
<a href="http://nginx.com/">nginx.com</a>.</p>

<p><em>Thank you for using nginx.</em></p>
</body>
</html>
user@Dell-G3-3500:~$
```

4.4 Using service IP


```
user@Dell-G3-3500:~$ kubectl get services
NAME                TYPE        CLUSTER-IP     EXTERNAL-IP   PORT(S)   AGE
kubernetes          ClusterIP   10.43.0.1      <none>        443/TCP   2d3h
nginx-deployment    ClusterIP   10.43.89.107   <none>        80/TCP    14m
user@Dell-G3-3500:~$ kubectl exec -it nginx-deployment-6d6565499c-ln9g7 -- curl 10.43.89.107
<!DOCTYPE html>
<html>
<head>
<title>Welcome to nginx!</title>
<style>
html { color-scheme: light dark; }
body { width: 35em; margin: 0 auto;
font-family: Tahoma, Verdana, Arial, sans-serif; }
</style>
</head>
<body>
<h1>Welcome to nginx!</h1>
<p>If you see this page, the nginx web server is successfully installed and
working. Further configuration is required.</p>

<p>For online documentation and support please refer to
<a href="http://nginx.org/">nginx.org</a>.<br/>
Commercial support is available at
<a href="http://nginx.com/">nginx.com</a>.</p>

<p><em>Thank you for using nginx.</em></p>
</body>
</html>
user@Dell-G3-3500:~$
```

5. Delete pods using a label selector:


```
user@Dell-G3-3500:~$ kubectl delete pods -l app=nginx-deployment
pod "nginx-deployment-6d6565499c-6pbvz" deleted
pod "nginx-deployment-6d6565499c-9bpm5" deleted
pod "nginx-deployment-6d6565499c-ln9g7" deleted
user@Dell-G3-3500:~$
```

6. Kill the main process of a container

```
user@Dell-G3-3500: ~                                                                                    —  □  ×
user@Dell-G3-3500:~$ kubectl get pods
NAME                                 READY   STATUS    RESTARTS      AGE
kubia-587b7cd6b7-pkvcd               1/1     Running   1 (27m ago)   43h
nginx-deployment-6d6565499c-8fth2    1/1     Running   0             4m40s
nginx-deployment-6d6565499c-grnm7    1/1     Running   0             4m40s
nginx-deployment-6d6565499c-zxsc9    1/1     Running   0             4m40s
user@Dell-G3-3500:~$ kubectl exec -it nginx-deployment-6d6565499c-zxsc9 -- /bin/sh
# kill 1
# command terminated with exit code 137
user@Dell-G3-3500:~$ kubectl get pods
NAME                                 READY   STATUS    RESTARTS      AGE
kubia-587b7cd6b7-pkvcd               1/1     Running   1 (30m ago)   43h
nginx-deployment-6d6565499c-8fth2    1/1     Running   0             7m20s
nginx-deployment-6d6565499c-grnm7    1/1     Running   0             7m20s
nginx-deployment-6d6565499c-zxsc9    1/1     Running   1 (4s ago)    7m20s
user@Dell-G3-3500:~$ kubectl get pods -o wide
NAME                                 READY   STATUS    RESTARTS      AGE    IP           NODE          NOMINATED NODE   READINESS GATE
S
kubia-587b7cd6b7-pkvcd               1/1     Running   1 (30m ago)   43h    10.42.0.26   dell-g3-3500  <none>           <none>
nginx-deployment-6d6565499c-8fth2    1/1     Running   0             7m41s  10.42.0.34   dell-g3-3500  <none>           <none>
nginx-deployment-6d6565499c-grnm7    1/1     Running   0             7m41s  10.42.0.32   dell-g3-3500  <none>           <none>
nginx-deployment-6d6565499c-zxsc9    1/1     Running   1 (25s ago)   7m41s  10.42.0.33   dell-g3-3500  <none>           <none>
user@Dell-G3-3500:~$ _
```

- Each screenshot captures both the command executed and its corresponding output. When attempting to terminate a process within a container, Kubernetes automatically restarts the pod. Upon successful termination, we should observe that the 'RESTARTS' column for the affected pod increments from 0 to 1.

- It's worth noting that when trying to display processes inside the Nginx container, commands such as `ps` or `htop` were unavailable. This is because the Nginx container image is intentionally stripped of these utilities to maintain a lightweight footprint. However, it's generally understood that the main process ID in a container is always 1, so we can target this PID directly for termination.

- When deleting pods, we can observe that Kubernetes promptly creates new pods to maintain the desired state. These new pods are assigned different IP addresses, while the associated service IP remains constant, ensuring uninterrupted accessibility.