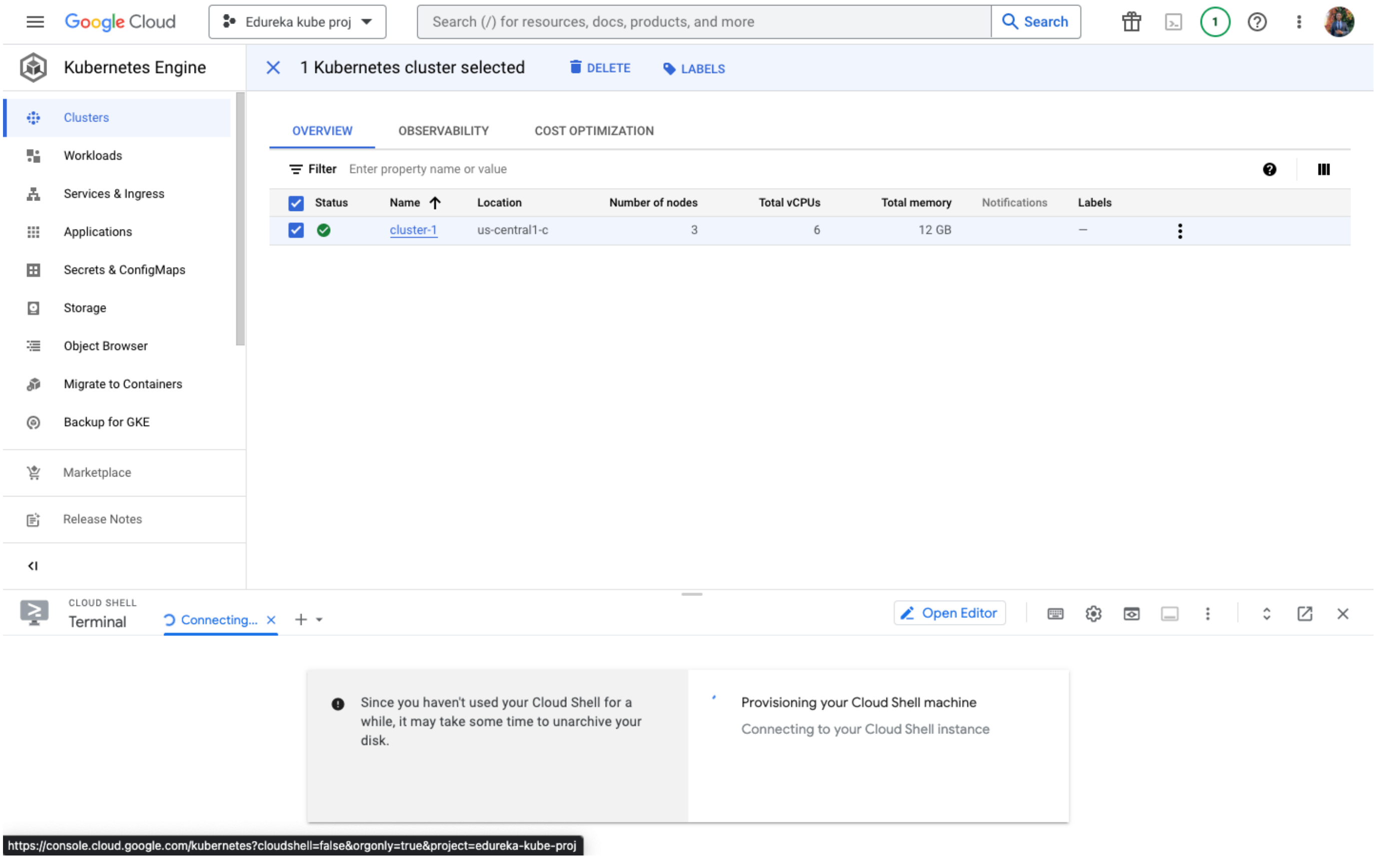
March 15th, 2023.

Google Cloud Provider: Kubernetes

Creating a Service Object of type: Cluster IP

Step 1: Create standard K8s cluster and connect to it.



☑	Status	Name ↑	Location	Number of nodes	Total vCPUs	Total memory	Notifications	Labels
☑	✓	cluster-1	us-central1-c	3	6	12 GB		–

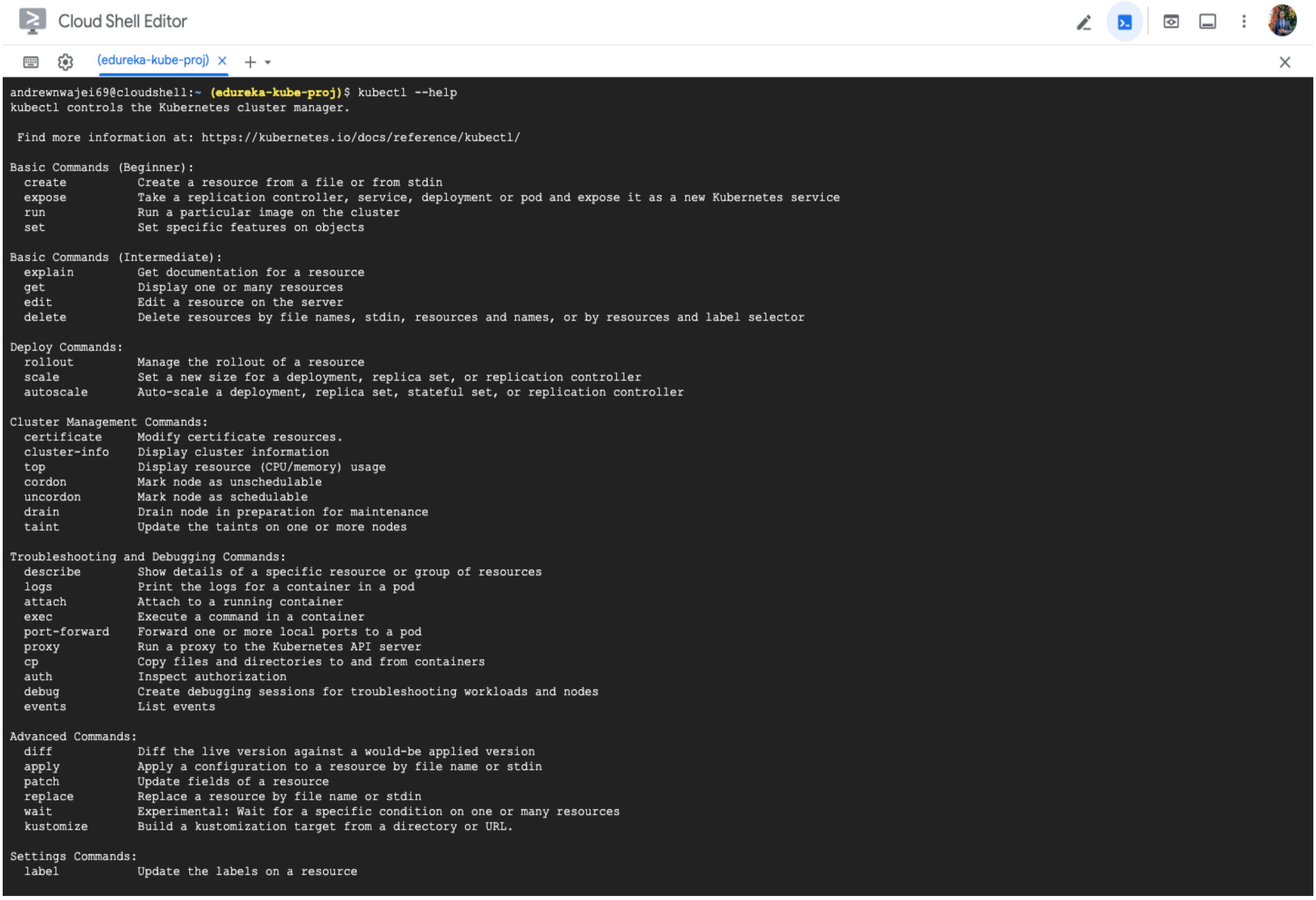
Since you haven't used your Cloud Shell for a while, it may take some time to unarchive your disk.

Provisioning your Cloud Shell machine

Connecting to your Cloud Shell instance

https://console.cloud.google.com/kubernetes?cloudshell=false&orgonly=true&project=edureka-kube-proj

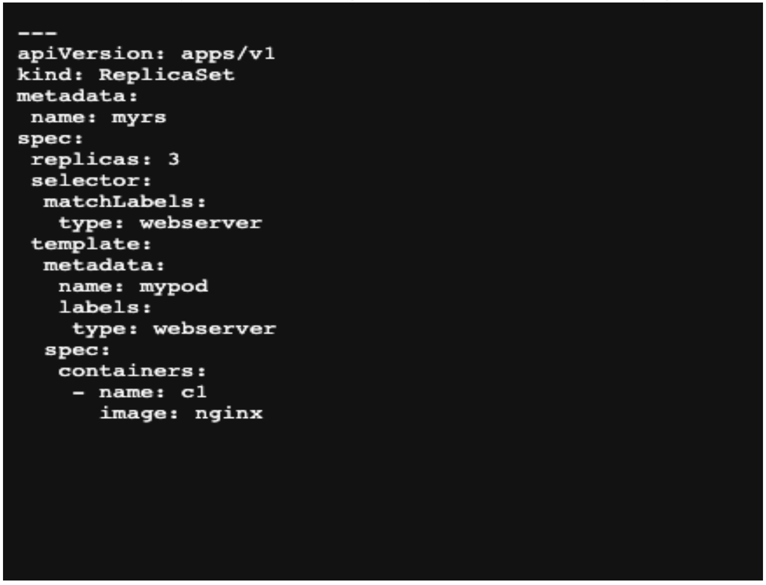
2. Using the: kubectl --help command to just list the available commands on Kubernetes.



```
andrewnwajei69@cloudshell:~ (edureka-kube-proj)$ kubectl --help
kubectl controls the Kubernetes cluster manager.

 Find more information at: https://kubernetes.io/docs/reference/kubectl/

Basic Commands (Beginner):
  create          Create a resource from a file or from stdin
  expose          Take a replication controller, service, deployment or pod and expose it as a new Kubernetes service
  run             Run a particular image on the cluster
  set             Set specific features on objects

Basic Commands (Intermediate):
  explain         Get documentation for a resource
  get             Display one or many resources
  edit            Edit a resource on the server
  delete          Delete resources by file names, stdin, resources and names, or by resources and label selector

Deploy Commands:
  rollout         Manage the rollout of a resource
  scale           Set a new size for a deployment, replica set, or replication controller
  autoscale       Auto-scale a deployment, replica set, stateful set, or replication controller

Cluster Management Commands:
  certificate     Modify certificate resources.
  cluster-info    Display cluster information
  top             Display resource (CPU/memory) usage
  cordon          Mark node as unschedulable
  uncordon        Mark node as schedulable
  drain           Drain node in preparation for maintenance
  taint           Update the taints on one or more nodes

Troubleshooting and Debugging Commands:
  describe        Show details of a specific resource or group of resources
  logs            Print the logs for a container in a pod
  attach          Attach to a running container
  exec            Execute a command in a container
  port-forward    Forward one or more local ports to a pod
  proxy           Run a proxy to the Kubernetes API server
  cp              Copy files and directories to and from containers
  auth            Inspect authorization
  debug           Create debugging sessions for troubleshooting workloads and nodes
  events          List events

Advanced Commands:
  diff            Diff the live version against a would-be applied version
  apply           Apply a configuration to a resource by file name or stdin
  patch           Update fields of a resource
  replace         Replace a resource by file name or stdin
  wait            Experimental: Wait for a specific condition on one or many resources
  kustomize       Build a kustomization target from a directory or URL.

Settings Commands:
  label           Update the labels on a resource
```

3. Using the get pods command before creating replicaset to show that there were no prior pods running or created.

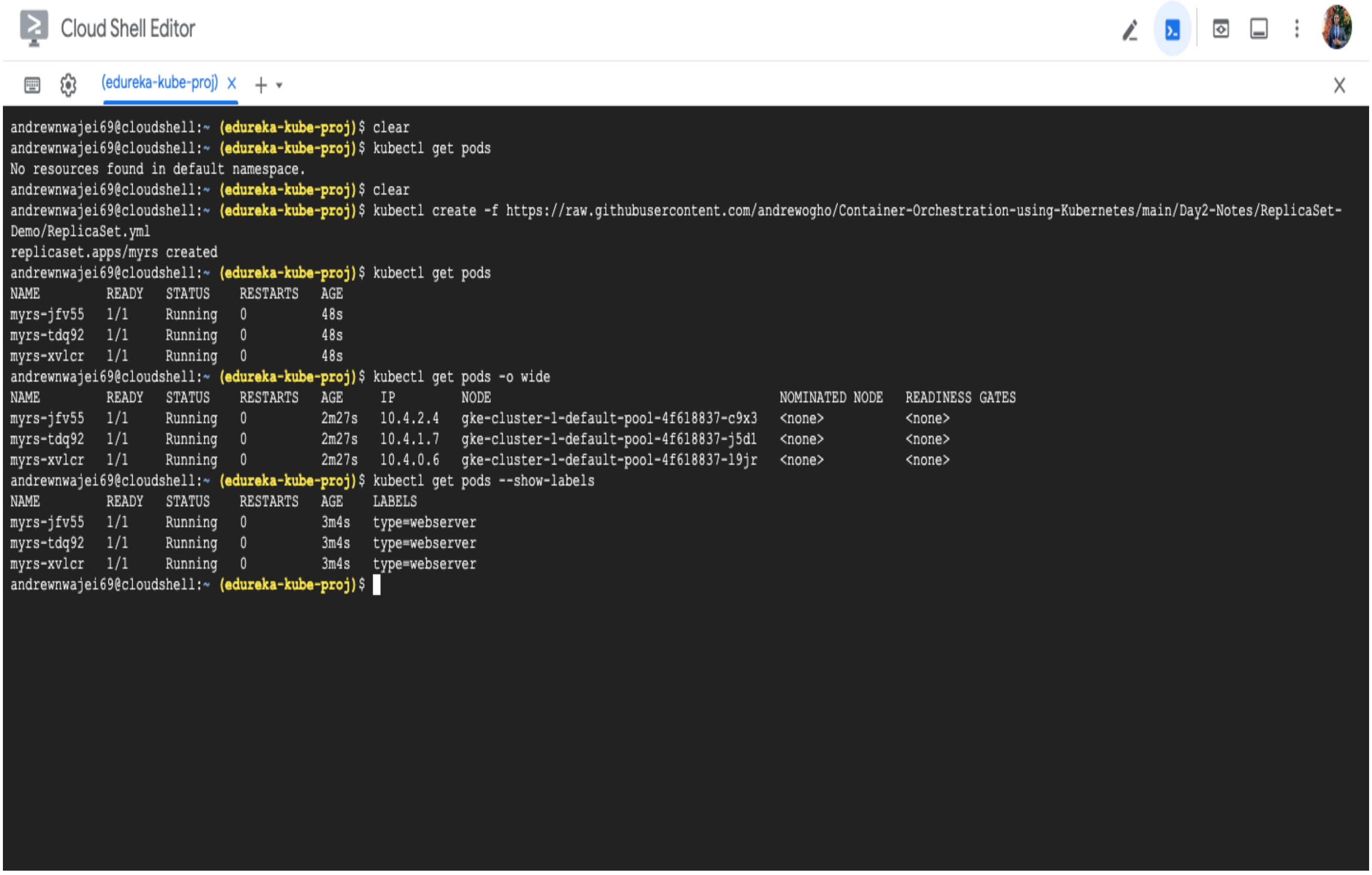
4. Create a replica set using raw yaml file from my github repo.



```
---
apiVersion: apps/v1
kind: ReplicaSet
metadata:
 name: myrs
spec:
 replicas: 3
 selector:
  matchLabels:
   type: webserver
 template:
  metadata:
   name: mypod
   labels:
    type: webserver
  spec:
   containers:
    - name: c1
      image: nginx
```

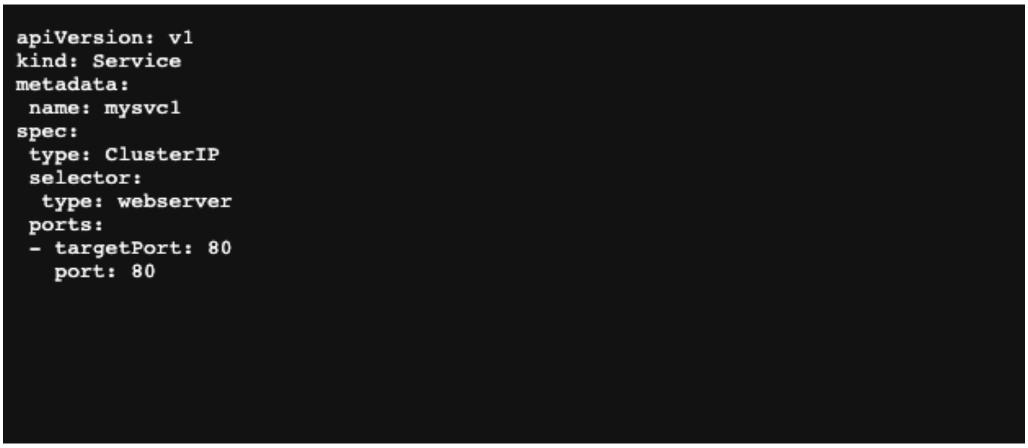
5. Use the get pods command to show that pods have been create and are running.

6. “-o wide” command used for more detailed information on the replicasets created



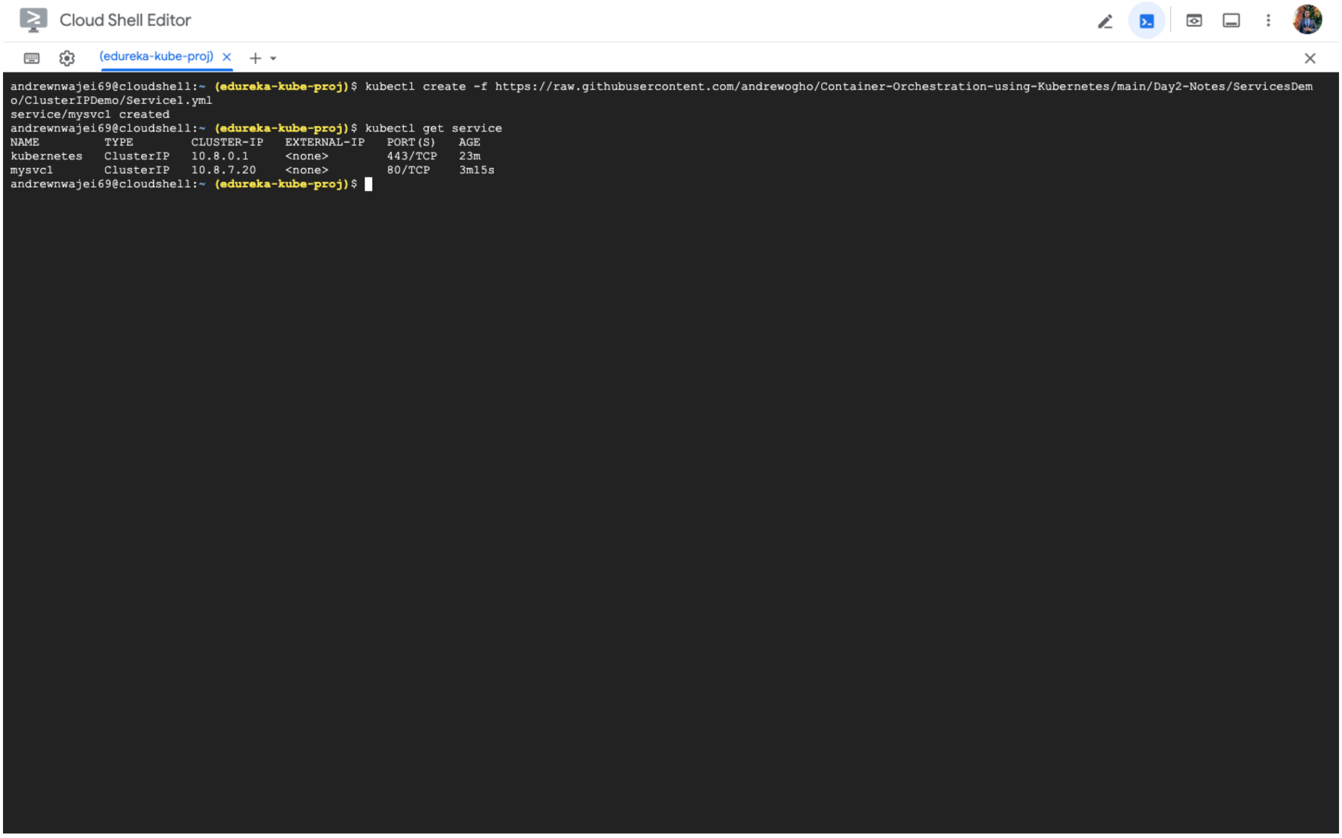
```
andrewnwajei69@cloudshell:~ (edureka-kube-proj)$ clear
andrewnwajei69@cloudshell:~ (edureka-kube-proj)$ kubectl get pods
No resources found in default namespace.
andrewnwajei69@cloudshell:~ (edureka-kube-proj)$ clear
andrewnwajei69@cloudshell:~ (edureka-kube-proj)$ kubectl create -f https://raw.githubusercontent.com/andrewogho/Container-Orchestration-using-Kubernetes/main/Day2-Notes/ReplicaSet-
Demo/ReplicaSet.yml
replicaset.apps/myrs created
andrewnwajei69@cloudshell:~ (edureka-kube-proj)$ kubectl get pods
NAME         READY   STATUS    RESTARTS   AGE
myrs-jfv55   1/1     Running   0          48s
myrs-tdq92   1/1     Running   0          48s
myrs-xvlcr   1/1     Running   0          48s
andrewnwajei69@cloudshell:~ (edureka-kube-proj)$ kubectl get pods -o wide
NAME         READY   STATUS    RESTARTS   AGE     IP         NODE                                         NOMINATED NODE   READINESS GATES
myrs-jfv55   1/1     Running   0          2m27s   10.4.2.4   gke-cluster-1-default-pool-4f618837-c9x3   <none>           <none>
myrs-tdq92   1/1     Running   0          2m27s   10.4.1.7   gke-cluster-1-default-pool-4f618837-j5dl   <none>           <none>
myrs-xvlcr   1/1     Running   0          2m27s   10.4.0.6   gke-cluster-1-default-pool-4f618837-19jr   <none>           <none>
andrewnwajei69@cloudshell:~ (edureka-kube-proj)$ kubectl get pods --show-labels
NAME         READY   STATUS    RESTARTS   AGE    LABELS
myrs-jfv55   1/1     Running   0          3m4s   type=webserver
myrs-tdq92   1/1     Running   0          3m4s   type=webserver
myrs-xvlcr   1/1     Running   0          3m4s   type=webserver
andrewnwajei69@cloudshell:~ (edureka-kube-proj)$
```

7. Create service using raw yaml file from my github repo.



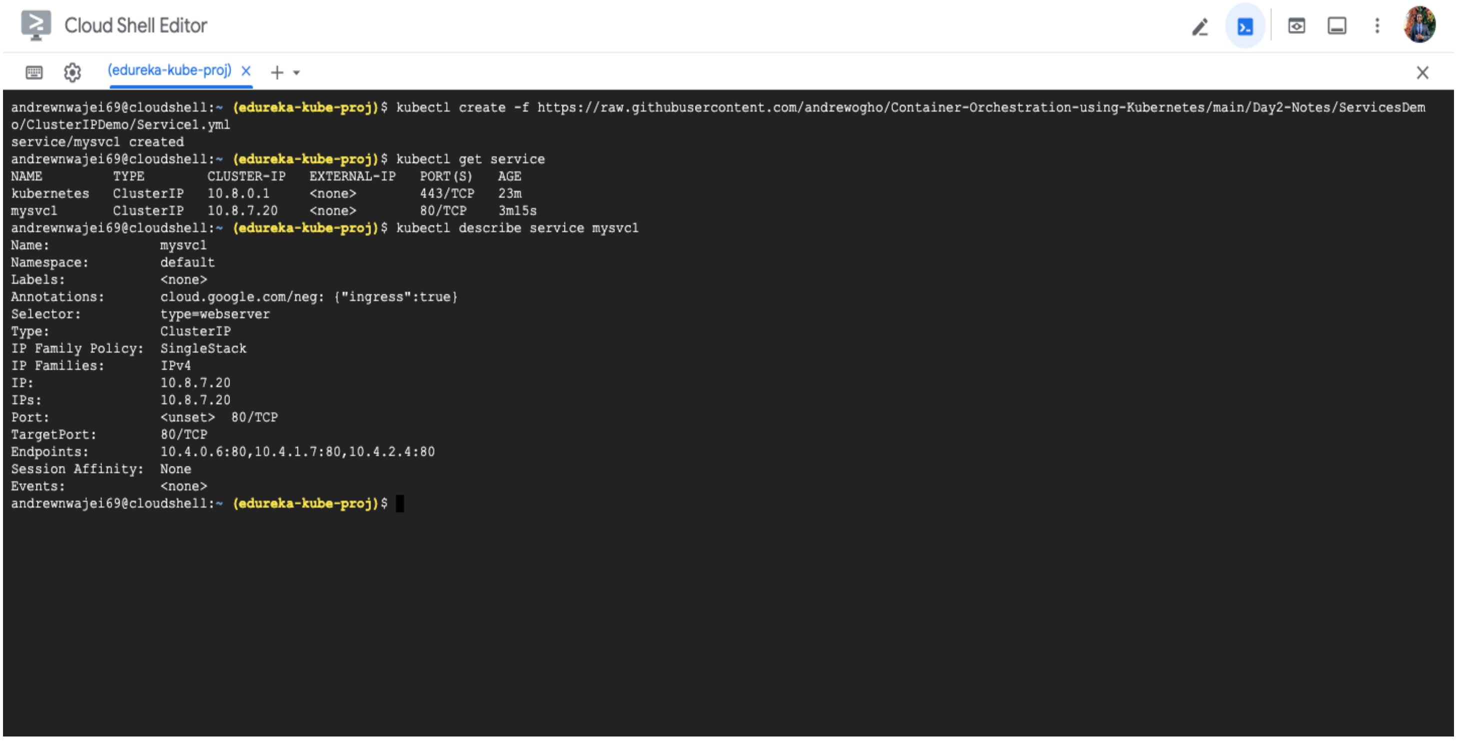
```
apiVersion: v1
kind: Service
metadata:
 name: mysvc1
spec:
 type: ClusterIP
 selector:
  type: webserver
 ports:
 - targetPort: 80
   port: 80
```

8. Kubectl get service to show “mysvc1” has been created with a clusterIP address.

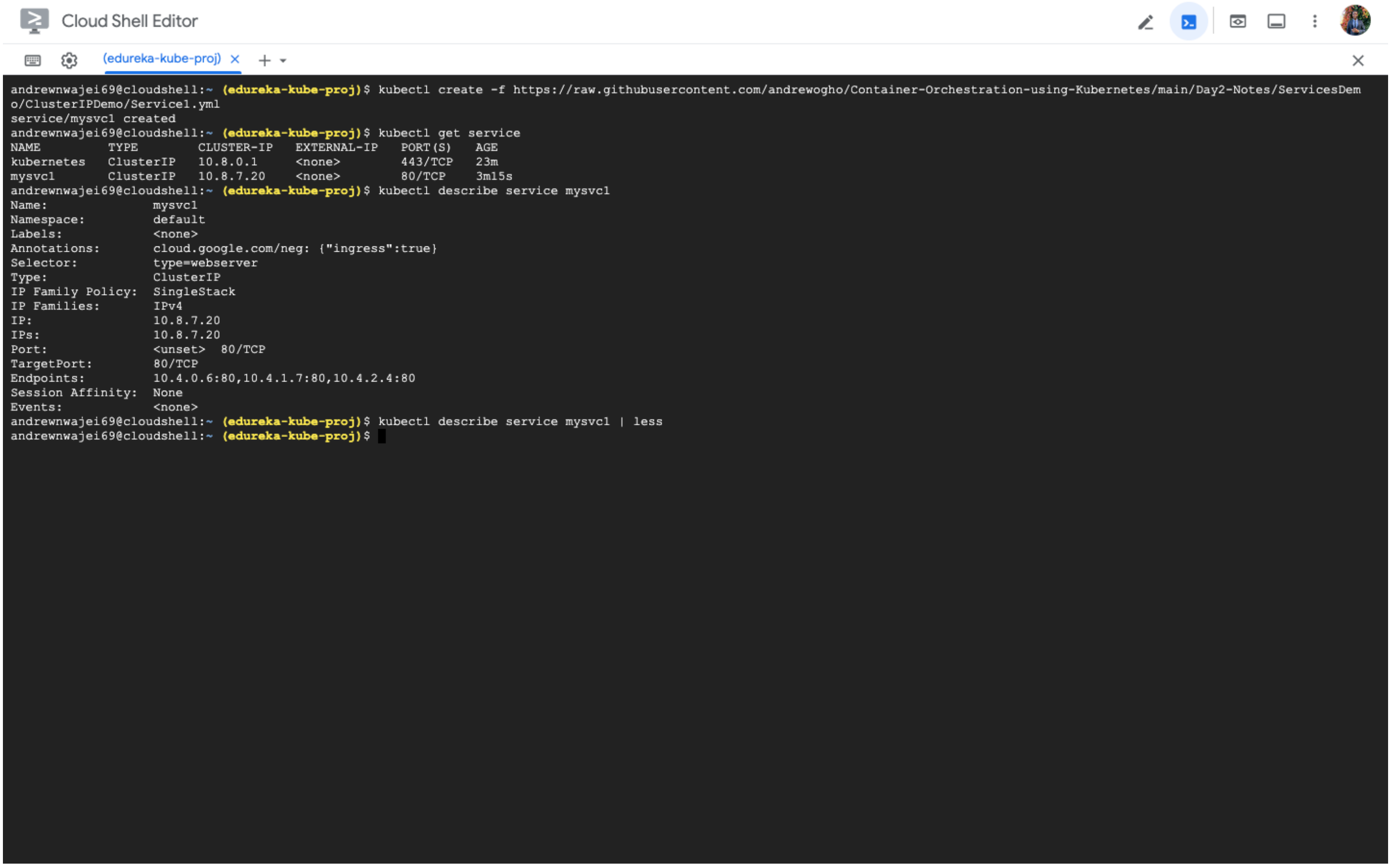


```
andrewnwajei69@cloudshell:~ (edureka-kube-proj)$ kubectl create -f https://raw.githubusercontent.com/andrewogho/Container-Orchestration-using-Kubernetes/main/Day2-Notes/ServicesDem
o/ClusterIPDemo/Service1.yml
service/mysvc1 created
andrewnwajei69@cloudshell:~ (edureka-kube-proj)$ kubectl get service
NAME         TYPE        CLUSTER-IP   EXTERNAL-IP   PORT(S)   AGE
kubernetes   ClusterIP   10.8.0.1     <none>        443/TCP   23m
mysvc1       ClusterIP   10.8.7.20    <none>        80/TCP    3m15s
andrewnwajei69@cloudshell:~ (edureka-kube-proj)$ 
```

9. Using the describe command tagging “mysvc1” to get details of the service created.

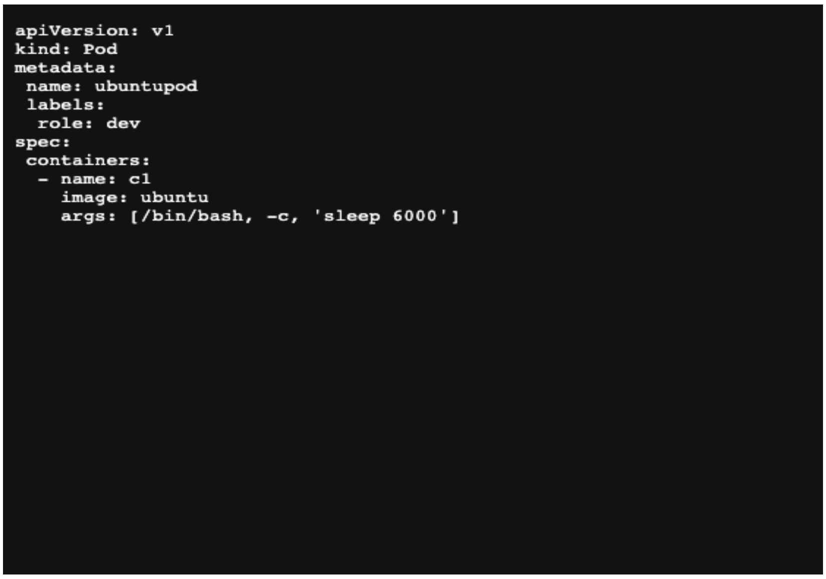


```
andrewnwajei69@cloudshell:~ (edureka-kube-proj)$ kubectl create -f https://raw.githubusercontent.com/andrewogho/Container-Orchestration-using-Kubernetes/main/Day2-Notes/ServicesDem
o/ClusterIPDemo/Service1.yml
service/mysvc1 created
andrewnwajei69@cloudshell:~ (edureka-kube-proj)$ kubectl get service
NAME         TYPE        CLUSTER-IP   EXTERNAL-IP   PORT(S)   AGE
kubernetes   ClusterIP   10.8.0.1     <none>        443/TCP   23m
mysvc1       ClusterIP   10.8.7.20    <none>        80/TCP    3m15s
andrewnwajei69@cloudshell:~ (edureka-kube-proj)$ kubectl describe service mysvc1
Name:              mysvc1
Namespace:         default
Labels:            <none>
Annotations:       cloud.google.com/neg: {"ingress":true}
Selector:          type=webserver
Type:              ClusterIP
IP Family Policy:  SingleStack
IP Families:       IPv4
IP:                10.8.7.20
IPs:               10.8.7.20
Port:              <unset>  80/TCP
TargetPort:        80/TCP
Endpoints:         10.4.0.6:80,10.4.1.7:80,10.4.2.4:80
Session Affinity:  None
Events:            <none>
andrewnwajei69@cloudshell:~ (edureka-kube-proj)$
```

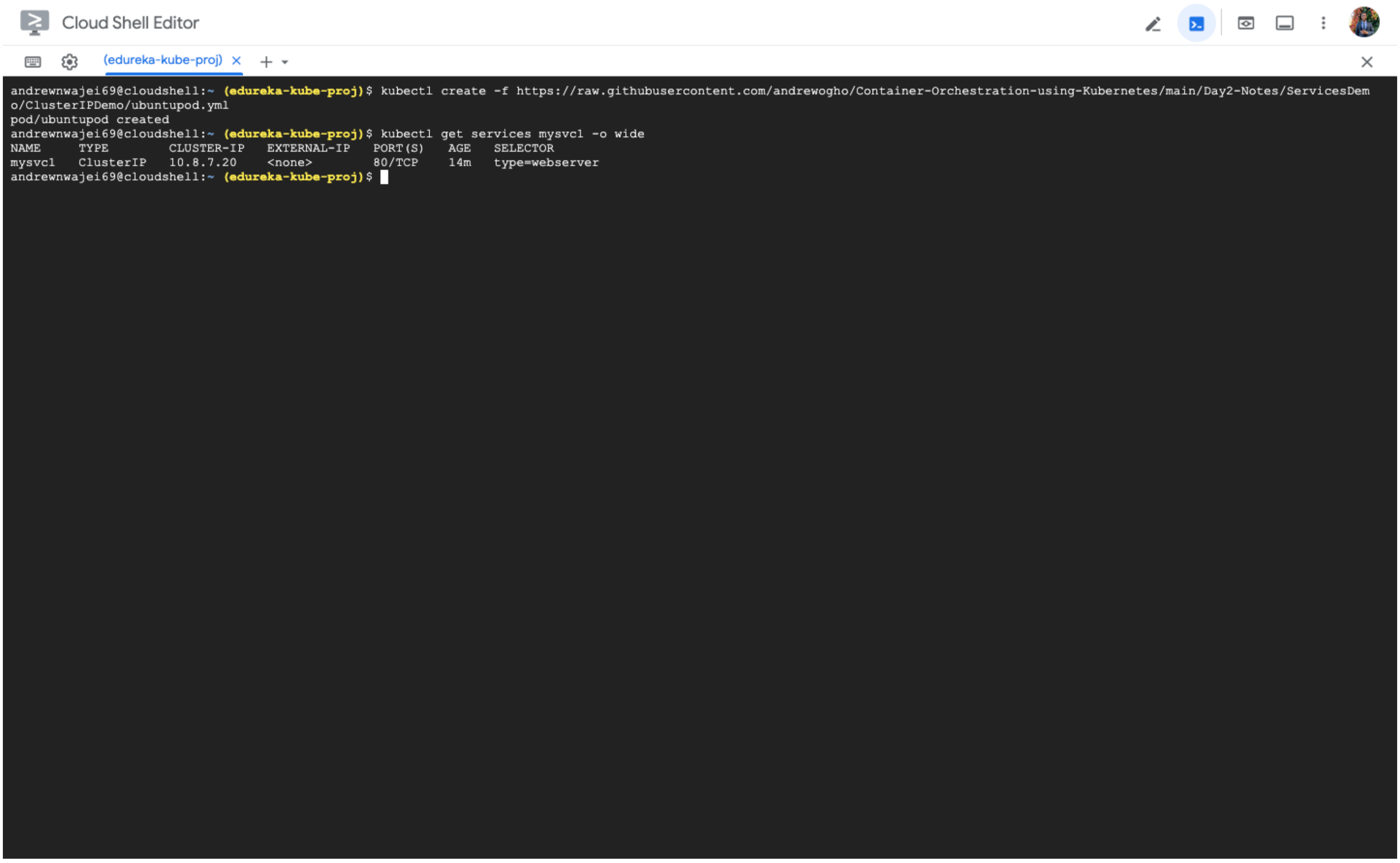


```
andrewnwajei69@cloudshell:~ (edureka-kube-proj)$ kubectl create -f https://raw.githubusercontent.com/andrewogho/Container-Orchestration-using-Kubernetes/main/Day2-Notes/ServicesDem
o/ClusterIPDemo/Service1.yml
service/mysvc1 created
andrewnwajei69@cloudshell:~ (edureka-kube-proj)$ kubectl get service
NAME         TYPE        CLUSTER-IP   EXTERNAL-IP   PORT(S)   AGE
kubernetes   ClusterIP   10.8.0.1     <none>        443/TCP   23m
mysvc1       ClusterIP   10.8.7.20    <none>        80/TCP    3m15s
andrewnwajei69@cloudshell:~ (edureka-kube-proj)$ kubectl describe service mysvc1
Name:              mysvc1
Namespace:         default
Labels:            <none>
Annotations:       cloud.google.com/neg: {"ingress":true}
Selector:          type=webserver
Type:              ClusterIP
IP Family Policy:  SingleStack
IP Families:       IPv4
IP:                10.8.7.20
IPs:               10.8.7.20
Port:              <unset>  80/TCP
TargetPort:        80/TCP
Endpoints:         10.4.0.6:80,10.4.1.7:80,10.4.2.4:80
Session Affinity:  None
Events:            <none>
andrewnwajei69@cloudshell:~ (edureka-kube-proj)$ kubectl describe service mysvc1 | less
andrewnwajei69@cloudshell:~ (edureka-kube-proj)$
```

10. Using this raw yaml file to create a database pod -ubuntu. To use to access the nginx replicaset created.

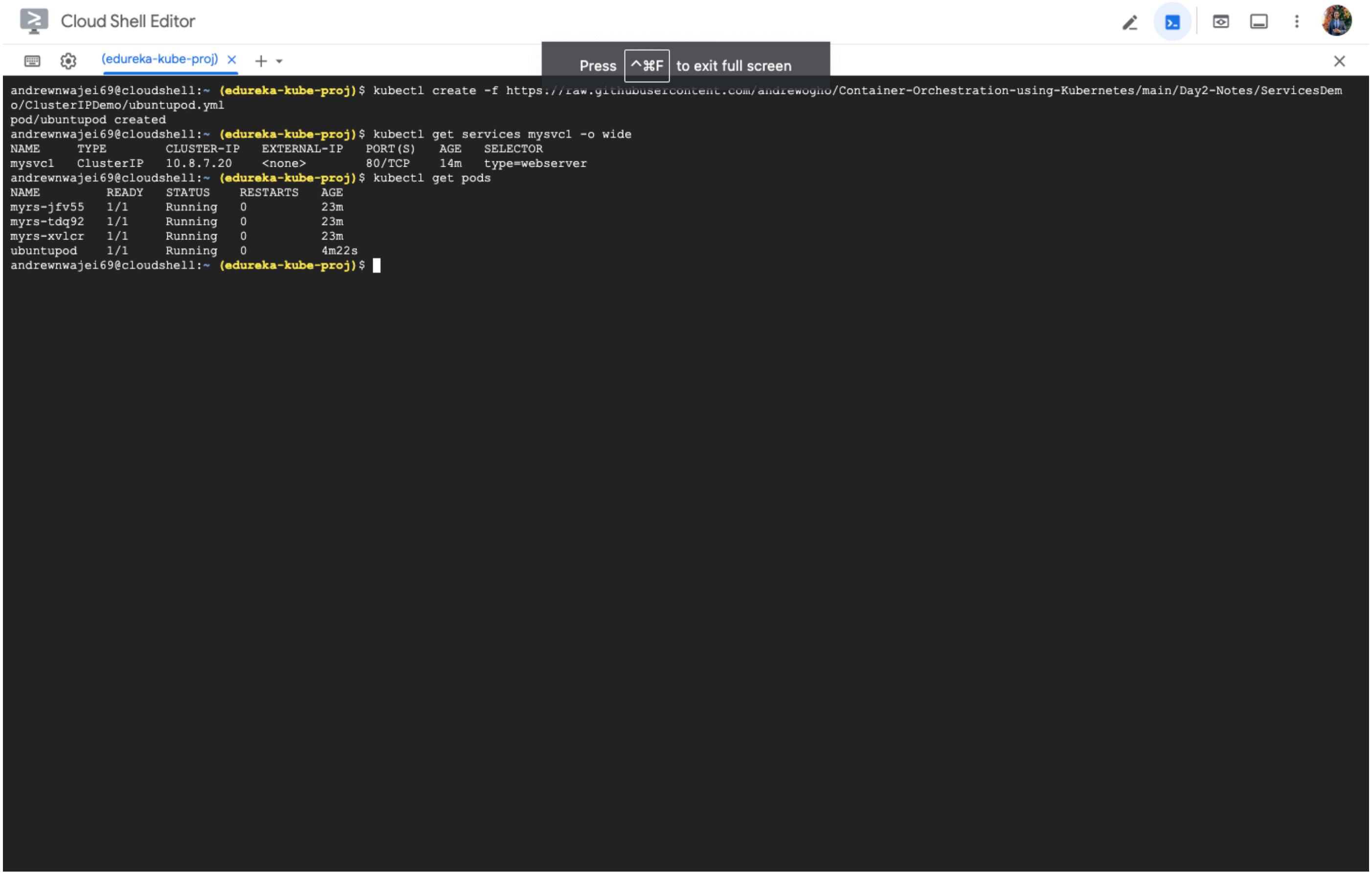


```
apiVersion: v1
kind: Pod
metadata:
 name: ubuntupod
 labels:
  role: dev
spec:
 containers:
  - name: c1
    image: ubuntu
    args: [/bin/bash, -c, 'sleep 6000']
```



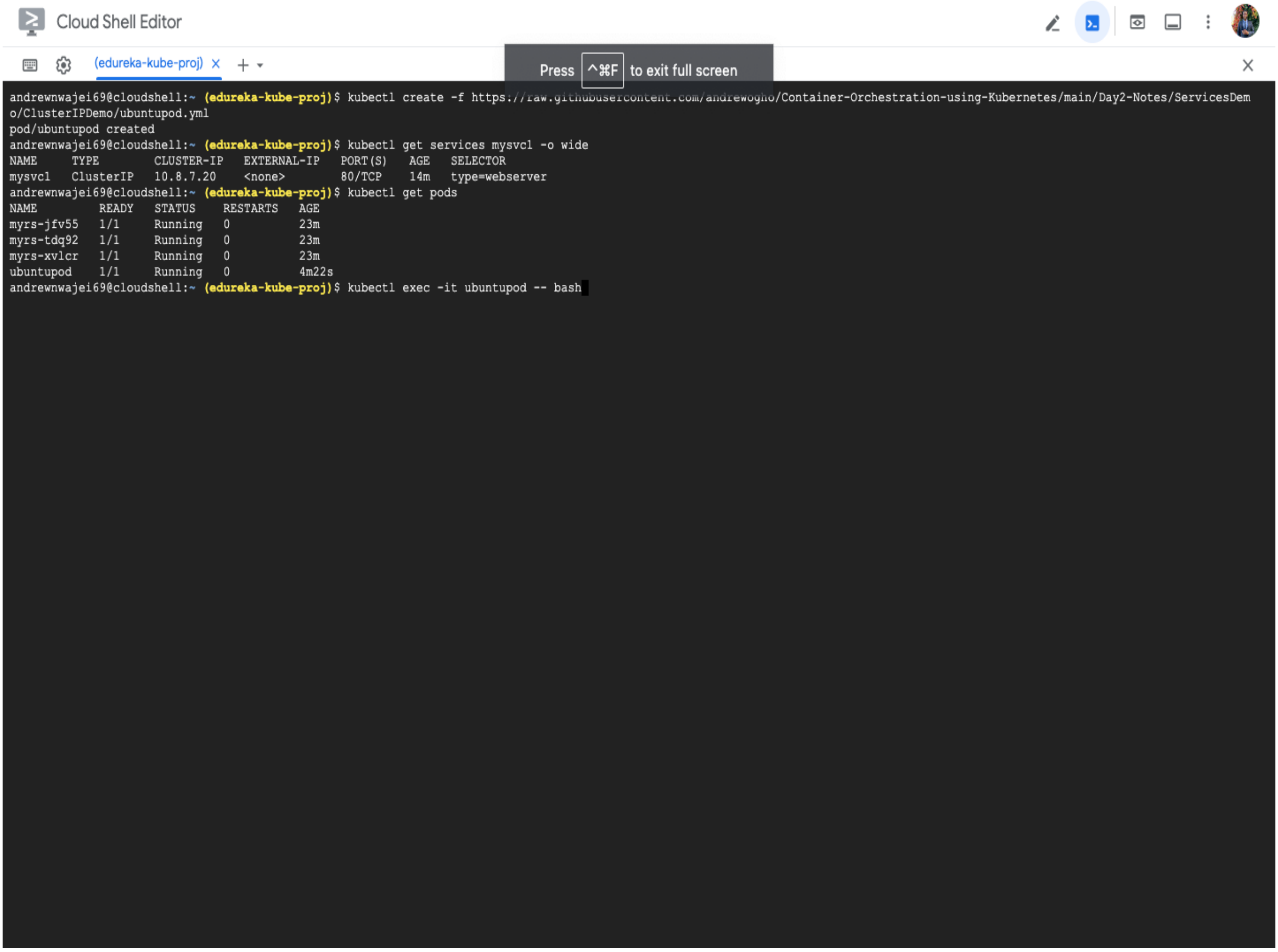
```
andrewnwajei69@cloudshell:~ (edureka-kube-proj)$ kubectl create -f https://raw.githubusercontent.com/andrewogho/Container-Orchestration-using-Kubernetes/main/Day2-Notes/ServicesDem
o/ClusterIPDemo/ubuntupod.yml
pod/ubuntupod created
andrewnwajei69@cloudshell:~ (edureka-kube-proj)$ kubectl get services mysvc1 -o wide
NAME     TYPE        CLUSTER-IP   EXTERNAL-IP   PORT(S)   AGE   SELECTOR
mysvc1   ClusterIP   10.8.7.20    <none>        80/TCP    14m   type=webserver
andrewnwajei69@cloudshell:~ (edureka-kube-proj)$
```

11. Command to show new pod ubuntu has been created.



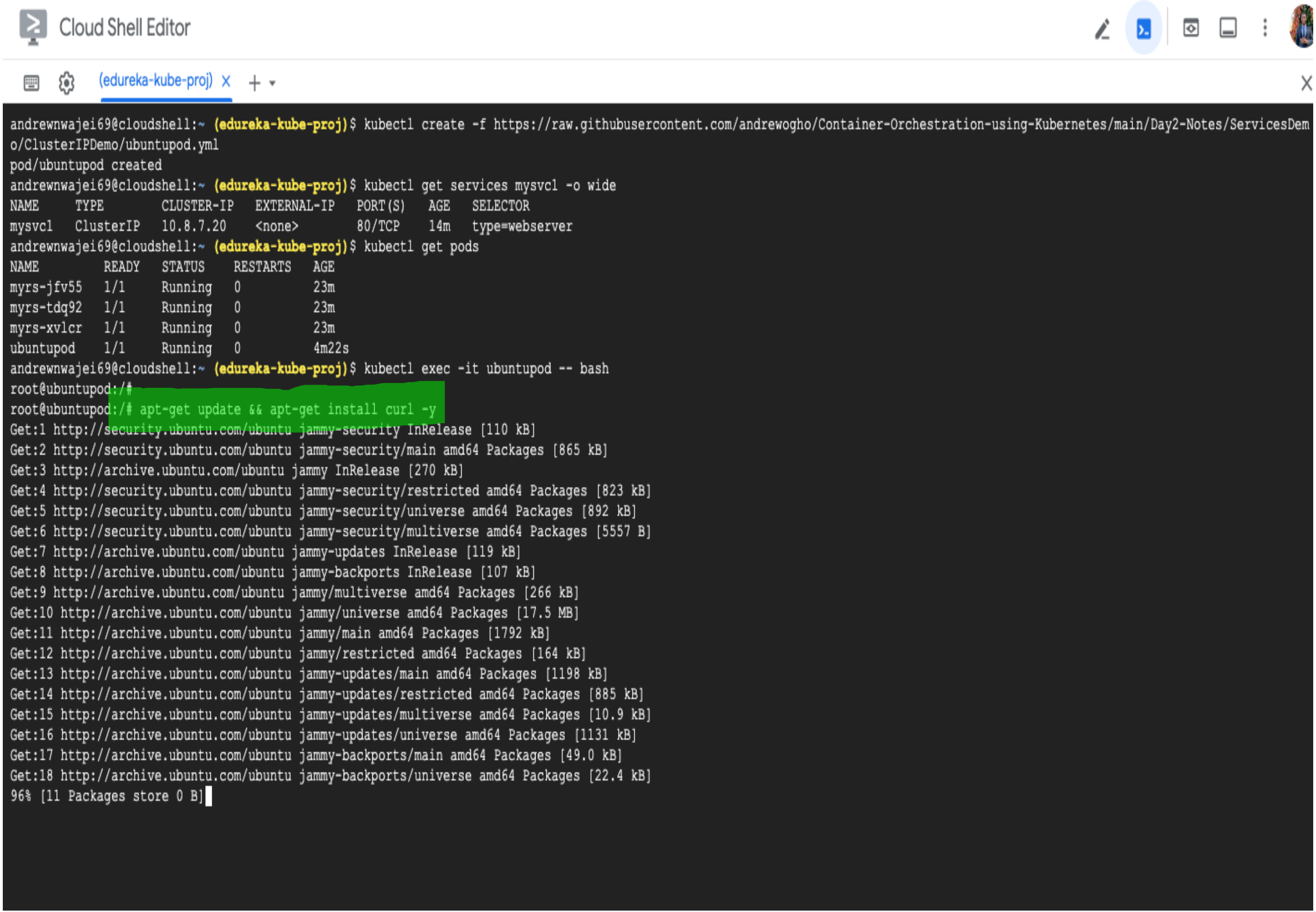
```
andrewnwajei69@cloudshell:~ (edureka-kube-proj)$ kubectl create -f https://raw.githubusercontent.com/andrewogho/Container-Orchestration-using-Kubernetes/main/Day2-Notes/ServicesDem
o/ClusterIPDemo/ubuntupod.yml
pod/ubuntupod created
andrewnwajei69@cloudshell:~ (edureka-kube-proj)$ kubectl get services mysvc1 -o wide
NAME     TYPE        CLUSTER-IP   EXTERNAL-IP   PORT(S)   AGE   SELECTOR
mysvc1   ClusterIP   10.8.7.20    <none>        80/TCP    14m   type=webserver
andrewnwajei69@cloudshell:~ (edureka-kube-proj)$ kubectl get pods
NAME          READY   STATUS    RESTARTS   AGE
myrs-jfv55    1/1     Running   0          23m
myrs-tdq92    1/1     Running   0          23m
myrs-xvlcr    1/1     Running   0          23m
ubuntupod     1/1     Running   0          4m22s
andrewnwajei69@cloudshell:~ (edureka-kube-proj)$ 
```

12. Using the kubectl exec command to enter the ubuntu pod CLI.

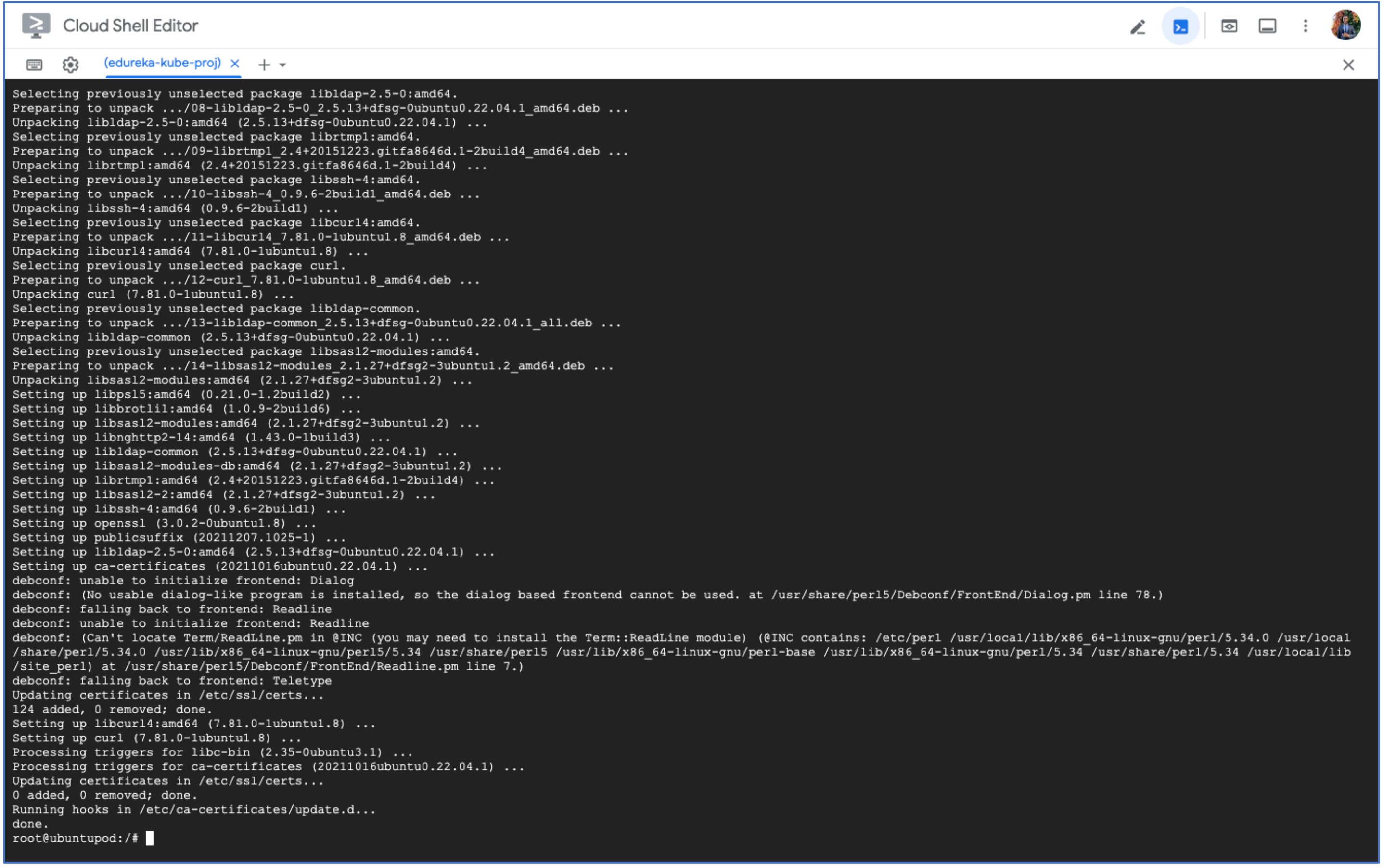


```
andrewnwajei69@cloudshell:~ (edureka-kube-proj)$ kubectl create -f https://raw.githubusercontent.com/andrewogho/Container-Orchestration-using-Kubernetes/main/Day2-Notes/ServicesDem
o/ClusterIPDemo/ubuntupod.yml
pod/ubuntupod created
andrewnwajei69@cloudshell:~ (edureka-kube-proj)$ kubectl get services mysvcl -o wide
NAME     TYPE        CLUSTER-IP   EXTERNAL-IP   PORT(S)   AGE   SELECTOR
mysvcl   ClusterIP   10.8.7.20    <none>        80/TCP    14m   type=webserver
andrewnwajei69@cloudshell:~ (edureka-kube-proj)$ kubectl get pods
NAME          READY   STATUS    RESTARTS   AGE
myrs-jfv55    1/1     Running   0          23m
myrs-tdq92    1/1     Running   0          23m
myrs-xvlcr    1/1     Running   0          23m
ubuntupod     1/1     Running   0          4m22s
andrewnwajei69@cloudshell:~ (edureka-kube-proj)$ kubectl exec -it ubuntupod -- bash
```

13. In the ubuntu pod: use the apt-get command to update & install curl to be able to ping the replicaset clusterIP.

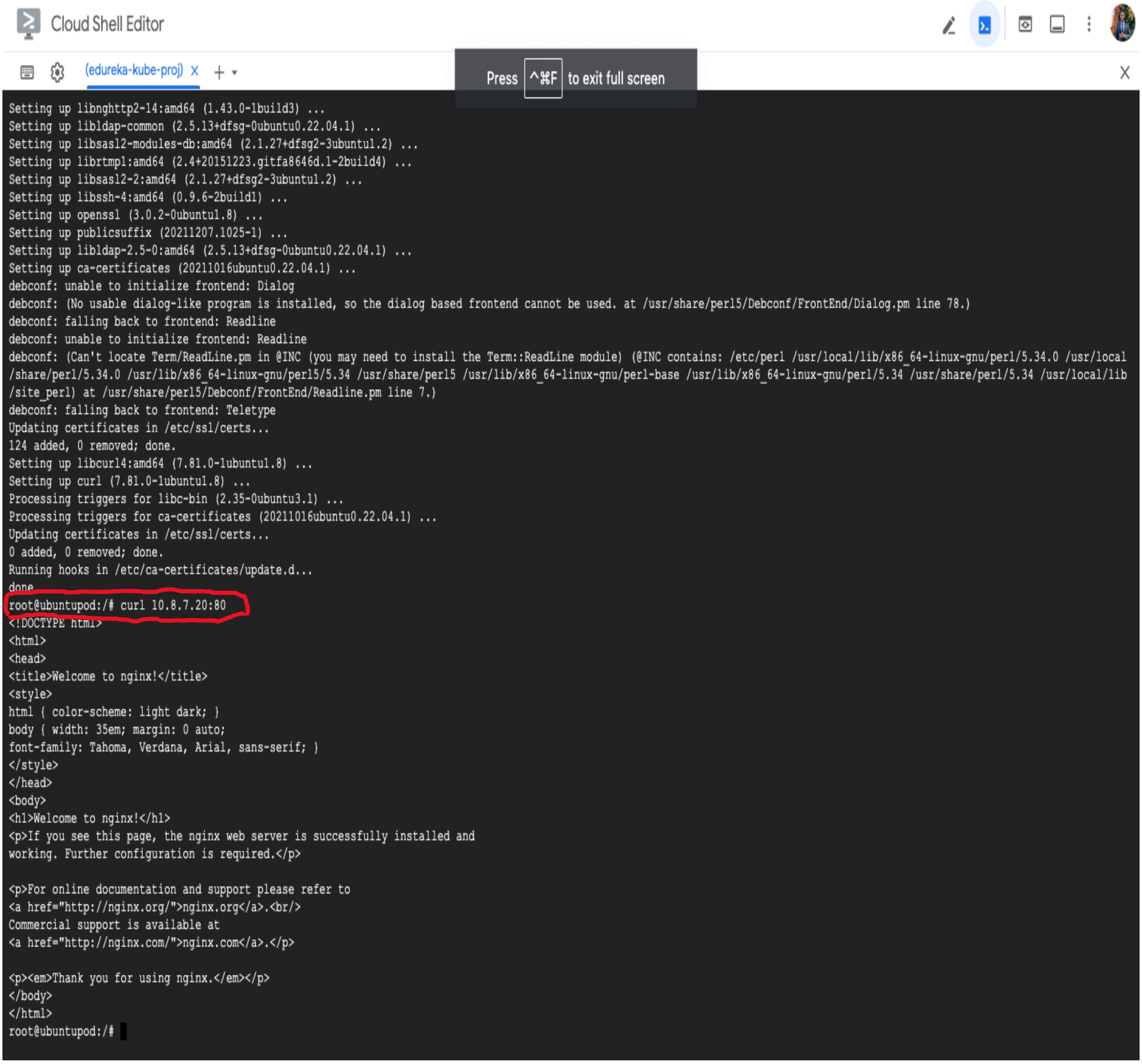


```
andrewnwajei69@cloudshell:~ (edureka-kube-proj)$ kubectl create -f https://raw.githubusercontent.com/andrewogho/Container-Orchestration-using-Kubernetes/main/Day2-Notes/ServicesDem
o/ClusterIPDemo/ubuntupod.yml
pod/ubuntupod created
andrewnwajei69@cloudshell:~ (edureka-kube-proj)$ kubectl get services mysvcl -o wide
NAME     TYPE        CLUSTER-IP   EXTERNAL-IP   PORT(S)   AGE   SELECTOR
mysvcl   ClusterIP   10.8.7.20    <none>        80/TCP    14m   type=webserver
andrewnwajei69@cloudshell:~ (edureka-kube-proj)$ kubectl get pods
NAME         READY   STATUS    RESTARTS   AGE
myrs-jfv55   1/1     Running   0          23m
myrs-tdq92   1/1     Running   0          23m
myrs-xvlcr   1/1     Running   0          23m
ubuntupod    1/1     Running   0          4m22s
andrewnwajei69@cloudshell:~ (edureka-kube-proj)$ kubectl exec -it ubuntupod -- bash
root@ubuntupod:/#
root@ubuntupod:/# apt-get update && apt-get install curl -y
Get:1 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
Get:2 http://security.ubuntu.com/ubuntu jammy-security/main amd64 Packages [865 kB]
Get:3 http://archive.ubuntu.com/ubuntu jammy InRelease [270 kB]
Get:4 http://security.ubuntu.com/ubuntu jammy-security/restricted amd64 Packages [823 kB]
Get:5 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 Packages [892 kB]
Get:6 http://security.ubuntu.com/ubuntu jammy-security/multiverse amd64 Packages [5557 B]
Get:7 http://archive.ubuntu.com/ubuntu jammy-updates InRelease [119 kB]
Get:8 http://archive.ubuntu.com/ubuntu jammy-backports InRelease [107 kB]
Get:9 http://archive.ubuntu.com/ubuntu jammy/multiverse amd64 Packages [266 kB]
Get:10 http://archive.ubuntu.com/ubuntu jammy/universe amd64 Packages [17.5 MB]
Get:11 http://archive.ubuntu.com/ubuntu jammy/main amd64 Packages [1792 kB]
Get:12 http://archive.ubuntu.com/ubuntu jammy/restricted amd64 Packages [164 kB]
Get:13 http://archive.ubuntu.com/ubuntu jammy-updates/main amd64 Packages [1198 kB]
Get:14 http://archive.ubuntu.com/ubuntu jammy-updates/restricted amd64 Packages [885 kB]
Get:15 http://archive.ubuntu.com/ubuntu jammy-updates/multiverse amd64 Packages [10.9 kB]
Get:16 http://archive.ubuntu.com/ubuntu jammy-updates/universe amd64 Packages [1131 kB]
Get:17 http://archive.ubuntu.com/ubuntu jammy-backports/main amd64 Packages [49.0 kB]
Get:18 http://archive.ubuntu.com/ubuntu jammy-backports/universe amd64 Packages [22.4 kB]
96% [11 Packages store 0 B]
```



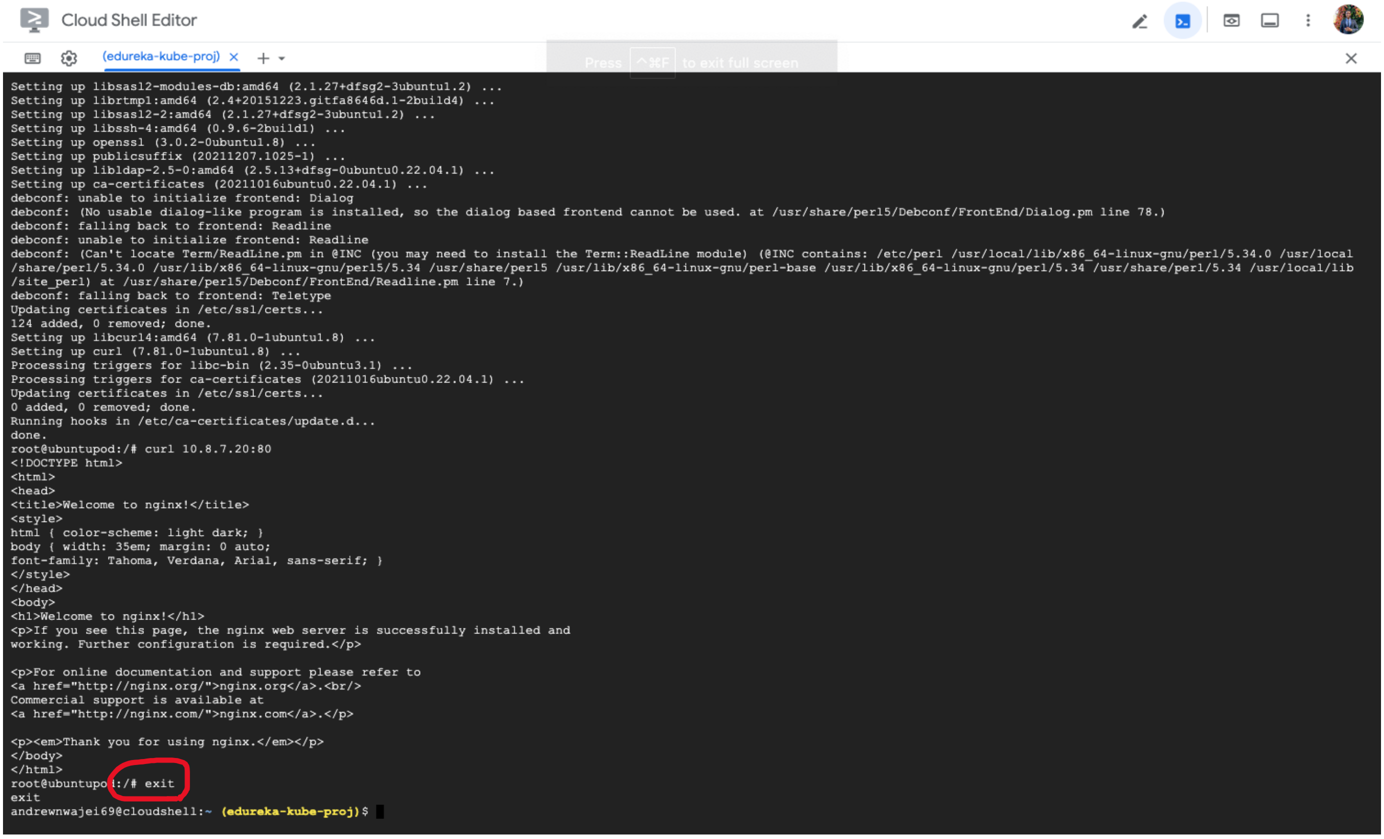
```
Selecting previously unselected package libldap-2.5-0:amd64.
Preparing to unpack .../08-libldap-2.5-0_2.5.13+dfsg-0ubuntu0.22.04.1_amd64.deb ...
Unpacking libldap-2.5-0:amd64 (2.5.13+dfsg-0ubuntu0.22.04.1) ...
Selecting previously unselected package librtmp1:amd64.
Preparing to unpack .../09-librtmp1_2.4+20151223.gitfa8646d.1-2build4_amd64.deb ...
Unpacking librtmp1:amd64 (2.4+20151223.gitfa8646d.1-2build4) ...
Selecting previously unselected package libssh-4:amd64.
Preparing to unpack .../10-libssh-4_0.9.6-2build1_amd64.deb ...
Unpacking libssh-4:amd64 (0.9.6-2build1) ...
Selecting previously unselected package libcurl4:amd64.
Preparing to unpack .../11-libcurl4_7.81.0-1ubuntu1.8_amd64.deb ...
Unpacking libcurl4:amd64 (7.81.0-1ubuntu1.8) ...
Selecting previously unselected package curl.
Preparing to unpack .../12-curl_7.81.0-1ubuntu1.8_amd64.deb ...
Unpacking curl (7.81.0-1ubuntu1.8) ...
Selecting previously unselected package libldap-common.
Preparing to unpack .../13-libldap-common_2.5.13+dfsg-0ubuntu0.22.04.1_all.deb ...
Unpacking libldap-common (2.5.13+dfsg-0ubuntu0.22.04.1) ...
Selecting previously unselected package libsasl2-modules:amd64.
Preparing to unpack .../14-libsasl2-modules_2.1.27+dfsg2-3ubuntu1.2_amd64.deb ...
Unpacking libsasl2-modules:amd64 (2.1.27+dfsg2-3ubuntu1.2) ...
Setting up libpsl5:amd64 (0.21.0-1.2build2) ...
Setting up libbrotli1:amd64 (1.0.9-2build6) ...
Setting up libsasl2-modules:amd64 (2.1.27+dfsg2-3ubuntu1.2) ...
Setting up libnghttp2-14:amd64 (1.43.0-1build3) ...
Setting up libldap-common (2.5.13+dfsg-0ubuntu0.22.04.1) ...
Setting up libsasl2-modules-db:amd64 (2.1.27+dfsg2-3ubuntu1.2) ...
Setting up librtmp1:amd64 (2.4+20151223.gitfa8646d.1-2build4) ...
Setting up libsasl2-2:amd64 (2.1.27+dfsg2-3ubuntu1.2) ...
Setting up libssh-4:amd64 (0.9.6-2build1) ...
Setting up openssl (3.0.2-0ubuntu1.8) ...
Setting up publicsuffix (20211207.1025-1) ...
Setting up libldap-2.5-0:amd64 (2.5.13+dfsg-0ubuntu0.22.04.1) ...
Setting up ca-certificates (20211016ubuntu0.22.04.1) ...
debconf: unable to initialize frontend: Dialog
debconf: (No usable dialog-like program is installed, so the dialog based frontend cannot be used. at /usr/share/perl5/Debconf/FrontEnd/Dialog.pm line 78.)
debconf: falling back to frontend: Readline
debconf: unable to initialize frontend: Readline
debconf: (Can't locate Term/ReadLine.pm in @INC (you may need to install the Term::ReadLine module) (@INC contains: /etc/perl /usr/local/lib/x86_64-linux-gnu/perl/5.34.0 /usr/local
/share/perl/5.34.0 /usr/lib/x86_64-linux-gnu/perl5/5.34 /usr/share/perl5 /usr/lib/x86_64-linux-gnu/perl-base /usr/lib/x86_64-linux-gnu/perl/5.34 /usr/share/perl/5.34 /usr/local/lib
/site_perl) at /usr/share/perl5/Debconf/FrontEnd/Readline.pm line 7.)
debconf: falling back to frontend: Teletype
Updating certificates in /etc/ssl/certs...
124 added, 0 removed; done.
Setting up libcurl4:amd64 (7.81.0-1ubuntu1.8) ...
Setting up curl (7.81.0-1ubuntu1.8) ...
Processing triggers for libc-bin (2.35-0ubuntu3.1) ...
Processing triggers for ca-certificates (20211016ubuntu0.22.04.1) ...
Updating certificates in /etc/ssl/certs...
0 added, 0 removed; done.
Running hooks in /etc/ca-certificates/update.d...
done.
root@ubuntupod:/#
```

14. Using the “curl” command ping the ClusterIP address and the nginx pod port number to access the file with the nginx pod. i.e: ClusterIP:portnumber.



```
Setting up libnghttp2-14:amd64 (1.43.0-1build3) ...
Setting up libldap-common (2.5.13+dfsg-0ubuntu0.22.04.1) ...
Setting up libsasl2-modules-db:amd64 (2.1.27+dfsg2-3ubuntu1.2) ...
Setting up librtmp1:amd64 (2.4+20151223.gitfa8646d.1-2build4) ...
Setting up libsasl2-2:amd64 (2.1.27+dfsg2-3ubuntu1.2) ...
Setting up libssh-4:amd64 (0.9.6-2build1) ...
Setting up openssl (3.0.2-0ubuntu1.8) ...
Setting up publicsuffix (20211207.1025-1) ...
Setting up libldap-2.5-0:amd64 (2.5.13+dfsg-0ubuntu0.22.04.1) ...
Setting up ca-certificates (20211016ubuntu0.22.04.1) ...
debconf: unable to initialize frontend: Dialog
debconf: (No usable dialog-like program is installed, so the dialog based frontend cannot be used. at /usr/share/perl5/Debconf/FrontEnd/Dialog.pm line 78.)
debconf: falling back to frontend: Readline
debconf: unable to initialize frontend: Readline
debconf: (Can't locate Term/ReadLine.pm in @INC (you may need to install the Term::ReadLine module) (@INC contains: /etc/perl /usr/local/lib/x86_64-linux-gnu/perl/5.34.0 /usr/local
/share/perl/5.34.0 /usr/lib/x86_64-linux-gnu/perl5/5.34 /usr/share/perl5 /usr/lib/x86_64-linux-gnu/perl-base /usr/lib/x86_64-linux-gnu/perl/5.34 /usr/share/perl/5.34 /usr/local/lib
/site_perl) at /usr/share/perl5/Debconf/FrontEnd/Readline.pm line 7.)
debconf: falling back to frontend: Teletype
Updating certificates in /etc/ssl/certs...
124 added, 0 removed; done.
Setting up libcurl4:amd64 (7.81.0-1ubuntu1.8) ...
Setting up curl (7.81.0-1ubuntu1.8) ...
Processing triggers for libc-bin (2.35-0ubuntu3.1) ...
Processing triggers for ca-certificates (20211016ubuntu0.22.04.1) ...
Updating certificates in /etc/ssl/certs...
0 added, 0 removed; done.
Running hooks in /etc/ca-certificates/update.d...
done.
root@ubuntupod:/# curl 10.8.7.20:80
<!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>
root@ubuntupod:/#
```

15. Type “exit” to come out of ubuntu pod.



```
Setting up libsasl2-modules-db:amd64 (2.1.27+dfsg2-3ubuntu1.2) ...
Setting up librtmp1:amd64 (2.4+20151223.gitfa8646d.1-2build4) ...
Setting up libsasl2-2:amd64 (2.1.27+dfsg2-3ubuntu1.2) ...
Setting up libssh-4:amd64 (0.9.6-2build1) ...
Setting up openssl (3.0.2-0ubuntu1.8) ...
Setting up publicsuffix (20211207.1025-1) ...
Setting up libldap-2.5-0:amd64 (2.5.13+dfsg-0ubuntu0.22.04.1) ...
Setting up ca-certificates (20211016ubuntu0.22.04.1) ...
debconf: unable to initialize frontend: Dialog
debconf: (No usable dialog-like program is installed, so the dialog based frontend cannot be used. at /usr/share/perl5/Debconf/FrontEnd/Dialog.pm line 78.)
debconf: falling back to frontend: Readline
debconf: unable to initialize frontend: Readline
debconf: (Can't locate Term/ReadLine.pm in @INC (you may need to install the Term::ReadLine module) (@INC contains: /etc/perl /usr/local/lib/x86_64-linux-gnu/perl/5.34.0 /usr/local
/share/perl/5.34.0 /usr/lib/x86_64-linux-gnu/perl5/5.34 /usr/share/perl5 /usr/lib/x86_64-linux-gnu/perl-base /usr/lib/x86_64-linux-gnu/perl/5.34 /usr/share/perl/5.34 /usr/local/lib
/site_perl) at /usr/share/perl5/Debconf/FrontEnd/Readline.pm line 7.)
debconf: falling back to frontend: Teletype
Updating certificates in /etc/ssl/certs...
124 added, 0 removed; done.
Setting up libcurl4:amd64 (7.81.0-1ubuntu1.8) ...
Setting up curl (7.81.0-1ubuntu1.8) ...
Processing triggers for libc-bin (2.35-0ubuntu3.1) ...
Processing triggers for ca-certificates (20211016ubuntu0.22.04.1) ...
Updating certificates in /etc/ssl/certs...
0 added, 0 removed; done.
Running hooks in /etc/ca-certificates/update.d...
done.
root@ubuntupod:/# curl 10.8.7.20:80
<!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>
root@ubuntupod:/# exit
exit
andrewnwajei69@cloudshell:~ (edureka-kube-proj)$
```

Done.

That is how to create a Service Object for type ClusterIP which allows pods within a cluster access each other using its ClusterIP created by giving it a service using the yaml file and the pods port number.