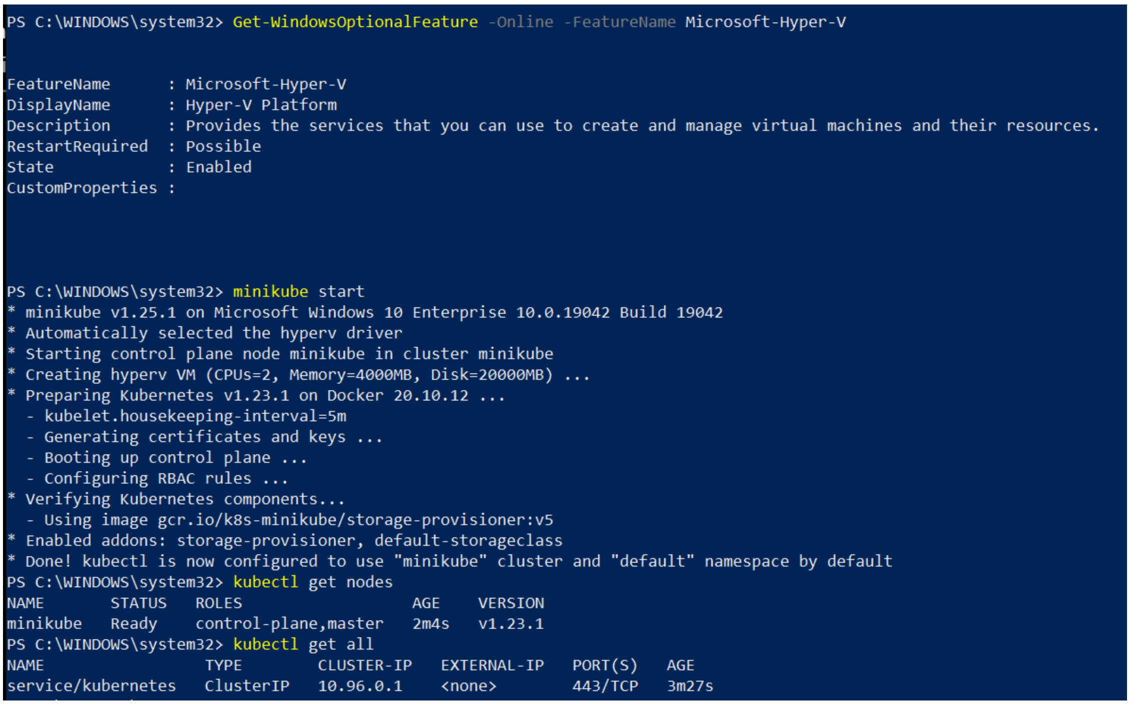
Running First Application on Minikube

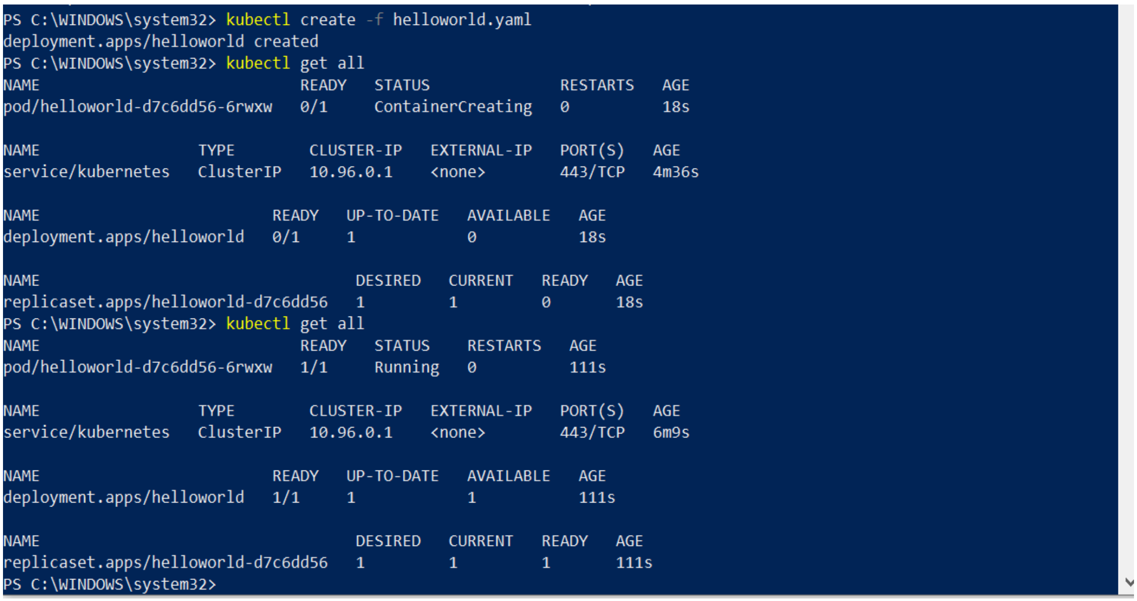
Part 1: Start a Minikube



```
PS C:\WINDOWS\system32> Get-WindowsOptionalFeature -Online -FeatureName Microsoft-Hyper-V


FeatureName      : Microsoft-Hyper-V
DisplayName      : Hyper-V Platform
Description      : Provides the services that you can use to create and manage virtual machines and their resources.
RestartRequired  : Possible
State            : Enabled
CustomProperties :




PS C:\WINDOWS\system32> minikube start
* minikube v1.25.1 on Microsoft Windows 10 Enterprise 10.0.19042 Build 19042
* Automatically selected the hyperv driver
* Starting control plane node minikube in cluster minikube
* Creating hyperv VM (CPUs=2, Memory=4000MB, Disk=20000MB) ...
* Preparing Kubernetes v1.23.1 on Docker 20.10.12 ...
  - kubelet.housekeeping-interval=5m
  - Generating certificates and keys ...
  - Booting up control plane ...
  - Configuring RBAC rules ...
* Verifying Kubernetes components...
  - Using image gcr.io/k8s-minikube/storage-provisioner:v5
* Enabled addons: storage-provisioner, default-storageclass
* Done! kubectl is now configured to use "minikube" cluster and "default" namespace by default
PS C:\WINDOWS\system32> kubectl get nodes
NAME       STATUS   ROLES                  AGE    VERSION
minikube   Ready    control-plane,master   2m4s   v1.23.1
PS C:\WINDOWS\system32> kubectl get all
NAME                 TYPE        CLUSTER-IP   EXTERNAL-IP   PORT(S)   AGE
service/kubernetes   ClusterIP   10.96.0.1    <none>        443/TCP   3m27s
```

1. ensure hyper-V is enabled using: Get-WindowsOptionalFeature -Online -FeatureName Microsoft-Hyper-V
    a. if not enable with command: Enable-WindowsOptionalFeature -Online -FeatureName Microsoft-Hyper-V -All
2. Start a MiniKube with command: minikube start
3. Ensure the Minikube is active using: kubectl get nodes
4. Check whats in the cluster using: kubectl get all

Part 2: Deploy application into cluster

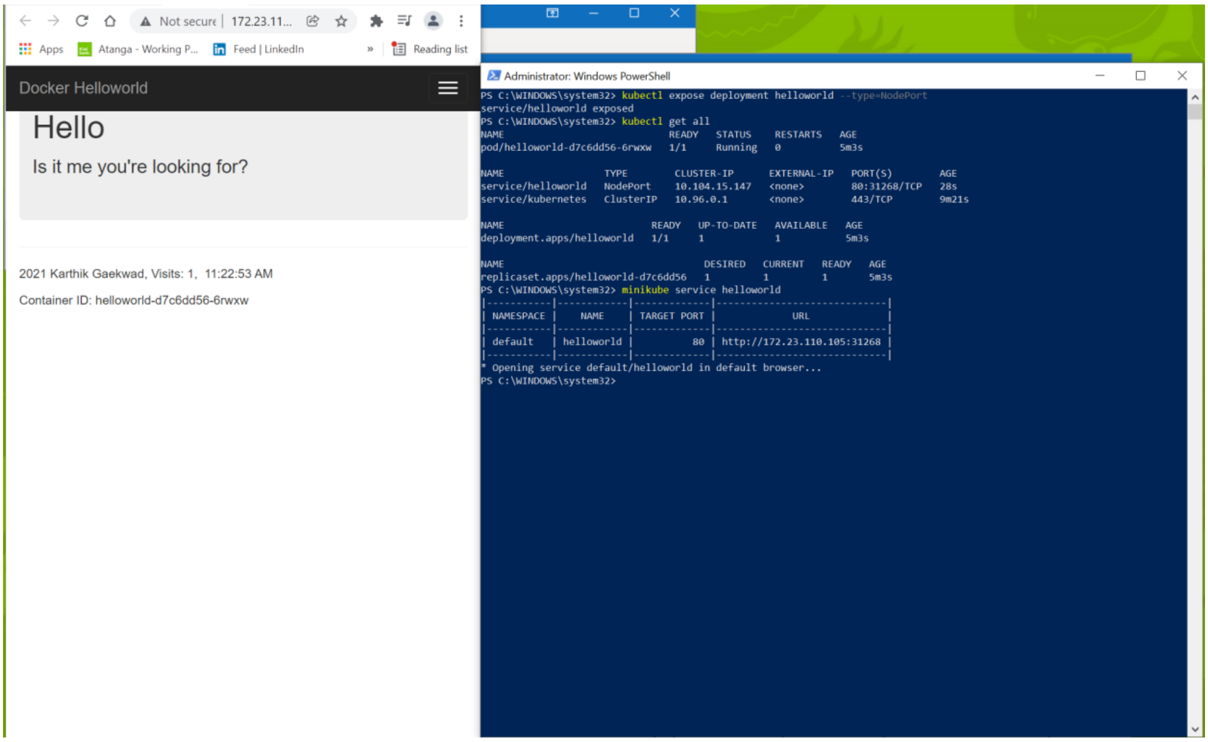


```
PS C:\WINDOWS\system32> kubectl create -f helloworld.yaml
deployment.apps/helloworld created
PS C:\WINDOWS\system32> kubectl get all
NAME                            READY   STATUS              RESTARTS   AGE
pod/helloworld-d7c6dd56-6rwxw   0/1     ContainerCreating   0          18s

NAME                 TYPE        CLUSTER-IP   EXTERNAL-IP   PORT(S)   AGE
service/kubernetes   ClusterIP   10.96.0.1    <none>        443/TCP   4m36s

NAME                         READY   UP-TO-DATE   AVAILABLE   AGE
deployment.apps/helloworld   0/1     1            0           18s

NAME                                   DESIRED   CURRENT   READY   AGE
replicaset.apps/helloworld-d7c6dd56   1         1         0       18s
PS C:\WINDOWS\system32> kubectl get all
NAME                            READY   STATUS    RESTARTS   AGE
pod/helloworld-d7c6dd56-6rwxw   1/1     Running   0          111s

NAME                 TYPE        CLUSTER-IP   EXTERNAL-IP   PORT(S)   AGE
service/kubernetes   ClusterIP   10.96.0.1    <none>        443/TCP   6m9s

NAME                         READY   UP-TO-DATE   AVAILABLE   AGE
deployment.apps/helloworld   1/1     1            1           111s

NAME                                   DESIRED   CURRENT   READY   AGE
replicaset.apps/helloworld-d7c6dd56   1         1         1       111s
PS C:\WINDOWS\system32>
```

1. To deploy an application use command: kubectl create -f [applicationname.type]

2. Check that the application has been added to the cluster using: kubectl get all

PART 3: Running an application



Docker Helloworld

Hello

Is it me you're looking for?

2021 Karthik Gaekwad, Visits: 1, 11:22:53 AM

Container ID: helloworld-d7c6dd56-6rwxw

```
PS C:\WINDOWS\system32> kubectl expose deployment helloworld --type=NodePort
service/helloworld exposed
PS C:\WINDOWS\system32> kubectl get all
NAME                            READY   STATUS    RESTARTS   AGE
pod/helloworld-d7c6dd56-6rwxw   1/1     Running   0          5m3s

NAME                  TYPE        CLUSTER-IP     EXTERNAL-IP   PORT(S)        AGE
service/helloworld    NodePort    10.104.15.147  <none>        80:31268/TCP   28s
service/kubernetes    ClusterIP   10.96.0.1      <none>        443/TCP        9m21s

NAME                         READY   UP-TO-DATE   AVAILABLE   AGE
deployment.apps/helloworld   1/1     1            1           5m3s

NAME                                   DESIRED   CURRENT   READY   AGE
replicaset.apps/helloworld-d7c6dd56    1         1         1       5m3s
PS C:\WINDOWS\system32> minikube service helloworld
|-----------|------------|-------------|-----------------------------|
| NAMESPACE |    NAME    | TARGET PORT |             URL             |
|-----------|------------|-------------|-----------------------------|
| default   | helloworld |          80 | http://172.23.110.105:31268 |
|-----------|------------|-------------|-----------------------------|
* Opening service default/helloworld in default browser...
PS C:\WINDOWS\system32>
```

1. Create service construct for Kubernetes and expose deployment as a service: kubectl expose deployment [applicationName] –type=NodePort
2. Check that the service was added to the cluster using: kubectl get all
3. Access the webservice using: minikube service [applicationName]