

# Learn Kubernetes Basics

## Lab 3

Using the tutorials

- Deploy a containerized application on a cluster.
- Scale the deployment.
- Update the containerized application with a new software version.
- Debug the containerized application.

# Module 1 – Create Kubernetes cluster

## Installing minikube

```
Windows PowerShell
[Empire:PowerShell]
copyright (C) Microsoft Corporation. All rights reserved.

writing web request
writing request stream... (Number of bytes written: 2970886)

Directory: C:\

File                LastWriteTime         Length Name
----                -
17/03/2025    18:19             4096 minikube

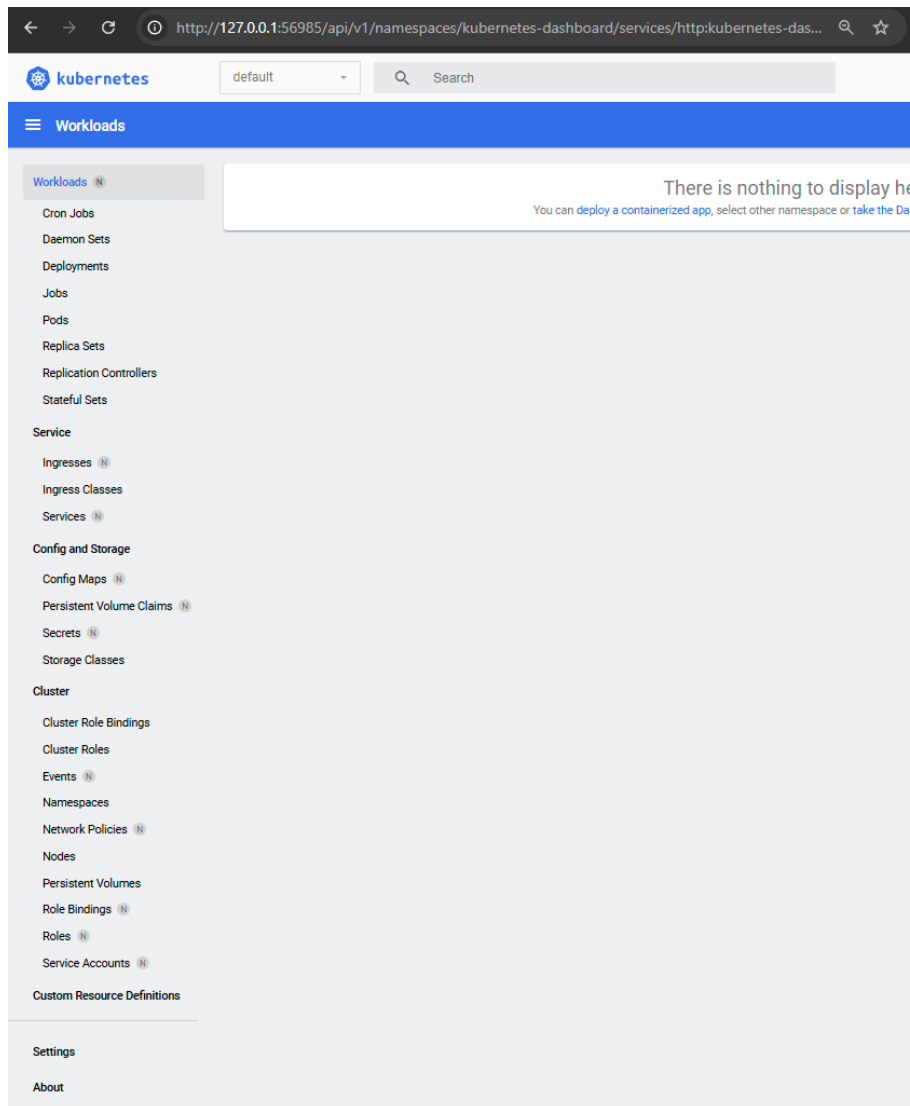
PS C:\Users\User> Invoke-WebRequest -OutFile "c:\minikube\minikube.exe" -Uri "https://github.com/kubernetes/minikube/releases/download/v1.35.0/minikube-windows-amd64.exe" -UseBasicParsing
```

```
PS C:\WINDOWS\system32> minikube version
minikube version: v1.35.0
commit: dd5d320e41b5451cdf3c01891bc4e13d189586ed-dirty
PS C:\WINDOWS\system32> kubectl version --client
Client Version: v1.31.4
Kustomize Version: v5.4.2
```

## Installed and running with docker

```
PS C:\WINDOWS\system32> minikube start --driver=docker
* minikube v1.35.0 on Microsoft Windows 10 Pro 10.0.19045.5608 Build 19045.5608
* Using the docker driver based on user configuration
* Using Docker Desktop driver with root privileges
* Starting "minikube" primary control-plane node in "minikube" cluster
* Pulling base image v0.0.46 ...
  > gcr.io/k8s-minikube/kicbase...: 500.31 MiB / 500.31 MiB 100.00% 5.98 Mi
* Creating docker container (CPUs=2, Memory=8100MB) ...
! Failing to connect to https://registry.k8s.io/ from inside the minikube container
* To pull new external images, you may need to configure a proxy: https://minikube.sigs.k8s.io/docs/reference/networking/proxy/
* Preparing Kubernetes v1.32.0 on Docker 27.4.1 ...
  - Generating certificates and keys ...
  - Booting up control plane ...
  - Configuring RBAC rules ...
* Configuring bridge CNI (Container Networking Interface) ...
* 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>
PS C:\WINDOWS\system32> minikube status
minikube
type: Control Plane
host: Running
kubelet: Running
apiserver: Running
kubeconfig: Configured
PS C:\WINDOWS\system32>
PS C:\WINDOWS\system32>
```

## Kubernetes dashboard running locally



## Module 2: Deploy an App

### Create a Deployment

Create a deployment for an Nginx app:

```
kubectl create deployment nginx --image=nginx
```

```
kubectl create deployment kubernetes-bootcamp --image=gcr.io/google-samples/kubernetes-bootcamp:v1
```

### Verify deployments

```
get deployments
```

```
PS C:\WINDOWS\system32> kubectl create deployment kubernetes-bootcamp --image=gcr.io/google-samples/kubernetes-bootcamp:v1
deployment.apps/kubernetes-bootcamp created
PS C:\WINDOWS\system32> kubectl get deployments
NAME                READY   UP-TO-DATE   AVAILABLE   AGE
kubernetes-bootcamp 0/1     1            0           4s
nginx               1/1     1            1           64s
PS C:\WINDOWS\system32>
```

I deployed 2 apps

## Module 3: Explore your app

### Kubernetes Pods:

A **Pod** is a group of one or more application containers, such as Docker containers, with shared resources, including storage (volumes), networking (unique IP address), and configuration details

Pods represent the basic unit of deployment in Kubernetes and are used to run applications.

A **Node** is a worker machine in the Kubernetes cluster, where Pods are scheduled to run. Each Node manages Pods and communicates with the Kubernetes control plane.

### Troubleshooting with kubectl

To troubleshoot applications in your cluster, you can use the following kubectl commands:

- **kubectl get:** List resources (e.g., Pods, Nodes).
- **kubectl describe:** Show detailed information about a resource.
- **kubectl logs:** Print logs from a container inside a Pod.
- **kubectl exec:** Execute a command on a container inside a Pod.

These commands allow you to verify application status, check configurations, and diagnose issues.

### Step 1: Check Application Configuration

To verify that the application you deployed is running properly:

1. **List Pods:** Use the kubectl get pods command to check the status of the Pods.

kubectl get pods

```
PS C:\WINDOWS\system32> kubectl get pods
NAME                                READY   STATUS    RESTARTS   AGE
kubernetes-bootcamp-9bc58d867-17gjd 1/1     Running   0           5m46s
nginx-5869d7778c-npz94              1/1     Running   0           6m45s
PS C:\WINDOWS\system32>
```

## 2. Describe Pods: To see details about the Pod, including container images and ports:

kubectl describe pods

This will display information like the Pod's IP address, ports, and lifecycle events.

```
Container ID:   docker://9b58d86717gjd
Image:          gcr.io/google-samples/kubernetes-bootcamp:v1
Image ID:       docker-pullable://gcr.io/google-samples/kubernetes-bootcamp@sha256:0d608ee63bb57c5f5b6156f446b3c3c143d233037f3a2f00e279c8fcc64af
Port:          <none>
Host Port:     <none>
State:         Running
Started:       Mon, 17 Mar 2025 18:45:03 +0000
Ready:         True
Restart Count: 0
Environment:   <none>
Mounts:
  /var/run/secrets/kubernetes.io/serviceaccount from kube-api-access-brkg7 (ro)
Conditions:
  Type             Status
  PodReadyToStartContainers  True
  Initialized       True
  Ready            True
  ContainersReady  True
  PodScheduled     True
Volumes:
  kube-api-access-brkg7:
    Type:              Projected (a volume that contains injected data from multiple sources)
    TokenExpirationSeconds: 3607
    ConfigMapName:      kube-root-ca.crt
    ConfigMapOptional:  <nil>
    DownwardAPI:       true
    BestEffort        <none>
  Node-Selectors:      node.kubernetes.io/not-ready:NoExecute op=Exists for 300s
  Tolerations:         node.kubernetes.io/unreachable:NoExecute op=Exists for 300s
Events:
  Type    Reason      Age    From          Message
  ----    -
  Normal  Scheduled   6m5s   default-scheduler  Successfully assigned default/kubernetes-bootcamp-9bc58d867-17gjd to minikube
  Normal  Pulling     6m5s   kubelet        Pulling image "gcr.io/google-samples/kubernetes-bootcamp:v1"
  Normal  Pulled      6m5s   kubelet        Successfully pulled image "gcr.io/google-samples/kubernetes-bootcamp:v1" in 6.424s (6.424s including waiting). Image size: 191998640 bytes.
  Normal  Created     6m5s   kubelet        Created container: kubernetes-bootcamp
  Normal  Started     6m5s   kubelet        Started container: kubernetes-bootcamp

Name:          nginx-5869d778c-npz94
Namespace:    default
Priority:      0
Service Account: default
Node:         minikube
Start Time:   Mon, 17 Mar 2025 18:43:57 +0000
Labels:       app=nginx
Annotations:  pod-template-hash=5869d778c
Status:       Running
IP:          10.0.2.15
IPs:         <none>
Controlled By: Replicaset/nginx-5869d778c
Containers:
  nginx:
    Container ID:   docker://89d7be4c3a14524b59804914c9fe89e1dc54a90a6ea68192c682d04d1760a58
    Image:          nginx
    Image ID:       docker-pullable://nginx@sha256:9d6b58feebd2dbd3c56ab5853333d627cc6e281011cf05050fa4bcf2072c9496
    Port:          <none>
    Host Port:     <none>
    State:         Running
    Started:       Mon, 17 Mar 2025 18:44:03 +0000
    Ready:         True
    Restart Count: 0
    Environment:   <none>
    Mounts:
      /var/run/secrets/kubernetes.io/serviceaccount from kube-api-access-xdgmw (ro)
Conditions:
  Type             Status
  PodReadyToStartContainers  True
  Initialized       True
  Ready            True
  ContainersReady  True
  PodScheduled     True
Volumes:
  kube-api-access-xdgmw:
    Type:              Projected (a volume that contains injected data from multiple sources)
    TokenExpirationSeconds: 3607
    ConfigMapName:      kube-root-ca.crt
    ConfigMapOptional:  <nil>
    DownwardAPI:       true
    BestEffort        <none>
  Node-Selectors:      node.kubernetes.io/not-ready:NoExecute op=Exists for 300s
  Tolerations:         node.kubernetes.io/unreachable:NoExecute op=Exists for 300s
Events:
  Type    Reason      Age    From          Message
  ----    -
  Normal  Scheduled   7m8s   default-scheduler  Successfully assigned default/nginx-5869d778c-npz94 to minikube
  Normal  Pulling     7m8s   kubelet        Pulling image "nginx"
  Normal  Pulled      7m5s   kubelet        Successfully pulled image "nginx" in 5.499s (5.499s including waiting). Image size: 191998640 bytes.
  Normal  Created     7m5s   kubelet        Created container: nginx
  Normal  Started     7m5s   kubelet        Started container: nginx
PS C:\WINDOWS\system32>
```

**Export Pod Name:** Get the Pod name and store it in a variable for easier reference:

**Access the Application:** Use curl to access the application via the proxy:

```
PS C:\WINDOWS\system32> $env:POD_NAME = $(kubectl get pods -o jsonpath='{.items[0].metadata.name}')
PS C:\WINDOWS\system32> echo "Name of the Pod: $env:POD_NAME"
Name of the Pod: kubernetes-bootcamp-9bc58d867-17gjd
PS C:\WINDOWS\system32>
```

## View Logs

To view the logs from the Pod, use the kubectl logs command:

```
kubectl logs "$POD_NAME"
```

## Execute Commands Inside the Container

You can execute commands directly in the container inside the Pod:

- To list environment variables:

```
kubectl exec "$POD_NAME" -- env
```

- To start an interactive bash session:

```
kubectl exec -ti $POD_NAME -- bash
```

```
kubernetes-bootcamp-9bc58d867-17gjd
PS C:\WINDOWS\system32> kubectl logs "$POD_NAME"
kubernetes Bootcamp App Started At: 2025-03-17T18:45:03.564Z | Running On: kubernetes-bootcamp-9bc58d867-17gjd

PS C:\WINDOWS\system32> kubectl exec "$POD_NAME" -- env
PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin
HOSTNAME=kubernetes-bootcamp-9bc58d867-17gjd
KUBERNETES_SERVICE_PORT_HTTPS=443
KUBERNETES_PORT=tcp://10.0.1.1:443
KUBERNETES_PORT_443_TCP=tcp://10.0.1.1:443
KUBERNETES_PORT_443_TCP_PROTO=tcp
KUBERNETES_PORT_443_TCP_PORT=443
KUBERNETES_PORT_443_TCP_ADDR=10.0.1.1
KUBERNETES_SERVICE_HOST=10.0.1.1
KUBERNETES_SERVICE_PORT=443
IPM_CONFIG_LOGLEVEL=info
NODE_VERSION=6.3.1
HOME=/root
PS C:\WINDOWS\system32> kubectl exec -ti $POD_NAME -- bash
root@kubernetes-bootcamp-9bc58d867-17gjd:/#
```

Once inside the container, you can check the application's code by running:

```
cat server.js
```

```
root@kubernetes-bootcamp-9bc58d867-17gjd:/# cat server.js
var http = require('http');
var requests=0;
var podname= process.env.HOSTNAME;
var startTime;
var host;
var handleRequest = function(request, response) {
  response.setHeader('Content-Type', 'text/plain');
  response.writeHead(200);
  response.write("Hello Kubernetes bootcamp! | Running on: ");
  response.write(host);
  response.end(" | v=1\n");
  console.log("Running On:", host, " | Total Requests:", ++requests, " | App Uptime:", (new Date() - startTime)/1000 , "seconds", " | Log Time:",new Date());
};
var www = http.createServer(handleRequest);
www.listen(8080,function () {
  startTime = new Date();
  host = process.env.HOSTNAME;
  console.log ("Kubernetes Bootcamp App Started At:",startTime, " | Running On: " ,host, "\n" );
});
root@kubernetes-bootcamp-9bc58d867-17gjd:/# exit
PS C:\WINDOWS\system32>
```

## Module 4: Using a Service to Expose Your App

Kubectl get services

```
PS C:\WINDOWS\system32> kubectl get services
NAME         TYPE        CLUSTER-IP   EXTERNAL-IP   PORT(S)    AGE
kubernetes   ClusterIP   10.0.1.1     <none>        443/TCP    30m
PS C:\WINDOWS\system32>
```

## Expose the Deployment

Use the kubectl expose command to create a Service:

kubectl expose deployment/kubernetes-bootcamp --type="NodePort" --port 8080

```
PS C:\WINDOWS\system32> kubectl get services
NAME         TYPE        CLUSTER-IP   EXTERNAL-IP   PORT(S)    AGE
kubernetes   ClusterIP   10.0.0.1      <none>        443/TCP    30m
PS C:\WINDOWS\system32> kubectl expose deployment/kubernetes-bootcamp --type="NodePort" --port 8080
service/kubernetes-bootcamp exposed
PS C:\WINDOWS\system32> kubectl get services
NAME              TYPE        CLUSTER-IP   EXTERNAL-IP   PORT(S)            AGE
kubernetes        ClusterIP   10.0.0.1      <none>        443/TCP            31m
kubernetes-bootcamp NodePort     10.0.0.2.210  <none>        8080:32639/TCP     5s
PS C:\WINDOWS\system32>
```

Detailed

```
kubernetes-bootcamp NodePort 10.0.0.2.210 <none> 8080:32639/TCP 5s
PS C:\WINDOWS\system32> kubectl describe services/kubernetes-bootcamp
Name:         kubernetes-bootcamp
Namespace:    default
Labels:       app=kubernetes-bootcamp
Annotations:  <none>
Selector:     app=kubernetes-bootcamp
Type:         NodePort
IP Family Policy: SingleStack
IP Families:  IPv4
IP:           10.0.0.210
IPs:          10.0.0.210
Port:         <unset> 8080/TCP
TargetPort:   8080/TCP
NodePort:     <unset> 32639/TCP
Endpoints:    10.0.0.2:8080
Session Affinity: None
External Traffic Policy: Cluster
Internal Traffic Policy: Cluster
Events:       <none>
PS C:\WINDOWS\system32>
```

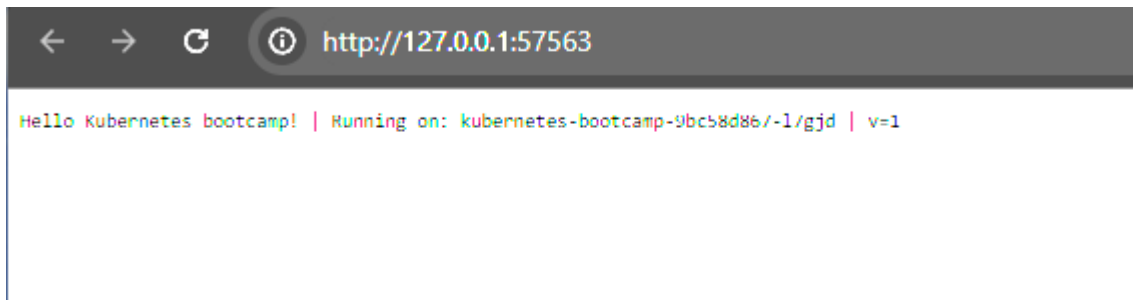
*kubectl describe services/kubernetes-bootcamp*

*\$env:NODE\_PORT = \$(kubectl get services/kubernetes-bootcamp -o go-template='{{{(index .spec.ports 0).nodePort}}}')'*

Check if the environment variable has been set correctly:

*echo \$env:NODE\_PORT*

**Now that your service is exposed, test it**



### Deleting and cleaning up

```
kubernetes-bootcamp-9bc58d86/-1/gjd 1/1 Running 0 31m
PS C:\WINDOWS\system32> kubectl label pods "$POD_NAME" version=v1
error: resource(s) were provided, but no name was specified
PS C:\WINDOWS\system32> kubectl get pods -l version=v1
No resources found in default namespace.
PS C:\WINDOWS\system32> kubectl delete service -l app=kubernetes-bootcamp
service "kubernetes-bootcamp" deleted
PS C:\WINDOWS\system32> kubectl get services
NAME         TYPE        CLUSTER-IP   EXTERNAL-IP   PORT(S)    AGE
kubernetes   ClusterIP   10.0.0.1     <none>        443/TCP    37m
PS C:\WINDOWS\system32> curl http://"$($minikube ip):$env:NODE_PORT"
```

## Module 5: scale up your app

The screenshot shows the process of exposing the **kubernetes-bootcamp** deployment with a LoadBalancer and scaling it to 4 replicas. After exposing the deployment using `kubectl expose`, the number of replicas was initially 1. Then, by running `kubectl scale`, the deployment was scaled to 4 replicas, and the updated pod statuses confirm the scaling was successful.

```
PS C:\WINDOWS\system32> kubectl expose deployment/kubernetes-bootcamp --type="LoadBalancer" --port 8080
service/kubernetes-bootcamp exposed
PS C:\WINDOWS\system32> kubectl get deployments
NAME          READY   UP-TO-DATE   AVAILABLE   AGE
kubernetes-bootcamp  1/1     1             1           34m
nginx         1/1     1             1           35m
PS C:\WINDOWS\system32> kubectl get rs
NAME                                DESIRED   CURRENT   READY   AGE
kubernetes-bootcamp-9bc58d867       1         1         1       34m
nginx-5869d7778c                    1         1         1       35m
PS C:\WINDOWS\system32> kubectl scale deployments/kubernetes-bootcamp --replicas=4
deployment.apps/kubernetes-bootcamp scaled
PS C:\WINDOWS\system32> kubectl get deployments
NAME          READY   UP-TO-DATE   AVAILABLE   AGE
kubernetes-bootcamp  2/4     4             2           34m
nginx         1/1     1             1           35m
PS C:\WINDOWS\system32> kubectl get pods -o wide
NAME                                READY   STATUS    RESTARTS   AGE   IP            NODE          NOMINATED NODE   READINESS GATES
kubernetes-bootcamp-9bc58d867-9hfbk  1/1     Running   0           3s    10.0.2.15     minikube     <none>            <none>
kubernetes-bootcamp-9bc58d867-fsgcv  1/1     Running   0           3s    10.0.2.16     minikube     <none>            <none>
kubernetes-bootcamp-9bc58d867-jwxn8  1/1     Running   0           3s    10.0.2.17     minikube     <none>            <none>
kubernetes-bootcamp-9bc58d867-17gjd  1/1     Running   0           34m   10.0.2.18     minikube     <none>            <none>
nginx-5869d7778c-npz94               1/1     Running   0           35m   10.0.2.19     minikube     <none>            <none>
```



## Check Deployment Events

To check the events and see the scale operation, run:

*kubectl describe deployments/kubernetes-bootcamp*

```
PS C:\WINDOWS\system32> kubectl describe deployments/kubernetes-bootcamp
Name:                kubernetes-bootcamp
Namespace:           default
CreationTimestamp:    Mon, 17 Mar 2025 18:44:56 +0000
Labels:              app=kubernetes-bootcamp
Annotations:         deployment.kubernetes.io/revision: 1
Selector:             app=kubernetes-bootcamp
Replicas:            4 desired | 4 updated | 4 total | 4 available | 0 unavailable
StrategyType:        RollingUpdate
MinReadySeconds:      0
RollingUpdateStrategy: 25% max unavailable, 25% max surge
Pod Template:
  Labels:  app=kubernetes-bootcamp
  Containers:
    kubernetes-bootcamp:
      Image:  gcr.io/google-samples/kubernetes-bootcamp:v1
      Port:  <none>
      Host Port:  <none>
      Environment:  <none>
      Mounts:  <none>
      Volumes:  <none>
      Node-Selectors:  <none>
      Tolerations:  <none>
  Conditions:
    Type           Status  Reason
    ----           -
    Progressing     True    NewReplicaSetAvailable
    Available       True    MinimumReplicasAvailable
  OldReplicaSets:  <none>
  NewReplicaSet:   kubernetes-bootcamp-9bc58d867 (4/4 replicas created)
Events:
  Type    Reason             Age    From                      Message
  ----    -
  Normal  ScalingReplicaSet  38m    deployment-controller     Scaled up replica set kubernetes-bootcamp-9bc58d867 from 0 to 1
  Normal  ScalingReplicaSet  4m32s  deployment-controller     Scaled up replica set kubernetes-bootcamp-9bc58d867 from 1 to 4
PS C:\WINDOWS\system32>
```

Before scaling down, there were 4 pods running for the kubernetes-bootcamp deployment, all listed as "Running" with their respective IPs and nodes. The deployment had 4 replicas, which were all available and up-to-date.

After scaling down, the number of pods decreased to 2. The deployment now reflects only 2 replicas instead of 4, confirming the scale-down operation.

```
PS C:\WINDOWS\system32> kubectl get pods --wide
NAME                                READY   STATUS    RESTARTS   AGE   IP              NODE             NOMINATED NODE   READINESS GATES
kubernetes-bootcamp-9bc58d867-9nfbx 1/1     Running   0          6m42s  10.0.0.8        minikube         <none>            <none>
kubernetes-bootcamp-9bc58d867-9nfcv 1/1     Running   0          6m42s  10.0.0.7        minikube         <none>            <none>
kubernetes-bootcamp-9bc58d867-9nwb 1/1     Running   0          6m42s  10.0.0.9        minikube         <none>            <none>
kubernetes-bootcamp-9bc58d867-9nqio 1/1     Running   0          40s    10.0.0.6        minikube         <none>            <none>
nginx-5868d7778c-np9m4              1/1     Running   0          41m    10.0.0.5        minikube         <none>            <none>

PS C:\WINDOWS\system32> kubectl describe deployments/kubernetes-bootcamp
Name:                kubernetes-bootcamp
Namespace:           default
CreationTimestamp:    Mon, 17 Mar 2025 18:44:56 +0000
Labels:              app=kubernetes-bootcamp
Annotations:         deployment.kubernetes.io/revision: 1
Selector:             app=kubernetes-bootcamp
Replicas:            4 desired | 4 updated | 4 total | 4 available | 0 unavailable
StrategyType:        RollingUpdate
MinReadySeconds:      0
RollingUpdateStrategy: 25% max unavailable, 25% max surge
Pod Template:
  Labels:  app=kubernetes-bootcamp
  Containers:
    kubernetes-bootcamp:
      Image:  gcr.io/google-samples/kubernetes-bootcamp:v1
      Port:  <none>
      Host Port:  <none>
      Environment:  <none>
      Mounts:  <none>
      Volumes:  <none>
      Node-Selectors:  <none>
      Tolerations:  <none>
  Conditions:
    Type           Status  Reason
    ----           -
    Progressing     True    NewReplicaSetAvailable
    Available       True    MinimumReplicasAvailable
  OldReplicaSets:  <none>
  NewReplicaSet:   kubernetes-bootcamp-9bc58d867 (4/4 replicas created)
Events:
  Type    Reason             Age    From                      Message
  ----    -
  Normal  ScalingReplicaSet  40m    deployment-controller     Scaled up replica set kubernetes-bootcamp-9bc58d867 from 0 to 1
  Normal  ScalingReplicaSet  6m45s  deployment-controller     Scaled up replica set kubernetes-bootcamp-9bc58d867 from 1 to 4
PS C:\WINDOWS\system32> kubectl get pods --wide
NAME                                READY   UP-TO-DATE   AVAILABLE   AGE   IP              NODE             NOMINATED NODE   READINESS GATES
kubernetes-bootcamp 4/4      4            4           40m    10.0.0.8        minikube         <none>            <none>
nginx                1/1      1            1           41m    10.0.0.5        minikube         <none>            <none>

PS C:\WINDOWS\system32> kubectl get pods --wide
NAME                                READY   STATUS    RESTARTS   AGE   IP              NODE             NOMINATED NODE   READINESS GATES
kubernetes-bootcamp-9bc58d867-9nfbx 1/1     Running   0          6m45s  10.0.0.8        minikube         <none>            <none>
kubernetes-bootcamp-9bc58d867-9nfcv 1/1     Running   0          6m45s  10.0.0.7        minikube         <none>            <none>
```

Module 6: update your app

*kubectl get pods*

This will display the Pods running for your deployment.

*kubectl describe pods*

Look for the Image field in the output to see the version of the application running (e.g., v1).

## Perform the Rolling Update

*kubectl set image deployments/kubernetes-bootcamp kubernetes-bootcamp=docker.io/jocatalin/kubernetes-bootcamp:v2*

This command notifies Kubernetes to update the Pods with the new image, starting the rolling update.

```
PS C:\WINDOWS\system32> kubectl get pods
NAME                                READY   STATUS    RESTARTS   AGE
kubernetes-bootcamp-9bc58d867-9hfbk 1/1     Running   0           9m41s
kubernetes-bootcamp-9bc58d867-fsgcv 1/1     Running   0           9m41s
kubernetes-bootcamp-9bc58d867-jwxn8 1/1     Running   0           9m41s
kubernetes-bootcamp-9bc58d867-17gjd 1/1     Running   0           43m
nginx-5869d7778c-npz94              1/1     Running   0           44m
PS C:\WINDOWS\system32> kubectl set image deployments/kubernetes-bootcamp kubernetes-bootcamp=docker.io/jocatalin/kubernetes-bootcamp:v2
deployment.apps/kubernetes-bootcamp image updated
PS C:\WINDOWS\system32> kubectl get pods
NAME                                READY   STATUS             RESTARTS   AGE
kubernetes-bootcamp-5c4f7cb664-77w7v 0/1     ContainerCreating   0           3s
kubernetes-bootcamp-5c4f7cb664-jt2np 0/1     ContainerCreating   0           3s
kubernetes-bootcamp-9bc58d867-9hfbk 1/1     Running             0           9m46s
kubernetes-bootcamp-9bc58d867-fsgcv 1/1     Running             0           9m46s
kubernetes-bootcamp-9bc58d867-jwxn8 1/1     Terminating       0           9m46s
kubernetes-bootcamp-9bc58d867-17gjd 1/1     Running             0           43m
nginx-5869d7778c-npz94              1/1     Running             0           44m
PS C:\WINDOWS\system32>
```

Showing here update took place and then deleting the clusters/cleaning up

```
kube-api-access-lmhc5:
  Type:          Projected (a volume that contains injected data from multiple sources)
  TokenExpirationSeconds: 3607
  ConfigMapName: kube-root-ca.crt
  ConfigMapOptional: <nil>
  DownwardAPI:   true
os 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   2m15s default-scheduler Successfully assigned default/kubernetes-bootcamp-5c4f7cb664-jt2np to minikube
Normal    Pulling     2m14s kubelet       Pulling image "docker.io/jocatalin/kubernetes-bootcamp:v2"
Normal    Pulled      2m12s kubelet       Successfully pulled image "docker.io/jocatalin/kubernetes-bootcamp:v2" in 1.766s (1.766s including waiting)
Normal    Created     2m12s kubelet       Created container: kubernetes-bootcamp
Normal    Started     2m12s kubelet       Started container kubernetes-bootcamp

Name:          nginx-5869d7778c-npz94
Namespace:     default
Priority:       0
ServiceAccount: default
Node:          minikube/192.168.1.5
Start Time:    Mon, 17 Mar 2025 18:43:57 +0000
Labels:        app=nginx
               pod-template-hash=5869d7778c
Annotations:   <none>
Status:        Running
IP:            192.168.1.5
IPs:           192.168.1.5
Controlled By: ReplicaSet/nginx-5869d7778c
Containers:
  nginx:
    Container ID:  docker://89d7be4c3a14524b59804914c9fe89e1dcd54a90a6ea68192c682d04d1760a58
    Image:         nginx
    Image ID:      docker-pullable://nginx@sha256:9d6b58feebd2dbd3c56ab5853333d627cc6e281011cfd6050fa4bcf2072c9496
    Port:          <none>
    Host Port:     <none>
    State:         Running
      Started:     Mon, 17 Mar 2025 18:44:03 +0000
      Ready:       True
      Restart Count: 0
    Environment:   <none>
    Mounts:
      /var/run/secrets/kubernetes.io/serviceaccount from kube-api-access-xdgwn (ro)
Conditions:
  Type              Status
  PodReadyToStartContainers  True
  Initialized         True
  Ready               True
  ContainersReady     True
  PodScheduled        True
Volumes:
  kube-api-access-xdgwn:
    Type:          Projected (a volume that contains injected data from multiple sources)
    TokenExpirationSeconds: 3607
    ConfigMapName: kube-root-ca.crt
    ConfigMapOptional: <nil>
    DownwardAPI:   true
os 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   47m   default-scheduler Successfully assigned default/nginx-5869d7778c-npz94 to minikube
Normal    Pulling     47m   kubelet       Pulling image "nginx"
Normal    Pulled      47m   kubelet       Successfully pulled image "nginx" in 5.499s (5.499s including waiting). Image size: 191998640
Normal    Created     47m   kubelet       Created container: nginx
Normal    Started     47m   kubelet       Started container nginx
S C:\WINDOWS\system32> kubectl rollout undo deployments/kubernetes-bootcamp
deployment.apps/kubernetes-bootcamp rolled back
S C:\WINDOWS\system32> kubectl delete deployments/kubernetes-bootcamp services/kubernetes-bootcamp
deployment.apps "kubernetes-bootcamp" deleted
service "kubernetes-bootcamp" deleted
S C:\WINDOWS\system32>
```