
Department of Computing

Bachelor of Science (Hons) in Software Development

# Cloud Data Centres - Y4

## Lab 4

**Student name:** Sebastian Firsaev

**Student Number:** C00263348

**Lecturer:** Dr Lei Shi

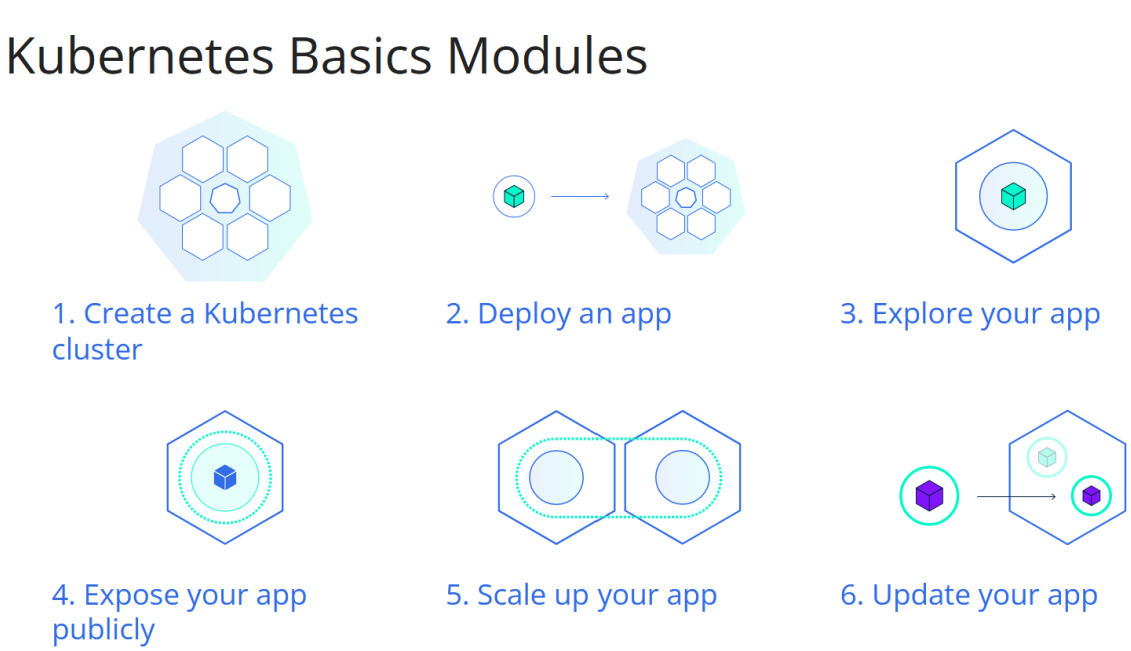
## Table of Contents

# Learning Kubernetes Basics

Undertaking the Kubernetes Basics tutorial afforded me a comprehensive examination of the foundational aspects of the Kubernetes cluster orchestration system. Each module within this tutorial provided essential contextual information regarding prominent Kubernetes features and concepts, supplemented by practical tutorials for hands-on engagement.

Throughout this educational journey, I acquired proficiency in:

- Deploying containerized applications on Kubernetes clusters, fostering a nuanced understanding of cluster management principles.
- Dynamically scaling deployments to accommodate varying workload requirements, optimizing resource allocation strategies.
- Seamlessly updating containerized applications with new software iterations, ensuring uninterrupted service continuity.
- Proficiently debugging containerized applications, refining my diagnostic capabilities within the Kubernetes environment.




In the contemporary landscape of web services, the imperative for applications to maintain 24/7 availability is paramount. Furthermore, as a developer, I acknowledge the significance of deploying new application versions frequently, often multiple times within a day. Containerization has emerged as a pivotal solution, facilitating the packaging of software for streamlined deployment and updates, sans any disruption to availability.

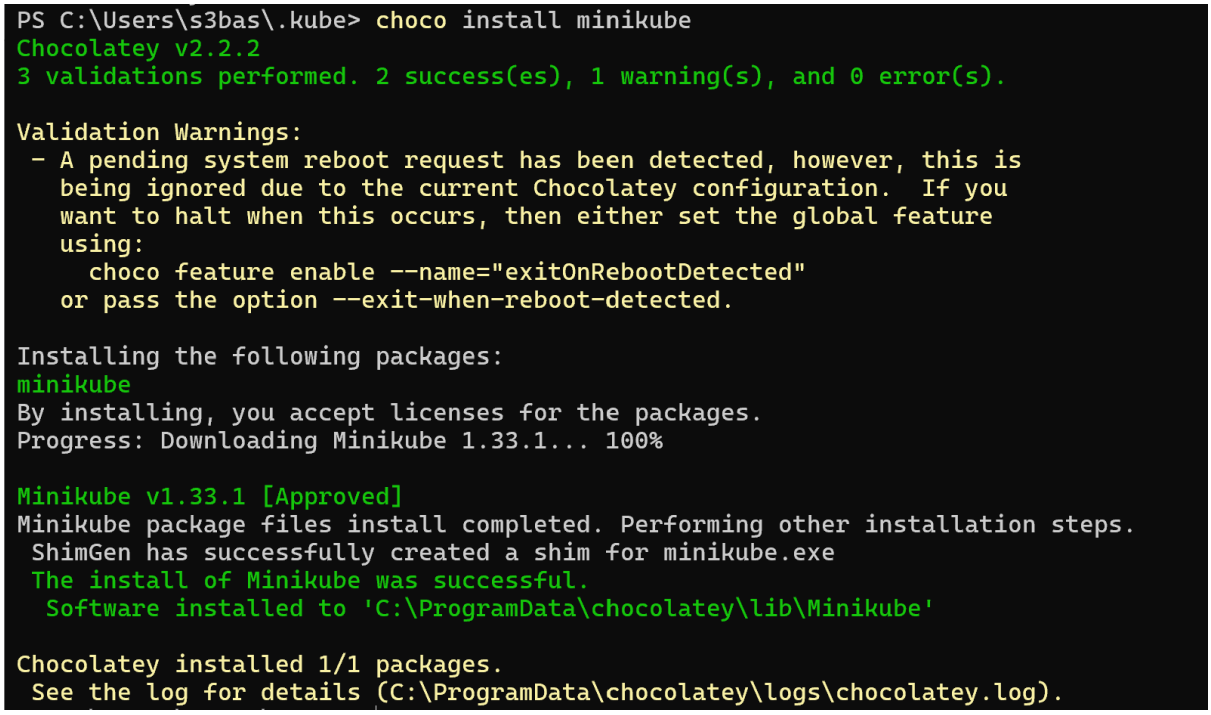
Kubernetes assumed a pivotal role in this educational pursuit, serving as a robust tool to ensure the reliable and efficient execution of containerized applications. Its

dynamic resource management capabilities and adept provision of support where necessary significantly augmented my comprehension of container orchestration principles.

As a production-ready, open-source platform, Kubernetes embodies Google's extensive expertise in container orchestration, complemented by innovative contributions from the broader community. Through this tutorial, I garnered valuable insights into the Kubernetes ecosystem, equipping me to adeptly navigate and harness its capabilities in future endeavours.

# Getting started with Minikube

## Installing minikube:



## Start of Kubernetes minikube:

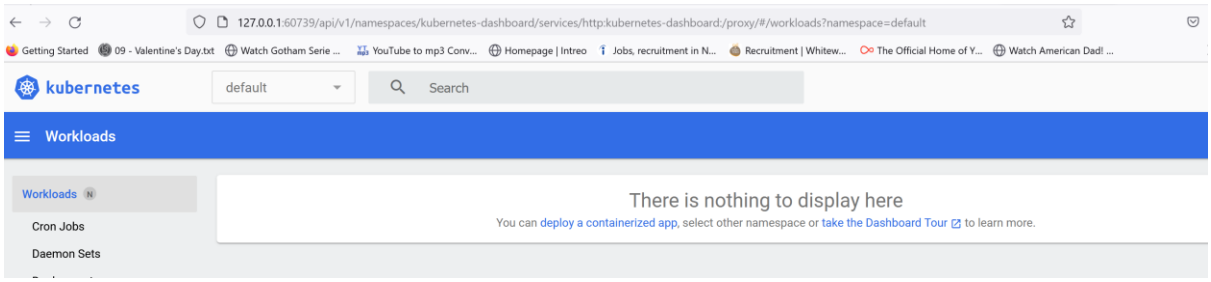
```
PS C:\Users\s3bas> minikube start
W0516 18:50:26.954407   28192 main.go:291] Unable to resolve the current Docker CLI context "default": context "default"
: context not found: open C:\Users\s3bas\.docker\contexts\meta\37a8eec1ce19687d132fe29051dca629d164e2c4958ba141d5f4133a3
3f0688f\meta.json: The system cannot find the path specified.
😄  minikube v1.33.1 on Microsoft Windows 11 Pro 10.0.22631.3447 Build 22631.3447
🆕  Kubernetes 1.30.0 is now available. If you would like to upgrade, specify: --kubernetes-version=v1.30.0
✨  Using the docker driver based on existing profile
👍  Starting "minikube" primary control-plane node in "minikube" cluster
🚜  Pulling base image v0.0.44 ...
🔄  Restarting existing docker container for "minikube" ...
❗  Image was not built for the current minikube version. To resolve this you can delete and recreate your minikube clus
ter using the latest images. Expected minikube version: v1.32.0 -> Actual minikube version: v1.33.1
🐳  Preparing Kubernetes v1.28.3 on Docker 24.0.7 ...- Bad local forwarding specification '0:localhost:8443'

🔗  Configuring bridge CNI (Container Networking Interface) ...
🔎  Verifying Kubernetes components...
    ▪ Using image docker.io/kubernetesui/dashboard:v2.7.0
    ▪ Using image docker.io/kubernetesui/metrics-scraper:v1.0.8
    ▪ Using image gcr.io/k8s-minikube/storage-provisioner:v5
💡  Some dashboard features require the metrics-server addon. To enable all features please run:

        minikube addons enable metrics-server

🌟  Enabled addons: storage-provisioner, default-storageclass, dashboard

❗  C:\ProgramData\chocolatey\bin\kubectl.exe is version 1.30.1, which may have incompatibilities with Kubernetes 1.28.3
.
    ▪ Want kubectl v1.28.3? Try 'minikube kubectl -- get pods -A'
🏄  Done! kubectl is now configured to use "minikube" cluster and "default" namespace by default
PS C:\Users\s3bas>
```

## list all the pods across all namespaces in Kubernetes cluster:

```
PS C:\Users\s3bas> kubectl get po -A
NAMESPACE              NAME                                         READY   STATUS    RESTARTS        AGE
kube-system            coredns-5dd5756b68-zj6vc                     1/1     Running   1 (106d ago)    106d
kube-system            etcd-minikube                                1/1     Running   1 (106d ago)    106d
kube-system            kube-apiserver-minikube                      1/1     Running   1 (106d ago)    106d
kube-system            kube-controller-manager-minikube             1/1     Running   1 (106d ago)    106d
kube-system            kube-proxy-4bvqt                             1/1     Running   1 (106d ago)    106d
kube-system            kube-scheduler-minikube                      1/1     Running   1 (106d ago)    106d
kube-system            storage-provisioner                          1/1     Running   3 (2m22s ago)   106d
kubernetes-dashboard   dashboard-metrics-scraper-7fd5cb4ddc-bmlm7   1/1     Running   1 (106d ago)    106d
kubernetes-dashboard   kubernetes-dashboard-8694d4445c-84lk2        1/1     Running   2 (2m21s ago)   106d
PS C:\Users\s3bas>
```

## Opening Kubernetes Dashboard:

```
PS C:\Users\s3bas> minikube dashboard
W0516 18:57:03.619596    8236 main.go:291] Unable to resolve the current Docker CLI context "default": context "default"
: context not found: open C:\Users\s3bas\.docker\contexts\meta\37a8eec1ce19687d132fe29051dca629d164e2c4958ba141d5f4133a3
3f0688f\meta.json: The system cannot find the path specified.
🤔  Verifying dashboard health ...
🚀  Launching proxy ...
🤔  Verifying proxy health ...
🎉  Opening http://127.0.0.1:60739/api/v1/namespaces/kubernetes-dashboard/services/http:kubernetes-dashboard:/proxy/ in
your default browser...
```



# Creating a sample deployment,

## Running test container image that includes a webserver:

```
PS C:\Users\s3bas> kubectl create deployment hello-node --image=registry.k8s.io/e2e-test-images/agnhost:2.39 -- /agnhost
 netexec --http-port=8080
deployment.apps/hello-node created
PS C:\Users\s3bas>
```

## Showing deployment:

```
PS C:\Users\s3bas> kubectl get deployments
NAME             READY   UP-TO-DATE   AVAILABLE   AGE
hello-minikube   1/1     1            1           20m
hello-node       1/1     1            1           86s
PS C:\Users\s3bas>
```

## Viewing the pod:

```
PS C:\Users\s3bas> kubectl get pods
NAME                              READY   STATUS    RESTARTS        AGE
hello-minikube-7f54cff968-dj7dd   1/1     Running   1 (7m23s ago)   22m
hello-node-ccf4b9788-tqpnd        1/1     Running   0               3m10s
PS C:\Users\s3bas>
```

## Viewing Cluster events:

```
PS C:\Users\s3bas> kubectl get events
LAST SEEN   TYPE     REASON                 OBJECT                                  MESSAGE
23m         Normal   Scheduled              pod/hello-minikube-7f54cff968-dj7dd     Successfully assigned default/hell
o-minikube-7f54cff968-dj7dd to minikube
23m         Normal   Pulling                pod/hello-minikube-7f54cff968-dj7dd     Pulling image "kicbase/echo-server
:1.0"
23m         Normal   Pulled                 pod/hello-minikube-7f54cff968-dj7dd     Successfully pulled image "kicbase
/echo-server:1.0" in 5.022s (5.022s including waiting)
23m         Normal   Created                pod/hello-minikube-7f54cff968-dj7dd     Created container echo-server
23m         Normal   Started                pod/hello-minikube-7f54cff968-dj7dd     Started container echo-server
5m8s        Normal   SandboxChanged         pod/hello-minikube-7f54cff968-dj7dd     Pod sandbox changed, it will be ki
lled and re-created.
4m58s       Normal   Pulled                 pod/hello-minikube-7f54cff968-dj7dd     Container image "kicbase/echo-serv
er:1.0" already present on machine
4m57s       Normal   Created                pod/hello-minikube-7f54cff968-dj7dd     Created container echo-server
4m56s       Normal   Started                pod/hello-minikube-7f54cff968-dj7dd     Started container echo-server
23m         Normal   SuccessfulCreate       replicaset/hello-minikube-7f54cff968    Created pod: hello-minikube-7f54cf
f968-dj7dd
```

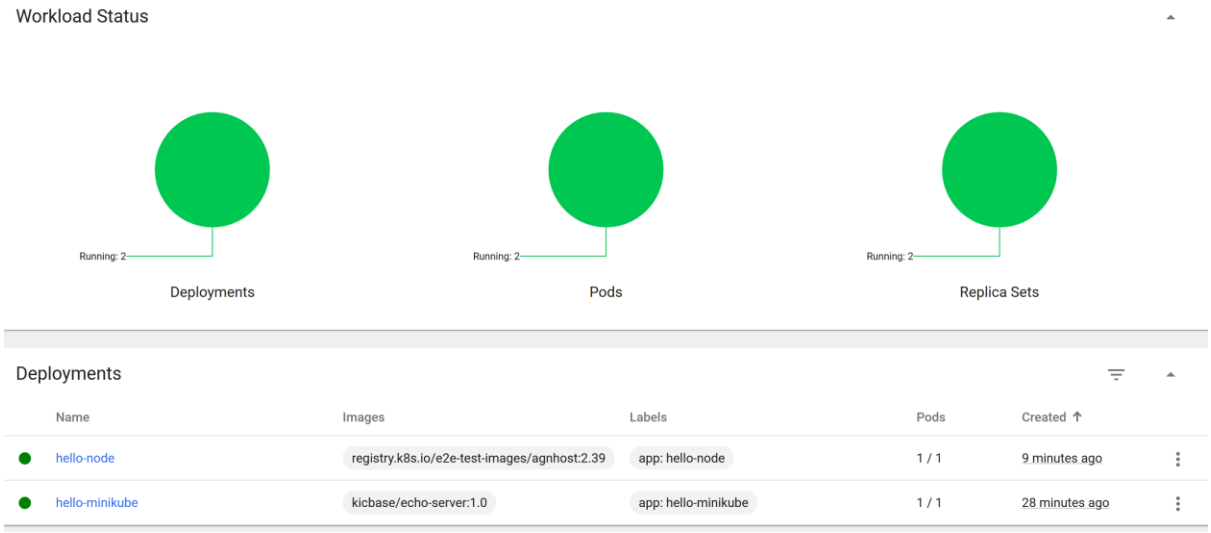
## Viewing the kubectl configuration:

```
PS C:\Users\s3bas> kubectl config view
apiVersion: v1
clusters:
- cluster:
    certificate-authority: C:\Users\s3bas\.minikube\ca.crt
    extensions:
    - extension:
```

## Viewing application logs for a container in a pod:

```
PS C:\Users\s3bas> kubectl logs hello-node-ccf4b9788-tqpnd
I0516 18:19:51.426711       1 log.go:195] Started HTTP server on port 8080
I0516 18:19:51.427239       1 log.go:195] Started UDP server on port  8081
PS C:\Users\s3bas>
```

## Updated Dashboard:




## Viewing created Service:

```
PS C:\Users\s3bas> kubectl get services
NAME             TYPE           CLUSTER-IP      EXTERNAL-IP   PORT(S)          AGE
hello-minikube   NodePort       10.99.179.172   <none>        8080:31368/TCP   31m
hello-node       LoadBalancer   10.102.20.61    <pending>     8080:31910/TCP   11s
kubernetes       ClusterIP      10.96.0.1       <none>        443/TCP          106d
PS C:\Users\s3bas>
```

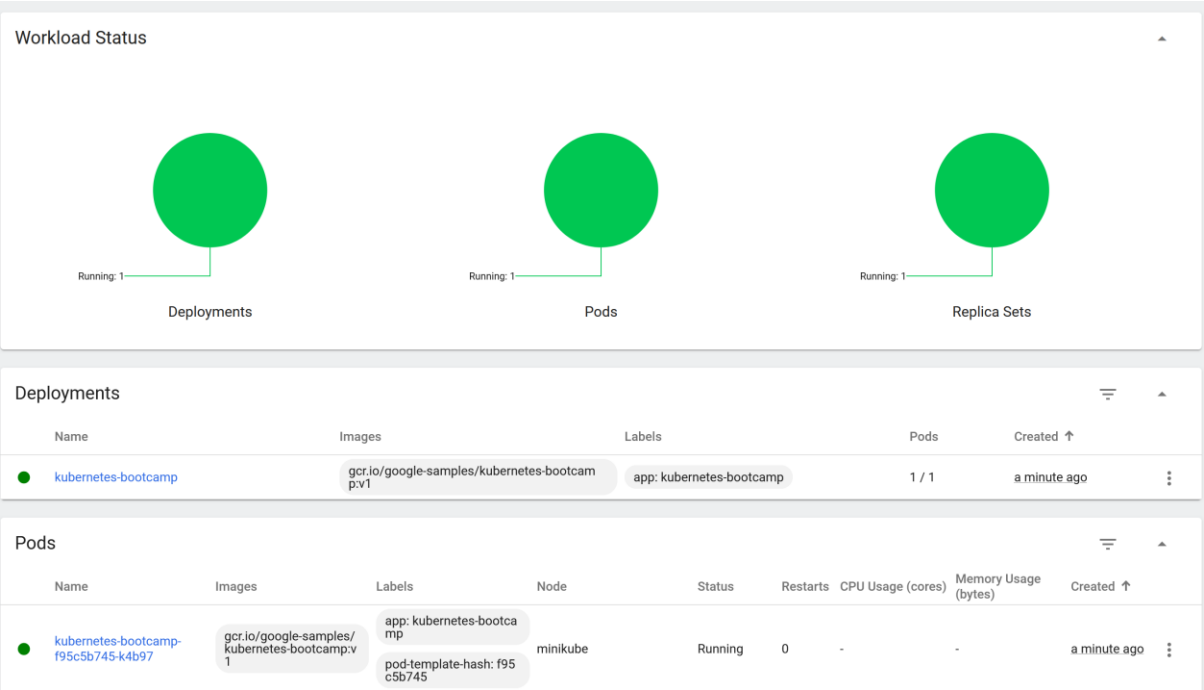
## Deploying Kubernetes bootcamp app image:

```
PS C:\Users\s3bas> kubectl create deployment kubernetes-bootcamp --image=gcr.io/google-samples/kubernetes-bootcamp:v1
deployment.apps/kubernetes-bootcamp created
PS C:\Users\s3bas>
```

## Listing Updated Deployments:

```
PS C:\Users\s3bas> kubectl get deployments
NAME                  READY   UP-TO-DATE   AVAILABLE   AGE
kubernetes-bootcamp   0/1     1            0           36s
PS C:\Users\s3bas>
```

## Checking Dashboard for new pod:




## Checking APIs hosted through the proxy endpoint:

```
PS C:\Users\s3bas> kubectl proxy
Starting to serve on 127.0.0.1:8001
```

```
PS C:\Users\s3bas> curl http://localhost:8001/version


StatusCode        : 200
StatusDescription : OK
Content           : {
                      "major": "1",
                      "minor": "28",
                      "gitVersion": "v1.28.3",
                      "gitCommit": "a8a1abc25cad87333840cd7d54be2efaf31a3177",
                      "gitTreeState": "clean",
                      "buildDate": "2023-10-18T11:33:18Z",
                      "goVersion"...
RawContent        : HTTP/1.1 200 OK
                    Audit-Id: 58b54b95-f503-47a8-a012-60f61ed4e047
                    X-Kubernetes-Pf-Flowschema-Uid: ceff8e65-d65d-4a44-b8c1-6b5fc74c0aff
                    X-Kubernetes-Pf-Prioritylevel-Uid: ec990d35-0242-44fe-b12b-f1b72e...
Forms             : {}
Headers           : {[Audit-Id, 58b54b95-f503-47a8-a012-60f61ed4e047], [X-Kubernetes-Pf-Flowschema-Uid,
                    ceff8e65-d65d-4a44-b8c1-6b5fc74c0aff], [X-Kubernetes-Pf-Prioritylevel-Uid,
                    ec990d35-0242-44fe-b12b-f1b72eb8c3f6], [Content-Length, 264]...}
Images            : {}
InputFields       : {}
Links             : {}
ParsedHtml        : mshtml.HTMLDocumentClass
RawContentLength  : 264
```

# Viewing Pods and Nodes:

## Check application configuration:

```
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  8m8s   default-scheduler  Successfully assigned default/kubernetes-bootcamp-f95c5b745-k4b97 to mini
kube
  Normal  Pulling    8m7s   kubelet            Pulling image "gcr.io/google-samples/kubernetes-bootcamp:v1"
  Normal  Pulled     7m24s  kubelet            Successfully pulled image "gcr.io/google-samples/kubernetes-bootcamp:v1"
in 43.069s (43.069s including waiting)
  Normal  Created    7m21s  kubelet            Created container kubernetes-bootcamp
  Normal  Started    7m21s  kubelet            Started container kubernetes-bootcamp
```

# Executing commands on the container

## Listing Available Commands:

```
PS C:\Users\s3bas> kubectl exec kubernetes-bootcamp-f95c5b745-k4b97 -- env
PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin
HOSTNAME=kubernetes-bootcamp-f95c5b745-k4b97
KUBERNETES_SERVICE_PORT_HTTPS=443
KUBERNETES_PORT_443_TCP_PROTO=tcp
HELLO_MINIKUBE_SERVICE_PORT=8080
HELLO_MINIKUBE_PORT_8080_TCP=tcp://10.99.179.172:8080
KUBERNETES_PORT_443_TCP=tcp://10.96.0.1:443
HELLO_MINIKUBE_PORT_8080_TCP_PROTO=tcp
HELLO_MINIKUBE_PORT_8080_TCP_PORT=8080
HELLO_MINIKUBE_PORT_8080_TCP_ADDR=10.99.179.172
KUBERNETES_PORT_443_TCP_PORT=443
KUBERNETES_PORT=tcp://10.96.0.1:443
KUBERNETES_PORT_443_TCP_ADDR=10.96.0.1
HELLO_MINIKUBE_SERVICE_HOST=10.99.179.172
HELLO_MINIKUBE_PORT=tcp://10.99.179.172:8080
KUBERNETES_SERVICE_HOST=10.96.0.1
KUBERNETES_SERVICE_PORT=443
NPM_CONFIG_LOGLEVEL=info
NODE_VERSION=6.3.1
HOME=/root
PS C:\Users\s3bas>
```

## Starting Bash Session in Pod's container and opening 'server.js':

```
PS C:\Users\s3bas> kubectl exec -ti kubernetes-bootcamp-f95c5b745-k4b97 -- bash
root@kubernetes-bootcamp-f95c5b745-k4b97:/# 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 , "sec
onds", "| 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-f95c5b745-k4b97:/#
```

## Verifying that the application is running:

```
root@kubernetes-bootcamp-f95c5b745-k4b97:/# curl http://localhost:8080
Hello Kubernetes bootcamp! | Running on: kubernetes-bootcamp-f95c5b745-k4b97 | v=1
root@kubernetes-bootcamp-f95c5b745-k4b97:/#
```

# Using a Service to Expose App:

## Creating a New Service:

```
PS C:\Users\s3bas> kubectl get services
NAME             TYPE        CLUSTER-IP      EXTERNAL-IP   PORT(S)          AGE
hello-minikube   NodePort    10.99.179.172   <none>        8080:31368/TCP   61m
kubernetes       ClusterIP   10.96.0.1       <none>        443/TCP          106d
PS C:\Users\s3bas>
```

```
PS C:\Users\s3bas> kubectl expose deployment/kubernetes-bootcamp --type="NodePort" --port 8080
service/kubernetes-bootcamp exposed
PS C:\Users\s3bas> kubectl get services
NAME                 TYPE        CLUSTER-IP      EXTERNAL-IP   PORT(S)          AGE
hello-minikube       NodePort    10.99.179.172   <none>        8080:31368/TCP   63m
kubernetes           ClusterIP   10.96.0.1       <none>        443/TCP          106d
kubernetes-bootcamp  NodePort    10.111.75.225   <none>        8080:30343/TCP   12s
PS C:\Users\s3bas>
```

## Finding the externally opened port on the service:

```
PS C:\Users\s3bas> 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:                       10.111.75.225
IPs:                      10.111.75.225
Port:                     <unset>  8080/TCP
TargetPort:               8080/TCP
NodePort:                 <unset>  30343/TCP
Endpoints:                10.244.0.15:8080
Session Affinity:         None
External Traffic Policy:  Cluster
Events:                   <none>
PS C:\Users\s3bas>
```

## Creating an environment variable called NODE_PORT that has the value of the Node port assigned:

```
PS C:\Users\s3bas> $env:NODE_PORT=$(kubectl get services/kubernetes-bootcamp -o go-template='{{(index .spec.ports 0).nod
ePort}}')
PS C:\Users\s3bas> echo "NODE_PORT=$NODE_PORT"
NODE_PORT=30343
PS C:\Users\s3bas>
```

## Testing that the app is exposed outside the cluster using curl:

```
PS C:\Users\s3bas> curl http://127.0.0.1:61155


StatusCode        : 200
StatusDescription : OK
Content           : {
                      "paths": [
                        "/.well-known/openid-configuration",
                        "/api",
                        "/api/v1",
                        "/apis",
                        "/apis/",
                        "/apis/admissionregistration.k8s.io",
                        "/apis/admissionregistration.k8s.io/v1",
                        ...
RawContent        : HTTP/1.1 200 OK
                    Audit-Id: a2ccf690-2eb1-4580-a6ce-bb8fb7ecbf6a
                    X-Kubernetes-Pf-Flowschema-Uid: ceff8e65-d65d-4a44-b8c1-6b5fc74c0aff
                    X-Kubernetes-Pf-Prioritylevel-Uid: ec990d35-0242-44fe-b12b-f1b72e...
Forms             : {}
Headers           : {[Audit-Id, a2ccf690-2eb1-4580-a6ce-bb8fb7ecbf6a], [X-Kubernetes-Pf-Flowschema-Uid,
                    ceff8e65-d65d-4a44-b8c1-6b5fc74c0aff], [X-Kubernetes-Pf-Prioritylevel-Uid,
                    ec990d35-0242-44fe-b12b-f1b72eb8c3f6], [Transfer-Encoding, chunked]...}
Images            : {}
InputFields       : {}
Links             : {}
ParsedHtml        : mshtml.HTMLDocumentClass
RawContentLength  : 7286
```

## Applying a new label:

```
PS C:\Users\s3bas> kubectl get pods -l app=kubernetes-bootcamp
NAME                                  READY   STATUS    RESTARTS   AGE
kubernetes-bootcamp-f95c5b745-k4b97   1/1     Running   0          39m
PS C:\Users\s3bas>
```

## Deleting a service:

```
PS C:\Users\s3bas> kubectl get services
NAME             TYPE        CLUSTER-IP      EXTERNAL-IP   PORT(S)          AGE
hello-minikube   NodePort    10.99.179.172   <none>        8080:31368/TCP   78m
kubernetes       ClusterIP   10.96.0.1       <none>        443/TCP          106d
PS C:\Users\s3bas>
```

## Confirming that the route is not exposed anymore:

```
PS C:\Users\s3bas> curl http://127.0.0.1:61155
curl : The underlying connection was closed: An unexpected error occurred on a receive.
At line:1 char:1
```

# Running Multiple Instances of App:

## Viewing the ReplicaSet created by the deployment:

```
PS C:\Users\s3bas> kubectl get deployments
NAME                  READY   UP-TO-DATE   AVAILABLE   AGE
kubernetes-bootcamp   1/1     1            1           50m
PS C:\Users\s3bas> kubectl get rs
NAME                            DESIRED   CURRENT   READY   AGE
kubernetes-bootcamp-f95c5b745   1         1         1       51m
PS C:\Users\s3bas>
```

## Scaling the deployment to 4 replicas:

```
PS C:\Users\s3bas> kubectl scale deployments/kubernetes-bootcamp --replicas=4
deployment.apps/kubernetes-bootcamp scaled
PS C:\Users\s3bas> kubectl get deployments
NAME                  READY   UP-TO-DATE   AVAILABLE   AGE
kubernetes-bootcamp   1/4     4            1           52m
PS C:\Users\s3bas>
```

```
PS C:\Users\s3bas> kubectl get pods -o wide
NAME                                  READY   STATUS    RESTARTS   AGE   IP            NODE       NOMINATED NODE   READI
NESS GATES
kubernetes-bootcamp-f95c5b745-gz2s6   1/1     Running   0          58s   10.244.0.16   minikube   <none>           <none
>
kubernetes-bootcamp-f95c5b745-j2wqw   1/1     Running   0          58s   10.244.0.18   minikube   <none>           <none
>
kubernetes-bootcamp-f95c5b745-k4b97   1/1     Running   0          53m   10.244.0.15   minikube   <none>           <none
>
kubernetes-bootcamp-f95c5b745-qg8d6   1/1     Running   0          58s   10.244.0.17   minikube   <none>           <none
>
PS C:\Users\s3bas>
```

## Scaling Down the application:

```
PS C:\Users\s3bas> kubectl scale deployments/kubernetes-bootcamp --replicas=2
deployment.apps/kubernetes-bootcamp scaled
PS C:\Users\s3bas> kubectl get deployments
NAME                  READY   UP-TO-DATE   AVAILABLE   AGE
kubernetes-bootcamp   2/2     2            2           59m
PS C:\Users\s3bas> kubectl get pods -o wide
NAME                                  READY   STATUS    RESTARTS   AGE     IP            NODE       NOMINATED NODE   REA
DINESS GATES
kubernetes-bootcamp-f95c5b745-k4b97   1/1     Running   0          60m     10.244.0.15   minikube   <none>           <no
ne>
kubernetes-bootcamp-f95c5b745-qg8d6   1/1     Running   0          7m53s   10.244.0.17   minikube   <none>           <no
ne>
PS C:\Users\s3bas>
```

## Performing an Update:

```
PS C:\Users\s3bas> kubectl set image deployments/kubernetes-bootcamp kubernetes-bootcamp=docker.io/jocatalin/kubernetes-
bootcamp:v2
deployment.apps/kubernetes-bootcamp image updated
PS C:\Users\s3bas> kubectl get pods
NAME                                  READY   STATUS        RESTARTS   AGE
kubernetes-bootcamp-9cfc76686-m6cjb   1/1     Running       0          4s
kubernetes-bootcamp-9cfc76686-vfx5p   1/1     Running       0          11s
```

## Running the roll out status command:

```
PS C:\Users\s3bas> kubectl rollout status deployments/kubernetes-bootcamp
deployment "kubernetes-bootcamp" successfully rolled out
PS C:\Users\s3bas>
```

## Cleaning up the local cluster:

```
PS C:\Users\s3bas> kubectl delete deployments/kubernetes-bootcamp services/kubernetes-bootcamp
deployment.apps "kubernetes-bootcamp" deleted
```