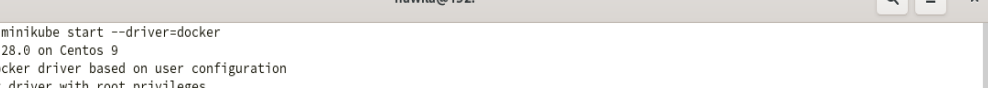


Pre requisite: Install k8s cluster (minikube) + kubectl



```
[hawila@192 ~]$ minikube start --driver=docker
🐳 minikube v1.28.0 on Centos 9
🔧 Using the docker driver based on user configuration
🔧 Using Docker driver with root privileges
👤 Starting control plane node minikube in cluster minikube
📶 Pulling base image ...
🔥 Creating docker container (CPUs=2, Memory=2200MB) ...
📦 Preparing Kubernetes v1.25.3 on Docker 20.10.20 ...
  • 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
🔧 kubectrl not found. If you need it, try: 'minikube kubectrl -- get pods -A'
🏠 Done! kubectrl is now configured to use "minikube" cluster and "default" namespace by default
```

1- Create a pod with the name "imperative-nginx" and with the image `nginx` and



```
hawila@localhost:~  
[hawila@localhost ~]$ kubectl run imperative-nginx --image=nginx  
pod/imperative-nginx created  
[hawila@localhost ~]$
```

2- Create a pod with the name `webserver-image"nginx123"`



The screenshot shows a terminal window with two panes. The top pane displays the contents of a file named `deploy.yaml` in a vim editor. The file defines a Kubernetes Deployment with the following configuration:

```
apiVersion: v1
kind: Pod
metadata:
  name: webserver
spec:
  containers:
  - name: nginx
    image: nginx:1.23
    ports:
    - containerPort: 80
```

The bottom pane shows the terminal commands and output:

```
[hawila@localhost ~]$ vim deploy.yaml
[hawila@localhost ~]$ kubectl apply -f deploy.yaml
pod/webserver created
[hawila@localhost ~]$
```

3- What is the nginx pod status?

```
hawila@localhost:~$ kubectl get pod minikube -o yaml
Restart Count: 0
Environment: <none>
Mounts:
  /var/run/secrets/kubernetes.io/serviceaccount from kube-api-access-6m6rw (ro)
Conditions:
  Type             Status
  Initialized       True
  Ready             True
  ContainersReady   True
  PodScheduled      True
Volumes:
  kube-api-access-6m6rw:
    Type: Projected (a volume that contains injected data from multiple sources)
    TokenExpirationSeconds: 3607
    ConfigMapName: kube-root-ca.crt
    ConfigMapOptional: <nil>
    DownwardAPI: true
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   6m55s  default-scheduler  Successfully assigned default/webserver to minikube
  Normal   Pulling     6m49s  kubelet        Pulling image "nginx:1.23"
  Normal   Pulled      6m46s  kubelet        Successfully pulled image "nginx:1.23" in 2.755088708s
  Normal   Created     6m46s  kubelet        Created container nginx
  Normal   Started     6m42s  kubelet        Started container nginx
[hawila@localhost ~]$
```

4- Change the nginx pod image to "nginx" check the status again

```
hawila@localhost:~$ kubectl get pod minikube -o yaml
Mounts:
  /var/run/secrets/kubernetes.io/serviceaccount from kube-api-access-6m6rw (ro)
Conditions:
  Type             Status
  Initialized       True
  Ready             True
  ContainersReady   True
  PodScheduled      True
Volumes:
  kube-api-access-6m6rw:
    Type: Projected (a volume that contains injected data from multiple sources)
    TokenExpirationSeconds: 3607
    ConfigMapName: kube-root-ca.crt
    ConfigMapOptional: <nil>
    DownwardAPI: true
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   8m6s   default-scheduler  Successfully assigned default/webserver to minikube
  Normal   Pulling     8m      kubelet        Pulling image "nginx:1.23"
  Normal   Pulled      7m57s  kubelet        Successfully pulled image "nginx:1.23" in 2.755088708s
  Normal   Killing     13s    kubelet        Container nginx definition changed, will be restarted
  Normal   Pulled      12s    kubelet        Container image "nginx" already present on machine
  Normal   Created     10s (x2 over 7m57s)  kubelet        Created container nginx
  Normal   Started     9s (x2 over 7m53s)  kubelet        Started container nginx
[hawila@localhost ~]$
```

5- How many pods are running in the system? Type the command to show this

```
hawila@localhost:~$ kubectl get pods -A
NAMESPACE   NAME                                READY   STATUS    RESTARTS   AGE
default     imperative-nginx                   1/1     Running   0           21m
default     webserver                          1/1     Running   1 (107s ago)  9m40s
kube-system coredns-565d847f94-htbtlh         1/1     Running   3 (35m ago)  20h
kube-system etcd-minikube                      1/1     Running   3 (40m ago)  20h
kube-system kube-apiserver-minikube 1/1     Running   3 (35m ago)  20h
kube-system kube-controller-manager-minikube 1/1     Running   6 (40m ago)  20h
kube-system kube-proxy-tmcwb         1/1     Running   3           20h
kube-system kube-scheduler-minikube  1/1     Running   3 (40m ago)  20h
kube-system storage-provisioner      1/1     Running   6 (32m ago)  20h
[hawila@localhost ~]$
```

6- What does READY column in the output of get pods

Determine if the pod is ready to perform operation

7- Delete first pod named imperative-nginx you just created. Type the command to do this

```
hawila@localhost:~$ kubectl get pods -A
NAMESPACE   NAME                                     READY   STATUS    RESTARTS   AGE
default     imperative-nginx                       1/1     Running   0           21m
default     webserver                             1/1     Running   1 (107s ago) 9m40s
kube-system coredns-565d847f94-htblh              1/1     Running   3 (35m ago) 20h
kube-system etcd-minikube                       1/1     Running   3 (40m ago) 20h
kube-system kube-apiserver-minikube     1/1     Running   3 (35m ago) 20h
kube-system kube-controller-manager-minikube 1/1     Running   6 (40m ago) 20h
kube-system kube-proxy-tmcwb           1/1     Running   3           20h
kube-system kube-scheduler-minikube     1/1     Running   3 (40m ago) 20h
kube-system storage-provisioner         1/1     Running   6 (32m ago) 20h
[hawila@localhost ~]$ kubectl delete pod imperative-nginx
pod "imperative-nginx" deleted
[hawila@localhost ~]$ kubectl get pods -A
NAMESPACE   NAME                                     READY   STATUS    RESTARTS   AGE
default     webserver                             1/1     Running   1 (6m50s ago) 14m
kube-system coredns-565d847f94-htblh              1/1     Running   3 (40m ago) 20h
kube-system etcd-minikube                       1/1     Running   3 (45m ago) 20h
kube-system kube-apiserver-minikube             1/1     Running   3 (40m ago) 20h
kube-system kube-controller-manager-minikube     1/1     Running   6 (45m ago) 20h
kube-system kube-proxy-tmcwb                     1/1     Running   3           20h
kube-system kube-scheduler-minikube              1/1     Running   3 (45m ago) 20h
kube-system storage-provisioner                   1/1     Running   6 (37m ago) 20h
[hawila@localhost ~]$
```

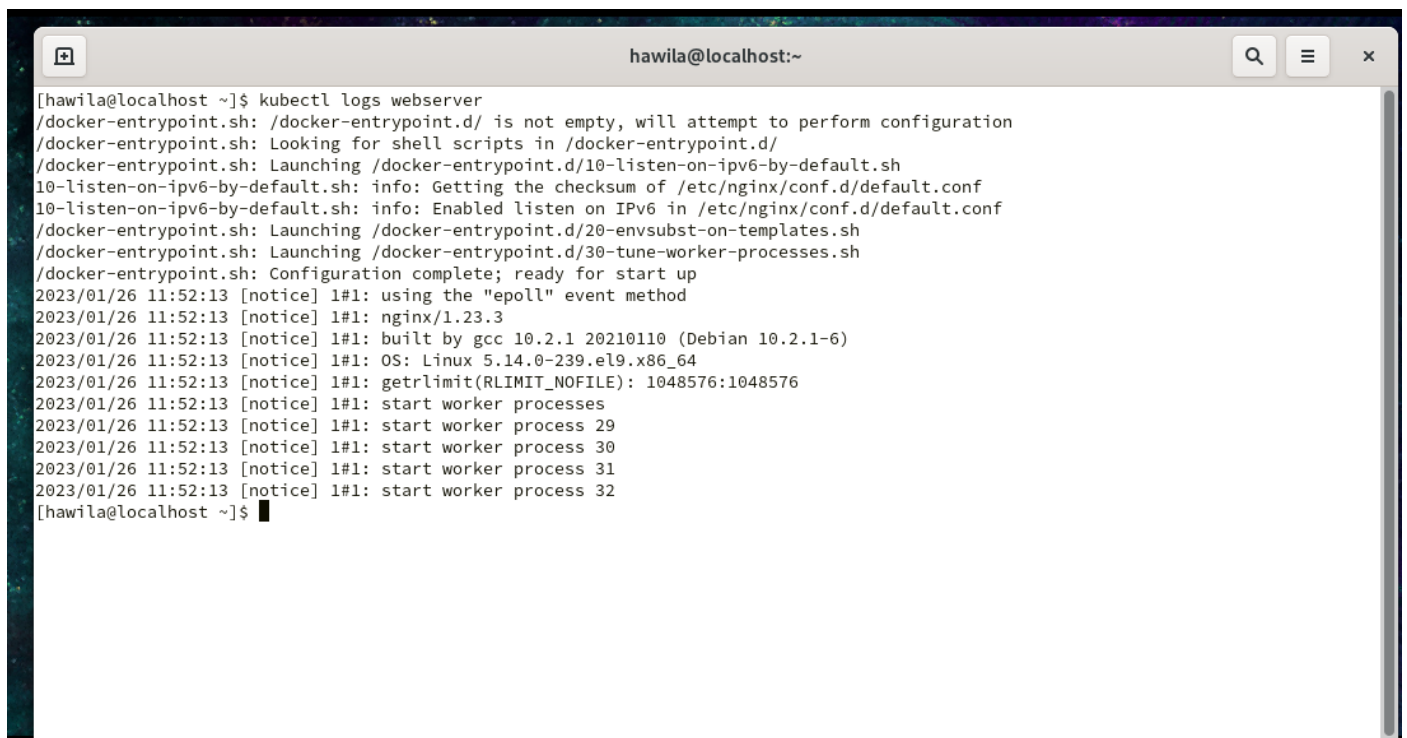
8- Which node is pod named webserver running on (list two commands to do this)

```
hawila@localhost:~$
[hawila@localhost ~]$
[hawila@localhost ~]$
[hawila@localhost ~]$ kubectl get pod webserver -o wide
NAME        READY   STATUS    RESTARTS   AGE   IP        NODE       NOMINATED NODE   READINESS GATES
webserver   1/1     Running   1 (55m ago) 63m   172.17.0.4 minikube   <none>           <none>
[hawila@localhost ~]$
```

- 9- Get a shell to the running container
- 10- Run cat /etc/os-release inside the container
- 11- Exit from the shell (/bin/bash) session

```
[hawila@localhost ~]$ kubectl exec --stdin --tty webserver -- /bin/bash
root@webserver:/# cat /etc/os-release
PRETTY_NAME="Debian GNU/Linux 11 (bullseye)"
NAME="Debian GNU/Linux"
VERSION_ID="11"
VERSION="11 (bullseye)"
VERSION_CODENAME=bullseye
ID=debian
HOME_URL="https://www.debian.org/"
SUPPORT_URL="https://www.debian.org/support"
BUG_REPORT_URL="https://bugs.debian.org/"
root@webserver:/#
```

- 12- Get logs of pod, what are logs and what they are used for?

A terminal window titled 'hawila@localhost:~' showing the output of the command 'kubectl logs webserver'. The logs show the initialization of the container, including the execution of scripts in /docker-entrypoint.d/, configuration of nginx, and the start of worker processes. The logs are timestamped 2023/01/26 11:52:13.

```
[hawila@localhost ~]$ kubectl logs webserver
/docker-entrypoint.sh: /docker-entrypoint.d/ is not empty, will attempt to perform configuration
/docker-entrypoint.sh: Looking for shell scripts in /docker-entrypoint.d/
/docker-entrypoint.sh: Launching /docker-entrypoint.d/10-listen-on-ipv6-by-default.sh
10-listen-on-ipv6-by-default.sh: info: Getting the checksum of /etc/nginx/conf.d/default.conf
10-listen-on-ipv6-by-default.sh: info: Enabled listen on IPv6 in /etc/nginx/conf.d/default.conf
/docker-entrypoint.sh: Launching /docker-entrypoint.d/20-envsubst-on-templates.sh
/docker-entrypoint.sh: Launching /docker-entrypoint.d/30-tune-worker-processes.sh
/docker-entrypoint.sh: Configuration complete; ready for start up
2023/01/26 11:52:13 [notice] 1#1: using the "epoll" event method
2023/01/26 11:52:13 [notice] 1#1: nginx/1.23.3
2023/01/26 11:52:13 [notice] 1#1: built by gcc 10.2.1 20210110 (Debian 10.2.1-6)
2023/01/26 11:52:13 [notice] 1#1: OS: Linux 5.14.0-239.el9.x86_64
2023/01/26 11:52:13 [notice] 1#1: getrlimit(RLIMIT_NOFILE): 1048576:1048576
2023/01/26 11:52:13 [notice] 1#1: start worker processes
2023/01/26 11:52:13 [notice] 1#1: start worker process 29
2023/01/26 11:52:13 [notice] 1#1: start worker process 30
2023/01/26 11:52:13 [notice] 1#1: start worker process 31
2023/01/26 11:52:13 [notice] 1#1: start worker process 32
[hawila@localhost ~]$
```

- 13- How many ReplicaSets exist on the system?

A terminal window titled 'hawila@localhost:~' showing the output of the command 'kubectl get replicaset'. The output indicates that there are 10 resources found in the default namespace.

```
hawila@localhost ~]$ kubectl get replicaset
10 resources found in default namespace.
hawila@localhost ~]$
```

14- create a ReplicaSet
withname= replica-set-1
image= busybox
replicas= 3

```
hawila@localhost:~ — vim rs.yaml
apiVersion: apps/v1
kind: ReplicaSet
metadata:
  name: replica-set-1
spec:
  replicas: 3
  selector:
    matchLabels:
      tier: my-app
  template:
    metadata:
      labels:
        tier: my-app
    spec:
      containers:
        - name: my-app
          image: busybox

~
~
```

```
hawila@localhost:~
[hawila@localhost ~]$ kubectl create -f rs.yaml
replicaset.apps/replica-set-1 created
[hawila@localhost ~]$
```

15- Scale the ReplicaSet replica-set-1 to 5 PODs.

16- How many PODs are READY in the replica-set-1?

```
hawila@localhost:~
[hawila@localhost ~]$ kubectl create -f rs.yaml
replicaset.apps/replica-set-1 created
[hawila@localhost ~]$ kubectl get pods
NAME                READY   STATUS              RESTARTS   AGE
replica-set-1-7bptc  0/1     CrashLoopBackOff    9 (17s ago) 23m
replica-set-1-jwwj5  0/1     CrashLoopBackOff    9 (37s ago) 23m
replica-set-1-sm2dx  0/1     Completed           9 (5m26s ago) 23m
webserver            1/1     Running             1 (101m ago) 109m
[hawila@localhost ~]$ kubectl scale --replicas=5 -f rs.yaml
replicaset.apps/replica-set-1 scaled
[hawila@localhost ~]$ kubectl get pods
NAME                READY   STATUS              RESTARTS   AGE
replica-set-1-7bptc  0/1     CrashLoopBackOff    9 (2m14s ago) 24m
replica-set-1-h7vjx  0/1     ContainerCreating   0           14s
replica-set-1-jwwj5  0/1     CrashLoopBackOff    9 (2m34s ago) 24m
replica-set-1-qprhc  0/1     ContainerCreating   0           14s
replica-set-1-sm2dx  0/1     CrashLoopBackOff    9 (2m8s ago) 24m
webserver            1/1     Running             1 (103m ago) 111m
[hawila@localhost ~]$
```

17- Delete any one of the 5 PODs then check How many PODs exist now?
Why are there still 5 PODs, even after you deleted one?

Because the desired number of replicas is 5 so replica set will automatically initialize pod

```
[hawila@localhost ~]$ kubectl get pods
NAME                READY   STATUS              RESTARTS   AGE
replica-set-1-7bptc  0/1     CrashLoopBackOff    9 (2m14s ago) 24m
replica-set-1-h7vjx  0/1     ContainerCreating   0           14s
replica-set-1-jwwj5  0/1     CrashLoopBackOff    9 (2m34s ago) 24m
replica-set-1-qprhc  0/1     ContainerCreating   0           14s
replica-set-1-sm2dx  0/1     CrashLoopBackOff    9 (2m8s ago) 24m
webserver            1/1     Running             1 (103m ago) 111m
[hawila@localhost ~]$ kubectl delete pod replica-set-1-7bptc
pod "replica-set-1-7bptc" deleted
```