

# Kubernetes avec Minikube

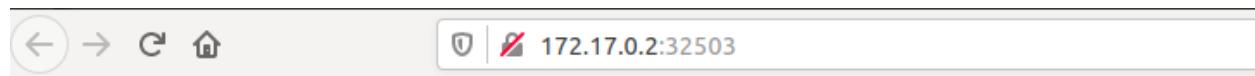
```
omar-devops@omardevops-Lenovo-Y520-15IKBN:~$ minikube start
minikube v1.9.2 sur Ubuntu 18.04
Using the docker driver based on existing profile
Starting control plane node m01 in cluster minikube
Pulling base image ...
Updating the running docker "minikube" container ...
Préparation de Kubernetes v1.18.0 sur Docker 19.03.2...
■ kubeadm.pod-network-cidr=10.244.0.0/16
Installation des addons: default-storageclass, storage-provisioner
Terminé ! kubectl est maintenant configuré pour utiliser "minikube".
Pour des résultats optimaux, installez kubectl à l'adresse suivante : https://kubernetes.io/docs/tasks/tools/install-kubectl/
omar-devops@omardevops-Lenovo-Y520-15IKBN:~$ ls ~/.kube/config
/home/omar-devops/.kube/config
omar-devops@omardevops-Lenovo-Y520-15IKBN:~$ cat ~/.kube/config
apiVersion: v1
clusters:
- cluster:
    certificate-authority: /home/omar-devops/.minikube/ca.crt
    server: https://172.17.0.2:8443
    name: minikube
contexts:
- context:
    cluster: minikube
    user: minikube
    name: minikube
current-context: minikube
kind: Config
preferences: {}
users:
- name: minikube
  user:
    client-certificate: /home/omar-devops/.minikube/profiles/minikube/client.crt
    client-key: /home/omar-devops/.minikube/profiles/minikube/client.key
omar-devops@omardevops-Lenovo-Y520-15IKBN:~$ minikube status
m01
host: Running
kubelet: Running
apiserver: Running
kubeconfig: Configured

omar-devops@omardevops-Lenovo-Y520-15IKBN:~$ [ ]
omar-devops@omardevops-Lenovo-Y520-15IKBN:~$ ##### install kubectl #####
omar-devops@omardevops-Lenovo-Y520-15IKBN:~$ curl -LO https://storage.googleapis.com/kubernetes-release/release/$(curl -s https://storage.googleapis.com/kubernetes-release/release/stable.txt)/bin/linux/amd64/kubectl
% Total    % Received % Xferd  Average Speed   Time   Time  Current
          Dload  Upload Total Spent   Left Speed
100 41.9M  100 41.9M    0     0  249k      0  0:02:52  0:02:52 ---:--- 381k
omar-devops@omardevops-Lenovo-Y520-15IKBN:~$ chmod +x ./kubectl
omar-devops@omardevops-Lenovo-Y520-15IKBN:~$ sudo mv ./kubectl /usr/local/bin/kubectl
[sudo] Mot de passe de omar-devops :
omar-devops@omardevops-Lenovo-Y520-15IKBN:~$ kubectl version --client
Client Version: version.Info{Major:"1", Minor:"18", GitVersion:"v1.18.1", GitCommit:"7879fc12a63337efff607952a323df90cdc7a335", GitTreeState:"clean", BuildDate:"2020-04-08T17:38:50Z", GoVersion:"go1.13.9", Compiler:"gc", Platform:"linux/amd64"}
omar-devops@omardevops-Lenovo-Y520-15IKBN:~$ kubectl
kubectl controls the Kubernetes cluster manager.

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

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
```

```
omar-devops@omardevops-Lenovo-Y520-15IKBN:~$ ### first run to test    kubectl ##"
omar-devops@omardevops-Lenovo-Y520-15IKBN:~$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~$ kubectl create deployment hello-minikube --image=k8s.gcr.io/echoserver:1.10
deployment.apps/hello-minikube created
omar-devops@omardevops-Lenovo-Y520-15IKBN:~$ ### expose port ###
omar-devops@omardevops-Lenovo-Y520-15IKBN:~$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~$ kubectl expose deployment hello-minikube --type=NodePort --port=8080
service/hello-minikube exposed
omar-devops@omardevops-Lenovo-Y520-15IKBN:~$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~$ ## Le pod "hello-minikube" est lancé ##
omar-devops@omardevops-Lenovo-Y520-15IKBN:~$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~$ ## on vérifie si le pod est opérationnel ###
omar-devops@omardevops-Lenovo-Y520-15IKBN:~$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~$ kubectl get pod
NAME                  READY   STATUS      RESTARTS   AGE
hello-minikube        1/1     Running    0          5m27s
hello-minikube-64b64df8c9-2hm7t  0/1     ContainerCreating   0          111s
omar-devops@omardevops-Lenovo-Y520-15IKBN:~$ ### Notre pod est en cours de création ###
omar-devops@omardevops-Lenovo-Y520-15IKBN:~$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~$ kubectl get pod
NAME                  READY   STATUS      RESTARTS   AGE
hello-minikube        1/1     Running    0          7m31s
hello-minikube-64b64df8c9-2hm7t  1/1     Running    0          3m55s
omar-devops@omardevops-Lenovo-Y520-15IKBN:~$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~$ ## voilà ! ## ■
omar-devops@omardevops-Lenovo-Y520-15IKBN:~$ ### Maintenant Obtenir l'URL ###
omar-devops@omardevops-Lenovo-Y520-15IKBN:~$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~$ minikube service hello-minikube --url
http://172.17.0.2:32503
```



Hostname: hello-minikube-64b64df8c9-2hm7t

Pod Information:  
-no pod information available-

Server values:  
server\_version=nginx: 1.13.3 - lua: 10008

Request Information:  
client\_address=172.18.0.1  
method=GET  
real\_path=/  
query=  
request\_version=1.1  
request\_scheme=http  
request\_uri=http://172.17.0.2:8080/

Request Headers:  
accept=text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,\*/\*;q=0.8  
accept-encoding=gzip, deflate  
accept-language=fr,fr-FR;q=0.8,en-US;q=0.5,en;q=0.3  
connection=keep-alive  
host=172.17.0.2:32503  
upgrade-insecure-requests=1  
user-agent=Mozilla/5.0 (X11; Ubuntu; Linux x86\_64; rv:75.0) Gecko/20100101 Firefox/75.0

Request Body:  
-no body in request-

```
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app$ cat Dockerfile
FROM node:12
WORKDIR /app
ADD . /app
RUN npm install
EXPOSE 3000
CMD npm start
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app$ cat index.js
var express = require('express');
var app = express();

app.get('/', function (req, res) {
  res.send('Hello World!');
});

var server = app.listen(3000, function () {
  var host = server.address().address;
  var port = server.address().port;

  console.log('Example app listening at http://%s:%s', host, port);
});
```

```

omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app$ cat package.json
{
  "name": "myapp",
  "version": "0.0.1",
  "private": true,
  "scripts": {
    "start": "node index.js",
    "test": "npm install inherits && mocha"
  },
  "engines": {
    "node": "^4.6.1"
  },
  "dependencies": {
    "express": "^4.14.0",
    "http-errors": "^1.7.0",
    "mysql": "^2.10.2"
  },
  "devDependencies": {
    "inherits": "^2.0.3",
    "mocha": "^5.2.0",
    "ms": "^2.1.1"
  }
}

omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app$ ## create dockerfile & index.js & package.json ##
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app$ ll
total 48
drwxr-xr-x 2 omar-devops omar-devops 4096 avril 10 11:12 .
drwxrwxr-x 4 omar-devops omar-devops 4096 avril 10 10:42 ..
-rwxrwxrwx 1 root      root      79 avril 10 10:45 Dockerfile*
-rwxrwxrwx 1 root      root      316 avril 10 10:45 index.js*
-rwxrwxrwx 1 root      root      397 avril 10 10:46 package.json*
-rwxrwxrwx 1 root      root      26116 avril 10 11:12 package-lock.json*
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app$ ## build image #####
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app$ docker build .
Sending build context to Docker daemon 31.23kB
Step 1/6 : FROM node:12
--> 0c601cba9f11
Step 2/6 : WORKDIR /app
--> Using cache
--> ad2804904f99
Step 3/6 : ADD . /app
--> e58b89156aa8
Step 4/6 : RUN npm install
--> Running in 0d061c86bc5f
added 94 packages from 485 contributors and audited 171 packages in 5.037s
found 1 low severity vulnerability
  run `npm audit fix` to fix them, or `npm audit` for details
Removing intermediate container 0d061c86bc5f
--> ac3c20daaa6f
Step 5/6 : EXPOSE 3000
--> Running in dc34c1c848db
Removing intermediate container dc34c1c848db
--> cd754660ce12
Step 6/6 : CMD npm start
--> Running in 5788288780b4
Removing intermediate container 5788288780b4
--> 1a4a120faa46
Successfully built 1a4a120faa46
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app$ ## run image #####
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app$ docker run -p 3000:3000 -it 1a4a120faa46
> myapp@0.0.1 start /app
> node index.js

Example app listening at http://:::3000

```

```

omar-devops@omardevops-Lenovo-Y520-15IKBN:~$ ### test hello world with nodejs ###
omar-devops@omardevops-Lenovo-Y520-15IKBN:~$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~$ curl localhost:3000
Hello World!omar-devops@omardevops-Lenovo-Y520-15IKBN:~$ 

omar-devops@omardevops-Lenovo-Y520-15IKBN:~$ ## to make image available to kubernetes, you need to push the image
to docker Registry, like Docker hub , ECR, etc ##
omar-devops@omardevops-Lenovo-Y520-15IKBN:~$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~$ ## connect with docker registry ##
omar-devops@omardevops-Lenovo-Y520-15IKBN:~$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~$ docker login
Login with your Docker ID to push and pull images from Docker Hub. If you don't have a Docker ID, head over to https://hub.docker.com to create one.
Username: iddocker98
Password:
WARNING! Your password will be stored unencrypted in /home/omar-devops/.docker/config.json.
Configure a credential helper to remove this warning. See
https://docs.docker.com/engine/reference/commandline/login/#credentials-store

Login Succeeded
omar-devops@omardevops-Lenovo-Y520-15IKBN:~$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~$ ### tag image ###
omar-devops@omardevops-Lenovo-Y520-15IKBN:~$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~$ docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED            STATUS              PORTS               NAMES
c73c103a413f        1a4a120faa46      "docker-entrypoint.s..."   14 minutes ago   Up 14 minutes    0.0.0.0:3000->3000/tcp   agitated_swanson
omar-devops@omardevops-Lenovo-Y520-15IKBN:~$ docker tag 1a4a120faa46 iddocker98/k8s-demo-nodejs
omar-devops@omardevops-Lenovo-Y520-15IKBN:~$ docker images
REPOSITORY          TAG                 IMAGE ID            CREATED           SIZE
iddocker98/k8s-demo-nodejs   latest              1a4a120faa46      47 minutes ago   923MB
node                12                  0c661cba9f11      15 hours ago    916MB
postgres            latest              9907cacfc001      9 days ago     314MB
gcr.io/k8s-minikube/kicbase   v0.0.8              11589cdcc9e4      2 weeks ago    964MB
738739307270.dkr.ecr.eu-west-3.amazonaws.com/medilib-booking-query   latest              43000f65704b      3 months ago   188MB
738739307270.dkr.ecr.eu-west-3.amazonaws.com/medilib-carrier-command   latest              43000f65704b      3 months ago   188MB
738739307270.dkr.ecr.eu-west-3.amazonaws.com/medilib-notification   latest              43000f65704b      3 months ago   188MB
738739307270.dkr.ecr.eu-west-3.amazonaws.com/medilib-oauth2       latest              43000f65704b      3 months ago   188MB
738739307270.dkr.ecr.eu-west-3.amazonaws.com/medilib-user-query   latest              43000f65704b      3 months ago   188MB
738739307270.dkr.ecr.eu-west-3.amazonaws.com/medilib-zuul        latest              43000f65704b      3 months ago   188MB
hello-world         latest              43000f65704b      3 months ago   188MB
738739307270.dkr.ecr.eu-west-3.amazonaws.com/medilib-booking-command   latest              43000f65704b      3 months ago   188MB
738739307270.dkr.ecr.eu-west-3.amazonaws.com/medilib-carrier-query   latest              43000f65704b      3 months ago   188MB
738739307270.dkr.ecr.eu-west-3.amazonaws.com/medilib-customer-command   latest              43000f65704b      3 months ago   188MB
738739307270.dkr.ecr.eu-west-3.amazonaws.com/medilib-customer-query   latest              43000f65704b      3 months ago   188MB
738739307270.dkr.ecr.eu-west-3.amazonaws.com/medilib-user-command   latest              43000f65704b      3 months ago   188MB
ubuntu              18.04              549b9b86cb8d      3 months ago   64.2MB
omar-devops@omardevops-Lenovo-Y520-15IKBN:~$ 

omar-devops@omardevops-Lenovo-Y520-15IKBN:~$ ### push image ###
omar-devops@omardevops-Lenovo-Y520-15IKBN:~$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~$ docker push iddocker98/k8s-demo-nodejs
The push refers to repository [docker.io/iddocker98/k8s-demo-nodejs]
3b4c7417c1c3: Pushed
1afbfc64b466: Pushed
9f6ede181cca: Pushed
217b797e3844: Mounted from library/node
5036395203c1: Mounted from library/node
be799ebe96b7: Mounted from library/node
45ac74adb5b4: Mounted from library/node
d485cbbe6a5e: Mounted from library/node
391c89959588: Mounted from library/node
588545a7a2a3: Mounted from library/node
8452468a5e50: Mounted from library/node
55b19a5e648f: Mounted from library/node
latest: digest: sha256:ab6acbcefce664e5b63a210f4e7f6c8a29ad5b28595a314c8a41e8ea6738e2bd0 size: 2841
omar-devops@omardevops-Lenovo-Y520-15IKBN:~$ 

```

```

omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernetes/first_app/k8s$ ### first create pod ##
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernetes/first_app/k8s$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernetes/first_app/k8s$ ## create helloworld.yml ####
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernetes/first_app/k8s$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernetes/first_app/k8s$ sudo nano helloworld.yml
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernetes/first_app/k8s$ cat helloworld.yml
apiVersion: v1
kind: Pod
metadata:
  name: nodehelloworld.com
  labels:
    app: helloworld
spec:
  containers:
  - name: k8s-demo
    image: iddocker98/k8s-demo-nodejs
    ports:
    - name: nodejs-port
      containerPort: 3000
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernetes/first_app/k8s$ ### create pod with kubectl ####
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernetes/first_app/k8s$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernetes/first_app/k8s$ kubectl create -f helloworld.yml
pod/nodehelloworld.com created
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernetes/first_app$ kubectl get pod
NAME           READY   STATUS    RESTARTS   AGE
hello-minikube 1/1     Running   1          17h
hello-minikube-64b64df8c9-2hm7t 1/1     Running   1          17h
nodehelloworld.example.com 1/1     Running   0          24m
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernetes/first_app$ kubectl describe pod nodehelloworld.example.com
Name:           nodehelloworld.example.com
Namespace:      default
Priority:       0
Node:          minikube/172.17.0.3
Start Time:    Fri, 10 Apr 2020 13:14:10 +0100
Labels:         app=helloworld
Annotations:   <none>
Status:        Running
IP:            172.18.0.6
IPs:
  IP: 172.18.0.6
Containers:
  k8s-demo:
    Container ID: docker://5ff6fd88490682aa24d616a4e41b46a7bf432cb15529b64363336b0eaf2a7495
    Image:         iddocker98/k8s-demo-nodejs
    Image ID:     docker-pullable://iddocker98/k8s-demo-nodejs@sha256:ab6acbefce664e5b63a210f4e7f6c8a29ad5b28595a314c8a
41e8ea6738e2bd0
    Port:         3000/TCP
    Host Port:   0/TCP
    State:       Running
      Started:   Fri, 10 Apr 2020 13:35:58 +0100
    Ready:        True
    Restart Count: 0
    Environment: <none>
    Mounts:
      /var/run/secrets/kubernetes.io/serviceaccount from default-token-qwsp (ro)
Conditions:
  Type      Status
  Initialized  True
  Ready      True
  ContainersReady  True
  PodScheduled  True
Volumes:
  default-token-qwsp:
    Type:      Secret (a volume populated by a Secret)
    SecretName: default-token-qwsp
    Optional:   false
QoS Class:  BestEffort
Node-Selectors: <none>
Tolerations:  node.kubernetes.io/not-ready:NoExecute for 300s
               node.kubernetes.io/unreachable:NoExecute for 300s
Events:
  Type  Reason  Age   From          Message
  ----  -----  ---   ----          -----
  Normal Scheduled  24m  default-scheduler  Successfully assigned default/nodehelloworld.example.com to minikube
  Normal Pulling   24m  kubelet, minikube  Pulling image "iddocker98/k8s-demo-nodejs"
  Normal Pulled    3m7s  kubelet, minikube  Successfully pulled image "iddocker98/k8s-demo-nodejs"
  Normal Created   2m56s  kubelet, minikube  Created container k8s-demo
  Normal Started   2m47s  kubelet, minikube  Started container k8s-demo

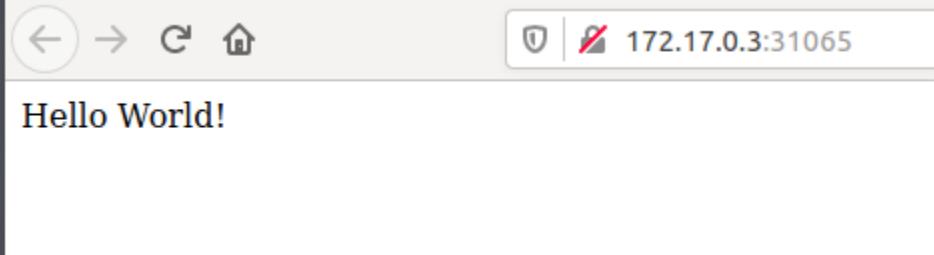
```

```

omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app$ ### use port-forward ####
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app$ kubectl port-forward nodehelloworld.example.com 8081:3000
Forwarding from 127.0.0.1:8081 -> 3000
Forwarding from [::1]:8081 -> 3000
Handling connection for 8081
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s$ curl localhost:8081
Hello World!omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s$ 

omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app$ ### expose pod ####
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app$ kubectl expose pod nodehelloworld.example.com --type=NodePort --name
e nodehelloworld-service
service/nodehelloworld-service exposed
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app$ ### url & ip to connecte ####
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app$ minikube service nodehelloworld-service --url
http://172.17.0.3:31065
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app$ ## get service ##
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app$ kubectl get service
NAME          TYPE      CLUSTER-IP    EXTERNAL-IP   PORT(S)        AGE
hello-minikube   NodePort  10.110.40.234  <none>       8080:32503/TCP  18h
kubernetes       ClusterIP 10.96.0.1    <none>       443/TCP       18h
nodehelloworld-service   NodePort  10.97.166.140  <none>       3000:31065/TCP  5m32s
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app$ 

```



```

omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app$ ### execute some cmd with kubectl ####
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app$ kubectl get service
NAME          TYPE      CLUSTER-IP    EXTERNAL-IP   PORT(S)        AGE
hello-minikube   NodePort  10.110.40.234  <none>       8080:32503/TCP  18h
kubernetes       ClusterIP 10.96.0.1    <none>       443/TCP       19h
nodehelloworld-service   NodePort  10.97.166.140  <none>       3000:31065/TCP  37m
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app$ ## kubectl attach to see process ##
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app$ kubectl attach nodehelloworld.example.com
Defaulting container name to k8s-demo.
Use 'kubectl describe pod/nodehelloworld.example.com -n default' to see all of the containers in this pod.
If you don't see a command prompt, try pressing enter.
^C
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app$ ## execute cmd in container ##
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app$ kubectl exec nodehelloworld.example.com -- ls /app
Dockerfile
index.js
node_modules
package-lock.json
package.json

```

```

omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s$ ### horizontally scale a pod with replication controller ##
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s$ ## create replication controller ##
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s$ cat helloworld-repl-controller.yml
apiVersion: v1
kind: ReplicationController
metadata:
  name: helloworld-controller
spec:
  replicas: 2
  selector:
    app: helloworld
  template:
    metadata:
      labels:
        app: helloworld
    spec:
      containers:
        - name: k8s-demo
          image: iddocker98/k8s-demo-nodejs
        ports:
          - name: nodejs-port
            containerPort: 3000
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s$ ## create replication controller ##
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s$ kubectl create -f helloworld-repl-controller.yml
replicationcontroller/helloworld-controller created
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s$ kubectl get pods
NAME           READY   STATUS    RESTARTS   AGE
hello-minikube 1/1     Running   1          20h
hello-minikube-64b64df8c9-lgz77 1/1     Running   0          7m26s
helloworld-controller-mcppn 0/1     ContainerCreating  0          15s
nodehelloworld.example.com 1/1     Running   0          3h19m
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s$ kubectl get rc
NAME       DESIRED   CURRENT   READY   AGE
helloworld-controller 2         2         1       32s
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s$ kubectl describe pod helloworld-controller-mcppn
Name:           helloworld-controller-mcppn
Namespace:      default
Priority:       0
Node:          minikube/172.17.0.3
Start Time:    Fri, 10 Apr 2020 16:33:40 +0100
Labels:         app=helloworld
Annotations:   <none>
Status:        Running
IP:            172.18.0.7
IPs:
  IP:          172.18.0.7
Controlled By: ReplicationController/helloworld-controller
Containers:
  k8s-demo:
    Container ID: docker://8242d2c96355e79a1b4a7093afa69606bcba3bb8160ab2803496ca85ef502bef
    Image:          iddocker98/k8s-demo-nodejs
    Image ID:      docker-pullable://iddocker98/k8s-demo-nodejs@sha256:ab0acbefce664e5b63a210f4e7f6c8a29ad5b28595a314c8a41e8ea6738e2bd0
    Port:          3000/TCP
    Host Port:    0/TCP
    State:        Running
      Started:   Fri, 10 Apr 2020 16:34:13 +0100
    Ready:        True
    Restart Count: 0
    Environment:  <none>
    Mounts:
      /var/run/secrets/kubernetes.io/serviceaccount from default-token-qwsp (ro)
Conditions:
  Type      Status
  Initialized  True
  Ready      True
  ContainersReady  True
  PodScheduled  True
Volumes:
  default-token-qwsp:
    Type:      Secret (a volume populated by a Secret)
    SecretName: default-token-qwsp
    Optional:   false
QoS Class:  BestEffort
Node-Selectors:  <none>
Tolerations:   node.kubernetes.io/not-ready:NoExecute for 300s
               node.kubernetes.io/unreachable:NoExecute for 300s
Events:
  Type  Reason  Age   From          Message
  ----  -----  ---   ----          -----
  Normal Scheduled 9m54s  default-scheduler  Successfully assigned default/helloworld-controller-mcppn to minikube
  Normal Pulling  9m33s  kubelet, minikube  Pulling image "iddocker98/k8s-demo-nodejs"
  Normal Pulled   9m30s  kubelet, minikube  Successfully pulled image "iddocker98/k8s-demo-nodejs"
  Normal Created   9m22s  kubelet, minikube  Created container k8s-demo
  Normal Started   9m18s  kubelet, minikube  Started container k8s-demo

```

```

omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s$ kubectl describe rc helloworld-controller
Name:      helloworld-controller
Namespace: default
Selector:  app=helloworld
Labels:    app=helloworld
Annotations: <none>
Replicas:  2 current / 2 desired
Pods Status: 2 Running / 0 Waiting / 0 Succeeded / 0 Failed
Pod Template:
  Labels:  app=helloworld
  Containers:
    k8s-demo:
      Image:      iddocker98/k8s-demo-nodejs
      Port:       3000/TCP
      Host Port: 0/TCP
      Environment: <none>
      Mounts:    <none>
      Volumes:   <none>
  Events:
    Type  Reason     Age   From           Message
    ----  -----     --   --   -----
    Normal  SuccessfulCreate  15m  replication-controller  Created pod: helloworld-controller-mcppn
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s$ ### scale replication ###
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s$ kubectl scale --replicas=4 -f helloworld-repl-controller.yml
replicationcontroller/helloworld-controller scaled
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s$ kubectl get rc
NAME        DESIRED  CURRENT  READY   AGE
helloworld-controller  4        4        2       24m
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s$ kubectl get pods
NAME                READY  STATUS    RESTARTS  AGE
hello-minikube      1/1    Running   1         21h
hello-minikube-64b64df8c9-tl5pb  1/1    Running   0         12m
helloworld-controller-m764s  0/1    ContainerCreating  0         23s
helloworld-controller-sjfc8  1/1    Running   0         7m58s
helloworld-controller-tgjzh  0/1    ContainerCreating  0         23s
nodehelloworld.example.com  1/1    Running   0         3h44m
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s$ ## on peut utilise le nom de replication 2 eme methode ##
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s$ kubectl scale --replicas=1 rc/helloworld-controller
replicationcontroller/helloworld-controller scaled
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s$ kubectl get rc
NAME        DESIRED  CURRENT  READY   AGE
helloworld-controller  1        1        1       27m
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s$ kubectl get pods
NAME                READY  STATUS    RESTARTS  AGE
hello-minikube      1/1    Running   1         21h
hello-minikube-64b64df8c9-tl5pb  1/1    Running   0         15m
helloworld-controller-m764s  1/1    Terminating  0         3m36s
helloworld-controller-sjfc8  1/1    Terminating  0         11m
helloworld-controller-tgjzh  1/1    Terminating  0         3m36s
nodehelloworld.example.com  1/1    Running   0         3h47m
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s$ ## delete replication ##
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s$ kubectl delete rc/helloworld-controller
replicationcontroller "helloworld-controller" deleted
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s$ kubectl get pods
NAME                READY  STATUS    RESTARTS  AGE
hello-minikube      1/1    Running   1         21h
hello-minikube-64b64df8c9-tl5pb  1/1    Running   0         17m
nodehelloworld.example.com  1/1    Terminating  0         3h49m
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s$ kubectl get rc
No resources found in default namespace.
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s$ 

```

```

omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ ##### let's go to the deployments ##
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ ### create deployment file ##
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ cat helloworld.yml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: helloworld-deployment
spec:
  replicas: 3
  selector:
    matchLabels:
      app: helloworld
  template:
    metadata:
      labels:
        app: helloworld
  spec:
    containers:
      - name: k8s-demo
        image: iddocker98/k8s-demo-nodejs
        ports:
          - name: nodejs-port
            containerPort: 3000

omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ ## create deployment ##
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ kubectl create -f helloworld.yml
deployment.apps/helloworld-deployment created
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ kubectl get deployments
NAME           READY   UP-TO-DATE   AVAILABLE   AGE
hello-minikube 1/1     1           1           21h
helloworld-deployment 0/3     3           0           22s
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ kubectl get deployments
NAME           READY   UP-TO-DATE   AVAILABLE   AGE
hello-minikube 1/1     1           1           22h
helloworld-deployment 3/3     3           3           42m
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ ### show replicas set ##
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ kubectl get rs
NAME           DESIRED   CURRENT   READY   AGE
hello-minikube-64b64df8c9 1         1         1         136m
helloworld-deployment-56bfb7695 3         3         3         72m
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ kubectl get pods
NAME           READY   STATUS   RESTARTS   AGE
hello-minikube 1/1     Running  1          22h
hello-minikube-64b64df8c9-tl5pb 1/1     Running  0          115m
helloworld-deployment-56bfb7695-74zrf 1/1     Running  0          73m
helloworld-deployment-56bfb7695-8snzg 1/1     Running  0          73m
helloworld-deployment-56bfb7695-ccqxm 1/1     Running  0          73m
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ ## show lables ##
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ kubectl get pods --show-lables
Error: unknown flag: --show-lables
See 'kubectl get --help' for usage.
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ kubectl get pods --show-labels
NAME           READY   STATUS   RESTARTS   AGE   LABELS
hello-minikube 1/1     Running  1          22h   run=hello-minikube
hello-minikube-64b64df8c9-tl5pb 1/1     Running  0          116m  app=hello-minikube,pod-template-hash=64b64df8c9
helloworld-deployment-56bfb7695-74zrf 1/1     Running  0          74m   app=helloworld,pod-template-hash=56bfb7695
helloworld-deployment-56bfb7695-8snzg 1/1     Running  0          74m   app=helloworld,pod-template-hash=56bfb7695
helloworld-deployment-56bfb7695-ccqxm 1/1     Running  0          74m   app=helloworld,pod-template-hash=56bfb7695
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ kubectl rollout status deployment/helloworld-deployment
deployment "helloworld-deployment" successfully rolled out
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ ### expose deployment ##
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ kubectl expose deployment helloworld-deployment --type=NodePort
service/helloworld-deployment exposed
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ kubectl get service
NAME           TYPE      CLUSTER-IP   EXTERNAL-IP   PORT(S)   AGE
hello-minikube  NodePort  10.110.40.234 <none>       8080:32503/TCP 23h
helloworld-deployment  NodePort  10.108.174.208 <none>       3000:31430/TCP 31s
kubernetes      ClusterIP  10.96.0.1    <none>       443/TCP      23h
nodehelloworld-service  NodePort  10.97.166.140 <none>       3000:31065/TCP 5h15m
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ ## voilà ! service create ##
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ 

```

```

omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app$ docker tag 01bb61216239 iddocker98/k8s-demo-nodejs:2
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app$ docker images
REPOSITORY          TAG        IMAGE ID      CREATED       SIZE
iddocker98/k8s-demo-nodejs   2          01bb61216239  3 minutes ago  923MB
iddocker98/exampleapp-ruby   k8s       25f098aef5a2  47 hours ago  923MB
exampleapp-ruby          latest     25f098aef5a2  47 hours ago  923MB
iddocker98/k8s-demo-nodejs   latest     1a4a120faa46  4 days ago   923MB
node                 12         0c601cba9f11  4 days ago   916MB
postgres             latest     9907cacf0c01  13 days ago   314MB
gcr.io/k8s-minikube/kicbase v0.0.8    11589cdc9ef4  3 weeks ago   964MB
hello-world           latest     43000f65704b  3 months ago  188MB
ubuntu               18.04     549b9b8ccb8d  3 months ago  64.2MB
ruby                 2.6.2     8d6721e9290e  12 months ago  870MB
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app$ ## push new version ###
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app$ docker push iddocker98/k8s-demo-nodejs:2
The push refers to repository [docker.io/iddocker98/k8s-demo-nodejs]
2e6d00a0589e: Pushed
86649ae5dc14: Pushed
9f6ede181cca: Layer already exists
217b797e3844: Layer already exists
5036395203c1: Layer already exists
be799ebe96b7: Layer already exists
45ac74adb5b4: Layer already exists
d485cbbe6a5e: Layer already exists
391c89959588: Layer already exists
588545a7a2a3: Layer already exists
8452468a5e50: Layer already exists
55b19a5e648f: Layer already exists
2: digest: sha256:3f73ec4cf253433d4bd99c48d3e36bf409492bef2c32f18ee24ca901ee460b99 size: 2841
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ kubectl rollout status deployment/helloworld-deployment
deployment "helloworld-deployment" successfully rolled out
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ kubectl expose deployment helloworld-deployment --type=NodePort
Error from server (AlreadyExists): services "helloworld-deployment" already exists
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ minikube service helloworld-deployment --url
http://172.17.0.2:30986
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ curl http://172.17.0.2:30986
Hello World!omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ kubectl set image deployment/helloworld-deployment k8s-demo=iddocker98/k8s-demo-nodejs:2
deployment.apps/helloworld-deployment image updated
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ kubectl rollout status deployment/helloworld-deployment
Waiting for deployment "helloworld-deployment" rollout to finish: 1 out of 3 new replicas have been updated...
Waiting for deployment "helloworld-deployment" rollout to finish: 1 out of 3 new replicas have been updated...
Waiting for deployment "helloworld-deployment" rollout to finish: 2 out of 3 new replicas have been updated...
Waiting for deployment "helloworld-deployment" rollout to finish: 2 out of 3 new replicas have been updated...
Waiting for deployment "helloworld-deployment" rollout to finish: 2 out of 3 new replicas have been updated...
Waiting for deployment "helloworld-deployment" rollout to finish: 1 old replicas are pending termination...
Waiting for deployment "helloworld-deployment" rollout to finish: 1 old replicas are pending termination...
deployment "helloworld-deployment" successfully rolled out
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ curl http://172.17.0.2:30986
Hello World! version 2omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ ### show rollout history ####
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ kubectl rollout history deployment/helloworld-deployment
deployment.apps/helloworld-deployment
REVISION  CHANGE-CAUSE
1          <none>
2          <none>

omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ ### rollout undo back to version: 1 ####
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ kubectl rollout undo deployment/helloworld-deployment
deployment.apps/helloworld-deployment rolled back
Hello World! version 2omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ kubectl rollout status deployment/helloworld-deployment
Waiting for deployment "helloworld-deployment" rollout to finish: 2 out of 3 new replicas have been updated...
Waiting for deployment "helloworld-deployment" rollout to finish: 2 out of 3 new replicas have been updated...
Waiting for deployment "helloworld-deployment" rollout to finish: 2 out of 3 new replicas have been updated...
Waiting for deployment "helloworld-deployment" rollout to finish: 1 old replicas are pending termination...
Waiting for deployment "helloworld-deployment" rollout to finish: 1 old replicas are pending termination...
deployment "helloworld-deployment" successfully rolled out
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ kubectl get pods
NAME            READY   STATUS    RESTARTS   AGE
consul-consul-qjst6   1/1    Running   2          2d13h
consul-consul-server-0   1/1    Running   2          2d13h
helloworld-deployment-56bfb7695-dsxwf  1/1    Running   6          13m
helloworld-deployment-56bfb7695-kb7gr  1/1    Running   0          13m
helloworld-deployment-56bfb7695-spkvd  1/1    Running   0          12m
vault_0          0/1    Running   2          2d13h
vault_1          0/1    Running   2          2d13h
vault_2          0/1    Running   2          2d13h
vault-agent-injector-ccc7d98cc-v4jcr  1/1    Running   4          2d13h
webapp-56499bb758-g7g92   1/1    Running   1          46h
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ curl http://172.17.0.2:30986
Hello World!omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ 
Hello World!omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ kubectl rollout history deployment/helloworld-deployment
deployment.apps/helloworld-deployment
REVISION  CHANGE-CAUSE
2          <none>
3          <none>

omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ ## on peut chager les paramètres ####
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ kubectl edit deployment/helloworld-deployment
Edit cancelled, no changes made.
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ 

```

```

omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ ### create deployment file to show nodeSelector ####
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployments
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ sudo nano helloworld-nodeselector.yml
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ cat helloworld-nodeselector.yml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: helloworld-deployment
spec:
  replicas: 3
  selector:
    matchLabels:
      app: helloworld
  template:
    metadata:
      labels:
        app: helloworld
    spec:
      containers:
        - name: k8s-demo
          image: iddocker98/k8s-demo-nodejs
          ports:
            - name: nodejs-port
              containerPort: 3000
      nodeSelector:
        hardware: high-spec
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ ## run deployment helloworld-nodeselector ####
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ kubectl create -f helloworld-nodeselector.yml
deployment.apps/helloworld-deployment created
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ kubectl get deployments
NAME           READY   UP-TO-DATE   AVAILABLE   AGE
helloworld-deployment   0/3     3           0          6m52s
vault-agent-injector   1/1     1           1          5d17h
webapp               1/1     1           1          5d2h
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ kubectl get pods
NAME           READY   STATUS    RESTARTS   AGE
consul-consul-qjst6   1/1     Running   4          5d17h
consul-consul-server-0   1/1     Running   3          5d17h
helloworld-deployment-7d88758cf4-cnfqw   0/1     Pending   0          7m24s
helloworld-deployment-7d88758cf4-w8ngp   0/1     Pending   0          7m24s
helloworld-deployment-7d88758cf4-wdj29   0/1     Pending   0          7m23s
vault-0             0/1     Running   3          5d17h
vault-1             0/1     Running   3          5d17h
vault-2             0/1     Running   3          5d17h
vault-agent-injector-ccc7d98cc-v4jcr   1/1     Running   9          5d17h
webapp-56499bb758-g7g92   1/1     Running   2          5d2h
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ kubectl describe pod helloworld-deployment-7d88758cf4-cnfqw
Name:           helloworld-deployment-7d88758cf4-cnfqw
Namespace:      default
Priority:       0
Node:           <none>
Labels:         app=helloworld
                pod-template-hash=7d88758cf4
Annotations:    <none>
Status:         Pending
IP:             <none>
IPs:            <none>
Controlled By: ReplicaSet/helloworld-deployment-7d88758cf4
Containers:
  k8s-demo:
    Image:      iddocker98/k8s-demo-nodejs
    Port:       3000/TCP
    Host Port:  0/TCP
    Environment: <none>
    Mounts:
      /var/run/secrets/kubernetes.io/serviceaccount from default-token-q966t (ro)
Conditions:
  Type      Status
  PodScheduled  False
Volumes:
  default-token-q966t:
    Type:      Secret (a volume populated by a Secret)
    SecretName: default-token-q966t
    Optional:   false
QoS Class:      BestEffort
Node-Selectors:  hardware:high-spec
Tolerations:    node.kubernetes.io/not-ready:NoExecute for 300s
                node.kubernetes.io/unreachable:NoExecute for 300s
Events:
  Type      Reason     Age   From           Message
  ----      ----     --   --            --
  Warning   FailedScheduling  <unknown>  default-scheduler  0/1 nodes are available: 1 node(s) didn't match node selector.
  Warning   FailedScheduling  <unknown>  default-scheduler  0/1 nodes are available: 1 node(s) didn't match node selector.

```

```

omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ ### il faut ajouter label pour notre node ###
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ kubectl label nodes minikube hardware=high-spec
node/minikube labeled
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ ## verifier labels ###
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ kubectl get nodes --show-labels
NAME           STATUS  ROLES   AGE    VERSION   LABELS
minikube       Ready   master  5d18h  v1.18.0   beta.kubernetes.io/arch=amd64,beta.kubernetes.io/os=linux,hardware=high-spec,kubernetes.io/arch=amd64,kubernetes.io/hostname=minikube,kubernetes.io/os=linux,minikube.k8s.io/commit=93af9c1e43cab9618e301bc9fa720c63d5efa393,minikube.k8s.io/name=minikube,minikube.k8s.io/updated_at=2020-07-01T13:38:45Z,minikube.k8s.io/version=v1.18.0,node-role.kubernetes.io/master=
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ kubectl get deployments
NAME          READY  UP-TO-DATE  AVAILABLE  AGE
helloworld-deployment  3/3     3          3          41m
vault-agent-injector  1/1     1          1          5d17h
webapp            1/1     1          1          5d3h
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ kubectl get pods
NAME          READY  STATUS  RESTARTS  AGE
consul-consul-qjst6  1/1     Running  4          5d18h
consul-consul-server-0  1/1     Running  3          5d18h
helloworld-deployment-7d88758cf4-cnfqw  1/1     Running  0          41m
helloworld-deployment-7d88758cf4-w8ngp  1/1     Running  0          41m
helloworld-deployment-7d88758cf4-wdj29  1/1     Running  0          41m
vault-0          0/1     Running  3          5d17h
vault-1          0/1     Running  3          5d17h
vault-2          0/1     Running  3          5d17h
vault-agent-injector-ccc7d98cc-v4jcr  1/1     Running  9          5d17h
webapp-56499bb758-g7g92  1/1     Running  2          5d3h
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ ##### health check #####
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ sudo nano helloworld-healthcheck.yml
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ cat helloworld-healthcheck.yml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: helloworld-deployment
spec:
  replicas: 3
  selector:
    matchLabels:
      app: helloworld
  template:
    metadata:
      labels:
        app: helloworld
    spec:
      containers:
        - name: k8s-demo
          image: iddocker98/k8s-demo-nodejs
          ports:
            - name: nodejs-port
              containerPort: 3000
      livenessProbe:
        httpGet:
          path: /
          port: nodejs-port
        initialDelaySeconds: 15
        timeoutSeconds: 30
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ kubectl create -f helloworld-healthcheck.yml
deployment.apps/helloworld-deployment created
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ kubectl get pods
NAME          READY  STATUS  RESTARTS  AGE
consul-consul-qjst6  1/1     Running  4          5d19h
consul-consul-server-0  1/1     Running  3          5d19h
helloworld-deployment-9b975d86c-dfzbl  1/1     Running  1          45m
helloworld-deployment-9b975d86c-t7fjq  1/1     Running  1          45m
helloworld-deployment-9b975d86c-zzmjp  1/1     Running  1          45m
vault-0          0/1     Running  3          5d18h
vault-1          0/1     Running  3          5d18h
vault-2          0/1     Running  3          5d18h
vault-agent-injector-ccc7d98cc-v4jcr  1/1     Running  9          5d18h
webapp-56499bb758-g7g92  1/1     Running  2          5d4h

```

```

omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/deployment$ kubectl describe pod helloworld-deployment-9b975d86c-t7fjq
Name:           helloworld-deployment-9b975d86c-t7fjq
Namespace:      default
Priority:      0
Node:          minikube/172.17.0.2
Start Time:    Fri, 17 Apr 2020 20:43:50 +0100
Labels:         app=helloworld
               pod-template-hash=9b975d86c
Annotations:   <none>
Status:        Running
IP:            172.18.0.13
IPS:
IP:            172.18.0.13
Controlled By: ReplicaSet/helloworld-deployment-9b975d86c
Containers:
  k8s-demo:
    Container ID: docker://73933193da0c1c6571ac36472251a96777643c580238093a1b59b278d7109031
    Image:          iddocker98/k8s-demo-nodejs
    Image ID:      docker-pullable://iddocker98/k8s-demo-nodejs@sha256:ab6acbefce664e5b63a210f4e7f6c8a29ad5b28595a314c8a41e8ea6738e2bd0
    Port:          3000/TCP
    Host Port:    0/TCP
    State:        Running
    Started:     Fri, 17 Apr 2020 20:58:54 +0100
    Last State:   Terminated
      Reason:      Error
      Exit Code:   137
      Started:    Fri, 17 Apr 2020 20:57:24 +0100
      Finished:   Fri, 17 Apr 2020 20:58:39 +0100
    Ready:        True
    Restart Count: 1
    Liveness:     http-get http://:nodejs-port/ delay=15s timeout=30s period=10s #success=1 #failure=3
    Environment:  <none>
    Mounts:
      /var/run/secrets/kubernetes.io/serviceaccount from default-token-q966t (ro)
Conditions:
  Type  Status
  Initialized  True
  Ready       True
  ContainersReady  True
  PodScheduled  True
Volumes:
  default-token-q966t:
    Type:  Secret (a volume populated by a Secret)
    SecretName: default-token-q966t
    Optional:  false
  QoS Class:  BestEffort
  Node-Selectors:  <none>
  Tolerations:  node.kubernetes.io/not-ready:NoExecute for 300s
                 node.kubernetes.io/unreachable:NoExecute for 300s
Events:
  Type  Reason  Age  From            Message
  ----  -----  ---  ----
  Normal  Scheduled  <unknown>  default-scheduler  Successfully assigned default/helloworld-deployment-9b975d86c-t7fjq to minikube
  Warning  Unhealthy  37m  (x3 over 37m)  kubelet, minikube  Liveness probe failed: Get http://172.18.0.13:3000/: dial tcp 172.18.0.13:3000: connect: connection refused

```

```

omar-devops@omardevops-Lenovo-Y520-15IKBN:~$ ### secrets ####
omar-devops@omardevops-Lenovo-Y520-15IKBN:~$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~$ ## generate secrets ##
omar-devops@omardevops-Lenovo-Y520-15IKBN:~$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~$ echo -b "root" | base64
LWIgcm9vdAo=
omar-devops@omardevops-Lenovo-Y520-15IKBN:~$ echo -b "password" | base64
LWIgcGFzc3dvcmQK

```

```

omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/secrets$ sudo nano secrets-db-secret.yml
[sudo] Mot de passe de omar-devops :
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/secrets$ cat secrets-db-secret.yml
apiVersion: v1
kind: Secret
metadata:
  name: db-secrets
type: Opaque
data:
  username: LWIgcm9vdAo=
  password: LWIgcGFzc3dvcmQK
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/secrets$ kubectl create -f secrets-db-secret.yml
secret/db-secrets created
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/secrets$ 

```

```

omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/secrets$ ## create secret using volume ##
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/secrets$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/secrets$ sudo nano secrets-volumes.yml
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/secrets$ cat secrets-volumes.yml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: helloworld-deployment
spec:
  replicas: 3
  selector:
    matchLabels:
      app: helloworld
  template:
    metadata:
      labels:
        app: helloworld
    spec:
      containers:
        - name: k8s-demo
          image: iddocker98/k8s-demo-nodejs
          ports:
            - name: nodejs-port
              containerPort: 3000
          volumeMounts:
            - name: cred-volume
              mountPath: /etc/creds
              readOnly: true
          volumes:
            - name: cred-volume
              secret:
                secretName: db-secrets
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/secrets$ kubectl create -f secrets-volumes.yml
deployment.apps/helloworld-deployment created
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/secrets$ 

```

```

omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/secrets$ kubectl describe pod helloworld-deployment-84b5c5f89f-dt44b
Name:           helloworld-deployment-84b5c5f89f-dt44b
Namespace:      default
Priority:       0
Node:           minikube/172.17.0.2
Start Time:     Mon, 20 Apr 2020 12:05:30 +0100
Labels:         app=helloworld
               pod-template-hash=84b5c5f89f
Annotations:   <none>
Status:         Running
IP:             172.18.0.8
IPs:
  IP:           172.18.0.8
Controlled By: ReplicaSet/helloworld-deployment-84b5c5f89f
Containers:
  k8s-demo:
    Container ID:  docker://32af70b29669606ec7c06d45a4fef18065106d15444b3cf7b9401d8d3d653ce
    Image:          iddocker98/k8s-demo-nodejs
    Image ID:      docker-pullable://iddocker98/k8s-demo-nodejs@sha256:ab6acbefce664e5b63a210f4e7f6c8a29ad5b28595a314c8a41e8ea6738e2bd0
    Port:          3000/TCP
    Host Port:    0/TCP
    State:        Running
      Started:   Mon, 20 Apr 2020 12:05:52 +0100
    Ready:        True
    Restart Count: 0
    Environment:  <none>
    Mounts:
      /etc/creds from cred-volume (ro)
      /var/run/secrets/kubernetes.io/serviceaccount from default-token-q966t (ro)
Conditions:
  Type        Status
  Initialized  True
  Ready        True
  ContainersReady  True
  PodScheduled  True
Volumes:
  cred-volume:
    Type:      Secret (a volume populated by a Secret)
    SecretName: db-secrets
    Optional:   false
  default-token-q966t:
    Type:      Secret (a volume populated by a Secret)
    SecretName: default-token-q966t
    Optional:   false
QoS Class:      BestEffort
Node-Selectors:  <none>
Tolerations:    node.kubernetes.io/not-ready:NoExecute for 300s
                node.kubernetes.io/unreachable:NoExecute for 300s
Events:
  Type  Reason  Age   From          Message
  ----  -----  ---   ----          -----
  Normal Scheduled  <unknown>  default-scheduler  Successfully assigned default/helloworld-deployment-84b5c5f89f-dt44b to minikube
  Normal Pulling   27m    kubelet, minikube  Pulling image "iddocker98/k8s-demo-nodejs"

```

```
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/first_app/k8s/secrets$ kubectl exec helloworld-deployment-84b5c5f89f-dt44b -i -t -- /bin/bash
root@helloworld-deployment-84b5c5f89f-dt44b:/app# cat /etc/creds/username
-b root
root@helloworld-deployment-84b5c5f89f-dt44b:/app# cat /etc/creds/username
-b root
root@helloworld-deployment-84b5c5f89f-dt44b:/app#
root@helloworld-deployment-84b5c5f89f-dt44b:/app# cat /etc/creds/password
-b password
root@helloworld-deployment-84b5c5f89f-dt44b:/app#
```

```
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/service-discovery$ ### server discovery ####
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/service-discovery$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/service-discovery$ ## create secret ####
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/service-discovery$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/service-discovery$ sudo nano secrets.yml
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/service-discovery$ cat secrets.yml
apiVersion: v1
kind: Secret
metadata:
  name: helloworld-secrets
type: Opaque
data:
  username: aGVsbG93b3JsZA==
  password: cGFzc3dvcmQ=
  rootPassword: cm9vdHBhc3N3b3Jk
  database: aGVsbG93b3JsZA==
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/service-discovery$ kubectl create -f secrets.yml
secret/helloworld-secrets created
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/service-discovery$ ## create pod for database ####
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/service-discovery$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/service-discovery$ sudo nano database.yml
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/service-discovery$ cat database.yml
apiVersion: v1
kind: Pod
metadata:
  name: database
  labels:
    app: database
spec:
  containers:
  - name: mysql
    image: mysql:5.7
    ports:
    - name: mysql-port
      containerPort: 3306
    env:
      - name: MYSQL_ROOT_PASSWORD
        valueFrom:
          secretKeyRef:
            name: helloworld-secrets
            key: rootPassword
      - name: MYSQL_USER
        valueFrom:
          secretKeyRef:
            name: helloworld-secrets
            key: username
      - name: MYSQL_PASSWORD
        valueFrom:
          secretKeyRef:
            name: helloworld-secrets
            key: password
      - name: MYSQL_DATABASE
        valueFrom:
          secretKeyRef:
            name: helloworld-secrets
            key: database
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/service-discovery$ kubectl create -f database.yml
pod/database created
```

```

omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/service-discovery$ ## create service for database ####
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/service-discovery$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/service-discovery$ sudo nano database-service.yml
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/service-discovery$ cat database-service.yml
apiVersion: v1
kind: Service
metadata:
  name: database-service
spec:
  ports:
  - port: 3306
    protocol: TCP
  selector:
    app: database
    type: NodePort
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/service-discovery$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/service-discovery$ ### create helloworld-db deployment ####
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/service-discovery$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/service-discovery$ sudo nano helloworld-db.yml
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/service-discovery$ cat helloworld-db.yml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: helloworld-deployment
spec:
  replicas: 3
  selector:
    matchLabels:
      app: helloworld-db
  template:
    metadata:
      labels:
        app: helloworld-db
    spec:
      containers:
      - name: k8s-demo
        image: iddocker98/k8s-demo
        command: ["node", "index-db.js"]
      ports:
      - name: nodejs-port
        containerPort: 3000
      env:
      - name: MYSQL_HOST
        value: database-service
      - name: MYSQL_USER
        value: root
      - name: MYSQL_PASSWORD
        valueFrom:
          secretKeyRef:
            name: helloworld-secrets
            key: rootPassword
      - name: MYSQL_DATABASE
        valueFrom:
          secretKeyRef:
            name: helloworld-secrets
            key: database
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/service-discovery$ kubectl create -f helloworld-db.yml
deployment.apps/helloworld-deployment created
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/service-discovery$ ## create service for deployment helloworld-db ####
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/service-discovery$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/service-discovery$ sudo nano helloworld-db-service.yml
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/service-discovery$ cat helloworld-db-service.yml
apiVersion: v1
kind: Service
metadata:
  name: helloworld-db-service
spec:
  ports:
  - port: 3000
    protocol: TCP
  selector:
    app: helloworld-db
    type: NodePort
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/service-discovery$ kubectl get pods
NAME           READY   STATUS    RESTARTS   AGE
database       1/1     Running   0          8m13s
helloworld-deployment-5cc9d97848-8z8nt  1/1     Running   0          4m17s
helloworld-deployment-5cc9d97848-9q7n5  1/1     Running   0          4m18s
helloworld-deployment-5cc9d97848-gwq2d  1/1     Running   0          4m17s
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/service-discovery$ 

```

```

omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/service-discovery$ minikube service helloworld-db-service --url
http://172.17.0.2:31901
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/service-discovery$ kubectl get pods
NAME          READY   STATUS    RESTARTS   AGE
database      1/1     Running   0          42m
helloworld-deployment-5cc9d97848-4ch6w  1/1     Running   0          41m
helloworld-deployment-5cc9d97848-4gs6q  1/1     Running   0          41m
helloworld-deployment-5cc9d97848-zjklg  1/1     Running   0          41m
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/service-discovery$ ## logs for pod ##
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/service-discovery$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/service-discovery$ kubectl logs helloworld-deployment-5cc9d97848-4ch6w
Example app listening at http://:::3000
Connection to db established
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/service-discovery$ curl http://172.17.0.2:31901
Hello World! You are visitor number 1omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/service-discovery$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/service-discovery$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/service-discovery$ curl http://172.17.0.2:31901
Hello World! You are visitor number 2omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/service-discovery$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/service-discovery$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/service-discovery$ curl http://172.17.0.2:31901
Hello World! You are visitor number 3omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/service-discovery$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/service-discovery$ curl http://172.17.0.2:31901
Hello World! You are visitor number 4omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/service-discovery$ curl http://172.17.0.2:31901
Hello World! You are visitor number 5omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/service-discovery$ 

```

```

omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/service-discovery$ kubectl exec database -i -t -- mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 6
Server version: 5.7.29 MySQL Community Server (GPL)

Copyright (c) 2000, 2020, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| helloworld         |
| mysql              |
| performance_schema |
| sys                |
+--------------------+
5 rows in set (0.02 sec)

```

```
mysql> show databases;
+-----+
| Database      |
+-----+
| information_schema |
| helloworld     |
| mysql          |
| performance_schema |
| sys            |
+-----+
5 rows in set (0.02 sec)

mysql> use helloworld;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> show tables;
+-----+
| Tables_in_helloworld |
+-----+
| visits                |
+-----+
1 row in set (0.00 sec)

mysql> select * from visits;
+---+-----+
| id | ts           |
+---+-----+
| 1  | 1587496902021 |
| 2  | 1587496949913 |
| 3  | 1587496952753 |
| 4  | 1587496957411 |
| 5  | 1587496961631 |
+---+-----+
5 rows in set (0.00 sec)
```

```
Hello World! version 2omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/configmap$ ## verify nginx server ##
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/configmap$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/configmap$ kubectl exec -i -t helloworld-nginx -c nginx -- bash
root@helloworld-nginx:/# ps x
  PID TTY      STAT      TIME COMMAND
    1 ?        Ss          0:00 nginx: master process nginx -g daemon off;
    8 pts/0    Ss          0:00 bash
   13 pts/0    R+          0:00 ps x
root@helloworld-nginx:/# cat /etc/nginx/conf.d/reverseproxy.conf
server {
    listen      80;
    server_name localhost;

    location / {
        proxy_bind 127.0.0.1;
        proxy_pass http://127.0.0.1:3000;
    }

    error_page  500 502 503 504  /50x.html;
    location = /50x.html {
        root   /usr/share/nginx/html;
    }
}
root@helloworld-nginx:/#
```

```
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/ingress$ ### create ingress file ####
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/ingress$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/ingress$ sudo nano ingress.yml
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/ingress$ cat ingress.yml
# An Ingress with 2 hosts and 3 endpoints
apiVersion: extensions/v1beta1
kind: Ingress
metadata:
  name: helloworld-rules
spec:
  rules:
    - host: helloworld-v1.example.com
      http:
        paths:
          - path: /
            backend:
              serviceName: helloworld-v1
              servicePort: 80
    - host: helloworld-v2.example.com
      http:
        paths:
          - path: /
            backend:
              serviceName: helloworld-v2
              servicePort: 80
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/ingress$ kubectl create -f ingress.yml
ingress.extensions/helloworld-rules created
```

```
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/ingress$ cat nginx-ingress-controller.yml
apiVersion: v1
kind: ReplicationController
metadata:
  name: nginx-ingress-controller
  labels:
    k8s-app: nginx-ingress-lb
spec:
  replicas: 1
  selector:
    k8s-app: nginx-ingress-lb
  template:
    metadata:
      labels:
        k8s-app: nginx-ingress-lb
        name: nginx-ingress-lb
    spec:
      terminationGracePeriodSeconds: 60
      containers:
        - image: gcr.io/google_containers/nginx-ingress-controller:0.8.3
          name: nginx-ingress-lb
          imagePullPolicy: Always
      readinessProbe:
        httpGet:
          path: /healthz
          port: 10254
          scheme: HTTP
      livenessProbe:
        httpGet:
          path: /healthz
          port: 10254
          scheme: HTTP
        initialDelaySeconds: 10
        timeoutSeconds: 1
      # use download API
      env:
        - name: POD_NAME
          valueFrom:
            fieldRef:
              fieldPath: metadata.name
        - name: POD_NAMESPACE
          valueFrom:
            fieldRef:
              fieldPath: metadata.namespace
      ports:
        - containerPort: 80
          hostPort: 80
        - containerPort: 443
          hostPort: 443
      args:
        - /nginx-ingress-controller
        - --default-backend-service=$(POD_NAMESPACE)/echoheaders-default
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/ingress$ kubectl create -f nginx-ingress-controller.yml
replicationcontroller/nginx-ingress-controller created
```

```
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/ingress$ ## create service ###
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/ingress$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/ingress$ sudo nano echoservice.yml
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/ingress$ cat echoservice.yml
apiVersion: v1
kind: ReplicationController
metadata:
  name: echoheaders
spec:
  replicas: 1
  template:
    metadata:
      labels:
        app: echoheaders
    spec:
      containers:
        - name: echoheaders
          image: gcr.io/google_containers/echoserver:1.0
          ports:
            - containerPort: 8080
---
apiVersion: v1
kind: Service
metadata:
  name: echoheaders-default
  labels:
    app: echoheaders
spec:
  type: NodePort
  ports:
    - port: 80
      nodePort: 30302
      targetPort: 8080
      protocol: TCP
      name: http
    selector:
      app: echoheaders
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/ingress$ kubectl create -f echoservice.yml
replicationcontroller/echoheaders created
service/echoheaders-default created
```

```
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/ingress$ sudo nano helloworld-v2.yml
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/ingress$ cat helloworld-v2.yml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: helloworld-v2-deployment
spec:
  replicas: 1
  selector:
    matchLabels:
      app: helloworld-v2
  template:
    metadata:
      labels:
        app: helloworld-v2
    spec:
      containers:
        - name: k8s-demo
          image: iddocker98/k8s-demo-nodejs:2
          ports:
            - name: nodejs-port
              containerPort: 3000
    ...
apiVersion: v1
kind: Service
metadata:
  name: helloworld-v2
spec:
  type: NodePort
  ports:
    - port: 80
      nodePort: 30304
      targetPort: 3000
      protocol: TCP
      name: http
  selector:
    app: helloworld-v2
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/ingress$ kubectl create -f helloworld-v2.yml
deployment.apps/helloworld-v2-deployment created
service/helloworld-v2 created
```

```

omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/ingress$ minikube ip
172.17.0.2
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/ingress$ curl 172.17.0.2
CLIENT VALUES:
client_address=('172.18.0.1', 55302) (172.18.0.1)
command=GET
path=/
real_path=/
query=
request_version=HTTP/1.1

SERVER VALUES:
server_version=BaseHTTP/0.6
sys_version=Python/3.5.0
protocol_version=HTTP/1.0

HEADERS RECEIVED:
Accept:/*
Connection:close
Host:172.17.0.2
User-Agent:curl/7.58.0
X-Forwarded-For:172.18.0.1
X-Forwarded-Host:172.17.0.2
X-Forwarded-Port:80
X-Forwarded-Proto:http
X-Original-URI:/
X-Real-IP:172.18.0.1
X-Request-ID:1bd58470c13144c2f5438b6f1a766a5c
X-Scheme:http
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/ingress$ curl 172.17.0.2 -H 'Host: helloworld-v1.example.com'
Hello World!!!!omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/ingress$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/ingress$ curl 172.17.0.2 -H 'Host: helloworld-v2.example.com'
Hello World! version 2omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/ingress$ kubectl get service
NAME          TYPE      CLUSTER-IP   EXTERNAL-IP  PORT(S)        AGE
echoheaders-default  NodePort  10.105.159.112 <none>       80:30302/TCP  16h
helloworld-v1    NodePort  10.96.170.1    <none>       80:30303/TCP  16h
helloworld-v2    NodePort  10.109.63.37   <none>       80:30304/TCP  16h
kubernetes      ClusterIP  10.96.0.1    <none>       443/TCP       16h
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/ingress$ 
```

```

omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/volumes$ ## Next create PersistentVolume ##
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/volumes$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/volumes$ sudo nano pv-volume.yml
[sudo] Mot de passe de omar-devops :
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/volumes$ cat pv-volume.yml
apiVersion: v1
kind: PersistentVolume
metadata:
  name: pv-volume
  labels:
    type: local
spec:
  storageClassName: manual
  capacity:
    storage: 10Gi
  accessModes:
    - ReadWriteOnce
  hostPath:
    path: "/mnt/data"
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/volumes$ kubectl create -f pv-volume.yml
persistentvolume/pv-volume created
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/volumes$ kubectl get pv pv-volume
NAME      CAPACITY   ACCESS MODES  RECLAIM POLICY  STATUS   CLAIM   STORAGECLASS  REASON   AGE
pv-volume  10Gi       RWO        Retain        Available   manual   23s
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/volumes$ 
```

```

omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/volumes$ ### create PersistentVolumeClaim ####
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/volumes$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/volumes$ sudo nano pvc-volume.yml
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/volumes$ cat pvc-volume.yml
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: pv-claim
spec:
  storageClassName: manual
  accessModes:
    - ReadWriteOnce
  resources:
    requests:
      storage: 3Gi
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/volumes$ kubectl create -f pvc-volume.yml
persistentvolumeclaim/pv-claim created
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/volumes$ kubectl get pv pv-volume
NAME      CAPACITY   ACCESS MODES   RECLAIM POLICY   STATUS   CLAIM           STORAGECLASS   REASON   AGE
pv-volume  10Gi       RWO          Retain          Bound    default/pv-claim   manual        11m
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/volumes$ kubectl get pvc pv-claim
NAME      STATUS   VOLUME   CAPACITY   ACCESS MODES   STORAGECLASS   AGE
pv-claim  Bound   pv-volume  10Gi       RWO          manual        7m54s
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/volumes$ ## Output shows that PersistentVolumeClaim is bound to PersistentVolume ####
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/volumes$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/volumes$ ## create deployment with volume ####
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/volumes$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/volumes$ sudo nano helloworld-with-volume.yml
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/volumes$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/volumes$ cat helloworld-with-volume.yml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: helloworld-deployment
spec:
  replicas: 1
  selector:
    matchLabels:
      app: helloworld
  template:
    metadata:
      labels:
        app: helloworld
    spec:
      containers:
        - name: k8s-demo
          image: iddocker98/k8s-demo-nodejs
          ports:
            - name: nodejs-port
              containerPort: 3000
          volumeMounts:
            - mountPath: /myvol
              name: myvolume
      volumes:
        - name: myvolume
          persistentVolumeClaim:
            claimName: pv-claim
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/volumes$ kubectl create -f helloworld-with-volume.yml
deployment.apps/helloworld-deployment created

```

```

omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/volumes$ kubectl get pods
NAME                               READY   STATUS    RESTARTS   AGE
helloworld-deployment-787b798f57-t8q9t   1/1    Running   0          26s
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/volumes$ kubectl describe pod helloworld-deployment-787b798f57-t8q9t
Name:           helloworld-deployment-787b798f57-t8q9t
Namespace:      default
Priority:       0
Node:           minikube/172.17.0.2
Start Time:    Sat, 25 Apr 2020 03:12:08 +0100
Labels:         app=helloworld
                pod-template-hash=787b798f57
Annotations:   <none>
Status:        Running
IP:            172.18.0.2
IPs:
  IP:          172.18.0.2
Controlled By: ReplicaSet/helloworld-deployment-787b798f57
Containers:
  k8s-demo:
    Container ID:  docker://f1b7189ee3c144e6cbc6e202c22dca31b586ec5ed3729ae02fe9f2d0cf6534e
    Image:         iddocker98/k8s-demo-nodejs
    Image ID:     docker-pullable://iddocker98/k8s-demo-nodejs@sha256:0255c561145f5d371b5200b42a17dbdacece6bfdeedd35a6a70b8a66d0baeda4
    Port:          3000/TCP
    Host Port:    0/TCP
    State:        Running
      Started:   Sat, 25 Apr 2020 03:12:15 +0100
    Ready:        True
    Restart Count: 0
    Environment:  <none>
    Mounts:
      /myvol from myvolume (rw)
      /var/run/secrets/kubernetes.io/serviceaccount from default-token-8gj4g (ro)
  Conditions:
    Type      Status
    Initialized  True
    Ready      True
    ContainersReady  True
    PodsScheduled  True
Volumes:
  myvolume:
    Type:      PersistentVolumeClaim (a reference to a PersistentVolumeClaim in the same namespace)
    ClaimName:  pv-claim
    ReadOnly:   false
  default-token-8gj4g:
    Type:      Secret (a volume populated by a Secret)
    SecretName: default-token-8gj4g
    Optional:   false
  QoS Class:  BestEffort
  Node-Selectors:  <none>
  Tolerations:   node.kubernetes.io/not-ready:NoExecute for 300s
                 node.kubernetes.io/unreachable:NoExecute for 300s
Events:
  Type  Reason  Age   From           Message
  ----  ----   --   --   --
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/volumes$ kubectl exec helloworld-deployment-787b798f57-glg5d -i -t -- bash
root@helloworld-deployment-787b798f57-glg5d:/app# ls -ahl /myvol/
total 8.0K
drwxr-xr-x 2 root root 4.0K Apr 25 02:37 .
drwxr-xr-x 1 root root 4.0K Apr 25 02:37 ..
root@helloworld-deployment-787b798f57-glg5d:/app# echo 'test' > /myvol/myvol.txt
root@helloworld-deployment-787b798f57-glg5d:/app# echo 'test2' > /test.txt
root@helloworld-deployment-787b798f57-glg5d:/app# ls -ahl /myvol/
total 12K
drwxr-xr-x 2 root root 4.0K Apr 25 02:49 .
drwxr-xr-x 1 root root 4.0K Apr 25 02:49 ..
-rw-r--r-- 1 root root  5 Apr 25 02:49 myvol.txt
root@helloworld-deployment-787b798f57-glg5d:/app# ls -ahl /test.txt
-rw-r--r-- 1 root root  6 Apr 25 02:49 /test.txt
root@helloworld-deployment-787b798f57-glg5d:/app# exit
exit
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/volumes$ kubectl drain minikube --force --ignore-daemonsets
node/minikube already cordoned
WARNING: ignoring DaemonSet-managed Pods: kube-system/kindnet-7cfq5, kube-system/kube-proxy-745dg; deleting Pods not managed by ReplicationController, m/storage-provisioner
evicting pod kube-system/coredns-66bff467f8-vc957
evicting pod kube-system/storage-provisioner
evicting pod default/helloworld-deployment-787b798f57-glg5d
evicting pod kube-system/coredns-66bff467f8-sc78d
pod/storage-provisioner evicted
pod/coredns-66bff467f8-vc957 evicted
pod/coredns-66bff467f8-sc78d evicted
pod/helloworld-deployment-787b798f57-glg5d evicted
node/minikube evicted
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/volumes$ kubectl get pods
NAME                               READY   STATUS    RESTARTS   AGE
helloworld-deployment-787b798f57-8wj2   0/1    Pending   0          64s
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/volumes$ kubectl get pods
NAME                               READY   STATUS    RESTARTS   AGE
helloworld-deployment-787b798f57-8wj2   0/1    Pending   0          15m
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/volumes$ kubectl get pods
NAME                               READY   STATUS    RESTARTS   AGE
helloworld-deployment-787b798f57-8wj2   1/1    Running   0          19m

```

```

omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/volumes$ kubectl describe pod helloworld-deployment-787b798f57-8wjd2
Name:           helloworld-deployment-787b798f57-8wjd2
Namespace:      default
Priority:       0
Node:          minikube/172.17.0.2
Start Time:    Sat, 25 Apr 2020 04:27:47 +0100
Labels:         app=helloworld
                pod-template-hash=787b798f57
Annotations:   <none>
Status:        Running
IP:            172.18.0.4
IPs:
IP:            172.18.0.4
Controlled By: ReplicaSet/helloworld-deployment-787b798f57
Containers:
  k8s-demo:
    Container ID:  docker://15f2ee0c984dd79911fa2887ef700f1d943c23b9147f2df402b7bd26f951f173
    Image:          iddock98/k8s-demo-nodejs
    Image ID:      docker-pullable://iddock98/k8s-demo-nodejs@sha256:0255c561145f5d371b5200b42a17dbdace6bfdeedd35a6a70b8a66d0baeda4
    Port:          3000/TCP
    Host Port:    0/TCP
    State:        Running
      Started:   Sat, 25 Apr 2020 04:28:12 +0100
    Ready:        True
    Restart Count: 0
    Environment:  <none>
    Mounts:
      /myvol from myvolume (rw)
      /var/run/secrets/kubernetes.io/serviceaccount from default-token-8gj4g (ro)
Conditions:
  Type      Status
  Initialized  True
  Ready      True
  ContainersReady  True
  PodsScheduled  True
Volumes:
  myvolume:
    Type:     PersistentVolumeClaim (a reference to a PersistentVolumeClaim in the same namespace)
    ClaimName:  pv-claim
    ReadOnly:   false
    default-token-8gj4g:
      Type:     Secret (a volume populated by a Secret)
      SecretName: default-token-8gj4g
      Optional:  false
    QoS Class:  BestEffort
    Node-Selectors:  <none>
    Tolerations:  node.kubernetes.io/not-ready:NoExecute for 300s
                  node.kubernetes.io/unreachable:NoExecute for 300s
Events:
  Type  Reason     Age   From           Message
  ----  ----      --   --   -----
  Warning FailedScheduling  3m56s (x16 over 19m)  default-scheduler  0/1 nodes are available: 1 node(s) were unschedulable.
  Normal Scheduled  2m25s   default-scheduler  Successfully assigned default/helloworld-deployment-787b798f57-8wjd2 to minikube
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/volumes$ kubectl exec helloworld-deployment-787b798f57-8wjd2 -i -t -- bash
root@helloworld-deployment-787b798f57-8wjd2:/app# ls -ahl /myvol/
total 12K
drwxr-xr-x 2 root root 4.0K Apr 25 02:49 .
drwxr-xr-x 1 root root 4.0K Apr 25 03:28 ..
-rw-r--r-- 1 root root  5 Apr 25 02:49 myvol.txt
root@helloworld-deployment-787b798f57-8wjd2:/app# ls -ahl /test.txt
ls: cannot access '/test.txt': No such file or directory
root@helloworld-deployment-787b798f57-8wjd2:/app# 
```

```
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/autoscaling$ ### Horizontal Pod Scaling ###
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/autoscaling$ sudo nano hpa-deployment.yml
[sudo] Mot de passe de omar-devops :
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/autoscaling$ cat hpa-deployment.yml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: hpa-example
spec:
  replicas: 3
  selector:
    matchLabels:
      app: hpa-example
  template:
    metadata:
      labels:
        app: hpa-example
    spec:
      containers:
        - name: hpa-example
          image: gcr.io/google_containers/hpa-example
          ports:
            - name: http-port
              containerPort: 80
          resources:
            requests:
              cpu: 200m
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/autoscaling$ sudo nano hpa-service.yml
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/autoscaling$ cat hpa-service.yml
apiVersion: v1
kind: Service
metadata:
  name: hpa-example
spec:
  ports:
    - port: 31001
      nodePort: 31001
      targetPort: http-port
      protocol: TCP
  selector:
    app: hpa-example
  type: NodePort
```



```

omar-devops@omardevops-Lenovo-Y520-15IKBN:~$ kubectl get hpa
NAME           REFERENCE          TARGETS      MINPODS   MAXPODS   REPLICAS   AGE
hpa-example-autoscaler Deployment/hpa-example 1%/50%     1          10         1          6m10s

omar-devops@omardevops-Lenovo-Y520-15IKBN:~$ kubectl get hpa
NAME           REFERENCE          TARGETS      MINPODS   MAXPODS   REPLICAS   AGE
hpa-example-autoscaler Deployment/hpa-example 366%/50%    1          10         1          6m32s

omar-devops@omardevops-Lenovo-Y520-15IKBN:~$ kubectl get hpa
NAME           REFERENCE          TARGETS      MINPODS   MAXPODS   REPLICAS   AGE
hpa-example-autoscaler Deployment/hpa-example 366%/50%    1          10         4          6m43s

omar-devops@omardevops-Lenovo-Y520-15IKBN:~$ kubectl get pods
NAME             READY   STATUS        RESTARTS   AGE
hpa-example-86d788887c-c8sck  0/1    Pending       0          8s
hpa-example-86d788887c-fs22m  0/1    Pending       0          9s
hpa-example-86d788887c-k5pjf  0/1    ContainerCreating 0          26s
hpa-example-86d788887c-khfct  0/1    ContainerCreating 0          27s
hpa-example-86d788887c-mpz94  1/1    Running       0          6m47s
hpa-example-86d788887c-smjn6  0/1    ContainerCreating 0          9s
hpa-example-86d788887c-xwhll  0/1    ContainerCreating 0          26s
hpa-example-86d788887c-z8b2t  0/1    ContainerCreating 0          10s
load-generator      1/1    Running       0          3m17s

omar-devops@omardevops-Lenovo-Y520-15IKBN:~$ kubectl get hpa
NAME           REFERENCE          TARGETS      MINPODS   MAXPODS   REPLICAS   AGE
hpa-example-autoscaler Deployment/hpa-example 366%/50%    1          10         8          6m57s

omar-devops@omardevops-Lenovo-Y520-15IKBN:~$ kubectl get pods
NAME             READY   STATUS        RESTARTS   AGE
hpa-example-86d788887c-c8sck  0/1    ContainerCreating 0          25s
hpa-example-86d788887c-fs22m  0/1    ContainerCreating 0          26s
hpa-example-86d788887c-k5pjf  0/1    ContainerCreating 0          43s
hpa-example-86d788887c-khfct  0/1    ContainerCreating 0          44s
hpa-example-86d788887c-mpz94  1/1    Running       0          7m4s
hpa-example-86d788887c-smjn6  0/1    ContainerCreating 0          26s
hpa-example-86d788887c-xwhll  0/1    ContainerCreating 0          43s
hpa-example-86d788887c-z8b2t  0/1    ContainerCreating 0          27s
load-generator      1/1    Running       0          3m34s

omar-devops@omardevops-Lenovo-Y520-15IKBN:~$ kubectl get hpa
NAME           REFERENCE          TARGETS      MINPODS   MAXPODS   REPLICAS   AGE
hpa-example-autoscaler Deployment/hpa-example 287%/50%    1          10         10         8m36s

omar-devops@omardevops-Lenovo-Y520-15IKBN:~$ kubectl get pods
NAME             READY   STATUS        RESTARTS   AGE
hpa-example-86d788887c-c8sck  1/1    Running       0          2m1s
hpa-example-86d788887c-fs22m  1/1    Running       0          2m2s
hpa-example-86d788887c-h87kz  1/1    Running       0          72s
hpa-example-86d788887c-hdpmnt 1/1    Running       0          69s
hpa-example-86d788887c-k5pjf  1/1    Running       0          2m19s
hpa-example-86d788887c-khfct  1/1    Running       0          2m20s
hpa-example-86d788887c-mpz94  1/1    Running       0          8m40s
hpa-example-86d788887c-smjn6  1/1    Running       0          2m2s
hpa-example-86d788887c-xwhll  1/1    Running       0          2m19s
hpa-example-86d788887c-z8b2t  1/1    Running       0          2m3s
load-generator      1/1    Running       0          5m10s

omar-devops@omardevops-Lenovo-Y520-15IKBN:~$ kubectl get hpa
NAME           REFERENCE          TARGETS      MINPODS   MAXPODS   REPLICAS   AGE
hpa-example-autoscaler Deployment/hpa-example 0%/50%     1          10         1          15m

omar-devops@omardevops-Lenovo-Y520-15IKBN:~$ kubectl get pods
NAME             READY   STATUS        RESTARTS   AGE
hpa-example-86d788887c-c8sck  1/1    Terminating   0          9m1s
hpa-example-86d788887c-fs22m  1/1    Terminating   0          9m2s
hpa-example-86d788887c-h87kz  1/1    Terminating   0          8m12s
hpa-example-86d788887c-hdpmnt 1/1    Terminating   0          8m9s
hpa-example-86d788887c-k5pjf  1/1    Terminating   0          9m19s
hpa-example-86d788887c-khfct  1/1    Terminating   0          9m20s
hpa-example-86d788887c-mpz94  1/1    Running       0          15m
hpa-example-86d788887c-smjn6  1/1    Terminating   0          9m2s
hpa-example-86d788887c-xwhll  1/1    Terminating   0          9m19s
hpa-example-86d788887c-z8b2t  1/1    Terminating   0          9m3s
load-generator      1/1    Running       0          12m

```

```

omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/resourcequotas$ ### resource quotas with Namespace ####
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/resourcequotas$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/resourcequotas$ ## create resourcequotas ##
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/resourcequotas$ sudo nano resourcequota.yml
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/resourcequotas$ cat resourcequota.yml
apiVersion: v1
kind: Namespace
metadata:
  name: myspace
---
apiVersion: v1
kind: ResourceQuota
metadata:
  name: compute-quota
  namespace: myspace
spec:
  hard:
    requests.cpu: "1"
    requests.memory: 1Gi
    limits.cpu: "2"
    limits.memory: 2Gi
---
apiVersion: v1
kind: ResourceQuota
metadata:
  name: object-quota
  namespace: myspace
spec:
  hard:
    configmaps: "10"
    persistentvolumeclaims: "4"
    replicationcontrollers: "20"
    secrets: "10"
    services: "10"
    services.loadbalancers: "2"
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/resourcequotas$ kubectl create -f resourcequota.yml
namespace/myspace created
resourcequota/compute-quota created
resourcequota/object-quota created
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/resourcequotas$ ## create deployment without quotas ##
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/resourcequotas$ sudo nano helloworld-no-quotas.yml
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/resourcequotas$ cat helloworld-no-quotas.yml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: helloworld-deployment
  namespace: myspace
spec:
  replicas: 3
  selector:
    matchLabels:
      app: helloworld
  template:
    metadata:
      labels:
        app: helloworld
    spec:
      containers:
        - name: k8s-demo
          image: iddocker98/k8s-demo-nodejs
          ports:
            - name: nodejs-port
              containerPort: 3000
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/resourcequotas$ kubectl create -f helloworld-no-quotas.yml
deployment.apps/helloworld-deployment created
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/resourcequotas$ kubectl get deploy --namespace=myspace
NAME           READY   UP-TO-DATE   AVAILABLE   AGE
helloworld-deployment   0/3     0          0          8m57s

```

```

omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/resourcequotas$ kubectl get rs --namespace=myspace
NAME           DESIRED  CURRENT  READY   AGE
helloworld-deployment-56bfbf7695  3        0        0       10m
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/resourcequotas$ kubectl describe rs helloworld-deployment-56bfbf7695 --namespace=myspace
Name:          helloworld-deployment-56bfbf7695
Namespace:     myspace
Selector:     app=helloworld,pod-template-hash=56bfbf7695
Labels:        app=helloworld
Annotations:   deployment.kubernetes.io/desired-replicas: 3
               deployment.kubernetes.io/max-replicas: 4
               deployment.kubernetes.io/revision: 1
Controlled By: Deployment/helloworld-deployment
Replicas:      0 current / 3 desired
Pods Status:   0 Running / 0 Waiting / 0 Succeeded / 0 Failed
Pod Template:
  Labels:  app=helloworld
           pod-template-hash=56bfbf7695
  Containers:
    k8s-demo:
      Image:      iddocker98/k8s-demo-nodejs
      Port:       3000/TCP
      Host Port:  0/TCP
      Environment: <none>
      Mounts:    <none>
      Volumes:   <none>
  Conditions:
    Type     Status  Reason
    ----   -----  -----
    ReplicaFailure  True   FailedCreate
Events:
  Type     Reason     Age   From           Message
  ----   -----     --   --            --
  Warning  FailedCreate  11m  replicaset-controller  Error creating: pods "helloworld-deployment-56bfbf7695-b2zsl" is forbidden: failed quota: compute-quota: must specify limits.cpu,limits.memory,requests.cpu,requests.memory
  Warning  FailedCreate  11m  replicaset-controller  Error creating: pods "helloworld-deployment-56bfbf7695-vqmz2" is forbidden: failed quota: compute-quota: must specify limits.cpu,limits.memory,requests.cpu,requests.memory
  Warning  FailedCreate  11m  replicaset-controller  Error creating: pods "helloworld-deployment-56bfbf7695-pcx2n" is forbidden: failed quota: compute-quota: must specify limits.cpu,limits.memory,requests.cpu,requests.memory
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/resourcequotas$ kubectl delete deploy helloworld-deployment --namespace=myspace
deployment.apps "helloworld-deployment" deleted
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/resourcequotas$ ## create deployment with quotas ##
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/resourcequotas$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/resourcequotas$ sudo nano helloworld-with-quotas.yml
[sudo] Mot de passe de omar-devops :
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/resourcequotas$ cat helloworld-with-quotas.yml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: helloworld-deployment
  namespace: myspace
spec:
  replicas: 3
  selector:
    matchLabels:
      app: helloworld
  template:
    metadata:
      labels:
        app: helloworld
    spec:
      containers:
        - name: k8s-demo
          image: iddocker98/k8s-demo-nodejs
          ports:
            - name: nodejs-port
              containerPort: 3000
          resources:
            requests:
              cpu: 200m
              memory: 0.5Gi
            limits:
              cpu: 400m
              memory: 1Gi
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/resourcequotas$ kubectl create -f helloworld-with-quotas.yml
deployment.apps/helloworld-deployment created
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/resourcequotas$ kubectl get pods --namespace=myspace
NAME           READY  STATUS    RESTARTS   AGE
helloworld-deployment-69f68c9f96-2pkgx  1/1   Running   0          3h10m
helloworld-deployment-69f68c9f96-p85bc  1/1   Running   0          3h10m

```

```

omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/resourcequotas$ kubectl get rs --namespace=myspace
NAME           DESIRED   CURRENT  READY   AGE
helloworld-deployment-69f68c9f96  3         2        2      3h58m
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/resourcequotas$ kubectl describe rs helloworld-deployment-69f68c9f96 --namespace=myspace
Name:          helloworld-deployment-69f68c9f96
Namespace:     myspace
Selector:     app=helloworld,pod-template-hash=69f68c9f96
Labels:        app=helloworld
               pod-template-hash=69f68c9f96
Annotations:   deployment.kubernetes.io/desired-replicas: 3
               deployment.kubernetes.io/max-replicas: 4
               deployment.kubernetes.io/revision: 1
Controlled By: Deployment/helloworld-deployment
Replicas:      2 current / 3 desired
Pods Status:   2 Running / 0 Waiting / 0 Succeeded / 0 Failed
Pod Template:
  Labels:  app=helloworld
           pod-template-hash=69f68c9f96
  Containers:
    k8s-demo:
      Image:  iddocker98/k8s-demo-nodejs
      Port:   3000/TCP
      Host Port:  0/TCP
      Limits:
        cpu:  400m
        memory:  1Gi
      Requests:
        cpu:   200m
        memory:  512Mi
      Environment: <none>
      Mounts:  <none>
      Volumes: <none>
  Conditions:
    Type     Status  Reason
    ----   -----
    ReplicaFailure  True    FailedCreate
Events:
  Type     Reason     Age   From           Message
  ----   -----   ----  ----
  Warning  FailedCreate  50m  replicaset-controller  Error creating: pods "helloworld-deployment-69f68c9f96-ht2bf" is forbidden: exceeded quota: compute-quota
  its.memory=1Gi,requests.memory=512Mi, used: limits.memory=2Gi,requests.memory=1Gi, limited: limits.memory=2Gi,requests.memory=1Gi
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/resourcequotas$ kubectl get quota --namespace=myspace
NAME          AGE   REQUEST
compute-quota 4h21m requests.cpu: 400m/1, requests.memory: 1Gi/1Gi
2, limits.memory: 2Gi/2Gi
object-quota  4h21m configmaps: 0/10, persistentvolumeclaims: 0/4, replicationcontrollers: 0/20, secrets: 1/10, services: 0/10, services.loadbalancers: 0/2
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/resourcequotas$ kubectl describe quota compute-quota --namespace=myspace
Name:          compute-quota
Namespace:     myspace
Resource       Used Hard
-----
limits.cpu     800m  2
limits.memory  2Gi   2Gi
requests.cpu   400m  1
requests.memory 1Gi   1Gi
-----
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/resourcequotas$ kubectl delete deployment helloworld-deployment --namespace=myspace
deployment.apps "helloworld-deployment" deleted
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/resourcequotas$ ## resolve this problem with limits ####
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/resourcequotas$ 
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/resourcequotas$ sudo nano defaults.yml
[sudo] Mot de passe de omar-devops :
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/resourcequotas$ cat defaults.yml
apiVersion: v1
kind: LimitRange
metadata:
  name: limits
  namespace: myspace
spec:
  limits:
  - default:
    - cpu: 200m
      memory: 512Mi
    defaultRequest:
      - cpu: 100m
        memory: 256Mi
      type: Container
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/resourcequotas$ kubectl create -f defaults.yml
limitrange/limits created
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/resourcequotas$ kubectl describe limits limits --namespace=myspace
Name:          limits
Namespace:     myspace
Type          Resource Min  Max  Default Request  Default Limit  Max Limit/Request Ratio
----          -----  ---  ---  -----  -----  -----  -----
Container     memory   -    -    256Mi      512Mi      -
Container     cpu     -    -    100m      200m      -
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/resourcequotas$ ## test deployment ##
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/resourcequotas$ kubectl create -f helloworld-no-quotas.yml
deployment.apps/helloworld-deployment created
omar-devops@omardevops-Lenovo-Y520-15IKBN:~/Bureau/kubernets/resourcequotas$ kubectl get pods --namespace=myspace
NAME           READY   STATUS    RESTARTS   AGE
helloworld-deployment-56bfbf7695-pr9df  1/1    Running   0          92s
helloworld-deployment-56bfbf7695-qj5pp  1/1    Running   0          94s
helloworld-deployment-56bfbf7695-t4lkg  1/1    Running   0          92s

```

