

Day 3

Info - StatefulSet

<https://medium.com/tektutor/deploying-stateful-applications-in-kubernetes-8ffd46920b55>

Lab - Listing all projects in OpenShift cluster

```
oc get projects  
oc get namespaces
```

Expected output



The screenshot shows a terminal window with five tabs open. The active tab is titled '(jegan@tektutor.org)-[~/openshift-sep-2023]'. The command 'oc get projects' is run, and the output lists various OpenShift projects. The output is as follows:

NAME	DISPLAY NAME	STATUS
aap		Active
default		Active
jegan		Active
kube-node-lease		Active
kube-public		Active
kube-system		Active
metallb-system		Active
openshift		Active
openshift-apiserver		Active
openshift-apiserver-operator		Active
openshift-authentication		Active
openshift-authentication-operator		Active
openshift-cloud-controller-manager		Active
openshift-cloud-controller-manager-operator		Active
openshift-cloud-credential-operator		Active
openshift-cloud-network-config-controller		Active
openshift-cluster-csi-drivers		Active
openshift-cluster-machine-approver		Active
openshift-cluster-node-tuning-operator		Active
openshift-cluster-samples-operator		Active
openshift-cluster-storage-operator		Active
openshift-cluster-version		Active
openshift-config		Active
openshift-config-managed		Active
openshift-config-operator		Active

```
(jegan@tektutor.org)-[~/openshift-sep-2023]
$ oc get namespaces
NAME          STATUS  AGE
aap           Active  4d22h
default       Active  6d5h
jegan         Active  34m
kube-node-lease  Active  6d5h
kube-public    Active  6d5h
kube-system    Active  6d5h
metallb-system Active  21h
openshift      Active  6d5h
openshift-apiserver  Active  6d5h
openshift-apiserver-operator  Active  6d5h
openshift-authentication  Active  6d5h
openshift-authentication-operator  Active  6d5h
openshift-cloud-controller-manager  Active  6d5h
openshift-cloud-controller-manager-operator  Active  6d5h
openshift-cloud-credential-operator  Active  6d5h
openshift-cloud-network-config-controller  Active  6d5h
openshift-cluster-csi-drivers  Active  6d5h
openshift-cluster-machine-approver  Active  6d5h
openshift-cluster-node-tuning-operator  Active  6d5h
openshift-cluster-samples-operator  Active  6d5h
openshift-cluster-storage-operator  Active  6d5h
openshift-cluster-version  Active  6d5h
openshift-config  Active  6d5h
openshift-config-managed  Active  6d5h
openshift-config-operator  Active  6d5h
openshift-console  Active  6d5h
```

Lab - Understanding Service Discovery in OpenShift/Kubernetes

- Service discovery is nothing but accessing a service by its name which gets resolved into its respective IP address
- DNS Server helps resolving the name of the service to the IP address of the service
- Service discovery works only within the OpenShift cluster and it is supported by every type of service

1. ClusterIP
2. NodePort and
3. LoadBalancer

Delete any existing service you have in your project

```
oc project
oc delete svc/nginx
oc expose deploy/nginx --type=ClusterIP --port=8080
oc get svc/nginx
oc describe svc/nginx
```

Now we need test pod with curl utility, hence let's create a pod as shown below

```
oc run hello --image=tektutor/spring-tektutor-hellos:latest
oc get po -w
oc exec -it hello sh
curl http://<cluster-ip-of-nginx-service>:8080
curl http://172.30.222.144:8080
```

Accessing a service by its name (Service Discovery), in the same hello pod shell you can try this

```
curl http://nginx:8080
```

How the Pod is able to resolve the nginx service name to its corresponding IP address is

```
cat /etc/resolv.conf
```

The above file uses the DNS nameserver 172.30.0.10, this entry is created in each Pod by kubelet at the time it creates the Pod on the respective node where kubelet is running.

If you wish to see the DNS Nameserver, you can try this

```
oc get svc --all-namespaces | grep -i dns
oc describe svc/dns-default -n openshift-dns
```

Lab - Getting help info about any OpenShift resource

```
oc explain deployment
```

Expected output

```
(jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$ oc explain deployment
KIND: Deployment
VERSION: apps/v1

DESCRIPTION:
Deployment enables declarative updates for Pods and ReplicaSets.

FIELDS:
apiVersion <string>
APIVersion defines the versioned schema of this representation of an
object. Servers should convert recognized schemas to the latest internal
value, and may reject unrecognized values. More info:
https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources

kind <string>
Kind is a string value representing the REST resource this object
represents. Servers may infer this from the endpoint the client submits
requests to. Cannot be updated. In CamelCase. More info:
https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds

metadata <Object>
Standard object's metadata. More info:
https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata

spec <Object>
Specification of the desired behavior of the Deployment.

status <Object>
Most recently observed status of the Deployment.

(jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
```

Lab - Declaratively creating nginx deployment

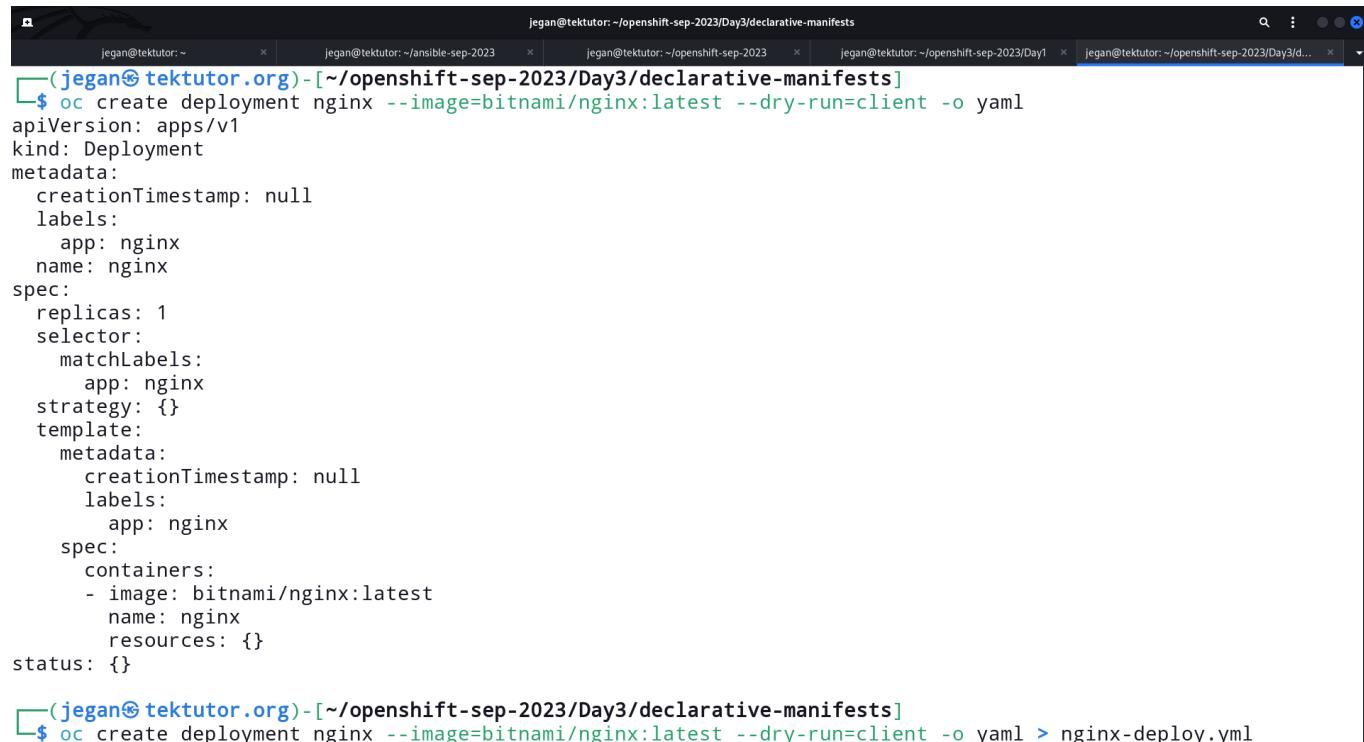
Let's delete the project and recreate project with the same name

```
oc delete project/jegan  
oc new-project jegan
```

Let's generate the nginx deployment yaml file as shown below

```
oc create deployment nginx --image=bitnami/nginx:latest --dry-run=client -o yaml  
oc create deployment nginx --image=bitnami/nginx:latest --dry-run=client -o yaml > nginx-deploy.yml  
cat nginx-deploy.yml
```

Expected output



The screenshot shows a terminal window with four tabs open. The active tab displays the command:

```
jegan@tektutor:~/openshift-sep-2023/Day3/declarative-manifests  
$ oc create deployment nginx --image=bitnami/nginx:latest --dry-run=client -o yaml
```

The output of the command is a YAML manifest for a Deployment named "nginx". The manifest includes fields for apiVersion, kind, metadata, spec (with replicas set to 1), and status. It also includes a template section with a spec containing a single container using the bitnami/nginx:latest image.

```
apiVersion: apps/v1  
kind: Deployment  
metadata:  
  creationTimestamp: null  
labels:  
  app: nginx  
name: nginx  
spec:  
  replicas: 1  
  selector:  
    matchLabels:  
      app: nginx  
  strategy: {}  
  template:  
    metadata:  
      creationTimestamp: null  
    labels:  
      app: nginx  
    spec:  
      containers:  
      - image: bitnami/nginx:latest  
        name: nginx  
        resources: {}  
status: {}
```

At the bottom of the terminal, the command is repeated:

```
jegan@tektutor:~/openshift-sep-2023/declarative-manifests  
$ oc create deployment nginx --image=bitnami/nginx:latest --dry-run=client -o yaml > nginx-deploy.yml
```

The screenshot shows a terminal window with multiple tabs. The active tab displays the contents of a YAML file named `nginx-deploy.yml`. The file defines a Deployment object with the following specifications:

```
apiVersion: apps/v1
kind: Deployment
metadata:
  creationTimestamp: null
  labels:
    app: nginx
    name: nginx
spec:
  replicas: 1
  selector:
    matchLabels:
      app: nginx
  strategy: {}
  template:
    metadata:
      creationTimestamp: null
      labels:
        app: nginx
    spec:
      containers:
        - image: bitnami/nginx:latest
          name: nginx
          resources: {}
status: {}
```

Lab - Creating a nginx deployment in declarative style using yaml(manifest) file

The `oc create` command should be used when the nginx deployment you are creating through file doesn't already exist in the OpenShift cluster. In otherwords, only the first time we should use create command and subsequent times we should use apply command to update any changes on the existing resources using the same yaml file.

Interestingly, the apply command can be used everytime unlike the create command.

```
cd ~/openshift-sep-2023
git pull
cd Day3/declarative-manifests
oc create -f nginx-deploy.yml --save-config
oc create -f nginx-deploy.yml --save-config
oc apply -f nginx-deploy.yml
oc get deploy,rs,po
```

Expected output

```
jegan@tektutor:~/openshift-sep-2023/Day3/declarative-manifests
[jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$ oc create -f nginx-deploy.yml --save-config
deployment.apps/nginx created

[jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$ oc create -f nginx-deploy.yml --save-config
Error from server (AlreadyExists): error when creating "nginx-deploy.yml": deployments.apps "nginx" already exists

[jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$ vim nginx-deploy.yml

[jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$ oc apply -f nginx-deploy.yml --save-config
error: unknown flag: --save-config
See 'oc apply --help' for usage.

[jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$ oc apply -f nginx-deploy.yml
deployment.apps/nginx configured

[jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$ oc get deploy,rs,po
NAME           READY   UP-TO-DATE   AVAILABLE   AGE
deployment.apps/nginx   2/2     2          2          3m46s

NAME           DESIRED   CURRENT   READY   AGE
replicaset.apps/nginx-5bccb79775   2        2        2        3m46s

NAME           READY   STATUS    RESTARTS   AGE
pod/nginx-5bccb79775-7f5cs   1/1     Running   0         3m46s
pod/nginx-5bccb79775-86xvk   1/1     Running   0         3m46s

[jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$
```

Lab - Scale up nginx deployment in declarative style

Update the nginx-deploy.yml, replicas value from 2 to 5 and save it before apply it.

```
jegan@tekktutor:~/openshift-sep-2023/Day3/declarative-manifests
jegan@tekktutor:~/ansible-sep-2023
jegan@tekktutor:~/openshift-sep-2023
jegan@tekktutor:~/openshift-sep-2023/Day1
jegan@tekktutor:~/openshift-sep-2023/Day3/declarative-manifests

apiVersion: apps/v1
kind: Deployment
metadata:
  labels:
    app: nginx
  name: nginx
spec:
  replicas: 5
  selector:
    matchLabels:
      app: nginx
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
      - image: bitnami/nginx:latest
        name: nginx

"nginx-deploy.yml" 19L, 295B written
```

Expected output

```
jegan@tektutor: ~/openshift-sep-2023/Day3/declarative-manifests
[jegan@tektutor.org]-(~/openshift-sep-2023)
$ cd Day3/declarative-manifests

[jegan@tektutor.org]-(~/openshift-sep-2023/Day3/declarative-manifests]
$ ls
nginx-deploy.yml

[jegan@tektutor.org]-(~/openshift-sep-2023/Day3/declarative-manifests]
$ vim nginx-deploy.yml

[jegan@tektutor.org]-(~/openshift-sep-2023/Day3/declarative-manifests]
$ oc apply -f nginx-deploy.yml
deployment.apps/nginx configured

[jegan@tektutor.org]-(~/openshift-sep-2023/Day3/declarative-manifests]
$ oc get po -w
NAME          READY   STATUS      RESTARTS   AGE
nginx-5bccb79775-5n6x5  0/1     ContainerCreating  0          4s
nginx-5bccb79775-7f5cs  1/1     Running     0          22m
nginx-5bccb79775-86xvk  1/1     Running     0          22m
nginx-5bccb79775-pp27q  0/1     ContainerCreating  0          4s
nginx-5bccb79775-vgrsl  0/1     ContainerCreating  0          4s
nginx-5bccb79775-vgrsl  1/1     Running     0          5s
nginx-5bccb79775-pp27q  1/1     Running     0          6s
nginx-5bccb79775-5n6x5  1/1     Running     0          6s
^C

[jegan@tektutor.org]-(~/openshift-sep-2023/Day3/declarative-manifests]
$ [REDACTED]

[jegan@tektutor.org]-(~/openshift-sep-2023/Day3/declarative-manifests]
$ oc apply -f nginx-deploy.yml
deployment.apps/nginx configured

[jegan@tektutor.org]-(~/openshift-sep-2023/Day3/declarative-manifests]
$ oc get po -w
NAME          READY   STATUS      RESTARTS   AGE
nginx-5bccb79775-5n6x5  0/1     ContainerCreating  0          4s
nginx-5bccb79775-7f5cs  1/1     Running     0          22m
nginx-5bccb79775-86xvk  1/1     Running     0          22m
nginx-5bccb79775-pp27q  0/1     ContainerCreating  0          4s
nginx-5bccb79775-vgrsl  0/1     ContainerCreating  0          4s
nginx-5bccb79775-vgrsl  1/1     Running     0          5s
nginx-5bccb79775-pp27q  1/1     Running     0          6s
nginx-5bccb79775-5n6x5  1/1     Running     0          6s
^C

[jegan@tektutor.org]-(~/openshift-sep-2023/Day3/declarative-manifests]
$ oc get po
NAME          READY   STATUS      RESTARTS   AGE
nginx-5bccb79775-5n6x5  1/1     Running     0          21s
nginx-5bccb79775-7f5cs  1/1     Running     0          22m
nginx-5bccb79775-86xvk  1/1     Running     0          22m
nginx-5bccb79775-pp27q  1/1     Running     0          21s
nginx-5bccb79775-vgrsl  1/1     Running     0          21s

[jegan@tektutor.org]-(~/openshift-sep-2023/Day3/declarative-manifests]
$ [REDACTED]
```

Lab - Scale down nginx deployment in declarative style

Update the nginx-deploy.yml file replicase value from 5 to 2, save it before applying it.

```
cd ~/openshift-sep-2023
git pull
cd Day3/declarative-manifests
gedit nginx-deploy.yml
cat nginx-deploy.yml
oc apply -f nginx-deploy.yml
oc get po
```

Expected output

The screenshot shows a code editor window with a dark theme. At the top, there's a toolbar with icons for Open, Save, and others. The file path is shown as ~/openshift-scp-2023/Dry/declarative-manifests. The code itself is a YAML manifest for a Deployment named 'nginx'. It specifies two replicas, both labeled 'app: nginx', and uses a template with matching labels. Each replica contains a single container named 'nginx' based on the 'bitnami/nginx' image.

```
1 apiVersion: apps/v1
2 kind: Deployment
3 metadata:
4   labels:
5     app: nginx
6   name: nginx
7 spec:
8   replicas: 2
9   selector:
10    matchLabels:
11      app: nginx
12   template:
13     metadata:
14       labels:
15         app: nginx
16     spec:
17       containers:
18         - image: bitnami/nginx:latest
19           name: nginx
```

At the bottom right of the editor, there are status indicators: YAML, Tab Width: 8, Ln 2, Col 14, and INS.

```
jegan@tektutor: ~ /openshift-sep-2023/Day3/declarative-manifests
$ cat nginx-deploy.yml
apiVersion: apps/v1
kind: Deployment
metadata:
  labels:
    app: nginx
    name: nginx
spec:
  replicas: 2
  selector:
    matchLabels:
      app: nginx
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
        - image: bitnami/nginx:latest
          name: nginx

(jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$ oc apply -f nginx-deploy.yml
deployment.apps/nginx configured

(jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$ 

jegan@tektutor: ~ /openshift-sep-2023/Day3/declarative-manifests
jegan@tektutor: ~/ansible-sep-2023 x jegan@tektutor: ~/openshift-sep-2023 x jegan@tektutor: ~/openshift-sep-2023/Day1 x jegan@tektutor: ~/openshift-sep-2023/Day3/d...
(jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
labels:
  app: nginx
  name: nginx
spec:
  replicas: 2
  selector:
    matchLabels:
      app: nginx
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
        - image: bitnami/nginx:latest
          name: nginx

(jegan@tektutor.org)-[~/openshift-sep-2023/declarative-manifests]
$ oc apply -f nginx-deploy.yml
deployment.apps/nginx configured

(jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$ oc get po
NAME           READY   STATUS    RESTARTS   AGE
nginx-5bccb79775-7f5cs  1/1     Running   0          29m
nginx-5bccb79775-86xvk  1/1     Running   0          29m

(jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$ 
```

Lab - Using Rolling update to upgrade your live application from one version to other without downtime

First you can delete the nginx deploy that is already running in the cluster.

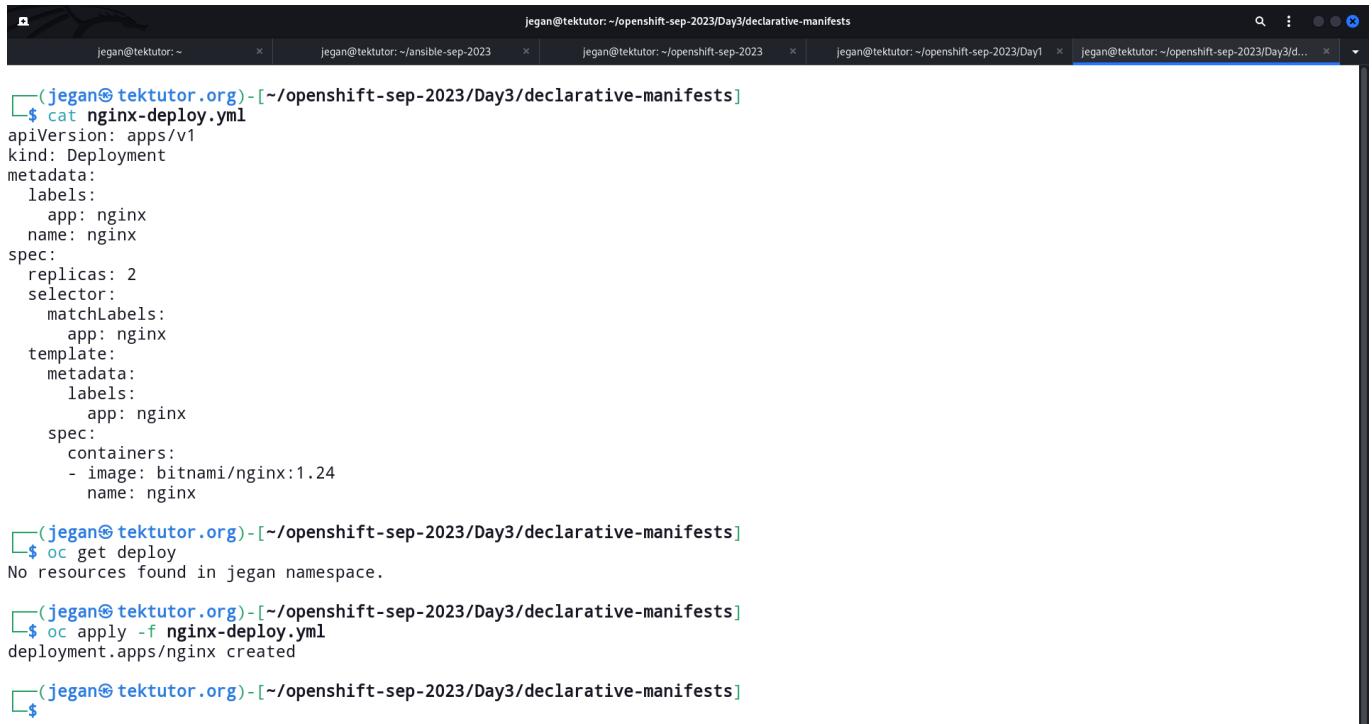
```
cd ~/openshift-sep-2023
git pull
cd Day3/declarative-manifests

oc delete -f nginx-deploy.yml
```

Then update the image in the nginx-deploy.yml from "bitnami/nginx:latest" to "bitnami/nginx:1.24" and save it.
 You can now apply/create this change into the cluster

```
cd ~/openshift-sep-2023
git pull
cd Day3/declarative-manifests
cat nginx-deploy.yml
oc get deploy
oc apply -f nginx-deploy.yml
```

Expected output

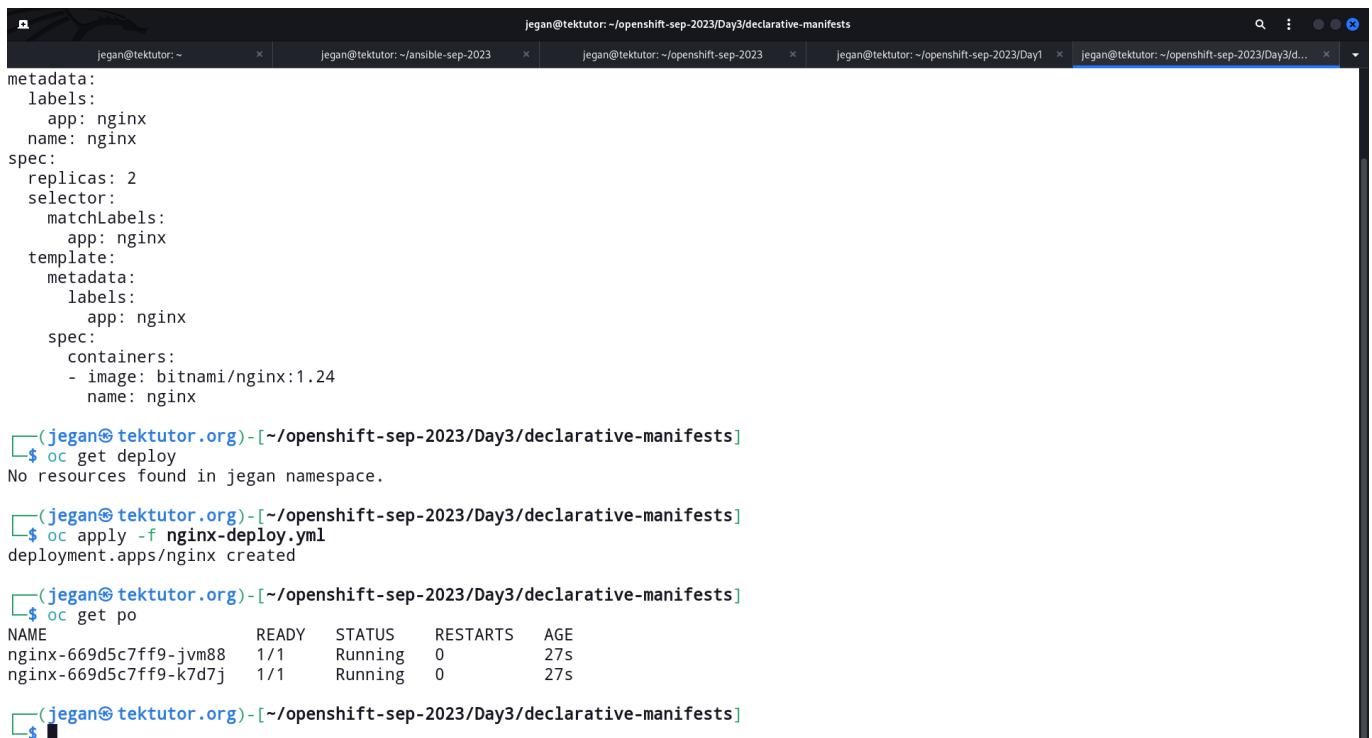


```
(jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$ cat nginx-deploy.yml
apiVersion: apps/v1
kind: Deployment
metadata:
  labels:
    app: nginx
  name: nginx
spec:
  replicas: 2
  selector:
    matchLabels:
      app: nginx
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
        - image: bitnami/nginx:1.24
          name: nginx

(jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$ oc get deploy
No resources found in jegan namespace.

(jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$ oc apply -f nginx-deploy.yml
deployment.apps/nginx created

(jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$
```

```
jegan@tektutor:~/openshift-sep-2023/Day3/declarative-manifests
jegan@tektutor:~ | jegan@tektutor:~/ansible-sep-2023 | jegan@tektutor:~/openshift-sep-2023 | jegan@tektutor:~/openshift-sep-2023/Day1 | jegan@tektutor:~/openshift-sep-2023/Day3/d...
metadata:
  labels:
    app: nginx
  name: nginx
spec:
  replicas: 2
  selector:
    matchLabels:
      app: nginx
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
        - image: bitnami/nginx:1.24
          name: nginx

(jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$ oc get deploy
No resources found in jegan namespace.

(jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$ oc apply -f nginx-deploy.yml
deployment.apps/nginx created

(jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$ oc get po
NAME           READY   STATUS    RESTARTS   AGE
nginx-669d5c7ff9-jvm88  1/1     Running   0          27s
nginx-669d5c7ff9-k7d7j  1/1     Running   0          27s

(jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$
```

Checking the rolling update status

```
oc rollout status deploy/nginx
```

Expected output

The screenshot shows a terminal window with five tabs open, all showing the same user: jegan@tektutor: ~. The active tab is the fourth one, titled 'jegan@tektutor: ~/openshift-sep-2023/Day3/declarative-manifests'. The terminal displays the following command and its output:

```
jegan@tektutor: ~/openshift-sep-2023/Day3/declarative-manifests
$ oc rollout status deploy/nginx
deployment "nginx" successfully rolled out
```

Below this, there is a large block of YAML manifest code for a deployment named 'nginx'.

```
spec:
  replicas: 2
  selector:
    matchLabels:
      app: nginx
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
        - image: bitnami/nginx:1.24
          name: nginx
```

At the bottom of the terminal window, there is a prompt: '(jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]\$'.

Checking the rolling update revision history

```
oc rollout history deploy/nginx
```

Expected output

```
jegan@tektutor: ~/openshift-sep-2023/Day3/declarative-manifests
jegan@tektutor: ~ | jegan@tektutor: ~/ansible-sep-2023 | jegan@tektutor: ~/openshift-sep-2023 | jegan@tektutor: ~/openshift-sep-2023/Day1 | jegan@tektutor: ~/openshift-sep-2023/Day3/d...
labels:
  app: nginx
spec:
  containers:
    - image: bitnami/nginx:1.24
      name: nginx

[jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$ oc get deploy
No resources found in jegan namespace.

[jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$ oc apply -f nginx-deploy.yml
deployment.apps/nginx created

[jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$ oc get po
NAME          READY   STATUS    RESTARTS   AGE
nginx-669d5c7ff9-jvm88  1/1     Running   0          27s
nginx-669d5c7ff9-k7d7j  1/1     Running   0          27s

[jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$ oc rollout status deploy/nginx
deployment "nginx" successfully rolled out

[jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$ oc rollout history deploy/nginx
deployment.apps/nginx
REVISION  CHANGE-CAUSE
1          <none>

[jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$
```

Let's update the image version from 1.24 to 1.25

```
cat nginx-deploy.yml
oc apply -f nginx-deploy.yml
oc get po -w
oc get po
```

Expected output

```
jegan@tektutor: ~/openshift-sep-2023/Day3/declarative-manifests
jegan@tektutor: ~ | jegan@tektutor: ~/ansible-sep-2023 | jegan@tektutor: ~/openshift-sep-2023 | jegan@tektutor: ~/openshift-sep-2023/Day1 | jegan@tektutor: ~/openshift-sep-2023/Day3/d...
[jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$ oc rollout history deploy/nginx
deployment.apps/nginx
REVISION  CHANGE-CAUSE
1          <none>

[jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$ vim nginx-deploy.yml
[jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$ cat nginx-deploy.yml
apiVersion: apps/v1
kind: Deployment
metadata:
  labels:
    app: nginx
  name: nginx
spec:
  replicas: 2
  selector:
    matchLabels:
      app: nginx
  template:
    metadata:
      labels:
        app: nginx
  spec:
    containers:
      - image: bitnami/nginx:1.25
        name: nginx

[jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$
```

```
jegan@tektutor:~/openshift-sep-2023/Day3/declarative-manifests
jegan@tektutor:~/ansible-sep-2023
jegan@tektutor:~/openshift-sep-2023
jegan@tektutor:~/openshift-sep-2023/Day1
jegan@tektutor:~/openshift-sep-2023/Day3/d...

spec:
  containers:
    - image: bitnami/nginx:1.25
      name: nginx

[jegan@tektutor.org] - [~/openshift-sep-2023/Day3/declarative-manifests]
$ oc apply -f nginx-deploy.yaml
deployment.apps/nginx configured

[jegan@tektutor.org] - [~/openshift-sep-2023/Day3/declarative-manifests]
$ oc get rs
NAME      DESIRED   CURRENT   READY   AGE
nginx-669d5c7ff9   2         2         2       3m17s
nginx-7567ff57d4   1         1         0       4s

[jegan@tektutor.org] - [~/openshift-sep-2023/Day3/declarative-manifests]
$ oc ge po -w
error: unknown command "ge" for "oc"

Did you mean this?
  cp
  get
  set

[jegan@tektutor.org] - [~/openshift-sep-2023/Day3/declarative-manifests]
$ oc get po -w
NAME        READY   STATUS    RESTARTS   AGE
nginx-669d5c7ff9-k7d7j  1/1     Running   0          3m24s
nginx-7567ff57d4-grl1v4 1/1     Running   0          11s
nginx-7567ff57d4-rcz54  0/1     ContainerCreating   0          1s
nginx-7567ff57d4-rcz54  0/1     ContainerCreating   0          1s
nginx-7567ff57d4-rcz54  1/1     Running   0          9s
nginx-669d5c7ff9-k7d7j  1/1     Terminating   0          3m32s
nginx-669d5c7ff9-k7d7j  0/1     Terminating   0          3m33s
nginx-669d5c7ff9-k7d7j  0/1     Terminating   0          3m33s
nginx-669d5c7ff9-k7d7j  0/1     Terminating   0          3m33s
^C

[jegan@tektutor.org] - [~/openshift-sep-2023/Day3/declarative-manifests]
$ oc get po
NAME        READY   STATUS    RESTARTS   AGE
nginx-7567ff57d4-grl1v4 1/1     Running   0          36s
nginx-7567ff57d4-rcz54  1/1     Running   0          26s
```

Rolling back from 1.25 to 1.24

```
oc rollout undo deploy/nginx
```

Expected output

```
jegan@tektutor:~/openshift-sep-2023/Day3/declarative-manifests
jegan@tektutor:~/ansible-sep-2023
jegan@tektutor:~/openshift-sep-2023
jegan@tektutor:~/openshift-sep-2023/Day1
jegan@tektutor:~/openshift-sep-2023/Day3/d...

[jegan@tektutor.org] - [~/openshift-sep-2023/Day3/declarative-manifests]
$ oc rollout status deploy/nginx
deployment "nginx" successfully rolled out

[jegan@tektutor.org] - [~/openshift-sep-2023/Day3/declarative-manifests]
$ oc rollout history deploy/nginx
deployment.apps/nginx
REVISION  CHANGE-CAUSE
1          <none>
2          <none>

[jegan@tektutor.org] - [~/openshift-sep-2023/Day3/declarative-manifests]
$ oc rollout undo deploy/nginx
deployment.apps/nginx rolled back

[jegan@tektutor.org] - [~/openshift-sep-2023/Day3/declarative-manifests]
$ oc get po -w
NAME        READY   STATUS    RESTARTS   AGE
nginx-669d5c7ff9-qqpww  1/1     Running   0          3s
nginx-669d5c7ff9-qzt7k  0/1     ContainerCreating   0          1s
nginx-7567ff57d4-rcz54  1/1     Running   0          8m55s
nginx-669d5c7ff9-qzt7k  0/1     ContainerCreating   0          1s
nginx-669d5c7ff9-qzt7k  1/1     Running   0          2s
nginx-7567ff57d4-rcz54  1/1     Terminating   0          8m56s
nginx-7567ff57d4-rcz54  0/1     Terminating   0          8m57s
nginx-7567ff57d4-rcz54  0/1     Terminating   0          8m57s
nginx-7567ff57d4-rcz54  0/1     Terminating   0          8m57s
^C

[jegan@tektutor.org] - [~/openshift-sep-2023/Day3/declarative-manifests]
$ oc get po
NAME        READY   STATUS    RESTARTS   AGE
nginx-669d5c7ff9-qqpww  1/1     Running   0          12s
nginx-669d5c7ff9-qzt7k  1/1     Running   0          10s

[jegan@tektutor.org] - [~/openshift-sep-2023/Day3/declarative-manifests]
```

To cross-check what image version the nginx pods are using after rollback, you can try the below command

```
oc edit pod/nginx-669d5c7ff9-qzt7k
```

Expected outt

```

jegan@tektutor:~/openshift-sep-2023/Day3/declarative-manifests
jegan@tektutor:~/ansible-sep-2023 x | jegan@tektutor:~/openshift-sep-2023 x | jegan@tektutor:~/openshift-sep-2023 x | jegan@tektutor:~/openshift-sep-2023/Day1 x | jegan@tektutor:~/openshift-sep-2023/Day3/d... x |
# Please edit the object below. Lines beginning with a '#' will be ignored,
# and an empty file will abort the edit. If an error occurs while saving this file will be
# reopened with the relevant failures.
#
apiVersion: v1
kind: Pod
metadata:
  annotations:
    k8s.v1.cni.cncf.io/network-status: |->
      [
        {
          "name": "openshift-sdn",
          "interface": "eth0",
          "ips": [
            "10.129.1.28"
          ],
          "default": true,
          "dns": {}
        }
      ]
    openshift.io/scc: restricted-v2
    seccomp.security.alpha.kubernetes.io/pod: runtime/default
  creationTimestamp: "2023-09-13T07:38:08Z"
  generateName: nginx-669d5c7ff9-
  labels:
    app: nginx
    pod-template-hash: 669d5c7ff9
  name: nginx-669d5c7ff9-qzt7k
  namespace: jegan
  ownerReferences:
  - apiVersion: apps/v1
    blockOwnerDeletion: true
    controller: true
    kind: ReplicaSet
    name: nginx-669d5c7ff9
    uid: 19b8cfa-8eaf-4bd4-b591-91dc3adeeaab
    resourceVersion: "2839544"
    uid: a57415e2-cb6c-43e9-8a0f-c50f58029c71
spec:
  containers:
  - image: bitnami/nginx:1.24
    imagePullPolicy: IfNotPresent
"/tmp/oc-edit-2296437425.yaml" 143L, 3896B

```

2,1 Top

Lab - Creating a clusterip internal service using declarative style

```
oc expose deploy/nginx --type=ClusterIP --port=8080 --dry-run=client -o
yaml
```

Let's redirect the output to a file

```
oc expose deploy/nginx --type=ClusterIP --port=8080 --dry-run=client -o
yaml > nginx-clusterip-svc.yaml
```

Expected output Expected output

```
jegan@tektutor: ~ | jegan@tektutor: ~/openshift-sep-2023 | jegan@tektutor: ~/ansible-sep-2023 | jegan@tektutor: ~/openshift-sep-2023 | jegan@tektutor: ~/openshift-sep-2023/Day1 | jegan@tektutor: ~/openshift-sep-2023/Day3/declarative-manifests
jegan@tektutor: ~ | jegan@tektutor: ~/openshift-sep-2023/Day3/declarative-manifests | jegan@tektutor: ~/openshift-sep-2023/Day3/declarative-manifests | jegan@tektutor: ~/openshift-sep-2023/Day3/declarative-manifests | jegan@tektutor: ~/openshift-sep-2023/Day3/declarative-manifests | jegan@tektutor: ~/openshift-sep-2023/Day3/declarative-manifests
NAME      READY   STATUS    RESTARTS   AGE
nginx-669d5c7ff9-qqpww  1/1     Running   0          55m
nginx-669d5c7ff9-qzt7k  1/1     Running   0          55m

(jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$ oc expose deploy/nginx --type=ClusterIP --port=8080 --dry-run=client -o yaml
apiVersion: v1
kind: Service
metadata:
  creationTimestamp: null
  labels:
    app: nginx
    name: nginx
spec:
  ports:
  - port: 8080
    protocol: TCP
    targetPort: 8080
  selector:
    app: nginx
  type: ClusterIP
status:
  loadBalancer: {}

(jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$ oc expose deploy/nginx --type=ClusterIP --port=8080 --dry-run=client -o yaml > nginx-clusterip-svc.yaml

(jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$
```

Let's create the ClusterIP Internal service using the yaml file

```
cd ~/openshift-sep-2023
git pull
cd Day3/declarative-manifests
oc create -f nginx-clusterip-svc.yaml --save-config
```

Expected output

```
jegan@tektutor: ~ | jegan@tektutor: ~/ansible-sep-2023 | jegan@tektutor: ~/openshift-sep-2023 | jegan@tektutor: ~/openshift-sep-2023/Day1 | jegan@tektutor: ~/openshift-sep-2023/Day3/declarative-manifests | jegan@tektutor: ~/openshift-sep-2023/Day3/declarative-manifests
jegan@tektutor: ~ | jegan@tektutor: ~/openshift-sep-2023/Day3/declarative-manifests | jegan@tektutor: ~/openshift-sep-2023/Day3/declarative-manifests | jegan@tektutor: ~/openshift-sep-2023/Day3/declarative-manifests | jegan@tektutor: ~/openshift-sep-2023/Day3/declarative-manifests | jegan@tektutor: ~/openshift-sep-2023/Day3/declarative-manifests
(jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$ oc create -f nginx-clusterip-svc.yaml --save-config
service/nginx created

(jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$ oc get svc
NAME      TYPE      CLUSTER-IP      EXTERNAL-IP      PORT(S)      AGE
nginx     ClusterIP  172.30.245.26  <none>           8080/TCP    4s

(jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$ oc describe svc/nginx
Name:           nginx
Namespace:      jegan
Labels:         app=nginx
Annotations:   <none>
Selector:       app=nginx
Type:          ClusterIP
IP Family Policy: SingleStack
IP Families:   IPv4
IP:             172.30.245.26
IPs:            172.30.245.26
Port:           <unset>  8080/TCP
TargetPort:     8080/TCP
Endpoints:     10.128.1.103:8080,10.129.1.28:8080
Session Affinity: None
Events:         <none>

(jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$
```

Lab - Creating a nodeport external service for nginx deployment in declarative style

```
oc delete -f nginx-clusterip-svc.yml  
oc expose deploy/nginx --type=NodePort --port=8080 --dry-run=client -o yaml > nginx-nodeport-svc.yml  
cat nginx-nodeport-svc.yml
```

Expected output

The screenshot shows a terminal window with multiple tabs open. The active tab is titled '(jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]'. The command history and output are as follows:

```
jegan@tektutor:~ | jegan@tektutor:~/ansible-sep-2023 | jegan@tektutor:~/openshift-sep-2023 | jegan@tektutor:~/openshift-sep-2023/Day1 | jegan@tektutor:~/openshift-sep-2023/Day3/d...  
[jegan@tektutor.org]-[~/openshift-sep-2023/Day3/declarative-manifests]  
$ oc delete -f nginx-clusterip-svc.yml  
service "nginx" deleted  
  
[jegan@tektutor.org]-[~/openshift-sep-2023/Day3/declarative-manifests]  
$ oc expose deploy/nginx --type=NodePort --port=8080 --dry-run=client -o yaml > nginx-nodeport-svc.yml  
  
[jegan@tektutor.org]-[~/openshift-sep-2023/Day3/declarative-manifests]  
$ cat nginx-nodeport-svc.yml  
apiVersion: v1  
kind: Service  
metadata:  
  creationTimestamp: null  
  labels:  
    app: nginx  
    name: nginx  
spec:  
  ports:  
  - port: 8080  
    protocol: TCP  
    targetPort: 8080  
  selector:  
    app: nginx  
  type: NodePort  
status:  
  loadBalancer: {}  
  
$
```

You could create the nodeport sevice as shown below

```
cd ~/openshift-sep-2023  
git pull  
cd Day3/declarative-manifests  
oc apply -f nginx-nodeport-svc.yml  
oc get svc  
oc describe svc/nginx
```

Expected output

```
jegan@tektutor: ~/openshift-sep-2023/Day3/declarative-manifests
jegan@tektutor: ~/ansible-sep-2023 x | jegan@tektutor: ~/openshift-sep-2023 x | jegan@tektutor: ~/openshift-sep-2023/Day1 x | jegan@tektutor: ~/openshift-sep-2023/Day3/d... x
status:
loadBalancer: {}

[jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$ oc apply -f nginx-nodeport-svc.yml
service/nginx created

[jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$ oc get svc
NAME      TYPE      CLUSTER-IP      EXTERNAL-IP      PORT(S)      AGE
nginx    NodePort   172.30.74.122  <none>        8080:31212/TCP   2s

[jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$ oc describe svc/nginx
Name:           nginx
Namespace:      jegan
Labels:         app=nginx
Annotations:   <none>
Selector:       app=nginx
Type:          NodePort
IP Family Policy: SingleStack
IP Families:   IPv4
IP:             172.30.74.122
IPs:            172.30.74.122
Port:           <unset>  8080/TCP
TargetPort:     8080/TCP
NodePort:       <unset>  31212/TCP
Endpoints:     10.128.1.103:8080,10.129.1.28:8080
Session Affinity: None
External Traffic Policy: Cluster
Events:        <none>

[jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$
```

Lab - Creating a LoadBalancer service in declarative style

Let's delete the existing service before creating loadbalancer external service

```
cd ~/openshift-sep-2023
git pull
cd Day3/declarative-manifests
oc delete -f nginx-nodeport-svc.yml
```

Let's create the loadbalancer service

```
cd ~/openshift-sep-2023
git pull
cd Day3/declarative-manifests

oc expose deploy/nginx --type=LoadBalancer --port=8080 --dry-run=client -o
yaml > nginx-lb-svc.yml
cat nginx-lb-svc.yml
oc apply -f nginx-lb-svc.yml
oc get svc
oc describe svc/nginx
```

Expected output

```
jegan@tektutor: ~ | jegan@tektutor: ~/ansible-sep-2023 | jegan@tektutor: ~/openshift-sep-2023 | jegan@tektutor: ~/openshift-sep-2023/Day1 | jegan@tektutor: ~/openshift-sep-2023/Day3/d... [~/openshift-sep-2023/declarative-manifests]
[jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$ oc delete -f nginx-nodeport-svc.yml
service "nginx" deleted

[jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$ oc expose deploy/nginx --type=LoadBalancer --port=8080 --dry-run=client -o yaml > nginx-lb-svc.yml

[jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$ cat nginx-lb-svc.yml
apiVersion: v1
kind: Service
metadata:
  creationTimestamp: null
  labels:
    app: nginx
    name: nginx
spec:
  ports:
    - port: 8080
      protocol: TCP
      targetPort: 8080
  selector:
    app: nginx
    type: LoadBalancer
status:
  loadBalancer: {}

[jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$ oc apply -f nginx-lb-svc.yml
service/nginx created

[jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$ 

[jegan@tektutor: ~ | jegan@tektutor: ~/ansible-sep-2023 | jegan@tektutor: ~/openshift-sep-2023 | jegan@tektutor: ~/openshift-sep-2023/Day1 | jegan@tektutor: ~/openshift-sep-2023/Day3/d... [~/openshift-sep-2023/declarative-manifests]
[jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$ oc get svc
NAME      TYPE      CLUSTER-IP      EXTERNAL-IP      PORT(S)      AGE
nginx    LoadBalancer   172.30.189.215  192.168.122.50  8080:30765/TCP  3m16s

[jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$ oc describe svc/nginx
Name:           nginx
Namespace:      jegan
Labels:         app=nginx
Annotations:   metallb.universe.tf/ip-allocated-from-pool: tektutor-metallb-addresspool
Selector:       app=nginx
Type:          LoadBalancer
IP Family Policy: SingleStack
IP Families:   IPv4
IP:             172.30.189.215
IPs:            172.30.189.215
LoadBalancer Ingress: 192.168.122.50
Port:           <unset> 8080/TCP
TargetPort:     8080/TCP
NodePort:       <unset> 30765/TCP
Endpoints:     10.128.1.103:8080,10.129.1.28:8080
Session Affinity: None
External Traffic Policy: Cluster
Events:
  Type  Reason  Age  From           Message
  ----  -----  ---  ---           ---
  Normal  IPAllocated  3m11s  metallb-controller  Assigned IP ["192.168.122.50"]
  Normal  nodeAssigned  3m11s  metallb-speaker    announcing from node "worker-2.ocp.tektutor-ocp-labs" with protocol "layer2"

[jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$ 
```

Info - What is Service?

- service represents a group of similar pods that belongs to a single deployment in Kubernetes/OpenShift
- they are load-balanced by default
- they are highly available
- application that run in Kubernetes/OpenShift aren't supposed to invoke Pod by IP/host, as Pods can be deleted/replaced any point of time
- so the only reliable/stable abstraction your application can depend is the service

- when a deployment is scaled up/down, the service pod endpoints are automatically updated, hence your application is decoupled from the Pod instance, this make it loosely coupled design

Info - How to decide which type of service is suitable for a application deployment

There are 3 types of Services in Kubernetes/OpenShift

1. ClusterIP Internal Service
2. NodePort External Service
3. LoadBalancer External Service

CluterIP Internal Service

- This type of Service is typically use for database deployments
- This service can only be accessed from within the OpenShift cluster
- As the application that needs database access runs within the OpenShift cluster, the ClusterIP db service will work perfectly
- This is inbuilt feature of Kubernetes/OpenShift, hence there won't be any additional cost implications when you create ClusterIP Services either locally or in public cloud based OpenShift cluster

NodePort External Service

- This type of Service is required to access the application from outside the OpenShift cluster
- Typically for any front-end application or Microservices will require this type of service
- Kubernetes/OpenShift has reserved 30000 to 32767 Port range on every Node in the Cluster for NodePort services
- For each NodePort service we create in the OpenShift cluster, OpenShift will assign a port out of the range 30000-32767 whichever port is available on all the nodes in cluster
- The NodePort will be opened up in every node in the OpenShift cluster
- So each time you create a nodeport service it opens that port on every server in the OpenShift cluster, this might lead to some security issues
- Also the way we access the nodeport service is not end-user friendly, hence this is suitable only for front-end application. Frontend application or Microservices can make use of NodePort.

LoadBalancer External Service

- This type of Service is used when OpenShift runs in public cloud like AWS/Azure/GCP/Digital Ocean, etc.,
- Each LoadBalancer service will create an External Load Balancer within AWS/Azure, hence there would be dedicated Server where the external Load Balancer works, unlike NodePort service or ClusterIp Service
- Hence, LoadBalancer Service is more reliable
- There is a cost implication each time you create a LoadBalancer Service in public cloud, as the public cloud vendor will charge on monthly basis for the LoadBalancer service
- Loadbalancer also gets allocation one unique static ip, which is accessible over Internet, hence only one person/company can own that unique static ip, so there is a cost from static ip as well

Lab - Creating a external route as an alternate for NodePort service

Things to note

- Route is only supported by OpenShift and not available in Kubernetes
- Route provides a public url that is accessible outside the cluster
- this serves as an alternate to nodeport
- this is based on Ingress feature in Kubernetes
- Ingress is not a service, it is a routing rule used in Load Balancer
- the Ingress Controller picks the Ingress rule and configures the Load Balancer with those routing rule
- Ingress supports routing calls to multiple different services, while the OpenShift route generally routes/forwards the call to just one service

Let's delete the existing lb service

```
cd ~/openshift-sep-2023
git pull
cd Day3/declarative-manifests
oc delete -f nginx-lb-svc.yml
```

Let's create a ClusterIP Internal Service

```
cd ~/openshift-sep-2023
git pull
cd Day3/declarative-manifests
oc apply -f nginx-clusterip-svc.yml
```

Expected output

```
(jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$ oc delete -f nginx-lb-svc.yml
service "nginx" deleted

(jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$ oc get svc
No resources found in jegan namespace.

(jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$ oc get deploy
NAME      READY   UP-TO-DATE   AVAILABLE   AGE
nginx    2/2       2           2           114m

(jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$ oc apply -f nginx-clusterip-svc.yml
service/nginx created

(jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$ oc expose svc/nginx --dry-run=client -o yaml > nginx-route.yaml

(jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$ cat nginx-route.yaml
apiVersion: route.openshift.io/v1
kind: Route
metadata:
  creationTimestamp: null
  labels:
    app: nginx
    name: nginx
spec:
  port:
    targetPort: 8080
  to:
    .. . . .
```

Let's expose the service via route to make it accessible outside the cluster with a public url

```
cd ~/openshift-sep-2023
git pull
cd Day3/declarative-manifests

oc expose svc/nginx --dry-run=client -o yaml > nginx-route.yml
cat nginx-route.yml
oc create -f nginx-route.yml --save-config
```

Expected output

The screenshot shows a terminal window with five tabs open. The active tab is titled 'jegan@tektutor: ~/openshift-sep-2023/Day3/declarative-manifests'. The command history shows:

- \$ oc expose svc/nginx --dry-run=client -o yaml > nginx-route.yml
- \$ cat nginx-route.yml
- apiVersion: route.openshift.io/v1
kind: Route
metadata:
 creationTimestamp: null
 labels:
 app: nginx
 name: nginx
spec:
 port:
 targetPort: 8080
 to:
 kind: ""
 name: nginx
 weight: null
status: {}
- \$ oc create -f nginx-route.yml
route.route.openshift.io/nginx created
- \$ oc get route
NAME HOST/PORT PATH SERVICES PORT TERMINATION WILDCARD
nginx nginx-jegan.apps.ocp.tektutor-ocp-labs nginx 8080 None

Accessing the external route

```
oc get route
curl http://nginx-jegan.apps.ocp.tektutor-ocp-labs:80
```

Expected output

```
jegan@tektutor: ~/openshift-sep-2023/Day3/declarative-manifests
[jegan@tektutor: ~] jegan@tektutor: ~/ansible-sep-2023 [jegan@tektutor: ~/openshift-sep-2023 [jegan@tektutor: ~/openshift-sep-2023/Day1 [jegan@tektutor: ~/openshift-sep-2023/Day3/d...
(jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$ oc get route
NAME    HOST/PORT        PATH  SERVICES   PORT  TERMINATION WILDCARD
nginx  nginx-jegan.apps.ocp.tektutor-ocp-labs  nginx      8080       None

(jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$ curl http://nginx-jegan.apps.ocp.tektutor-ocp-labs:80
<!DOCTYPE html>
<html>
<head>
<title>Welcome to nginx!</title>
<style>
html { color-scheme: light dark; }
body { width: 35em; margin: 0 auto;
font-family: Tahoma, Verdana, Arial, sans-serif; }
</style>
</head>
<body>
<h1>Welcome to nginx!</h1>
<p>If you see this page, the nginx web server is successfully installed and
working. Further configuration is required.</p>
<p>For online documentation and support please refer to
<a href="http://nginx.org/">nginx.org</a>.<br/>
Commercial support is available at
<a href="http://nginx.com/">nginx.com</a>.</p>
<p><em>Thank you for using nginx.</em></p>
</body>
</html>

(jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$
```

Lab - Creating a Pod without replicaset/deployment

Things to Note

- Creating a Pod without replicaset/deployment is considered a bad practice
- In case the Pod crashes, as there is no replicaset to monitor and repair, we need to manually fix the error
- Hence, this must be done only for testing purpose and not in production

```
oc run hello --image=tektutor/spring-tektutor-hellos:latest
```

Expected output

```
(jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$ oc get po
NAME READY STATUS RESTARTS AGE
nginx-669d5c7ff9-qqpww 1/1 Running 0 120m
nginx-669d5c7ff9-qzt7k 1/1 Running 0 120m

(jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$ oc run hello --image=tektutor/spring-tektutor-hellos:latest
Warning: would violate PodSecurity "restricted:v1.24": allowPrivilegeEscalation != false (container "hello" must set securityContext.allowPrivilegeEscalation=false), unrestricted capabilities (container "hello" must set securityContext.capabilities.drop=["ALL"]), runAsNonRoot != true (pod or container "hello" must set securityContext.runAsNonRoot=true), seccompProfile (pod or container "hello" must set securityContext.seccompProfile.type to "RuntimeDefault" or "Localhost")
pod/hello created

(jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$ oc get deploy,rs,po
NAME READY UP-TO-DATE AVAILABLE AGE
deployment.apps/nginx 2/2 2 2 133m

NAME DESIRED CURRENT READY AGE
replicaset.apps/nginx-669d5c7ff9 2 2 2 133m
replicaset.apps/nginx-7567ff57d4 0 0 0 129m

NAME READY STATUS RESTARTS AGE
pod/hello 1/1 Running 0 7s
pod/nginx-669d5c7ff9-qqpww 1/1 Running 0 120m
pod/nginx-669d5c7ff9-qzt7k 1/1 Running 0 120m

(jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$
```

Lab - Creating a ReplicaSet without Deployment

Let's delete the project

```
oc delete project/jegan
```

Let's create a new project

```
cd ~/openshift-sep-2023
git pull
cd Day3/declarative-manifests

oc new-project jegan
ls
oc get all
oc create -f nginx-rs.yml
oc get deploy,rs,po
oc delete pod/nginx-rs-gsj4j
oc delete rs/nginx-rs
oc get all
```

Expected output

```
jegan@tektutor:~/openshift-sep-2023/Day3/declarative-manifests
$ oc delete project/jegan
project.project.openshift.io "jegan" deleted

(jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$ oc new-project jegan
Already on project "jegan" on server "https://api.ocp.tektutor-ocp-labs:6443".

You can add applications to this project with the 'new-app' command. For example, try:

  oc new-app rails-postgresql-example

to build a new example application in Ruby. Or use kubectl to deploy a simple Kubernetes application:

  kubectl create deployment hello-node --image=registry.k8s.io/e2e-test-images/agnhost:2.43 -- /agnhost serve-hostname

(jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$ ls
nginx-clusterip-svc.yml nginx-deploy.yml nginx-lb-svc.yml nginx-nodeport-svc.yml nginx-route.yml nginx-rs.yml

(jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$ oc get all
No resources found in jegan namespace.

(jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$ oc create -f nginx-rs.yml
replicaset.apps/nginx-rs created

(jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$ oc get deploy,rs,po
NAME           DESIRED   CURRENT   READY   AGE
replicaset.apps/nginx-rs   2         2         2       6s

(jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$ oc get all
No resources found in jegan namespace.

(jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$ oc create -f nginx-rs.yml
replicaset.apps/nginx-rs created

(jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$ oc get deploy,rs,po
NAME           DESIRED   CURRENT   READY   AGE
replicaset.apps/nginx-rs   2         2         2       6s
NAME          READY   STATUS    RESTARTS   AGE
pod/nginx-rs-gsj4j  1/1     Running   0          6s
pod/nginx-rs-qmp5n  1/1     Running   0          6s

(jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$ oc delete pod/pod/nginx-rs-gsj4j
error: arguments in resource/name form may not have more than one slash

(jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$ oc delete pod/nginx-rs-gsj4j
pod "nginx-rs-gsj4j" deleted

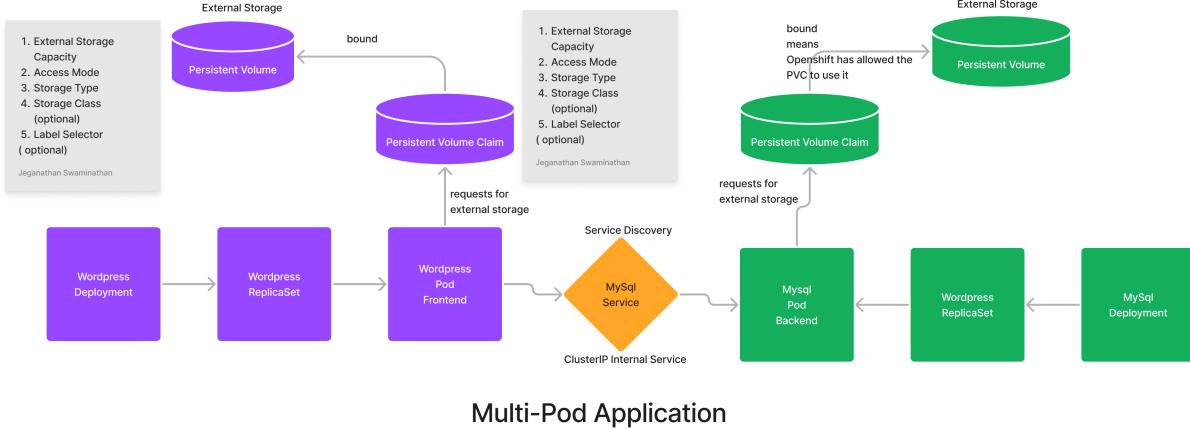
(jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$ oc get po -w
NAME           READY   STATUS    RESTARTS   AGE
nginx-rs-94hx7 1/1     Running   0          4s
nginx-rs-qmp5n  1/1     Running   0          38s
^C

(jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$ oc delete rs/nginx-rs
replicaset.apps "nginx-rs" deleted

(jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
$ oc get all
No resources found in jegan namespace.

(jegan@tektutor.org)-[~/openshift-sep-2023/Day3/declarative-manifests]
```

Lab - Deploying a Wordpress and MySQL multi-pod application that uses external storage



First, let's deploy mysql deployment with all its dependent resources

```
cd ~/openshift-sep-2023
git pull
cd Day3/declarative-manifests

oc apply -f wordpress-secrets.yml
oc apply -f wordpress-cm.yml

oc apply -f mysql-pv.yml
oc apply -f mysql-pvc.yml
oc apply -f mysql-deploy.yml
oc apply -f mysql-svc.yml
```

Expected output

```
jegan@tektutor:~/openshift-sep-2023/Day3/wordpress-configmap-and-secrets$ ls -l
total 44
-rw-r--r-- 1 jegan jegan 836 Sep 13 15:54 mysql-deploy.yml
-rw-r--r-- 1 jegan jegan 302 Sep 13 15:54 mysql-pvc.yml
-rw-r--r-- 1 jegan jegan 396 Sep 13 16:13 mysql-pv.yml
-rw-r--r-- 1 jegan jegan 175 Sep 13 15:54 mysql-svc.yml
-rw-r--r-- 1 jegan jegan 119 Sep 13 15:54 wordpress-cm.yml
-rw-r--r-- 1 jegan jegan 1194 Sep 13 15:54 wordpress-deploy.yml
-rw-r--r-- 1 jegan jegan 306 Sep 13 15:54 wordpress-pvc.yml
-rw-r--r-- 1 jegan jegan 482 Sep 13 16:07 wordpress-pv.yml
-rw-r--r-- 1 jegan jegan 231 Sep 13 15:54 wordpress-route.yml
-rw-r--r-- 1 jegan jegan 146 Sep 13 15:54 wordpress-secrets.yml
-rw-r--r-- 1 jegan jegan 186 Sep 13 15:54 wordpress-svc.yml

(jegan@tektutor.org)-[~/openshift-sep-2023/Day3/wordpress-configmap-and-secrets]$ oc apply -f mysql-pv.yml
persistentvolume/mysql-pv-jegan created

(jegan@tektutor.org)-[~/openshift-sep-2023/Day3/wordpress-configmap-and-secrets]$ oc apply -f mysql-pvc.yml
persistentvolumeclaim/mysql-pvc-jegan created

(jegan@tektutor.org)-[~/openshift-sep-2023/Day3/wordpress-configmap-and-secrets]$ oc apply -f mysql-deploy.yml
deployment.apps/mysql created

(jegan@tektutor.org)-[~/openshift-sep-2023/Day3/wordpress-configmap-and-secrets]$ oc apply -f mysql-svc.yml
service/mysql created

(jegan@tektutor.org)-[~/openshift-sep-2023/Day3/wordpress-configmap-and-secrets]$ oc apply -f wordpress-cm.yml
configmap/wordpress-cm created

(jegan@tektutor.org)-[~/openshift-sep-2023/Day3/wordpress-configmap-and-secrets]$ oc apply -f wordpress-secrets.yml
secret/wordpress-secrets created

(jegan@tektutor.org)-[~/openshift-sep-2023/Day3/wordpress-configmap-and-secrets]$ cat wordpress-secrets.yml
```

```
jegan@tektutor: ~ | jegan@tektutor: ~/ansible-sep-2023 | jegan@tektutor: ~/openshift-sep-2023 | jegan@tektutor: ~/openshift-sep-202... | jegan@tektutor: ~/openshift-sep-202... | jegan@tektutor: ~
$ oc apply -f mysql-pv.yml
persistentvolume/mysql-pv-jegan created

(jegan@tektutor.org)-[~/openshift-sep-2023/Day3/wordpress-configmap-and-secrets]
$ oc apply -f mysql-pvc.yml
persistentvolumeclaim/mysql-pvc-jegan created

(jegan@tektutor.org)-[~/openshift-sep-2023/Day3/wordpress-configmap-and-secrets]
$ oc apply -f mysql-deploy.yml
deployment.apps/mysql created

(jegan@tektutor.org)-[~/openshift-sep-2023/Day3/wordpress-configmap-and-secrets]
$ oc apply -f mysql-svc.yml
service/mysql created

(jegan@tektutor.org)-[~/openshift-sep-2023/Day3/wordpress-configmap-and-secrets]
$ oc apply -f wordpress-cm.yml
configmap/wordpress-cm created

(jegan@tektutor.org)-[~/openshift-sep-2023/Day3/wordpress-configmap-and-secrets]
$ oc apply -f wordpress-secrets.yml
secret/wordpress-secrets created

(jegan@tektutor.org)-[~/openshift-sep-2023/Day3/wordpress-configmap-and-secrets]
$ cat wordpress-secrets.yml
apiVersion: v1
kind: Secret
metadata:
  labels:
    app: wordpress
  name: wordpress-secrets
data:
  username: cm9vdA==
  password: cm9vdEAxMjM=

(jegan@tektutor.org)-[~/openshift-sep-2023/Day3/wordpress-configmap-and-secrets]
$ echo -n cm9vdEAxMjM= | base64 -d
root@123

(jegan@tektutor.org)-[~/openshift-sep-2023/Day3/wordpress-configmap-and-secrets]
$
```

From the OpenShift webconsole

The screenshot shows the Red Hat OpenShift webconsole interface. The top navigation bar includes links for 'Meeting is in progress...', 'Digital Learning Platform', 'Topology - Red Hat OpenShift', 'openshift-sep-2023/Day3', 'WordPress - FigJam', and 'openshift-aug-2023/Day3'. The user is logged in as 'kube:admin'. The left sidebar menu has 'Developer' selected and lists 'Topology', 'Observe', 'Search', 'Builds', 'Helm', 'Project', 'ConfigMaps', and 'Secrets'. The main content area displays the MySQL application details. At the top, it says 'You are logged in as a temporary administrative user. Update the cluster OAuth configuration to allow others to log in.' Below this is a search bar with 'Find by name' and a 'mysql' button. To the right, there's a large circular icon with a red and white design. The 'mysql' application card shows a 'Health checks' section with a note about missing health checks. The 'Resources' tab is selected, showing a 'Pods' section with one pod named 'mysql-886b485d4-v2tw6' in 'Running' status, and a 'Services' section with a service port 'TCP/3306 → Pod port: 3306'. The 'Details' and 'Observe' tabs are also present.

You are logged in as a temporary administrative user. Update the [cluster OAuth configuration](#) to allow others to log in.

Project: jegan

Pods > Pod details

P mysql-886b485d4-v2tw6 Running

Actions ▾

Details Metrics YAML Environment Logs Events Terminal

Pod details

Name	mysql-886b485d4-v2tw6	Status	Running
Namespace	NS jegan	Restart policy	Always restart
Labels	app=mysql, pod-template-hash=686b485d4	Active deadline seconds	Not configured
Node selector	No selector	Pod IP	10.128.11.9
Tolerations	2 tolerations	Host IP	192.168.122.196
Annotations	3 annotations	Node	master-3.ocptektor-ocp-labs
Created at	2023-09-13 10:47:44 UTC	Image pull secret	default-dockercfg-xglnz

Logs

Log streaming... **mysql** Current log **Search**

85 lines

```

1 mariadb 10:47:42.17
2 mariadb 10:47:42.17 Welcome to the Bitnami mariadb container
3 mariadb 10:47:42.17 Subscribe to project updates by watching https://github.com/bitnami/containers
4 mariadb 10:47:42.17 Submit issues and feature requests at https://github.com/bitnami/containers/issues
5 mariadb 10:47:42.17
6 mariadb 10:47:42.18 INFO => ** Starting MariaDB setup **
7 mariadb 10:47:42.19 INFO => Validating settings in MYSQL_*/MARIADB_* env vars
8 mariadb 10:47:42.20 INFO => Initializing mariadb database
9 mariadb 10:47:42.21 INFO => Updating "my.cnf" with custom configuration
10 mariadb 10:47:42.22 INFO => Setting slow_query_log option
11 mariadb 10:47:42.22 INFO => Setting long_query_time option
12 mariadb 10:47:42.23 INFO => Installing database
13 /opt/bitnami/mariadb/bin/mysql: Deprecate program name. It will be removed in a future release, use '/opt/bitnami/mariadb/bin/mariadb' instead
14 mariadb 10:47:42.28 INFO => Starting mariadb in background
15 2023-09-13 10:47:44 0 [Note] Starting MariaDB 11.0.3-MariaDB source revision 70905bcb9059dcc40db3b73bc46a36c7d40fie10 as process 107
16 2023-09-13 10:47:44 0 [Note] InnoDB: Compressed tables use zlib 1.2.11
17 2023-09-13 10:47:44 0 [Note] InnoDB: Number of transaction pools: 1
18 2023-09-13 10:47:44 0 [Note] InnoDB: Using crc32 + pclmulqdq instructions

```

Actions ▾

The screenshot shows the Red Hat OpenShift web interface. The left sidebar has a 'Developer' dropdown set to 'Project: jegan'. Under 'Logs', the 'mysql' tab is selected. The main area displays the log output for the pod 'mysql-886b485d4-v2tw6'. The logs show MySQL startup messages, including the creation of the 'ibtmp1' temporary tablespace and the loading of buffer pool(s). A search bar at the top right allows filtering by line number (85 lines) or content. Action buttons for 'Wrap lines', 'Raw', 'Download', and 'Expand' are also present.

```
66 2023-09-13 10:47:47 0 [Note] mysqld: Q_TMPFILE is not supported on /opt/bitnami/mariadb/tmp (disabling future attempts)
67 2023-09-13 10:47:47 0 [Note] InnoDB: Using Linux native AIO
68 2023-09-13 10:47:47 0 [Note] InnoDB: Initializing buffer pool, total size = 128.000MiB, chunk size = 2.000MiB
69 2023-09-13 10:47:47 0 [Note] InnoDB: Completed initialization of buffer pool
70 2023-09-13 10:47:47 0 [Note] InnoDB: Buffered log writes (block size=512 bytes)
71 2023-09-13 10:47:47 0 [Note] InnoDB: End of log at LSN=47263
72 2023-09-13 10:47:47 0 [Note] InnoDB: Opened 3 undo tablespaces
73 2023-09-13 10:47:47 0 [Note] InnoDB: 128 rollback segments in 3 undo tablespaces are active.
74 2023-09-13 10:47:47 0 [Note] InnoDB: Setting file './ibtmp1' size to 12.000MB. Physically writing the file full; Please wait ...
75 2023-09-13 10:47:47 0 [Note] InnoDB: File './ibtmp1' size is now 12.000MB.
76 2023-09-13 10:47:47 0 [Note] InnoDB: log sequence number 47263; transaction id 14
77 2023-09-13 10:47:47 0 [Note] Plugin 'FEEDBACK' is disabled.
78 2023-09-13 10:47:47 0 [Note] Plugin 'wsrep-provider' is disabled.
79 2023-09-13 10:47:47 0 [Note] InnoDB: Loading buffer pool(s) from /bitnami/mariadb/data/ib_buffer_pool
80 2023-09-13 10:47:47 0 [Note] InnoDB: Buffer pool(s) load completed at 230913 10:47:47
81 2023-09-13 10:47:47 0 [Note] Server socket created on IP: '0.0.0.0'.
82 2023-09-13 10:47:47 0 [Warning] 'proxies_priv' entry '% root@mysql-886b485d4-v2tw6' ignored in --skip-name-resolve mode.
83 2023-09-13 10:47:47 0 [Note] /opt/bitnami/mariadb/sbin/mysqld: ready for connections
84 Version: '11.0.3-MariaDB' socket: '/opt/bitnami/mariadb/tmp/mysql.sock' port: 3306 Source distribution
85 2023-09-13 10:48:55 3 [Warning] Access denied for user 'root'@'localhost' (using password: YES)
```