Deploying Stateless Application with Deployment Objects.

Kubectl - kubectl controls the Kubernetes cluster manager

You can run an application by creating a Kubernetes Deployment object, and you can describe a Deployment in a YAML file. For example, this YAML file describes a Deployment that runs the **<your-docker-hub-username>/ril:v1** Docker image

You have to replace this with the one you pushed to your **docker hub** account in **docker lab 9**.

1. Login to your AWS Instance and make a dir /home/<your-user-name>/application

$ cd /home/<your-username>
$ mkdir application
$ cd application/

$ vim <your-name>-deployment.yaml # paste the below text in the vim editor

#Note : press 'i' to start the edit mode in the vim editor.

```
apiVersion: apps/v1 # for versions before 1.9.0 use apps/v1beta2
kind: Deployment
metadata:
  name: <your-name>-deployment
spec:
  selector:
    matchLabels:
      app: <your-app-name>
  replicas: 2 # tells deployment to run 2 pods matching the template
  template:
    metadata:
      labels:
        app: <your-app-name>
    spec:
      containers:
      - name: <your-container-name>
        image: <docker-hub-image>/image:tag > #ex : asyed755/ril:v1 - it should be the same as on docker hub
        ports:
        - containerPort: 80
```

2. Create a Deployment based on the YAML file:

$ kubectl apply -f <your-name>-**deployment.yaml**

3. Display information about the Deployment:

$ kubectl describe deployment deployment

The output is similar to this:

user@computer:~/website$ kubectl describe deployment ril-deployment
Name: <your-name>-deployment
Namespace: default
CreationTimestamp: Tue, 30 Aug 2016 18:11:37 -0700
Labels: app=ril
Annotations: deployment.kubernetes.io/revision=1
Selector: app=ril
Replicas: 2 desired | 2 updated | 2 total | 2 available | 0 unavailable
StrategyType: RollingUpdate
MinReadySeconds: 0
RollingUpdateStrategy: 1 max unavailable, 1 max surge
Pod Template:
Labels: app=ril
Containers:
nginx:
Image: asyed755/ril:v1
Port: 80/TCP
Environment: <none>
Mounts: <none>
Volumes: <none>
Conditions:
Type Status Reason
---- ------ ------
Available True MinimumReplicasAvailable
Progressing True NewReplicaSetAvailable
OldReplicaSets: <none>
NewReplicaSet: nginx-deployment-1771418926 (2/2 replicas created)
No events.

4. List the pods created by the deployment:

$ kubectl get pods -l app=<your-app-name>
The output is similar to this:

```
NAME                                READY   STATUS    RESTARTS   AGE
ril-deployment-1471416983-7o5ac     1/1     Running   0          16h
ril-deployment-1541148254-318ad     1/1     Running   0          16h
```

5. To display information about a pod:

$ kubectl describe pod <pod-name>