**DaemonSet**

Create a daemonset with

name : my-ds

image: nginx

| **apiVersion**: apps/v1  **kind**: DaemonSet  **metadata**:  **name**: my-ds  **spec**:  **template**:  **metadata**:  **name**: pod-for-ds  **labels**:  **app**: ds  **spec**:  **containers**:  **- image**: nginx  **name**: nginx  **selector**:  **matchLabels**:  **app**: ds |
| --- |

**Service**

Expose your deployment with the following command:

kubectl expose deployment “Your-deployment-name” --name mysvc --port 80

Get the IP of the svc

kubectl get svc -o wide

Access the application running behind the svc

curl <ClusterIP> command

Create Nodeport svc

kubectl expose deployment mydep --name mysvc --port 80 --type NodePort

Create loadbalancer svc

kubectl expose deployment mydep --name mysvc-lb --port 80 --type LoadBalancer

**Storage:**

**Volumes**

| **apiVersion**: v1  **kind**: Pod  **metadata**:  **name**: random-number-generator  **spec**:  **containers** :  **- image**: alpine  **name**: alpine  **command**: ["/bin/sh","-c","shuf -i 0-100 -n 1 >> /opt/number.out && sleep 3600"]  **volumeMounts**:  **- mountPath**: /opt  **name**: mydata  **volumes**:  **- name**: mydata  **hostPath**:  **path**: /tmp  **type**: Directory |
| --- |

Check the random number generated

kubectl exec random-number-generator -- cat /opt/number.out

Delete the pod, and recreate it.

kubectl delete pod random-number-generator

Check the random number generated (you should see the previous number preserved)

kubectl exec random-number-generator -- cat /opt/number.out

**PV - PVC and Pods**

1. Create a PV with 1Gi storage

| **apiVersion**: v1  **kind**: PersistentVolume  **metadata**:  **name**: pv1  **spec**:  **capacity**:  **storage**: 1Gi  **accessModes**:  - ReadWriteOnce  **hostPath**:  **path**: /tmp/data |
| --- |

1. Create a PVC with 500Mi of storage

| **apiVersion**: v1  **kind**: PersistentVolumeClaim  **metadata**:  **name**: claim1  **spec**:  **accessModes**:  - ReadWriteOnce  **resources**:  **requests**:  **storage**: 500Mi |
| --- |

1. Check with PVC is bound or not

kubectl get pvc

1. Mount the storage in Pod

| **apiVersion**: v1  **kind**: Pod  **metadata**:  **name**: mypod  **spec**:  **containers**:  **- name**: myfrontend  **image**: nginx  **volumeMounts**:  **- mountPath**: "/tmp/html"  **name**: my-pvc  **volumes**:  **- name**: my-pvc  **persistentVolumeClaim**:  **claimName**: claim1 |
| --- |

Commands

| kubectl apply -f daemonset.yaml  kubectl get ds  kubectl get pods  kubectl get nodes  kubectl get pods -o wide  kubectl describe nodes ip-172-31-19-184 | grep Taint  kubectl taint node ip-172-31-19-184 node-role.kubernetes.io/master:NoSchedule-  kubectl get pods -o wide  kubectl taint node ip-172-31-19-184 node-role.kubernetes.io/master:NoExecute  kubectl get pods -o wide  kubectl get pods -A -o wide  kubectl describe pod kube-scheduler-ip-172-31-19-184 -n kube-system  kubectl get pods -A -o wide  kubectl describe pod kube-proxy-npr2h -n kube-system  kubectl taint node ip-172-31-19-184 node-role.kubernetes.io/master:NoExecute-  kubectl taint node ip-172-31-19-184 node-role.kubernetes.io/master:NoSchedule  cat daemonset.yaml  kubectl get pods -o wide  curl 192.168.189.89  kubectl delete pod mydep-6b669d478c-2gmkx  curl 192.168.189.89  kubectl get pods -o wide  curl 192.168.235.151  kubectl delete -f daemonset.yaml deployments.apps  kubectl delete -f daemonset.yaml -f deployments.apps  kubectl delete -f daemonset.yaml -f deployment.yaml  kubectl get pods -o wide  cat deployment.yaml  kubectl apply -f deployment.yaml  kubectl get pods -o wide  kubectl expose deployment mydep --name mysvc --port 80  kubectl get svc -o wide  curl 10.102.26.73  kubectl describe svc mysvc  kubectl scale deployment mydep --replicas=10  kubectl describe svc mysvc  kubectl expose deployment mydep --name mysvc --port 80 --dry-run -o yaml  curl 10.102.26.73  kubectl delete svc mysvc  kubectl expose deployment mydep --name mysvc --port 80 --type NodePort  kubectl get svc -o wide  curl localhost:31802  kubectl expose deployment mydep --name mysvc-lb --port 80 --type Loadbalancer  kubectl expose deployment mydep --name mysvc-lb --port 80 --type LoadBalancer  kubectl get svc -o wide  history  kubectl get pods  kubectl delete all --all  vim random-number-generator.yaml  kubectl apply random-number-generator.yaml  kubectl apply -f random-number-generator.yaml  vim random-number-generator.yaml  kubectl apply -f random-number-generator.yaml  vim random-number-generator.yaml  kubectl apply -f random-number-generator.yaml  vim random-number-generator.yaml  kubectl apply -f random-number-generator.yaml  vim random-number-generator.yaml  kubectl apply -f random-number-generator.yaml  kubectl get pods  kubectl exec random-number-generator -- bash  kubectl exec random-number-generator -- /bin/bash  kubectl exec random-number-generator -- cat /opt/number.out  kubectl delete pod random-number-generator  kubectl apply -f random-number-generator.yaml  kubectl exec random-number-generator -- cat /opt/number.out  kubectl delete pod random-number-generator  kubectl get pods  vim random-number-generator.yaml  kubectl apply -f random-number-generator.yaml  vim random-number-generator.yaml  kubectl apply -f random-number-generator.yaml  kubectl exec random-number-generator -- cat /opt/number.out  kubectl delete pod random-number-generator  kubectl apply -f random-number-generator.yaml  kubectl exec random-number-generator -- cat /opt/number.out  kubectl describe pod random-number-generator  cat random-number-generator.yaml  vim random-number-generator.yaml  vim pv.yaml  kubectl apply -f pv.yaml  kubectl get pv  vim pvc.yaml  kubectl apply pvc.yaml  kubectl apply -f pvc.yaml  vim pvc.yaml  kubectl apply -f pvc.yaml  kubectl get pvc  kubectl get pv  vim pod-with-pvc.yaml  kubectl apply -f pod-with-pvc.yaml  kubectl get pods  cp pvc.yaml pvc-5G.yaml  vim pvc-5G.yaml  kubectl apply -f pvc-5G.yaml  kubectl get pvc  kubectl get pv  cp pvc-5G.yaml pvc-700Mi.yaml  vim pvc-700Mi.yaml  kubectl apply -f pvc-700Mi.yaml  kubectl get pvc  ll  vim pv.yaml  kubectl get pv  kubectl describe pv pv1  history |
| --- |