**Environment Variables**

Create a pod with below configurations:

Image: gcr.io/google-samples/node-hello:1.0

Set below environment variables

DEMO\_GREETING = Hello Simplilearn

DEMO\_FAREWELL = Bye SImplilearn

Either use kubectl run or the yaml

kubectl run envvar-demo --image gcr.io/google-samples/node-hello:1.0 --env “DEMO\_GREETING=Hello Simplilearn” --env “DEMO\_FAREWELL=Bye SImplilearn”

Print environment variables

kubectl exec <podname> -- printenv

| **apiVersion**: v1  **kind**: Pod  **metadata**:  **name**: envar-demo  **labels**:  **purpose**: demonstrate-envars  **spec**:  **containers**:  **- name**: envar-demo-container  **image**: gcr.io/google-samples/node-hello:1.0  **env**:  **- name**: DEMO\_GREETING  **value**: "Hello Simplilearn"  **- name**: DEMO\_FAREWELL  **value**: "Bye Simplilearn" |
| --- |

**Secrets**

Create a secret

kubectl create secret generic --from-literal password=simplilearn mysql-secret

Use the secret in pod:

| **apiVersion**: v1  **kind**: Pod  **metadata**:  **name**: mysql-pod  **spec**:  **containers**:  **- name**: mysql  **image**: mysql  **env**:  **- name**: MYSQL\_ROOT\_PASSWORD  **valueFrom**:  **secretKeyRef**:  **name**: mysql-secret  **key**: password |
| --- |

Verify:

sudo apt update && sudo apt install mysql-client-core-8.0

mysql -h <Pod IP Address> --password=simplilearn --user=root

**ConfigMap**

Create a file called max\_packets.cnf with following content:

[mysqld]

max\_allowed\_packet = 100M

Create a configmap

kubectl create configmap --from-file=max\_packets.cnf mysql-config

Use the configMap in a pod

| **apiVersion**: v1  **kind**: Pod  **metadata**:  **name**: mysql-pod  **spec**:  **containers**:  **- name**: mysql  **image**: mysql  **volumeMounts**:  **- name**: config-file  **mountPath**: /etc/mysql/conf.d  **env**:  **- name**: MYSQL\_ROOT\_PASSWORD  **valueFrom**:  **secretKeyRef**:  **name**: mysql-secret  **key**: password  **volumes**:  **- name**: config-file  **configMap**:  **name**: mysql-config  **items**:  **- key**: max\_packets.cnf  **path**: max\_allowed\_packets.cnf |
| --- |

Verify

kubectl exec mysql-pod -- cat /etc/mysql/conf.d/max\_allowed\_packets.cnf

**Jobs:**

Print a hello statement using ubuntu image; job should be executed 10 times

Image: ubuntu

Command: /bin/sh echo Hello

Completions: 10

| **apiVersion**: batch/v1  **kind**: Job  **metadata**:  **name**: hello-job  **spec**:  **completions**: 10  **template**:  **spec:**  **containers**:  **- name**: ubuntu  **image**: ubuntu  **command**: ["echo" ,"Hello"]  **restartPolicy**: Never |
| --- |

kubectl get pods --watch

**CronJob**

| **apiVersion**: batch/v1beta1  **kind**: CronJob  **metadata**:  **name**: my-cron  **spec**:  **schedule**: "\* \* \* \* \*"  **jobTemplate**:  **spec**:  **template**:  **spec**:  **containers**:  **- name**: ubuntu  **image**: ubuntu  **command**: ["echo" ,"Hello"]  **restartPolicy**: Never |
| --- |

kubectl get pods --watch

Commands:

| kubectl version  kubectl get nodes  vim pod-mysql.yaml  kubectl apply -f pod-mysql.yaml  vim pod-mysql.yaml  kubectl apply -f pod-mysql.yaml  kubectl get pods  kubectl describe pod mysql-pod  kubectl logs mysql-pod  cat pod-mysql.yaml  vim pod-mysql.yaml  kubectl apply -f  kubectl apply -hj  kubectl apply -h  kubectl delete -f pod-mysql.yaml  kubectl apply -h  kubectl apply -f pod-mysql.yaml  kubectl get pods  kubectl get pods -o wide  kubectl run -h  cat pod-mysql.yaml  kubectl create secret generic --from-literal password=simplilearn --from-literal somedata=somevalue mysql-secret  kubectl get secrets mysql-secret  kubectl get secrets mysql-secret -o wide  kubectl describe secrets mysql-secret  kubectl get secrets mysql-secret -o yaml  base64 simplilearn  base64 -h  base64 --help  echo simplilearn | base64  echo c2ltcGxpbGVhcm4K | base64 --decode  kubectl get secrets mysql-secret -o yaml  vim pod-mysql.yaml  kubectl delete pod mysql-pod  kubectl apply -f pod-mysql.yaml  kubectl get pods  kubectl describe pod mysql-pod  cat pod-mysql.yaml  kubectl get pod -o wide  vim max\_packets.cnf  kubectl create configmap --from-file=max\_packets.cnf mysql-config  kubectl describe configmaps mysql-config  vim pod-mysql.yaml  kubectl describe configmaps mysql-config  vim pod-mysql.yaml  kubectl delete pod mysql-pod  kubectl apply -f pod-mysql.yaml  vim pod-mysql.yaml  kubectl apply -f pod-mysql.yaml  vim pod-mysql.yaml  kubectl apply -f pod-mysql.yaml  kubectl get pods  kubectl describe pod mysql-pod  kubectl exec mysql-pod -- ll /etc/mysql/conf.d/  kubectl exec mysql-pod -- ls -l /etc/mysql/conf.d/  kubectl exec mysql-pod -- cat /etc/mysql/conf.d/max\_allowed\_packets.cnf  cat max\_packets.cnf  cat pod-mysql.yaml  vim job.yaml  kubectl apply job.yaml  kubectl apply -f job.yaml  vim job.yaml  kubectl apply -f job.yaml  kubectl get pods  kubectl get pods --watch  kubect get job  kubectl get job  vim job.yaml  kubectl logs pi-l729f  vim job.yaml  kubectl delete job pi  kubectl apply -f job.yaml  kubectl get pods --watch  kubectl get pods  kubectl delete job pi  vim job.yaml  kubectl apply -f job.yaml && kubectl get pods --watch  vim job.yaml  cat job.yaml  vim myjob.yaml  kubectl apply -f myjob.yaml  kubectl get pods  kubectl get pods --watch  kubectl describe pod hello-job-k297r  kubectl logs hello-job-k297r  kubectl delete job hello-job  vim myjob.yaml  kubectl apply -f myjob.yaml  kubectl get pods  kubectl logs hello-job-2fdzh  vim cronjob.yaml  kubectl apply -f cronjob.yaml  kubectl explain cronjob  vim cronjob.yaml  kubectl apply -f cronjob.yaml  kubectl get pods  kubectl get cronjobs.batch  kubectl get pods -w  kubectl get pods  cat cronjob.yaml  kubectl get pods -A  kubectl delete jobs.batch hello-job pi  kubectl delete cronjobs.batch my-cron  kubectl get pods -A  kubectl get -n kube-system pod kube-controller-manager-ip-172-31-19-184  kubectl get -n kube-system pod kube-controller-manager-ip-172-31-19-184 -o yaml  vim /etc/kubernetes/controller-manager.conf  netstat  apt install net-tools  netstat -ntlp  ll /etc/cni/  ll /etc/cni/net.d/  vim /etc/cni/net.d/10-calico.conflist  vim /etc/cni/net.d/calico-kubeconfig  kubectl get -n kube-system pod  kubectl get -n kube-system pod calico-kube-controllers-5f6cfd688c-mdk7k -o yaml  kubectl get pods -o wide  kubectl apply -f myjob.yaml  kubetl get pods -o wide  kubectl get pods -o wide  kubectl get -n kube-system pod  kubectl create deployment --image nginx  kubectl create deployment mydep --image nginx --replicas 5  kubectl expose deployment mydep --port 80  kubectl get svc  kubectl run --image ubuntu -- sleep 3600  kubectl get pods  kubectl describe pod sleep  kubectl run --image ubuntu -h  kubectl run --image ubuntu --command -- sleep 3600  kubectl delete pod sleep  kubectl run --image ubuntu --command -- sleep 3600  kubectl delete jobs.batch hello-job  kubectl get pods  kubectl describe pod sleep  kubectl run --image ubuntu -h  kubectl delete pod sleep  kubectl run testpod --image ubuntu --command -- sleep 3600  kubectl get pod  kubectl exec -it testpod  kubectl exec -it testpod bash  kubectl get svc  netstat -ntlp  kubectl get pods -A  kubectl get pods -n kube-system coredns-74ff55c5b-9d4dq -o yaml  vim /etc/coredns/Corefile  kubectl get svc -A |
| --- |