

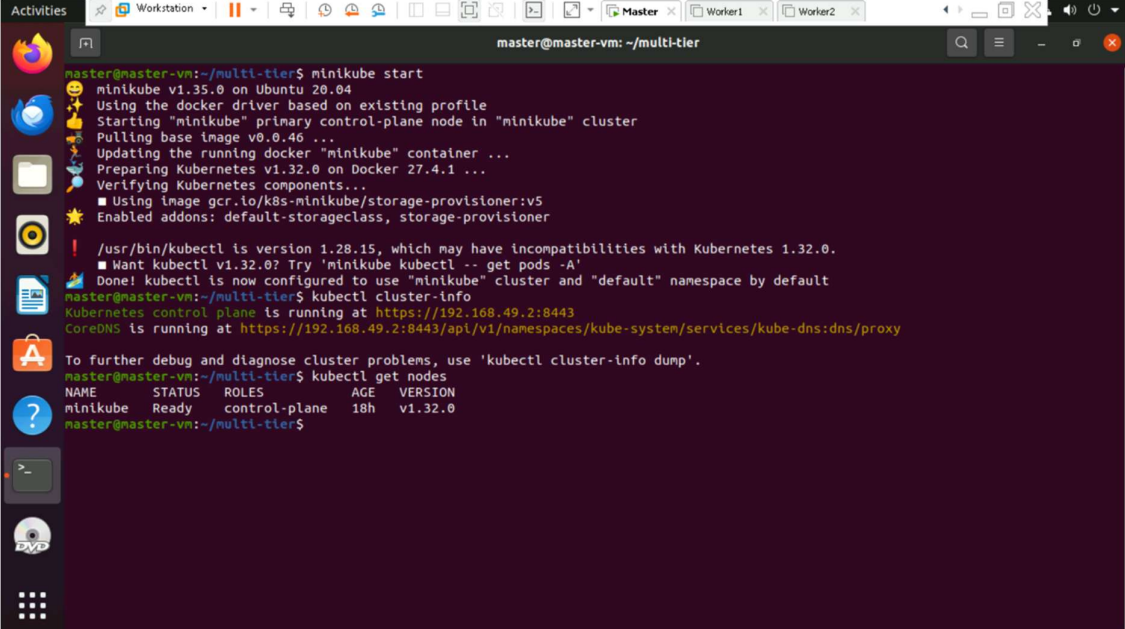
PROJECT 3: DEPLOY A MULTI-TIER WEB APPLICATION ON KUBERNETES

1. Setup Kubernetes Cluster

minikube start

kubectl cluster-info

kubectl get nodes

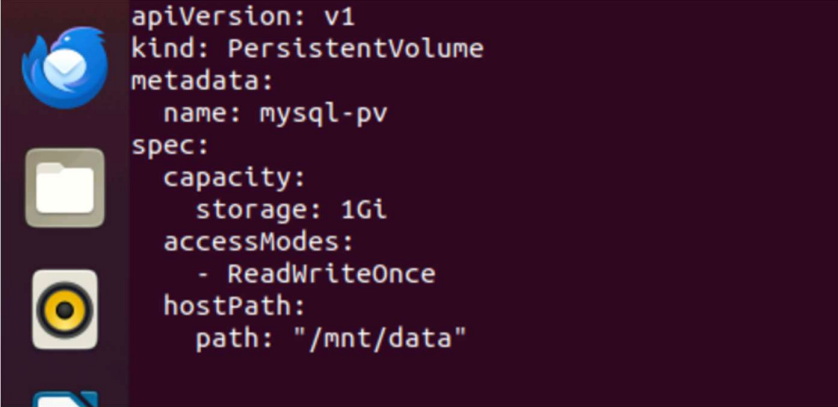


```
master@master-vm: ~/multi-tier
master@master-vm:~/multi-tier$ minikube start
minikube v1.35.0 on Ubuntu 20.04
Using the docker driver based on existing profile
Starting "minikube" primary control-plane node in "minikube" cluster
Pulling base image v0.0.46 ...
Updating the running docker "minikube" container ...
Preparing Kubernetes v1.32.0 on Docker 27.4.1 ...
Verifying Kubernetes components...
  ■ Using image gcr.io/k8s-minikube/storage-provisioner:v5
  ■ Enabled addons: default-storageclass, storage-provisioner
! /usr/bin/kubectl is version 1.28.15, which may have incompatibilities with Kubernetes 1.32.0.
  ■ Want kubectl v1.32.0? Try 'minikube kubectl -- get pods -A'
Done! kubectl is now configured to use "minikube" cluster and "default" namespace by default
master@master-vm:~/multi-tier$ kubectl cluster-info
Kubernetes control plane is running at https://192.168.49.2:8443
CoreDNS is running at https://192.168.49.2:8443/api/v1/namespaces/kube-system/services/kube-dns:dns/proxy

To further debug and diagnose cluster problems, use 'kubectl cluster-info dump'.
master@master-vm:~/multi-tier$ kubectl get nodes
NAME        STATUS   ROLES    AGE   VERSION
minikube    Ready    control-plane  18h   v1.32.0
master@master-vm:~/multi-tier$
```

2. Deploy MySQL Database (StatefulSet)

Step 1: Create a Persistent Volume for MySQL

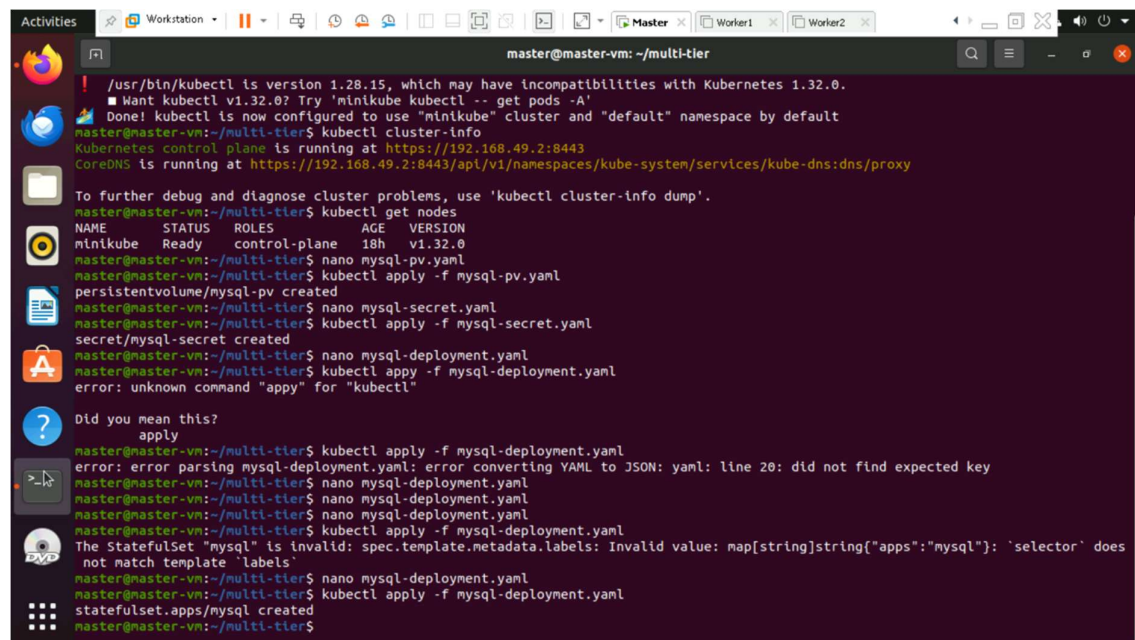


```
apiVersion: v1
kind: PersistentVolume
metadata:
  name: mysql-pv
spec:
  capacity:
    storage: 1Gi
  accessModes:
    - ReadWriteOnce
  hostPath:
    path: "/mnt/data"
```

Step 2: Create MySQL Secret for Password Storage

```
apiVersion: v1
kind: Secret
metadata:
  name: mysql-secret
type: Opaque
data:
  mysql-root-password: cGFzc3dvcmQ=
```

Step 3: Deploy MySQL as a StatefulSet



```
master@master-vm: ~/multi-tier
! /usr/bin/kubectl is version 1.28.15, which may have incompatibilities with Kubernetes 1.32.0.
  ■ Want kubectl v1.32.0? Try 'minikube kubectl -- get pods -A'
  Done! kubectl is now configured to use "minikube" cluster and "default" namespace by default
master@master-vm:~/multi-tier$ kubectl cluster-info
Kubernetes control plane is running at https://192.168.49.2:8443
CoreDNS is running at https://192.168.49.2:8443/api/v1/namespaces/kube-system/services/kube-dns:dns/proxy

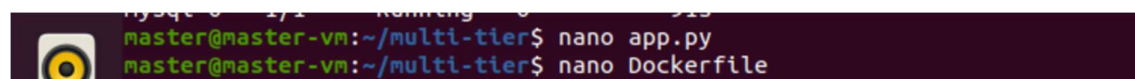
To further debug and diagnose cluster problems, use 'kubectl cluster-info dump'.
master@master-vm:~/multi-tier$ kubectl get nodes
NAME       STATUS   ROLES    AGE   VERSION
minikube   Ready    control-plane  18h   v1.32.0
master@master-vm:~/multi-tier$ nano mysql-pv.yaml
master@master-vm:~/multi-tier$ kubectl apply -f mysql-pv.yaml
persistentvolume/mysql-pv created
master@master-vm:~/multi-tier$ nano mysql-secret.yaml
master@master-vm:~/multi-tier$ kubectl apply -f mysql-secret.yaml
secret/mysql-secret created
master@master-vm:~/multi-tier$ nano mysql-deployment.yaml
master@master-vm:~/multi-tier$ kubectl apply -f mysql-deployment.yaml
error: unknown command "apply" for "kubectl"

Did you mean this?
  apply
master@master-vm:~/multi-tier$ kubectl apply -f mysql-deployment.yaml
error: error parsing mysql-deployment.yaml: error converting YAML to JSON: yaml: line 20: did not find expected key
master@master-vm:~/multi-tier$ nano mysql-deployment.yaml
master@master-vm:~/multi-tier$ nano mysql-deployment.yaml
master@master-vm:~/multi-tier$ kubectl apply -f mysql-deployment.yaml
The StatefulSet "mysql" is invalid: spec.template.metadata.labels: Invalid value: map[string]string["apps":"mysql"]: 'selector' does not match template 'labels'
master@master-vm:~/multi-tier$ nano mysql-deployment.yaml
master@master-vm:~/multi-tier$ kubectl apply -f mysql-deployment.yaml
statefulset.apps/mysql created
master@master-vm:~/multi-tier$
```

3. Deploy Flask Backend

Step 1: Create a Flask App (Dockerized)

Step 2: Create a Dockerfile



```
master@master-vm:~/multi-tier$ nano app.py
master@master-vm:~/multi-tier$ nano Dockerfile
```

Build & push Docker image:

