**Task1**

1. gcloud container clusters create CLUSTER\_NAME --num-nodes=2 --zone=ZONE
2. **gcloud container clusters create onlineboutique-cluster-674 --num-nodes=2 --machine-type=e2-standard-2 --release-channel=rapid --zone=us-west1-b**
3. same
4. kubectl create namespace team-a && \

kubectl create namespace team-b

**kubectl create namespace dev && \**

**kubectl create namespace prod**

1. Run this :

**git clone https://github.com/GoogleCloudPlatform/microservices-demo.git &&**

**cd microservices-demo && kubectl apply -f ./release/kubernetes-manifests.yaml --namespace dev**

**Task 2**

1. Create a new node pool with a larger machine type:

**gcloud container node-pools create** optimized-pool-3381**\**

**--cluster=hello-demo-cluster \**

**--machine-type=custom-2-3584 \**

**--num-nodes=2 \**

**--zone=us-west1-b**

1. Same
2. same
3. Cordon and drain
4. Cordon
5. for node in $(kubectl get nodes -l cloud.google.com/gke-nodepool=default-pool -o=name); do  
    kubectl cordon "$node";  
   done
6. Drain

**for node in $(kubectl get nodes -l cloud.google.com/gke-nodepool=**default-pool **-o=name); do**

**kubectl drain --force --ignore-daemonsets --delete-local-data --grace-period=10 "$node";**

**done**

1. **Check**

**kubectl get pods -o=wide**

1. **Delete**

**gcloud container node-pools delete default-pool --cluster=onlineboutique-cluster-674 --zone=us-west1-b**

**Task 3**

1. **Create disruption budget**

**kubectl create poddisruptionbudget onlineboutique-frontend-pdb --selector run=frontend --min-available 1**

1. kubectl get poddisruptionbudgets
2. Get the frontend deployment’s yaml manifest file.

In the Cloud console, from Kubernetes Engine > Workloads > click on **frontend**. Select the **YAML** tab. Click on **EDIT** on the horizontal menu at the top to edit

1. In the editor, under .spec.containers, replace **image: gcr.io/google-samples/microservices-demo/frontend:v0.8.1** to **image: gcr.io/qwiklabs-resources/onlineboutique-frontend:v2.1**  
   One line down, replace **imagePullPolicy: IfNotPresent** to **imagePullPolicy: Always**  
   In the Cloud console, click **SAVE** after editing.
2. Verify :from Cluster > Nodes > frontend services.

**Task 4. Autoscale from estimated traffic**

1. **kubectl autoscale deployment/frontend --cpu-percent=50 --min=1 --max=10**
2. gcloud container clusters update [Cluster Name] --enable-autoscaling --min-nodes=1 --max-nodes=6

**exam**

**europe-west1-d**

**onlineboutique-cluster-858**

**gcloud container clusters create onlineboutique-cluster-858 --num-nodes=2 --machine-type=e2-standard-2 --release-channel=rapid --zone=europe-west1-d**

**kubectl create namespace dev && \**

**kubectl create namespace prod**

**git clone https://github.com/GoogleCloudPlatform/microservices-demo.git &&**

**cd microservices-demo && kubectl apply -f ./release/kubernetes-manifests.yaml --namespace dev**

**gcloud container node-pools create optimized-pool-9397\**

**--cluster=onlineboutique-cluster-858 \**

**--machine-type=custom-2-3584 \**

**--num-nodes=2 \**

**--zone=europe-west1-d**

**for node in $(kubectl get nodes -l cloud.google.com/gke-nodepool=default-pool -o=name); do**

**kubectl cordon "$node";**

**done**

**for node in $(kubectl get nodes -l cloud.google.com/gke-nodepool=default-pool -o=name); do**

**kubectl drain --force --ignore-daemonsets --delete-local-data --grace-period=10 "$node";**

**done**

**kubectl get pods -o=wide**

**kubectl create poddisruptionbudget onlineboutique-frontend-pdb \**

**--selector app=frontend --min-available 1 --namespace dev**

**KUBE\_EDITOR="nano" kubectl edit deployment/frontend --namespace dev**

**gcr.io/qwiklabs-resources/onlineboutique-frontend:v2.1**

**kubectl autoscale deployment frontend --cpu-percent=50 --min=1 --max=10 --namespace dev**

**gcloud container clusters update onlineboutique-cluster-858 --enable-autoscaling --min-nodes=1 --max-nodes=6 --zone=europe-west1-d**

**extras**

**kubectl exec $(kubectl get pod --namespace=dev | grep 'loadgenerator' | cut -f1 -d ' ') -it --namespace=dev -- bash -c 'export USERS=8000; locust --host="http://YOUR\_FRONTEND\_EXTERNAL\_IP" --headless -u "8000" 2>&1'**