

**Kubernetes Engine Course**  
**Kubernetes Engine and gcloud Command List**  
**Updated Jan 2020**

**CLI /SDK Project Commands**

**-List Projects**

gcloud projects list

**-Set Your Default Project In GCP**

gcloud config set project mykubeproject

**-Set Your Default Region**

gcloud config set compute/region "us-east1"

**-List Compute Regions**

gcloud compute zones list

**-Describe List Compute Zones**

gcloud compute zones list

***-Print List All the URI in a zone***

```
gcloud compute zones list --uri
```

### **Describe a Project**

```
gcloud compute project-info describe --project mykubeproject
```

## **Setup IDE Environment**

**This will let you authenticate with Google Cloud SDK which obtains access credentials via a web-based authorization flow and sets the configuration.**

```
gcloud auth login
```

After downloading your version we need to initialize the IDE environment and connect to GCP.

**Gcloud init will initialize and configure your SDK with GCP. Multiple account can be managed as well.**

### **#“gcloud init”**

Then pick a configuration (Project) and follow prompts. Set account and project, region and zone

Install Emulators.. > Available for

- Bigtable
- Datastore
- Firestore
- Pub/Sub

***Use the follow commands to install emulator for Pub/Sub***

`gcloud components install pubsub-emulator`

`gcloud components update`

To start the Pub/Sub emulator

`gcloud beta emulators pubsub start`

```
C:\Users\HPE Workstation\AppData\Local\Google\Cloud SDK>gcloud components install pubsub-emulator
All components are up to date.

C:\Users\HPE Workstation\AppData\Local\Google\Cloud SDK>gcloud components update
All components are up to date.

C:\Users\HPE Workstation\AppData\Local\Google\Cloud SDK>gcloud beta emulators pubsub start
Executing: cmd /c C:\Users\HPE Workstation\AppData\Local\Google\Cloud SDK\google-cloud-sdk\platform\pubsub
emulator\bin\cloud-pubsub-emulator.bat --host=localhost --port=8085
[pubsub] This is the Google Pub/Sub fake.
[pubsub] Implementation may be incomplete or differ from the real system.
[pubsub] Jun 06, 2019 6:57:36 PM com.google.cloud.pubsub.testing.v1.Main main
[pubsub] INFO: IAM integration is disabled. IAM policy methods and ACL checks are not supported
[pubsub] Jun 06, 2019 6:57:37 PM io.gapi.emulators.netty.NettyUtil applyJava7LongHostnameWorkaround
[pubsub] INFO: Unable to apply Java 7 long hostname workaround.
[pubsub] Jun 06, 2019 6:57:37 PM com.google.cloud.pubsub.testing.v1.Main main
[pubsub] INFO: Server started, listening on 8085
```

## **Install Kubectl and Minikube**

**Install kubectl for SDK use.**

`gcloud components install kubectl`

**Install minikube cli so you can access from SDK**

`gcloud components install minikube`

## **Snapshots**

**Get list of snapshots in your current project**

`gcloud compute snapshots list`

Snapshot from an existing disk

gcloud compute disks snapshot

## Roles

gcloud iam roles copy

## Deployment Manager

gcloud deployment-manager deployments create example-deployment --config configuration-file.yaml \ --preview

## Kubernetes Engine

**Container Commands for GCP Cloud Developer Exam**

**You may want to practice these... Very important to understand how to increase cluster size and enable autoscaling for the cluster**

**Be sure to Create a Kubernetes Cluster and configure it to host an application**

**Understand how to make the cluster auto repairable and upgradable. Hint – Node auto-upgrades and auto-repairing feature**

**-Setup**

export PROJECT\_ID="\$(gcloud config get-value project -q)"

```
docker build -t gcr.io/$PROJECT_ID/hello-app:v1 .docker images
```

## **- Gcloud Container Commands**

### **Remember to set project**

```
gcloud config set project
```

### **Remember to set zone or region**

```
gcloud config set compute/zone us-central1-b
```

### **Kube Login credentials**

```
gcloud auth application-default login
```

### **Create container cluster with three nodes in US Central**

```
gcloud container clusters create hello-cluster --num-nodes=3 --zone us--central-b
```

### **Obtain credentials from cluster.**

```
gcloud container clusters get-credentials mykubeccluster
```

### **View context**

```
kubectl config current-context
```

### **List Clusters**

```
gcloud container clusters list
```

### **Describe cluster**

```
gcloud container clusters describe cluster-name
```

### **Resize cluster to 4 nodes.**

```
gcloud container clusters resize mygkecluster --num-nodes --size 4
```

### **- *Kubectl Commands***

kubectl run hello-web --image=gcr.io/\${PROJECT\_ID}/hello-app:v1 --port 8080

kubectl get pods

kubectl get nodes

kubectl expose deployment hello-web --type=LoadBalancer --port 8080

kubectl get services

kubectl scale deployment hello-web --replicas=3 Add (Expand)

kubectl apply -f myapp.yaml

kubectl logs -f myapp-pod

Delete Changes to a Yaml file

kubectl delete -f myapp.yaml

Apply Changes to a Yaml file

kubectl apply -f myapp.yaml

## **GSUTIL - Managing buckets and objects in Cloud Storage**

Create nearline Storage Bucket named mybucket

```
gsutil mb -c nearline gs://devopsbucket
```

**List all storage buckets in project**

```
gsutil ls
```

**View bucket information**

```
gsutil ls -L -b gs://mybucket
```

## Cloud Build

**Run a Cloud Build Script**

```
gcloud builds submit --config helloworld.yaml
```

## Cloud Source Repository

**Create New Repository**

```
gcloud source repos create mynewrepository
```

**List available repositories**

```
gcloud source repos list
```

**Clone a Cloud Source Repositories**

```
gcloud source repos clone CloudDeveloper --project=cloud-developer-242517
```

**Describe a repository**

```
gcloud source repos describe
```

## Compute Engine Certificate Mapping

***Verify your certificate has been provisioned for your resources***

gcloud app domain-mappings list

***Map certificates and update***

gcloud app domain-mappings update DOMAIN --certificate-management='AUTOMATIC'

***Upload Certificates***

gcloud app ssl-certificates create --display-name CERT\_DISPLAY\_NAME --certificate CERT\_DIRECTORY\_PATH --private-key KEY\_DIRECTORY\_PATH

***List Create and Delete Compute Engine Certificates***

gcloud compute ssl-certificates

***Add Metadata***

gcloud compute project-info add-metadata --metadata <KEY>=<VALUE>

## **API**

**Enable API**

gcloud services enable pubsub.googleapis.com

**Disable API**

gcloud services disable pubsub.googleapis.com

**List Services**

gcloud services list

**Curl Command**

Gcurl

<https://serviceusage.googleapis.com/v1/projects/357084163378/services/pubsub.googleapis.com:enable>



## SSH Keys

### View Project wide SSH Keys

```
gcloud compute project-info describe
```

### View Compute Instance SSH Keys

```
gcloud compute instance describe gcpvm1
```

generate SSH Key Linux

```
ssh-keygen -t rsa -f ~/.ssh/[KEY_FILENAME] -C [USERNAME]
```

## Networking

### Create Custom VPC

```
gcloud compute --project=massive-dynamo-244818 networks create devops1 --  
description="This is a test vpc" --subnet-mode=custom
```

### Create Custom Subnets

```
gcloud compute --project=massive-dynamo-244818 networks subnets create NAME --  
network=devops1 --region=REGION --range=IP_RANGE
```

### Create Auto Mode VPC

```
gcloud compute --project=massive-dynamo-244818 networks create devops1 --  
description="This is a test vpc" --subnet-mode=auto
```

### Create Subnets Auto Mode and add FW Rule for ingress SSH

```
gcloud compute --project=massive-dynamo-244818 firewall-rules create devops1-allow-ssh --  
description="Allows TCP connections from any source to any instance on the network using port  
22." --direction=INGRESS --priority=65534 --network=devops1 --action=ALLOW --rules=tcp:22 --  
source-ranges=0.0.0.0/0
```

