Learn about Google Kubernetes Engine (GKE)

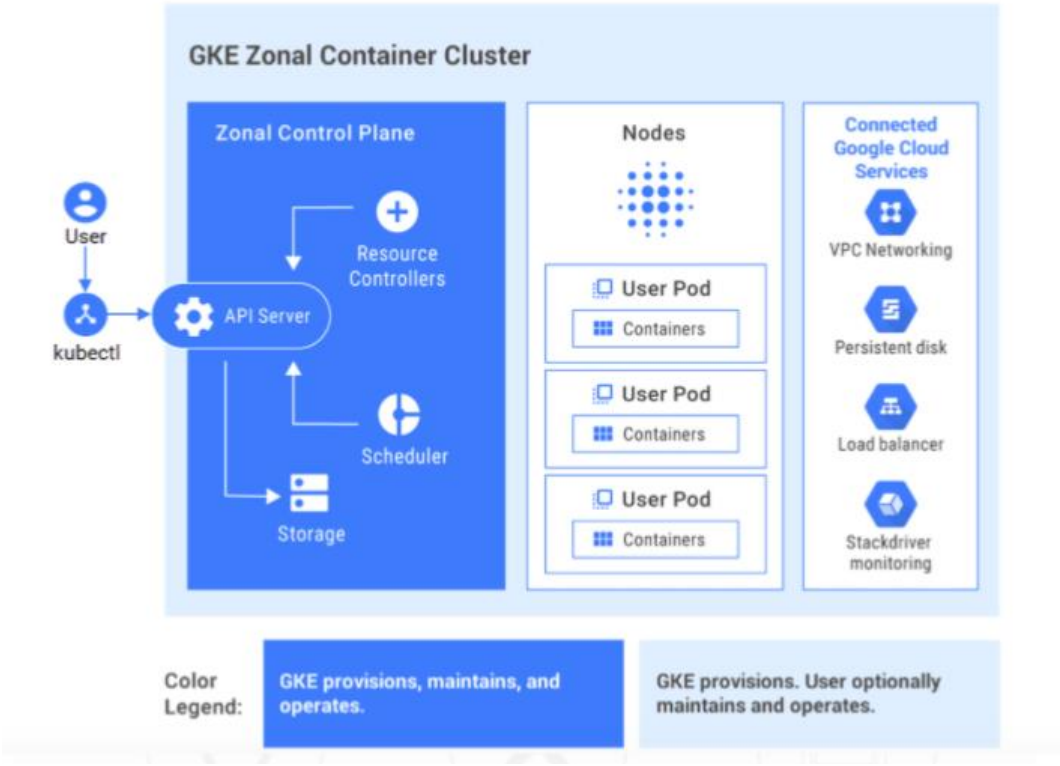
By

Stanley Stephen

Linkedin: https://www.linkedin.com/in/contactstanley/
Github: https://github.com/stanleymca/Learn_from_Stanley/
Email: s.stanley.mca@gmail.com

Google Kubernetes Engine (GKE)

Google Kubernetes Engine (also known as GKE) is a managed, production-ready environment for running Docker containers in the Google cloud.

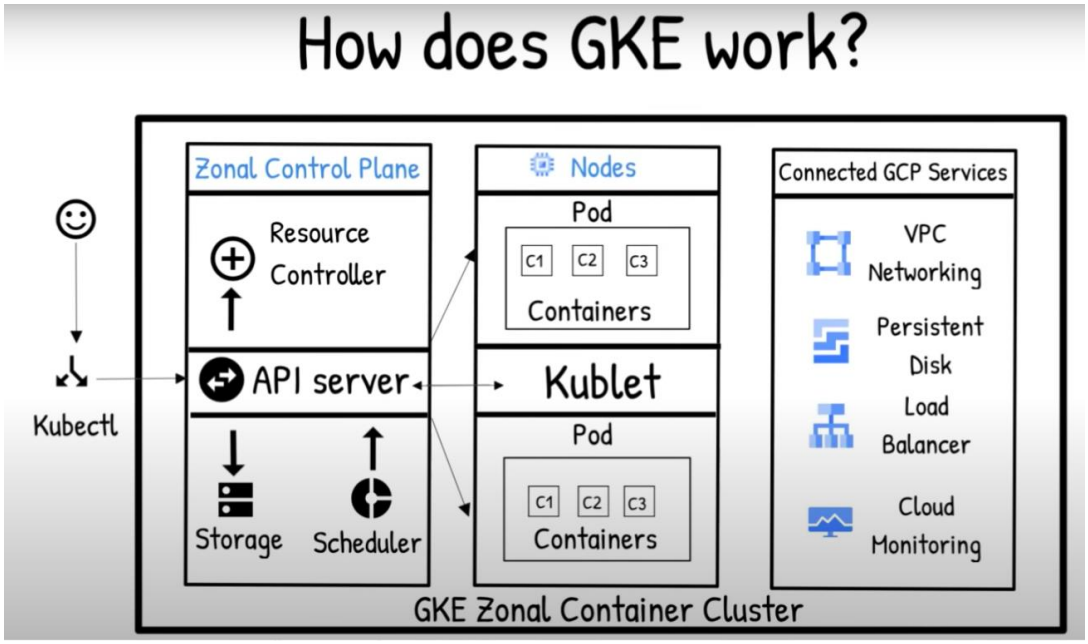


It permits you to form multiple-node clusters whereas conjointly providing access to any or all Kubernetes options.

Stanley Stephen, M.C.A.,
Linkedin: https://www.linkedin.com/in/contactstanley/
Github: https://github.com/stanleymca/Learn_from_Stanley/GKE_by_Stanley.pdf
Email: s.stanley.mca@gmail.com

How do GKE works?

GKE works with docker applications. These applications are dockerized into platform-independent, isolated user-space instances. Before you deploy any workloads on a GKE cluster, you need to always firstly dockerize the deployment.

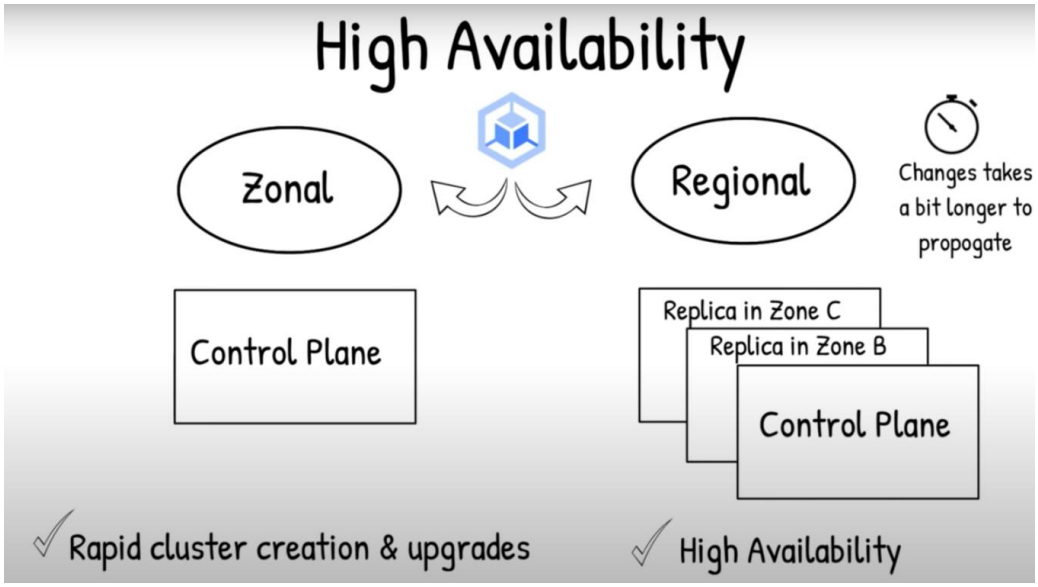


How to use GKE?

Stanley Stephen, M.C.A.,
Linkedin: https://www.linkedin.com/in/contactstanley/
Github: https://github.com/stanleymca/Learn_from_Stanley/GKE_by_Stanley.pdf
Email: s.stanley.mca@gmail.com

High Availability in GKE

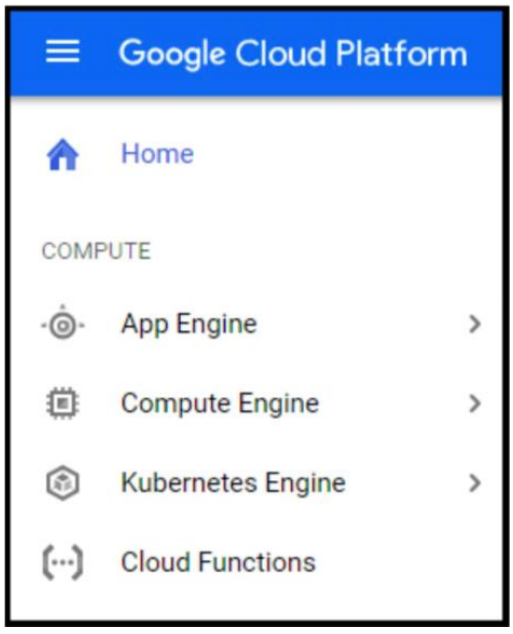
GKE offers two types of clusters: regional and zonal. … Regional clusters consist of a three Kubernetes control planes quorum, offering **higher availability** than a zonal cluster can provide for your cluster's control plane API.



Create A Kubernetes Cluster using the GCP console

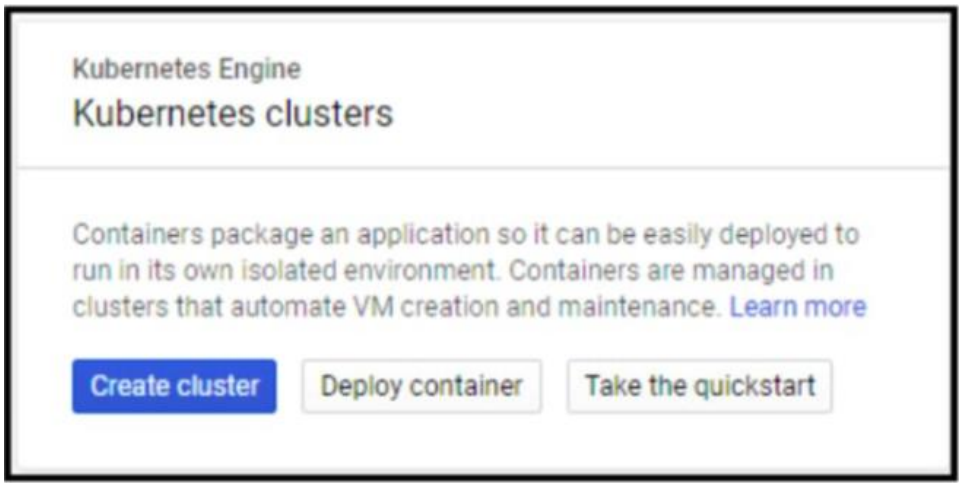
Login into the GCP console with your Gmail account. By default, it will create a project "My Project" for you. It is mandatory to complete the billing verification to use GCP services. Once the Billing account is verified you can use all services at $300 credit.

1. Click on the left side menu bar, scroll down, and click on **Kubernetes Engine.**

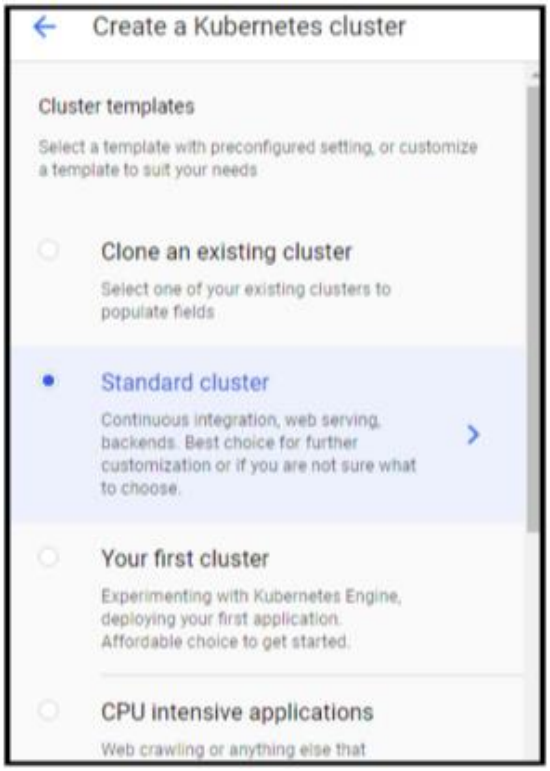


Stanley Stephen, M.C.A.,
Linkedin: https://www.linkedin.com/in/contactstanley/
Github: https://github.com/stanleymca/Learn_from_Stanley/GKE_by_Stanley.pdf
Email: s.stanley.mca@gmail.com

2. Let's go to the cluster page, you must click "**Create cluster**" to work on it

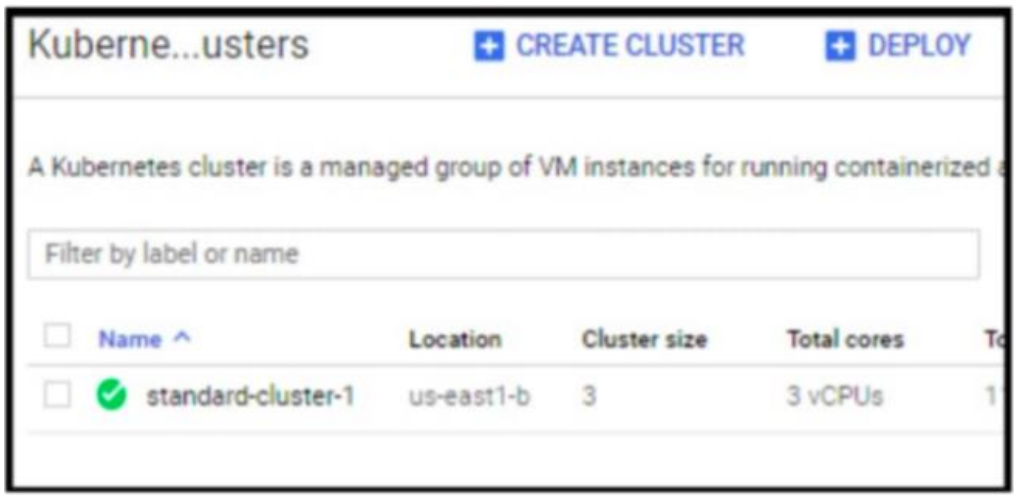


3. On the next page, choose cluster type and modify the details as per your requirement and you must click on the "**creating button**". It won't take more than 3 min for you to get a ready cluster

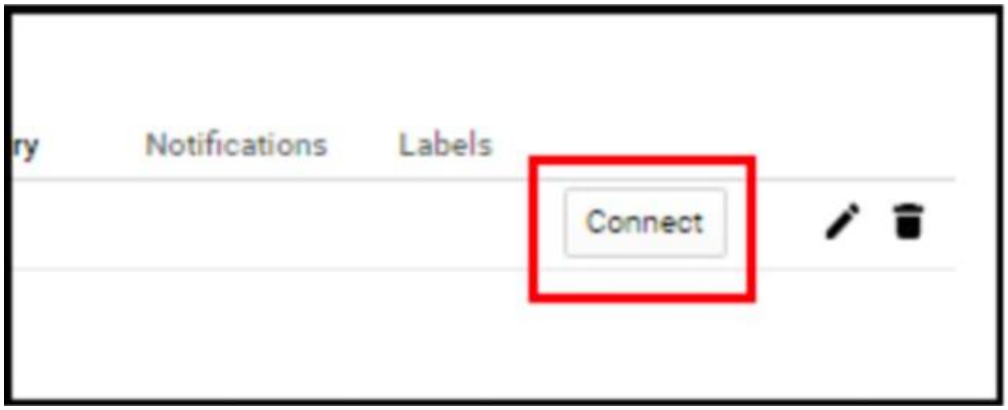


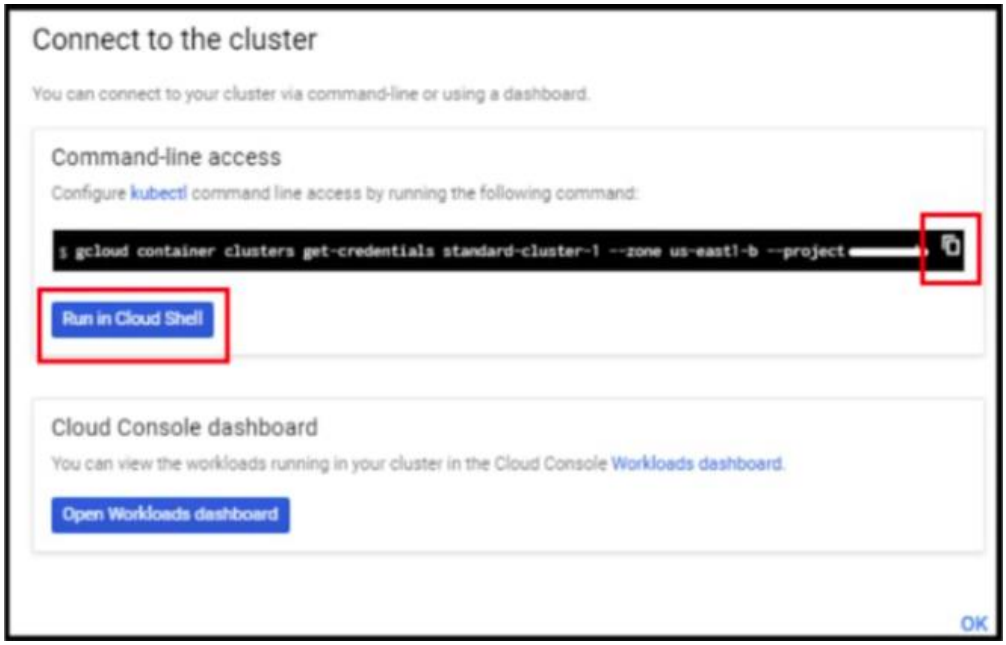
Stanley Stephen, M.C.A.,
Linkedin: https://www.linkedin.com/in/contactstanley/
Github: https://github.com/stanleymca/Learn_from_Stanley/GKE_by_Stanley.pdf
Email: s.stanley.mca@gmail.com

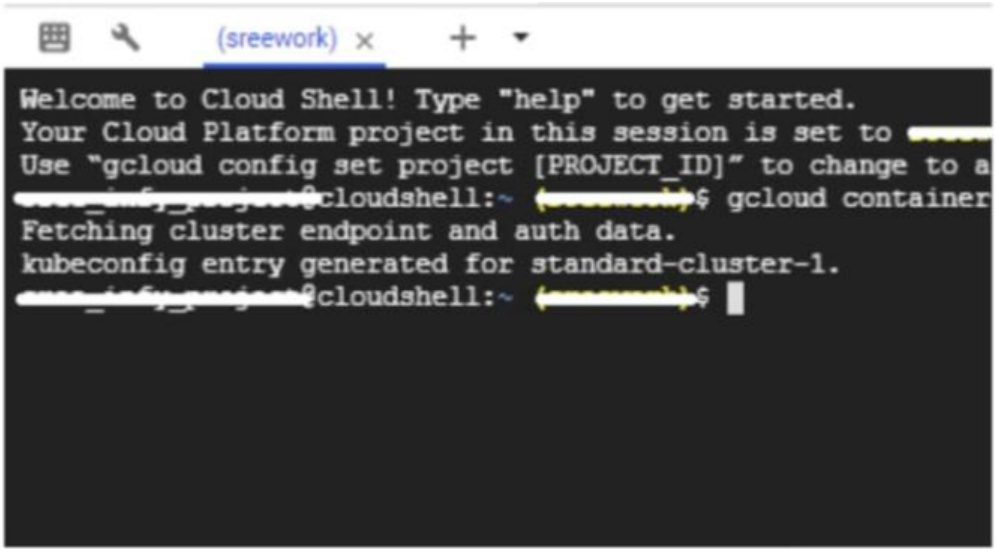
4. Once the cluster is created, Now you click on the “connect” option. It will give you a command to run on a cloud shell. Copy the command and run in the cloud shell.



Stanley Stephen, M.C.A.,
Linkedin: https://www.linkedin.com/in/contactstanley/
Github: https://github.com/stanleymca/Learn_from_Stanley/GKE_by_Stanley.pdf
Email: s.stanley.mca@gmail.com







Stanley Stephen, M.C.A.,
Linkedin: https://www.linkedin.com/in/contactstanley/
Github: https://github.com/stanleymca/Learn_from_Stanley/GKE_by_Stanley.pdf
Email: s.stanley.mca@gmail.com

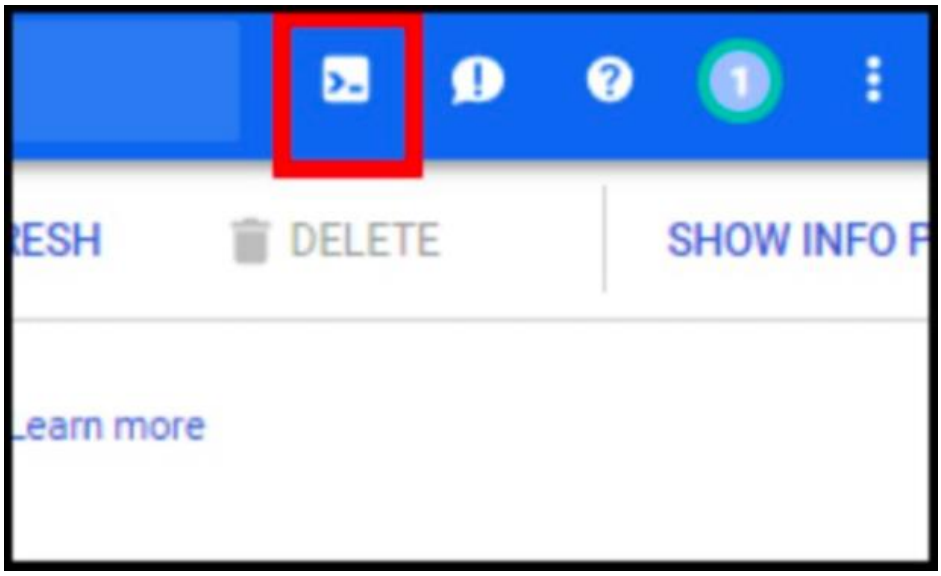
5. Kubectl is available in the cloud shell by default. To check the nodes, run the "**kubectl get nodes**" command in the cloud shell.

```
$ kubectl get nodesNAME STATUS ROLES AGE VERSION
gke-my-cluster-default-pool-2d75839c-f1h3 Ready < none > 5m
v1.11.6-gke.2
gke-my-cluster-default-pool-2d75839c-m3j7 Ready < none > 5m
v1.11.6-gke.2
gke-my-cluster-default-pool-2d75839c-vx44 Ready < none > 5m
v1.11.6-gke.2
```

Now, you can deploy your **containerized application** on Google Kubernetes Engine.

Create a cluster using gcloud CLI commands

Connect to cloud shell using the **cloud shell** button and click on the new window icon to open in the new tab.



Step 1: Set the **region** to launch cluster nodes

```
$ gcloud config set compute/zone us-east1-b
```


Stanley Stephen, M.C.A.,
Linkedin: https://www.linkedin.com/in/contactstanley/
Github: https://github.com/stanleymca/Learn_from_Stanley/GKE_by_Stanley.pdf
Email: s.stanley.mca@gmail.com

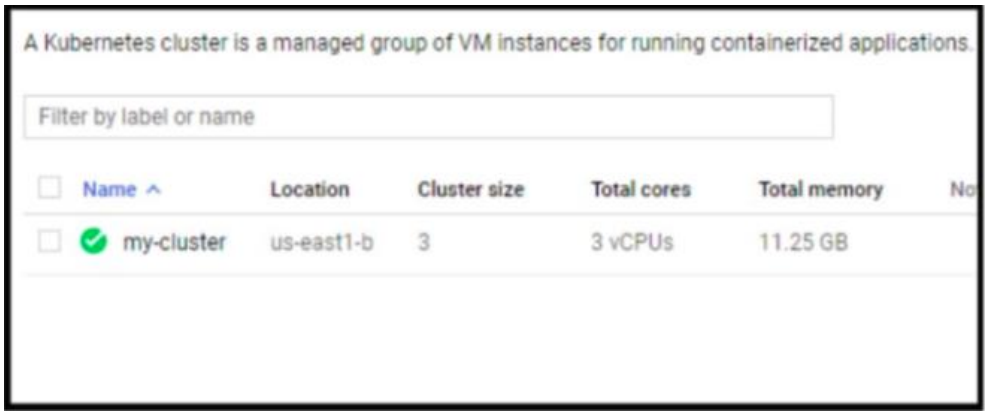
Step 2: Create a **network** for your cluster, or you can use the default network.

```
$ gcloud compute networks create my-cluster-network
```

Step 3: Create a cluster with the required configuration

```
$ gcloud container clusters create my-cluster \
— enable-cloud-logging \
— enable-cloud-monitoring \
— num-nodes 3 \
— disk-size 30G \
— network my-cluster-network
```

The above command will create a **new cluster** for you.



Step 4: Get credentials for your cluster. GKE will be using these secret credentials to allow you to access the newly provisioned cluster.

```
$ gcloud container clusters get-credentials my-cluster
```

Now check with **kubectl**

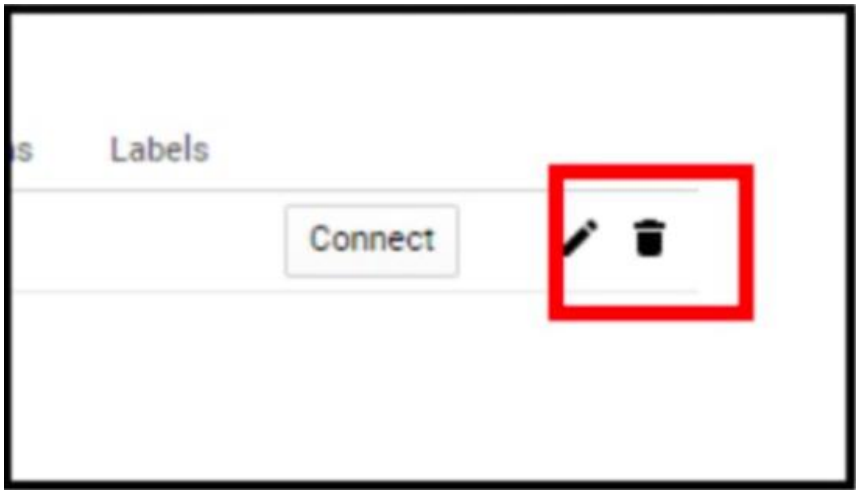
```
$ kubectl get nodesNAME STATUS ROLES AGE VERSION
gke-my-cluster-default-pool-2d75839c-f1h3 Ready < none > 5m
v1.11.6-gke.2
gke-my-cluster-default-pool-2d75839c-m3j7 Ready < none > 5m
v1.11.6-gke.2
gke-my-cluster-default-pool-2d75839c-vx44 Ready < none > 5m
v1.11.6-gke.2
```


Stanley Stephen, M.C.A.,
Linkedin: https://www.linkedin.com/in/contactstanley/
Github: https://github.com/stanleymca/Learn_from_Stanley/GKE_by_Stanley.pdf
Email: s.stanley.mca@gmail.com

To **delete** the cluster, run the following command:

```
$ gcloud container clusters delete my-cluster
```

You can also delete the cluster from the console as well.



Stanley Stephen, M.C.A.,
Linkedin: https://www.linkedin.com/in/contactstanley/
Github: https://github.com/stanleymca/Learn_from_Stanley/GKE_by_Stanley.pdf
Email: s.stanley.mca@gmail.com