



Kubernetes in a cloud guru

Setting in Google cloud sandbox

**Google Cloud Sandbox**
Jump into a Google Cloud Platform sandbox session to try things out on your own.
[▶ Start Google Cloud Sandbox](#)

**Google Cloud Sandbox**
Jump into a Google Cloud Platform sandbox session to try things out on your own.

Username	<input type="text" value="cloud_user_p_af62ed..."/>
Password	<input type="password" value="QeHSqqtK"/>
URL	<input type="text" value="https://console.clou..."/>

Auto Shutdown **2:25 pm**
The sandbox will be shut down and cleaned at this time.

[Open Sandbox](#) [Delete Sandbox](#)

⚠ Remember to **Right Click** and "**Open Link in incognito Window**" or your browser's equivalent mode.

Why open the sandbox in an incognito window?
To access the cloud console, we recommend you open the link by using your browser's "incognito" or "Private Browsing" mode.
This is important in case you have ever accessed the console before, as cookies and other factors may interfere with what is supposed to be a clean, sandboxed experience.
Learn more about why this matters in our [knowledge base article](#).

Go to Kubernetes than cluster

Recent

[View all products](#)

PINNED

API

APIs & Services

>

Billing

IAM & Admin

>

Marketplace

Compute Engine

>

Kubernetes Engine

>

Clusters

Workloads

Services & Ingress

Applications

Secrets & ConfigMaps

Storage

Object Browser

Migrate to Containers

Backup for GKE

Config Management

Protect

Google Cloud Platform

90-day, \$300 free trial to get you started
Always free products to keep you going

[TRY FOR FREE](#)

Cloud Storage

A powerful, simple and cost effective object storage service


Cloud SQL

A fully-managed MySQL, PostgreSQL, and SQL Server database service

Cloud Run

Fully managed > out for deploying and scaling containerized applications

Click on enable button

**Kubernetes Engine API**
[Google Enterprise API](#)
Builds and manages container-based applications, powered by the open source Kubernetes technology.

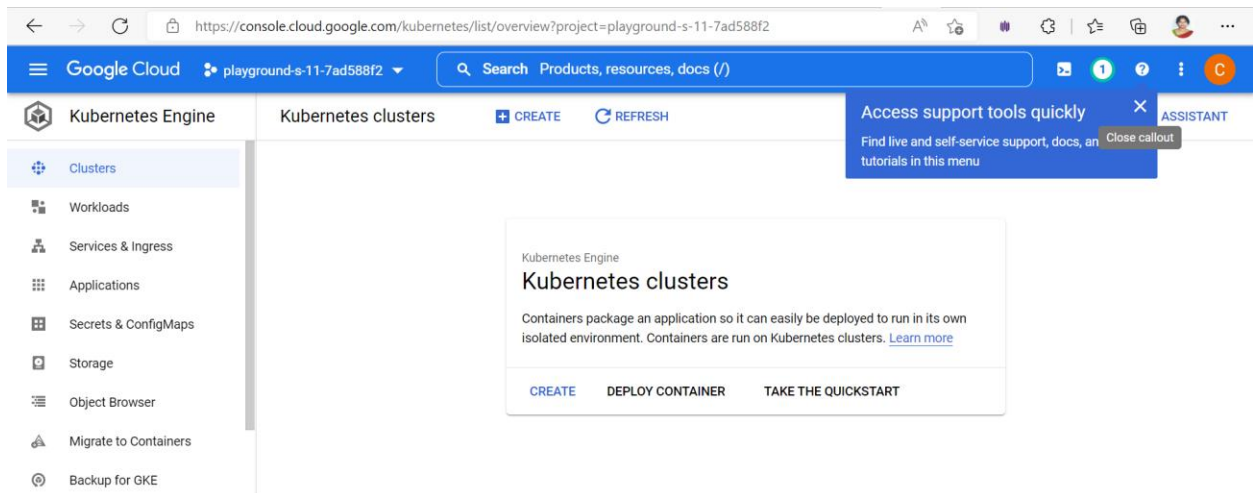
[ENABLE](#) [TRY THIS API](#)

[OVERVIEW](#) [DOCUMENTATION](#)

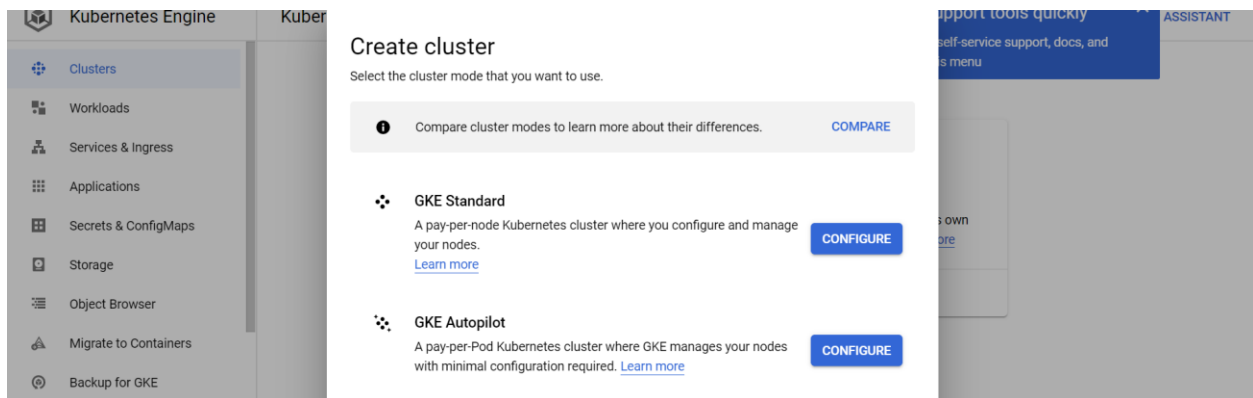
Access support tools quickly

Find live and self-service support, docs, and tutorials in this menu

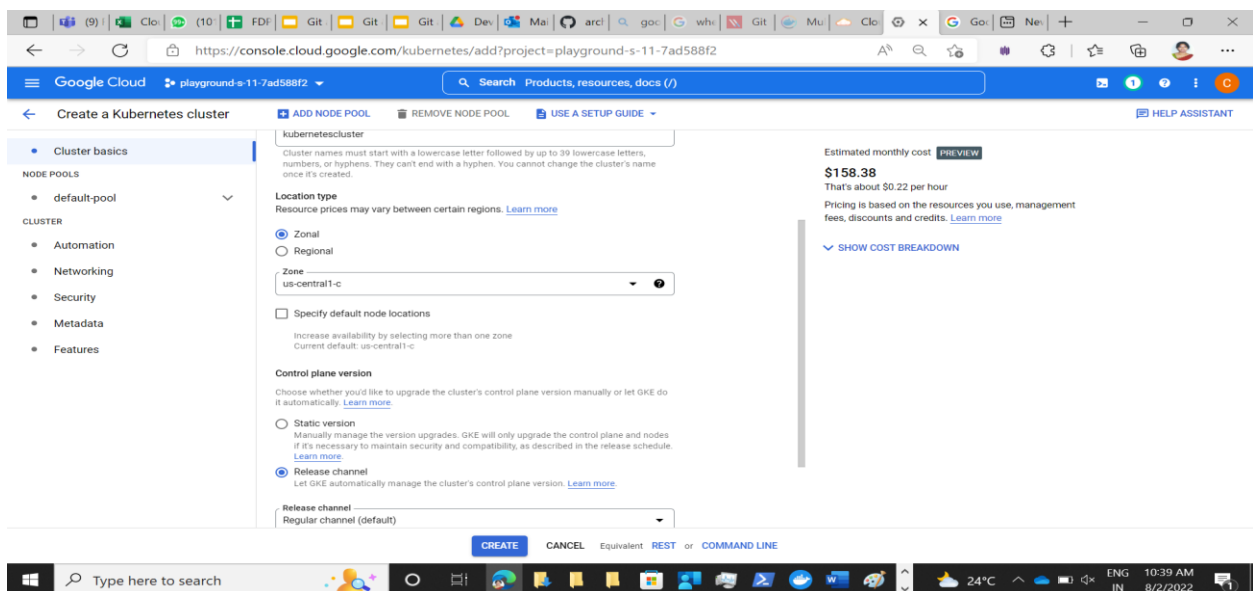
Click on create



select on GKE standard



Give the name of cluster and keep all as default then click on create



We can see cluster has been created with three nodes

Google Cloud playground-s-11-7ad588f2

Kubernetes Engine 1 Kubernetes cluster selected

OVERVIEW OBSERVABILITY PREVIEW COST OPTIMIZATION

Filter Enter property name or value

Status	Name	Location	Number of nodes	Total vCPUs	Total memory	Notifications	Labels
<input checked="" type="checkbox"/>	kubernetescluster	us-central1-c	3	6	12 GB		

Click on connect to go console

Google Cloud playground-s-11-7ad588f2

Kubernetes Engine 1 Kubernetes cluster selected

OVERVIEW OBSERVABILITY PREVIEW COST OPTIMIZATION

Filter Enter property name or value

Status	Name	Location	Number of nodes	Total vCPUs	Total memory	Notifications	Labels
<input checked="" type="checkbox"/>	kubernetescluster	us-central1-c	3	6	12 GB		

Edit Connect Delete

Click on RUN IN CLOUD SHELL

Connect to the cluster

You can connect to your cluster via command-line or using a dashboard.

Command-line access

Configure [kubectl](#) command line access by running the following command:

```
$ gcloud container clusters get-credentials kubernetescluster --zone us-central1-c --project playground-s-11-7ad5
```

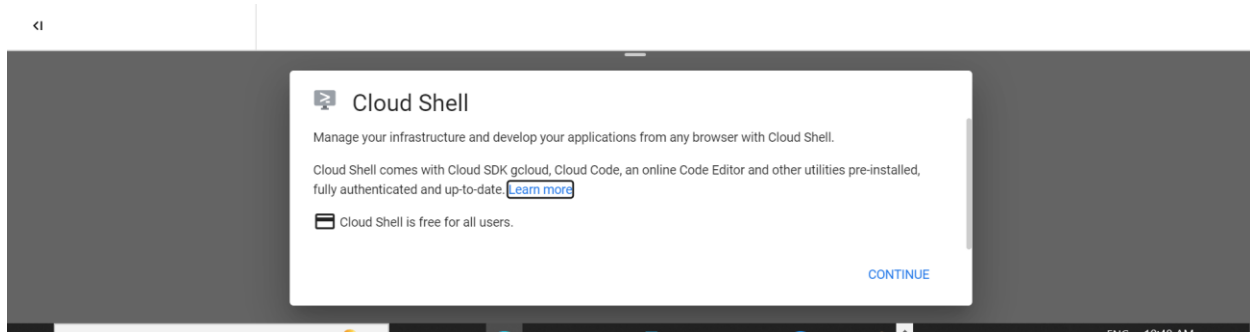
[RUN IN CLOUD SHELL](#)

Cloud Console dashboard

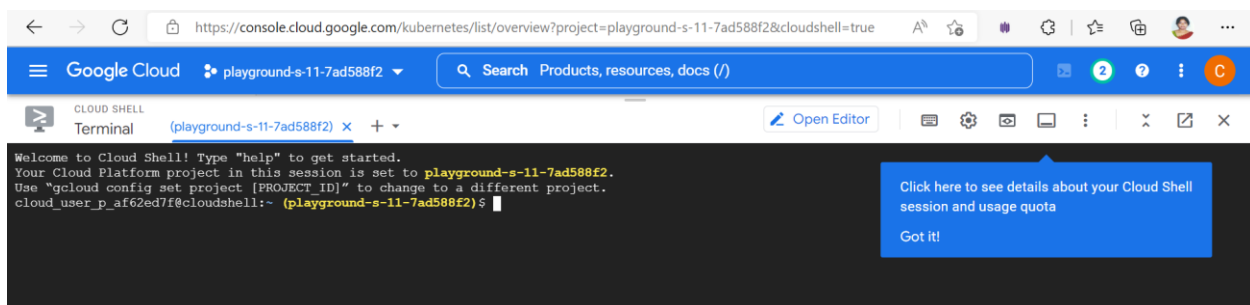
You can view the workloads running in your cluster in the Cloud Console [Workloads dashboard](#).

[OPEN WORKLOADS DASHBOARD](#)

OK



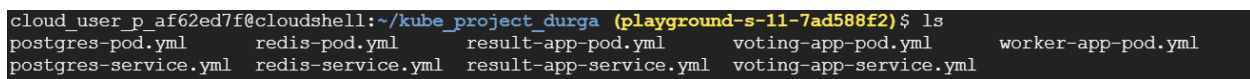
Now console is ready



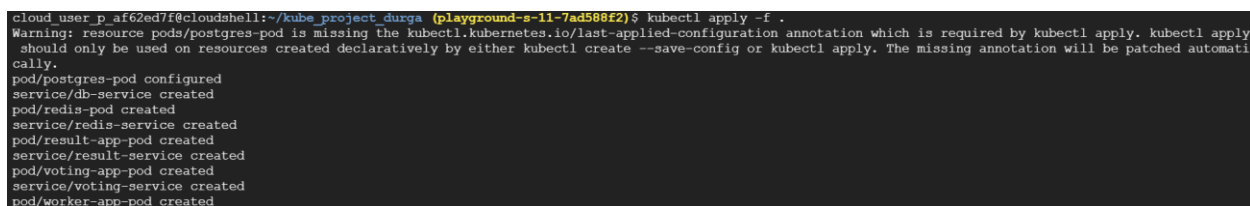
Clone the project from github repository



Check for list of files by ls



Now run the commands by kubectl apply -f



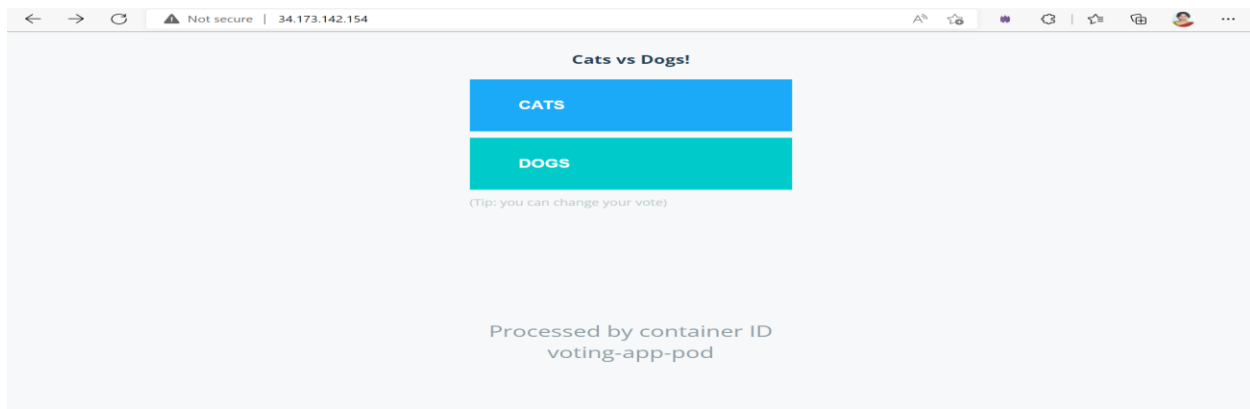
Now check for kubectl get all

```
cloud_user_p_af62ed7f@cloudshell:~/kube_project_durga (playground-s-11-7ad588f2)$ kubectl get all
```

NAME	READY	STATUS	RESTARTS	AGE
pod/postgres-pod	0/1	CrashLoopBackOff	8 (3m10s ago)	19m
pod/redis-pod	1/1	Running	0	40s
pod/result-app-pod	1/1	Running	0	38s
pod/snappass	1/1	Running	2 (21m ago)	22m
pod/snappass-nginx	1/1	Running	0	22m
pod/snappass-redis	1/1	Running	0	22m
pod/voting-app-pod	1/1	Running	0	36s
pod/worker-app-pod	0/1	Error	1 (11s ago)	34s

NAME	TYPE	CLUSTER-IP	EXTERNAL-IP	PORT(S)	AGE
service/db-service	ClusterIP	10.4.7.124	<none>	5432/TCP	42s
service/kubernetes	ClusterIP	10.4.0.1	<none>	443/TCP	27m
service/redis-service	ClusterIP	10.4.1.130	<none>	6379/TCP	41s
service/result-service	LoadBalancer	10.4.4.169	34.122.130.146	80:31563/TCP	39s
service/snappass	ClusterIP	10.4.5.213	<none>	5000/TCP	22m
service/snappass-nginx	LoadBalancer	10.4.0.235	34.134.96.83	443:31735/TCP	22m
service/snappass-redis	ClusterIP	10.4.4.10	<none>	6379/TCP	22m
service/voting-service	LoadBalancer	10.4.12.153	34.173.142.154	80:32101/TCP	37s

Now check for external ip [34.134.96.83]and check in browser(nginx)



Now check for external ip [34.173.142.154]and check in browser

