

# Deploying Docker Images on Google Cloud Platform(GCP)

1. Log In to GCP and search kubernetes and select Kubernetes Engine.

https://console.cloud.google.com/kubernetes?project=superb-shelter-259010

2. Create Kubernetes clusters by clicking 'CREATE'.

Kubernetes clusters - Kubernetes Engine

Kubernetes clusters

CREATE

REFRESH

Kubernetes Engine

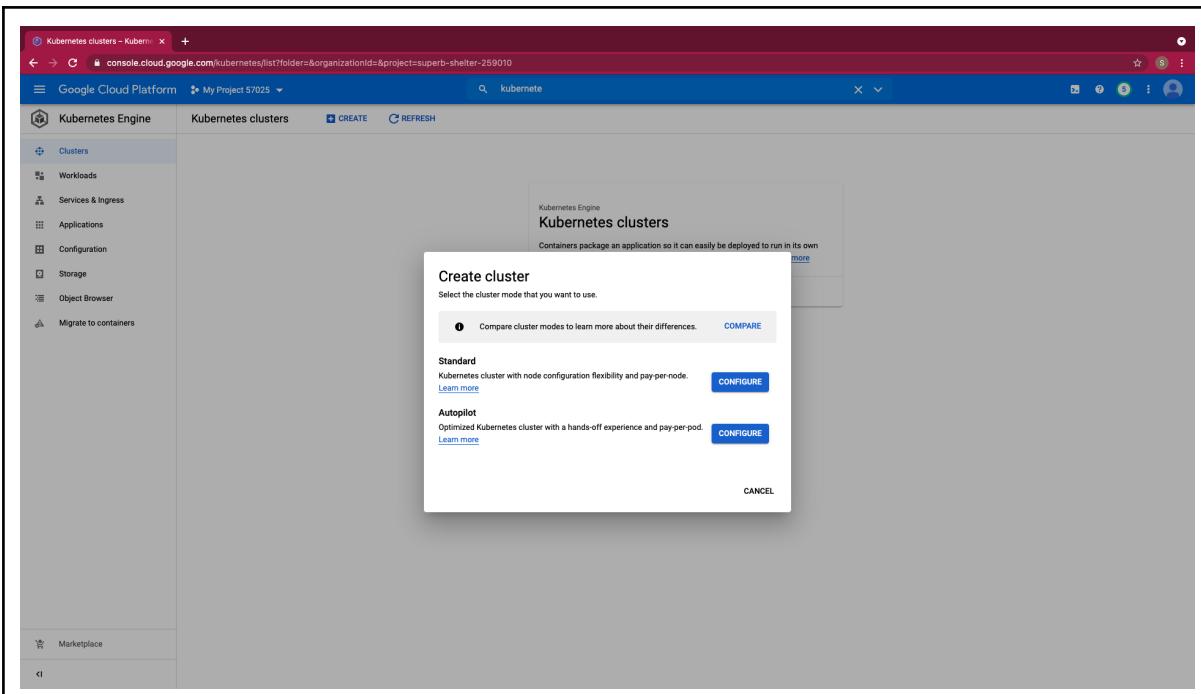
Kubernetes clusters

Containers package an application so it can easily be deployed to run in its own isolated environment. Containers are run on Kubernetes clusters. [Learn more](#)

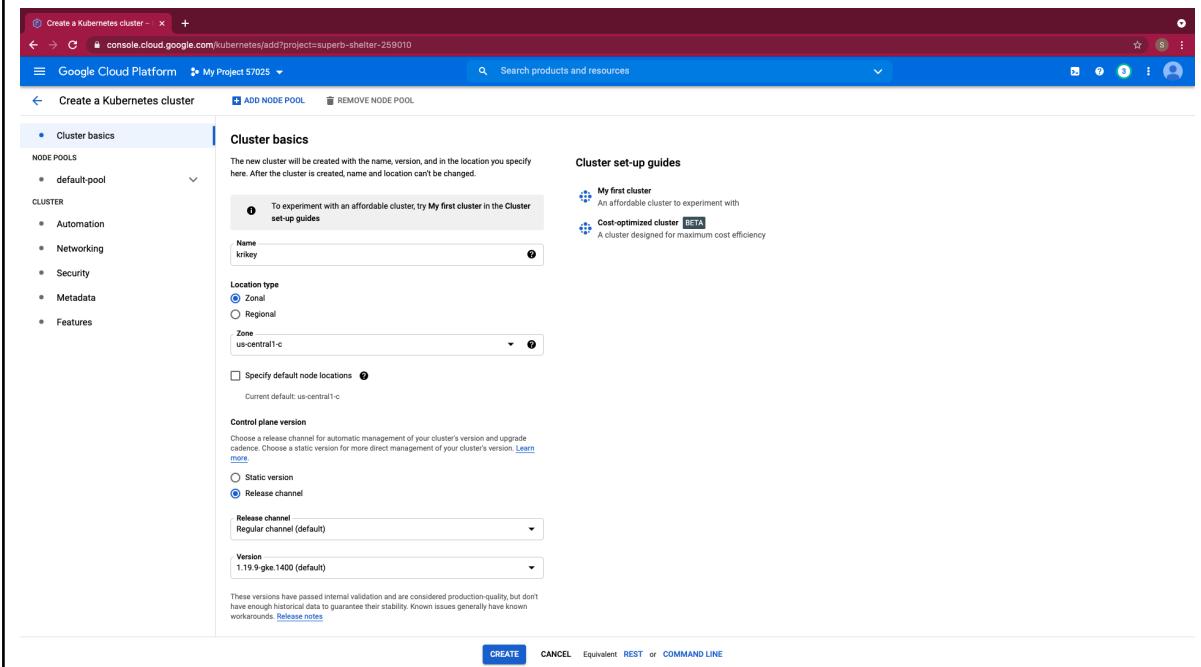
CREATE DEPLOY CONTAINER TAKE THE QUICKSTART

Marketplace

3. Click the 'CONFIGURE' button under 'Standard'.



4. Change the Name if necessary and click 'CREATE'.



5. After a cluster is created, click the 'Actions' button and select 'connect'.

Kubernetes clusters - Kubernetes

Google Cloud Platform My Project 57025 Search products and resources

Kubernetes Engine Kubernetes clusters CREATE DEPLOY REFRESH DELETE HIDE INFO PANEL LEARN

**Clusters**

- Workloads
- Services & Ingress
- Applications
- Configuration
- Storage
- Object Browser
- Migrate to containers

**Introducing Autopilot mode**

An optimized cluster with a hands-off experience. When you create a cluster in Autopilot mode, Google provisions and manages the entire cluster's underlying infrastructure, including nodes and node pools.

Compare cluster modes

- ✓ Get a production-ready cluster based on your workload requirements
- ✓ Eliminate the overhead of node management
- ✓ Pay per Pod, only for the resources that you use
- ✓ Increase security with Google best practices built-in
- ✓ Gain higher workload availability

TRY THE DEMO LEARN MORE

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

No clusters selected

Labels help organize your resources (e.g., cost\_center=sales or env=prod). Learn more

No clusters selected

Marketplace

Kubernetes clusters - Kubernetes

Google Cloud Platform My Project 57025 Search products and resources

Kubernetes Engine Kubernetes clusters CREATE DEPLOY REFRESH DELETE HIDE INFO PANEL LEARN

**Clusters**

- Workloads
- Services & Ingress
- Applications
- Configuration
- Storage
- Object Browser
- Migrate to containers

**Introducing Autopilot mode**

An optimized cluster with a hands-off experience. When you create a cluster in Autopilot mode, Google provisions and manages the entire cluster's underlying infrastructure, including nodes and node pools.

Compare cluster modes

- ✓ Get a production-ready cluster based on your workload requirements
- ✓ Eliminate the overhead of node management
- ✓ Pay per Pod, only for the resources that you use
- ✓ Increase security with Google best practices built-in
- ✓ Gain higher workload availability

TRY THE DEMO LEARN MORE

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

No clusters selected

Labels help organize your resources (e.g., cost\_center=sales or env=prod). Learn more

No clusters selected

Marketplace

6. Select 'RUN IN CLOUD SHELL' and click 'Continue'.

Kubernetes clusters – Kubernetes

Google Cloud Platform My Project 57025 Search products and resources

**Kubernetes Engine** Clusters CREATE DEPLOY REFRESH DELETE HIDE INFO PANEL LEARN

**Kubernetes clusters**

Introducing Autopilot mode

An optimized cluster with a hands-off experience. When you create a cluster in Autopilot mode, Google provisions and manages the entire cluster's underlying infrastructure, including nodes and node pools.

Compare cluster modes

- ✓ Get a production-ready cluster based on your workload requirements
- ✓ Eliminate the overhead of node management
- ✓ Pay per Pod, only for the resources that you use
- ✓ Increase security with Google best practices built-in
- ✓ Gain higher workload availability

TRY THE DEMO LEARN MORE

Filter Enter property name or value

Name	Location	Number of nodes	Total vCPUs	Total memory	Notifications	Labels
krikey	us-central1-c	3	6	12 GB	--	--

Marketplace

No clusters selected

Labels help organize your resources (e.g., cost\_center=sales or env=prod). [Learn more](#)

Cloud Shell

Manage your infrastructure and develop your applications from any browser with Cloud Shell.

Cloud Shell comes with Cloud SDK gcloud, Cloud Code, an online Code Editor and other utilities pre-installed, fully authenticated and up-to-date. [Learn more](#)

Cloud Shell is free for all users.

Continue

<https://cloud.google.com/shell/docs/>

7. Press the 'Enter' key and click 'Authorize'.

**Kubernetes clusters - KubeSphere**

Google Cloud Platform My Project 57025

Kubernetes Engine Kubernetes clusters CREATE DEPLOY REFRESH DELETE HIDE INFO PANEL LEARN

**Clusters**

- Workloads
- Services & Ingress
- Applications
- Configuration
- Storage
- Object Browser
- Migrate to containers

**Introducing Autopilot mode**

An optimized cluster with a hands-off experience. When you create a cluster in Autopilot mode, Google provisions and manages the entire cluster's underlying infrastructure, including nodes and node pools.

Compare cluster modes

- ✓ Get a production-ready cluster based on your workload requirements
- ✓ Eliminate the overhead of node management
- ✓ Pay per Pod, only for the resources that you use
- ✓ Increase security with Google best practices built-in
- ✓ Gain higher workload availability

TRY THE DEMO LEARN MORE

Filter Enter property name or value

	Name ↑	Location	Number of nodes	Total vCPUs	Total memory	Notifications	Labels
	krikey	us-central1-c	3	6	12 GB	-	

No clusters selected

Labels help organize your resources (e.g., cost\_center sales or env.prod). Learn more

No clusters selected

CLOUD SHELL Terminal (superb-shelter-259010) + Open Editor

```
Welcome to Cloud Shell! Type "help" to get started.
Your Cloud Platform project in this session is set to superb-shelter-259010.
Use "gcloud config set project [PROJECT_ID]" to change to a different project.
gcloud container clusters get-credentials krikey --zone us-central1-c --project superb-shelter-259010 sungohryu@cloudshell:~ (superb-shelter-259010)$ gcloud container clusters get-credentials krikey --zone us-central1-c --project superb-shelter-259010 [1]
```

**Kubernetes clusters - KubeSphere**

Google Cloud Platform My Project 57025

Kubernetes Engine Kubernetes clusters CREATE DEPLOY REFRESH DELETE HIDE INFO PANEL LEARN

**Clusters**

- Workloads
- Services & Ingress
- Applications
- Configuration
- Storage
- Object Browser
- Migrate to containers

**Introducing Autopilot mode**

An optimized cluster with a hands-off experience. When you create a cluster in Autopilot mode, Google provisions and manages the entire cluster's underlying infrastructure, including nodes and node pools.

Compare cluster modes

- ✓ Get a production-ready cluster based on your workload requirements
- ✓ Eliminate the overhead of node management
- ✓ Pay per Pod, only for the resources that you use
- ✓ Increase security with Google best practices built-in
- ✓ Gain higher workload availability

TRY THE DEMO LEARN MORE

Filter Enter property name or value

	Name ↑	Location	Number of nodes	Total vCPUs	Total memory	Notifications	Labels
	krikey	us-central1-c	3	6	12 GB	-	

No clusters selected

Labels help organize your resources (e.g., cost\_center sales or env.prod). Learn more

No clusters selected

CLOUD SHELL Terminal (superb-shelter-259010) + Open Editor

```
+259010
sungohryu@cloudshell:~ (gcloud.container.clusters.get-credentials) You do not currently have
Please run:
$ gcloud auth login
to obtain new credentials.
If you have already logged in with a different account:
$ gcloud config set account ACCOUNT
to select an already authenticated account to use.
sungohryu@cloudshell:~ (superb-shelter-259010)$ [1]
```

Authorize Cloud Shell

gcloud is requesting your credentials to make a GCP API call.  
Click to authorize this and future calls that require your credentials.

Authorize Reject

**8. Type 'git clone <https://github.com/seongohr/Krikey.git>' and cd Krikey to go into the cloned folder.**

```
sungohryu@cloudshell:~ (superb-shelter-259010)$ git clone https://github.com/seongohr/Krikey.git
Cloning into 'Krikey'...
remote: Enumerating objects: 92, done.
remote: Counting objects: 100% (92/92), done.
remote: Compressing objects: 100% (80/80), done.
remote: Total 92 (delta 27), reused 30 (delta 6), pack-reused 0
Unpacking objects: 100% (92/92), done.
sungohryu@cloudshell:~ (superb-shelter-259010)$ ls
Krikey README-cloudshell.txt
sungohryu@cloudshell:~ (superb-shelter-259010)$ cd Krikey
sungohryu@cloudshell:~/Krikey (superb-shelter-259010)$ ls
app.js bin database.sql docker-compose.yml Dockerfile img initialization.js package.json package-lock.json public README.md
```

**9. Type 'docker build -t krikey/app'.**

<important>The red-lined one should match with

docker-compose.yml file

the image name under app in the 'docker-compose.yml' file. See the right image ->

You can check the created docker image by typing 'docker images'.

```
version: "3.3"
services:
  postgresql:
    image: postgres
    depends_on:
      - app
  environment:
    POSTGRES_PASSWORD: [REDACTED]
    POSTGRES_USER: [REDACTED]
    POSTGRES_DB: [REDACTED]
  volumes:
    - ./pgdata:/var/lib/postgresql/data
  ports:
    - '5432:5432'
  app:
    image: krikey/app
    ports:
      - "8080:8080"
```

```
sungohryu@cloudshell:~/Krikey (superb-shelter-259010)$ docker build -t krikey/app .
Sending build context to Docker daemon 327.2kB
Step 1/7 : FROM node
latest: Pulling from library/node
d960726af2be: Pull complete
e8d62473a22d: Pull complete
8962bc0fad55: Pull complete
65d943ee54c1: Pull complete
532f6f723709: Pull complete
f8463f32765b: Pull complete
39c1cd906e85: Pull complete
9b89015c57b4: Pull complete
6a93d724110f: Pull complete
Digest: sha256:025d33d7797aaad52c0e3c96699814946093d4b50f792ec35cbd2f57278b94f2
Status: Downloaded newer image for node:latest
--> 7493e35c7ffa
Step 2/7 : WORKDIR /usr/src/krikey
--> Running in 83ea00087b6a
Removing intermediate container 83ea00087b6a
--> b3172046ea0e
Step 3/7 : COPY package*.json ./
--> 4995781366e3
Step 4/7 : RUN npm install
--> Running in 0af13e4bc9ce

added 85 packages, and audited 86 packages in 2s

found 0 vulnerabilities
npm notice
npm notice New minor version of npm available! 7.13.0 -> 7.14.0
npm notice Changelog: <https://github.com/npm/cli/releases/tag/v7.14.0>
npm notice Run `npm install -g npm@7.14.0` to update!
npm notice
Removing intermediate container 0af13e4bc9ce
--> 64ff7f445f5e
Step 5/7 : COPY .
--> 35ee4c7c8c9f
Step 6/7 : EXPOSE 8080
--> Running in 8ea14091ffd5
Removing intermediate container 8ea14091ffd5
--> ffe2e8a93baf
Step 7/7 : CMD npm run initial && npm run start
--> Running in 47e0fe663a86
Removing intermediate container 47e0fe663a86
--> 365b52746f05
Successfully built 365b52746f05
Successfully tagged krikey/app:latest
sungohryu@cloudshell:~/Krikey (superb-shelter-259010)$ docker images
REPOSITORY      TAG          IMAGE ID      CREATED        SIZE
krikey/app      latest       365b52746f05   2 minutes ago  913MB
node            latest       7493e35c7ffa   2 days ago   908MB
sungohryu@cloudshell:~/Krikey (superb-shelter-259010)$
```

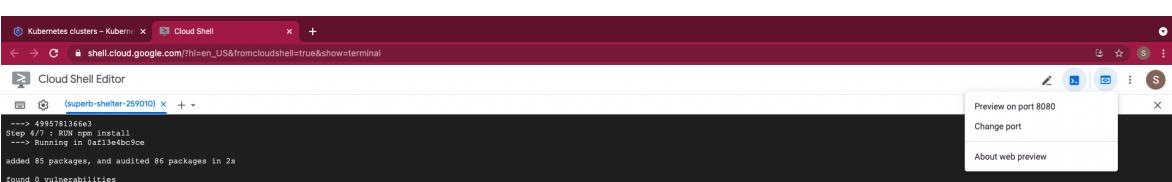
10. Type docker-compose up -d.

```
sungohryu@cloudshell:~/Krikey (superb-shelter-259010)$ docker-compose up -d
Creating network "krikey_default" with the default driver
Pulling postgresql (postgres:)... 
latest: Pulling from library/postgres
69692152171a: Pull complete
a31b993d5cc6: Pull complete
f65921886500: Pull complete
b9c1a94e4ca8: Pull complete
435dd99ceb68: Pull complete
d3ee8e88c67c: Pull complete
84b08674f942: Pull complete
7d358e850d3e: Pull complete
c7dcc5801f3b: Pull complete
f6eeeca01c79c: Pull complete
392faa2e3ddd: Pull complete
3e77feaf6319: Pull complete
9b42e6c9c7ba: Pull complete
5fce2660d75c: Pull complete
Digest: sha256:117c3ea384ce21421541515edfb11f2997b2c853d4fdd58a455b77664cladc20
Status: Downloaded newer image for postgres:latest
Creating krikey_app_1 ... done
Creating krikey_postgresql_1 ... done
sungohryu@cloudshell:~/Krikey (superb-shelter-259010)$
```

## 11. Check the created list of containers by typing ‘docker container list’.

```
sungohryu@cloudshell:~/Krikey (superb-shelter-259010)$ docker container list
CONTAINER ID        IMAGE               COMMAND             CREATED            STATUS              PORTS               NAMES
d06c7ae6877b        postgres            "docker-entrypoint.s..."   2 minutes ago      Up 2 minutes       0.0.0.0:5432->5432/tcp, :::5432->5432/tcp   krikey_postgresql_1
c5cb165cd1ee        krikey/app         "docker-entrypoint.s..."   2 minutes ago      Up 2 minutes       0.0.0.0:8080->8080/tcp, :::8080->8080/tcp   krikey_app_1
```

## 12. Click the ‘Web Preview’ and select ‘Preview on port 8080’.

## 13. You can see the docker images are successfully deployed by the responses from the api.

