

## Deploy to google cloud run .

We need to deploy a docker image . We need to deploy docker image in this .

Source code - app.py , requirements.txt, Dockerfile

Push the changes to repo .

cat Dockerfile :

FROM python

RUN pip install flask

WORKDIR /myapp

COPY ./main.py /myapp/

CMD ["python","/myapp/main.py"]

---

```
from flask import Flask
```

```
app = Flask(__name__)
```

```
@app.route('/')
```

```
def index():
```

```
    return 'Web App with Python Flask!'
```

```
if __name__ == '__main__':
```

```
    app.run(host='0.0.0.0',port=8080)
```

Cloud build - Docker build, docker push, gcloud run deploy

cloudbuild.yaml - in file rename project id with your own project id .

steps:

#Build the container image

- name: 'gcr.io/cloud-builders/docker'  
args: ['build', '-t', 'gcr.io/your-project-id/imagename', '.']

#Push the container image to container registry

- name: 'gcr.io/cloud-builders/docker'  
args: ['push', 'gcr.io/project-id/imagename']

#Deploy container image to cloud run

- name: 'gcr.io/google.com/cloudsdktool/cloud-sdk'  
entrypoint: gcloud  
args: ['run', 'deploy', 'myfirstrun', '--image', 'gcr.io/project-id/imagename', '--region', 'us-central1', '--allow-unauthenticated']

#images is a global variable which we can utilize in cloudbuild to specify image name .

images:

- gcr.io/project-id/imagename

option:

logging: CLOUD\_LOGGING\_ONLY

---

cat cloudbuild.yaml

steps:

- name: 'gcr.io/cloud-builders/docker'

```

  args: ['build', '-t', 'gcr.io/qwiklabs-gcp-01-97a5175489fe/image-new', '.']
- name: 'gcr.io/cloud-builders/docker'
  args: ['push', 'gcr.io/qwiklabs-gcp-01-97a5175489fe/image-new']
- name: 'gcr.io/google.com/cloudsdktool/cloud-sdk'
  entrypoint: gcloud
  args:
['run', 'deploy', 'image-new', 'gcr.io/qwiklabs-gcp-01-97a5175489fe/image-new', '--region', 'us-centra
l1', '--allow-unauthenticated']
images:
- gcr.io/qwiklabs-gcp-01-97a5175489fe/image-new
options:
  logging: CLOUD_LOGGING_ONLY

```

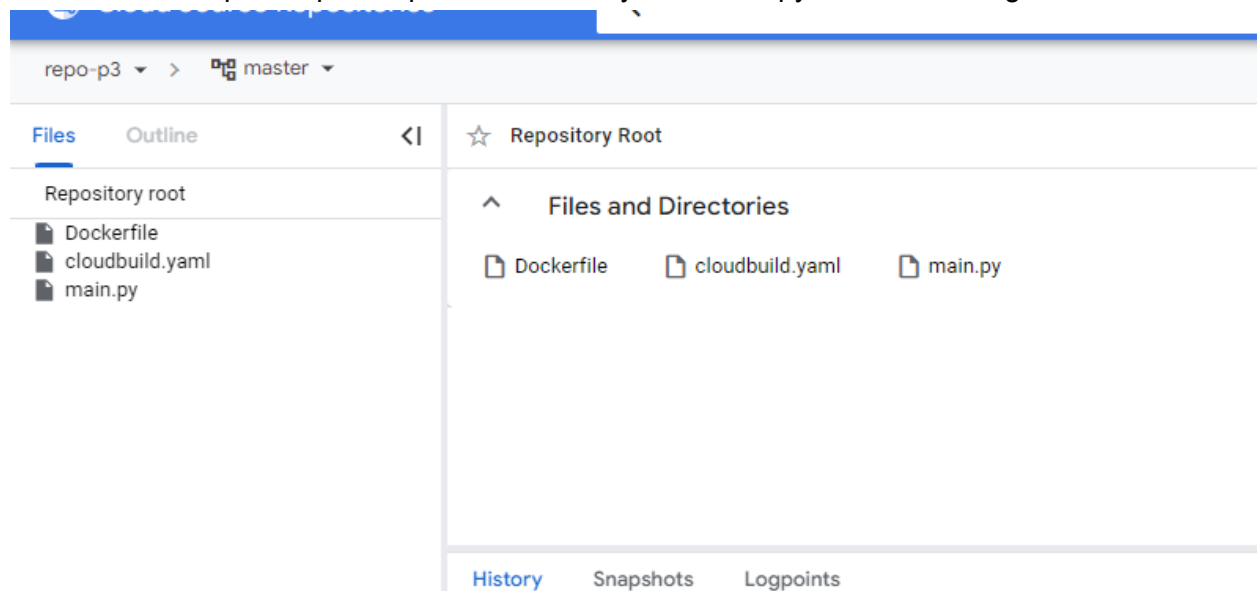
Your build failed to run: generic::invalid\_argument: if 'build.service\_account' is specified, the build must either (a) specify 'build.logs\_bucket' (b) use the CLOUD\_LOGGING\_ONLY logging option, or (c) use the NONE logging option

We have to specify CLOUD\_LOGGING\_ONLY if we use any service account .

Deploy to cloud run

We will implement multiple steps in cloud build .

Create source repo > repo-4 > push cloudbuild.yaml , main.py , Dockerfile to git .



Cloud build > Trigger > based on yaml > on repo master branch event > create

repository event that invokes ti

- ☒ Push to a branch
- ☐ Push new tag
- ☐ Pull request  
Not available for Cloud Sourc

Or in response to

- ☐ Manual invocation
- ☐ Pub/Sub message
- ☐ Webhook event

## Source

**Repository \*** \_\_\_\_\_  
repo-p3 (Cloud Source Repos  
Select the repository to watch fo

**Branch \*** \_\_\_\_\_  
^master\$  
Trigger only for a branch that m:

Type

- ☒ Cloud Build configuration file (yaml or json)
- ☐ Dockerfile
- ☐ Buildpacks


Location

- ☒ Repository  
repo-p3 (Cloud Source Repositories)
- ☐ Inline  
Write inline YAML

**Cloud Build configuration file location \*** \_\_\_\_\_  
/ cloudbuild.yaml

Specify the path to a Cloud Build configuration file i


Run trigger . We can check in the container registry whether the image is pushed or not .


Container Registry

Images

Settings

Repositories



Transi

Artifact R  
still supp  
Artifact R

TRY ART

qwiklabs-gcp-01-97a5175489

Filter

Enter property name or v

Name ↑	Hostname
<a href="#">image-new</a>	gcr.io

We can check in the cloud whether our deployment is done or not .

If `--allow-unauthenticated` will not work then we have to add permission manually for all users or we can add 1 step in cloudbuild for this part .

Allusers - cloud run incoker role so that application will be accessible on internet .

## Add principals

Principals are users, groups, domains, or service accounts. [Learn more about principals in IAM](#)

New principals

allUsers x



## Assign roles

Roles are composed of sets of permissions and determine what the principal can do with this resource. [Learn more](#)

Role \*

Cloud Run Invoker



Can invoke a Cloud Run service.

← → ↻ 🔒 runwithcid-csf5jso5kq-uc.a.run.app

## Web App with Python Flask!

Deployed with cloud run .

```
Step #2: ERROR: (gcloud.run.deploy) PERMISSION_DENIED: Cloud Run Admin AP
has not been used in project 354858291637 before or it is disabled. Enabl
it by visiting
https://console.developers.google.com/apis/api/run.googleapis.com/overvie
project=354858291637 then retry. If you enabled this API recently, wait a
```

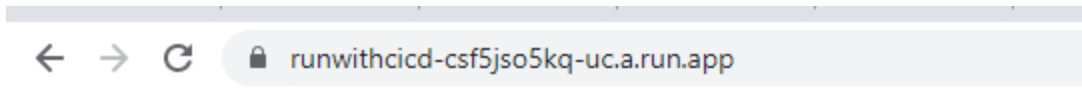
Enable cloud run api .

Was getting erro bcz of cloudbuild.yaml syntax mistake .

```
Branch 'master' set up to track remote branch 'master' from 'origin'.
student_00_e909601900c8@cloudshell:~/repo-p3 (qwiklabs-gcp-01-97a5175489fe)$ cat cloudbuild.yaml
steps:
- name: 'gcr.io/cloud-builders/docker'
  args: ['build', '-t', 'gcr.io/qwiklabs-gcp-01-97a5175489fe/image-new', '.']
- name: 'gcr.io/cloud-builders/docker'
  args: ['push', 'gcr.io/qwiklabs-gcp-01-97a5175489fe/image-new']
- name: 'gcr.io/google.com/cloudsdktool/cloud-sdk'
  entrypoint: gcloud
  args: ['run', 'deploy', 'runwithcid', '--image', 'gcr.io/qwiklabs-gcp-01-97a5175489fe/image-new', '--region', 'us-central1', '--allow-unauthenticated']
images:
- gcr.io/qwiklabs-gcp-01-97a5175489fe/image-new
options:
  logging: CLOUD_LOGGING_ONLY
student_00_e909601900c8@cloudshell:~/repo-p3 (qwiklabs-gcp-01-97a5175489fe)$
```

Use service account if you get permission issues .

Update main.py and check whether trigger runs or not .



Web App with Python Flask hi!

Cloud run is deployed after updating main.py