

Quickstart: Build

This page explains how to use Cloud Build to build a Docker image and push the image to Container Registry. Container Registry provides secure private Docker image storage on Google Cloud.

You will first build the image using a `Dockerfile`, which is the Docker configuration file, and then build the same image using the Cloud Build configuration file.

Note: This quickstart shows you how to build an image with `docker`, but Cloud Build supports most build tools and programming languages.

Before you begin

1. [Sign in](https://accounts.google.com/Login) (https://accounts.google.com/Login) to your Google Account.

If you don't already have one, [sign up for a new account](https://accounts.google.com/SignUp) (https://accounts.google.com/SignUp).

2. In the Cloud Console, on the project selector page, select or create a Cloud project.

★ **Note:** If you don't plan to keep the resources that you create in this procedure, create a project instead of selecting an existing project. After you finish these steps, you can delete the project, removing all resources associated with the project.

[Go to the project selector page](https://console.cloud.google.com/projectselector2/home/dashboard) (https://console.cloud.google.com/projectselector2/home/dashboard)

3. Make sure that billing is enabled for your Google Cloud project. [Learn how to confirm billing is enabled for your project](/billing/docs/how-to/modify-project) (/billing/docs/how-to/modify-project).

4. Enable the Cloud Build API.

[Enable the API](https://console.cloud.google.com/flows/enableapi?apiid=cloudbuild.googleapis.com&) (https://console.cloud.google.com/flows/enableapi?apiid=cloudbuild.googleapis.com&)

5. [Install and initialize the Cloud SDK](/sdk/docs) (/sdk/docs).

- ★ If you've already installed Cloud SDK previously, make sure you have the latest available version by running `gcloud components update`.

Preparing source files

You'll need some sample source code to package into a container image. In this section, you'll create a simple shell script and a `Dockerfile`. A `Dockerfile` is a text document that contains instructions for Docker to build an image.

1. Open a terminal window.
2. Create a new directory named `quickstart-docker` and navigate into it:

```
mkdir quickstart-docker
cd quickstart-docker
```

3. Create a file named `quickstart.sh` with the following contents:

[GitHub](https://github.com/GoogleCloudBuild/code-examples/blob/master/quickstart-build/quickstart.sh) (https://github.com/GoogleCloudBuild/code-examples/blob/master/quickstart-build/quickstart.sh)

```
#!/bin/sh
echo "Hello, world! The time is $(date)."
```

4. Create a file named `Dockerfile` with the following contents:

[on GitHub](https://github.com/GoogleCloudBuild/code-examples/blob/master/quickstart-build/Dockerfile) (https://github.com/GoogleCloudBuild/code-examples/blob/master/quickstart-build/Dockerfile)

```
FROM alpine
COPY quickstart.sh /
CMD ["/quickstart.sh"]
```

5. Run the following command to make `quickstart.sh` executable:

```
chmod +x quickstart.sh
```

Build using Dockerfile

Cloud Build allows you to build a Docker image using a `Dockerfile`. You don't require a separate Cloud Build config file.

To build using a `Dockerfile`:

1. Get your Cloud project ID by running the following command:

```
gcloud config get-value project
```

2. Run the following command from the directory containing `quickstart.sh` and `Dockerfile`, where ***project-id*** is your Cloud project ID:

```
gcloud builds submit --tag gcr.io/project-id/quickstart-image
```

After the build is complete, you will see an output similar to the following:

DONE

ID	CREATE_TIME	DURATION	SOURCE
\$BUILD_ID	2016-09-28T13:46:29+00:00	9S	gs://[PROJECT

You've just built a Docker image named `quickstart-image` using a `Dockerfile` and pushed the image to Container Registry.

Build using a build config file

In this section you will use a Cloud Build config file to build the same Docker image as above. The build config file instructs Cloud Build to perform tasks based on your specifications.

1. In the same directory that contains `quickstart.sh` and the `Dockerfile`, create a file named `cloudbuild.yaml` with the following contents. This file is your build config file. At build time, Cloud Build automatically replaces `$PROJECT_ID` with your project ID.

[tHub](https://github.com/GoogleCloudBuild/code-examples/blob/master/quickstart-build/cloudbuild.yaml) (https://github.com/GoogleCloudBuild/code-examples/blob/master/quickstart-build/cloudbuild.yaml)

```
steps:
- name: 'gcr.io/cloud-builders/docker'
  args: [ 'build', '-t', 'gcr.io/$PROJECT_ID/quickstart-image', '.' ]
images:
- 'gcr.io/$PROJECT_ID/quickstart-image'
```

2. Start the build by running the following command:

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

When the build is complete, you will see an output similar to the following:

DONE

```
-----
ID                                CREATE_TIME                        DURATION SOURCE
$BUILD_ID                         2016-09-28T13:46:29+00:00        8S      gs://[PROJECT
```

You've just built `quickstart-image` using the build config file and pushed the image to Container Registry.

View the build details

1. Open the Cloud Build page in the Google Cloud Console.

[Open the Cloud Build page](https://console.cloud.google.com/cloud-build) (https://console.cloud.google.com/cloud-build)

2. Select your project and click **Open**.

You will see the **Build history** page:




Build history		STOP STREAMING BUILDS	
Filter builds			
Build	Tags	Source	Branch
608c1028		Google Cloud Storage	-
6bccfe6c		Google Cloud Storage	-

3. Click on a particular build.

You will see the **Build details** page.

4. To view the artifacts of your build, under **Build Summary**, click **Build Artifacts**.

You will see an output similar to the following:

Build Summary	
BUILD LOG	EXECUTION DETAILS
BUILD ARTIFACTS	
Filter artifacts	
Artifact Name	Type
gs://843828483119.cloudbuild-logs.googleusercontent.com/log-608c1028-da6c-466e-b078-c9bb676070...	Build log 
gcr.io/[redacted]/quickstart-image	Image 
gcr.io/[redacted]/quickstart-image:latest	Image 

You can download your build log and view your image details in Container Registry from this page.

Clean up

To avoid incurring charges to your Google Cloud account for the resources used in this quickstart, follow these steps.

1. Open the Container Registry page in the Google Cloud Console.

[Open the Container Registry page](https://console.cloud.google.com/gcr) (https://console.cloud.google.com/gcr)

2. Select your project and click **Open**.

3. Click **quickstart-image**.
4. Select all the images and click **Delete**.

You have now deleted the images that you created as part of this quickstart.

That's it, you completed this quickstart!

What's next

- Learn how to create a basic [build config file](/cloud-build/docs/configuring-builds/create-basic-configuration) (/cloud-build/docs/configuring-builds/create-basic-configuration).
- Learn how to [deploy using Cloud Build](/cloud-build/docs/quickstart-deploy) (/cloud-build/docs/quickstart-deploy).
- Learn how to [build containers](/cloud-build/docs/building/build-containers) (/cloud-build/docs/building/build-containers).
- Learn about the [cloud builders](/cloud-build/docs/cloud-builders) (/cloud-build/docs/cloud-builders) that you can use to execute your tasks in Cloud Build.
- Read our resources about [DevOps](/devops) (/devops) and explore our [research program](https://www.devops-research.com/research.html) (https://www.devops-research.com/research.html).

Except as otherwise noted, the content of this page is licensed under the [Creative Commons Attribution 4.0 License](https://creativecommons.org/licenses/by/4.0/) (https://creativecommons.org/licenses/by/4.0/), and code samples are licensed under the [Apache 2.0 License](https://www.apache.org/licenses/LICENSE-2.0) (https://www.apache.org/licenses/LICENSE-2.0). For details, see the [Google Developers Site Policies](https://developers.google.com/site-policies) (https://developers.google.com/site-policies). Java is a registered trademark of Oracle and/or its affiliates.

Last updated 2020-08-06 UTC.