The Cloud SDK Docker Image is essentially Cloud SDK installed on top of a Debian-based OS image.

This allows you to pull your desired version of Cloud SDK as a Docker image from Dockerhub and quickly execute `gcloud` commands within an isolated, correctly configured container.

The Docker image itself is hosted on both Container Registry (/sdk/docs/gcr.io/google.com/cloudsdktool/cloud-sdk) and Docker Hub (https://hub.docker.com/r/google/cloud-sdk/), with the following repository names:

- Container Registry: `gcr.io/google.com/cloudsdktool/cloud-sdk`.
- Docker Hub: `google/cloud-sdk`.

The Cloud SDK Docker images comes in three flavors; latest, slim, and alpine. You can specify your preference by using the appropriate tag (after the host repository name):

- `:latest`, `:VERSION`: Large (Debian-based) image with additional components pre-installed
- `:slim`, `:VERSION-slim`: Smaller (Debian-based) image with no components pre-installed
- `:alpine`, `:VERSION-alpine`: Smallest (Alpine-based) image with no additional components installed

The following workflow uses the Container Registry image. To use the Docker Hub image instead, replace all instance
`o/google.com/cloudsdktool/cloud-sdk` with **`google/cloud-sdk`**.

1. To use the image of the latest Cloud SDK release, `gcr.io/google.com/cloudsdktool/cloud-sdk:latest`, pull it from Container Registry (/sdk/docs/gcr.io/google.com/cloudsdktool/cloud-sdk) by running the following command:

2. Verify the installation (if you've pulled the latest version) by running:

   Alternatively, run this command for a specific version, 266.0.0:

3. Authenticate with the gcloud command-line tool by running:

   Once you've authenticated successfully, credentials are preserved in the volume of the `gcloud-config container`.

   ★ **Note:** `gcloud-config container` now has a volume containing your Google Cloud credentials. Do not use `gcloud-config` volume in other containers.

4. List compute instances using these credentials to verify by running the container with `--volumes-from`:

By default, the latest images (`gcr.io/google.com/cloudsdktool/cloud-sdk:latest` and `gcr.io/google.com/cloudsdktool/cloud-sdk:VERSION`) have all the `gcloud` components installed.

The `gcr.io/google.com/cloudsdktool/cloud-sdk:slim` and `gcr.io/google.com/cloudsdktool/cloud-sdk:alpine` images, however, do not come with additional components pre-installed. You can extend these images by following the instructions below:

To install an additional component, like `google-cloud-sdk-datastore-emulator`, run the following:

To install additional components for Alpine-based images, create a Dockerfile that uses the Cloud SDK image as the base image. For example, to add `kubectl` and `app-engine-java` components, create a Dockerfile to look like:

Once done, run the following command:

```
docker build -t my-cloud-sdk-docker:alpine .
```

If these additional components require supporting resources, install these dependencies manually.

To install specific Cloud SDK versions, pass your preferred version in when running the `docker build` command, like so: