Getting started

This page explains how developers can get started using Google Cloud APIs.

If you are using Google Cloud APIs for the first time, you can follow the steps in this guide to call the APIs using `curl` commands. You can use `curl` commands to experiment with an API before you develop your application.

Creating a Google account

To use Google Cloud APIs in your applications, you first need to have a Google account. This allows you to use Google developer products, including Google Cloud Console (https://console.cloud.google.com/), Cloud SDK (/sdk), Cloud Logging (/logging), and Cloud Monitoring (/monitoring). If you're new to Google Cloud, create an account (https://console.cloud.google.com/freetrial) to evaluate how our products perform in real-world scenarios. New customers also get $300 in free credits to run, test, and deploy workloads.

Creating a Google project

To use Cloud APIs, you also need to have a Google project. A project is equivalent to a developer account. It serves as a resource container for your Google Cloud resources. It also provides an isolation boundary for your usage of Google Cloud services, so you can manage quota limits and billing independently at the project level. Usage telemetry and dashboards are grouped by projects as well. If you don't already have a project, you can create one using the Cloud Console (https://console.cloud.google.com/).

A project can own a wide range of resources, including API keys, OAuth clients, service accounts, Compute Engine VMs, Cloud Storage buckets, and BigQuery datasets. When an application calls a Cloud API, the project that owns the application credentials is called the *client project*, and the project that owns the target resource is called the *resource project*. One API request may touch multiple resources, hence multiple resource projects are involved.

If you want to stop using Google Cloud for any reasons, you can use Cloud Console (https://console.cloud.google.com/) to delete your project. Your project and all resources in the project will be deleted after the retention window. Note that different types of data have different retention periods.

Discovering APIs

Before using any Cloud APIs, you should use Google Cloud Console API Library (https://console.cloud.google.com/apis/library/browse) to browse available Cloud APIs and discover the ones that best meet your business needs. For more information about a specific Cloud API, visit its public documentation site, such as Spanner API (/spanner).

Enabling APIs

To use a Cloud API, you must enable it for your project. Depending on which services and which projects are involved from your application, including the client project and resource projects, you may need to enable an API for multiple projects.

Enabling an API requires you to accept the Terms of Service and billing responsibility for the API. You need permission `serviceusage.services.enable` on the project to enable APIs. For more information, see Service Usage Access Control (/service-usage/docs/access-control).

Note: You need the **`serviceusage.services.enable`** permission on the project to enable APIs. The service to enable must either be public, or else the service owner must grant the user the **`servicemanagement.services.bind`** permission on the private service. See Service Management Access Control (https://cloud.google.com/service-infrastructure/docs/service-management/access-control) for more information.

To enable an API for a project using the console:

1. Go to the Cloud Console API Library (https://console.cloud.google.com/apis/library?project=_).
2. From the projects list, select the project you want to use.
3. In the API Library, select the API you want to enable. If you need help finding the API, use the search field and/or the filters.
4. On the API page, click **ENABLE**.

From the same page you can disable an API for your project if you no longer use it to avoid misuse and accidental billing charges. You can also enable and disable Cloud APIs using the Cloud SDK (/sdk/docs) and the Service Usage API (https://cloud.google.com/service-usage/docs/reference/rest/v1/services/enable):

```
$ gcloud services enable pubsub.googleapis.com
$ gcloud services disable pubsub.googleapis.com
```

Enabling billing

Some Cloud APIs charge for usage. You need to enable billing for your project before you can start using these APIs in your project. The API usage in a project is charged to the billing account associated with the project.

If you don't have a billing account, go to the Cloud Console billing page (https://console.cloud.google.com/billing) and follow the instructions to create one. Then link your billing account (https://console.cloud.google.com/billing/linkedaccount?project=_) to your project.

Getting application credentials

Cloud APIs only accept API requests from *registered* applications. This requirement helps API producers associate and report API usage to the right project owning the application.

Cloud APIs use *application credentials* for identifying the calling applications. Credential types include API keys, OAuth 2.0 clients, and service accounts. You can use Cloud Console (https://console.cloud.google.com/apis/credentials?project=_) to create, retrieve, and manage your application credentials. For more information about application credentials, see Authentication Overview (/docs/authentication).

Using application credentials

If you are new to Google Cloud APIs, we highly recommend that you use `oauth2l` (https://github.com/google/oauth2l) to experiment with your application credentials and Cloud APIs before writing any application code. `oauth2l` can accept any application credentials and use them to make calls to Google Cloud APIs using the `curl` command.

Example:

```
oauth2l curl --credentials ./creds.json --scope cloud-platform --url https://pubsub.googleapis.co
```

Building applications

If you are building an application using Cloud APIs, we recommend you to use Google Cloud Client Libraries (/apis/docs/cloud-client-libraries) if available. The client libraries can handle common API features for your convenience, such as authentication, error handling, retry, and payload validation. You need to pass your application credentials to the client libraries during initialization, so the client libraries can make calls to Google Cloud APIs on behalf of your application.

For more information, see Client Libraries Explained (/apis/docs/client-libraries-explained).

More information

For more information about authentication, see Authentication overview (/docs/authentication).

For more information about error handling, see Handling Errors (/apis/design/errors#handling_errors).

For more information about billing, see Create, modify, or close your billing account (/billing/docs/how-to/manage-billin ount).

For more information about enabling billing on your project, see Modify a project's billing settings (/billing/docs/how-to/modify-project).

For more information about enabling and disabling APIs, see Enabling and disabling services (/service-usage/docs/enable-disable).

Last updated 2021-12-10 UTC.