Use the Google Cloud Console to perform simple storage management tasks for Cloud Storage. Some typical uses for the Cloud Console include:

- Enabling the Cloud Storage API for a project.
- Creating and deleting buckets.
- Uploading, downloading, and deleting objects.
- Managing Identity and Access Management (IAM) policies.

This page provides an overview of the Cloud Console, including the tasks you can accomplish using the Cloud Console to manage your data. For more advanced tasks, use the gsutil (/storage/docs/gsutil) command line tool or any of the client libraries (/storage/docs/reference/libraries) that support Cloud Storage.

The Cloud Console requires no setup or installation, and you can access it directly in a browser. Depending on your use case, you access Cloud Console in slightly different ways. If you are:

A user granted access to a project

Use: `https://console.cloud.google.com/`.

In order to use Google Cloud Console as a project member, you must be added to the project's member list. A current project owner can give you access, which applies to all buckets and objects defined in the project. For more information, see Adding a member to a project (/storage/docs/access-control/using-iam-permissions#project-add).

A user granted access to a bucket

Use: `https://console.cloud.google.com/storage/browser/[BUCKET_NAME]`.

In this use case, a project owner gives you access to an individual bucket within a larger project. The owner then sends you the bucket name which you substitute into the URL above. You are able to only work with objects in the specified bucket. This is useful for users who are not project members, but who need to access

a bucket. When you access the URL, you are prompted to authenticate with a Google account if you are not already signed in.

A variation of this use case is when a project owner grants **All Users** permission to read objects in a bucket. This creates a bucket whose contents are publicly readable. For more information, see Setting permissions and metadata (#_bucketpermission) below.

A user granted access to an object

Use:
`https://console.cloud.google.com/storage/browser/_details/[BUCKET_NAME]/[OBJECT_NAME]`

In this use case, a project owner gives you access to single objects within a bucket and sends you the URL to access the objects. When you access the URLs, you are prompted to authenticate with a Google account if you are not already signed in.

Note that the form of the URL above is different from the URL for objects that are shared publicly (/storage/docs/access-control/making-data-public). When you share a link publicly, the URL is of the form: `https://storage.googleapis.com/[BUCKET_NAME]/[OBJECT_NAME]`. This public URL does not require a recipient to authenticate with Google and can be used for non-authenticated access to an object.

The Cloud Console enables you to perform basic storage management tasks with your data using a browser. To use the Cloud Console, you must authenticate with Google and have appropriate permission to complete a given task. If you are the account owner who created the project, it is likely you already have all the permission you need to complete the tasks below. Otherwise, you can be added as a project member (Adding a member to a project (/storage/docs/access-control/using-iam-permissions#project-add)) or be given permission to perform actions on a bucket (Setting bucket permissions (#_bucketpermission)).

Cloud Storage uses a flat namespace to store your data but you can use the Cloud Console to create folders and mimic a folder hierarchy. Your data isn't physically stored in a hierarchical structure, but

is displayed like that in the Cloud Console.

Because Cloud Storage has no notion of folders, the folder suffix and object name delimiters are visible when you view your folders using gsutil (/storage/docs/gsutil) or any other command-line tools that may work with Cloud Storage.

See Creating Storage Buckets (/storage/docs/creating-buckets#storage-create-bucket-console) for a step-by-step guide to creating buckets using the Cloud Console.

You can upload data to your bucket by uploading one or more files or a folder containing files. When you upload a folder, the Cloud Console maintains the same hierarchical structure of the folder, including all of the files and folders it contains. You can track the progress of uploads to the Cloud Console using the upload progress window. You can minimize the progress window and continue working with your bucket.

See Uploading Objects (/storage/docs/uploading-objects) for a step-by-step guide to uploading objects to your buckets using the Cloud Console.

You can also upload objects to the Cloud Console by dragging and dropping files and folders from your desktop or file manager tool to a bucket or sub-folder in Cloud Console.

Dragging folders onto the Cloud Console pane is only supported in Chrome. Dragging one or more files onto the Clou
le pane is supported in all browsers.

See Downloading Objects (/storage/docs/downloading-objects) for a step-by-step guide to downloading objects from your buckets using the Cloud Console.

You can also view details of an object by clicking it. If the object can be displayed, the details page includes a preview of the object itself.

If you want to download many objects at once, consider using the gsutil tool (/storage/docs/gsutil).

Because the Cloud Storage system has no notion of folders, folders created in the Cloud Console are a convenience to help you organize objects in a bucket. As a visual aid, the Cloud Console shows folders with a folder icon image to help you distinguish folders from objects.

From within a bucket (or a folder in a bucket) you can create a new folder by clicking the **Create Folder** button. Unlike buckets, folders don't have to be globally unique. That is, while a bucket name can only be used if there are no buckets already in existence with that name, folder names can be used repeatedly so long as they don't reside in the same bucket or sub-folder.

Objects added to a folder appear to reside within the folder in the Cloud Console. In reality, all objects exist at the bucket level, and simply include the directory structure in their name. For example, if you create a folder named `pets` and add a file `cat.jpeg` to that folder, the Cloud Console makes the file appear to exist in the folder. In reality, there is no separate folder entity: the file simply exists in the bucket and has the name `pets/cat.jpeg`.

When navigating folders in the Cloud Console, you can access higher levels of the directory by clicking the desired folder or bucket name in the breadcrumb trail above the file lists.

When you use other tools to work with your buckets and data, the presentation of folders may be different than as presented in the Cloud Console. For example, to see how gsutil interprets folders, see How Subdirectories Work (/storage/docs/gsutil/addlhelp/HowSubdirectoriesWork).

In the Cloud Console, you can filter the objects you see by specifying a prefix in the **Filter by prefix...** text box located above the list of objects. This filter displays objects beginning with the specified prefix. The prefix only filters objects in your current bucket view: it does not select objects contained in folders.

You can configure an object's metadata (/storage/docs/metadata) in the Cloud Console. Object metadata controls aspects of how requests are handled, including what type of content your data represents and how your data is encoded. Use the Cloud Console to set metadata on one object at a time. Use gsutil setmeta (/storage/docs/gsutil/commands/setmeta) to set metadata on multiple objects simultaneously.

See Viewing and Editing Object Metadata (/storage/docs/viewing-editing-metadata) for a step-by-step guide to viewing and editing an object's metadata.

You cannot set metadata on a folder.

You can delete any bucket, folder, or object in the Google Cloud Console by selecting the checkbox next to it, clicking the **Delete** button, and confirming you want to proceed with the action. When you delete a folder or bucket, you also delete all objects inside it, including any objects marked as **Public**.

See Deleting Objects (/storage/docs/deleting-objects) for a step-by-step guide to removing objects from your buckets using the Cloud Console.

See Deleting Buckets (/storage/docs/deleting-buckets) for a step-by-step guide to deleting buckets from your project using the Cloud Console.

When you share an object publicly, a link icon appears in the object's *public access* column. Clicking on this link reveals a public URL for accessing the object.

The public URL is different from the link associated with directly right-clicking on an object. Both links provide access but using the public URL works without having to sign into a Google account. See Request Endpoints age/docs/request-endpoints) for more information.

See Making Data Public (/storage/docs/access-control/making-data-public) for step-by-step guides to sharing your objects with others by making them publicly accessible.

See Accessing Public Data (/storage/docs/access-public-data) for ways to access a publicly shared object.

To stop sharing an object publicly:

You can stop publicly sharing an object by removing any permission entries that have **allUsers** or **allAuthenticatedUsers** as members.

- For buckets where you share only certain objects publicly, edit the ACL of the individual object (/storage/docs/access-control/create-manage-lists#changing-acls).

- For buckets where you share all objects publicly, remove the IAM access to allUsers (/storage/docs/access-control/using-iam-permissions#bucket-remove).

Both buckets and objects in the Cloud Console have a *public access* column that indicates when resources are shared publicly.

Bucket-level public access column

A bucket is considered public if it has an IAM role that meets these criteria:

- The role contains the member **allUsers** or **allAuthenticatedUsers**.
- The role has at least one storage permission (/storage/docs/access-control/iam-permissions) that is not `storage.buckets.create` or `storage.buckets.list`.

If these conditions are true, the public access column for the bucket reads **Public**.

If these conditions are not true, the public access column for the bucket reads **Per object**. This is because it's still possible that individual objects within the bucket are publicly accessible, depending on their individual Access Control Lists (ACLs) (/storage/docs/access-control/lists).

Object-level public access column

An object is considered public if either of these conditions are true:

1. The Access Control List (ACL) (/storage/docs/access-control/lists) for the object includes an entry for **allUsers** or **allAuthenticatedUsers**.
2. The bucket containing the object has an IAM role that meets these criteria:
    - The role contains the member **allUsers** or **allAuthenticatedUsers**.
    - The role has at least one of the following storage permissions (/storage/docs/access-control/iam-permissions): `storage.objects.get`, `storage.objects.getIamPolicy`, `storage.objects.setIamPolicy`, `storage.objects.update`.

If either of these conditions are true, the public access column for the object reads **Public**.

If none of these conditions are true, the public access column for the object reads **Not public**.

You can control access to a Cloud Storage bucket by using Identity and Access Management (IAM) permissions (/storage/docs/access-control/using-iam-permissions). For example, you can set a bucket's permissions to allow an entity such as a user or group to view or create objects in your bucket. You might do this in cases when it isn't appropriate to add a user at the project level. The entity specified in the IAM permission must authenticate by signing in to Google when accessing the bucket. Share the bucket URL with the user(s) as `https://console.cloud.google.com/storage/browser/[BUCKET_NAME]/`.

You can easily and uniformly control access to objects in a bucket by using Identity and Access Management (IAM) permissions (/storage/docs/access-control/iam) in the Cloud Console. If you want to customize access for individual objects within a bucket, use Signed URLs (/storage/docs/access-control/signed-urls) or Access Control Lists (ACLs) (/storage/docs/access-control/lists) instead.

See Using IAM Permissions (/storage/docs/access-control/using-iam-permissions) for step-by-step guides to viewing and editing IAM permissions.

To view or change permissions for individual objects, see Changing ACLs (/storage/docs/access-control/create-manage-lists#changing-acls).

You cannot set permissions on a folder.

When you create a project, you are given the **Owner** IAM role (/storage/docs/access-control/iam). Other entities, such as collaborators, must be given their own roles in order to work with your project's buckets and objects.

Once you have been given a role for the project, the project name appears in your list of projects. If you are an existing project owner, you can add a member to your project. See Using IAM with projects (/storage/docs/access-control/using-iam-permissions#project-add) for step-by-step guides to adding and removing access at the project level.

In general, set the minimum permission possible while still giving the team member the required access. For example, nember only needs to read objects stored in a project, select the **Storage Object Viewer** permission. Similarly, if the te er needs full control of objects, but not buckets, in a project, select **Storage Object Admin**.

Cloud Data Loss Prevention (Cloud DLP (/dlp)) is a service allowing you to identify and protect sensitive data in your buckets. Cloud DLP can help you meet compliance requirements by finding and redacting information such as:

- Credit card numbers
- IP addresses
- Other forms of personally identifiable information (PII)

For a list of the types of data Cloud DLP detects, see the Infotype detector reference (/dlp/docs/infotypes-reference).

You can initiate a Cloud DLP scan for a bucket by clicking the three-dot menu for the bucket and selecting **Scan with Cloud Data Loss Prevention**. For a guide to performing a Cloud DLP scan on a bucket, see Inspecting a Cloud Storage location (/dlp/docs/inspecting-storage#inspecting-gcs).