This tutorial describes how to configure a Cloud Storage bucket to host a static website for a domain you own. Static web pages can contain client-side technologies such as HTML, CSS, and JavaScript. They cannot contain dynamic content such as server-side scripts like PHP. For more information on static web pages, such as examples and tips, see the Static Website page (/storage/docs/static-website/).

This tutorial also applies to hosting static assets for a dynamic website.

Caution: This tutorial applies to using HTTP. We recommend that you don't serve content that contains sensitive or private data via HTTP from your Cloud Storage bucket. For options for serving content on a custom domain over HTTPS, see the related troubleshooting topic (/storage/docs/troubleshooting#https).

In this tutorial you will:

- Point your domain to Cloud Storage by using a `CNAME` record.
- Create a bucket that is linked to your domain.
- Upload and share your site's files.
- Test the website.

This tutorial uses the following billable component of Google Cloud:

- Cloud Storage

See the Monitoring your storage charges tip (/storage/docs/static-website#tip-charges) for details on what charges may be incurred when hosting a static website, and see the Pricing (/storage/pricing) page for details on Cloud Storage costs.

1. Sign in (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).

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)

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).

4. Have a domain that you own or manage. If you don't have an existing domain, there are many services through which you can register a new domain, such as Google Domains (https://domains.google.com/).

   This tutorial uses the domain `example.com`.

5. Verify that you own or manage the domain that you will be using (/storage/docs/domain-name-verification#verification). Make sure you are verifying the top-level domain, such as `example.com`, and not a subdomain, such as `www.example.com`.

   Note: If you own the domain you are associating to a bucket, you might have already performed this step in the past. If you purchased your domain through Google Domains, verification is automatic.

A `CNAME` record is a type of DNS record. It directs traffic that requests a URL from your domain to the resources you want to serve, in this case objects in your Cloud Storage buckets. For `www.example.com`, the `CNAME` record might contain the following information:

For more information about `CNAME` redirects, see URI for `CNAME` aliasing (/storage/docs/request-endpoints#cname).

To connect your domain to Cloud Storage:

1. Create a `CNAME` record that points to `c.storage.googleapis.com.`.

   Your domain registration service should have a way for you to administer your domain, including adding a `CNAME` record. For example, if you use Google Domains, instructions for adding a `CNAME` record can be found on the Google Domains Help page (https://support.google.com/domains/answer/9211383).

★ **Note:** DNS does not support creating a CNAME record on a root domain, such as **`example.com`**, so in most cases, you must create your **`CNAME`** record on a subdomain (/storage/docs/static-website#tip-subdomain), such as **`www.example.com`** or **`myblog.example.com`**. Some DNS providers offer a non-standard feature that allows a root domain to be aliased to another name without the use of **`CNAME`**. Consult your DNS provider documentation to determine if such a feature is available.

Create a bucket whose name matches the `CNAME` you created for your domain.

For example, if you added a `CNAME` record pointing `www.example.com` to `c.storage.googleapis.com.`, then create a bucket with the name "www.example.com".

To create a bucket:

To add to your bucket the files you want your website to serve:

Note: If you are migrating an existing website, or copying a large number of files, use `gsutil rsync` (/storage/docs/gsutil/commands/rsync) to move your content. You can use the `-R` option to recursively copy directory trees. For example, to synchronize a local directory named `local-dir` with a bucket, use:

You can either make all files in your bucket publicly accessible, or you can set individual objects to be accessible through your website. Generally, making all files in your bucket accessible is easier and faster.

- To make all files accessible, follow the Cloud Storage guide for making groups of objects publicly readable (/storage/docs/access-control/making-data-public#buckets).

Caution: Before making your bucket publicly accessible, make sure that the files in your bucket do not contain sensitive or private information.

- To make individual files accessible, follow the Cloud Storage guide for making individual objects publicly readable (/storage/docs/access-control/making-data-public#objects).

If you choose to control the accessibility of individual files, you can set the default object ACL (/storage/docs/access-control/create-manage-lists#defaultobjects) for your bucket so that subsequent files uploaded to your bucket are shared by default.

Note: Visitors receive a **`http 403`** response code when requesting the URL for a non-public or non-existent file. See the next section for information on how to add an error page that uses a **`http 404`** response code.

You can assign an index page suffix, which is controlled by the `MainPageSuffix` property and a custom error page, which is controlled by the `NotFoundPage` property. Assigning either is optional, but without an index page, nothing is served when users access your top-level site, for example, `http://www.example.com`.

An index page (also called a webserver directory index (https://wikipedia.org/wiki/Webserver_directory_index)) is a file served to visitors when they request a URL that doesn't have an associated file. When you assign a `MainPageSuffix`, Cloud Storage looks for a file with that name whose prefix matches the URL the visitor requested.

For example, say you set the `MainPageSuffix` of your static website to `index.html`. Additionally, say you have no file named `directory` in your bucket `www.example.com`. In this situation, if a user requests the URL `http://www.example.com/directory`, Cloud Storage attempts to serve the file `www.example.com/directory/index.html`. If that file also doesn't exist, Cloud Storage returns an error page.

The `MainPageSuffix` also controls the file served when users request the top level site. Continuing the above example, if a user requests `http://www.example.com`, Cloud Storage attempts to serve the file `www.example.com/index.html`.

For more information on the cases in which the index page is served, see Website configuration examples (/storage/docs/static-website#examples).

The error page is the file returned to visitors of your static site who request a URL that does not correspond to an existing file. If you have assigned a `MainPageSuffix`, Cloud Storage only returns the error page if there is neither a file with the requested name nor an applicable index page.

When returning an error page, the http response code is `404`. The property that controls which file acts as the error page is `NotFoundPage`. If you don't set `NotFoundPage`, users receive a generic error page.

In the following sample, the `MainPageSuffix` is set to `index.html` and `NotFoundPage` is set to `404.html`:

Verify that content is served from the bucket by requesting the domain name in a browser. You can do this with a path to an object or with just the domain name, if you set the `MainPageSuffix` property.

For example, if you have an object named `test.html` stored in a bucket named `www.example.com`, check that it's accessible by going to `www.example.com/test.html` in your browser.

After you've finished the Hosting a Static Website tutorial, you can clean up the resources that you created on GCP so they won't take up quota and you won't be billed for them in the future. The following sections describe how to delete or turn off these resources.

The easiest way to eliminate billing is to delete the project that you created for the tutorial.

To delete the project:

> **Caution**: Deleting a project has the following effects:
>
> - **Everything in the project is deleted.** If you used an existing project for this tutorial, when you delete it, you also delete any other work you've done in the project.
> - **Custom project IDs are lost.** When you created this project, you might have created a custom project ID that you want to use in the future. To preserve the URLs that use the project ID, such as an **`appspot.com`** URL, delete selected resources inside the project instead of deleting the whole project.
>
> If you plan to explore multiple tutorials and quickstarts, reusing projects can help you avoid exceeding project quota limits.

1. In the Cloud Console, go to the **Manage resources** page.

   Go to the Manage resources page (https://console.cloud.google.com/iam-admin/projects)

2. In the project list, select the project you want to delete and click **Delete** 🗑.
3. In the dialog, type the project ID, and then click **Shut down** to delete the project.

If you do not want to delete the entire project, delete the bucket that you used for hosting your website:

- See examples and tips for using buckets to host a static website (/storage/docs/static-website/).
- Visit the troubleshooting section for hosting a static website (/storage/docs/troubleshooting#static-website).
- Learn about hosting static assets for a dynamic website (/storage/docs/static-website#tip-dynamic).
- Go more in-depth with the Cloud Storage Office Hours for hosting a static website (https://www.youtube.com/watch?feature=player_embedded&v=dRhJG0z4qVE).
- Learn about all web serving options (/solutions/web-serving-overview).
- Try other Google Cloud tutorials that use Cloud Storage (/docs/tutorials#%22cloud%20storage%22).