Bottom of Form

Creating Project Pages using the command line

If you're familiar with command-line Git, it's straightforward to manually create a Project Pages site.

**Make a fresh clone**

To set up a Project Pages site, you need to create a new "orphan" branch (a branch that has no common history with an existing branch) in your repository. The safest way to do this is to start with a fresh clone:

git clone https://github.com/*user*/*repository*.git

# Clone our repository

Cloning into '*repository*'...

remote: Counting objects: 2791, done.

remote: Compressing objects: 100% (1225/1225), done.

remote: Total 2791 (delta 1722), reused 2513 (delta 1493)

Receiving objects: 100% (2791/2791), 3.77 MiB | 969 KiB/s, done.

Resolving deltas: 100% (1722/1722), done.

**Create a master branch**

Once you have a clean repository, you'll need to create a new master branch unless your cloned repository already has a master branch.

**Tip**: You can also create a gh-pages branch for your Project Pages site instead. To learn more about your options, including the option to publish your Project Page from a /docs folder on your master branch, see "[User, Organization, and Project Pages](https://help.github.com/articles/user-organization-and-project-pages/#project-pages)."

1. Switch directories into your new cloned repository:
2. cd *repository*
3. Check if your repository already has a master branch:
4. git branch
5. # shows a list of branches for your repository
6. \* branch-name
7. \* branch-name
8. If you don't already have a master branch in your repository, create a new master branch:
9. git checkout --orphan master
10. # Creates a master branch, without any parents (it's an orphan!)
11. Switched to a new branch 'master'

**Tip**: If you just created the master branch, it won't appear in your list of branches when you use the git branch command until you make your first commit.

**Remove all files to create an empty working directory**

Once you have a master branch, you'll need to remove all content from the working directory and index:

git rm -rf .

# Remove all files from the old working tree

rm '.gitignore'

**Add content and push**

In order to trigger a build when you push to your Page's repository, you must first [verify your email address](https://help.github.com/articles/verifying-your-email-address).

Now you have an empty working directory. You can create some content in this branch and push it to GitHub. For example:

echo "My Page" > index.html

git add index.html

git commit -a -m "First pages commit"

git push origin master

Your GitHub Pages site should now be available. You'll receive an email if your build is unsuccessful.

**Load your new GitHub Pages site**

After your push to the master branch, your Project Pages site will be available at 'http(s)://<username>.github.io/<projectname>' . Note that Pages are always publicly accessible when published, even if their repository is private.

To set up a custom domain for GitHub Pages sites, see "[Setting up a custom domain with GitHub Pages](https://help.github.com/articles/setting-up-a-custom-domain-with-github-pages)."