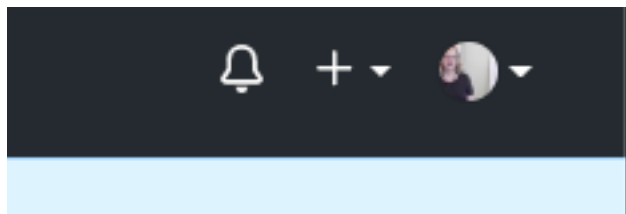


Create a GitHub Portfolio for Class Work Using GitHub Pages

A large portion of the class grade will be the work you upload to your portfolio. This will be a combination of code, code narrative, reports, and presentations. The three steps to set up and create the portfolio are described below.

Step 1: Set up a GitHub account (if you don't have one) and a repo for your classwork

- Go to www.github.com and set up a username and password. The basic account is free.
- Optional: set up your profile so that potential employers can find you. There is an option to show your contributions, and another one to check if you are available for hire. There are many other interesting options to read through.
- Install GitHub Desktop on your computer. If you are familiar with git commands, you can just use the command line options, but GitHub desktop makes uploading and modifying your work easier. Google Desktop: <https://desktop.github.com/>
- Create a repository specifically for this class. Click the + at the upper right and select "New repository".




- Give it a name, description, and make it public. Also check that you want to set up a README, a gitignore and a License. The license you choose is up to you. This screen looks like this:

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository](#).

Owner *

 kjmazidi ▾

Repository name *

Sample_Portfolio ✓

Great repository names are short and memorable. Need inspiration? How about [didactic-waddle?](#)

Description (optional)

This is a sample portfolio

☒  **Public**

Anyone on the internet can see this repository. You choose who can commit.

☐  **Private**

You choose who can see and commit to this repository.

Initialize this repository with:

Skip this step if you're importing an existing repository.

☒ **Add a README file**

This is where you can write a long description for your project. [Learn more](#).

Add .gitignore


Choose which files not to track from a list of templates. [Learn more](#).

.gitignore template: None ▾

Choose a license

A license tells others what they can and can't do with your code. [Learn more](#).

License: GNU General Public... ▾

This will set  **main** as the default branch. Change the default name in your [settings](#).

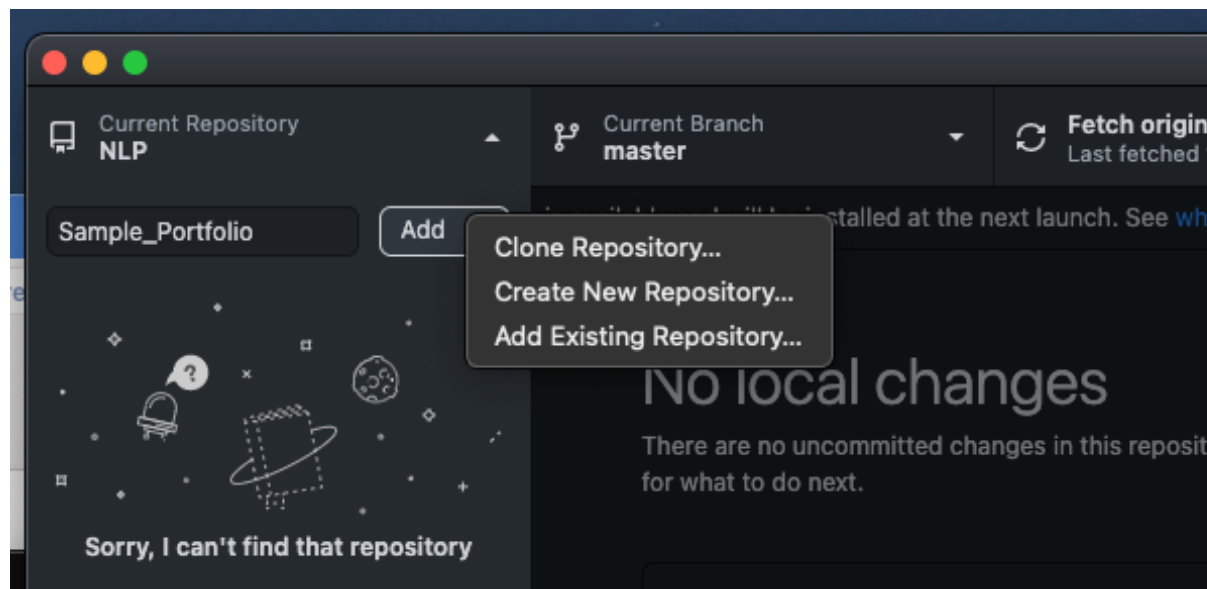
 You are creating a public repository in your personal account.

Create repository

Click Create when you are ready

You will see the repo with the README.md file.

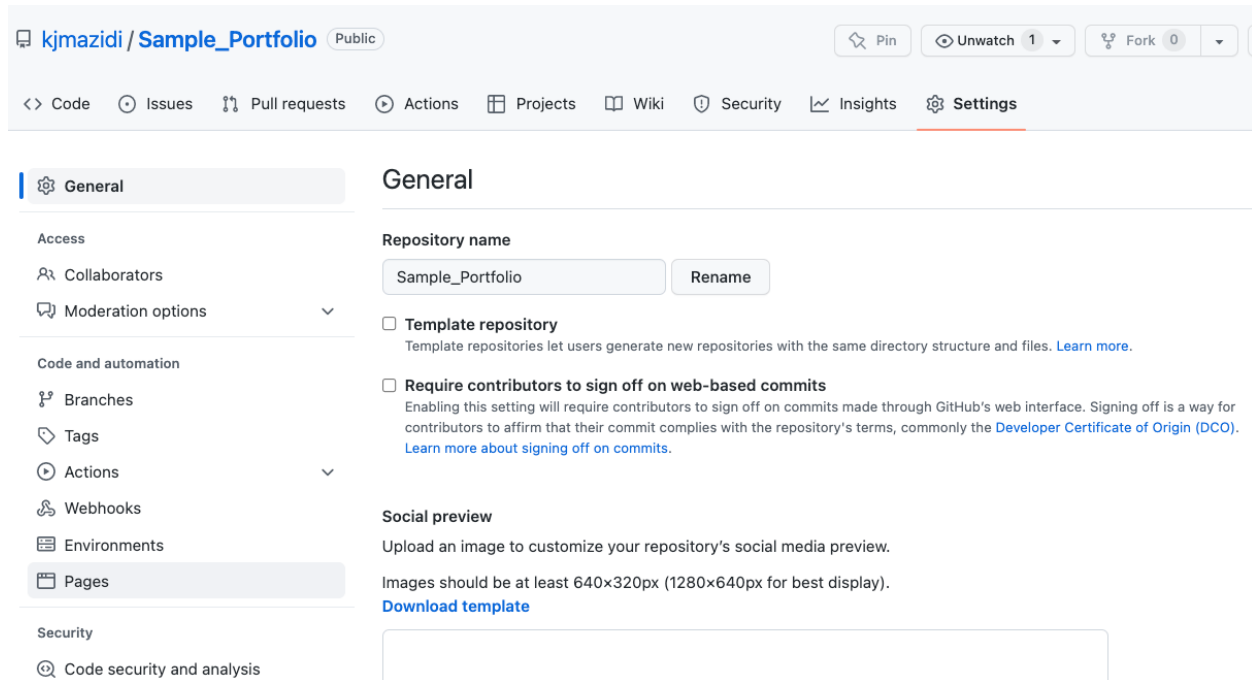
- Now that you have a repo, sync this with your GitHub Desktop app.
- In the left panel of GitHub Desktop, type in the name and click the arrow to the right of add. Choose Clone Repository.
- GitHub may take you to a second screen where you type in the name again.



Step 2: Enable GitHub Pages for this repo

1. Enable GitHub Pages in your class repo. Go to your repo on github.com. Select settings for this repo and then click “Pages” in settings or in the sidebar

This screen shot is showing Settings for the repo. Select Pages.



Once you click “Pages” you end up on the screen below.

1. Source – choose Deploy from a branch
2. Branch – choose main

GitHub Pages

GitHub Pages is designed to host your personal, organization, or project pages from a GitHub repository.

Build and deployment


Source

Deploy from a branch ▾

Branch

GitHub Pages is currently disabled. Select a source below to enable GitHub Pages for this repository. [Learn more.](#)

 main ▾

 / (root) ▾

Save



Publish privately to people with read access to this repository

Try risk-free for 30 days using a GitHub Enterprise organization, or [learn more about changing the visibility of your GitHub Pages site.](#) ✕

Click Save.

You are then taken to a page that lets you specify a custom domain. Ignore that.

If you go page to Settings -> Pages for your repo, you will see that GitHub Pages is live at your address. Mine was:

https://kjmazidi.github.io/Sample_Portfolio/

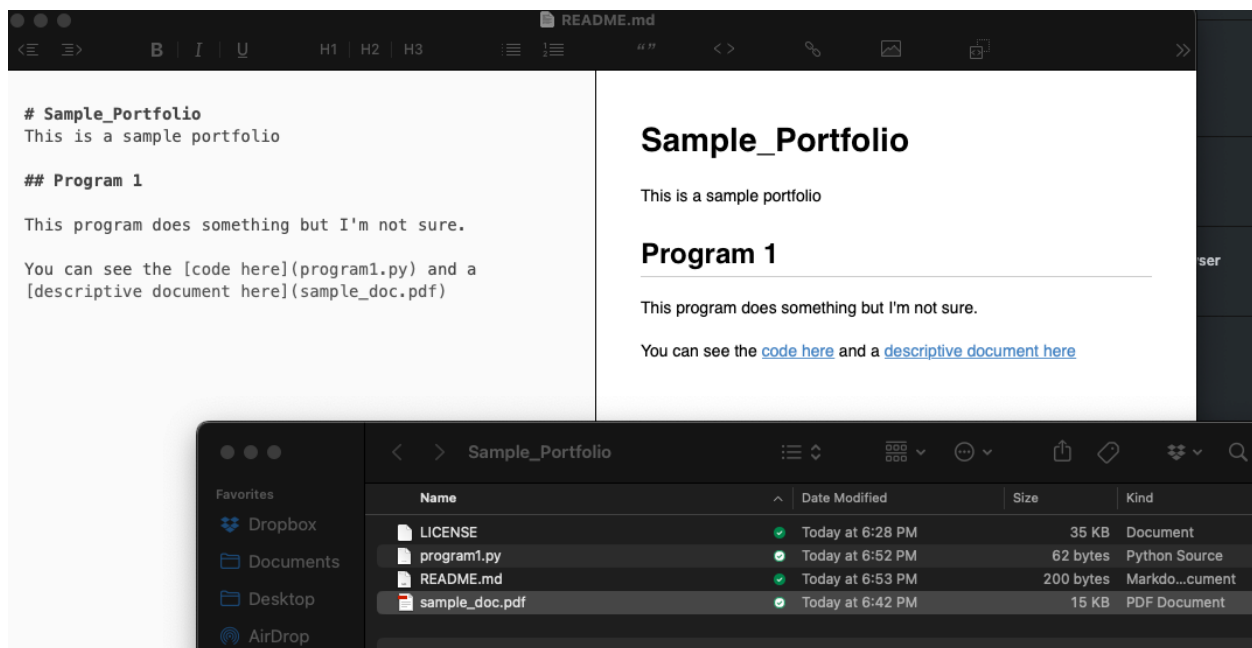
If you go back to your repo on GitHub you will see your README.md which is serving as the landing page for your GitHub pages.

Step 3: Organize and upload material as follows

1. As you add material, provide a link to it on your main page (the README)

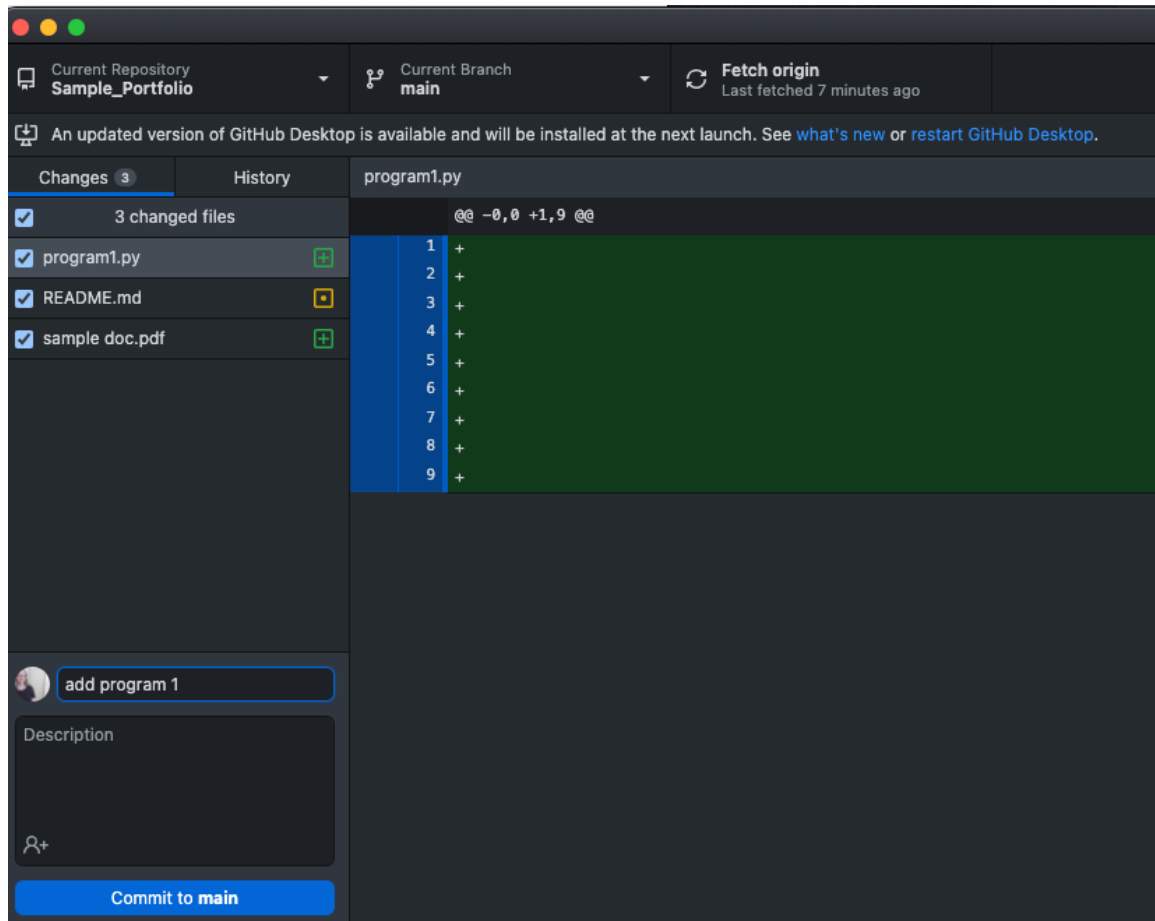
I think the easiest way is to add files to your local repo. Then push to origin in GH Desktop.

This screen shot shows the files I added to the repo in the dark box, and above that a markdown editor I use on the Mac called MacDown to edit the README. You can use a simple text editor as well. MacDown and similar editors show the markdown code on the left and how it will render on the right.

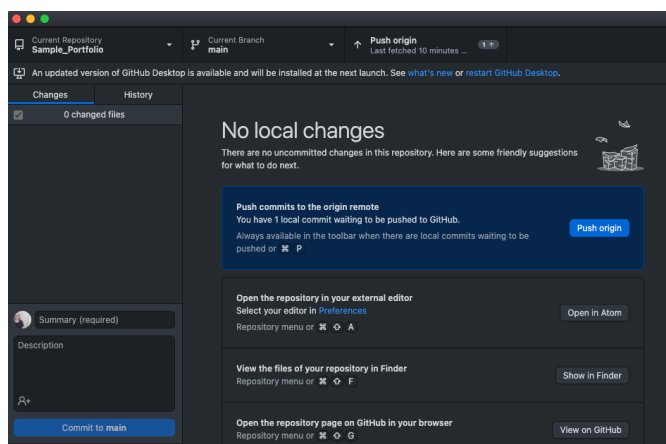


Now I'm ready to push these changes to GitHub. See next page.

Here is a reference sheet for markdown: <https://www.markdownguide.org/cheat-sheet/>



The screenshot above shows GH Desktop telling me that I've made changes to the local repo. It wants a comment. I wrote "add program 1". Now all I have to do is click the blue "Commit to main", then the Push to Origin button which will pop up, as shown below:



Now when I go back to my GitHub repo online, I see this:

The screenshot shows a GitHub repository page for 'kjmazidi / Sample_Portfolio'. The repository is public. The navigation bar includes links for Code, Issues, Pull requests, Actions, Projects, Wiki, Security, Insights, and Settings. Below the navigation bar, there are buttons for 'main' branch, '1 branch', and '0 tags'. There are also buttons for 'Go to file', 'Add file', and 'Code'. The commit history shows a commit by 'kjmazidi' titled 'update program 1' with commit hash 'e0a79ca' and '20 seconds ago'. The commit message is 'update program 1'. The commit includes four files: 'LICENSE' (Initial commit, 30 minutes ago), 'README.md' (update program 1, 20 seconds ago), 'program1.py' (update program 1, 20 seconds ago), and 'sample_doc.pdf' (update program 1, 20 seconds ago). Below the commit history, the 'README.md' file is displayed. The README content includes the title 'Sample_Portfolio', a description 'This is a sample portfolio', a section 'Program 1', and a description 'This program does something but I'm not sure.' followed by a link to the code and a link to the descriptive document.

kjmazidi / Sample_Portfolio Public

< > Code Issues Pull requests Actions Projects Wiki Security Insights Settings

main 1 branch 0 tags Go to file Add file Code

kjmazidi update program 1 e0a79ca 20 seconds ago 3 commits

LICENSE	Initial commit	30 minutes ago
README.md	update program 1	20 seconds ago
program1.py	update program 1	20 seconds ago
sample_doc.pdf	update program 1	20 seconds ago

README.md

Sample_Portfolio

This is a sample portfolio

Program 1

This program does something but I'm not sure.

You can see the [code here](#) and a [descriptive document here](#)

Notes:

1. Remember that assignments are uploaded to eLearning for the TA to grade, as well as being uploaded here for the portfolio.
2. GitHub does not play nicely with file names with spaces so use underscores.