

Basic Git Tutorial

Using GitHub Desktop

Baltimore Indie Game Devs – Updated: 12 December 2020

A version control system stores versions of your project's files on a server. Advantages:

- It's a backup of your project.
- You can revert to older versions if broke something in your project.
- Multiple people can share their changes.

Git is a version control system. It's powerful. It's free. It's also frustratingly complex to learn.

Git is a command line tool. This tutorial covers absolute minimum you need to know to use git through a GUI interface called GitHub Desktop. By default, GitHub Desktop assumes you'll be using it with GitHub (a version control service that offers free private repositories), but it works with any git server. This tutorial will use GitHub's service.

Step 1: Create an account on GitHub.

github.com/join

Join GitHub

Create your account

Username *

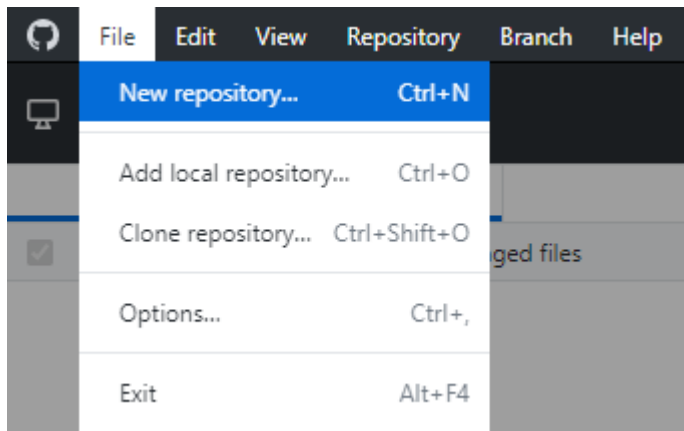
Email address *

Password *

Step 2: Download and install GitHub Desktop.

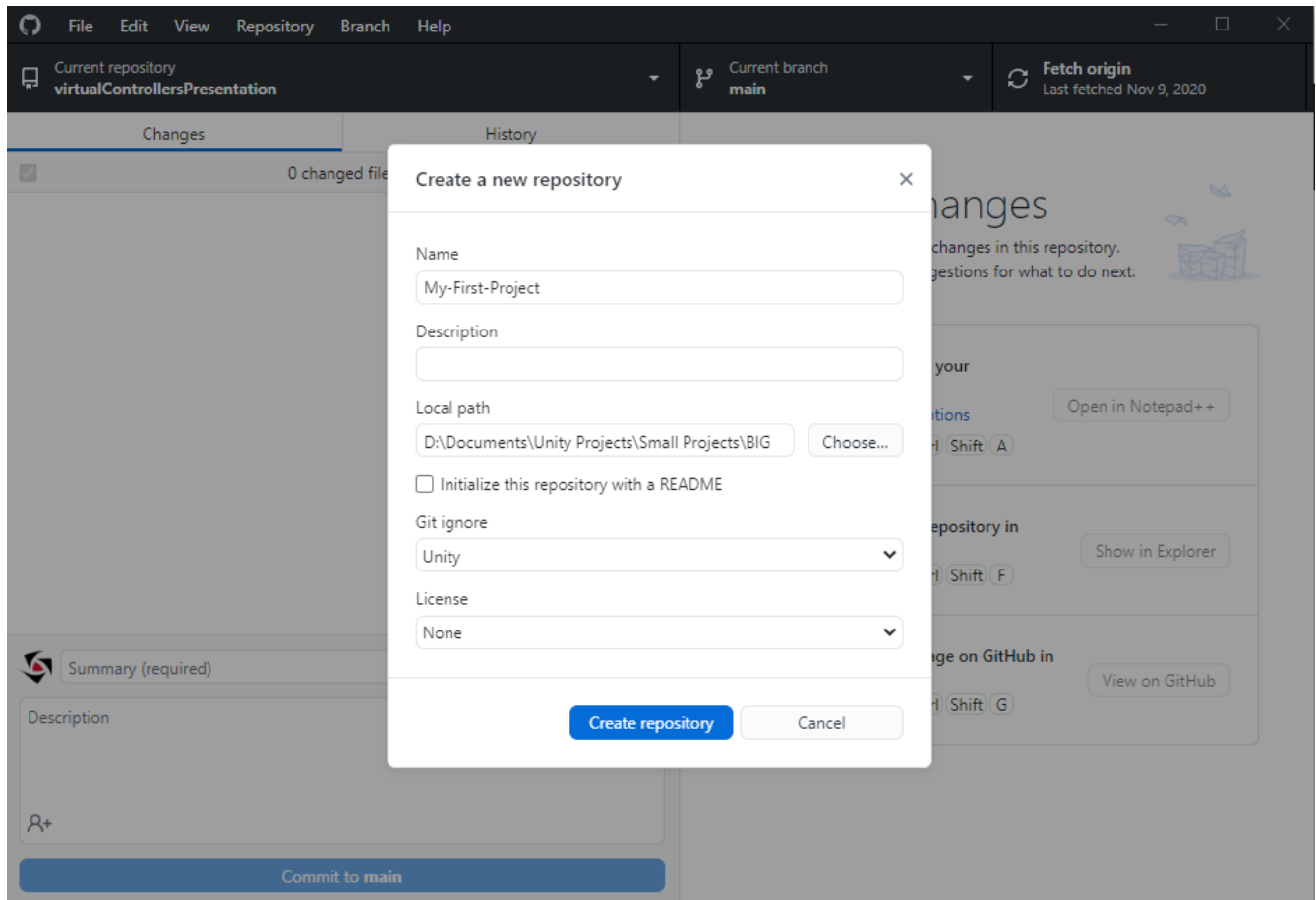


Step 3: Run GitHub Desktop. Select File > New Repository:

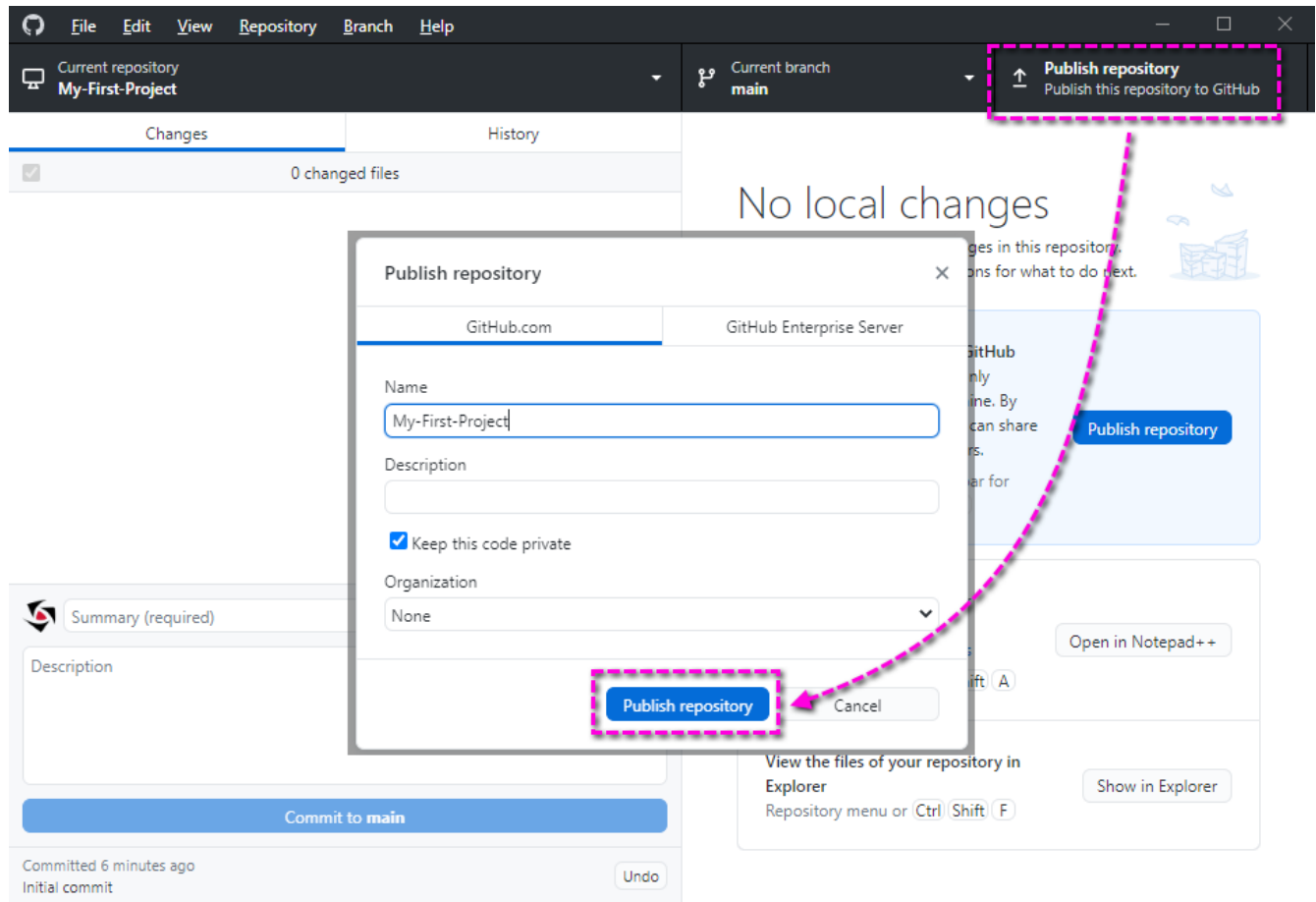


Step 4: Enter a name for your new repository and select where you want to store the local copy on your computer. You can optionally specify a *Git ignore* definition. This will tell git which files in your project should *not* be uploaded to the version control server. If you're making a Unity project, set the *Git ignore* dropdown to Unity.

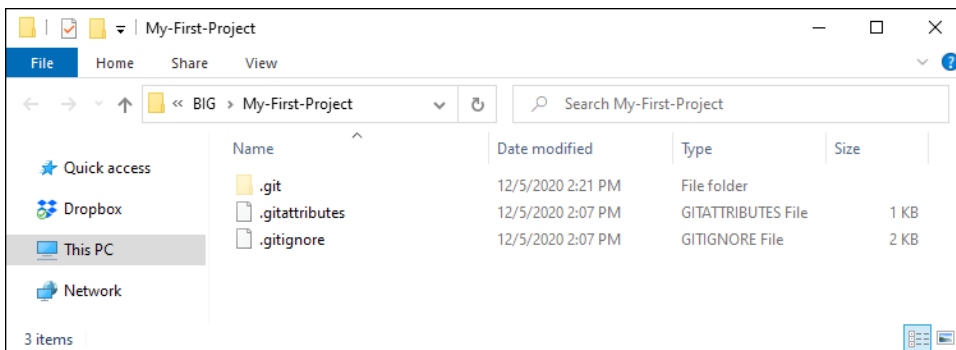
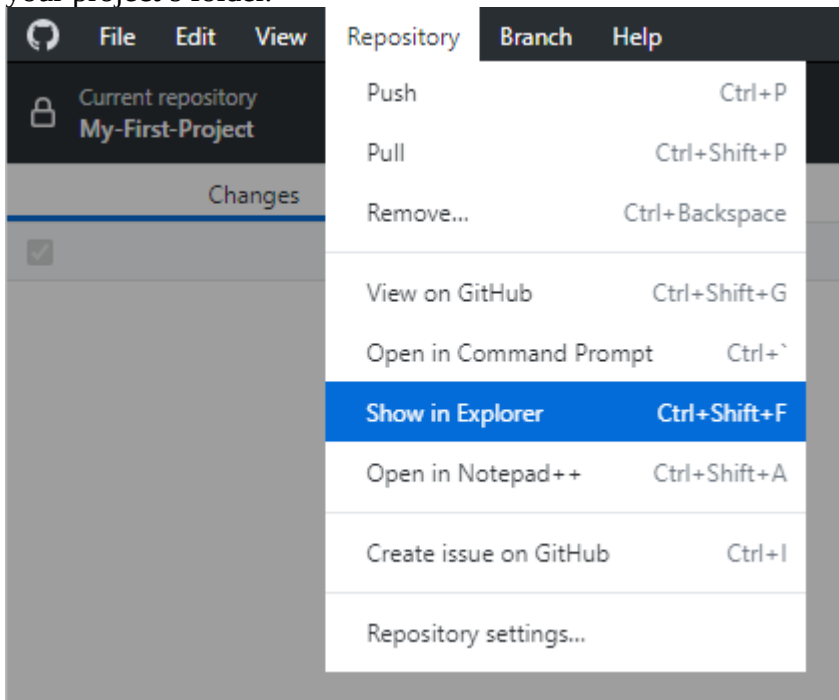
Then click *Create Repository*. This will create an empty local version on your computer containing just a few setup files such as a file named `.gitignore` if you selected something in the *Git ignore* dropdown.



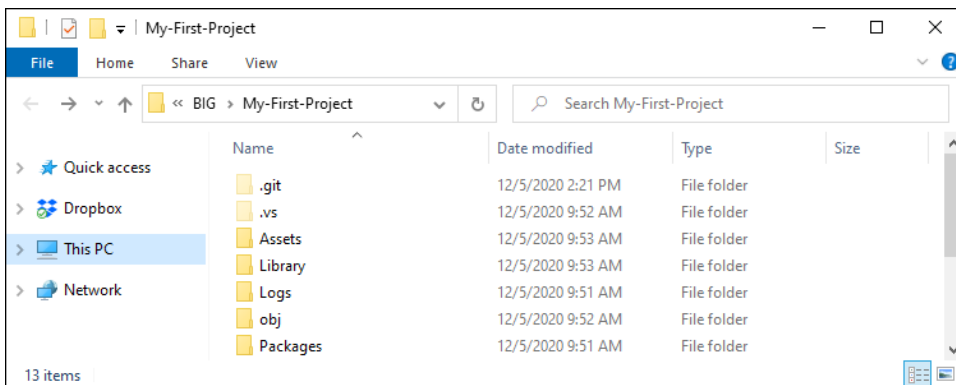
Step 5: Click *Publish repository*. It will open a window in which you can specify whether the project should be kept private or shared with the world. In this window, click *Publish repository*. This will upload your local copy to GitHub.



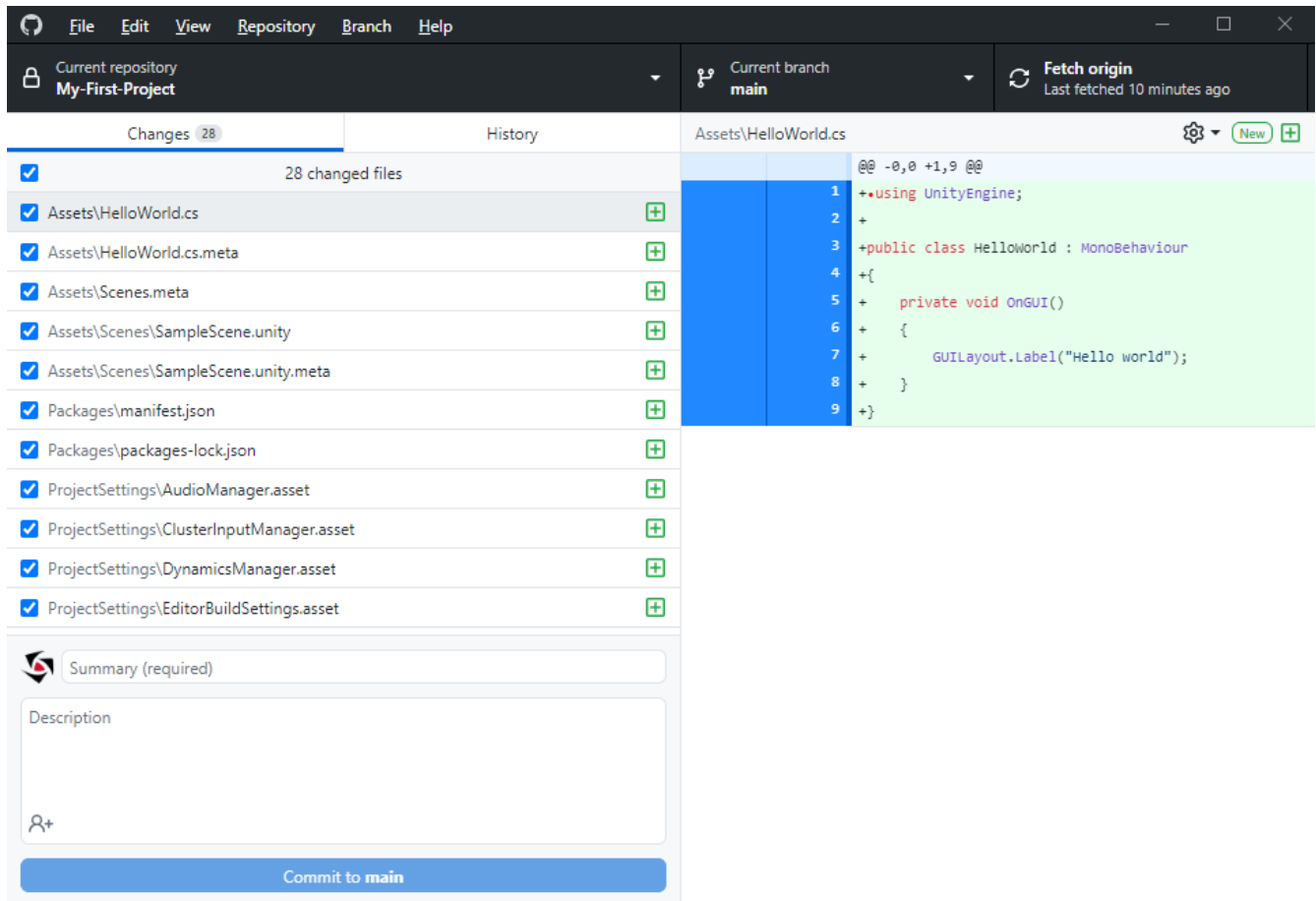
Step 6: Select Repository > Show in Explorer (or equivalent). This will open an OS file window in your project's folder.



Add files to your project. For Unity projects, you may find it easiest to create an empty project in another folder and then move that folder's contents into this folder.



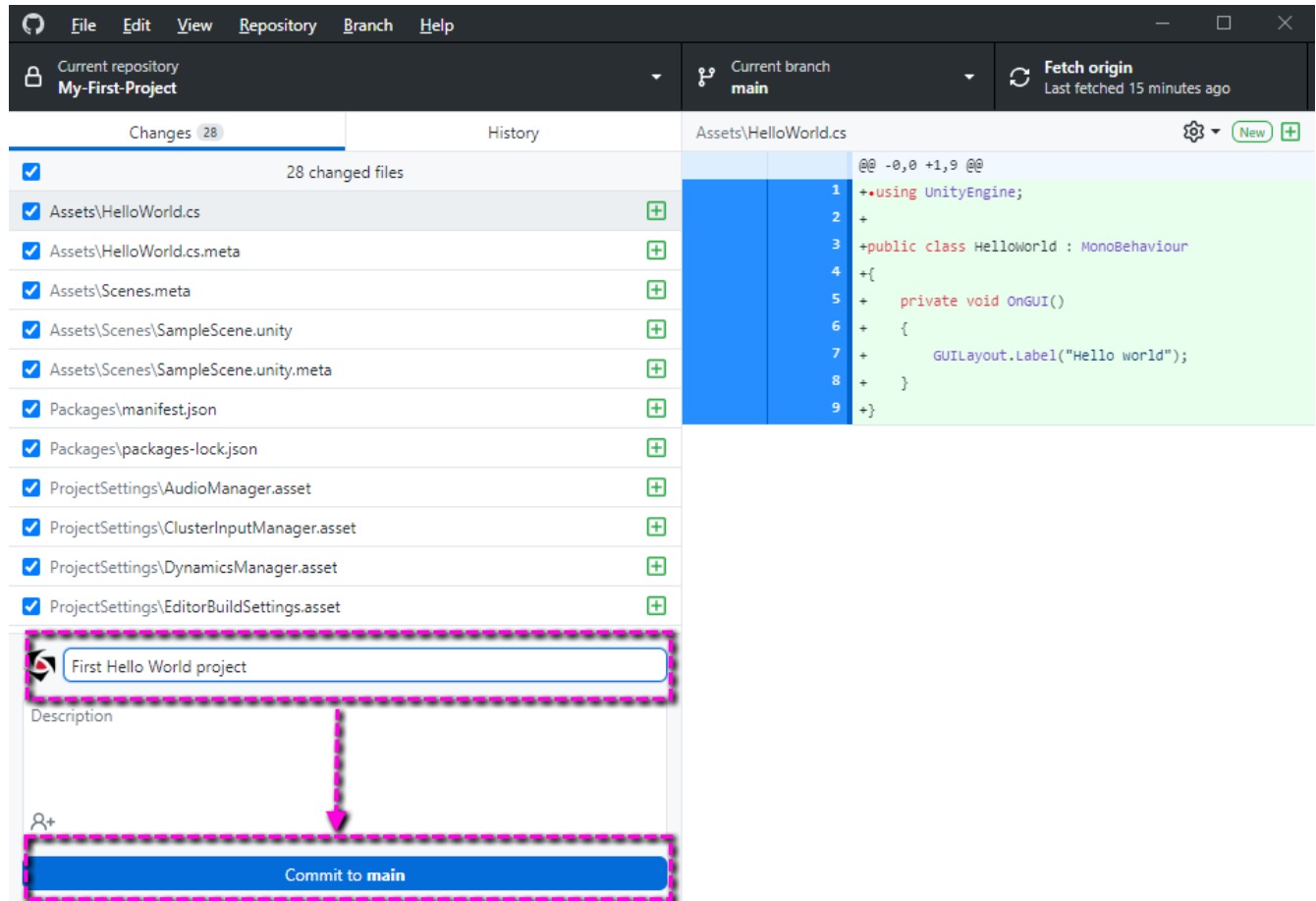
Step 7: Return to the GitHub Desktop window. The files that you added (and later, changed or deleted) will appear in the left pane. If you click on a file, the right pane will show the changes since the last time you committed changes to the server.



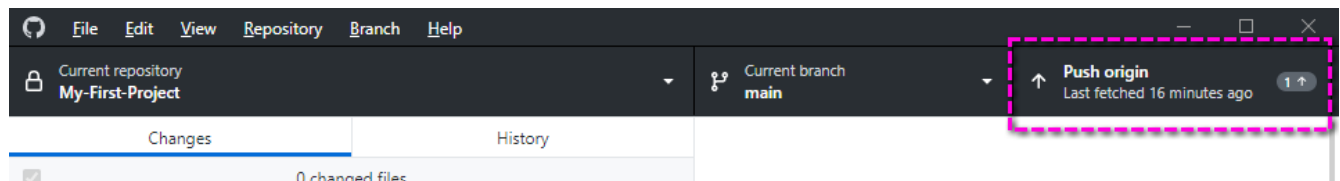
Step 8: There are two steps to upload files to the server:

1. *Commit* the file changes, which marks the changes as ready to be uploaded.
2. *Push* the changes up to the server.

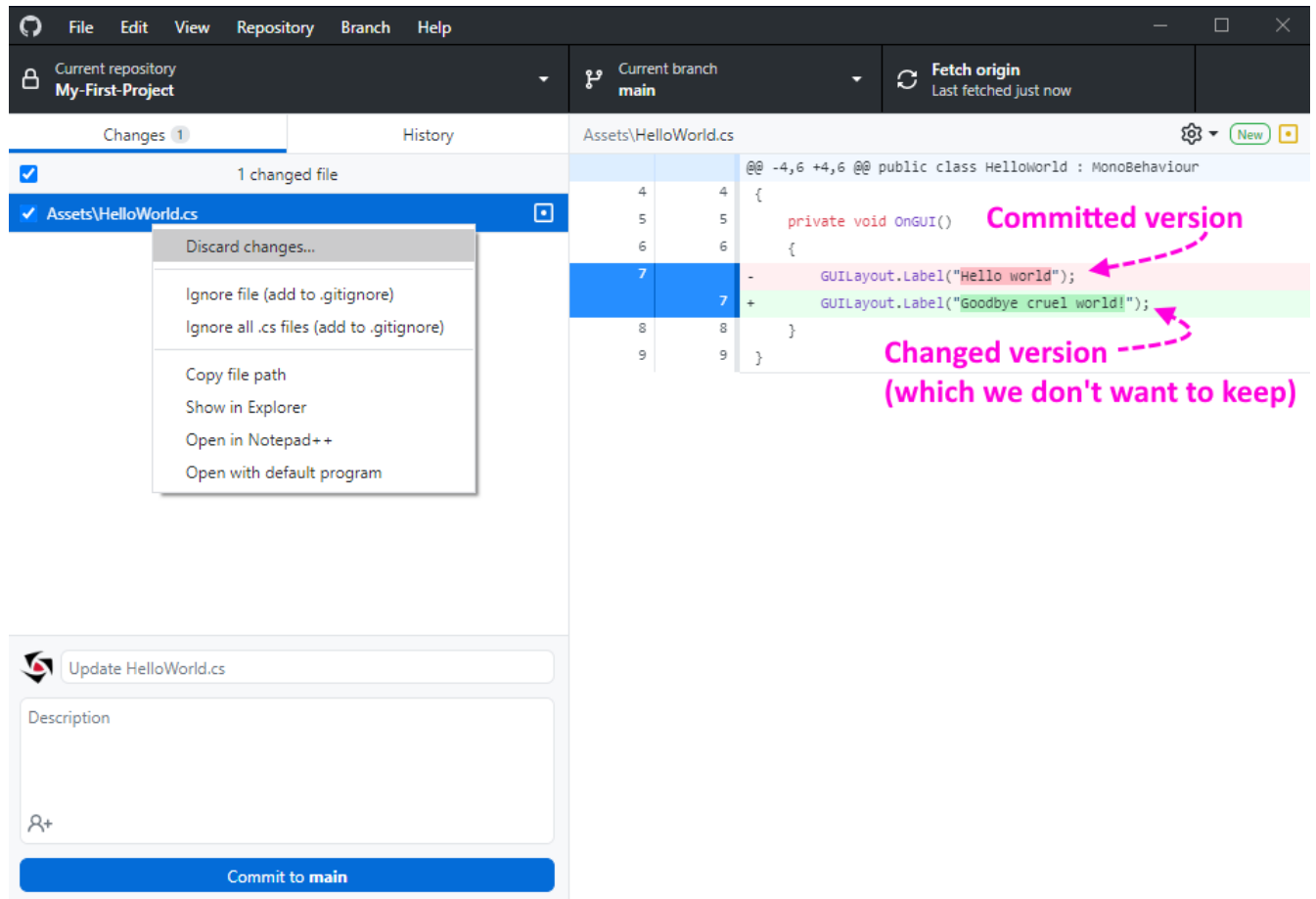
Enter a summary that describes the changes. Then click *Commit to main*.



Finally, click *Push origin*. This will upload your changes to the server.

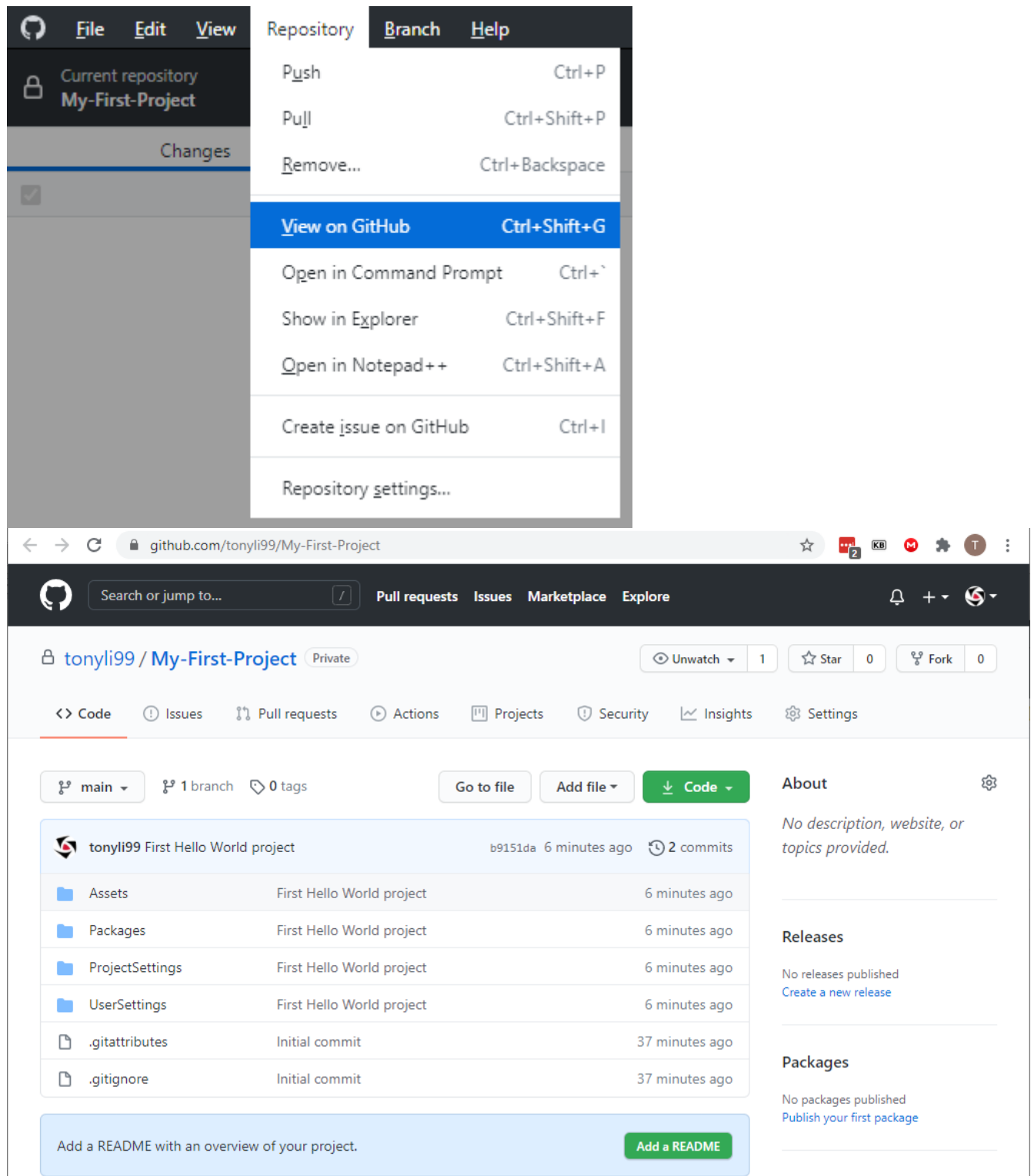


Step 9: If you make a change that you don't want to keep, right-click on the file(s) and select *Discard changes*.



Give Access To Others

Step 1: To give someone else access, select Repository > View on GitHub.



Step 2: Click Settings > Manage access, then Invite a collaborator. Enter their email address or GitHub account name.

The screenshot shows the GitHub repository settings page for 'tonyli99 / My-First-Project'. The repository is private. The 'Settings' tab is selected in the top navigation bar. On the left, the 'Options' menu is open, and 'Manage access' is highlighted. The main content area is titled 'Who has access' and shows two access types: 'PRIVATE REPOSITORY' (locked) and 'DIRECT ACCESS' (unlocked). The 'PRIVATE REPOSITORY' section states 'Only those with access to this repository can view it.' and has a 'Manage' link. The 'DIRECT ACCESS' section states '0 collaborators have access to this repository. Only you can contribute to this repository.' Below this, the 'Manage access' section is titled, and a message says 'You haven't invited any collaborators yet' with an 'Invite a collaborator' button.

tonyli99 / My-First-Project Private

Unwatch 1 Star 0 Fork 0

Code Issues Pull requests Actions Projects Security Insights Settings

Options

Manage access

Security & analysis

Branches

Webhooks

Notifications

Integrations

Deploy keys

Actions

Secrets

Who has access

PRIVATE REPOSITORY

Only those with access to this repository can view it.

Manage

DIRECT ACCESS

0 collaborators have access to this repository. Only you can contribute to this repository.

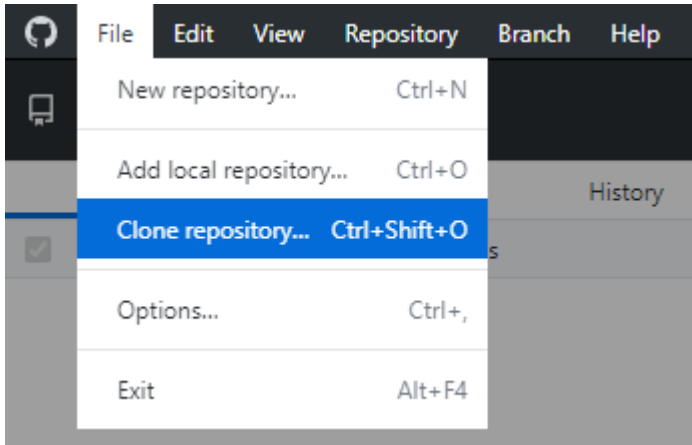
Manage access

You haven't invited any collaborators yet

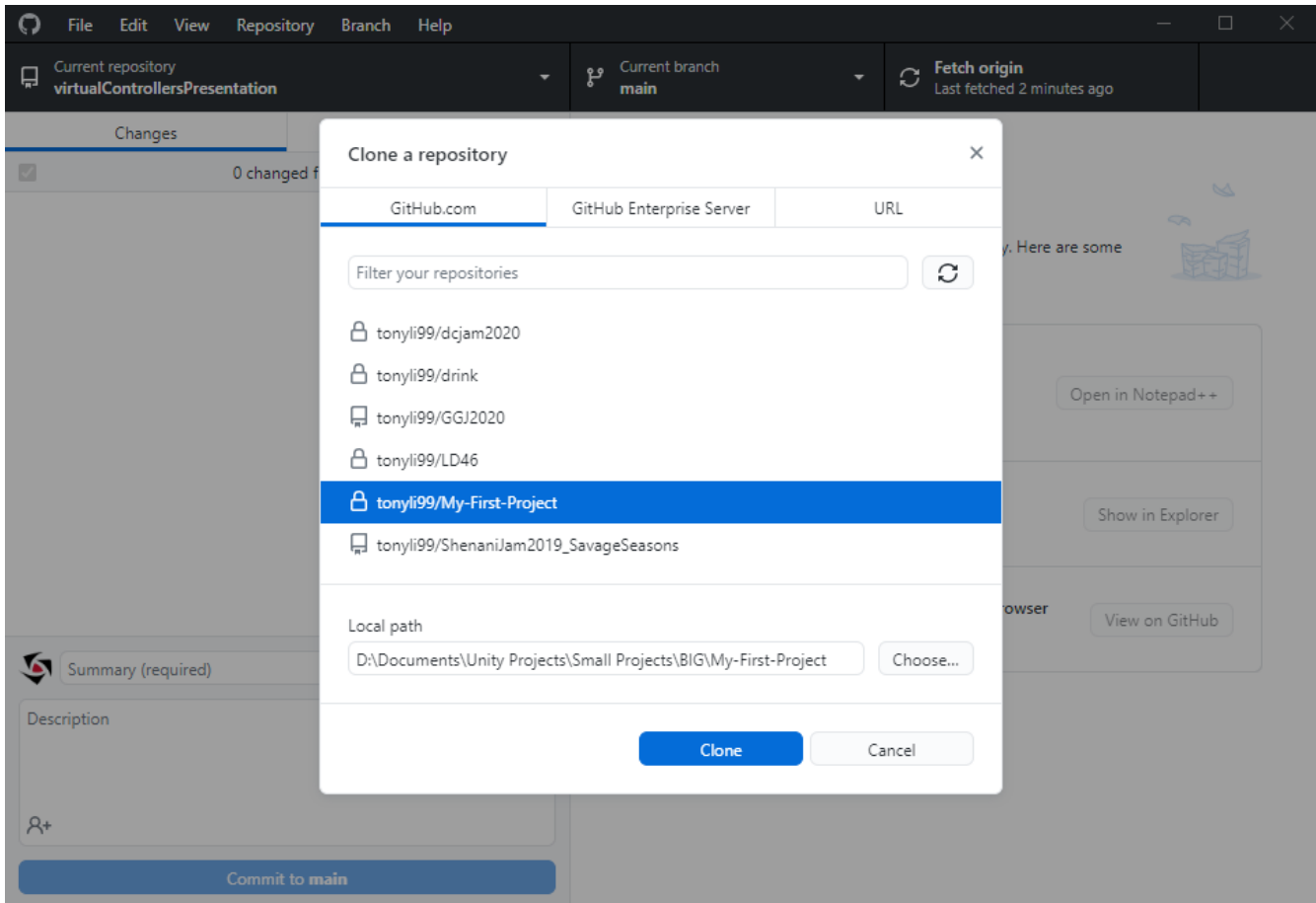
Invite a collaborator

Join An Existing Repository

Step 1: In GitHub Desktop, select File > Clone repository.



If the repository is hosted on GitHub and you have already been given access, you should be able to pick it from a list.



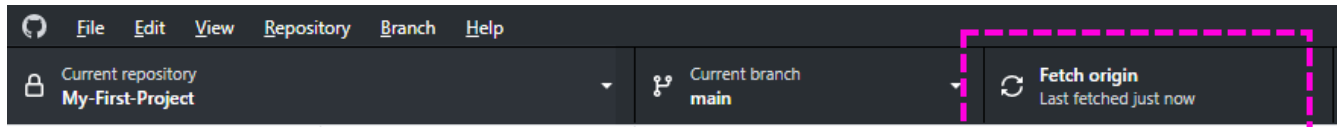
Otherwise you will need the repository's URL, such as:

`https://github.com/tonyli99/My-First-Project.git`

Click URL and enter the URL.

In either case, specify the Local path where you want to download a copy of the project, and click Clone.

Whenever a teammate makes a change, click this area twice to pull down the changes:



Whenever you make changes, follow these steps:

1. Click that area twice to pull down any changes that your teammates have already pushed up.
2. Enter a summary.
3. Click Commit to main.
4. Click the same area to push up your changes.

Final Notes

Git does so much more than what's covered in this document. But this should be enough to get you started with basic version control and collaborating with others.