# Learning git and github for LESGO users

1 Why do we use git?

- Full version history. Did you make a mistake or want to go back and look at previous versions of the code? It's all right there in the repository. This also serves as a kind of backup.

- Collaboration. Git makes it easy to save changes and share it with others.

- Branching. You can work on several new features at the same time using different branches. When you're done, you can easily merge the new feature branch into the main code.

2 Using the fork and pull workflow with Github using the learn-git repository

1. Navigate to the main repository https://github.com/lesgo-jhu/learn-git. Feel free to play around with this repository. It's a playground to learn what you're doing.

2. Fork your own copy of the repository to your account. This will give you a new repository where you can make changes without affecting the main source code used by others.
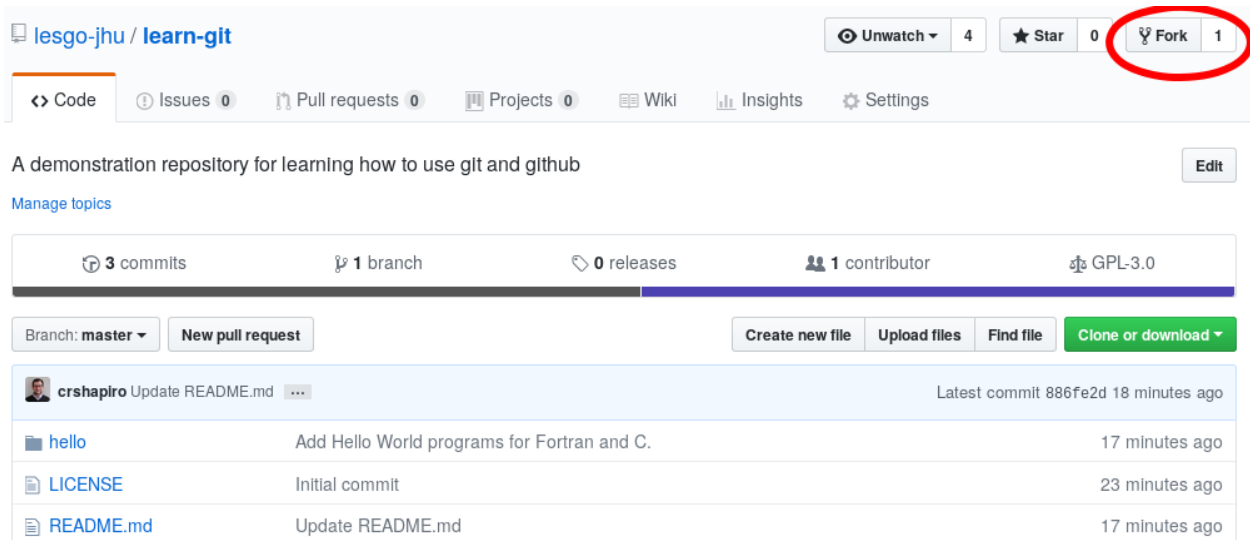
3. Clone your new repository to your local machine, where you will do development

```
git clone https://github.com/crshapiro/learn-git.git
```

4. By default, your personal server-side repository is called "origin." We also need to tell git where the official server-side repository is, which we'll call "upstream."

```
git remote add upstream https://github.com/lesgo-jhu/learn-git.git
```

You can always see the url of "origin" and "upstream" using

```
git remote show origin
git remote show upstream
```

5. Now, we can start making some changes. Write a "Hello World" program in your favorite language. Once this is complete, you can commit the changes:

```
git add (list for files)
```

To see the status of the added files use:

```
git status
```

Now commit the changes to your local repository

```
git commit -m "Enter a concise message detailing changes."
```

6. The edits you have made are still only on your local machine. To copy the changes to your server-side repository (origin) use

```
git push
```

This will push to the first remote repository listed in `git remote`.

Starting August 2021, performing HTTPS git operations on command line also require authentication using a *personal access token (PAT)*. Follow the steps in `https://docs.github.com/en/authentication/keeping-your-account-and-data-secure/creating-a-personal-access-token` to create a token. If the command line is not already authenticated, execute `git push` with PAT as follows:

```
git push https://<GITHUB_ACCESS_TOKEN>@github.com/<GITHUB_USERNAME>/<
    REPOSITORY_NAME>.git
```

7. Once you're satisfied with your revisions and want to integrate into the official repository, you need to initiate a new pull request. (You're requesting that the official repository pull your changes). Follow the instructions on screen to check that everything merges fine. If there are issues, you should merge the official repository into your server-side repository before making the pull request. If everything is ok, create the pull request.

Your changes will now show up as a pull request on the official repository waiting for approval from the maintainers.

lesgo-jhu / **demo**

👁 Unwatch ▾ 4   ★ Star 0   ⑂ Fork 1

‹› Code   ⓘ Issues **0**   Pull requests **1**   Ⅲ Projects **0**   ▤ Wiki   Ⅲ Insights   ⚙ Settings

Label issues and pull requests for new contributors   Dismiss
Now, GitHub will help potential first-time contributors discover issues
labeled with `help wanted` or `good first issue`

Filters ▾   🔍 is:pr is:open   Labels   Milestones   **New pull request**

☐ 1 Open ✓ 0 Closed   Author ▾   Labels ▾   Projects ▾   Milestones ▾   Reviews ▾   Assignee ▾   Sort ▾

☐ **Add fortran Hello World program.**
#1 opened a minute ago by crshapiro

💡 **ProTip!** What's not been updated in a month: updated:<2018-12-29.

8. Now, you need to update origin to be even with the changes in upstream

```
git fetch upstream
git rebase upstream/master
git push -f
```

3 Other important features

- Branching. Check out https://www.atlassian.com/git/tutorials/using-branches

- Merging. Check out https://www.atlassian.com/git/tutorials/using-branches/git-merge

- Merging vs. rebasing. Checkout: https://www.atlassian.com/git/tutorials/merging-vs-rebasing