

Beginner's Guide to using Git

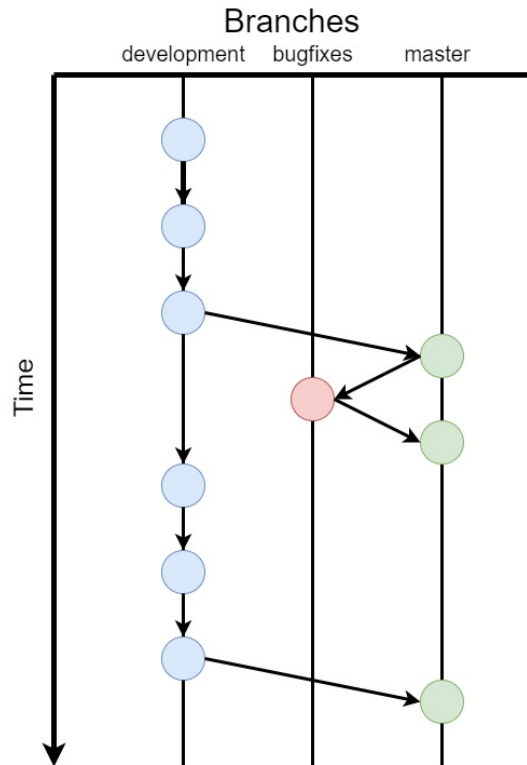
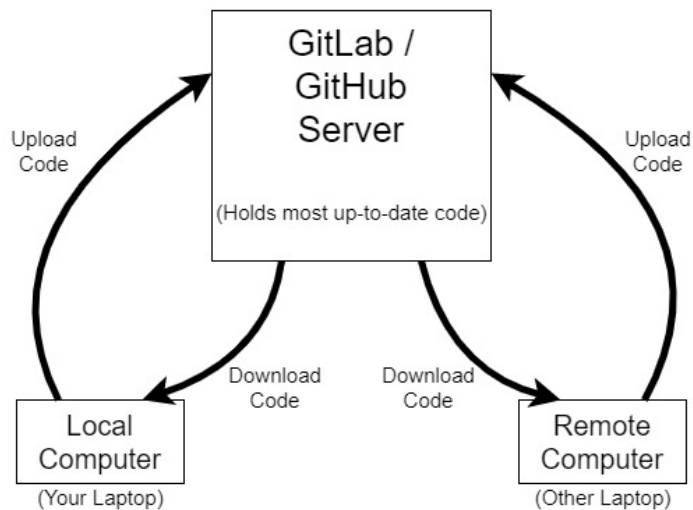
Written By: Kyle Swygert
Voiland College Tutoring Services
WSU CptS 223 and onward

What is Git?

Git is a tool that programmers use to upload their code to servers online so they can write a program from many different computers, create different versions of their programs, revert their changes if they upload broken code, as well as collaborating with other team members and many other reasons.

GitHub and GitLab are online servers and websites that are used to store code with Git online.

Code is uploaded to a place called a “repository,” or a repo. Each repository can have many different “branches” that hold different versions of a program or, in our case, different assignments. A Git repository that you are working on may have many different branches, but you can only have one of those branches downloaded to your computer at a time.



Installing Git

Linux (Ubuntu)

```
File Edit View Search Terminal Help
incredibr0@daily-driver:~$ sudo apt-get install git|
```

- Install the program from the terminal with the command:

Windows

- Download the Windows tool here: <https://git-scm.com/downloads>

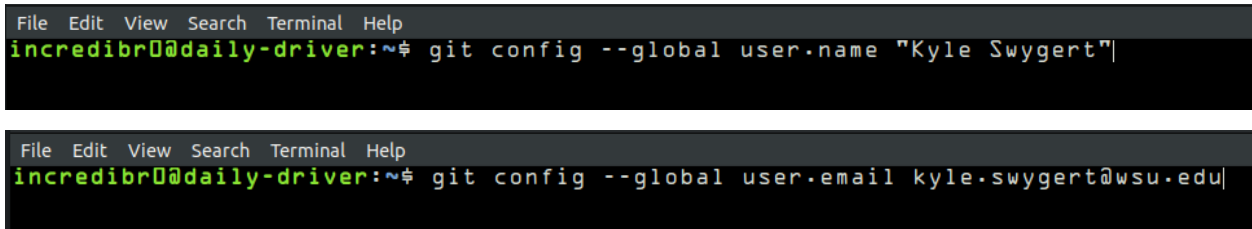
Mac

- Download the macOS tool here: <https://git-scm.com/downloads>

Set up before Use

Before we even start to work with the Git repositories online or on your computer, you need to add your Git username and email. Setting this information lets other people know who did what in a certain Git repo and is required to use Git. To do so, enter the commands below with your name and email:

- **git config --global user.name "First Last"**
- **git config --global user.email first.last@wsu.edu**



```
File Edit View Search Terminal Help
incredibr00@daily-driver:~$ git config --global user.name "Kyle Swygert"

File Edit View Search Terminal Help
incredibr00@daily-driver:~$ git config --global user.email kyle.swygert@wsu.edu
```

Using SSH Keys with Git

SSH keys are used with GitLab and GitHub to prove your identity so you don't have to enter your username and password to the Git server every time that you upload code to the servers. Using SSH keys is not required to use Git, but they make long term use simpler.

- <https://docs.gitlab.com/ee/ssh/>

Starting Using Git

Git requires a few steps to get your code that is online onto your computer, but these steps only need to be completed **once** per Git repository. After the initial start up, only a few steps are used. There are **two** ways to initialize a Git repository on your computer: **Cloning and Manual Setup**.

Cloning

1. Go to your newly created Git repository in the GitLab website.
2. Click the “Clone” button on the main page of the repository and copy the HTTPS clone URL to your clipboard.
3. Open a terminal and cd into the directory that you want to hold your Git repo.
4. Enter the command below to clone the entire repo into your directory:
 - a. **git clone (paste the URL that was copied before)**

```
File Edit View Search Terminal Help
incredibr0@daily-driver:~/Desktop/learning-git$ git clone https://gitlab.eecs.wsu.edu/43659/learning-git
```

```
File Edit View Search Terminal Help
incredibr0@daily-driver:~/Desktop/learning-git$ git clone https://gitlab.eecs.wsu.edu/43659/learning-git
Cloning into 'learning-git'...
Username for 'https://gitlab.eecs.wsu.edu': kswygert
Password for 'https://kswygert@gitlab.eecs.wsu.edu':
```

```
File Edit View Search Terminal Help
incredibr0@daily-driver:~/Desktop/learning-git$ git clone https://gitlab.eecs.wsu.edu/43659/learning-git
Cloning into 'learning-git'...
Username for 'https://gitlab.eecs.wsu.edu': kswygert
Password for 'https://kswygert@gitlab.eecs.wsu.edu':
warning: redirecting to https://gitlab.eecs.wsu.edu/43659/learning-git.git/
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0)
Unpacking objects: 100% (3/3), done.
incredibr0@daily-driver:~/Desktop/learning-git$
```

5. Congratulations, your repo is now on your computer!

Manual Setup

1. Go to either GitLab or GitHub and start a new repository with just a single README.txt
2. Create a directory for your Git repository and cd into the new directory.
 - a. Linux:
 - i. Open a terminal, cd to the location you want to make a folder, make a folder with mkdir, then cd into the new folder

b. Windows:

- i. Open the Git Bash application, cd to the location you want to make a folder, make a folder with mkdir, then cd into the new folder

c. MacOS:

- i. Open a terminal, cd to the location you want to make a folder, make a folder with mkdir, then cd into the new folder

3. Once inside the new directory, enter the command

a. git init

```
File Edit View Search Terminal Help
incredibr0@daily-driver:~/Desktop/learning-git$ git init
Initialized empty Git repository in /home/incredibr0/Desktop/learning-git/.git/
incredibr0@daily-driver:~/Desktop/learning-git$ |
```

- i. This will make the directory that you are in into a Git repository and this is where your files from this Git project will be stored.

4. We now need to add the source of the project, so we need to add a remote. Enter the following commands:

a. git remote add origin (URL for Git Repository online)

```
File Edit View Search Terminal Help
incredibr0@daily-driver:~/Desktop/learning-git$ git remote add origin https://gitlab.eecs.wsu.edu/43659/learning-git|
```

b. git pull origin master

```
File Edit View Search Terminal Help
incredibr0@daily-driver:~/Desktop/learning-git$ git pull origin master|
```

```
File Edit View Search Terminal Help
incredibr0@daily-driver:~/Desktop/learning-git$ git pull origin master
Username for 'https://gitlab.eecs.wsu.edu': kswygert
Password for 'https://kswygert@gitlab.eecs.wsu.edu': |
```

```
File Edit View Search Terminal Help
incredibr0@daily-driver:~/Desktop/learning-git$ git pull origin master
Username for 'https://gitlab.eecs.wsu.edu': kswygert
Password for 'https://kswygert@gitlab.eecs.wsu.edu':
warning: redirecting to https://gitlab.eecs.wsu.edu/43659/learning-git.git/
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0)
Unpacking objects: 100% (3/3), done.
From https://gitlab.eecs.wsu.edu/43659/learning-git
 * branch          master      -> FETCH_HEAD
 * [new branch]   master      -> origin/master
incredibr0@daily-driver:~/Desktop/learning-git$ |
```

5. Congratulations, you now have the files from the master branch on your local machine!

Changing Branches in Git

In a Git repository, you can have multiple branches, which are different versions of the same project, and you may need to change branches on your computer depending on the section of a project that is currently being worked on.

1. We can list the available branches by entering the following command:
 - **git branch -a**

```
File Edit View Search Terminal Help
incredibr0@daily-driver:~/Desktop/learning-git# git branch -a
* master
remotes/origin/HEAD -> origin/master
remotes/origin/master
incredibr0@daily-driver:~/Desktop/learning-git# |
```

2. When the branch names are displayed, the branch name is the string after the final '/' character in the whole string. To change to that branch, we enter the command:
 - **git checkout (branchName)**

```
File Edit View Search Terminal Help
incredibr0@daily-driver:~/Desktop/learning-git# git checkout newBranch|
```

3. We can also **Add** a branch to the repository from the terminal as well. We can do this by entering the following command:
 - **git checkout -b (branchName)**

```
File Edit View Search Terminal Help
incredibr0@daily-driver:~/Desktop/learning-git# git checkout -b newBranch|
```

4. Congratulations, you are now on a different branch!

Uploading using Git

When you upload using Git, a new version of the software is added to the history of the project that can be used to revert to, keep track of changes, and many other things.

1. To upload the files, we first need to add the files to be uploaded. Add files by entering the command:

a. git add *

```
File Edit View Search Terminal Help
incredibr0@daily-driver:~/Desktop/learning-git$ git add *
```

- i. This adds all of the files that are in the current directory.

2. We now need to add a message describing what has changed since the last upload. Add a comment by entering the command:

a. git commit -m "message about this commit"

```
File Edit View Search Terminal Help
incredibr0@daily-driver:~/Desktop/learning-git$ git commit -m "updated the README.md file"
```

```
File Edit View Search Terminal Help
incredibr0@daily-driver:~/Desktop/learning-git$ git commit -m "updated the README.md file"
[newBranch 7bf8cde] updated the README.md file
1 file changed, 1 insertion(+)
incredibr0@daily-driver:~/Desktop/learning-git$ |
```

- i. This may seem unnecessary, but this is a required step and will help if you ever need to revert your project to a previous state. The message should describe the changes that you made in the most recent update to the code.

3. Our new code is ready to be uploaded, but now we need to specify which branch to upload and finally upload the code. Upload the code by entering the command:

a. git push origin (branchName)

```
File Edit View Search Terminal Help
incredibr0@daily-driver:~/Desktop/learning-git$ git push origin master |
```

```
File Edit View Search Terminal Help
incredibr0@daily-driver:~/Desktop/learning-git$ git push origin master
Username for 'https://gitlab.eecs.wsu.edu': kswygert
Password for 'https://kswygert@gitlab.eecs.wsu.edu': |

File Edit View Search Terminal Help
incredibr0@daily-driver:~/Desktop/learning-git$ git push origin master
Username for 'https://gitlab.eecs.wsu.edu': kswygert
Password for 'https://kswygert@gitlab.eecs.wsu.edu':
warning: redirecting to https://gitlab.eecs.wsu.edu/43b59/learning-git.git/
Everything up-to-date
incredibr0@daily-driver:~/Desktop/learning-git$ |
```

- i. When you press enter after this command is typed out, your computer will have pushed the code to the branch on the online servers.
4. Congratulations, you have pushed new code to your Git repository! You can see the changes that you uploaded by going to the repository website and clicking in the file you changed.

Continuing Using Git

As of now, you now know how to create a Git repository, download the code, change branches, and upload code. As stated before, not all of these steps need to be done every time.

Whenever you are going to work on a Git repository project again, it is recommended to run the command: “git pull origin” to download any changes that may have been uploaded to the repository since the last time you worked on it. Other than that recommendation, keep using Git as described in the “Changing Branches in Git” and the “Uploading using Git” sections of this Tutorial.

Additional Resources

- YouTube Playlist for learning Git for Beginners:
<https://www.youtube.com/playlist?list=PLRqwx-V7Uu6ZF9C0YMKuns9sLDzK6zoiV>
- Interactive Website to learn Git terminology: <https://ndpsoftware.com/git-cheatsheet.html>
- GitHub Cheat Sheet:
<https://github.github.com/training-kit/downloads/github-git-cheat-sheet.pdf>
- Git Documentation: <https://www.git-scm.com/docs>