

Downloading the Github Repository

Note: Windows version

- This will be done through the command prompt
- Check that you first have Git installed, allowing you to use git commands
 - ↳ type `git version`
 - ↳ If it works, it should be the similar to the image below. Else you need to download Git

```
C:\Users\mattl>git version
git version 2.37.3.windows.1
```

- Now that you have Git installed, you can optionally navigate to the folder you want to download the repo.

↳ Some simple commands for the command prompt

`cd/` Navigate to the root

`D:` Change drives, the letter depends on the letter of the drive

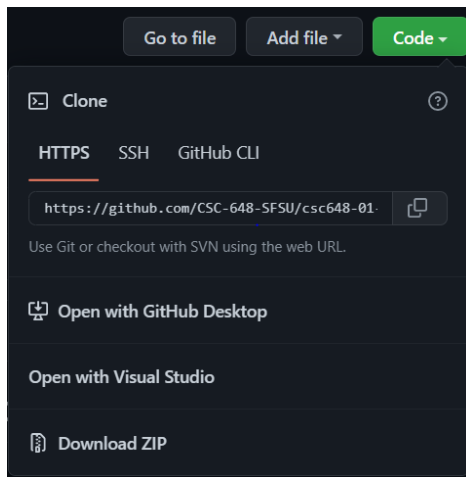
`cd folderName` Navigates to the folder, if available

- Once in the correct folder, use the command `git clone githubRepoURL`

↳ You can get the Github Repo URL from the github repo page

↳ Click the code button, should be green and around the top of the page

↳ Copy the URL



- Once you run `git clone githubRepoURL`, it should prompt you for a password or token
- After entering the token, it should show it downloading. Something similar to this:

```
D:\CSC648>git clone https://github.com/CSC-648-SFSU/csc648-01-fa22-team02.git
Cloning into 'csc648-01-fa22-team02'...
remote: Enumerating objects: 35, done.
remote: Counting objects: 100% (35/35), done.
remote: Compressing objects: 100% (31/31), done.
remote: Total 35 (delta 8), reused 9 (delta 0), pack-reused 0
Receiving objects: 100% (35/35), 7.25 KiB | 1.81 MiB/s, done.
Resolving deltas: 100% (8/8), done.
```

Downloading Git

[Instructions page on installation of Git](#)

Creation of a Token

- Go to the settings of your account on Github
- Click the Developer Settings in the left column
- Click the Personal Access Tokens button
- Click the Generate a New Token button

↳ I believe it needs only the repo scope, but you can add more if you like

Note: Once the token is generated, copy and save it somewhere since you can't get it back after you click away

Some Github Commands

Note: Can be entered in the terminal of your preferred IDE

<code>git status</code>	Shows the status of the current branch
<code>git add fileName.extension</code>	Adds the file to be committed
↳ <code>fileName</code> can be replaced with <code>*.fileExtension</code> to add all files with the same extension	
<code>git commit -m "comment"</code>	Commits all added files, comment is displayed on the repository
<code>git reset fileName</code>	Unstages the file from the commit, does not reset the changes
<code>git reset --hard</code>	Discards all history and restores to the last commit
<code>git push</code>	Pushes all committed files into the repository
<code>git checkout branchName</code>	Switches to an existing branch
↳ <code>git checkout master</code> switches to the master/origin branch	
<code>git checkout -b branchName</code>	Creates a new branch and switches to it
<code>git merge branchName</code>	Merges the history of the specified branch into the current one
<code>git pull</code>	Fetches and merges changes from the remote server to repository