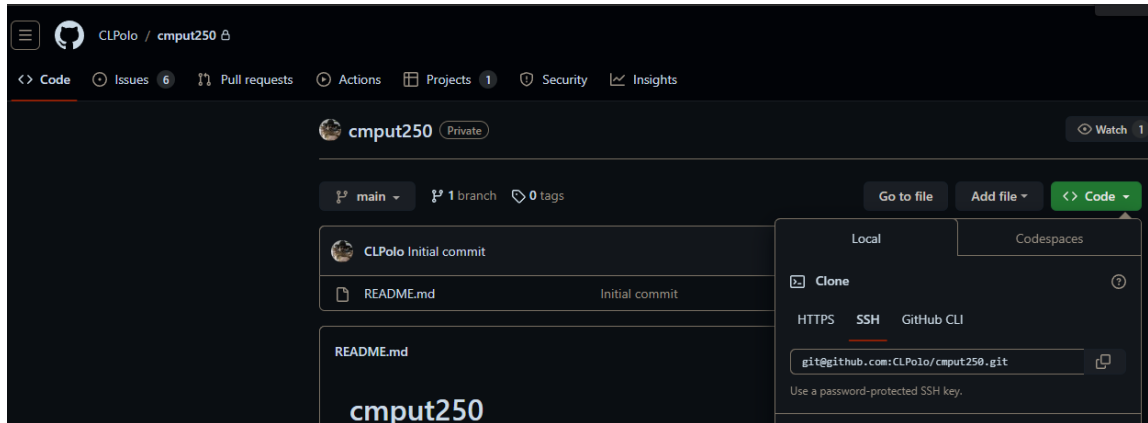


Part 1: Installing Git (if you don't already have it)

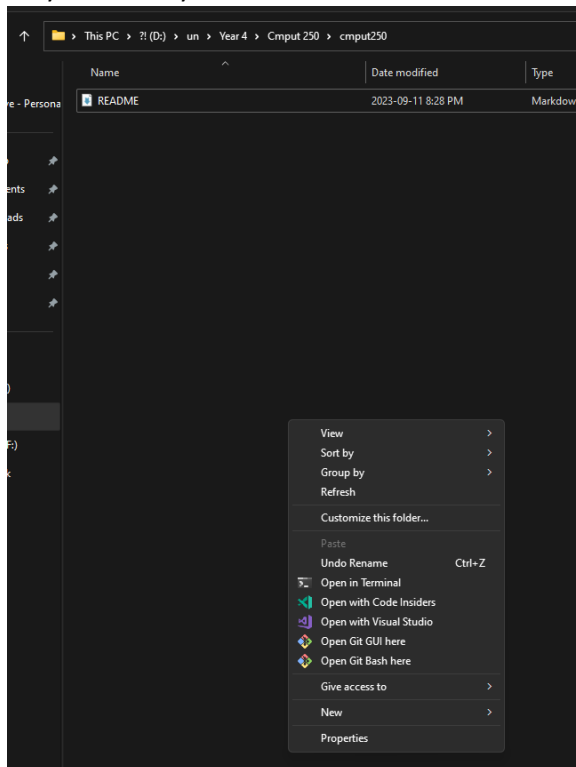
1. <https://git-scm.com/> download the installer from here.
2. Run the installer.
 - a. Ensure git bash is installed (this will allow showing branch names by the directory path)
3. That's it you are done!

Part 2: Accessing the git repository

1. Go to the repository: <https://github.com/CLPolo/cmpu250>
2. Copy the ssh link from the code section:



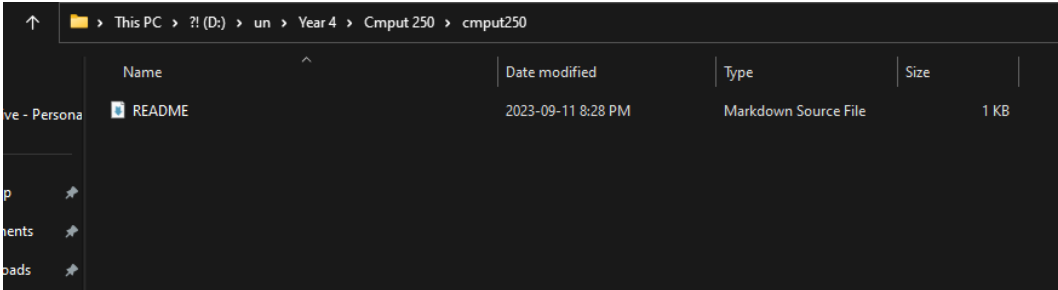
3. Open git bash (right click > show more options first) where you want the repository to be located on your local system:



4. Type ``git clone <ssh-link>`` replacing `<ssh-link>` with the link previously copied

```
Gurtaj K@Desktop310 MINGW64 /d/un/Year 4/Cmpu 250
$ git clone git@github.com:CLPollo/cmpu250.git
```

5. The files will now be there, you are done!



Part 2: Adding to the git repository

1. Traverse into the repository using ``CD: cmpu250`` from the previous folder

```
Gurtaj K@Desktop310 MINGW64 /d/un/Year 4/Cmpu 250
$ cd cmpu250

Gurtaj K@Desktop310 MINGW64 /d/un/Year 4/Cmpu 250/cmpu250 (main)
$ |
```

2. Create a branch for your changes using ``git branch <branch-name>`` (detailed names help) and switch to it ``git switch <branch-name>``
 - a. Alternatively use ``git brsw <branch-name>`` to create and switch to it (assuming config is set up, see last page for git configs)

```
Gurtaj K@Desktop310 MINGW64 /d/un/Year 4/Cmpu 250/cmpu250 (main)
$ git branch adding-git-instructions-to-repo

Gurtaj K@Desktop310 MINGW64 /d/un/Year 4/Cmpu 250/cmpu250 (main)
$ git switch adding-git-instructions-to-repo
Switched to branch 'adding-git-instructions-to-repo'

Gurtaj K@Desktop310 MINGW64 /d/un/Year 4/Cmpu 250/cmpu250 (adding-git-instructions-to-repo)
$ |
```

3. Make any changes to the file system as needed and type ``git add --all`` to add your changes to the selection of tracked files, then type ``git commit -m "<message>"`` to create a commit

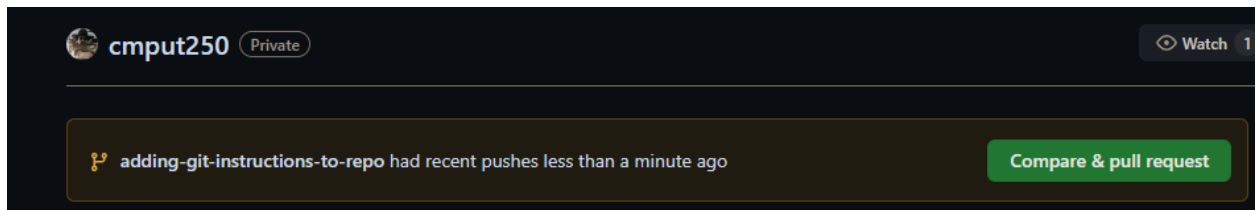
```
Gurtaj K@Desktop310 MINGW64 /d/un/Year 4/Cmpu 250/cmpu250 (adding-git-instructions-to-repo)
$ git add --all

Gurtaj K@Desktop310 MINGW64 /d/un/Year 4/Cmpu 250/cmpu250 (adding-git-instructions-to-repo)
$ git commit -m "added a word document with instructions on using git"
[adding-git-instructions-to-repo 03a984a] added a word document with instructions on using git
2 files changed, 0 insertions(+), 0 deletions(-)
create mode 100644 Documentation/Git Setup Help.docx
create mode 100644 Documentation/~$t Setup Help.docx
```

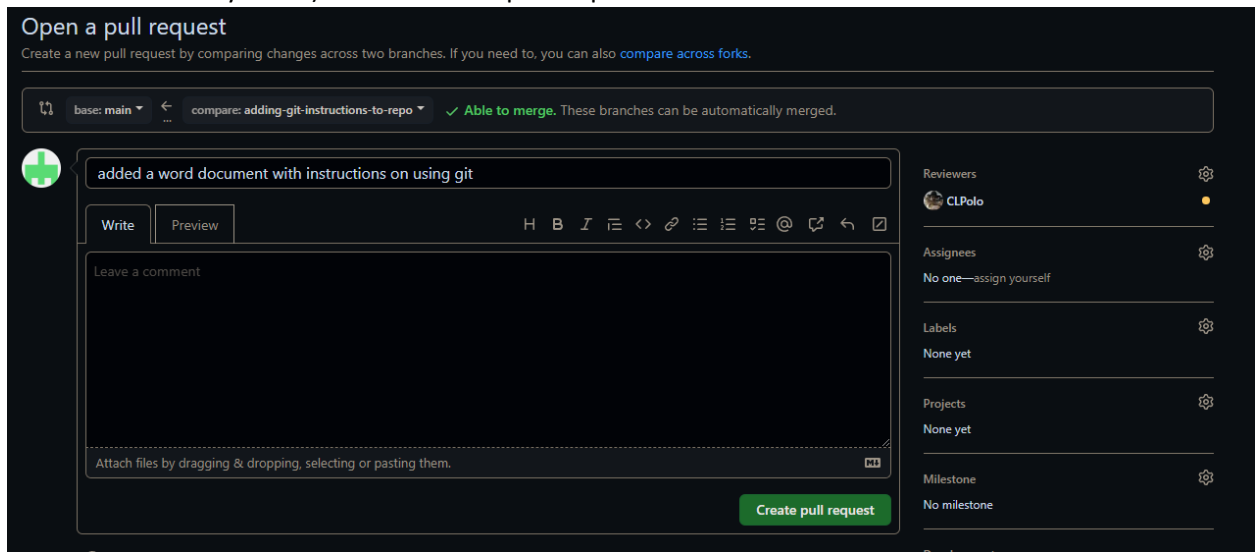
4. Push your changes using ``git push --set-upstream origin <branch-name>`` (or simply ``git pushu`` if config is set up [see last page])

```
Gurtaj K@Desktop310 MINGW64 /d/un/Year 4/Cmpu 250/cmpu250 (adding-git-instructions-to-repo)
$ git push --set-upstream origin adding-git-instructions-to-repo
Enumerating objects: 6, done.
Counting objects: 100% (6/6), done.
Delta compression using up to 6 threads
Compressing objects: 100% (5/5), done.
Writing objects: 100% (5/5), 141.49 KiB | 2.48 MiB/s, done.
Total 5 (delta 0), reused 0 (delta 0), pack-reused 0
remote:
remote: Create a pull request for 'adding-git-instructions-to-repo' on GitHub by visiting:
remote:   https://github.com/CLPolo/cmpu250/pull/new/adding-git-instructions-to-repo
remote:
To github.com:CLPolo/cmpu250.git
 * [new branch]      adding-git-instructions-to-repo -> adding-git-instructions-to-repo
branch 'adding-git-instructions-to-repo' set up to track 'origin/adding-git-instructions-to-repo'.
```

5. Go onto the github repository and create a pull request for your branch to be merged with the main branch:

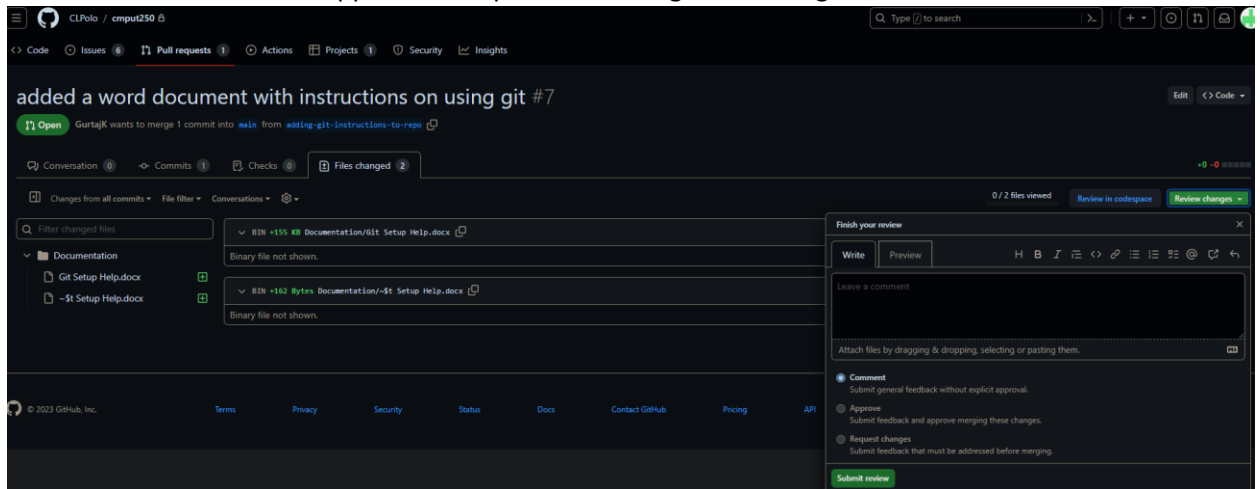


6. Add any reviewers, labels, link any issues (use #number in the comment section or the title and it will automatically link it) and create the pull request.



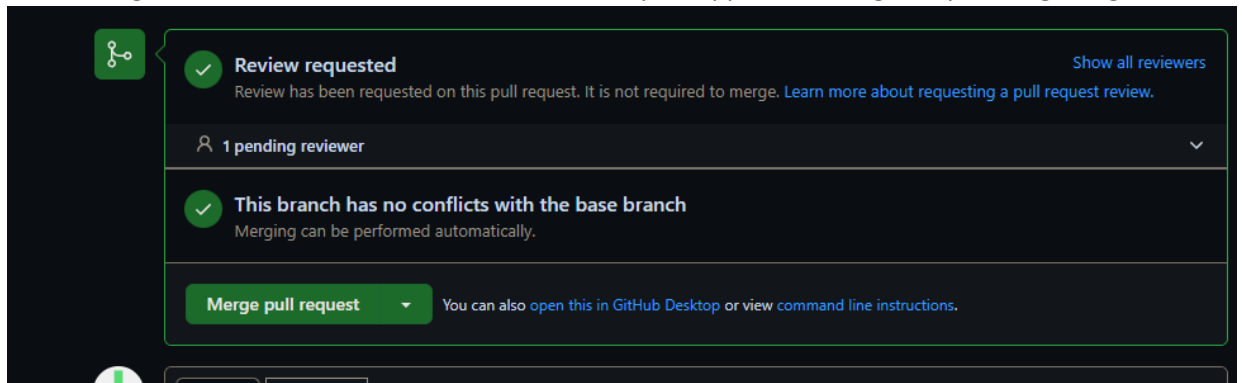
Reviewing PRs:

1. Add comments and either approve or request something to be changed:

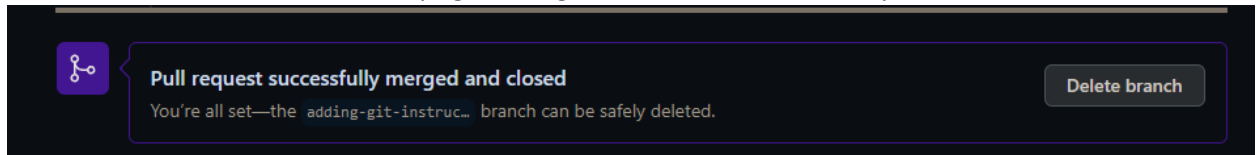


Merging PRs:

1. After changes from reviews have been made and the pr is approved, merge it by clicking merge:

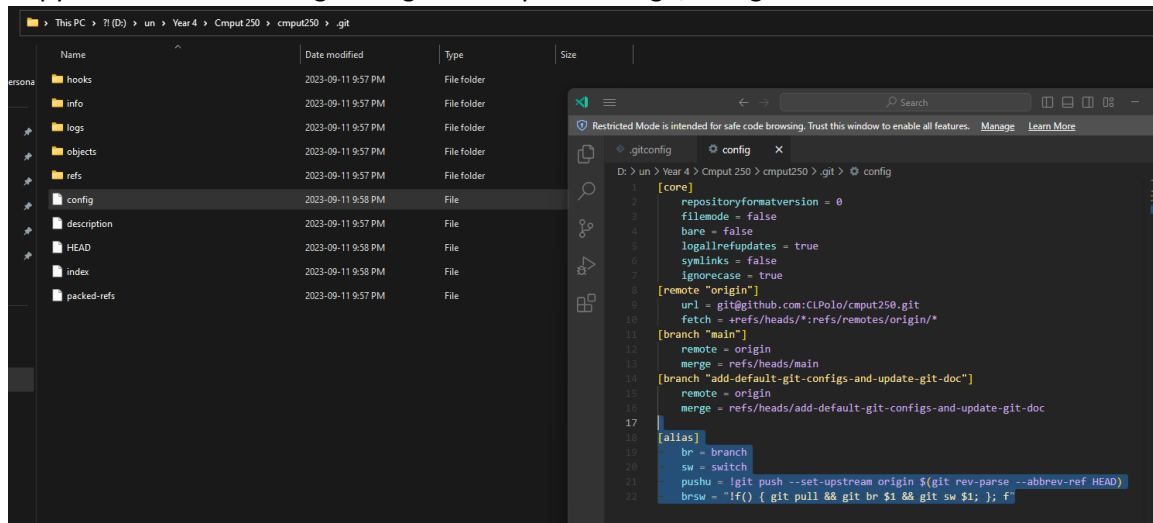


2. Delete the excess branch after the pr gets merged to avoid clutter basically



(OPTIONAL): Set up git configs

1. Copy the contents in the .gitconfig file into your local .git/config file:



The screenshot shows a Windows File Explorer window on the left, displaying the contents of the `.git` directory. The files listed are `hooks`, `info`, `logs`, `objects`, `refs`, `config`, `description`, `HEAD`, `index`, and `packed-refs`. The `config` file is highlighted. On the right, a VS Code editor window shows the contents of the `.gitconfig` file. The file is a standard Git configuration file with sections for `core`, `remote`, `branch`, and `alias`.

```
1 [core]
2   repositoryformatversion = 0
3   filemode = false
4   bare = false
5   logallrefupdates = true
6   symlinks = false
7   ignorecase = true
8
9 [remote "origin"]
10  url = git@github.com:CLPolo/cmpu250.git
11  fetch = +refs/heads/*:refs/remotes/origin/*
12
13 [branch "main"]
14   remote = origin
15   merge = refs/heads/main
16
17 [branch "add-default-git-configs-and-update-git-doc"]
18   remote = origin
19   merge = refs/heads/add-default-git-configs-and-update-git-doc
20
21 [alias]
22   br = branch
23   sw = switch
24   pushu = !git push --set-upstream origin $(git rev-parse --abbrev-ref HEAD)
25   brsw = "!(f) { git pull && git br $1 && git sw $1; }; f"
```