

CIS3534C GPA 9: Git and GitHub

In this assignment we will complete the following tasks:

- Use the Windows Command Prompt application to create a local Git repository (repo)
- Commit files to the local repo
- Sync the local repo to a new remote repo in your GitHub account
- Modify a file in the local repo, commit the change, and push the change to the remote GitHub repo

Assignment Prerequisites

- A Windows system with Git installed (the Horizon system meets this prerequisite)
- A GitHub account, preferably associated with your FSCJ student email address
- Access to GPA 8 files (unsolved)
- Access to **your** solution to GPA 8

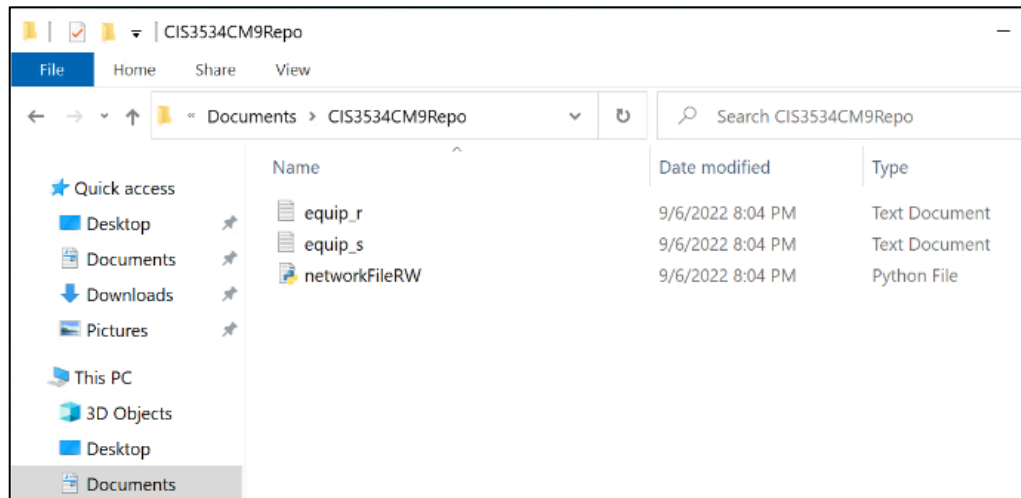
Instructions

Follow the steps listed here to complete this assignment. **Note that if you are doing this assignment on the Horizon system, you need to complete the first six steps up to and including "Push your local repo to the GitHub repo" before you log off from the system, otherwise you will lose your work.**

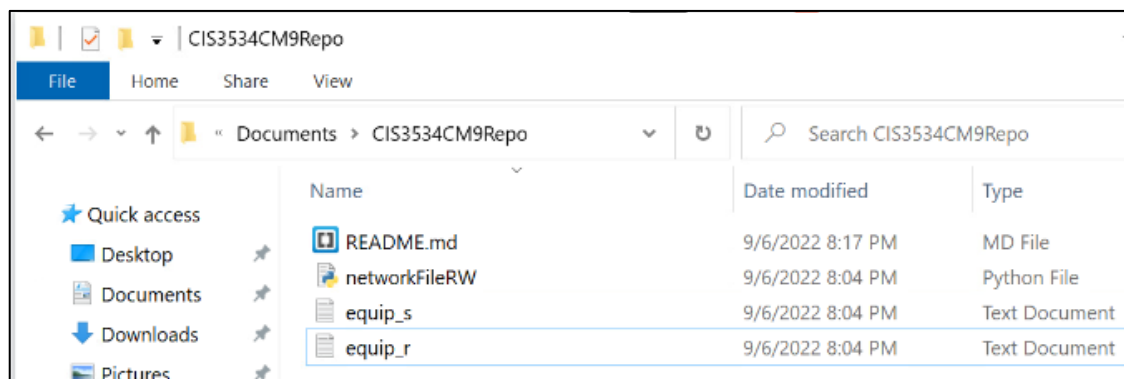
1. Create a new local Git repo.
 - Open a Windows command prompt app, navigate to your Documents folder, create a new folder with the name of your course and assignment (e.g. CIS3534CM9Repo), navigate to that folder, then run "git init ." (don't forget the dot). Here are the commands to execute; replace <youruser> with your Windows user ID:

```
C:\> c:
C:\> cd \users\<youruser>\Documents
C:\Users\<youruser>\Documents> mkdir CIS3534CM9Repo
C:\Users\<youruser>\Documents> cd CIS3534CM9Repo
C:\Users\<youruser>\Documents\CIS3534CM9Repo> git init .
```

2. Copy the **unsolved version** of the GPA 8 Python file and the associated data files to your local repo folder created in Step 1. (Data files are *networkFileRW.py*; *equip_r.txt*; *equip_s.txt*).

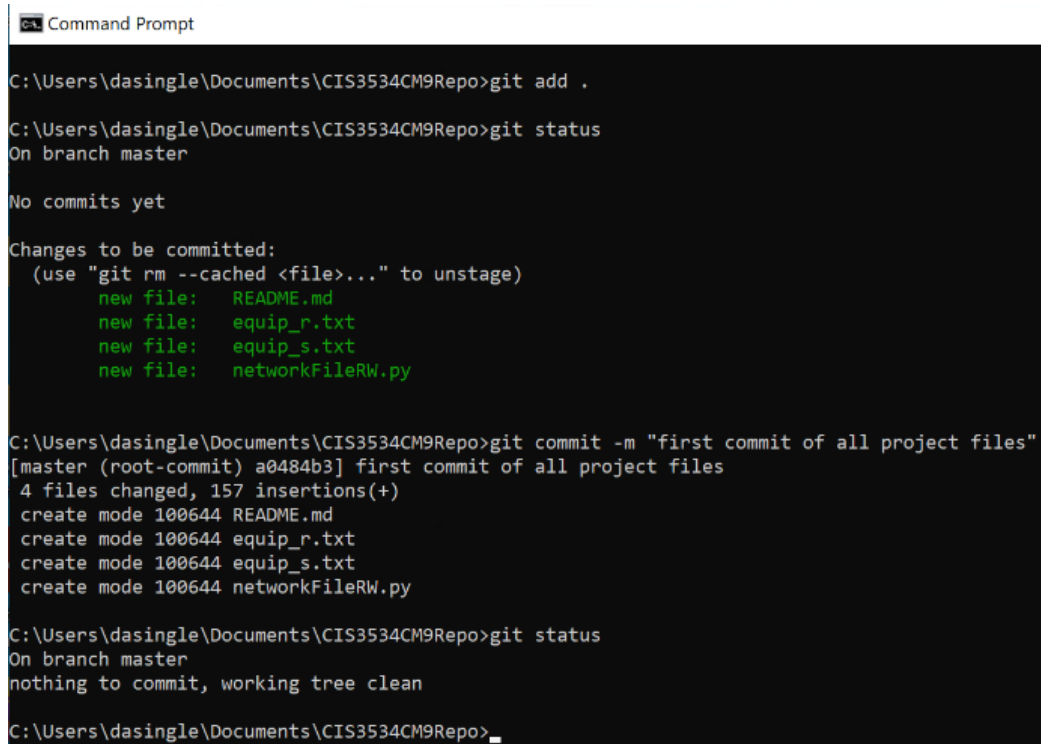


3. Create a README.md file in your repo folder which describes the repo – you should include the purpose of the Python source code file and the associated data files. You can use any text editor, but your file must be named README.md (md is short for “markdown”).



4. Use "git add ." to add ("stage") your files to the local repo, then commit them with a descriptive commit message (don't forget the dot). Use "git status" to verify the repo status after each operation.

```
git add .
git status
git commit -m "first commit for all project files"
git status
```



```
Command Prompt

C:\Users\dasingle\Documents\CIS3534CM9Repo>git add .

C:\Users\dasingle\Documents\CIS3534CM9Repo>git status
On branch master

No commits yet

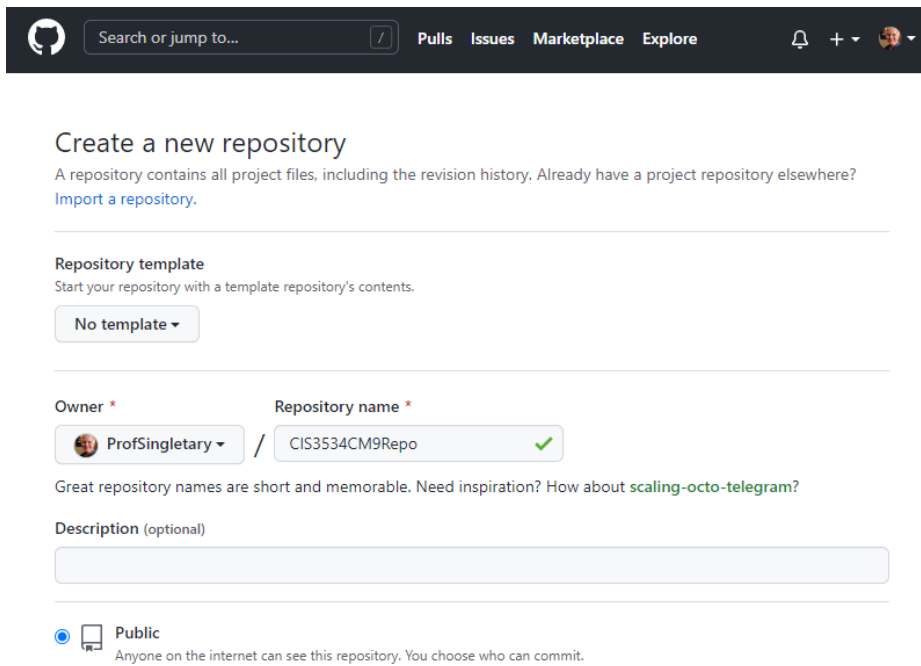
Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
    new file:   README.md
    new file:   equip_r.txt
    new file:   equip_s.txt
    new file:   networkFileRW.py

C:\Users\dasingle\Documents\CIS3534CM9Repo>git commit -m "first commit of all project files"
[master (root-commit) a0484b3] first commit of all project files
 4 files changed, 157 insertions(+)
 create mode 100644 README.md
 create mode 100644 equip_r.txt
 create mode 100644 equip_s.txt
 create mode 100644 networkFileRW.py

C:\Users\dasingle\Documents\CIS3534CM9Repo>git status
On branch master
nothing to commit, working tree clean

C:\Users\dasingle\Documents\CIS3534CM9Repo>_
```

5. Create a new GitHub repo using the same name as the local repo.



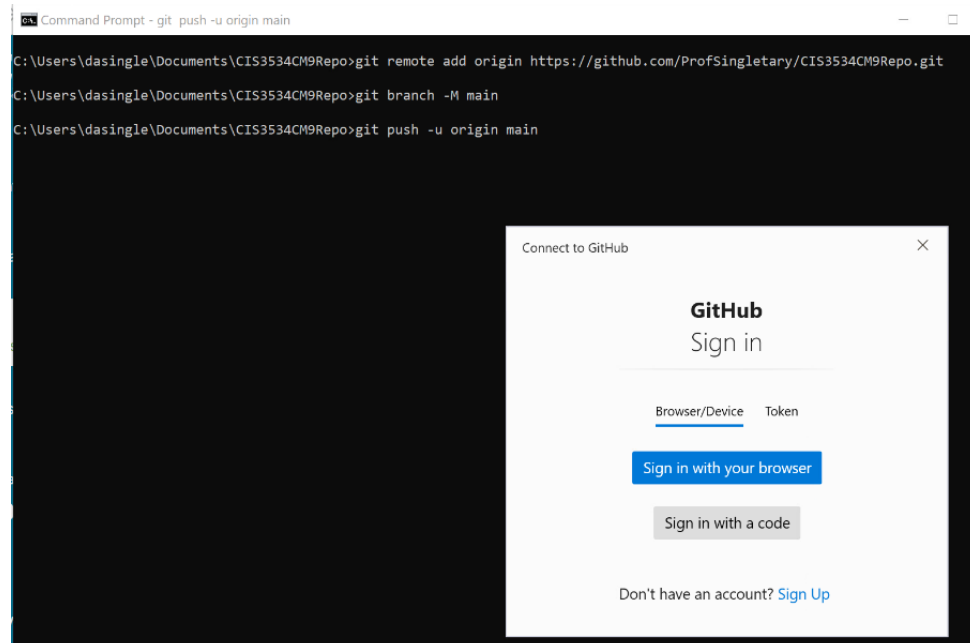
The screenshot shows the GitHub 'Create a new repository' page. At the top, there's a navigation bar with the GitHub logo, a search bar, and links for Pulls, Issues, Marketplace, and Explore. Below the navigation bar, the heading 'Create a new repository' is followed by a subtext: 'A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)'. Under the 'Repository template' section, there's a button labeled 'No template'. The 'Owner' field is set to 'ProfSingletary' and the 'Repository name' field is set to 'CIS3534CM9Repo', which has a green checkmark. Below these fields, a hint says: 'Great repository names are short and memorable. Need inspiration? How about [scaling-octo-telegram](#)?'. There's a 'Description (optional)' text area. At the bottom, the 'Public' radio button is selected, with a note: 'Anyone on the internet can see this repository. You choose who can commit.'

6. Push your local repo to the GitHub repo (you may be prompted to authenticate to GitHub in a browser during the push step).

```
git remote add origin <your GitHub repo URL>
```

```
git branch -M main
```

```
git push -u origin main
```



At this point if you are working on the Horizon system you can log off without losing your work as it has been safely stored on GitHub.

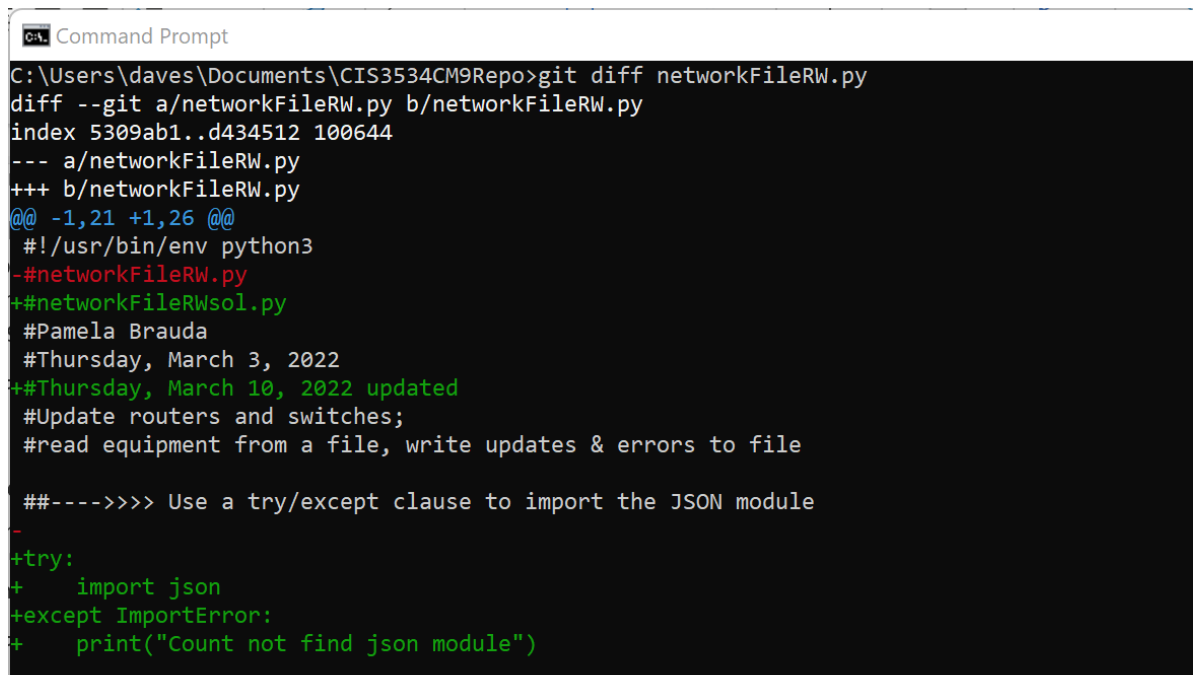
If you have not logged off and wish to continue working on this project, move directly to step 7.

If you have logged off and need to recreate your local repo, log on and open a new Windows command prompt app, navigate to your Documents folder and execute the following command (replace <github userid> with your GitHub user ID, and use the repo name of your remote repo if it is not CIS3534CM9Repo. This will replicate your remote repo in a folder with the same name.

```
git clone https://github.com/<github userid>/CIS3534CM9Repo
```

7. Modify the Python source file in your local repo so that it reflects the changes in your GPA 8 solution and test it to verify your changes work correctly (your Python source file ID header should provide a brief summary of the changes).
8. Run git diff and note the differences between the original file and the modified file.

```
git diff networkFileRW.py
```



```
Command Prompt
C:\Users\daves\Documents\CIS3534CM9Repo>git diff networkFileRW.py
diff --git a/networkFileRW.py b/networkFileRW.py
index 5309ab1..d434512 100644
--- a/networkFileRW.py
+++ b/networkFileRW.py
@@ -1,21 +1,26 @@
-#!/usr/bin/env python3
-#networkFileRW.py
+#networkFileRWsol.py
 #Pamela Brauda
 #Thursday, March 3, 2022
+#Thursday, March 10, 2022 updated
 #Update routers and switches;
 #read equipment from a file, write updates & errors to file

-##---->>> Use a try/except clause to import the JSON module
-
+try:
+    import json
+except ImportError:
+    print("Count not find json module")
```

9. Check the status of your local repo, verify the change is seen, then stage your modified file to the local repo and check the status again.

```
git status
git add .
git status
```

10. Commit your modified file to the local repo with a descriptive commit message.

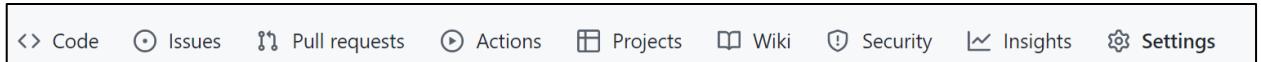
```
git commit -m "your commit message" networkFileRW.py
```

11. Push your local repo to the GitHub repo.

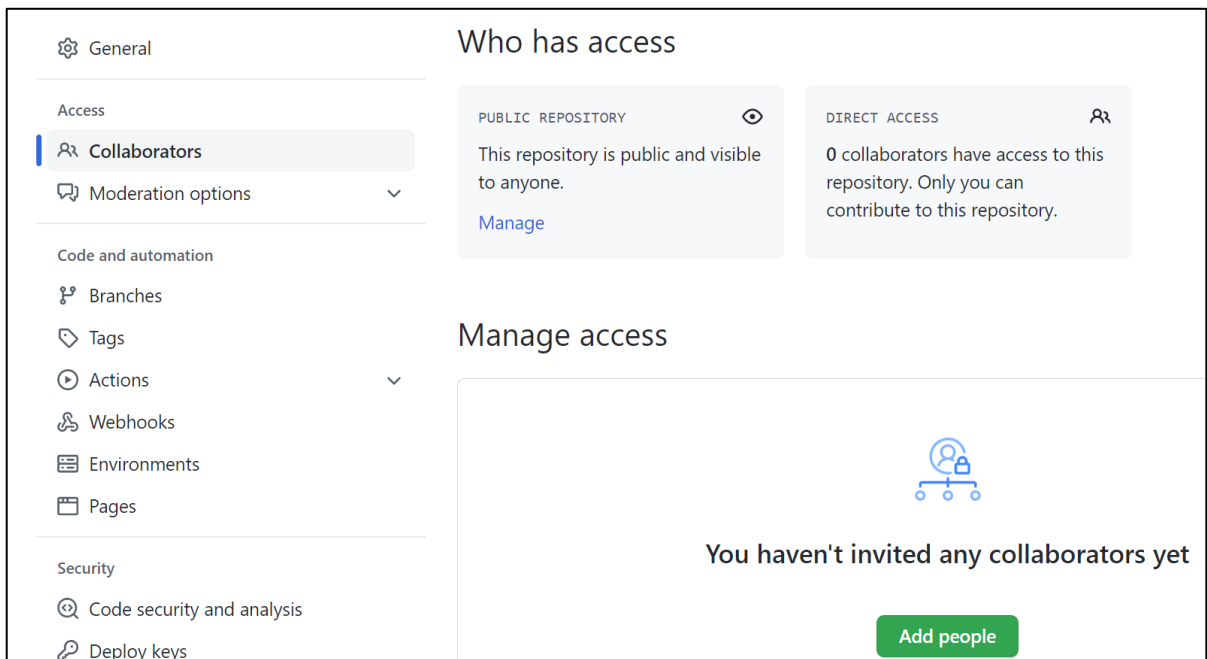
```
git push
```

12. Invite your instructor to your GitHub repo as a collaborator.

On your GitHub repo, go to **Settings**:



Then go to **Collaborators** and **Add people** (Use the GitHub user name provided by your instructor):



13. Return to **GPA 9 in Canvas** and use the **Text Entry** option to submit your **GitHub user name** and **link to your repo** to confirm assignment completion. **This step must be completed in order to receive credit for this assignment.**