# **Git and Github Fundamentals**

***<One submission of this assignment per group>***

First, answer the following questions regarding Git and Github:

1. What’s the difference between Git and Github?
2. Explain the concept of origin and master.
3. What is a Github branch?
4. What does push do? What about pull? Fork? Clone?
5. What is the difference between fork and clone?
6. Check out a professor’s Github page at <https://github.com/shadsluiter>. How many repositories does he have? Name one of the repositories.
7. Can there be multiple branches open at the same time in a repo? Can multiple people work on the same branch?
8. What is a tracked file? What is a staged file?
9. What is a gitignore file? How do you add files to gitignore in VS Code?

We need to do a few things to make git commands available in VS Code and to get you signed in to your Github account:

1. Download git for Windows. In VS Code, on the left side of the app, there are some icons. About halfway down the list is one for source control. Click the button to download git for Windows. We want the 64-bit version. Run the executable file to install git. Afterwards, click on the reload button in the source control panel or restart VS Code.
2. To sign into VS Code properly, first let’s make sure you’re logged out (can click on the Accounts icon on the left part of the interface). Reopen VS Code then run the following commands in the terminal window within VS Code:

***git config --global user.name "John Doe"***

***git config --global user.email johndoe@example.com***

Close VS Code again. Restart and then sign in via Account tab. If you get an error running git commands (git not found or something like that), run the commands in Windows Powershell instead, then restart VS Code. If you don’t see an option to sign in within the Accounts icon, go to “Clone Repository” so we can force it to have you sign in.

Watch the short video [here](https://www.youtube.com/watch?v=i_23KUAEtUM&ab_channel=VisualStudioCode), then do the following as you go through the video:

1. On one of your computers, open the folder that contains your recent work in R. Add the Word documents or Google Docs from the assignments to the folder.
2. Create a new repository using this folder.
3. Rename the main branch to “test-branch” as done in the video, then rename it back to “main”.
4. Add only the R files to the tracked files, then write a commit message and commit the changes to your repository.
5. Create a new branch called “include-assignments” and add the Word docs/Google Docs to the tracked files for this branch. (Note that normally we wouldn’t make a separate branch for adding files. This is just a proof of concept.), then write a commit message and commit the changes.
6. Open one of the R files and make some random additions and deletions to the files. They do not have to be syntactically correct. Note the gutter highlighting that indicates these changes. Now discard these changes.
7. Add some comments to one of the R files. Do a diff of the old and new version of the file and paste a screenshot here (standard or inline view is fine).
8. Stage the changes you made and then make a commit with an appropriate commit message.
9. Switch to the “main” branch. Now merge the “include-assignments” branch to the “main” branch.
10. Publish the “main” branch to github, then go look at the repository on Github.
11. Have your teammate or teammates clone your repository to their computer.

You can also watch [this](https://www.youtube.com/watch?v=LdSwWxVzUpo&ab_channel=VisualStudioCode) video about pull requests. Lastly, please watch [this](https://www.youtube.com/watch?v=HosPml1qkrg&ab_channel=VisualStudioCode) helpful video about merge conflicts. These won’t happen often during capstone, but it’s good to be prepared for the possibility and to know that it’s not really a big deal.