GitHub Classroom Introductory Assignment

The goal of this assignment is to introduce you to using GitHub Classroom for assignments. We'll be using this platform for nearly all of your individual and group assignments throughout the semester, so it's important that you've got everything setup properly and understand how to use these technologies.

Background

GitHub Classroom is a tool that helps me organize and manage GitHub repositories that are used for class assignments and group project deliverable. GitHub Classroom automates repository creation and access control, making it easy to distribute starter code and collect assignments on GitHub.

A GitHub Classroom assignment is a GitHub repository with access control setup so both you and your instructor can access it. When you accept an assignment, GitHub Classroom will automatically create a new, personalized copy of the assignment repository for you. The assignment repository will belong to your course's organization account on GitHub, but you and your teacher will have access to it.

Once an assignment has been accepted, students no longer interact with GitHub classroom. They just interact with GitHub.

Git and GitHub Desktop

In order to complete most assignments for this course, you will need to be able to use git and be familiar with GitHub. If you join a data science (or any engineering) team someday, you'll almost certainly do most of your work in code, and you'll likely collaborate with your team via git and GitHub.

You probably already have experience with these tools from prior classes, but just in case, I'd encourage you to watch this excellent tutorial video that summarizes git, GitHub, and GitHub Desktop. It's 22 minutes long, but it's clear and comprehensive (and you can up the playback speed if you want).

Terms to Know

While you're watching the video above and playing with git/GitHub for this assignment, be sure that you are familiar with the following terms. You won't be explicitly tested on these terms, but you need to know them to be successful in completing and submitting assignments. If you can't explain these terms to a friend, I suggest doing some additional research online until you're comfortable with them.

- (Understand the difference between git, GitHub, and GitHub Desktop)
- Repository
- Branch
- Clone
- Add
- Commit
- .gitignore
- Push
- Pull
- Publish
- Pull Request
- Merge

Git Utilities

There are dozens of great git tools that allow you to use git and GitHub, ranging from the command line git tool (for the hardcore among you), to GUI tools like GitHub's own GitHub Desktop, to plugins for most common code editors through which you can add, commit, branch, and push/pull files from within your IDE.

If you have a favorite tool, you're welcome to use it. If you don't, I'd recommend using one of the following tools this semester. The list is ordered, meaning that the first in the list is my default recommendation.

- Use GitHub Desktop. It's available (free) for both Mac and Windows, it's built by GitHub, and it's
 fully capable of doing everything you'll need to do this semester. The (minor) disadvantage is that
 it's a separate application from RStudio, but it's not a huge inconvenience. <u>Download GitHub</u>
 <u>Desktop here</u>.
- Use RStudio's built-in git support. You can certainly use this to accomplish all of the git operations for the class. It's not especially pretty, but it works, and it saves you a step because it's right there in RStudio.

Assignment Instructions

Important: The following instructions assume that you've already created a GitHub account (if necessary), accepted the first assignment using the link from the course schedule in Learning Suite, and properly linked your GitHub account with your name from the roster. It also assumes that you have completed Part 1 of this assignment (R and RStudio Environment Setup). If you have not yet done those things, please follow those instructions first.

Also Important: If you get stuck on any of the steps below, it's important that you reach out to a TA or me so we can help you get comfortable with everything happening in this assignment.

- 1. Clone your assignment repository to your local machine. The easiest way to do this is from the web interface. On the main "Code" tab, there is an attractive green "Code" button, which drops down to provide several options for cloning. Assuming you're using GitHub desktop, just use the "Open with GitHub Desktop" option and you'll hopefully be redirected to the GitHub Desktop app with the details pre-populated. If that doesn't work, you can also copy the provided <a href="https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://link.pub.edu/https://lin
- 2. Head to RStudio and ensure that you can see your newly cloned repository from within the "Files" tab in the lower right pane. (If you followed Part 1 of this assignment and setup your RStudio environment, the directory shown by default in that pane should be your main git directory, and the file referenced in the next step should be nestled under the cloned repository folder in that directory.)
- 3. Open the editme.txt file and add a few sentences describing the most unique or funny way someone recently showed you love or provided you a service and why you're grateful for that act.
- 4. When you have finished editing the file, save the changes and commit them to your repository, adding an appropriate commit message.
- 5. Create a new R Script file called added_file.R and save it to this repository. Add the block of code below to the file.

```
library(tidyverse)
string_1 <- "This is a string."</pre>
```

6. Step through each of the commands that you pasted from above (remember that Cmd-Enter / Ctrl-Enter shortcut!). You don't have to understand everything that's happening. Just make sure that the code runs without any errors and that at the end, when you run the last count command, you see something like this displayed in the console:

```
> sw_data %>% count(homeworld)
# A tibble: 5 × 2
homeworld n
<chr> <int>
1 Alderaan 1
2 Naboo 1
3 Stewjon 1
4 Tatooine 4
5 NA 2
```

7. Save the script file above, include it in a new commit, adding an appropriate commit message.

Just to be sure, before you commit and push your changes in the next step, your local folder should contain the following files: added file.R, editme.txt, README.md.

8. **Most importantly**, make sure to **push your changes** to GitHub, probably using the "Push Origin" button in GitHub Desktop. *This is how you submit your assignment*. If you don't, we can't see them and give you credit. I'll try to remind you to do this at the end of each assignment, but you'll be responsible for submitting your code using GitHub for all assignments distributed this way.

That's it! Adding changes to files or adding new files to repositories like this will be the way that all coderelated assignments are submitted for the semester. That wasn't so bad, right?!?