GitHub Repository Setup

Before beginning the Module 1 assignments, following the directions below to create a GitHub repository using Git and the CLI. The repository will be used throughout the course to host the coding assignments. If you have not already installed Git, please do so before continuing: [Install and set up Git](https://support.atlassian.com/bitbucket-cloud/docs/install-and-set-up-git/) (atlassian, 2024). Make sure that you complete the steps to set the global username and global email address.

**Local Directory Setup Instructions**

* Open a command prompt window.
  + If you are on a Windows machine, use PowerShell or cmd (press the Windows key and start typing the name of the command prompt you want to use).
  + If you are on macOS, use the Terminal.
  + If you are on Linux, use the Terminal.
* Navigate to your user directory.
  + If you are on Windows enter:  cd \
  + If you are on macOS/Linux enter: cd ~
* Create a new directory and name it csd
  + mkdir csd
* Change directory into the csd directory.
  + cd csd

# **Repository Setup Instructions**

* Go to GitHub and sign up.
* Sign into GitHub.
* Create a new GitHub repository and name it csd-430.
* Enter a description.
* Make the repository public and "Initialize this repository with a README.md" file.
* Add .gitignore file (Python).
* Choose "Create repository".

# **Clone Instructions**

* Open a command prompt window.
  + If you are on a Windows machine, use PowerShell or cmd (press the Windows key and start typing the command prompt you want to use).
  + If you are on macOS, use the Terminal.
  + If you are on Linux, use the Terminal.
* Sign-in to GitHub (if you are not already).
* Click on your csd-430 repository.
* Click on the green Code button and copy the URL for csd-430’s repository.
* In the command prompt or terminal window, verify you are in the correct working directory (you should be in the csd directory) and enter git clone [copied URL]
  + Verify the repository was created.
  + For Windows enter: dir
  + For macOS/Linux enter: ls

# **Git Stage, Commit, Push**

One thing to keep in mind is that Git/GitHub needs to stay in synch. If you make changes to your local directory, you need to push them to GitHub. If you make changes to GitHub, you need to pull them into your local directory. If you make changes to GitHub, and then try and push new changes from your local directory without pulling the first change from GitHub, you will get an error message telling you just that.