These are the basic strings of git commands needed to interface with Github. Use this after you watch the video or read the powerpoint. ONLY USE THIS FOR COMMAND INSTRUCTIONS - THIS DOESN'T EXPLAIN NORMS OR BEST PRACTICES. This is not exhaustive, just a beginners command list.

WITH EXPLANATIONS

Establish a Local Repository

- Navigate to your directory

    ("dir" command will show the current directory navigation points, "cd ***directory***" where ***directory*** is the name of the folder you want to navigate to. If you just type "cd", the computer will bring you back to the base. "cd ../" will navigate up the folder.")

- Use the command "git init".

Connect a Repository to GitHub

- On GitHub, navigate to the repository you want to connect. Then, click on the "Clone or Download" button and copy the link.
- Navigate to the directory on your computer that you want to this repository to be stored in.
- Use the command "git clone ***COPY AND PASTE LINK***"

Start a new GitHub Repository

- Click the "New" button on the MNGSC Ballooning home page.
- Create a repository (Include a README, do not include a .gitignore or a user license)
- Connect the repository to your computer.

Pull from GitHub

- Navigate into the clone repository.
- Use the command "git pull origin ***BRANCH NAME***" (The branch name that you want to pull- this defaults to "master")

Save files to a repository

- Navigate to the repository.
- Navigate to the branch in the repository you want to save into. (Use "git branch" to see what branches exist. Use "git branch ***BRANCH NAME***" to create a new branch. Use "git checkout ***BRANCH NAME***" to move into an existing branch.
- Use "git status" - everything in red is an unsaved change
- Use git add command
    - "git add ." will add every file (ONLY USE IN THE INITIAL COMMIT TO A REPOSITORY)
    - "git add -u" will add every EDITED file
    - "git add ***FILE NAME***" will add a specific file

- Use “git status” to make sure that everything is green. If it is not green, it will not be committed!
- Use “git commit -m “***STATE CHANGES***” to save the files.

Push to GitHub

- Save the files to the clone repository.
- Use “git push origin ***BRANCH NAME***”

Merging GitHub Branches

- GitHub cannot merge branches. Merge the branches locally (Using “git merge ***BRANCH NAME***”), (the branch name will state the branch you want to merge with the one you are currently checked into) and then push the change. Merge will save the files automatically.
- If your branch is not up to date, use “git fetch origin ***BRANCH NAME***” in a new branch, and then merge the branches locally.
    - All merges for github should be done locally. Try to pull new code as frequently as possible to avoid having to resolve code within the Arduino IDE.

Delete a Branch

- On github, use the settings in the repository
- On git, use ‘git branch -d ***BRANCH NAME***”

COMMAND LISTS

File Navigation

- cd directory
- cd
- cd ../
- dir
- ls

Branch Navigation

- git checkout <branch>
- git branch <branch>
- git branch

Establish a Local Repository

- git init

Connect a Repository to GitHub

- git clone <link>

Start a new GitHub Repository

- On GitHub

Pull from GitHub

- git pull origin <branch>

Save Files to a Repository

- git add . OR git add -u OR git add <fileName>
- git commit -m “MESSAGE”

Push to GitHub

- git push origin <branch>

Merging GitHub Branches

- git fetch origin <branch>
- git merge origin <branch>

If a pull brings you to a weird page…

- Press the “i” key.
- Arrow down once, type your commit message
- Press escape
- Type :wq
- Press enter

Delete a Branch

- git branch -d <branch>

Other Useful Commands

- git status
- git reset (To undo an add staging)