What is Git??

GIT is a version control system. \* saves history of changes \*revert to prior versions \* merge work from multiple collaborators

Git origin story:

Linux by Linus Torvalds is a big open-source software project started in the 90s.

The creator had people emailing him asking him to change things, so he invented Git so they could do it themselves.

Linus then became known as the time lord.

Git tips and tricks:

Commit early and often.

Saving every 5 minutes with descriptive commit messages is important.

Could your team members work with your commit message?

Don't put git repos inside of git repos.

DEFINE repository:

A repository is a directory (aka folder) where you are saving all the code files for any given project. First, you make a folder and then use git init to make it into a git repository.

Branching

"On branch master" (means on the main project, not a side branch)

to add to main branch use -m

to add to side branch use -t

Git commands to type in terminal:

git init (initializes repository (tells git to look for files in this folder))

ONCE PER PROJECT

git add . (stages files for a commit; the dot adds all files in the directory)

git commit -m "comment" (creates a commit aka save. This is a save command in git; save often and frequently to track your changes)

the -m commits to the master branch, e.g. the main branch

git log (shows history of changes made by all contributing members)

git status (shows you what’s added ("staged") but not yet committed, and tells you if any files have changed since last commit)

git commit -am "comment" (adds and commits in one step)

git add -u (tells commit to remove files you've deleted)

git push origin master (sends updates to GitHub)

git pull origin master (pulls updates from GitHub)

Working with shared repositories on GitHub:

To turn in homework/quiz/etc, push back to your account, then do a PULL REQUEST (PR) (request that they pull code from yours)

Clone onto local machine: \* fork from someone else's GitHub to your own \* navigate in terminal to the place where you want to put that folder. \* git clone (copy from your own GitHub account to your home computer) \* git remote add upstream (link your home computer to the upstream repository)

GitHub accounts are remote. git status doesn't know about edits made on GitHub that have been committed

In order for Git to respect the deletion of files, a "git add" modifier is needed. git add -u (adds the "deleted-ness" of a deleted file)

GitHub commands to type from terminal:

git push origin master (means push to GitHub my (local) master branch)

git pull origin master (pulls new changes from the master branch on GitHub to local repository)

git clone

git remote add upstream

git remote -v (shows all GitHub repositories)

git pull upstream master

git push upstream master

To start a git repository:

1. in terminal, navigate to the folder you want to use

2. git init (just do this once per project)

3. make some files and put them in the directory

4. git add . (the dot says to add all the files in the directory and below)

Or use: git add README.md (adds just one file) (this stages the files)

5. git commit -m "commit message"

(summarize what changes you made, be specific in present tense; you are committing these files to a certain version)

6. change your file and save it in sublime text

7. git status (check to see if you have made changes since the last commit)

8. git add <filename>

9. git commit (creates a commit(SAVE GAME))

10. git log: Displays the commits along with messages.

11. go to GitHub and make a new repository on GitHub

12. link GitHub repository to your computer's local repository:

git remote add origin <https://github.com/USERNAME/REPO_NAME.git>

git push -u origin master

13. git push origin master

14. git push

15. git pull origin master

16. git pull