CSI 34:
Git
Why git?!

• Version history!
• Access files from anywhere!
• Great for collaboration!
• Great for maintaining large code bases!
• In CS134, you use it for
  • Receiving code for lab assignments from us
  • Submitting your lab work to us
git clone

You only need clone each lab once on a machine

The first time you use a new machine, you must clone the lab files for the current lab you are working on first

git clone https://URL--here.git
git add

- Used for staging files, essentially telling git that "I edited this file, I want to include it in the next snapshot of my code."

```bash
git add myfilename.py
```
git commit -m

• Commit changes to a file (that you have added) takes a snapshot of your code in files you have added and assigns it a version number

git commit -m "Message here"

• If you use the flag -am it commits changes to all files that have ever been added to the current lab repository
  • Essentially a short term to avoid adding each file individually every time you change it

git commit -am "Message here"
git push

• Sends all the committed changes on your local machine to the CS server (*evolene*)
Check the Website to See Your Edits

- https://evolene.cs.williams.edu
• If you are **resuming work on a machine** where you have already cloned the lab files (sometime in the past), always git pull first to ensure you are starting from the most up-to-date version.
Summary

• **git clone**: copy code from server to a **new** machine for the first time. Only run this once for each assignment on each machine!

• **git add `<files>`**: add new or modified files to the next commit (this basically allows you to choose which files you plan to commit)

• **git commit -m “<message>”**: create a local snapshot of the added files (this does **not** copy anything back to the server!)

• **git push**: copy changes from your machine back to our server

• **git pull**: copy latest version of code from our server to your local machine (this can only be done **after** you have run **git clone** on this machine)

• **git commit -am “<message>”**: commits an already added file (a shortcut)
Things to Remember

• We use git commands in the **Terminal**

• You need your CS account to log-in to evolvene (the CS server that hosts all your lab files)

• Always git add/commit/push before you leave lab!

• Lab instructions are on the course website
CS 134:  
Unix (Terminal) Commands
Directories in Unix

• 'Folders' on your computers are called 'directories' in Unix-based operating systems

• Your ‘current directory’ is important when executing commands on the Terminal

• For example, Python programs that run as a script, such as helloworld.py, must be in the same directory as where you execute the command python3 helloworld.py in your Terminal

• Otherwise your computer doesn’t know which program to run!

• Similarly, when you git pull, you need to be in the correct directory

• Useful to learn how to navigate between directories with the Terminal!
Useful Unix Commands

- **pwd** - print working directory
- **mkdir <dir name>** - make new directory (or folder)
- **cd <dir name>** - change directory (like moving into a folder)
- Special directory names in Unix
  - single dot, current directory
  - two dots, parent directory
  - tilde, home directory
- **cd ..** - takes you to the parent directory
- **cd** - takes you “home”
- **ls** - shows contents of current directory