Saving and Sharing your code with

Git & Github

What are they? | Common Workflows

LEARN

It might git confusing

LEARN

But…

LEARN

Don’t Panic!

LEARN

Let’s break it down:

LEARN

Overview

- What are git and Github?
- What are they used for?
- Common commands
- A common workflow
- Practice

LEARN

Git and Github are 2 different things

LEARN

The command line tool

Git

- A version control system
  - also referred to as a VCS
  - allows you to track the changes in your project
  - a project with history or versions that can be recalled
- Used in the command line

LEARN

The online hosting service

Github

- Provides hosting and git version control online
  - allows collaboration with integrated version control
  - version control through graphical user interface
  - a bunch of other tools to help manage collaboration

LEARN

Still confused?

psst. That's totally normal.

LEARN

Key points

- git = manage version control locally (on your machine)
- Github = share and collaborate on code using version control remotely (on the web)

LEARN

With that in mind…

LEARN

Most Common Commands

- **git add** (ex. => git add .)
  - used to ‘stage’ or ‘track’ changes
- **git commit -m** (ex. => git commit -m ‘adds tests’)
  - used to save changes on ‘staged’ files
- **git pull** (ex. => git pull origin master)
  - get remote changes to local copy
- **git push** (ex. => git push origin feature_branch)
  - get local changes to remote copy

LEARN

More Common Commands

- **git init**
  - makes the current folder into a git repository
- **git clone** (ex. => git clone <copied git repo url>)
  - creates a copy of the repo at the the url locally
- **git status** (super useful!)
  - logs the current status of a branch

LEARN

Useful in class

- **git config user.name**
  - logs/sets the name associated with commits
- **git config user.email**
  - logs/sets email associated with commits
- **git remote**
  - logs/set/removes the remote address for a repo

LEARN