Git and Github

Edwin Kofler

Fall 2022

Agenda

1. Introduce Git and Github
2. Tour GitHub (create account, explore site)
3. Download Git and GitHub Desktop
4. Make a GitHub Contribution
5. Bonus Points: Using Git over the Command Line
6. Resources

Introducing Git

From the official Git website:

> *Git is a free and open source distributed version control system...*

But what does that mean in practice? Git...

- Lets you "look at" a file at a previous point in time
- Allows multiple people to work on the same codebase at a time
- Lets you experiment with your code (and keep the original copy safe)

Introducing GitHub

- Sort of a combination of Dropbox and Twitter
- Navigate to GitHub
- See "Sign up" at top-right and follow prompts

Tour of GitHub

- Create GitHub account
- Explore "home" view
- Show "profile" view
- Create a "profile README"
- Show small repository: hyperupcall/website
- Show large repository: xournalpp/xournalpp

Downloading Git and GitHub

Git must be installed before using GitHub Desktop

- For **Git**
  - Go to git-scm.com
  - See middle-right part of page for download button
- For **GitHub Desktop**
  - Go to desktop.github.com (Linux users, see shiftkey/desktop)
  - See download button

Make a GitHub Contribution

1. Find a repository to contribute too (ex. ecc-cs-club/practice)
2. Find an issue to "fix" (optional)
3. Fork the repository (usually required)
4. Clone your "forked" repository to your computer with a Git client (ex. GitHub Desktop)
5. Open the repository with a code editor (ex. VSCode)
6. Make the code or text changes required
7. Make a Git commit, then push your changes
8. Create a Pull Request (PR)
9. Wait

Resources

- The Coding Train: Git
- Using Git on the command line (video)
- Git cheatsheet by GitHub
- Git cheatsheet by GitLab
- How to contribute video