Git Tutorial: Presentation Outline

Slides - Motivation for a Version Controlled Workflow and GitHub - [10 min]

Overview: Through the next few slides, I hope to motivate the use of version control in general and show some examples of Git and GitHub being utilized in practice with my own projects.

1. The general coding workflow
    - map out project on paper
    - Search via stack exchange and other online resources to identify solutions
    - Try various implementations for reaching a goal. Until you narrow in on a
      good method.
    - Use comments to explain how to use your code (for your self and others)
    - Difficult to locate old versions or manual versioning with file names:
      (e.g. `compute_1.1`, `compute_1.2`)
    - Working on multiple computers can prove very difficult even if

the code is in the
cloud. Directory names are different, line endings get messed up.
    - Dropbox or Google Drive (potentially email) for sharing code with other people.
2. The Version Controlled Workflow
    -
        - Saving of the file itself alongside commits to a local and remote repository.
3. The Version Controlled Workflow with GitHub
    - Simultaneous collaboration with as many people as desired
    - Sharing of software for release with
    - Published ReadMe to explain the purpose for your code and instructions for
      it's use.
4. What is Git
    - "Git is a distributed version control system"
    - Basically everywhere that there is a copy of your project, the entire
      change history and in-progress parts are stored as well.
5. What is GitHub
6. Consolidation Example - MATLAB Utilities
    - **Git is a framework where copy-pasting is obsolete.**
7. Collaboration Example - Seaborn Pull Request
    - **GitHub is the forum where code is improved.**
    - Not only did this solve a problem for me, but there is now a

public

version of the Seaborn software with this feature which anyone can use at

any time.

8. Released Software Example - Volume Point Picker
    - **GitHub is the platform for software to be released.**
9. Yes this is a lot, but the Energy Barrier is low.
    - You could just throw all your code in a folder and commit once a day,

      just for the versioned backups.
    - Quick and dirty example: Put existing project on GitHub, then update.
    i. `$ cd to/my/project/`
    ii. `$ git init`
    iii. `$ git add -A`
    iv. `$ git commit -m "initial commit of existing proj."`
    v. `$ git remote add origin http://github.com/CorbanSwain/Example-Proj`
    vi. `$ git push -u origin master`
    vii. `$ emacs test.py` … *edit file* … *save file.*
    viii. `$ git add -A`
    ix. `$ git commit -m "Added feature to test"`
    x. `$ git push`

Exercise 1 - Creating a Local Repository & Pushing to Github [5 min]

Overview …

Commandline

1. set config variables

```
shell $ git config --global user.name "Corban Swain" $
git config --global user.email "CorbanSwain@gmail.com"
```

Exercise 2 - Collaborating on a project through Github. [10 min]

- Begin with a local code base. Some code that you want to start tracking
- .gitignore file
- Web calculator git branch demo

Slides - More Complex (But Important) Topics

- Managing code alongside large files
- Merging conflicting commits