Developing with Python:

Part III: Git

Friday, July 18, 2014

Guang Yang

Why Version Control?

- Made a change to code, realized it was wrong and wanted to revert back
- Lost code or backup
- Wanted to submit a change to someone else’s code
- Had to maintain multiple version of a product
- Wanted to review the history of some code
- Wanted to let other people work on your code
- Wanted to experiment with a new feature without interfering with the working code

Source: Si. @ stackoverflow

Why Git

- Steady increase in popularity since 2009 (contributing to SVN’s decline in popularity)
- Creates local repositories that do not require internet to commit
- Github provides a web-based graphical interface that makes learning git very easy

Git Basics

- **git clone /path/to/repository:** creates a working copy of a local repo
- **git add <filename>:** add changes (to Index)
- **git commit –m “Commit Message”:** commit changes (to Head)
- **git push origin master:** push changes to remote repo
- **git pull:** updates the local repo to the newest commits
- **git status:** see what’s going in with the current repo

Branching & Pull Request

git checkout –b my_new_branch: creates a new branch named “my_new_branch” and switch to it

git push origin <branch>: push the branch to remote repo

git checkout master: switch back to master

git branch –d my_new_branch: delete the branch “my_new_branch”

(Demo)

Your Turn

- Download Git
- Make a GitHub account
- Start practicing!

References

- Check out my git repository, where I will be posting slides for future/past tutorials and other relevant resources.
- Here's a simple illustrative guide to common git commands
- Special thanks to Prof John DeNero for the lovely template, check out his *CS61A course* in Python