Git

A version control system

by S Dattalkar

21-Aug-23

Version control systems

- Version control (or revision control, or source control) is all about managing multiple versions of documents, programs, web sites, etc.
  - Almost all “real” projects use some kind of version control
  - Essential for team projects, but also very useful for individual projects

Why version control?

- For working by yourself:
  - Gives you a “time machine” for going back to earlier versions
  - Gives you great support for different versions (standalone, web app, etc.) of the same basic project
- For working with others:
  - Greatly simplifies concurrent work, merging changes
- For getting an internship or job:
  - Any company with a clue uses some kind of version control
  - Companies without a clue are bad places to work

Download and install Git

- There are online materials that are better than any that I could provide
  - Standard one: http://git-scm.com/downloads
- Install Git on your machine from http://git-scm.com/downloads
  - Accept context menu items
- Git access:
  - Shift+Right click from windows explorer
  - Open Powershell windows here(Click in this option)

Introduce yourself to Git

- Start git with below commands
- git init (initialize git)
- Enter these lines (with appropriate changes):
  - `git config --global user.name “username"`
  - `git config --global user.email xyz@gmail.com`
- You only need to do this once
- If you want to use a different name/email address for a particular project, you can change it for just that project

Choose an editor

- Add files to git
  - git add file_name1 file_nam2 file_name3
  - Or
  - git add . (to add all files)
- Commit changes to repository
  - Commit your staged changes in your repository
    - git commit -m "the reason for the change"
  - Add remote origin
    - git remote add origin url
    - Above command run only once

Using your repository

- Push changes to the repository
  - update data to the repository:
    - git push -u origin master
- See origin of repository
  - git remote -v
- Get what is on repository
  - git pull origin master
  - If it says to resolve manually, just vi that file and see the head which is yours

Resubmit your changes

- Put changes back up into repository
  - Commit your staged changes in your repository
    - git commit -m "the reason for the change"
  - update data to the repository:
    - git push origin master