# Assignment #01: Setup Course Tools & Git Fundamentals

**DUE:** Wednesday, 2016-09-17 before class.

1. Create an account on GitHub (`https://github.com`).
2. Download and install `git` at `https://git-scm.com`. We will be using Git Bash, *not* a GUI client.
3. Setup an SSH key to seamlessly push/pull to/from your GitHub repositories: `https://help.github.com/articles/connecting-to-github-with-ssh/`
4. Download and install `python3` at `https://www.python.org/`. Be sure to install Python 3.6, *not* Python 2.7. Note that if you are using Windows, you should consider either:
   - Installing and using the Ubuntu Linux Subsystem (Windows 10), and running `python3` from within that environment, or
   - Install Conda python from `https://www.continuum.io/downloads`.

   Using "vanilla" python on Windows can have challenges with importing some packages, such as `numpy`, which do not exist in compiled wheels for Windows.
5. You will want a code writing environment that makes life easier for you as your projects get more complex. Options include:
   - VIM [`http:\www.vim.org`] (ideal for terminal usage)
   - GitHub Atom [`https://atom.io/`]
   - Visual Studio Code [`https://code.visualstudio.com/`]
   - PyCharm [`https://www.jetbrains.com/pycharm/`] (full-featured IDE)
6. Never used git before? Start with these resources:
   - `https://try.github.io/`
   - `https://www.codecademy.com/learn/learn-git`
   - `https://www.git-tower.com/learn/cheat-sheets/vcs-workflow`
   - `http://gitimmersion.com/`
   - `https://www.atlassian.com/git/tutorials/comparing-workflows#feature-branch-workflow`
7. Familiar with git (or just completed the exercises above)? Give this a try: `http://learngitbranching.js.org/`
8. Having trouble? We'll be reviewing some of these tools in lecture on Thursday. Also checkout the Duke Co-Lab, which hosts regular office hours and has an online Slack team: `https://colab.duke.edu/` We have a specific channel on there for this class, including:
   - `#git`
   - `#python`
9. Learning Objectives:
   - Create a git repository on your local computer.
   - Create a local file, then add and commit it to your local repository.
   - Edit your local file, adding and committing those edits.

- Create a remote repository on GitHub that has the same name as your local repository.
- Add the remote repository (origin) URL to your local repository.
- Push your local repostiory to GitHub.
- Create a local branch, create/add/commit a new file.
- Merge new local branch commit(s) into local master.
- Push updated master branch to GitHub.