

# GitExercises

February 21, 2015

## 0.1 Using Git and GitHub

This tutorial only covers the *bare* basics. When you are comfortable using Git in the basic way shown in this tutorial, there are plenty of online and dead tree resources for more advanced Git usage that you can learn from.

### 0.1.1 Basic concepts to understand

- Version control
- Git versus GitHub
- Working directory, staging area and snapshot archive
- Remotes and origin
- Branches and master

### 0.1.2 Useful references and further reading:

- [Basic Git commands](#)
- [Git cheatsheet](#)
- [A more advanced tutorial from the creator of IPython notebook](#)
- [Git for scientists](#)
- [Pro Git book](#)

**Exercise 1.** Create a repository on GitHub called ‘monster-cookies’ and clone to your local drive.

In [] :

**Exercise 1.1.** Set your username and email if necessary.

In [] :

**Exercise 2.** Add two files `phd_thesis.txt` and `java_manuscript.txt` with the contents `My very short thesis.` and `$$\pi = 3$ almost everywhere.` respectively to your local repository. How do you check the status of your local repository? Add and commit your changes. After doing so, check the GitHub web page.

In [] :

**Exercise 3.** Modify `phd_thesis.txt` by adding another line `This is a very good thesis.`

- You decide that you don’t want these changes but have not done `git add`. What do you do?
- You decide that you don’t want these changes but have just done `git add`. What do you do?
- You decide that you don’t want these changes but have just done `git commit`. What do you do?
- You decide that you don’t want these changes but have just done `git push`. What do you do?

In [] :

**Exercise 4:** Modify “`java_manuscript.txt`” by replacing the number 3 with 22/7. How do you find what is different from the working copy of the file and the current snapshot in the repository? Add, commit and push your changes.

In `[]`:

**Exercise 5.** You accidentally type `rm phd_thesis.txt` and realize you haven’t backed up your laptop in 3 years. What do you do?

In `[]`:

**Exercise 6.** You accidentally deleted the `monster-cookies` folder on your hard drive with `rm -rf monster_cookies`. What do you do?

In `[]`:

**Exercise 7.** Your laptop falls into the lake at Sarah P Duke gardens and is eaten by a turtle. What do you do?

In `[]`:

**Exercise 8.** You realize that  $\pi = 3$  after all, and want to go back to the version before you made the 22/7 switch. What do you do?

In `[]`:

**Question.** What does `java_manuscript.txt` look like on GitHub now?

In `[]`:

**Exercise 9.** You want to show off your productivity to your thesis advisor. What do you do?

In `[]`:

**Exercise 10.** You run the following Python command from a `make_cookies.py` script

```
for i in range(10):  
    open('f%02d.dat' % i, 'w').write('This contains intermediate results #%d' % i)
```

Now you have lots of `.dat` files that you don’t want to commit but still wish to keep in the folder. What do you do?

In `[]`:

**Exercise 11.** You modify `phd_thesis.txt` by adding `This thesis is super awesome`. You modify `java_manuscript.txt` by replacing 3 with 3.14. Add, commit and push each file so that each file has its own commit message (`making my thesis more awesomer` and `international pie day is March 14` respectively).

In `[]`:

**Exercise 12.** You decide to collaborate on `java_manuscript.txt` with your friend (choose the person on your left or right, depending on who you think is better looking). You want to grant write permission to your friend so that he/she can make changes to your repository. What do you do?

In `[]`:

**Exercise 13.** Your friend changes 3.14 to 2.718 in `java_manuscript.txt` and commits the changes to GitHub (add, commit, push). At the same time, you want to change  $\pi$  to  $e$  so that the final version in GitHub reads  $e = 2.718$ . What do you do?

In []:

**Exercise 14.** You are flying back to RDU on a 34 hour plane trip with no internet access when you have a brilliant idea that you want to record. You pull out your laptop, create a new repository called `my-first-original-thought`, write down your thoughts in `brilliant.txt`, then add and commit it to your local repository. When you arrive, you put your local repository on GitHub so that it can be archived and shared. How did you do this?

In []:

**Exercise 15.** Your JASA manuscript is accepted after all 3 reviewers acclaim its brilliance with comments like “I never realized the most common character in the English language had a numeric value!”. You want to tag the current (accepted) version of the manuscript as 1.0 with the annotation “Version accepted by JASA”. What do you do?

In []:

## 0.2 Branching and merging (optional)

**Exercise 16.** You decide to add some new experimental code to `make_cookies.py` that you are not sure will work. In order to preserve the structure of the existing repository, you decide to use *branching* in Git. Create a new branch called `hack-01` and switch to that branch.

In []:

**Exercise 17.** Modify `make_cookies.py` in the `hack-01` branch so that the new code reads as follows:

```
if __name__ == '__main__':
    for i in range(10):
        open('f%02d.dat' % i, 'w').write('This contains intermediate results #%d' % i)
```

Add and commit the changes.

In []:

**Exercise 18.** At the same time, you decide to add new results to `phd_thesis.txt` in the master branch. Switch back to the master branch and Append the following line to your thesis `This thesis is now complete..` Add, commit and push the changes.

In []:

**Exercise 19.** You are happy with your changes in `hack-01` and want them to be incorporated into the main (master) branch. Merge the `hack-01` branch with the master branch. Check that the changes have been incorporated, then push to GitHub.

In []:

**Exercise 20.** Now that the changes have been merged, `hack-01` is no longer needed. Delete the `hack-01` branch.

In []: