

Git and GitHub problem set

Kyle Gorman

In this problem set you will practice:

- Using Git locally
- Creating a remote GitHub repository and uploading to it
- Submitting a "pull request" to a remote repository owned by others

These are likely to be unfamiliar behaviors so the goal is to get a feel for how these things are done.

Working locally

Problem: Create a local git repository called "GitPractice".

Solution:

```
[~]$ mkdir GitPractice
[~]$ cd GitPractice/
[GitPractice]$ git init
Initialized empty Git repository in /home/kbg/GitPractice/.git/
```

Problem: Make a file called `AUTHORS` which contains your name, then add and commit it to the repository created in the previous step.

Solution:

```
[GitPractice]$ code AUTHORS
# Or whatever text editor you like.
[GitPractice]$ git add AUTHORS
[GitPractice]$ git commit -m "Added my name to AUTHORS"
[master (root-commit) 3c3aec5] Added my name to AUTHORS
1 file changed, 1 insertion(+)
create mode 100644 AUTHORS
```

Problem: Add the name of the person sitting to your left or right to `AUTHORS`, and update the Git repository as before.

```
[GitPractice]$ code AUTHORS
```

```
# Or whatever text editor you like.
[GitPractice]$ git add AUTHORS
[GitPractice]$ git commit -m "Added Hussein's name to AUTHORS"
[master 703ab42] Added Hussein's name to AUTHORS
Date: Mon Feb 18 15:41:47 2019 -0500
1 file changed, 1 insertion(+)
```

Problem: Now, remove the AUTHORS file from the repository.

```
[GitPractice]$ git rm AUTHORS
rm 'AUTHORS'
[GitPractice]$ git commit -m "Removed the AUTHORS file"
[master 07061bc] Removed the AUTHORS file
1 file changed, 2 deletions(-)
delete mode 100644 AUTHORS
```

Working remotely

To prepare for this component, remove the GitPractice directory like so:

```
[GitPractice]$ cd ..
[~]$ rm -rf GitPractice
```

Problem: Create a remote repository on GitHub and make a local clone.

Solution:

1. Go to <https://github.com/> and log in if necessary.
2. Click on the green "New" button.
3. Fill in the "Repository name" field with the name `GitPractice` then click on the green "Create repository" button.
4. Instructions will pop up for setting up the repository under the heading "...or create a new repository on the command line", which will give you a command for setting your local repository to track the "remote" on GitHub. Assuming you are currently inside the `GitPractice` directory on the command line:

```
[~]$ mkdir GitPractice
[~]$ cd GitPractice
[GitPractice]$ git init
Initialized empty Git repository in /home/kbg/GitPractice/.git/
# This next command will vary depending on your GitHub username and on
```

```
# the name you give the repository. I copied this directory from the
# GitHub "...or create a new repository on the command line"
# instructions.
[GitPractice]$ git remote \
    https://github.com/kylebgorman/GitPractice.git
```

Problem: Create a Python script called `hello.py` which simply prints out `hello, world`, and add it to your local repository. Then, push the changes to the remote repository.

```
[GitPractice]$ code hello.py
# Or whatever you like. The script reads:
#
# print("hello, world")
[GitPractice]$ git add hello.py
[GitPractice]$ git commit -m "Adds hello-world script"
[master (root-commit) b2a3a25] Adds hello-world script
1 file changed, 1 insertion(+)
create mode 100644 hello.py
[GitPractice]$ git push origin master
Username for 'https://github.com': kylebgorman
Password for 'https://kylebgorman@github.com':
Counting objects: 3, done.
Writing objects: 100% (3/3), 247 bytes | 0 bytes/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To https://github.com/kylebgorman/GitPractice.git
* [new branch]      master -> master
```

Problem: Kyle has created a public GitHub repository called `PracticumAttendance`, located at:

<https://github.com/methods-in-computational-linguistics-2/PracticumAttendance>

When you copy a public repository on GitHub, you "fork" it. Fork it, and clone it.

Solution:

1. Visit the above URL.
2. Click on the "Fork" button on the top right.
3. Once this is complete you will be taken to your remote fork's page. Click on the green "Clone or download" button on the right, and copy the text.
4. Then clone your fork:

```
# Note the URL here is copied from step 3; it will vary depending on
your GitHub username.
```

```
[~]$ git clone https://github.com/kylebgorman/PracticumAttendance.git
Cloning into 'PracticumAttendance'...
remote: Enumerating objects: 4, done.
remote: Counting objects: 100% (4/4), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 4 (delta 0), reused 4 (delta 0), pack-reused 0
Unpacking objects: 100% (4/4), done.
Checking connectivity... done.
```

Problem: In your local fork of PracticumAttendance, create a new branch called name. In this branch, add your name to attendance.txt and push it to the remote fork.

Solution:

```
[~]$ cd PracticumAttendance
[PracticumAttendance]$ git checkout -b name
Switched to a new branch 'name'
[PracticumAttendance]$ code attendance.txt
# Or whatever.
[PracticumAttendance]$ git add attendance.txt
[PracticumAttendance]$ git commit -m "Added my name"
[name e9330f2] Added my name
 1 file changed, 1 insertion(+)
# Note that here we use `origin name` instead of `origin master` since
# we're sending the `name` branch.
[PracticumAttendance]$ git push origin name
Username for 'https://github.com': kylebgorman
Password for 'https://kylebgorman@github.com':
Counting objects: 3, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 297 bytes | 0 bytes/s, done.
Total 3 (delta 0), reused 0 (delta 0)
remote:
remote: Create a pull request for 'name' on GitHub by visiting:
remote:
https://github.com/kylebgorman/PracticumAttendance/pull/new/name
remote:
To https://github.com/kylebgorman/PracticumAttendance.git
 * [new branch]      name -> name
```

Problem: File a pull request for your name branch.

Solution:

1. In the printout above, Git prints out a suggestion for how to file a pull request for this branch; visit that URL.
2. Type a description of your pull request ("added my name" would suffice) where it says "Leave a comment", then press the green button labeled "Create Pull Request".

This is the standard method used when you are submitting "patches" or "fixes" to open-source projects hosted on GitHub.