

# How I Use GitHub at the Command Line

Ryan T. Moore\*

12 September 2019

You would like to contribute to a project that's stored on GitHub. You have a GitHub account and are comfortable at the command line. I use iTerm2 and Terminal on macOS.

1. Clone the repository. From your Desktop (or wherever you want the repo to live),

```
git clone git@github.com:username/reponame.git
```

For example,

```
git clone git@github.com:ryantmoore/mytestrepo.git
```

2. Check that you're on the correct branch. Sometimes, you might commit to **master**:

```
git branch
```

3. Each work session, I start by making sure there's nothing in my local directory that I've changed, and nothing on the remote that I need to incorporate. For the first,

```
git status
```

Then, if I have no local work to commit, I

```
git fetch
```

```
git status
```

```
git pull (if needed)
```

```
git status (if needed)
```

If instead I do have local work to commit, I follow [6](#) below, then do the above.

4. Make your edits to the local files in your text editor, IDE, etc. Save your edits.

5. Add your work to the next commit:

```
git add <files>
```

6. Commit your work. A *commit* is a snapshot of the repo at a moment in time. A commit should be minimal and complete. To add a commit message easily, use **-m**:

```
git commit -m "Fix typos"
```

7. When you're ready to send your commits back to the remote on GitHub,

```
git push
```

Hope this helps!

---

\*Department of Government, American University, Ward Circle Building 226, 4400 Massachusetts Avenue NW, Washington DC 20016-8130. tel: 202.885.6470; fax: 202.885.2967; rtm (at) american (dot) edu; <http://ryantmoore.org>.