

Week 8 — Git On

Z620: Quantitative Biodiversity, Indiana University

March 6, 2015

OVERVIEW

A major goal of Quantitative Biodiversity has been to provide you with tools to conduct reproducible science. One of these tools is Github. As you know, Github allows one to manage changes to data and code to help ensure the integrity of a research project. Github is great for version control, but is also valuable for collaborating with other researchers. Up to this point, we've been using an Enterprise version of Github. Our Github Enterprise is restricted to people with an IU affiliation. Obviously, this may hinder collaboration with your colleagues at other institutions. In this handout, we present a workflow that will assist you reproducible science upon “graduation” from Quantitative Biodiversity.

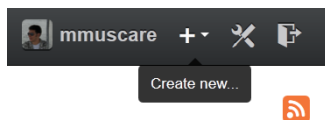
After completing this exercise you will know how to:

1. Create a source-code file
2. Open a github.com account (not restricted to IU)
3. Create a Github repository
4. Create and modify a README.md file
5. Properly license your work

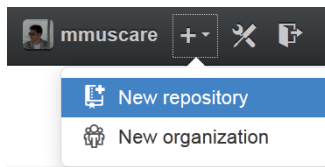
1) Make QBTools Repository

If you are going to use Github, and you start a new project, you are going to want to create a repository. For example, this semester, you have been working with a repository that we created for you (2015-QuantitativeBiodiversity). On the first day of class, you forked and cloned this repository. Now, we are going to go through the steps of creating your own repository:

1. Open an web browser and navigate to <https://github.iu.edu/>
2. At the top of page there is black banner. On the right-ahdn side there is “+” symbol. If you hover over this symbol, it will say “Create new...”.



3. Click on the symbol and select “New Repository”



4. Give your repository a name (e.g., QBTools)
5. Decide if you want your repository to be accessible to all IU affiliates (Public) or not (Private).

6. Click the button that says “Initialize this repository with a README”
7. Optional: you can click on the “Add .gitignore” button and select R. This will...

Owner: mmuscare / Repository name: QBTools

Great repository names are short and memorable. Need inspiration? How about [secret-octo-lana](#).

Description (optional)

☒ Public
Any logged in user can see this repository. You choose who can commit.

☐ Private
You choose who can see and commit to this repository.

☒ Initialize this repository with a README
This will allow you to `git clone` the repository immediately. Skip this step if you have already run `git init` locally.

Add .gitignore: R

Create repository

8. When you have completed the items above, you will be able to click on the green button that says “Create repository”

Create repository

9. Now, we are going to clone your new repository to your local machine

```
cd ~/GitHub
git clone https://github.iu.edu/User_Name/QBTools
cd ./QBTools
git status
```

10. At this point you are ready to go, but it is always good to check your remotes. You should have just one remote: origin

```
git remote -v
```

1) CREATE SOURCE CODE FILE

Last week we learned the benefit of using a R source code file. As a reminder, a source code file has user-defined functions that are required for certain analyses. The benefit of source files is that they contain “vetted” code that can be used across multiple projects. Last week, that file contained code for importing files created by the popular community sequencing software [mothur](http://www.mothur.org/) (<http://www.mothur.org/>). This week, we will make our own source code file containing code (functions) that we created during QB.

1. Create a blank R file.
2. In your menu bar, choose File > New File > R Script (alternatively, Ctrl + Shift + N)

3. Add a file header.

This should be a commented out section that give all of the required information for users to use this script. Additionally, this may contain notes for the user.

Here is an example header:

```
#####
#                                                                 #
# Quantitative Biodiversity Functions Source Code                #
#                                                                 #
# Written by: Student X                                         #
#                                                                 #
# Last update: 2015/03/06                                       #
#                                                                 #
# Notes:                                                         #
#                                                                 #
#####
```

There are a few things to notice about the header:

1. The comment character (#) prevents R from interpreting the information in the header
2. The header is formatted for stylistic purposes; it is 80 characters wide (as should your file), and it is boxed off using comment characters.

There are other ways of stylizing your header, this is just one option.

4. Add content A. The first thing you will want to do is add and require commands. This is a bit different than before because we need to make sure that the packages are installed. By default, the following code will try to load a package and will then install if not found.

```
require("vegan")||install.packages("vegan");require("vegan")
```

You can cusotmize this code for any package that you want loaded for our source file

B. Add functions: The major goal this week is to make a source file that contains all of the functions we wrote this semester. So now you are going to populate this file with those function. Go through all of your assignments and find the functions you want to save. Copy those functions into this new source file. Add comments about each function before the code. For example:

```
# My_Function: has some cool code
# Inputs: x, y, z

My_Function <- function(x, y, z){
  ...
}
```

2) CREATE A GITHUB.COM ACCOUNT

3) MAKING a GITHUB REPO

So far this semester you have been working with a repository that was created

Make students choose “public”. If they don’t feel comfortabel with this for whatever reason, they have the following options

- 1) don't migrate your files
- 2) choose private at which time Github is likely to ask for your credit card information
- 3) make public for now and then request private repos from github

4) CREATE A README.md FILE

We talked about supplying them with a template, which might have info such as description, software versions, author information, contact information, bugs/files/patches, etc.

In the beginning of the course, we had you edit a README.md file, which gave us some information about you. In general, README files provide:

- 1.) More or less detailed information on the content of a repository
- 2.) The version of programs and software, data, etc. needed to run the code
- 3.) Contact information for the owner of the repository (e.g., email)
- 4.) Potential bugs

This last one is commonplace among people who develop and distribute software (e.g. ecologists who develop R packages) and among folks who engage seriously in reproducible science.

5) MIGRATING TO GITHUB.COM

notes, add: 1. how to setup 2. create a repo... GitHub will ask whether you want to add a README.md. 3. add a license... should the licensing section be moved here, as a subsection? 4. add source file

6) LICENSING YOUR CODE

(this should come after "MIGRATING" because you can't automatically license code on IU's GitHub. If you figure out how to add a license, it won't mean much because IU probably owns whatever you put on their Enterprise, i.e., IU rights supercede your license)

When setting up public repositories on GitHub, you will want to license your work because it will be in the public domain. Licenses are what allow you to establish the right to waive responsibilities and liabilities, to attract copyright, and to legally establish intellectual primacy. Licensing allows us to be visible, transparent, and fearless professionals; protected by putting our ideas and work into the public domain while maintaining rights and restrictions to (re)use and (re)distribution.

GitHub offers a broad selection of licenses that range from short and tidy (MIT) to the thorough but lengthy (GPL V3); see <http://choosealicense.com/>. 9 Simple ways to make it easier to (re)use your data. There's lots of goodies of in it, the repo which is found here: <https://github.com/weecology/data-sharing-paper>.