**Agenda**

**Stock Assessment Tempalte with Git and R markdown**

NWFSC

Seattle, WA

October 25-26, 2016

**Prior to Tuesday, October 25**

Please work through the software installation below. The irony of an agenda written in Word did not escape me ☺ Feel free to email with questions prior to Tuesday.

**Tuesday, October, 25, 2016**

**8:30 am:**  Gather in 370W, install software as needed, set up G-hangout, snacks and coffee

**9 am – 12:30 pm:**  Basics of Git and GitHub and R markdown

**12:30 - 1:30 pm**: Lunch / Melissa will be available to fix remaining installation/rendering issues

**1:30 – 5:00 pm:**  Continue / Finish R markdown basics and start template demo

* Template Workflow
* Document structure
* R code

Post-workshop dinner?

**Wednesday, October 26 2016**

**8:30 am** Gather in 370W, set up G-hangout, snacks and coffee

**9am -12:30 pm**  Continue working with the template

* Tables
* Figures
* References
* Add in your own assessment

**12:30 -1:30 pm**Lunch on your own /Melissa will be available

**1:30-3pm:** Work session / Melissa will be available

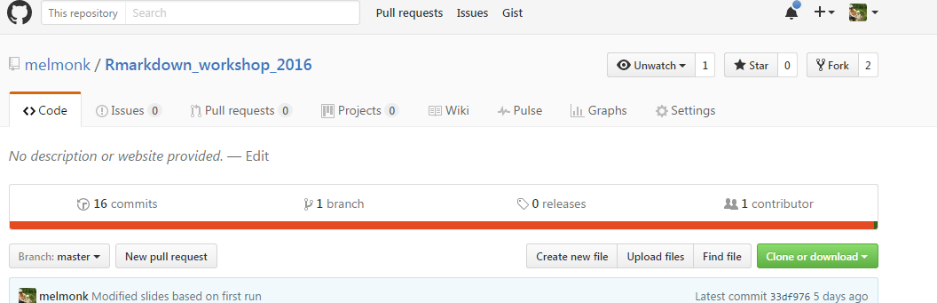
**Pre-workshop software installation and set-up**

Software installation and set-up with Git and RStudio can be painful (especially if you don’t have permissions on your computer). Do NOT download RStudio Preview – it contains a bug and the template will crash the program.

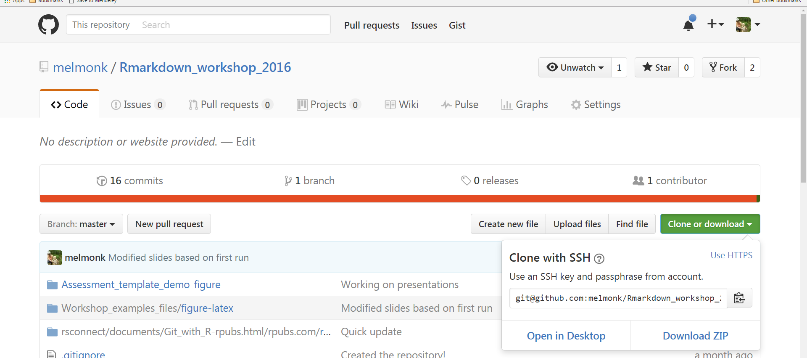
Please work through Section 2.2 of the ReadMe file found [here](https://github.com/melmonk/StockAssessment_template/blob/master/ReadMe/ReadMe.pdf), and then try and work through the below steps. It’s a little backwards working through this prior to the workshop, but this workflow will eventually be second-nature. The actual assessment template now lives [here](https://github.com/melmonk/StockAssessment_template/blob/master/ReadMe/ReadMe.pdf).

All of the workshop materials live on my GitHub page in the [Rmarkdown\_workshop\_2016](https://github.com/melmonk/Rmarkdown_workshop_2016) repository, including the presentations (authored in R presentations).

1. Navigate to the [Rmarkdown\_workshop\_2016](https://github.com/melmonk/Rmarkdown_workshop_2016) repository
2. Click Fork (this makes a copy of the repository on your account)

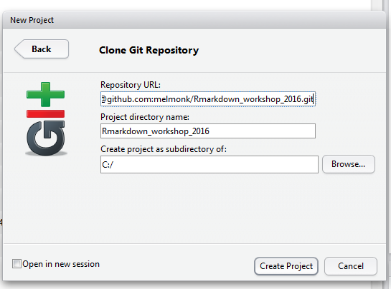


1. GitHub automatically takes you to the forked repository on your account. Now click the green Clone or Download button and click the little clipboard. You can choose to clone with SSH (if you set that up) or with HTTPS.



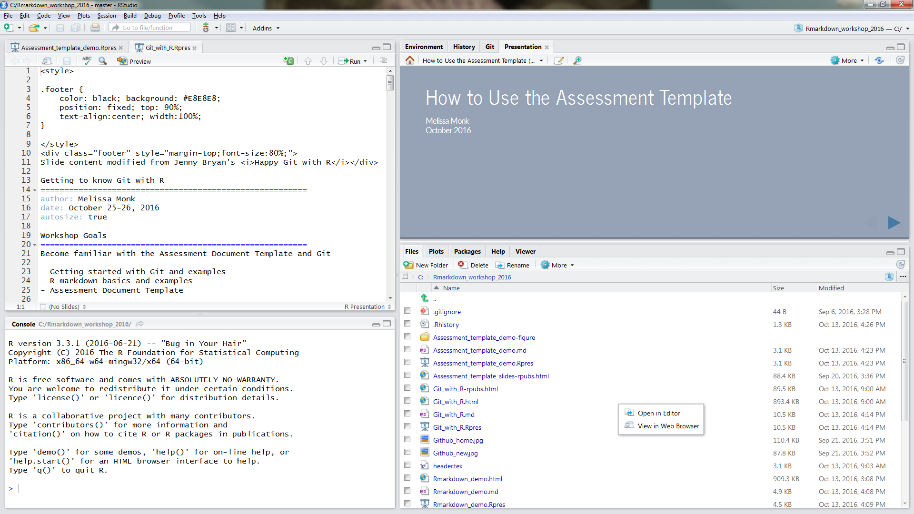
1. Open RStudio. Navigate to File -> New Project. Select Version Control, then Git. Paste the link you copied from GitHub in the Repository URL box. You can Create the project in any subdirectory.

Take charge of -- or at least notice! -- the local directory for the Project. A common rookie mistake is to have no idea where you are saving files or what your working directory is. Pay attention. Be intentional.



1. If all goes well, you now have the workshop materials. The .Rpres files are the presentations, which also have an associated .Md file and .html file. If you want to preview the presentations, use the file navigator in the bottom right and find one of the presentations, e.g., GIt\_with\_R.html. Click on it and you have the option to view it in the Editor or in the web browser.

In the workshop, we’re actually going to open the .Rpres file, e.g., Git\_withR.Rpres, and then click preview (in green below). This pops up the presentation in the top right panel of RStudio. You can simultaneously work on the presentation and the demo examples in a different file.



I will likely update the presentations prior to the workshop. If you notice GitHub telling you you’re behind the master branch, and want to update your fork, follow these instructions:

1. Open your fork repository on GitHub.
2. Click on Pull Requests.
3. Click on New Pull Request. By default, GitHub will compare the original with your fork, and there shouldn’t be nothing to compare if you didn’t make any changes.
4. Change the "Base" drop down's so both point to your fork and then you'll get a prompt to "Compare across repos" (if no changes were made in the fork) or click Edit and **switch the base manually**. Now GitHub will compare your fork with the original, and you should see all the latest changes.
5. Click on Create to create a pull request for this comparison and assign a predictable name to your pull request (e.g., Update from original).
6. Click on Send pull request.
7. Scroll down and click Merge pull request and finally Confirm merge (*If your fork didn’t have any changes, you will be able to merge it automatically*).