

Step by step instructions for getting set up with the RR/NR codebases, and setting up a source code repository of your own important work (see Appendix A.4 of [RR](#) for why...):

1. Sign up for an account at <https://github.com/signup>
2. Install Github Desktop (<https://desktop.github.com/>) on your computer.
3. Use Github Desktop to clone the RR codebases on your computer:
 - A. Open Github Desktop.
 - B. Under the File menu, click **clone** repository, and click the URL tab in the right column.
 - C. For the Repository URL, paste (exactly!): <https://github.com/tbewley/RR.git>
 - D. For the Local Path, choose a simple name for a new (initially empty) directory (e.g., put it in your home directory, let's say it is "checkers", so you can find it easily)
 - On a Mac, probably use something like: /Users/checkers/RR
 - On a Windows machine, probably use something like: C:\Users\checkers\RR
 - E. Click the **clone** button.
4. Repeat step 3 for the NR codebase, available at: <https://github.com/tbewley/NR.git>
5. Now, create a new repository for the stuff you create for all your classes.
 - A. Go back to Github Desktop.
 - B. Under the File menu, click **New Repository**
 - C. For Name, type **Classes**. For Description, again type **Classes** (or something similar)
 - D. For Local Path, choose a simple name for a new (initially empty) directory (e.g., put it in your home directory, so you can get to it easily) (for the love of all that is holy, do NOT put spaces in directory or file names!)
 - On a Mac, probably use something like: /Users/checkers/classes
 - On a Windows machine, probably use something like: C:\Users\checkers\classes
 - E. Turn on the button for "Initialize this repository with a readme".
 - F. Click the **Create Repository** button.
6. Now, put some stuff in your new directory, and save it to the repository:
 - A. Navigate to the new classes directory that you just created on your computer.
 - B. Create a subdirectory within this directory for each class you are currently taking.
 - C. Navigate to the subdirectory relevant for this class, and create a simple text file named, e.g., **hw0.m**, edit this text file, and put the command **disp('hello world')** into it.
 - D. Open Matlab, change directories to this particular subdirectory, and run hw0.
 - Debug this amazing new code, if necessary, until it runs properly.
 - E. Finally, save the stuff you created in your repository:
 - Go back to Github Desktop.
 - In the upper-left corner of the window, under Current Repository, select **Classes**
 - In the lower-left corner, write a short Summary of progress (e.g., **debugged hw0.m**)
 - Click **Commit to main**, then click **Push origin**
7. Now, every time you start a work session on your computer:
 - A. First, open Github Desktop.
 - B. Under Current repository, click **RR**
 - Click **Fetch Origin**, then if Changes are detected click **Pull origin**
 - C. Repeat step B for **NR** and **Classes**
8. Navigate to classes, and create/edit the file(s) you want to work on, like **hw1.m**
9. Every time you finish a work session on your computer, upload progress back to repository:
 - Go back to Github Desktop.
 - Under Current repository, select **Classes**
 - Write short Summary of your recent progress (e.g., **started hw1.m**)
 - Click **Commit to main**, then click **Push origin**

Step by step instructions for initializing your Matlab path to more easily run the RR/NR codebases (see Appendix A.5 of [RR](#) for why...):

1. Open Matlab
2. Type the command **userpath** and copy the name of the directory that it returns.
3. IN THAT DIRECTORY (important!), edit a file named startup.m (create one if it doesn't exist)
 - A. At a minimum, on a Mac, put the following two lines in this file:
RRbase= '/Users/checkers/RR'; cd (RRbase); RR_path_init
NRbase= '/Users/checkers/NR'; cd (NRbase); NR_path_init
or on a Windows machine, put the following two lines in this file:
RRbase= 'C:\Users\checkers\RR'; cd (RRbase); RR_path_init
NRbase= 'C:\Users\checkers\NR'; cd (NRbase); NR_path_init
(of course, change the directory names to those used in steps 3-4 on previous page)
 - B. At your discretion, add additional lines that might be useful to your startup.m file.
A couple of suggestions:
RR_physical_constants
cd /Users/checkers/classes
(of course, change the directory name above to that which you'd like to start out in)
4. Finally, restart Matlab. You should be good to go. :)

To suggest corrections/improvements to these instructions, please text me at 858.997.8369.
Thanks! - Tom Bewley