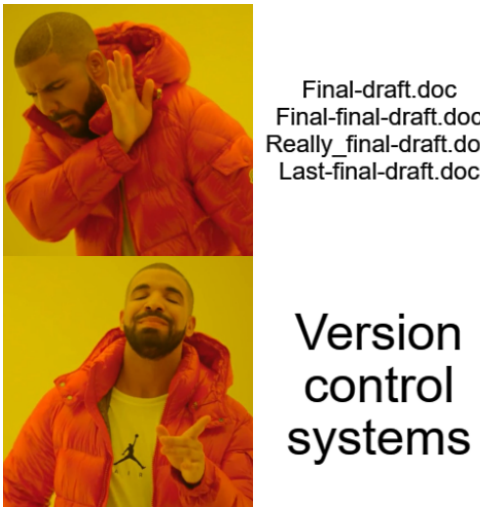
GitHub: A Gentle Introduction

Dr. Aline Coutinho

About version control systems

Version control are systems that record changes made to a file. Think, for example, of the track changes feature of Word documents. You may experience feeling overwhelmed with so many copies of the same file. Version controls record changes to the file in an organized way. You won't be working with multiple copies, but with a single updated one.



Version control systems keep a single updated version of a file, but you are still able to access previous versions, if need be. This is a powerful tool, especially if you work on long-term projects and in collaborative teams.

Here are some benefits of version control systems:

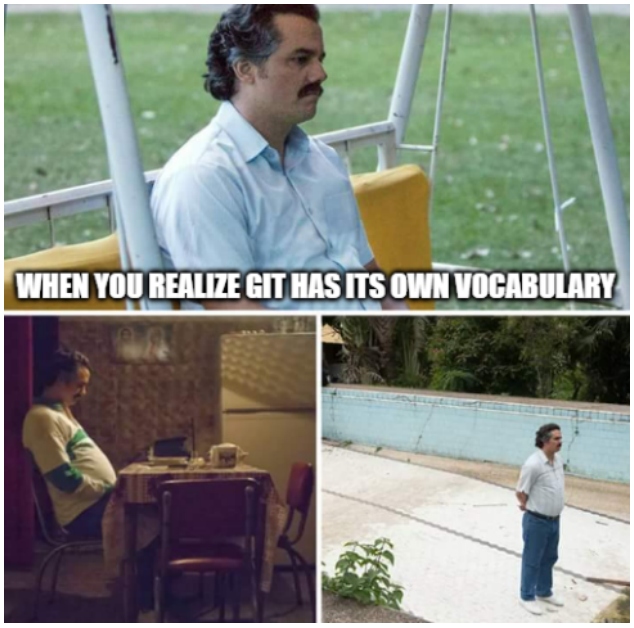
1. They record all changes to a file, including who changed the file and when changes were made;
2. They allow a better integration of all versions and edits of a file. This is especially important for those working with collaborators;
3. They allow working offline and then sync all files once you return online;
4. They are a good tool if you are developing or testing new code, especially if you want to identify sources of error messages.

Git is a type of version control systems that we will be using.

GitHub

GitHub is an online interface for Git. It hosts the files and records any changes made to the file.

It is quite common to feel lost when starting to learn Git. That is because Git has its own vocabulary that is not immediately obvious to lay persons. I felt like the guy in the meme below when I first started using GitHub. To be honest, I still feel like that sometimes, but it does get better - that I promise!



The following are some terms you might encounter sooner than later, and what they mean in the GitHub interface:

Repository or repo: the folder directory;

Commit: saves edits or changes made;

Push: updates the repo with your edits. Everybody else with access to the repository will have access to your edits;

Pulling: updates your local version with the edits others have done to the repository;

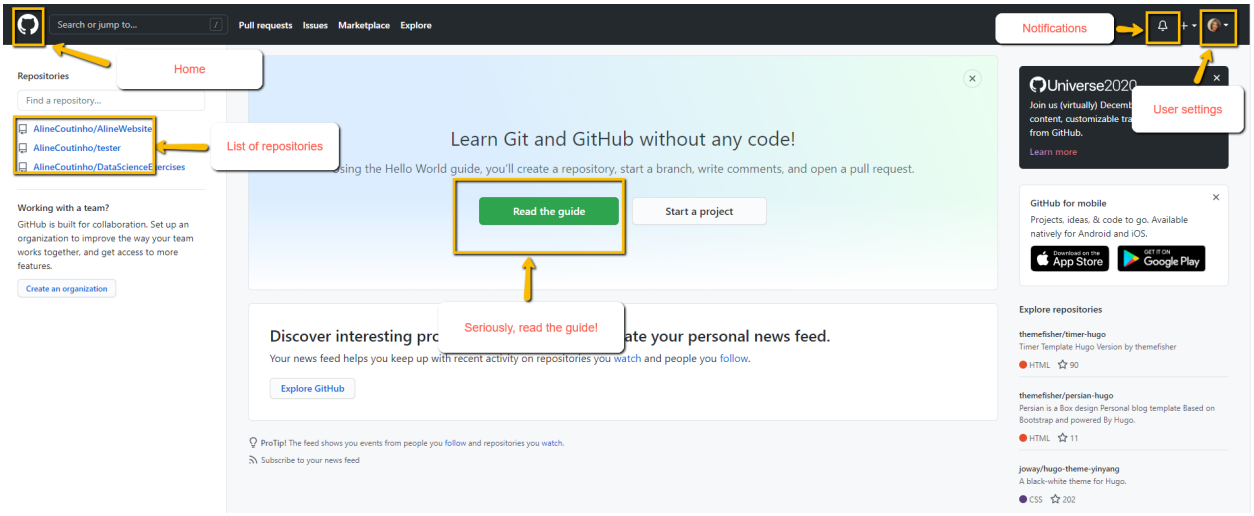
Staging: refers to preparing a file for a commit;

Clone: refers to making a copy of an existing Git repository, either yours or somebody else's;

Fork: it is a personal copy of a repository so you can explore the project without making changes to the original repo.

How to sign up for GitHub?

First, you must visit the address https://github.com/. In their homepage, where you should fill in your information and click *Sign up for GitHub*. Once you have signed up and logged in, you will find the following window:



In the homepage, you will find:

- A link to the homepage at the upper left corner;
- User settings in the upper right corner;
- Notifications, also on the upper right corner, where you can find messages and notifications for all the repositories, teams, and conversations you are part of;
- The GitHub guide, a useful introduction to GitHub which I also included at the end of this tutorial.
- A list of all your repositories on the left of your screen (it will be empty when you first sign up for GitHub).

Downloading Git on your device

In order to link the GitHub and RStudio, you need to download and install Git on your device.

First, go to https://git-scm.com/download. You should arrive at this webpage:

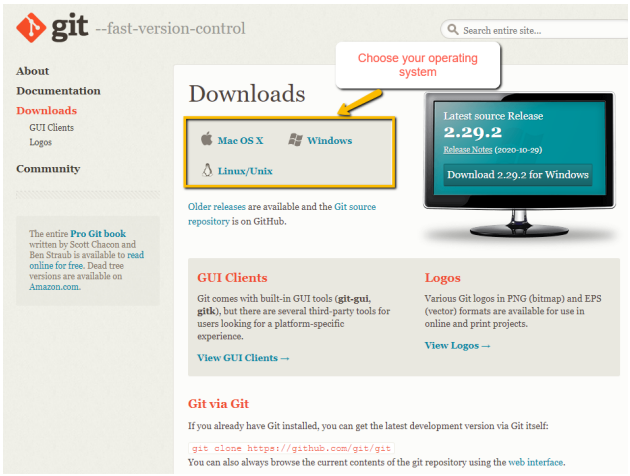


Figure 1: The Git homepage

Choose your operating system to download Git. Follow the usual prompts.

Once the download process is finalized, a command environment will open. You will find a *Start* menu shortcut to launch *Git Bash* (on Windows) in the future.

Essential steps to configure Git

After installing Git, you need to configure it in order to link it with RStudio.

First, you need to tell Git your username and email address. In the command prompt, *GitBash* or *Terminal*, depending what operating system you use), type the code:

```
git config --global user.name "insert your username here"
```

Then, type the following code in the command prompt:

```
git config --global user.email (your email, but without parenthesis)
```

If you want to confirm your configuration settings, type the following code:

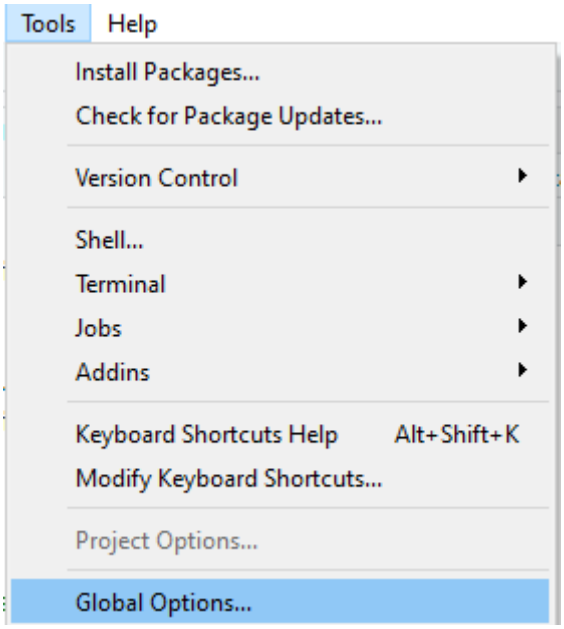
```
git config --list
```

In the last two lines, you will find the the user name and the user email you have chosen.

Important: You must use the same email you signed up for GitHub

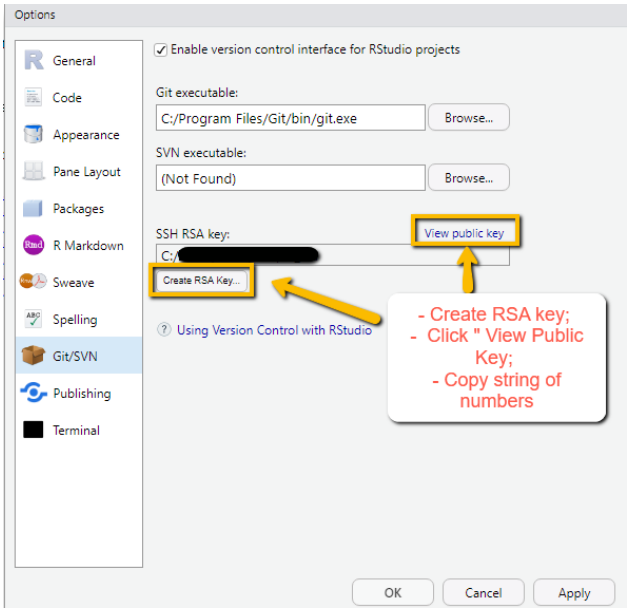
Linking Github and RStudio

The last step of this tutorial is to link GitHub and RStudio. To do so, go to *Tools* in RStudio. Click on *Global Options*, then *Git/SVN*



Choose Git as your version control system. Click *OK* or *Apply*. RStudio and Git are now linked.

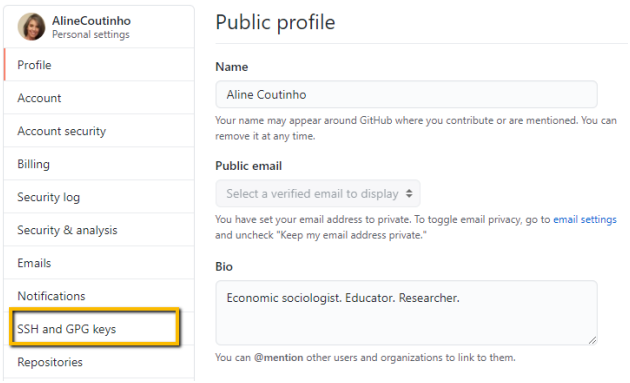
Now, click *Create RSA Key* and then click *Close*.



Then, click *View public key* and copy the code.

Finally, generate an RSA key. This key is specifically to your RStudio and you must inform GitHub so you can commit changes from RStudio.

To do so, go to https://github.com/. In your account settings, click *SSH and GPG keys* and then click *New SSH key*. This is where you paste the public key you have copied from RStudio. Confirm.



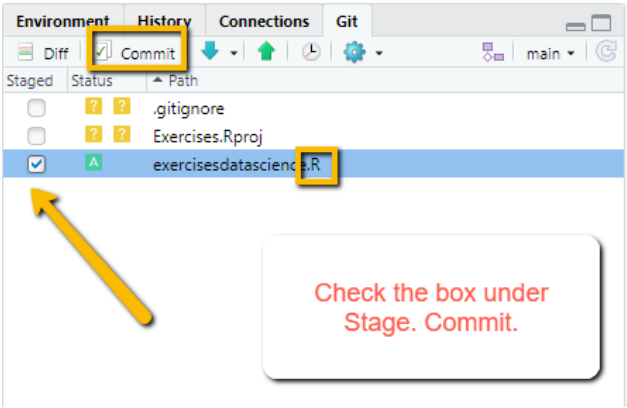
Your GitHub account and RStudio are now linked.

Create a repository on GitHub and edit in RStudio

On GitHub, create a new repository. Copy the URL for your new repository. Then go to RStudio ans create a new project (*File* > *New Project*). You should then select *Version Control*. Remember to use Git as your version control software. Paste in the repository URL from before, and then select the location to store your project. Click on *Create Project*. And voila! You have a project linked to your GitHub repository.

You can create a new R script (*File* > *New File* > *R Script*), write whatever code you need, and save the file.

In the Git tab of the *environment window*, you will see the file you created. Note that the R script ends with `.R`. Click the checkbox under *Staged*. Click *Commit*. In the new window that opens, you will see all the changes in the file. Write a message in the *Commit message* box and click *Commit*. Close the window.



Don't forget to **push** your changes to the GitHub repository. You can do so by clicking the green upper faced arrow.

Congratulations! You are all set to produce reproducible research.

Additional resources

GitHub guide

This online course on Coursera is very informative