

**GitHub Exercices**  
**Club iEES-Paris**  
**1/12/2021**  
**Rosalie Bruel & Jerome Mathieu**

**Resources for an in depth understanding of Git :**

<https://gist.github.com/peterhurford/4d43aa5d6de114c0c741ba664c9c5ff5>

<https://docs.github.com/en>

glossary : <https://docs.github.com/en/get-started/quickstart/github-glossary>

## I - Get started with GitHub (online)

### 1.1° Create an account on GitHub

<https://github.com/>

### 1.2° Get familiar with your space

In the top right of the web explorer,

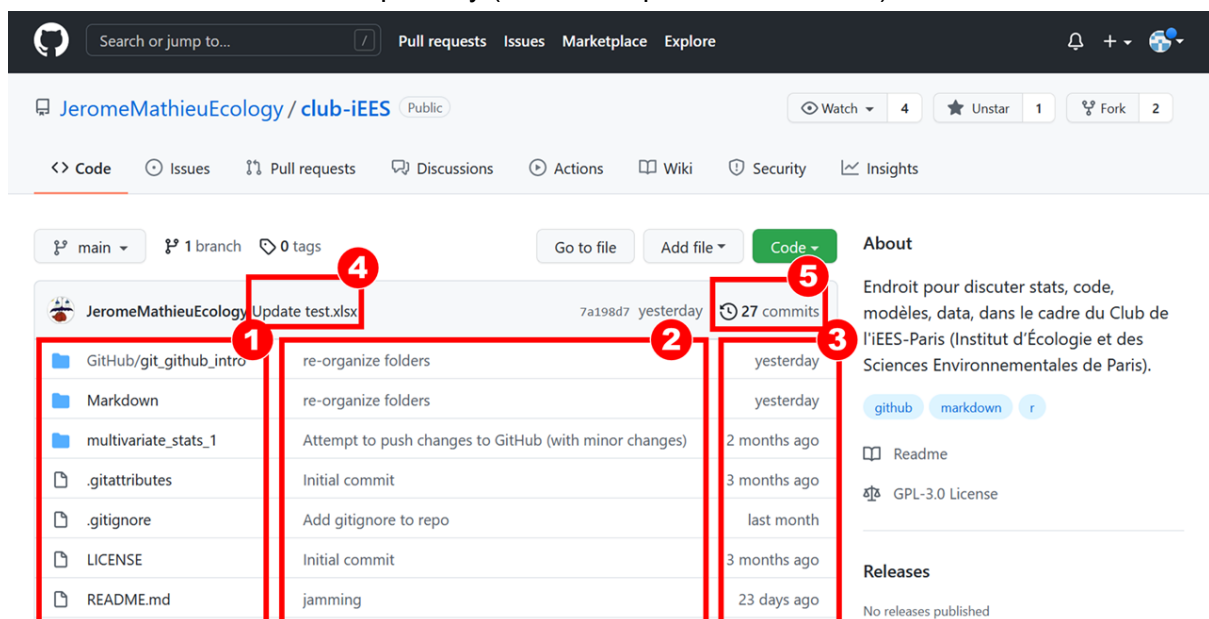
- go to “your profile” → this is your public profile
- go to “Repositories” → This is where all your repositories will be accessible

### 1.3° Find a public repository

- Click on the github logo in the top left of the screen
- Do a search in the search bar.
- Clic again on the GitHub logo and do a search for “iees”

### 1.4° Explore a GitHub repository

- Go to the club-iees repository (Hint : see question 3 to find it)



**Figure 1.** (1) List of ALL the files in the repository. (2) Summary description of the latest commit. (3) Date of the last commit. To access the whole file history, you need to click on the file name (in (1)). To

access only the changes done in one specific commit, click the text in (2). The latest commit will always be displayed above, in (4) to access the latest commit. Click on (5) to see the history of all previous commits.

### **Understand the README file**

- The text displayed at the bottom ("Club iEES, Prochaines séances thématiques [...]") and its displaying style are in the README.md file.

### **Open/download a file**

- Go to README.md and look inside the file (click on "raw")
- Go to [Markdown/fichiers\\_demo](#) , click on Rscript.R → scripts are automatically displayed. To save it, click on Raw then right click/save as

### **Understand the structure of the repository**

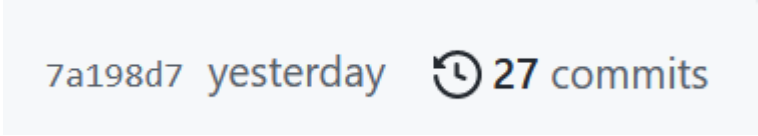
- Explore the folders and files of a repository
- go to [/Markdown](#) then to [/fichiers/demo](#) → which types of file can be stored in a repo?
- 

## **1.5 Explore the history of changes in a repo**

You can access to different aspects of the history of a repository :

### **See the history of commits**

In the area n°5 you see on the left the name of the last commit, the date, and the numbers of commits. To see all commits, click on "commits" → you see the history of commits



7a198d7 yesterday ⌚ 27 commits

### **See all files affected by the same commit**

- Access the files concerned by any commit

Go to history of commits (see above), then click on the name of a commit. Do it for "re\_organize folders" → you see all files that were changed during this particular commit.

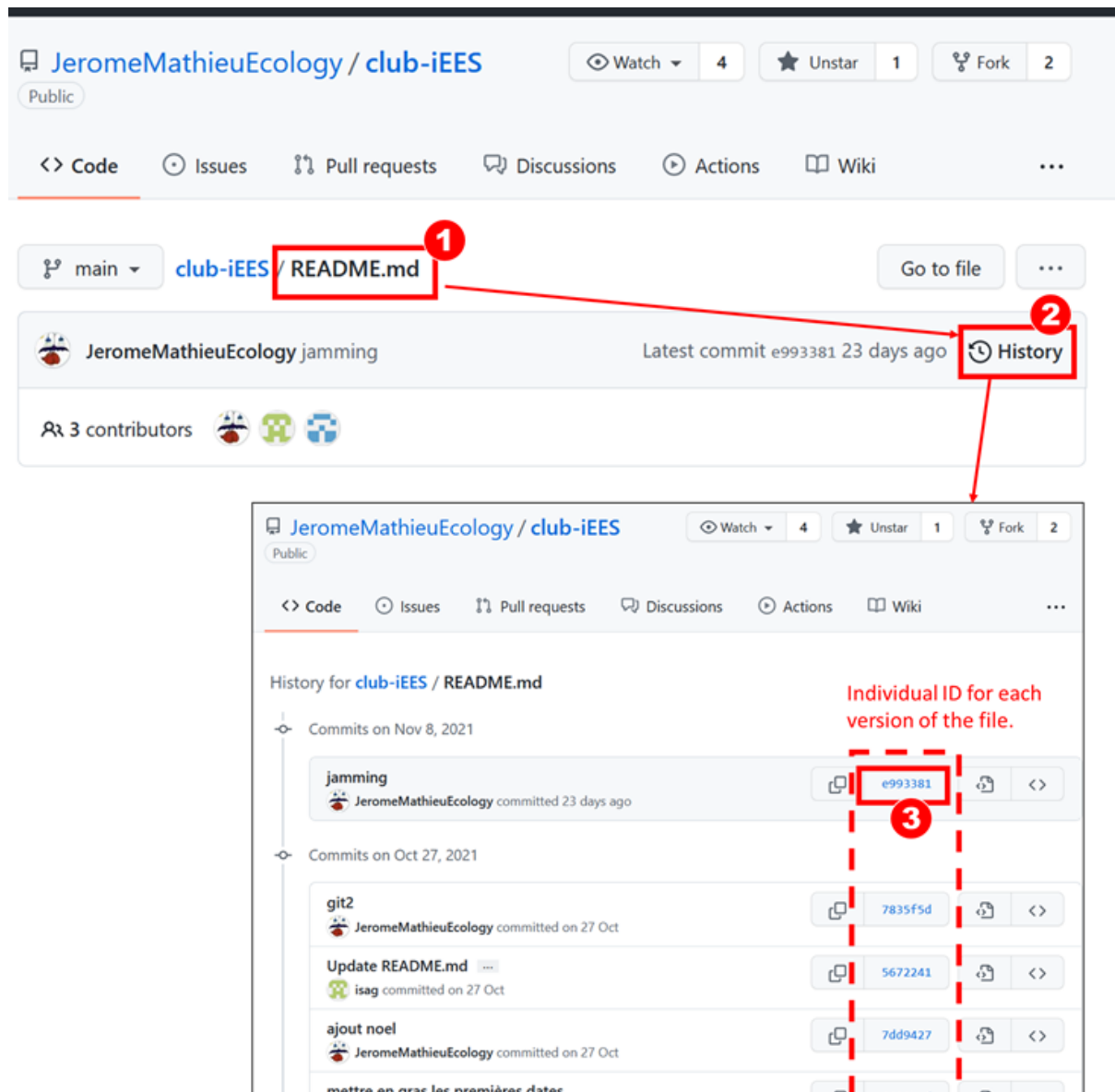
- Access the files concerned by the *last* relevant commit

- Go to the root of the club iEES repo, then in area n°2, click on the commit "re-organize folders" → You see all files

-




### **See the history of each file separately**

To do that, click on the name of a file in area n°1 of fig.1 then click on history (see fig. 2)




**Figure 2.** See one single file history. Click on a file name (see area (1) from Fig. 1), then History (2 in this figure). The window now shows all the different versions of your file. Click on (3) to see the diffs for a file (go to Figure 3).


### **See the different versions of a file**

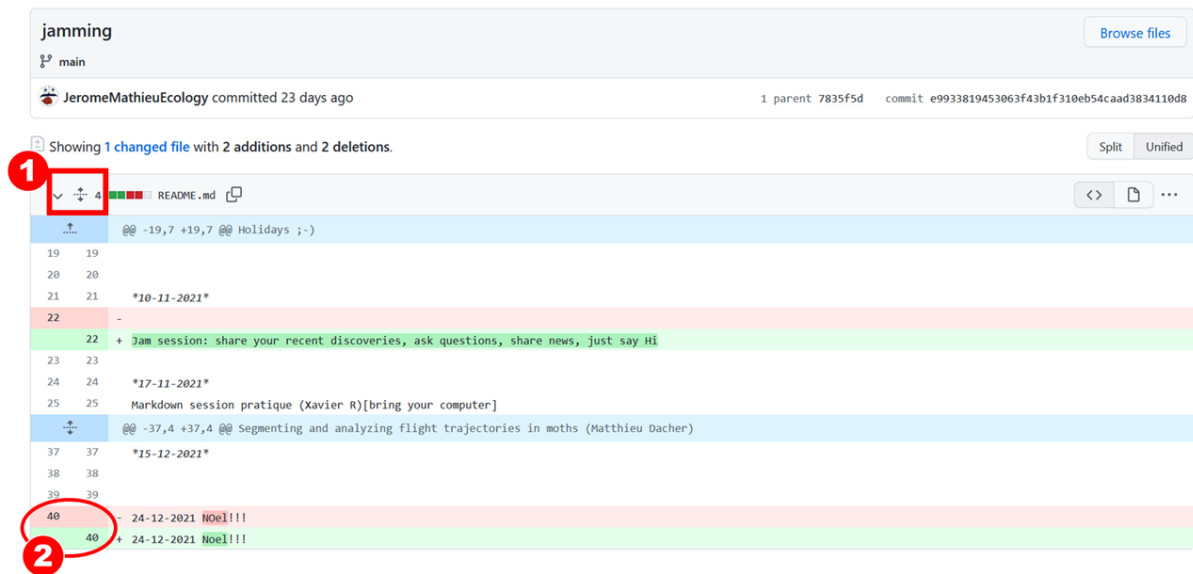
- Click on  to see the version of the file at a given point in history.
- Click on  to see the source of the file at this given point in history.
- You can do versioning with different file formats. For instance with excel. Demo :
- Go to the root of the repository of the club
- click on test.xlsx / history
- click on  of the version at the bottom (older version)
- click on “view raw” or “Download”

### See the different versions of a repository

- Click on , you will see the whole repository in its state at a given point of history. **Warning:** the interface will look the same (that can be confusing!), but you are indeed browsing the repository as it was at that point in history.

### See the detailed log of changes for a file (diff)

- Click on a version number (3 in Figure 2) () , and visualize the detailed log of changes (= diffs) (Figure 3).



**Figure 3.** Visualize the differences between old and new versions. By default, only the lines with changes are shown, but click (1) to see the whole file. (2) Example of differences: old line 40, in red, read “NOel”, while new line 40, in green, reads “Noel”.

## II - Work with your own repositories

### First, Install Git on your computer

Use Git with R studio : <https://r-pkgs.org/git.html#git-setup>

Use Git with GitHub Desktop : <https://desktop.github.com/>

(other resources are available, including GitHub for Atom <https://github.atom.io/>)

### Exercises

You can do them with R studio or GitHub Desktop. The explanations for each system are given below.

- 1° Make a new Git repository in your computer, with a csv file.
- 2° Publish your repository on Github.com and check online that it is updated
- 3° Make/update in the readme with a text containing two levels of titles (hint : use # and ##) and italic words (hint : use \*xxx\*). Make changes in the csv file.
- 4° Commit and push to GitHub.com
- 5° Check the update on GitHub.com (F5 to refresh the webpage)
- 6° Go to the settings of the repository and explore them.
- 7° Go to Options and scroll down to the danger zone → you can make your repository public or private and you can delete your repo.

### With R Studio

#### **2.1° Create your own repository**

<https://r-pkgs.org/git.html#git-init>

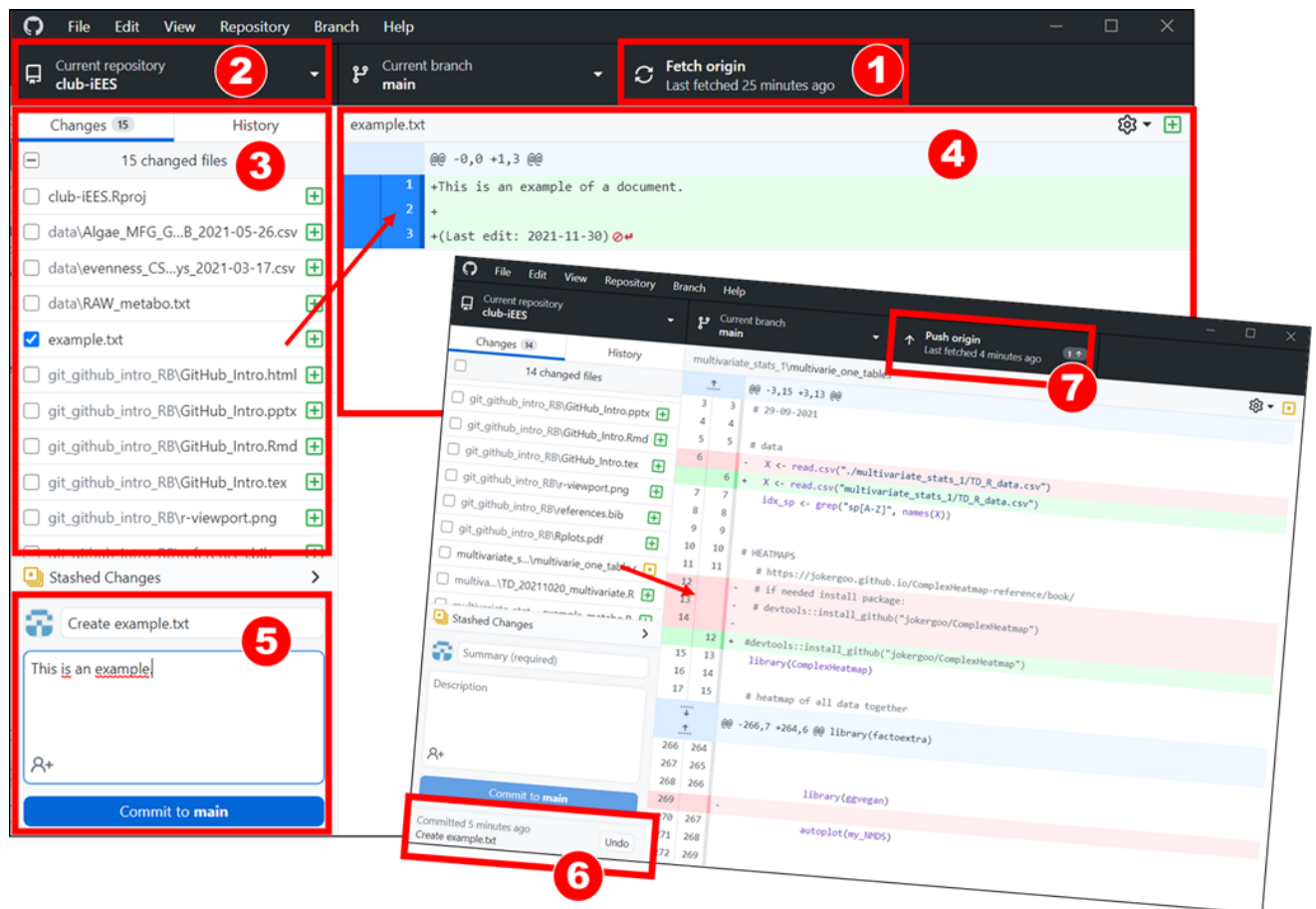
#### **2.2° Do a local modification in the repo: “commit”**

With R Studio : <https://r-pkgs.org/git.html#git-status>

#### **2.3° send your modification on GitHub: “push”**

With R Studio : <https://r-pkgs.org/git.html#github-init>

## With GitHub Desktop

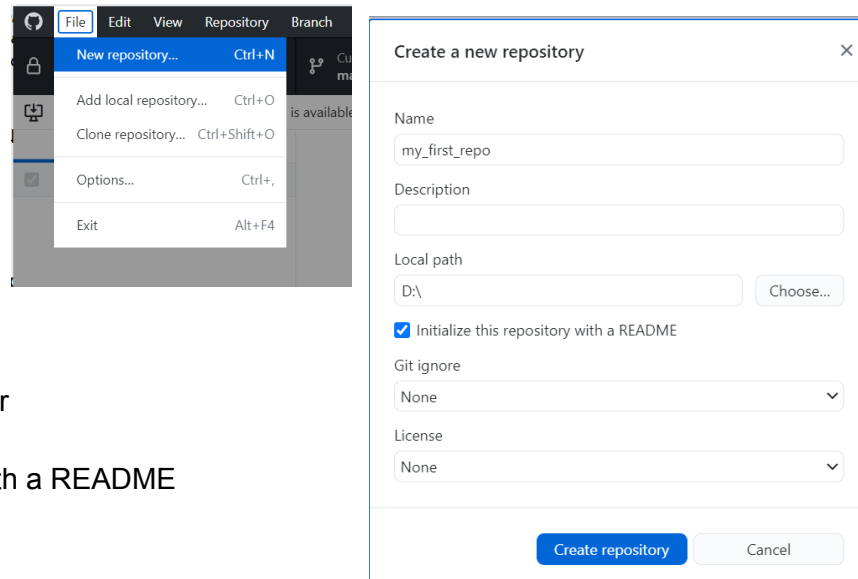


**Figure 4.** GitHub Desktop interface. (1) Before you start working, check you are up-to-date with the online repository by clicking “Fetch”. If there have been some changes, the icon will change to “Pull”. Click the same button again. (2) “Current repository”: drop-down menu with all the repositories you cloned. (3) List of files with differences between your local version and the online version. When you select a file (here, “example.txt”, it appears on the right side (4). The additions appear in green and the deletion appears in red. If you want to commit your changes, check the boxes near the file names in (3), then fill-in the commit field (5). You need at the very list a summary. Click “Commit to main” when you are ready. You can undo the commit while it hasn’t been pushed (6). Finally, once you are ready, click (7) to push your commit to the online repository.

## 2.1° Create your own repository

### Option 1 : Create a git folder in your computer and publish it to GitHub

1° File/New repository



2° Dialog box

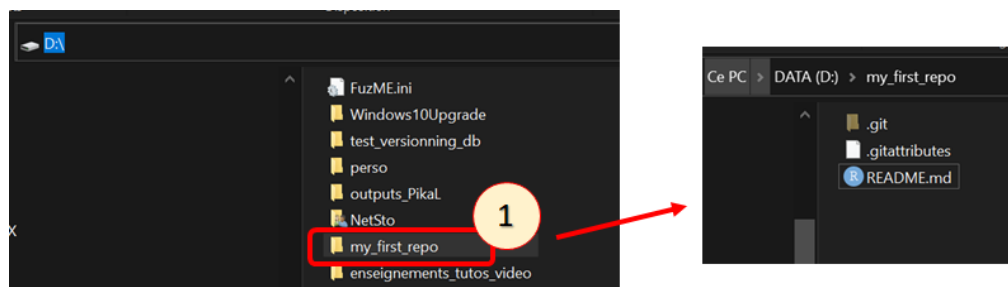
Name = name of the folder

Path = root of the folder

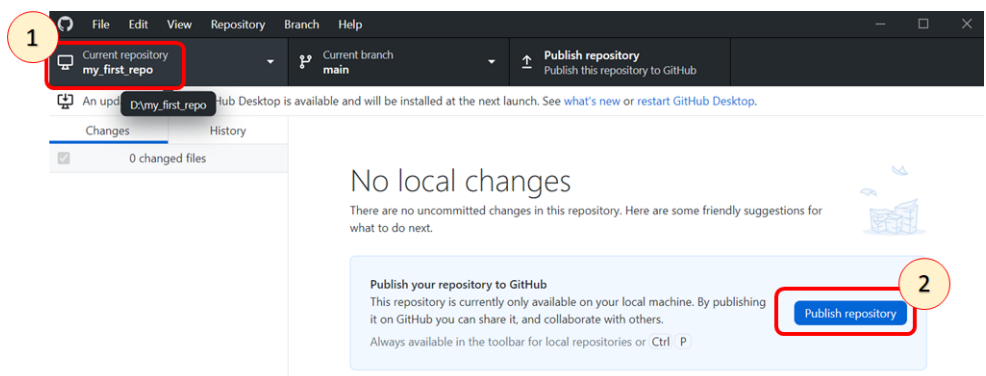
Initialize this repository with a README  
creates a readme

3° You should have a new repo (1)

In your PC :



In GitHub Desktop



4) To Publish your  
repo online,

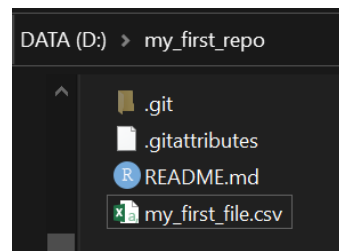
click on Publish repository (2) → You're done !

### Option 2 : create on Github.com then add it to your computer

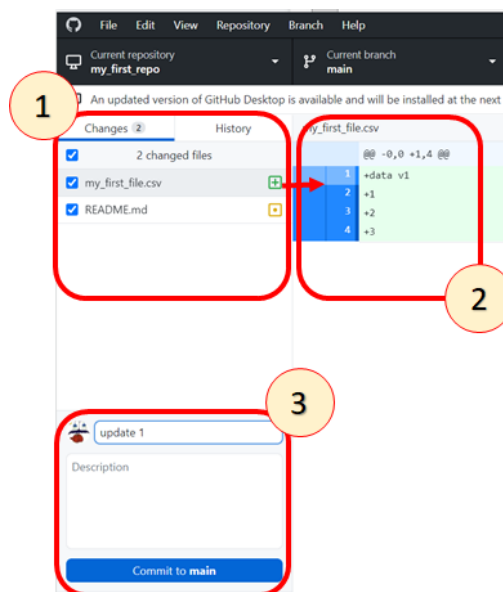
This is not recommended and will not be presented !!

## 2.2° Do a local modification in the repository: “commit”

Let's say we modify the readme file (with a text editor) and add a csv file “my\_first\_file.csv”



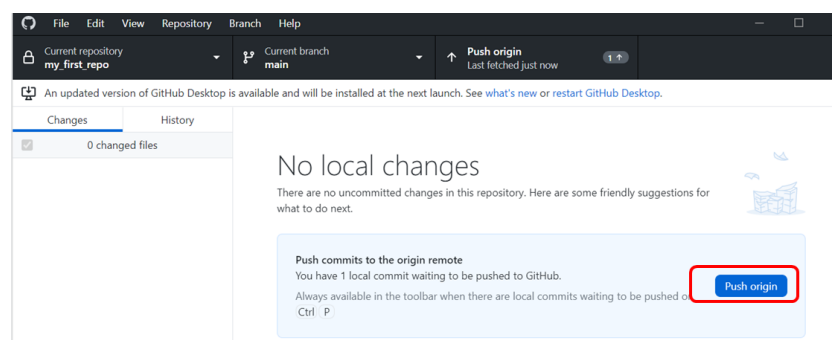
GitHub Desktop should automatically detect the files that has been changed (1)  
If you click on a file, the changes (the “diffs”) will be displayed on side of it (2)



In order to save these modifications into the versioning system in your computer (nothing happens online so far), you need to “commit” this new version to the Git System (3)  
You need to write a title of the commit and click “**Commit to main**” (3)

## 2.3° send your modifications on GitHub.com : “push”

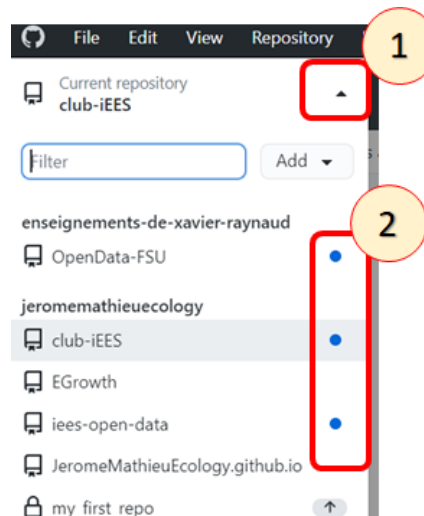
So far your modifications happened only in your computer.  
To send the modifications online to GitHub.com, click “**Push Origin**”  
This should send the modifs to the online folder





## 2.4° Check which folders needs to be pushed

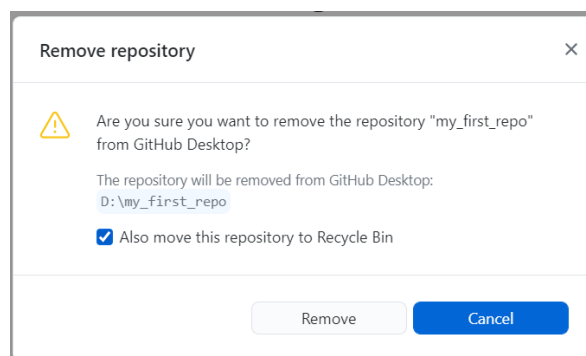
In GitHub Desktop, click on the triangle on the side of the current repository (1)  
All folders with changes that need to be pushed will be marked by a circle (2).



## 2.5 Remove / delete a folder

Go to Repository/remove

You can just remove the folder from the git system (the folder stays in your pc), or you can completely delete the folder by checking delete the folder

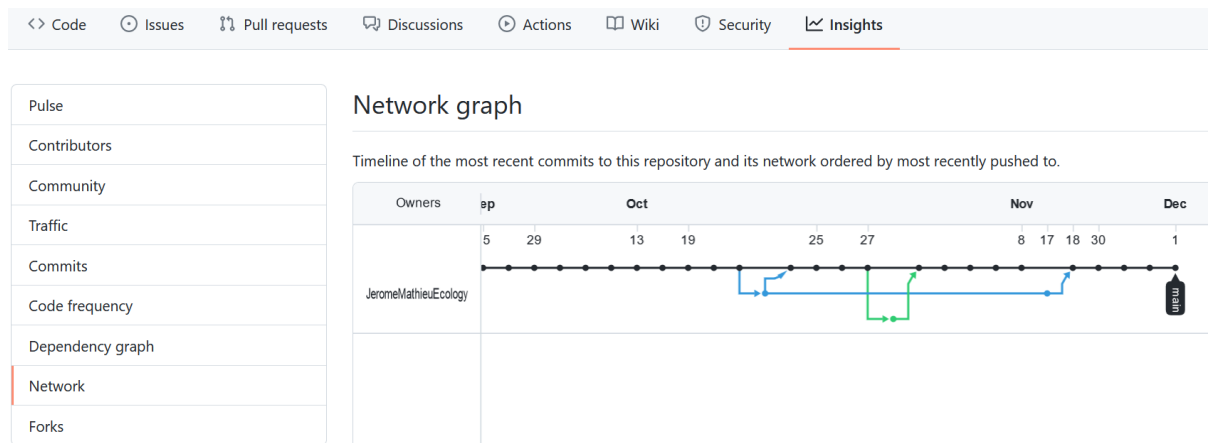


## III - Work with others

### 3.1° Visualize multiple branches - what your project may look like with multiple collaborators

From the home page of a project, click on Insight>Network and see the different branches for the same project.

Example: <https://github.com/JeromeMathieuEcology/club-iEES/network>



*Figure 5. Network graph for the club-iEES, as of 01/12/2021.*

### 3.2° Issue vs pull request

### 3.3° Make a pull to a public repository

With R Studio :

With GitHub Desktop

### 3.4° Contribute to repository as a contributor

- 
- Send to Jerome (or another contributor??) your GitHub ID → He will include you in the list of contributors to the Club iEES
- clone the club iEES to your computer
- make a change to a file/ add a file
- commit the change (if you are 2+ people in the room, choose only one person to do that!)
- push the change [before being able to push the change, you need to “fetch” and “pull” any modifications from the online repository!]
- update the web page (F5) → you should see your changes!