Git versioning:
Keep track of your changes

Physalia course 2023

Instructor: Jacques Serizay

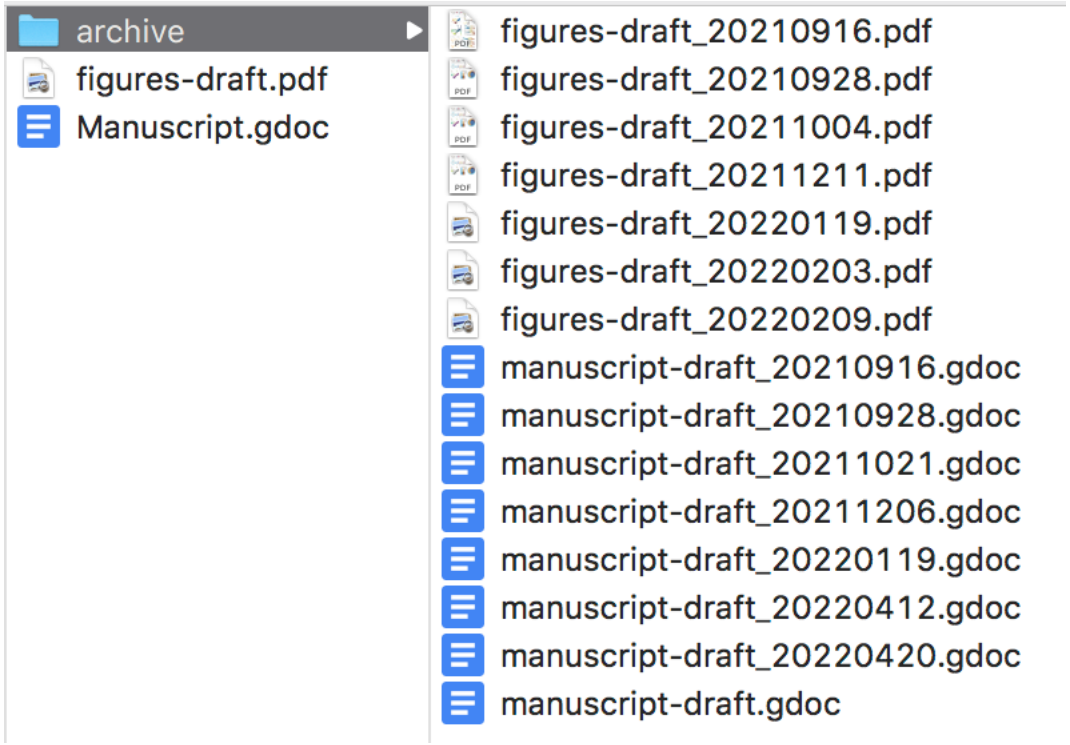
Keeping track of changes with git

https://happygitwithr.com/

- Git is a **version control system**.
- Git manages the evolution of a set of files – called a **repository** – in a sane, highly structured way.
- Its original purpose was to help groups of developers work collaboratively on big software projects.
- Essentially, this is "Track Changes" features from Microsoft Word on steroids.

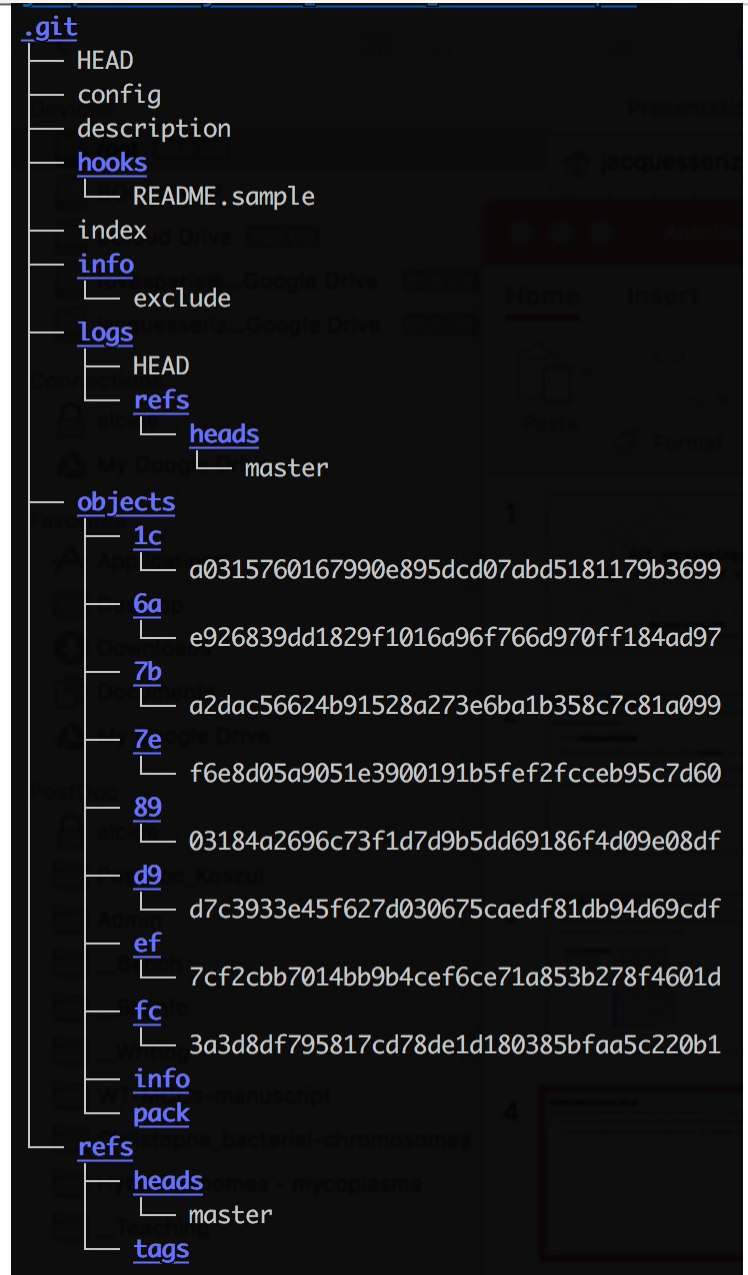
Keeping track of changes with git

- Ever time-stamped a file? Git does the same, for one or several files, and allows you to revert back to previous files or compare changes (for text files).



Keeping track of changes with git

- Your working directory only contains the 2 files (pdf and gdoc) you are working on, and changes of these files are stored in the `.git` subfolder, in a compressed way.



Keeping track of changes with git

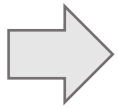
Git's main downside is… git is not user-friendly!

- It's painful to visualize changes, to browse history, to compare files, …
- No native graphical interface

Git vs github

- Git is a **software**, locally installed on one's computer

```
$> git --help
```

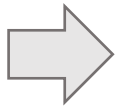


"Track Changes" features

Git vs github

- Git is a **software**, locally installed on one's computer

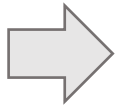
```
$> git
```



"Track Changes" features

- Github is an Internet hosting service, storing your local `git` repository (files AND changes) on-line ("remotely"):

```
https://github.com/
```



Dropbox features

Git vs github

- You are working on your manuscript draft in Word: you may rewrite sentences, resolve previous comments, reply to others still open, … As long as you are working on your own, this works.
- Now you want to share it with others → you decide to put the document online as a Google Doc, allow access to others, and enable synchronization with your local Google Drive.
- Your colleagues will eventually rewrite/resolve/reply together with you (sometimes with conflicting changes!), and everyone will see who is doing what, etc.

The Google Doc is your "remote repository" and your local Google Drive folder is your "local repository"

Typically, both are continuously synced, whereas Git and Github are not sync-ed "live", you do it manually and whenever you want.

Git vs github

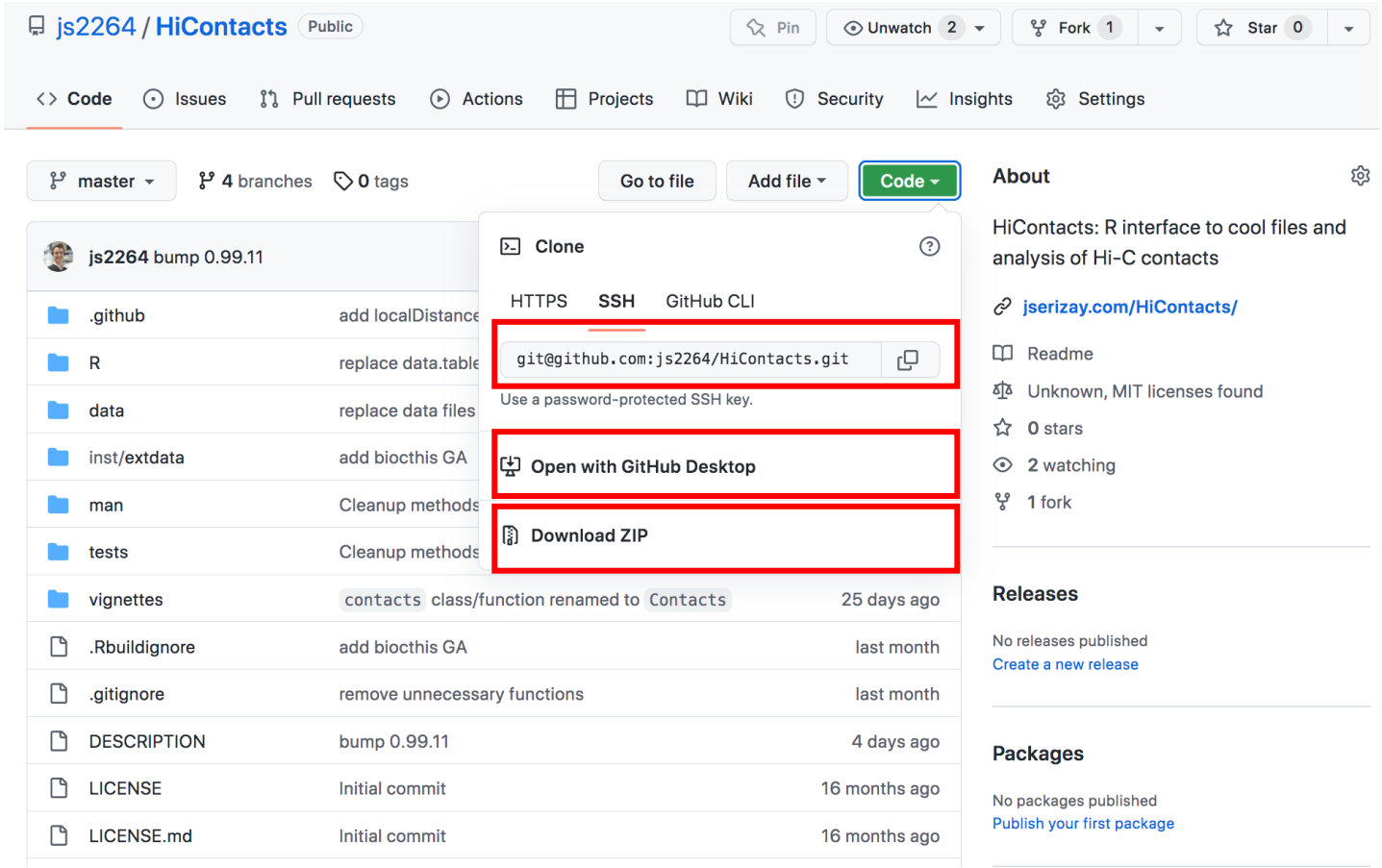
Github, gitlab, bitbucket: all different internet hosting services doing the same thing:

1. Storing your local git repositories online, to make available to others.
2. Allow specific online-based features: issues and pull requests
3. Easy “time-travel” (i.e. browsing files back in time) with permanent links
4. Search function!

Git vs github

Github, gitlab, bitbucket: all different internet hosting services doing the same thing:

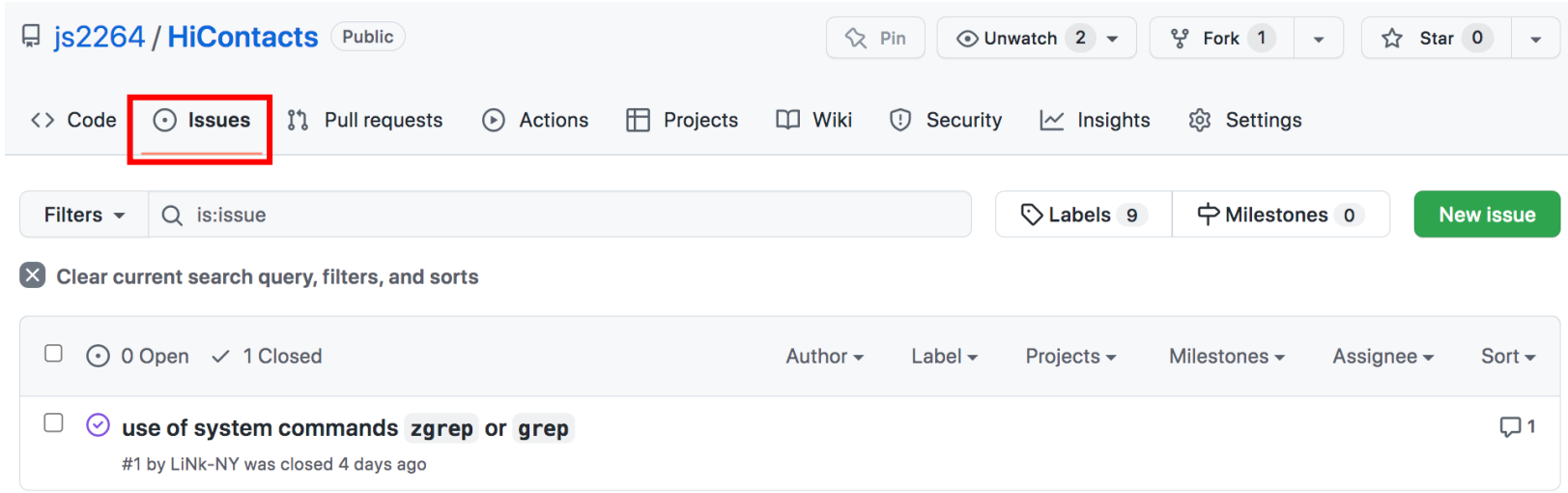
1. Storing your local git repositories online, to make available to others.



Git vs github

Github, gitlab, bitbucket: all different internet hosting services doing the same thing:

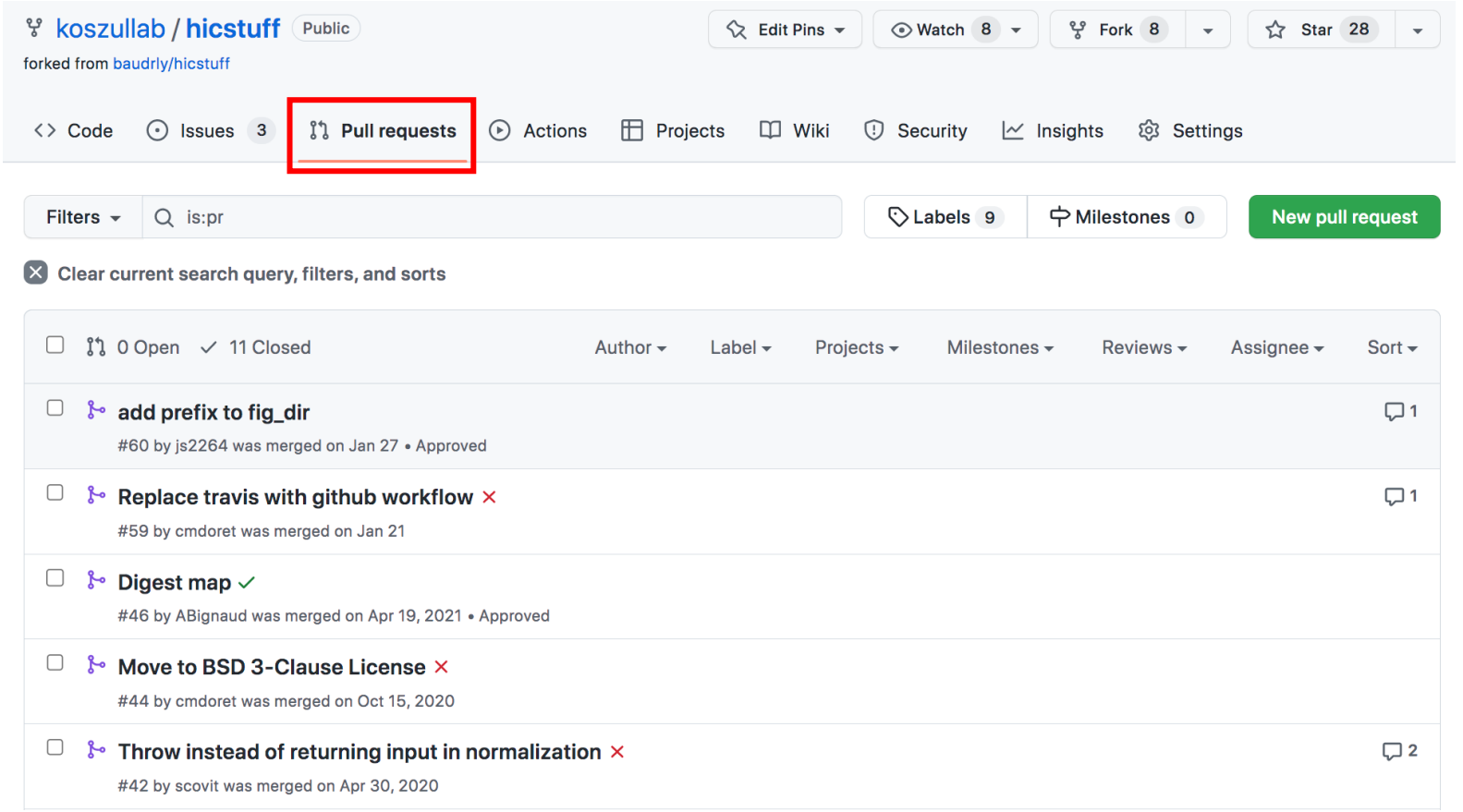
2. Allow specific online-based features: issues and pull requests



Git vs github

Github, gitlab, bitbucket: all different internet hosting services doing the same thing:

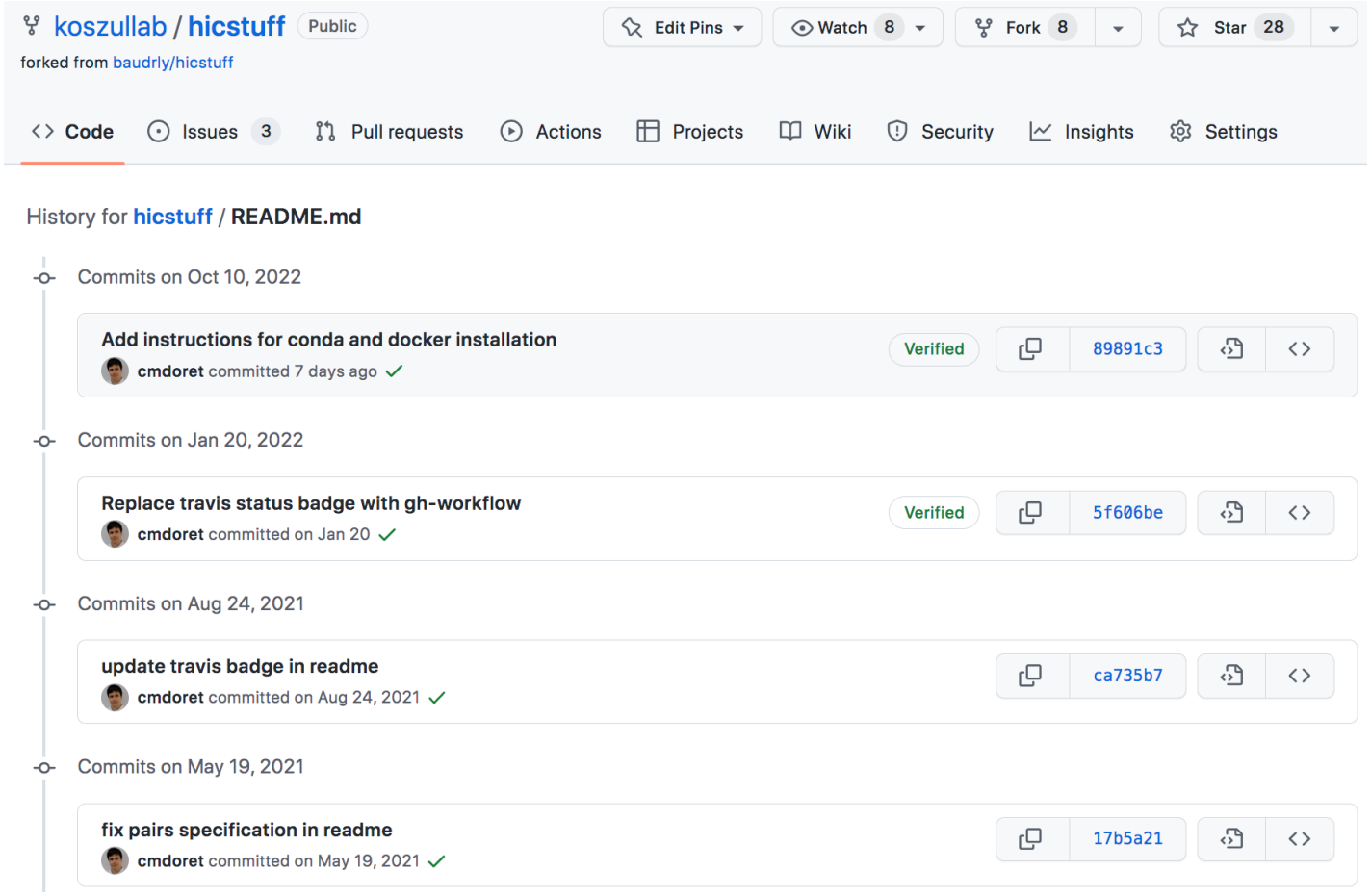
2. Allow specific online-based features: issues and pull requests



Git vs github

Github, gitlab, bitbucket: all different internet hosting services doing the same thing:

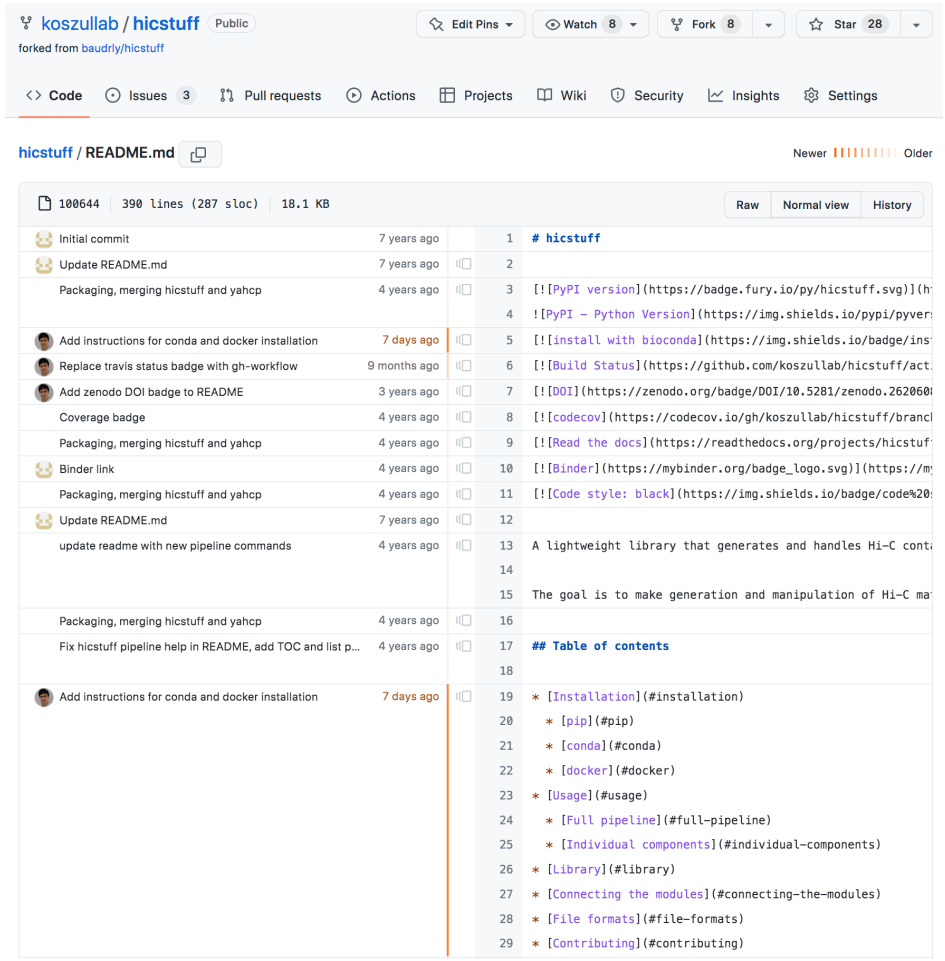
3. Easy "time-travel" (i.e. browsing files back in time) with permanent links



Git vs github

Github, gitlab, bitbucket: all different internet hosting services doing the same thing:

3. Easy “time-travel” (i.e. browsing files back in time) with permanent links



Github and organizations

- A public repository is readable by the world. The owner can grant higher levels of permission to others, such as the ability to push commits.
- A private repository is invisible to the world. The owner can grant read, write (push), or admin access to others.
- There is also a notion of an organization, which can be useful for managing repository permissions for entire teams of people.

Github and organizations

- A public repository is readable by the world. The owner can grant higher levels of permission to others, such as the ability to push commits.
- A private repository is invisible to the world. The owner can grant read, write (push), or admin access to others.
- There is also a notion of an organization, which can be useful for managing repository permissions for entire teams of people.

→ Git has been re-purposed by the data science community.

- We now also use it, IN COLLABORATION WITH A CLOUD-BASED GIT HOST, to manage the diverse collection of files that make up typical data analytical projects, which often consist of data, figures, reports, and, yes, source code.

Github and organizations

R infrastructure

Overview | Repositories 136 | Projects | Packages | People 15

Pinned

devtools Public
Tools to make an R developer's life easier
R ☆ 2.2k ⑂ 747

roxygen2 Public
Generate R package documentation from inline R comments
R ☆ 521 ⑂ 208

testthat Public
An R 📦 to make testing 😀
R ☆ 810 ⑂ 299

pkgdown Public
Generate static html documentation for an R package
R ☆ 636 ⑂ 307

usethis Public
Set up commonly used 📦 components
R ☆ 718 ⑂ 233

httr Public
httr: a friendly http package for R
R ☆ 959 ⑂ 2k

tidyverse

The tidyverse is a collection of R packages that share common principles and are designed to work together seamlessly

361 followers | http://tidyverse.org | Verified

Overview | Repositories 37 | Projects | Packages | People 27

Pinned

dplyr Public
dplyr: A grammar of data manipulation
R ☆ 4.2k ⑂ 1.5k

ggplot2 Public
An implementation of the Grammar of Graphics in R
R ☆ 5.6k ⑂ 1.9k

readr Public
Read flat files (csv, tsv, fwf) into R
R ☆ 924 ⑂ 278

purrr Public
A functional programming toolkit for R
R ☆ 1.1k ⑂ 229

tidyr Public
Tidy Messy Data
R ☆ 1.2k ⑂ 393

rvest Public
Simple web scraping for R
R ☆ 1.4k ⑂ 332

Bioconductor

Software for the analysis and comprehension of high-throughput genomic data

49 followers | https://bioconductor.org

Overview | Repositories 260 | Projects 2 | Packages | People 2

Pinned

Contributions Public
Contribute Packages to Bioconductor
☆ 115 ⑂ 32

BiocStickers Public
Stickers for some Bioconductor packages - feel free to contribute and/or modify.
R ☆ 110 ⑂ 77

bioconductor_docker Public
Docker Containers for Bioconductor - NEW!
Shell ☆ 50 ⑂ 18

bioconductor.org Public
Source code for the Bioconductor website
HTML ☆ 16 ⑂ 33

pkgrevdocs Public
Training Material for Community Reviewers
TeX ☆ 6 ⑂ 11

BBS Public
The Bioconductor Build System
Python ☆ 7 ⑂ 6

rOpenSci

190 followers | Berkeley, CA | https://ropensci.org/ | @rOpenSci | info@ropensci.org | Verified

Overview | Repositories 328 | Projects 1 | Packages | People 79

README.md

👋 Welcome to the rOpenSci Project on GitHub. Our repositories are currently distributed across a few different organizations.

You can explore our resources page to find what you're after, it includes a list of all our GitHub organizations.

Also refer to our code of conduct.

Pinned

software-review Public
rOpenSci Software Peer Review.
R ☆ 249 ⑂ 103

software-review-meta Public
For organizing projects related to rOpenSci Software Peer Review
☆ 11 ⑂ 3

dev_guide Public
rOpenSci Packages: Development, Maintenance, and Peer Review
R ☆ 133 ⑂ 49

rtweet Public
R client for interacting with Twitter's [stream and REST] APIs
R ☆ 751 ⑂ 202

How to set things up

1. Register for a GitHub account

2. MAKE SURE YOU ARE RUNNING R/RStudio/BiocManager WITH THE APPROPRIATE VERSION!

3. Install Git: https://git-scm.com/downloads, or:

    - Linux: `sudo apt-get install git`
    - Mac: `xcode-select –install` (needs Xcode)
    - Windows: `choco install git.install` (needs Chocolatey)

How to set things up

1. Register for a GitHub account

2. MAKE SURE YOU ARE RUNNING R/RStudio/BiocManager WITH THE APPROPRIATE VERSION!

3. Install Git: https://git-scm.com/downloads, or:

    - Linux: `sudo apt-get install git`
    - Mac: `xcode-select –install` (needs Xcode)
    - Windows: `choco install git.install` (needs Chocolatey)

4. Configure **git** from R (easy)

```
usethis::git_sitrep()
gert::git_config_global_set("user.name", "Jacques")
gert ::git_config_global_set("user.email", "jacquesserizay@gmail.com")
usethis ::git_vaccinate()
usethis ::use_git()
usethis ::use_git_ignore("*.Rproj")

# Don't forget to make an initial commit!
```

How to set things up

5. Configure **github** (harder):

   - Git can communicate with a remote server using one of two protocols, HTTPS or SSH, and the different protocols use different credentials.

   - We **must** use **SSH**, as this means of communications is required later on, for Bioconductor submission.

   - For that, we need to generate a pair of private/public SSH keys, connect to GitHub from R and push the public key to GitHub.

How to set things up

5. Configure **github** (harder):

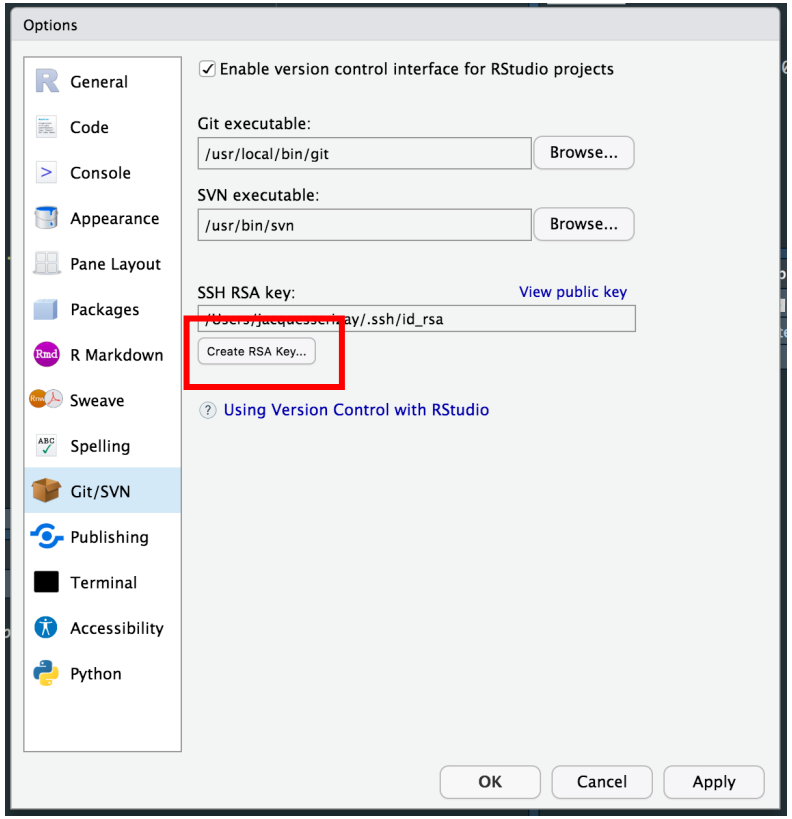
- Git can communicate with a remote server using one of two protocols, HTTPS or SSH, and the different protocols use different credentials.
- We **must** use **SSH**, as this means of communications is required later on, for Bioconductor submission.
- For that, we need to generate a pair of private/public SSH keys, connect to GitHub from R and push the public key to GitHub.

```
gh::gh_whoami()
usethis::gh_token_help()

#Create a GH PAT
usethis ::create_github_token()
gitcreds ::gitcreds_set()
#Paste your GitHub token, not your password!

#Add an SSH key to GitHub
usethis ::use_github()
credentials::ssh_setup_github()
```

How to set things up

5. Configure **github** (harder):

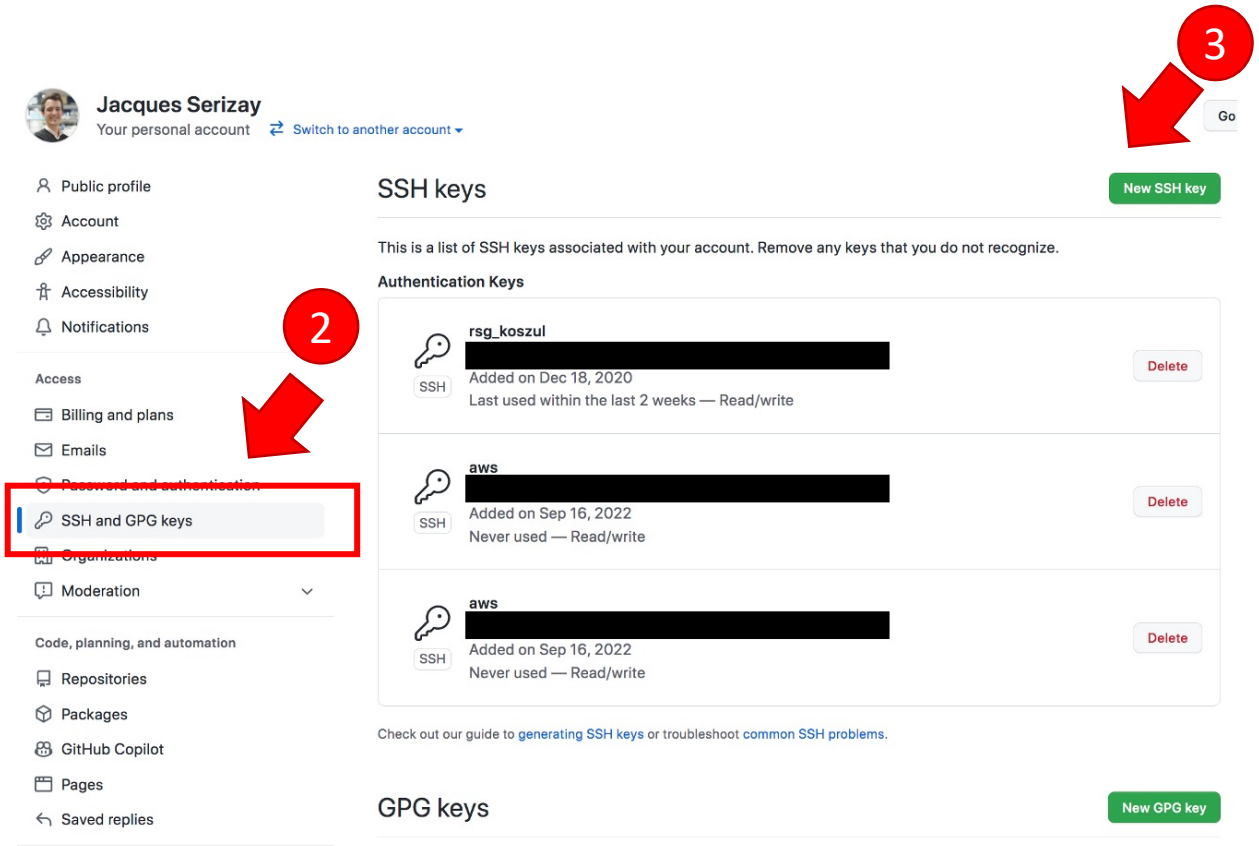
   - Fallback approach: create a key in RStudio

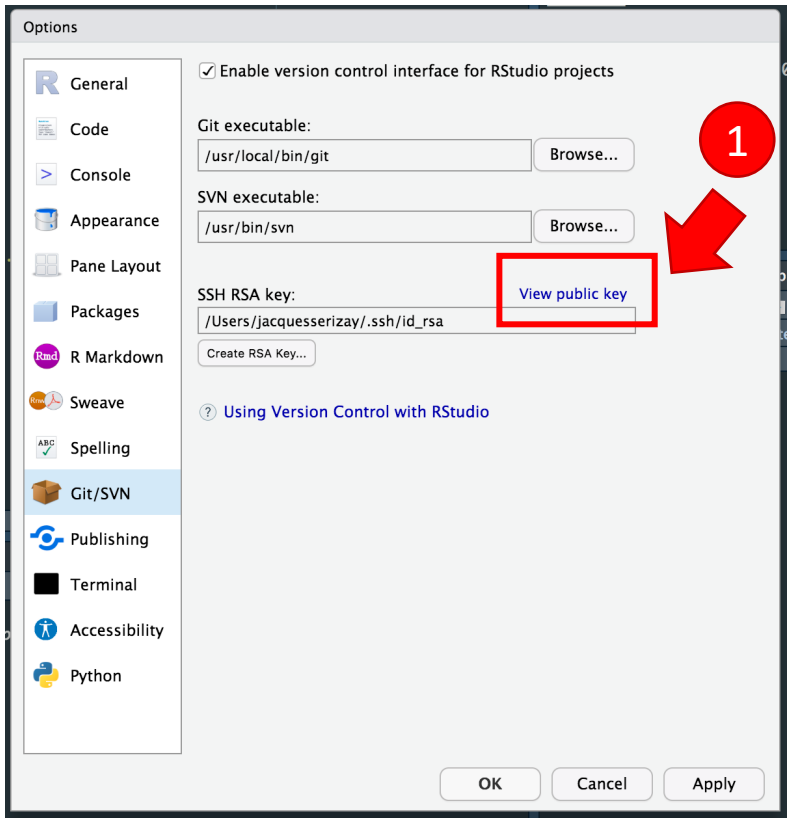


How to set things up

5. Configure **github** (harder):

   - Then, we need to upload the public key to GitHub

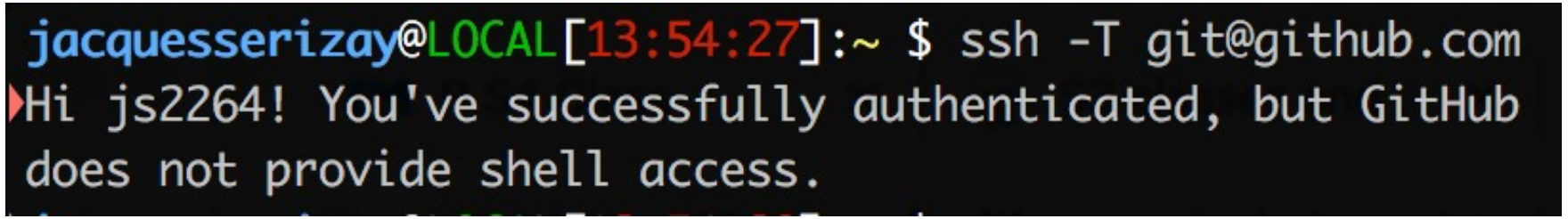




How to set things up

5. Configure **github** (harder):

   - Test it with `ssh -T git@github.com`

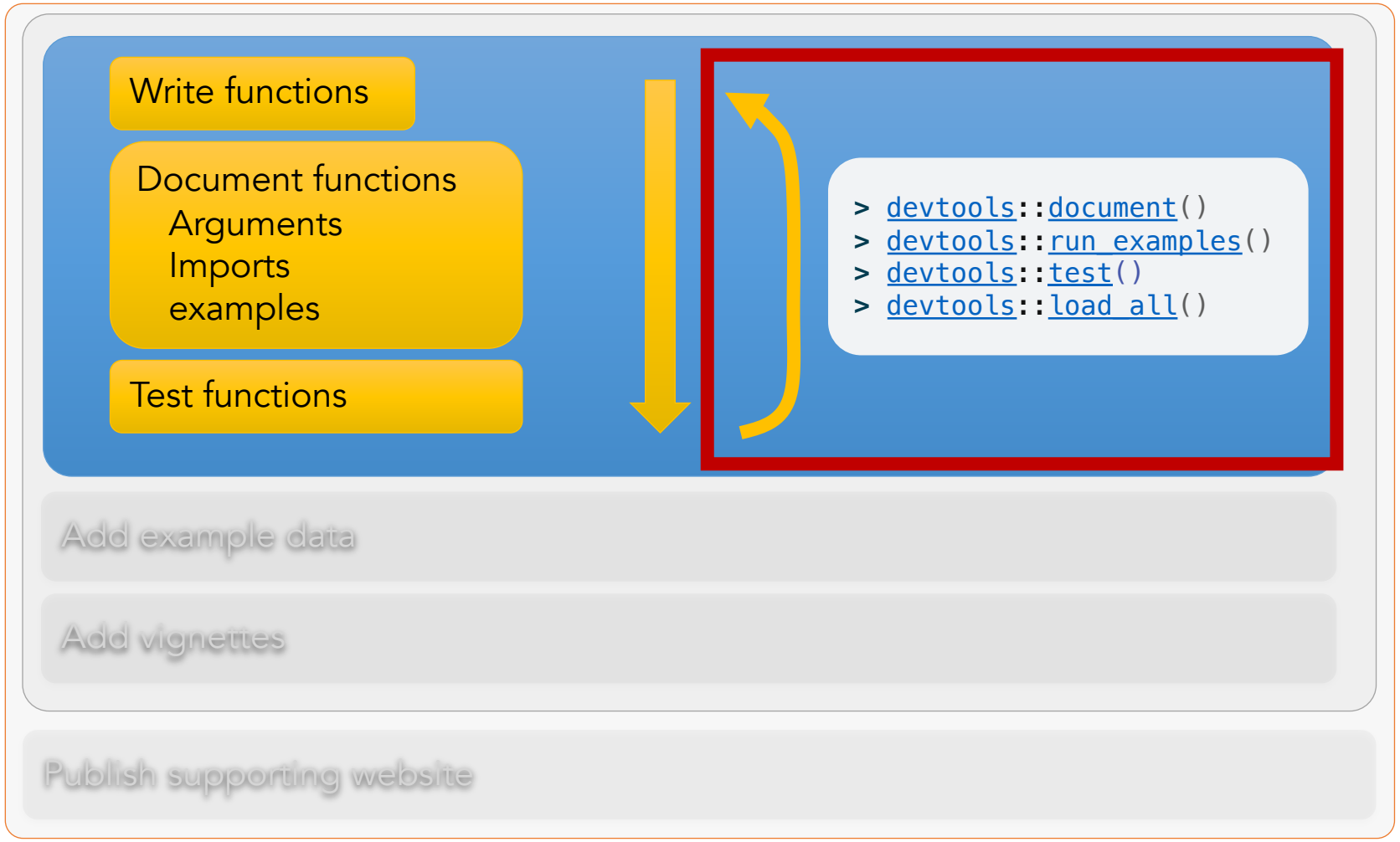


How to set things up

If you are still struggling…

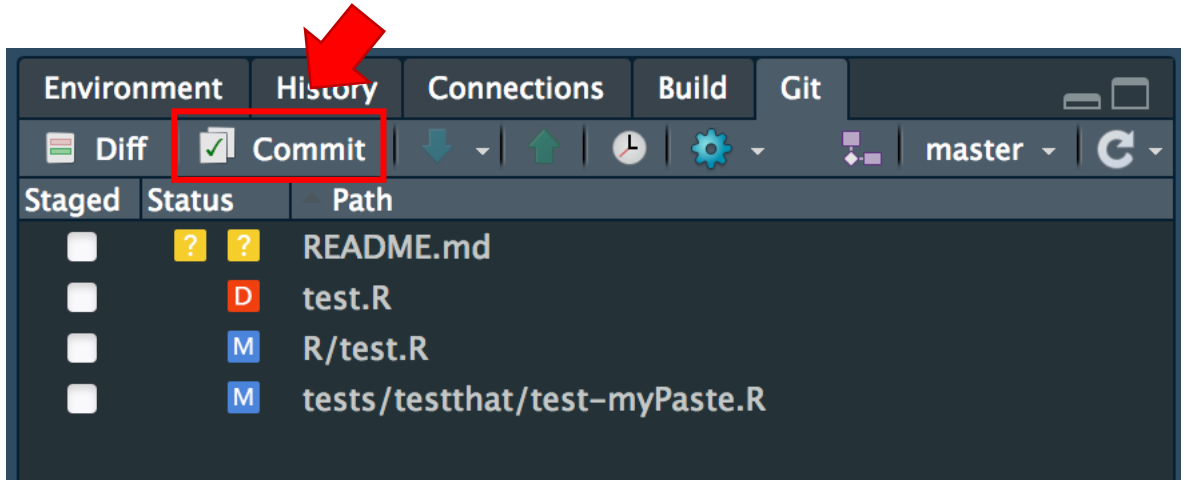
https://happygitwithr.com/usage-intro.html#usage-intro

Lather-rinse-repeat

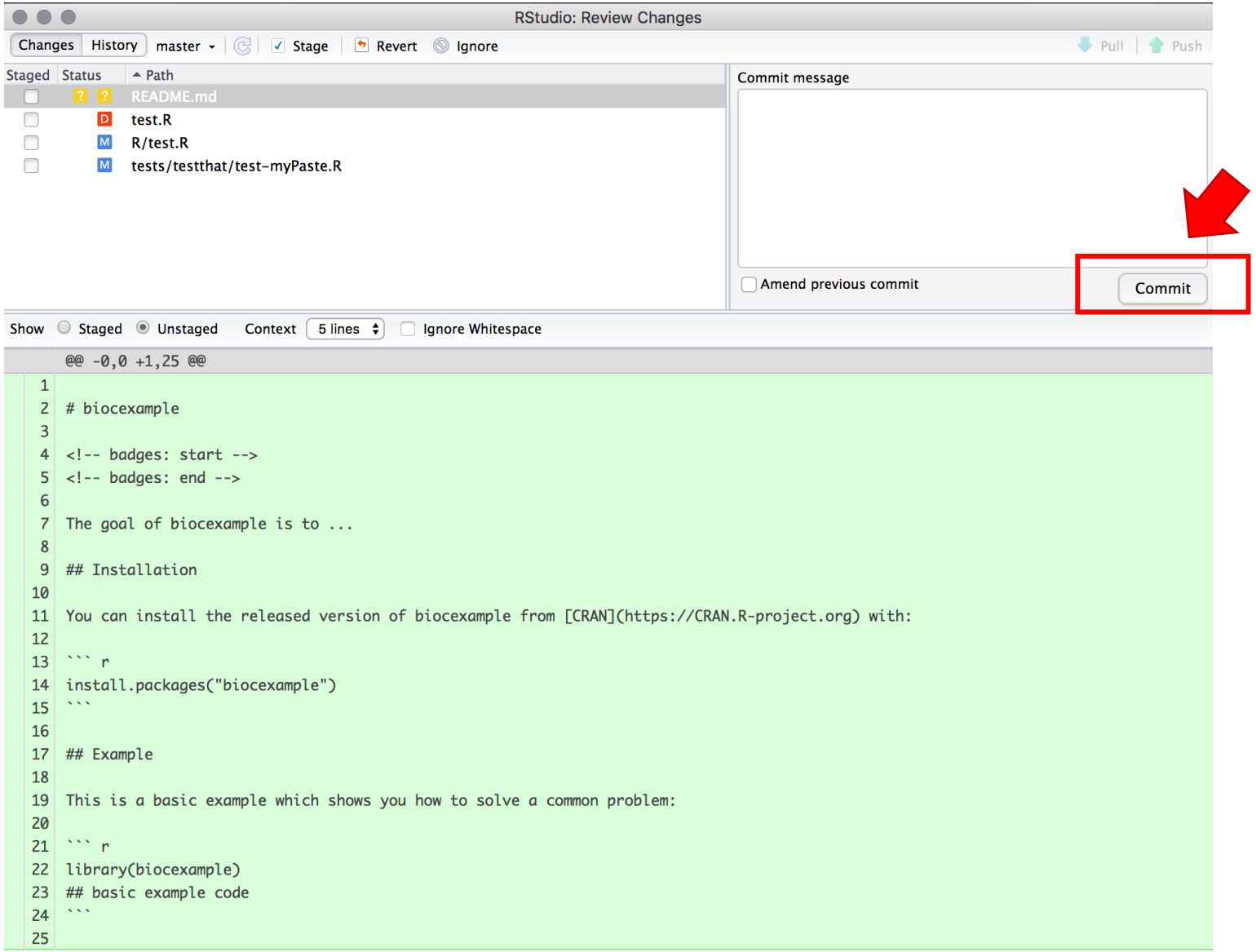



```
Git commit
Github push
```

Lather-rinse-repeat



Lather-rinse-repeat



Lather-rinse-repeat