
Git + R = collaborative coding

Ainhoa Leger
Gonzalo Garcia

Introduction



- **What is GitHub?** It is an online place to
 - ❖ store your code
 - ❖ track the changes that we have made on the code
 - ❖ avoid multiple copies or versions
 - ❖ roll back my code to a previous version
 - ❖ collaborate effectively with others
- **Workshop aim:** Learn how to version control an R project using GitHub within Rstudio



Roadmap

- Part 1: Ainhoa
 - ❖ **Git + R for individual use**
- Part 2: Gonzalo
 - ❖ **Git + R for collaborative use**
 - ❖ **Sneak into more advanced Git + R features**
- Both parts :Guidelines (15 min) + Hands on! (15 min)

PART 1

(individual use)

Topics

Individual use

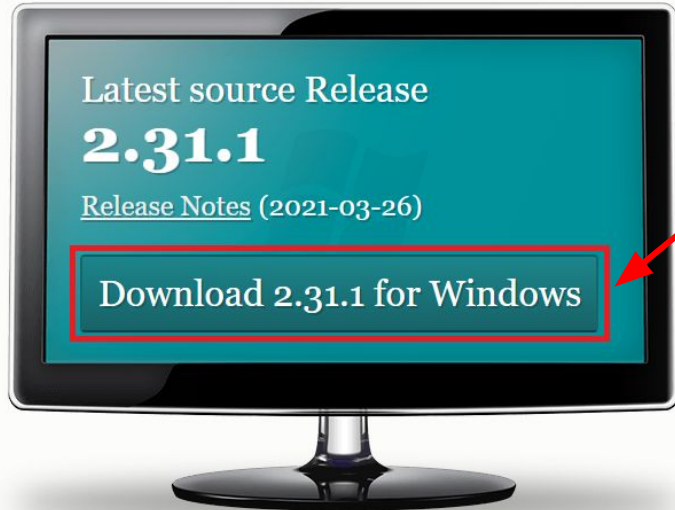
- 1. Install Git + Setup a GitHub account**
2. Create a GitHub repository
3. Connect a GitHub repository to an RStudio project
4. Version control the files
 - a. Commit action
 - b. Push action

Collaborative use

5. Fork GitHub repositories
6. Create Pull Request
7. Accept/Merge changes
8. Refresh forked repo

1.1 Install Git

- In order to use the Git version control system we first need to install it
- Git is the machinery which works behind Github and Rstudio

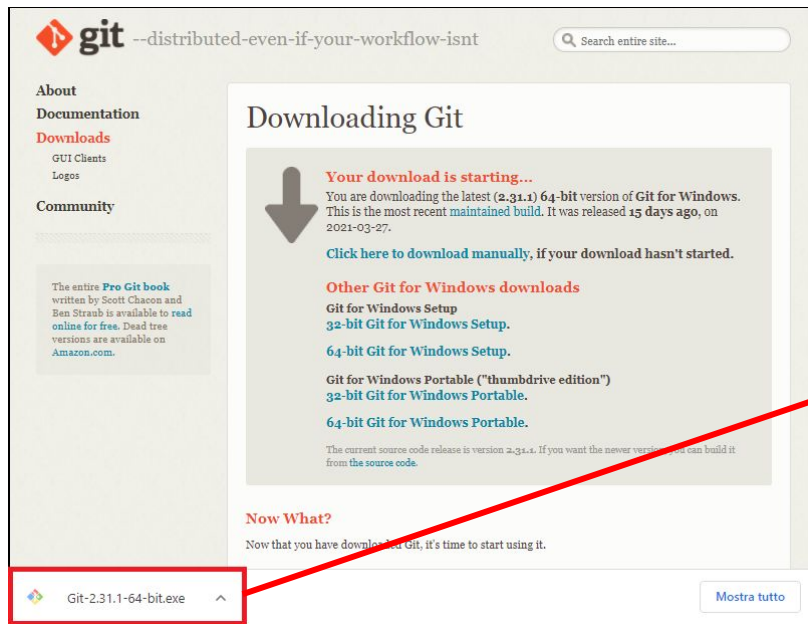


Steps to install Git on your machine

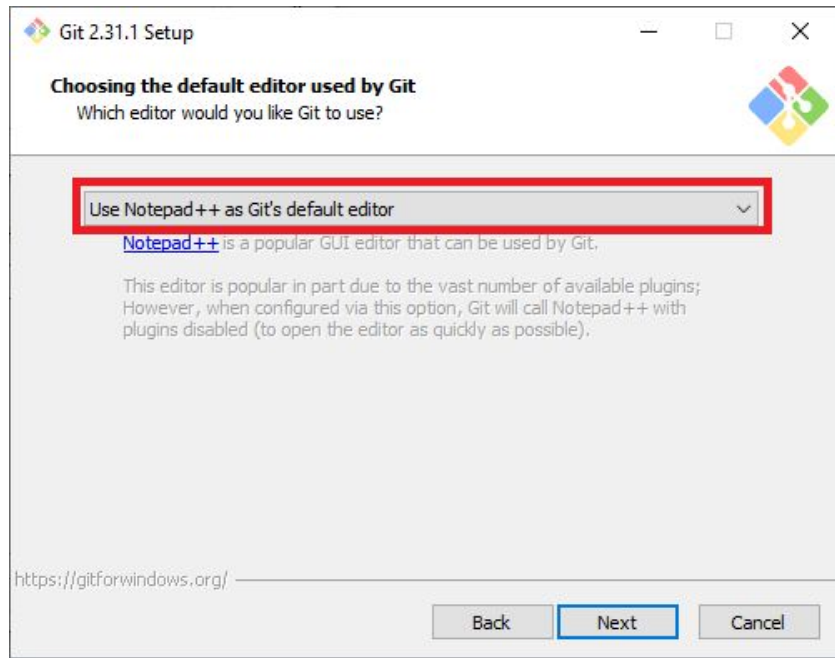
- Go to <https://git-scm.com/downloads>
- Download executable file (.exe)

1.1 Install Git

→ Execute the file and follow the instructions (by clicking next) to complete the installation



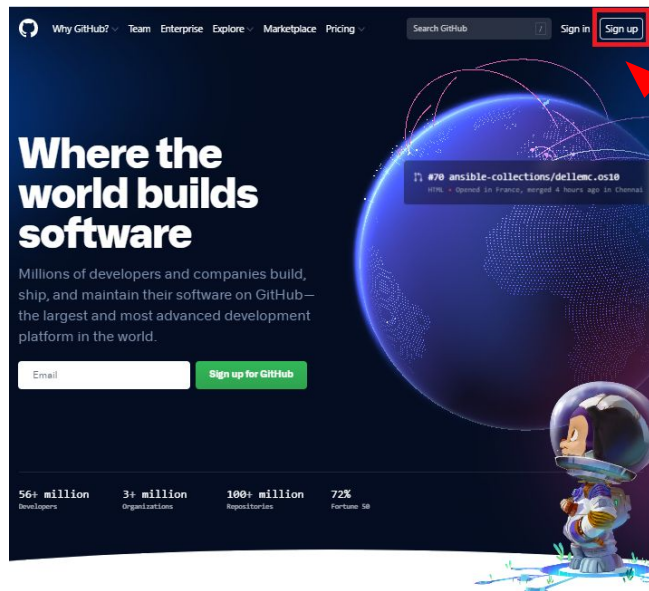
1.1 Install Git



Advice: There are several editors to choose from. Notepad++ is a simple editor that can be useful in general to have installed on your machine. However please note that when coding in R we use RStudio to edit our scripts.

1.2 Setup a GitHub account

- We need to create the online platform where you have your repositories, version control your projects and collaborate with others



Steps to setup a Github account

- Go to <https://github.com/>
- Just follow the Sign up process (top right of the page)

1.2 Setup a GitHub account

Create your account

Username *

Email address *

Password *

Make sure it's at least 15 characters OR at least 8 characters including a number and a lowercase letter.
[Learn more.](#)

Email preferences

☐ Send me occasional product updates, announcements, and offers.

Verify your account

Risolvere questo puzzle, così sappiamo
che è una persona reale

Verifica

Create account

By creating an account, you agree to the [Terms of Service](#). For more information about GitHub's
privacy practices, see the [GitHub Privacy Statement](#). We'll occasionally send you account-related
emails.

→ Type a username, email address and password

It is a good idea to use a username that can easily be associated with you.

For example: first name + last name

Tim Riffe => **timriffe**

Diego Alburez-Gutierrez => **alburezg**

→ Do the verification process



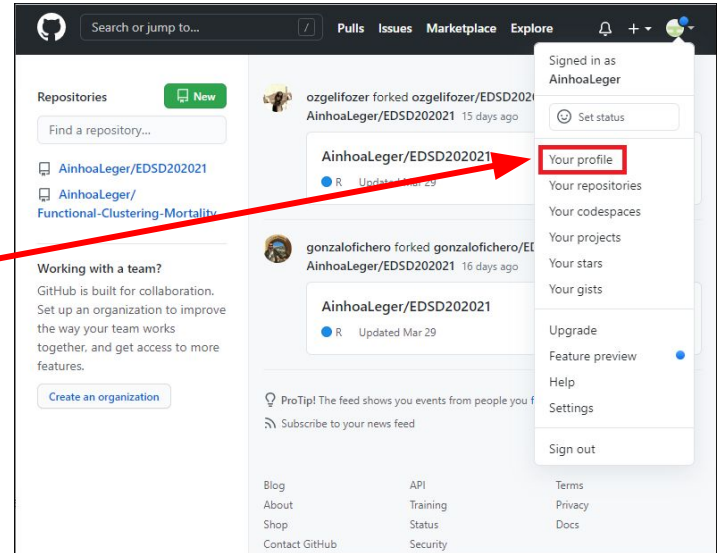
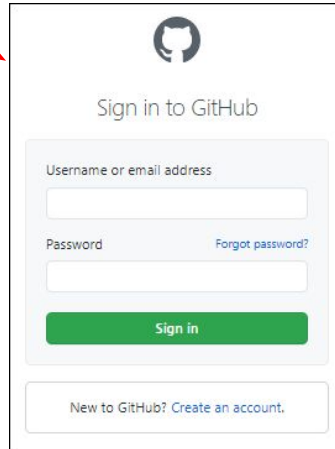
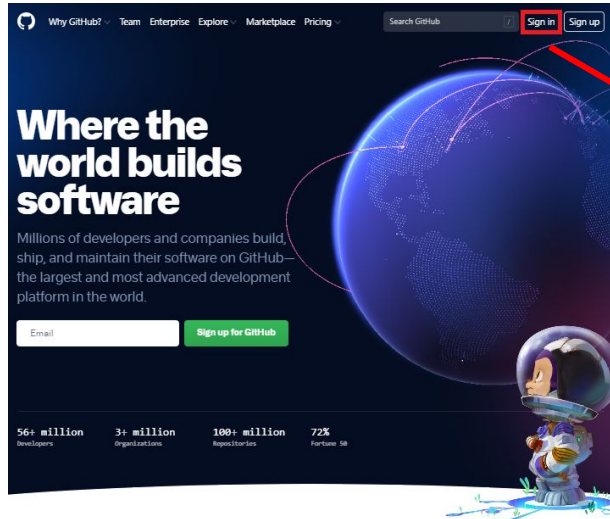
Welcome to GitHub

Woohoo! You've joined millions of developers who are doing their best work on GitHub. Tell us what you're interested in. We'll help you get there.



1.2 Your GitHub account

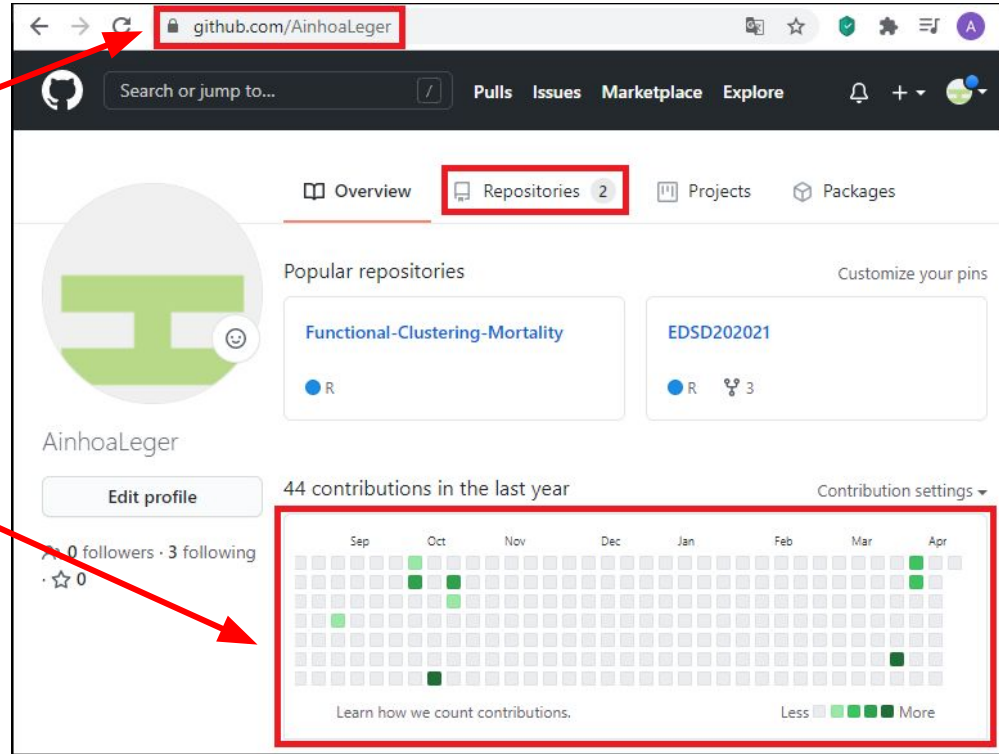
- You can sign in to your GitHub account from the webpage
- The **drop down menu** (topright corner) gives access to the profile and repositories



1.2 Your GitHub account

This is the **URL** that others can go to and see your public activities

This is the **contribution graph**; the darker green the square the more contributions for that day



Topics

Individual use

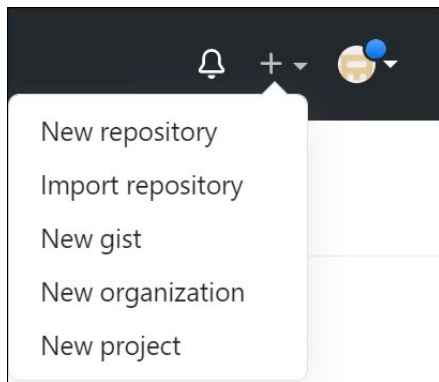
1. Install Git + Setup a GitHub account
- 2. Create a GitHub repository**
3. Connect a GitHub repository to an RStudio project
4. Version control the files
 - a. Commit action
 - b. Push action

Collaborative use

5. Fork GitHub repositories
6. Create Pull Request
7. Accept/Merge changes
8. Refresh forked repo

2. Create a GitHub repository

- We can now create a new **public repository**, where your folder and files (data, R scripts, images) are stored and version controlled
- Each project that you work on usually has its own dedicated repository



The “+” **button** (topright corner) allows you to create a new repository

2. Create a GitHub repository

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere?

[Import a repository.](#)

Owner *

AinhoaLeger ▾

Repository name *

Great repository names are short and memorable. Need inspiration? How about [verbose-octo-spoon?](#)

Description (optional)



Public

Anyone on the internet can see this repository. You choose who can commit.



Private

You choose who can see and commit to this repository.

Initialize this repository with:

Skip this step if you're importing an existing repository.

☐ Add a README file

This is where you can write a long description for your project. [Learn more.](#)

☐ Add .gitignore

Choose which files not to track from a list of templates. [Learn more.](#)

☐ Choose a license

A license tells others what they can and can't do with your code. [Learn more.](#)

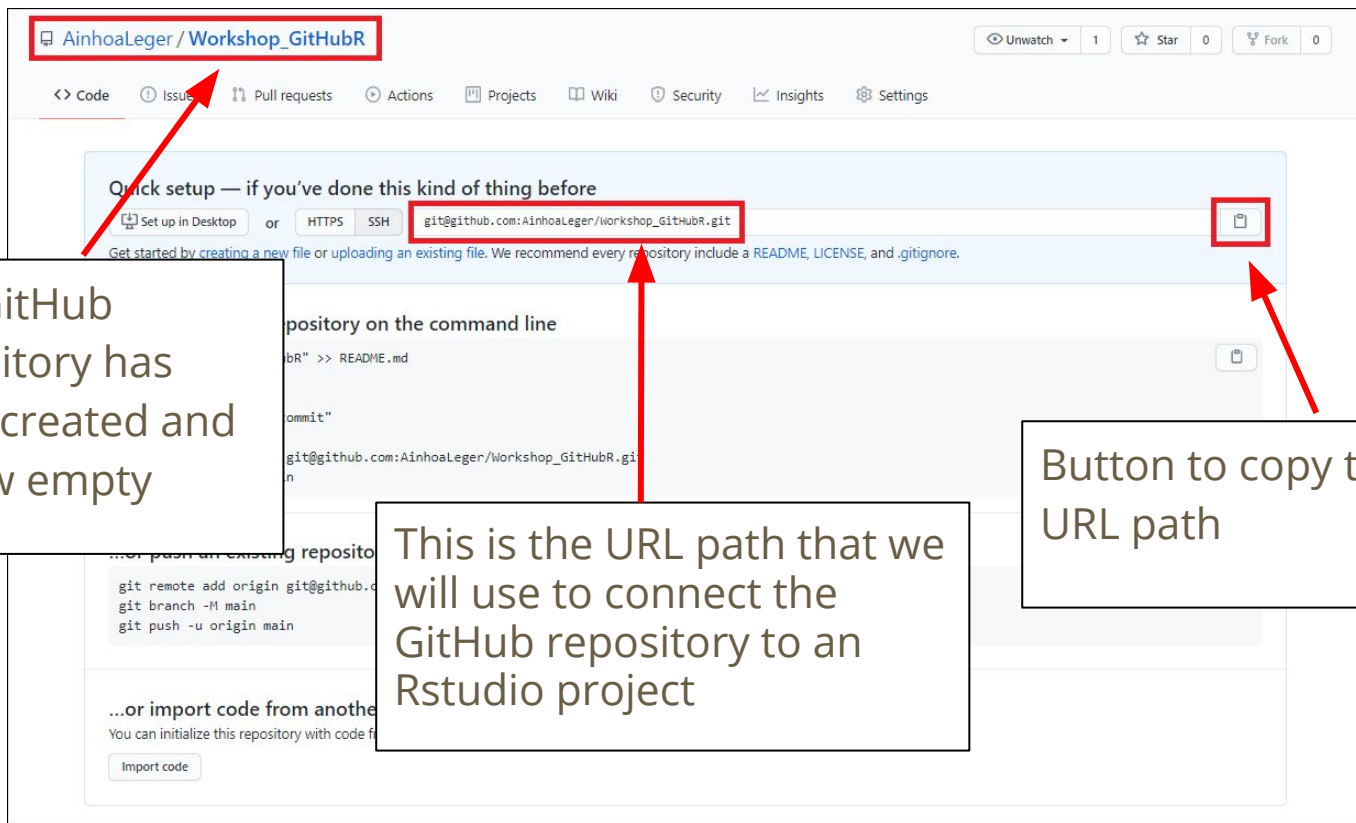
Create repository

This is the repository name. This is also the URL address that others can use to go straight into that project

For example: Workshop_GitHubR

Select the repository type, in this workshop we will work with a public repository

2. Create a GitHub repository



The GitHub repository has been created and is now empty

This is the URL path that we will use to connect the GitHub repository to an Rstudio project

Button to copy the URL path

Topics

Individual use

1. Install Git + Setup a GitHub account
2. Create a GitHub repository
- 3. Connect a GitHub repository to an RStudio project**
4. Version control the files
 - a. Commit action
 - b. Push action

Collaborative use

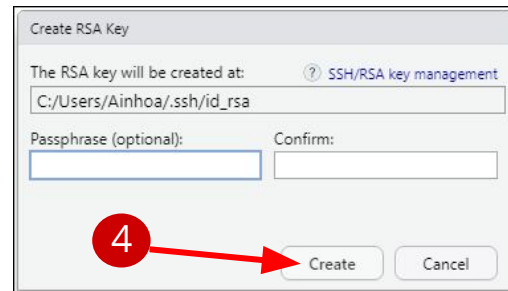
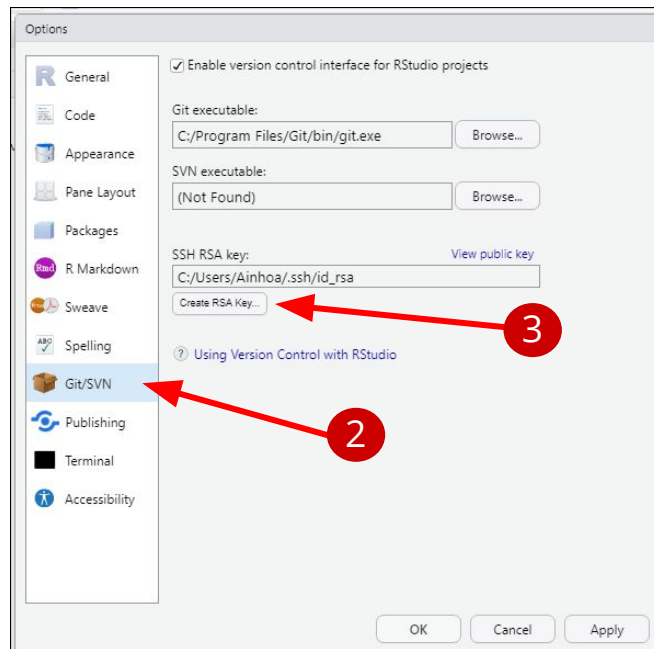
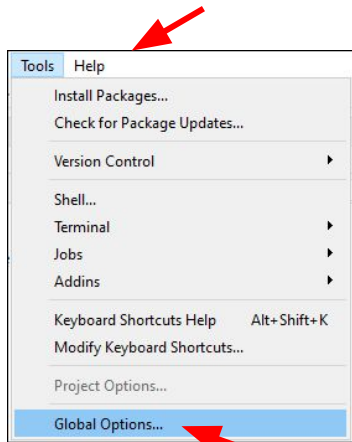
5. Fork GitHub repositories
6. Create Pull Request
7. Accept/Merge changes
8. Refresh forked repo

3. Connect a GitHub repository to an RStudio project

- In order to link up our GitHub account with RStudio (desktop) we need to establish a secure connection. This is done by creating and using an **SSH key** and it only needs to be done once.
- SSH stands for Secure Shell and is a “cryptographic network protocol for operating network services securely over an unsecured network”.
- **Create an SSH key from your RStudio desktop**
Rstudio > Tools > Global options > GIT/SVN > Create RSA key > View public key > copy the key
- **Associate the SSH key with your GitHub account**
Github > Settings > SSH and GPG keys > New SSH key > paste the key to the clipboard
- **Clone a GitHub repository to an R project**
Rstudio > File > New Project > Version Control > Git
Github > Copy the URL path to clipboard
Rstudio > Paste the URL path to Repository URL

3. Connect a GitHub repository to an RStudio project

Create an SSH key from your RStudio desktop

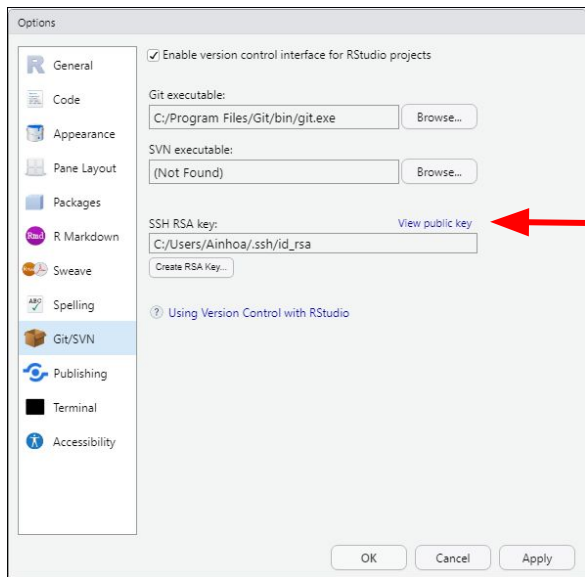


Advice: you can leave the passphrase empty

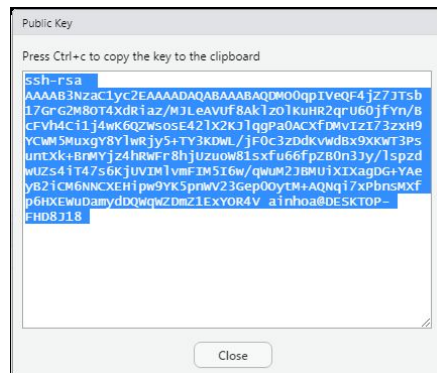


3. Connect a GitHub repository to an RStudio project

Your SSH key



6

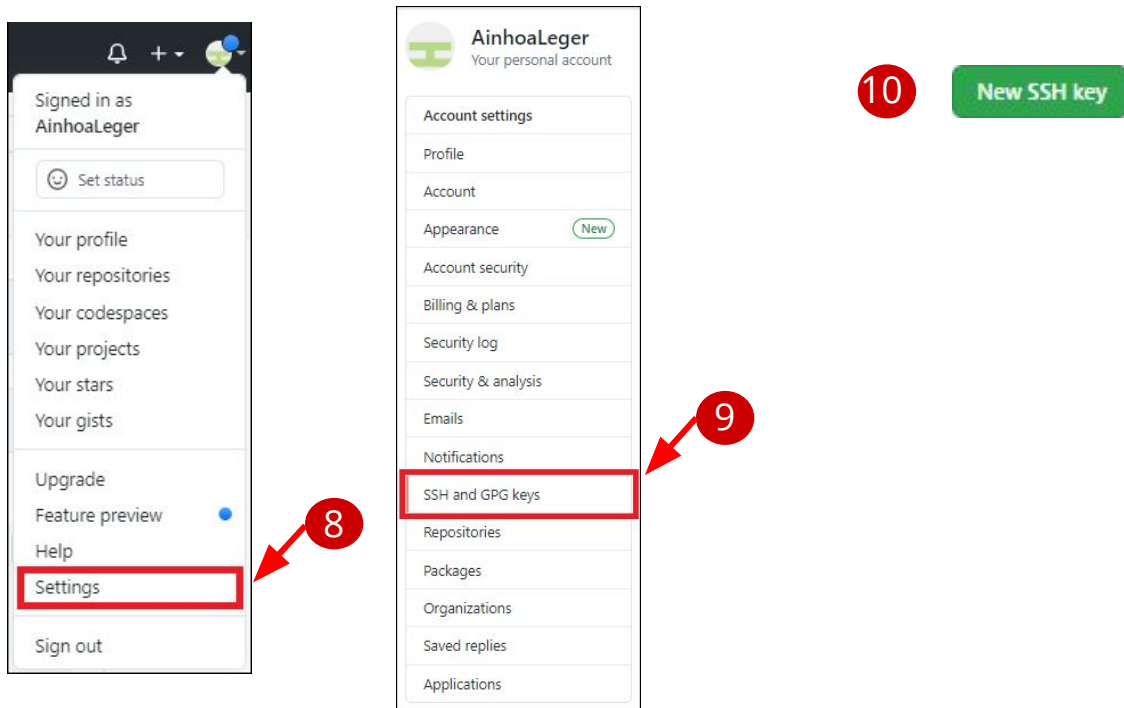


7

Press Ctrl+c to copy the key to the clipboard

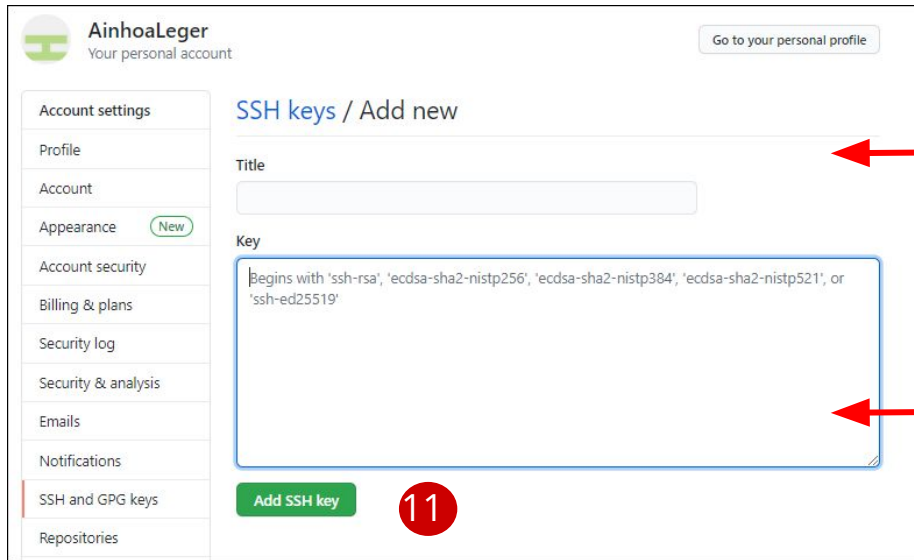
3. Connect a GitHub repository to an RStudio project

Associate the SSH key with your GitHub account



3. Connect a GitHub repository to an RStudio project

Associate the SSH key with your GitHub account



The screenshot shows the GitHub 'SSH keys / Add new' page for user AinhoaLeger. The page has a sidebar with navigation links: Account settings, Profile, Account, Appearance (marked 'New'), Account security, Billing & plans, Security log, Security & analysis, Emails, Notifications, SSH and GPG keys (highlighted), and Repositories. The main content area is titled 'SSH keys / Add new' and contains a 'Title' input field and a 'Key' text area. A red circle with the number '11' is next to the 'Add SSH key' button.

Give this key a title

Paste here the SSH key that you copied earlier from RStudio

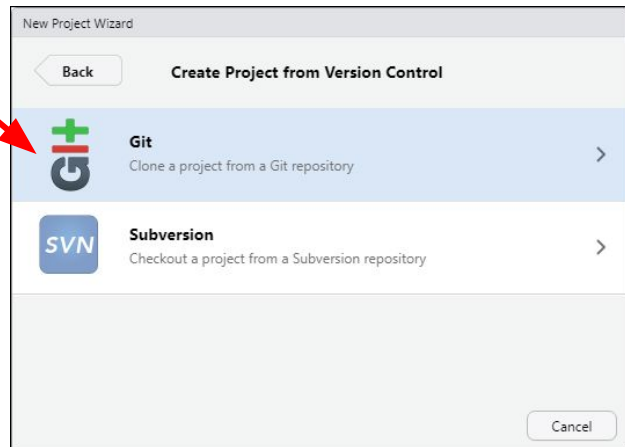
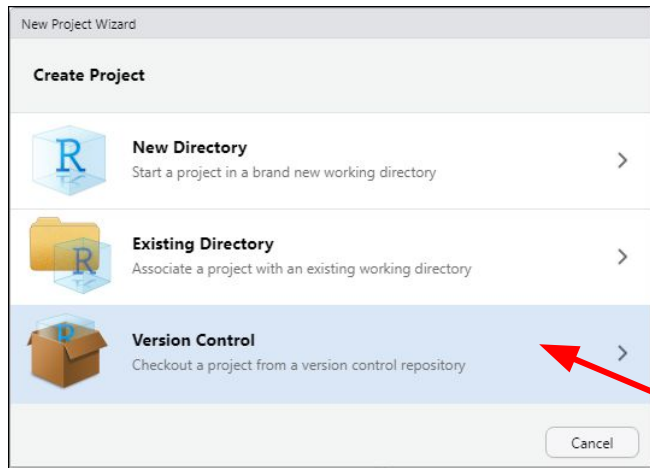
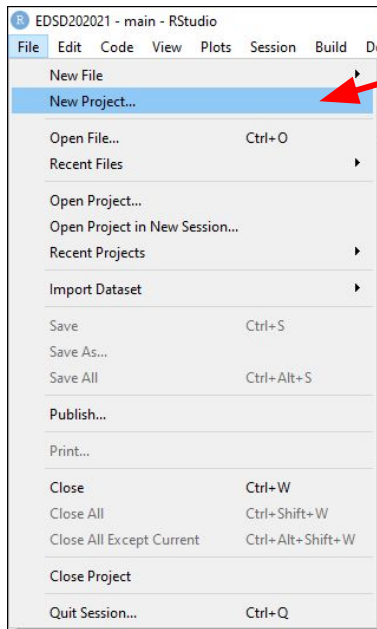
3. Connect a GitHub repository to an RStudio project

Clone a GitHub repository to an R project

- We now want to connect the GitHub repository that we have just created with an R project that will live on your machine
- R projects allow you to divide your work in a structured way.
- Each project has its own setup such as a working directory, workspace and history.
- This is even better and more efficient when it is version controlled!

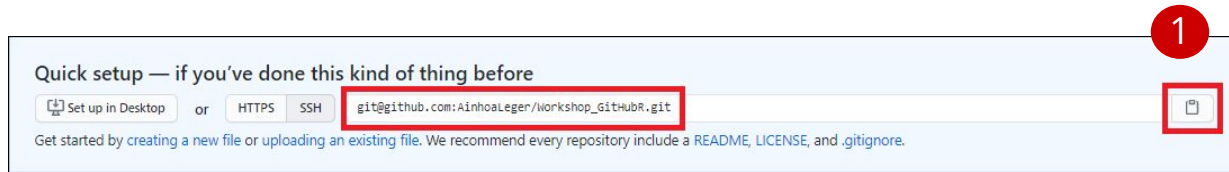
3. Connect a GitHub repository to an RStudio project

Clone a GitHub repository to an R project

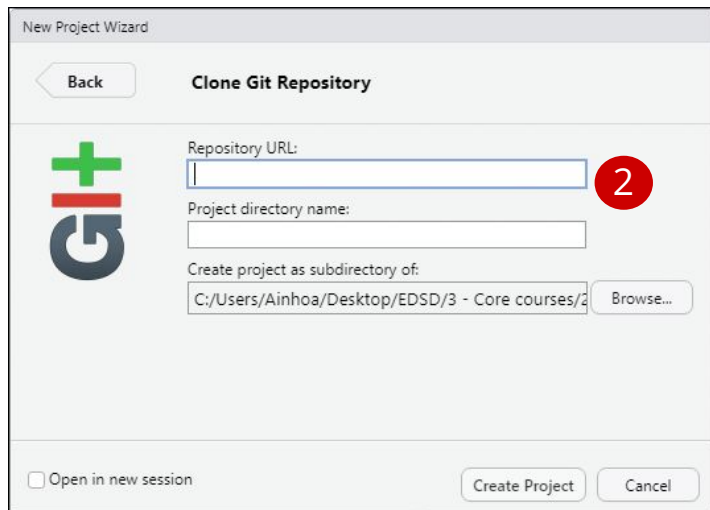


3. Connect a GitHub repository to an RStudio project

Clone a GitHub repository to an R project



1. Copy the URL path to clipboard



2. Paste the URL path to Repository URL. You can edit the project directory name and the folder

3. Click on create project

3. Connect a GitHub repository to an RStudio project

Clone a GitHub repository to an R project

The screenshot shows the RStudio interface with the 'Workshop_GitHubR - RStudio' window. The 'Git' tab is selected in the top toolbar, indicated by a red box and a red arrow. The 'Files' pane on the right shows the project directory 'C:\Users\Ainhoa\Desktop\Github Repositories\Workshop_GitHubR'. The file explorer lists files: '..', '.gitignore', and 'Workshop_GitHubR.Rproj'. The '.Rproj' file is highlighted with a red box and a red arrow. The console shows the R version 4.0.2 (2020-06-22) and the message 'Taking off Again'.

This tells us that we are inside the R project

This is the ".new" tab that should appear that tells you that your R project has added Git functionality, i.e. it is version controlled

This is the .Rproj file associated with this project

Topics

Individual use

1. Install Git + Setup a GitHub account
2. Create a GitHub repository
3. Connect a GitHub repository to an RStudio project

4. Version control the files

a. Commit action

b. Push action

Collaborative use

5. Fork GitHub repositories
6. Create Pull Request
7. Accept/Merge changes
8. Refresh forked repo

4. Version control the files

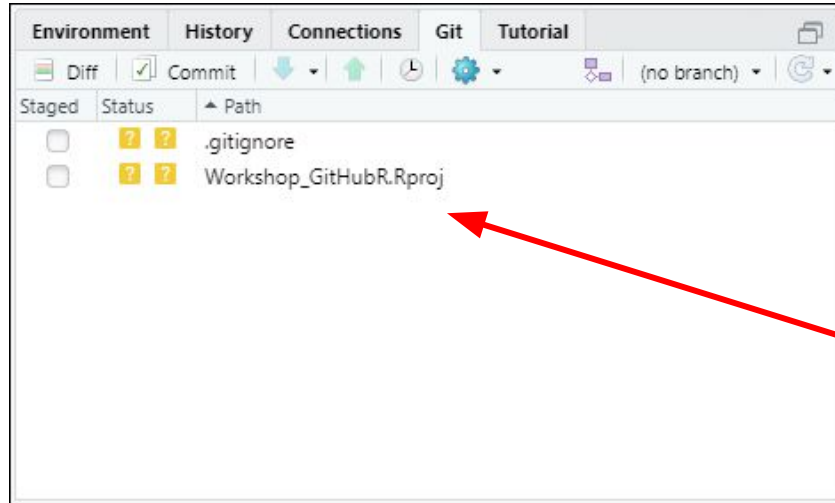
Basic version control processes

Now that our project is version controlled let's cover some of the basic processes to version control a file, keeping things as simple as possible. A file in our repository can be one of four statuses:

1. **Untracked:** The file is not being tracked, i.e. it is not version controlled
2. **Unmodified:** The file is tracked but has not changed
3. **Modified:** The file is tracked and has been changed
4. **Staged:** The file is tracked and the changes are ready to be version controlled

4. Version control the files

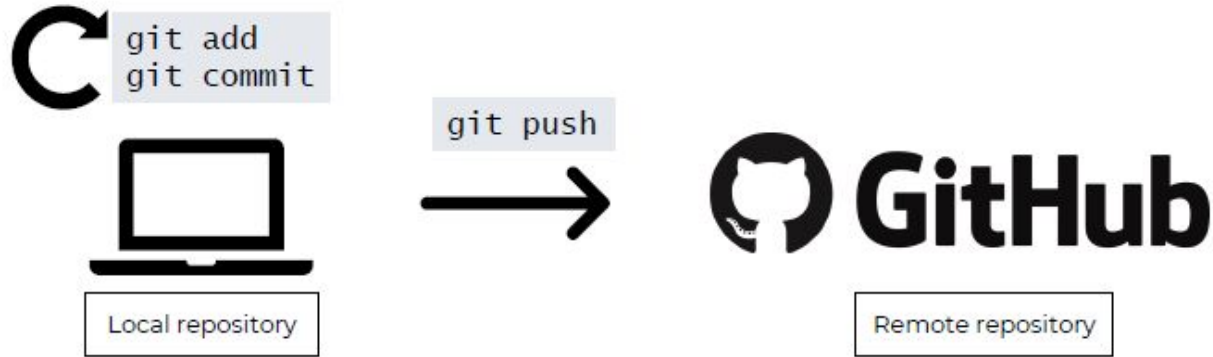
- If we go in “Git” in the top right panel we can see that we have two files with a yellow question mark
- This means that the files are unknown to Github (untracked)



These files are currently untracked

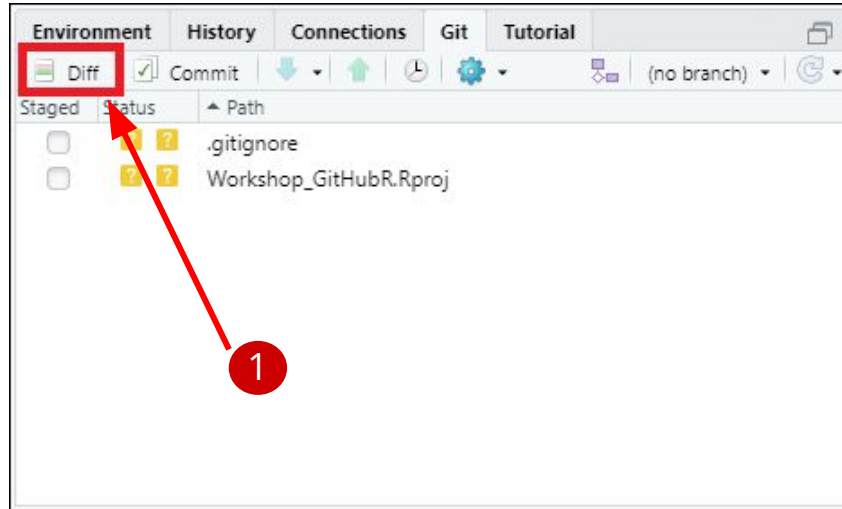
4. Version control the files

- We want to “send” the files stored locally (local repository) or the changes to the file into the remote (online) repository in GitHub
- In this way the file is being version controlled
- We need to add the file to the “staging area”, commit the file and all of its changes, and push them into the GitHub repository



4.a. Commit action

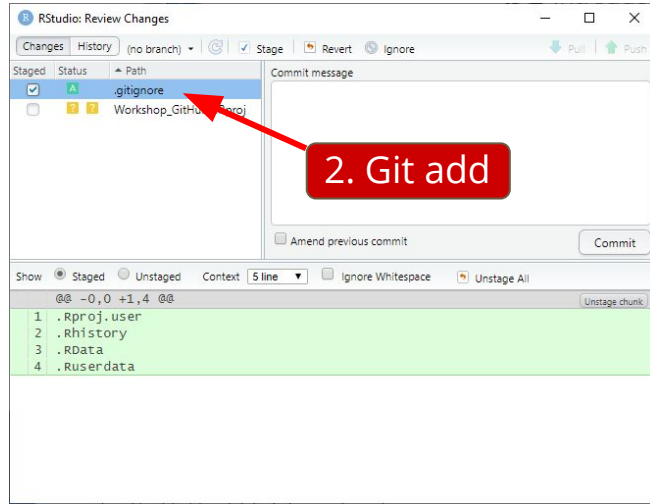
- We will “send” to GitHub the .gitignore file
- The .gitignore is a very useful (hidden) file that communicates to our GitHub repository what files/folders should be ignored and not version controlled



Steps to commit a file/change

1. Click on the Diff button in the Git environment

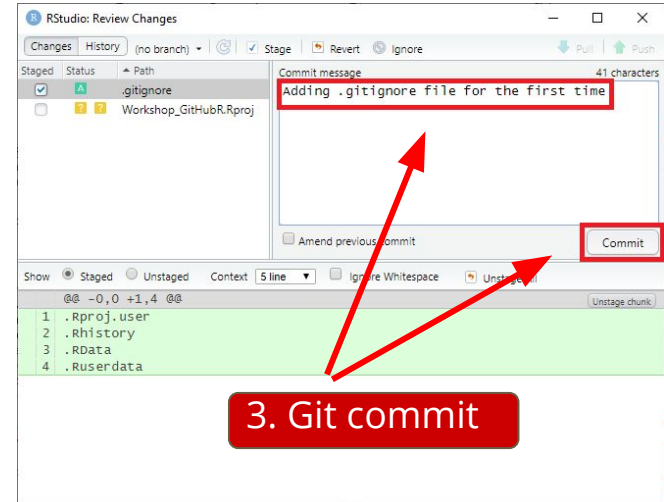
4.a. Commit action



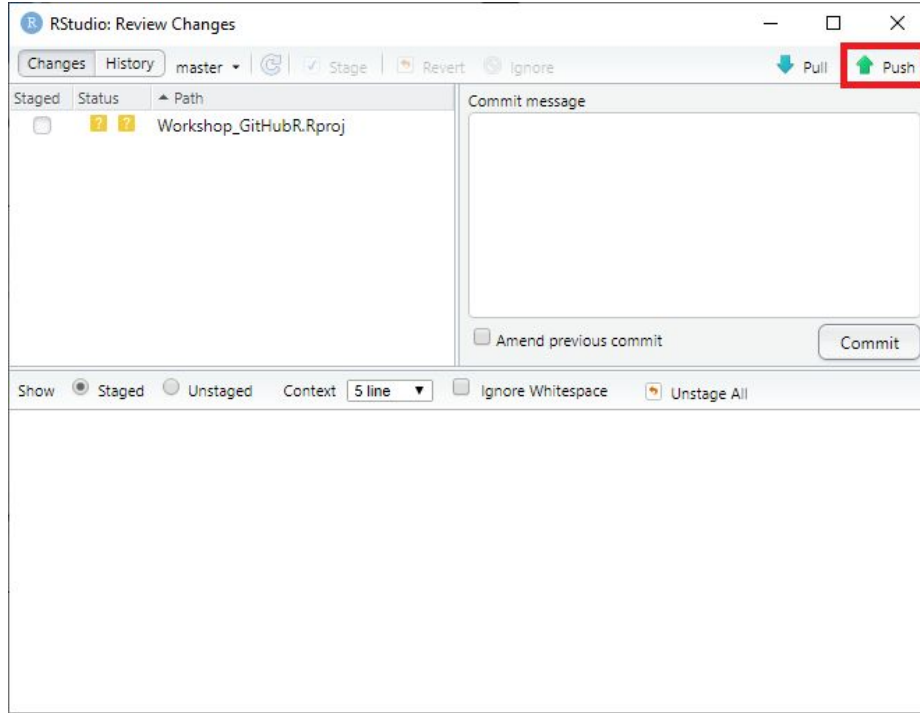
2. Click on the file or changes you want to add to "staging area" (green highlight is added changes, red highlight is deleted changes)

3. Commit the file and all of its changes with some useful comment about the changes

4. Close the pop-up window



4.b. Push action



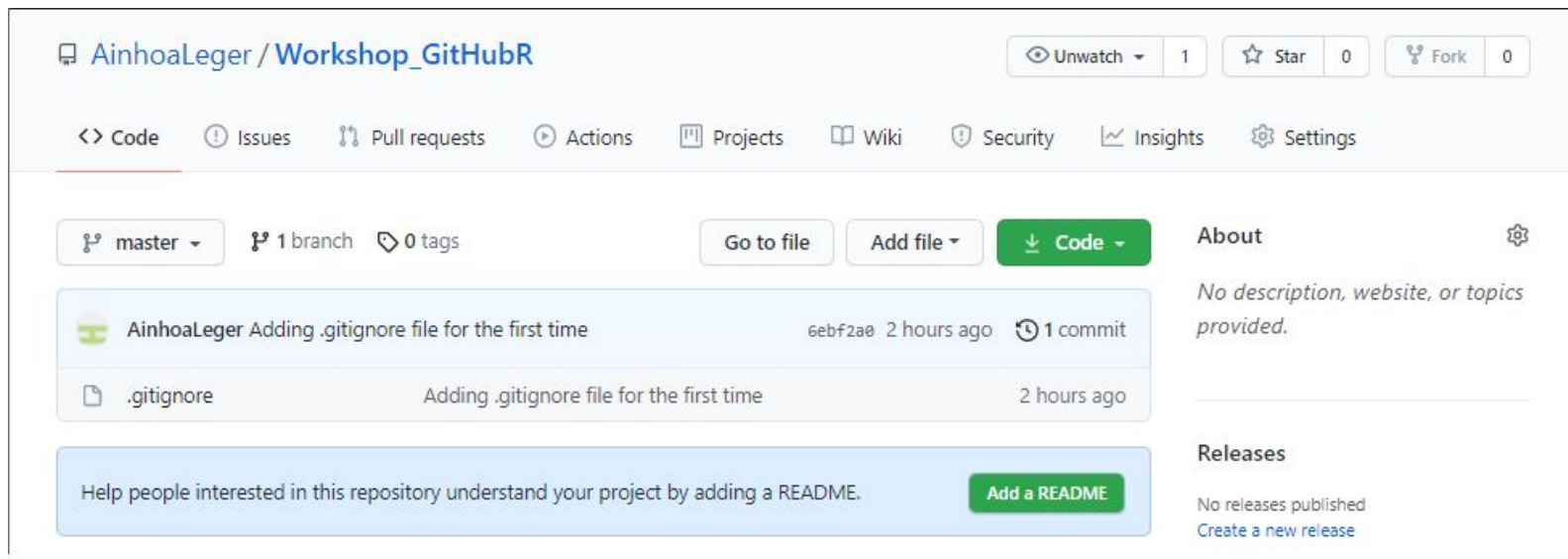
4. Git push

Click the push button to send the files/changes to your remote (online) repository in GitHub

A master branch has been created

4.b. Push action

Congratulations! You have just added, committed and pushed a file into a GitHub repository within an R project in RStudio desktop!



PART 2

Now it gets collaborative

Topics

Individual use

1. Install Git + Setup a GitHub account
2. Create a GitHub repository
3. Connect a GitHub repository to an RStudio project
4. Version control the files
 - a. Commit action
 - b. Push action

Collaborative use

5. Fork GitHub repositories

6. Create Pull Request
7. Accept/Merge changes
8. Refresh forked repo

5. Forking* repos

Fork a repo



A fork is a copy of a repository. Forking a repository allows you to freely experiment with changes without affecting the original project.

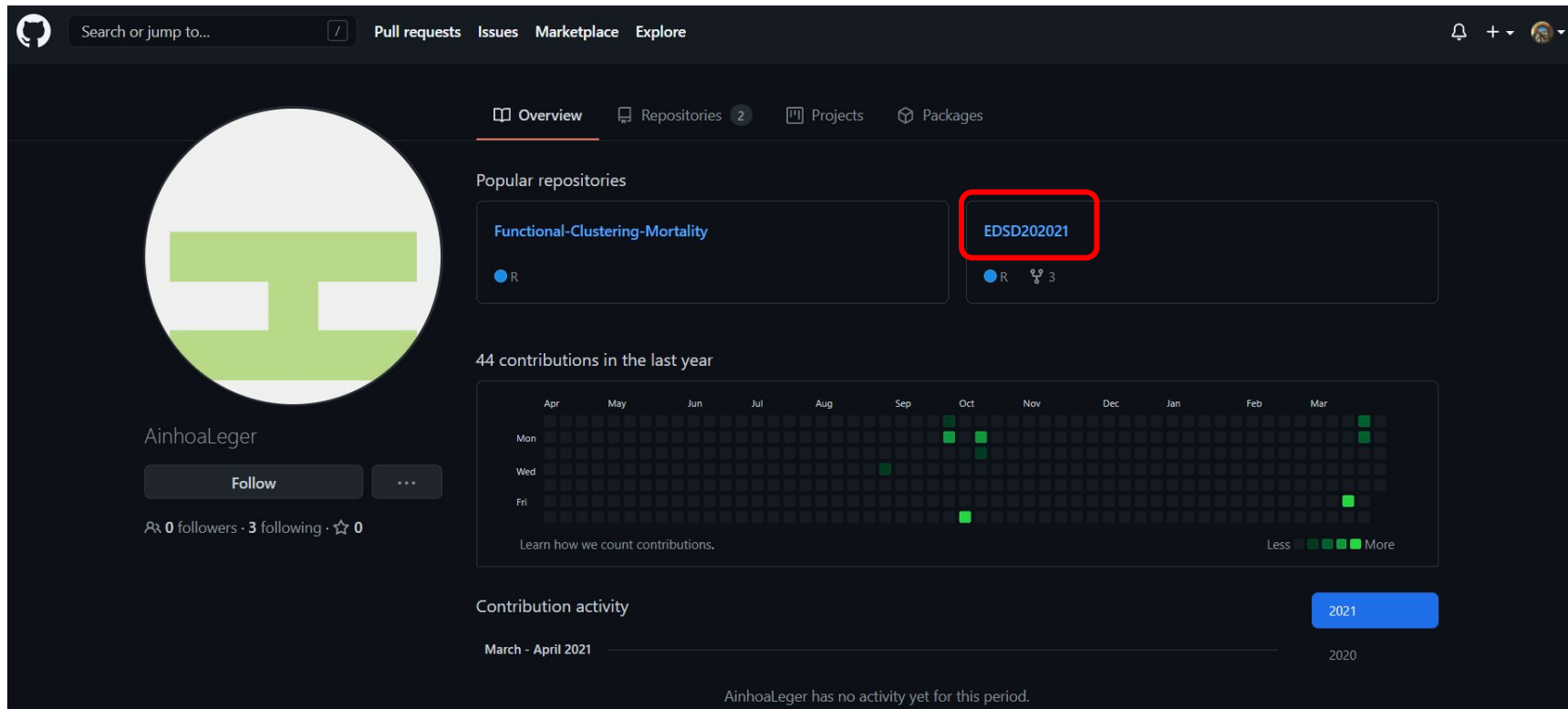
Most commonly, forks are used to either propose changes to someone else's project or to use someone else's project as a starting point for your own idea.

Propose changes to someone else's project

For example, you can use forks to propose changes related to fixing a bug. Rather than logging an issue for a bug you've found, you can:

- Fork the repository.
- Make the fix.
- Submit a pull request to the project owner.

5. Forking repos



Search or jump to... Pull requests Issues Marketplace Explore

Overview Repositories 2 Projects Packages

Popular repositories

- Functional-Clustering-Mortality
- EDSD202021

44 contributions in the last year

Learn how we count contributions.

Contribution activity

March - April 2021

AinhoaLeger has no activity yet for this period.

5. Forking repos

AinhoaLeger / EDSD202021

Watch 1 Star 0 Fork 3

<> Code Issues Pull requests Actions Projects Wiki Security Insights

main 1 branch 0 tags

Go to file Add file Code

AinhoaLeger · 57e507d 10 days ago 38 commits

Basic Mathematics E120	.	6 months ago
Computer Programming E140	.	6 months ago
Stable Populations	.	10 days ago
.gitignore	Adding .gitignore to project	6 months ago
EDSD202021.Rproj	First commit.	6 months ago
README.md	Initial commit	6 months ago

README.md

EDSD202021

About

No description, website, or topics provided.

Readme

Releases

No releases published

Packages

No packages published

Contributors 3

- AinhoaLeger
- gonzalofichero

5. Forking repos

The screenshot shows a GitHub repository page for **gonzalofichero / EDSD202021**. The repository is a fork of **AinhoaLeger / EDSD202021**. The interface includes navigation tabs for Code, Pull requests, Actions, Projects, Wiki, Security, Insights, and Settings. The main content area shows the repository's state: 1 branch (main), 0 tags, and 40 commits. A message indicates the branch is 2 commits ahead of the parent repository. A list of files and folders is displayed, including Basic Mathematics E120, Computer Programming E140, Stable Populations, .gitignore, EDSD202021.Rproj, and README.md. The right sidebar contains sections for About, Releases, Packages, and Languages.

Repository Information:

- Owner: gonzalofichero
- Repository: EDSD202021
- Forked from: AinhoaLeger/EDSD202021
- Watch: 0
- Star: 0
- Fork: 3

Navigation:

- <> Code
- Pull requests
- Actions
- Projects
- Wiki
- Security
- Insights
- Settings

Repository State:

- main (1 branch)
- 0 tags
- Go to file
- Add file
- Code

Commit History:

- This branch is 2 commits ahead of AinhoaLeger:main.
- gonzalofichero Merge pull request #3 from AinhoaLeger/main ... b4eca4b 10 days ago 40 commits

Files and Folders:

- Basic Mathematics E120 (6 months ago)
- Computer Programming E140 (6 months ago)
- Stable Populations (10 days ago)
- .gitignore (Adding .gitignore to project, 6 months ago)
- EDSD202021.Rproj (First commit, 6 months ago)
- README.md (Initial commit, 6 months ago)

Right Sidebar:

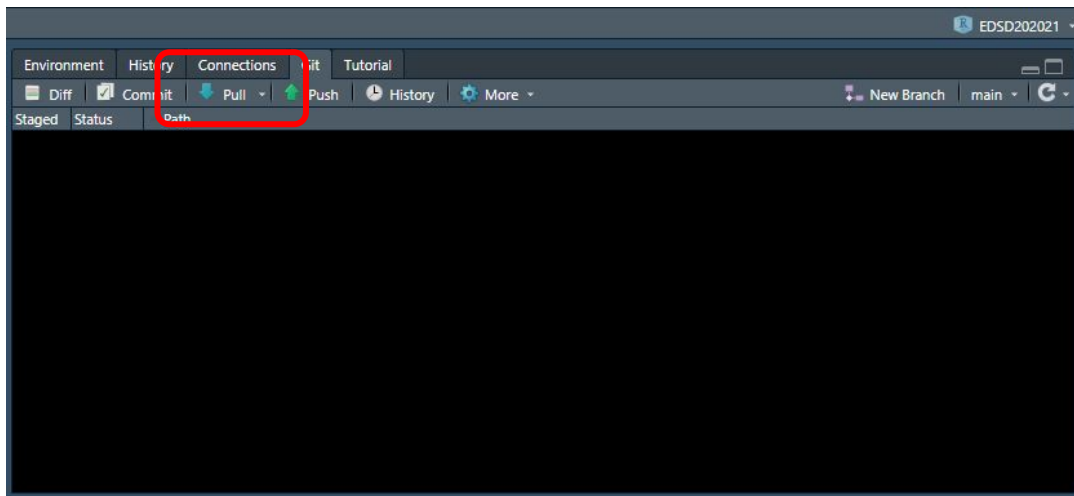
- About:** No description, website, or topics provided.
- Releases:** No releases published. [Create a new release](#)
- Packages:** No packages published. [Publish your first package](#)
- Languages:** (Progress bar showing C++ and R)

5.a. Pull* action

Instead of pushing (sending), this time we need to bring information (pulling).

Just 2 simple steps:

1. Create R project just like Ainhoa showed
- 2.



5.b. Commit + Push on forked repo

- Same as Ainhua actions.
- What's different then?
 - Changes will be pushed to repository in our **own** Github
 - Other people won't be modified (yet)
 - We are 1 step closer to work collaborative

Topics

Individual use

1. Install Git + Setup a GitHub account
2. Create a GitHub repository
3. Connect a GitHub repository to an RStudio project
4. Version control the files
 - a. Commit action
 - b. Push action

Collaborative use

5. Fork GitHub repositories
- 6. Create Pull Request**
7. Accept/Merge changes
8. Refresh forked repo

6. Pull Request*

About pull requests



Pull requests let you tell others about changes you've pushed to a branch in a repository on GitHub. Once a pull request is opened, you can discuss and review the potential changes with collaborators and add follow-up commits before your changes are merged into the base branch.

6. Pull Request

gonzalofichero / EDSD202021

Forked from AinhoaLeger/EDSD202021

Code Pull requests Actions Projects Wiki Security Insights Settings

main 1 branch 0 tags

Go to file Add file Code

This branch is 2 commits ahead of AinhoaLeger:main. Pull request Compare

gonzalofichero Merge pull request #3 from AinhoaLeger/main ... b4eca4b 10 days ago 40 commits

Basic Mathematics E120	.	6 months ago
Computer Programming E140	.	6 months ago
Stable Populations	.	10 days ago
.gitignore	Adding .gitignore to project	6 months ago
EDSD202021.Rproj	First commit.	6 months ago
README.md	Initial commit	6 months ago

README.md

About

No description, website, or topics provided.

Readme

Releases

No releases published
[Create a new release](#)

Packages

No packages published
[Publish your first package](#)

Languages

R 74.3% TeX 25.7%

6. Pull Request

Comparing changes

Choose two branches to see what's changed or to start a new pull request. If you need to, you can also [compare across forks](#).

base repository: AinhoaLeger/EDSD202021 ▾ base: main ▾ ← head repository: gonzalofichero/EDSD202021 ▾ compare: main ▾

✓ **Able to merge.** These branches can be automatically merged.



Discuss and review the changes in this comparison with others. [Learn about pull requests](#)

Create pull request

2 commits

0 files changed

0 comments

1 contributor

Commits on Mar 29, 2021

Merge pull request #2 from AinhoaLeger/main ...

Verified c2bd2fa

Merge pull request #3 from AinhoaLeger/main ...

Verified b4eca4b

Showing 0 changed files with 0 additions and 0 deletions.

Unified Split

Now is the other guy's problem...

Topics

Individual use

1. Install Git + Setup a GitHub account
2. Create a GitHub repository
3. Connect a GitHub repository to an RStudio project
4. Version control the files
 - a. Commit action
 - b. Push action

Collaborative use

5. Fork GitHub repositories
6. Create Pull Request
- 7. Accept/Merge changes**
8. Refresh forked repo

7. Accept/Merge changes

7. Accept/Merge changes

Step 1

7. Accept/Merge changes

You got Mail

7. Accept/Merge changes

The screenshot shows a GitHub pull request interface. At the top, the repository name 'AinhoaLeger / EDSD202021' is highlighted with a red box. Below it, the pull request title 'Adding Comments to Ex1 #5' is displayed. A green 'Open' button is visible. The pull request description states: 'gonzalofichero wants to merge 4 commits into AinhoaLeger:main from gonzalofichero:main'. Below the description, there are tabs for 'Conversation' (0), 'Commits' (4), 'Checks' (0), and 'Files changed' (1). The 'Commits' tab is selected, showing a list of four commits added 22 days ago by gonzalofichero. These commits are highlighted with a red box: 'Merge pull request #2 from AinhoaLeger/main', 'Merge pull request #3 from AinhoaLeger/main', 'Merge pull request #4 from AinhoaLeger/main', and 'add comments to all ex1'. To the right of the commit list, there are verification status indicators (Verified) and commit hashes (c2bd2fa, b4eca4b, 711f8db, 637bc34). On the far right, there are sections for 'Reviewers' (AinhoaLeger), 'Assignees' (No one—assign yourself), and 'Labels' (None yet).

AinhoaLeger / EDSD202021

Unwatch 1 Star 0 Fork 4

Code Issues Pull requests 1 Actions Projects Wiki Security Insights Settings

Adding Comments to Ex1 #5

Edit Open with

Open gonzalofichero wants to merge 4 commits into AinhoaLeger:main from gonzalofichero:main

Conversation 0 Commits 4 Checks 0 Files changed 1 +40 -4

gonzalofichero commented 5 days ago Contributor

Adding comments to Ex1 Sequence Analysis
Adding print() to Question 12 to force showing summary results

gonzalofichero added 4 commits 22 days ago

- Merge pull request #2 from AinhoaLeger/main Verified c2bd2fa
- Merge pull request #3 from AinhoaLeger/main Verified b4eca4b
- Merge pull request #4 from AinhoaLeger/main Verified 711f8db
- add comments to all ex1 637bc34

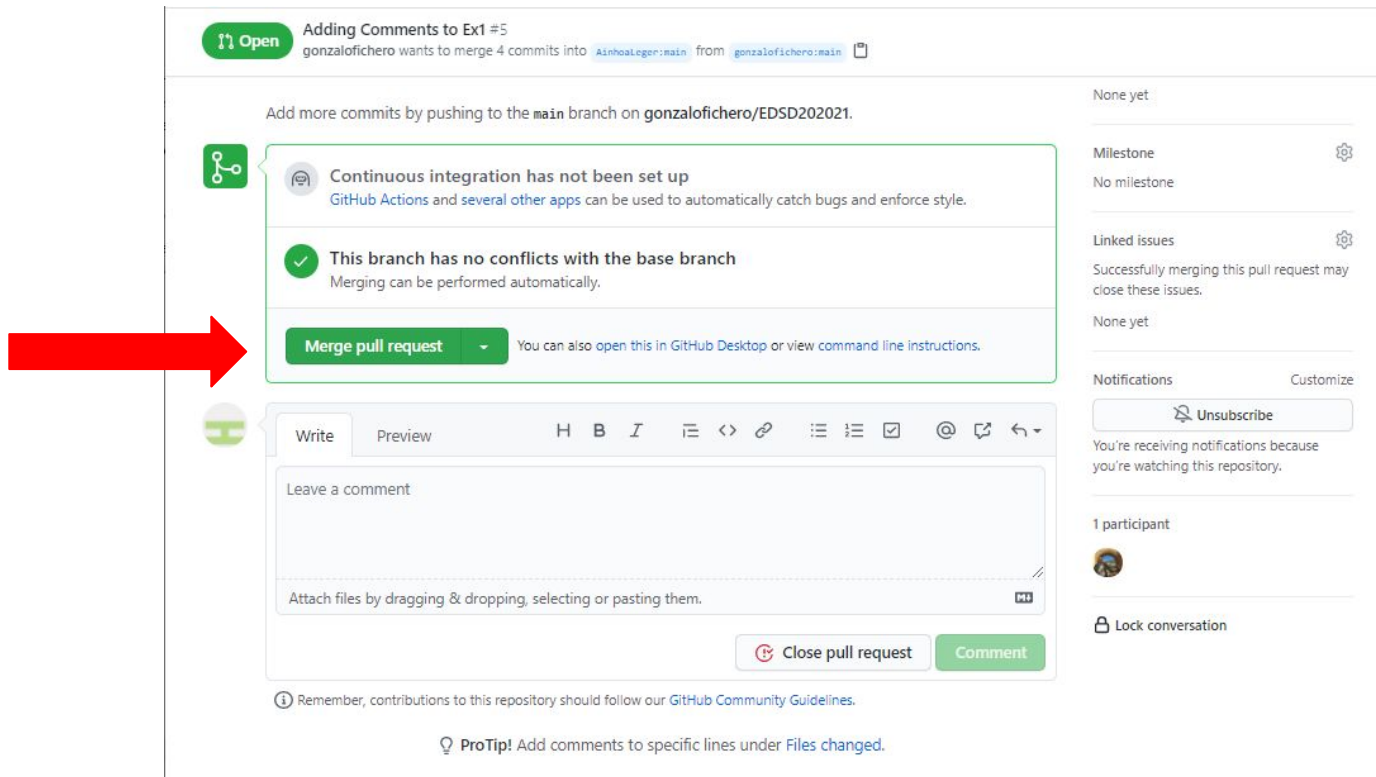
Reviewers
Suggestions
AinhoaLeger Request
Still in progress? Convert to draft

Assignees
No one—assign yourself

Labels
None yet

Projects
None yet

7. Accept/Merge changes



The screenshot shows a GitHub pull request interface. At the top, a green button labeled "Open" is next to the title "Adding Comments to Ex1 #5". Below the title, it says "gonzalofichero wants to merge 4 commits into `AinhoaLeger:main` from `gonzalofichero:main`".

Below this, a message says "Add more commits by pushing to the `main` branch on `gonzalofichero/EDSD202021`."

The main content area has a green border and contains three sections:

- A warning icon and text: "Continuous integration has not been set up. GitHub Actions and several other apps can be used to automatically catch bugs and enforce style."
- A green checkmark icon and text: "This branch has no conflicts with the base branch. Merging can be performed automatically."
- A green button labeled "Merge pull request" with a dropdown arrow. To its right, it says "You can also open this in GitHub Desktop or view command line instructions."

A large red arrow points from the left towards the "Merge pull request" button.

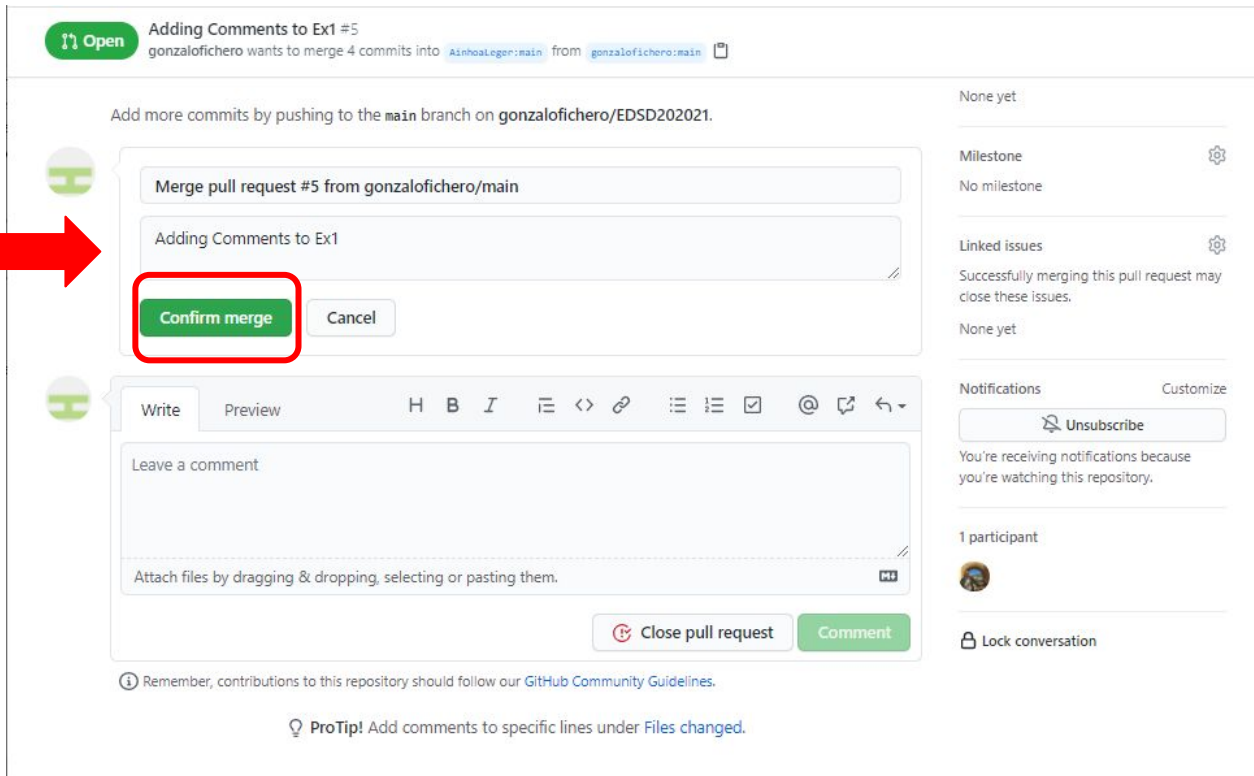
Below the merge button is a comment section with a "Write" tab, a "Preview" tab, and a rich text editor. The editor has a placeholder "Leave a comment" and a note "Attach files by dragging & dropping, selecting or pasting them." At the bottom of the comment section are two buttons: "Close pull request" and "Comment".

At the bottom of the page, there is a footer with a reminder: "Remember, contributions to this repository should follow our GitHub Community Guidelines." and a "ProTip!" that says "Add comments to specific lines under Files changed."

On the right side of the interface, there are several sections:

- "None yet"
- "Milestone" with a gear icon and "No milestone"
- "Linked issues" with a gear icon and "Successfully merging this pull request may close these issues."
- "None yet"
- "Notifications" with a "Customize" link and an "Unsubscribe" button. Below this, it says "You're receiving notifications because you're watching this repository."
- "1 participant" with a profile picture of a person.
- "Lock conversation" with a lock icon.

7. Accept/Merge changes



Open Adding Comments to Ex1 #5
gonzalofichero wants to merge 4 commits into `AinhoaLeger:main` from `gonzalofichero:main`

Add more commits by pushing to the `main` branch on `gonzalofichero/EDSD202021`.

Merge pull request #5 from `gonzalofichero/main`

Adding Comments to Ex1

Confirm merge Cancel

Write Preview H B I

Leave a comment

Attach files by dragging & dropping, selecting or pasting them.

Close pull request **Comment**

Remember, contributions to this repository should follow our [GitHub Community Guidelines](#).

ProTip! Add comments to specific lines under [Files changed](#).

None yet

Milestone
No milestone

Linked issues
Successfully merging this pull request may close these issues.

None yet

Notifications [Customize](#)
 Unsubscribe
You're receiving notifications because you're watching this repository.

1 participant

Lock conversation

7. Accept/Merge changes

The screenshot displays a GitHub pull request interface. At the top, a purple button with a checkmark and the word "Merged" is highlighted with a red rectangle. To its right, the title "Adding Comments to Ex1 #5" is shown, followed by the text "AinhoaLeger merged 4 commits into AinhoaLeger:main from gonzalofichero:main" and a clock icon indicating "16 seconds ago".

Below this, a commit history section shows a list of commits by gonzalofichero, added 22 days ago. The commits are:

- Merge pull request #2 from AinhoaLeger/main (Verified, c2bd2fa)
- Merge pull request #3 from AinhoaLeger/main (Verified, b4eca4b)
- Merge pull request #4 from AinhoaLeger/main (Verified, 711f0db)
- add comments to all ex1 (637bc34)

Below the commit history, a green checkmark icon is followed by the text "AinhoaLeger merged commit 50b6e2f into AinhoaLeger:main 16 seconds ago". A "Revert" button is visible to the right.

The bottom section features a comment input area with a "Write" tab selected. It includes a rich text editor with formatting options (H, B, I, list, link, etc.) and a "Preview" tab. The input area contains the placeholder text "Leave a comment" and a message "Attach files by dragging & dropping, selecting or pasting them." A green "Comment" button is at the bottom right of the input area.

On the right side, there are several sections:

- Labels:** None yet
- Projects:** None yet
- Milestone:** No milestone
- Linked issues:** Successfully merging this pull request may close these issues. None yet
- Notifications:** Customizable. An "Unsubscribe" button is present. A message states: "You're receiving notifications because you're watching this repository."
- 2 participants:** Two user avatars are shown.
- Lock conversation:** A lock icon and the text "Lock conversation".

At the bottom, a reminder states: "Remember, contributions to this repository should follow our GitHub Community Guidelines." A "ProTip!" icon is followed by the text: "Add comments to specific lines under Files changed."

7. Accept/Merge changes

main 1 branch 0 tags

Go to file Add file Code

AinhoaLeger Merge pull request #5 from gonzalofichero/m... 50b6e2f 2 minutes ago 49 commits

Basic Mathematics E120	.	6 months ago
Computer Programming E140	.	6 months ago
Sequence Analysis	add comments to all ex1	5 days ago
Stable Populations	.	22 days ago
.gitignore	Adding .gitignore to project	6 months ago
EDSD202021.Rproj	First commit.	6 months ago
README.md	Initial commit	6 months ago

README.md

EDSD202021

About

No description, website, or topics provided.

Readme

Releases

No releases published
[Create a new release](#)

Packages

No packages published
[Publish your first package](#)


Contributors 3

AinhoaLeger

gonzalofichero

7. Accept/Merge changes

main ▾ EDSD202021 / Sequence Analysis / **Always last commit** Go to file Add file ▾ ...

 gonzalofichero add comments to all ex1 ... 637bc34 5 days ago ⌚ History

..		
Assignment.Rmd	add comments to all ex1	5 days ago
Assignment.pdf	.	6 days ago
seqanalysis_ref.bib	.	6 days ago

Topics

Individual use

1. Install Git + Setup a GitHub account
2. Create a GitHub repository
3. Connect a GitHub repository to an RStudio project
4. Version control the files
 - a. Commit action
 - b. Push action

Collaborative use

5. Fork GitHub repositories
6. Create Pull Request
7. Accept/Merge changes
- 8. Refresh forked repo**

8. Refresh forked repo

8. Refresh forked repo

The screenshot shows a GitHub repository interface. At the top, the repository name 'gonzalofichero / EDSD202021' is highlighted with a red box, with a note 'forked from AinhoaLeger/EDSD202021'. Navigation tabs include Code, Pull requests, Actions, Projects, Wiki, Security, Insights, and Settings. On the right, there are buttons for Watch (0), Star (0), and Fork (4). Below the navigation, a status bar shows 'main' branch, '1 branch', and '0 tags'. A red box highlights a message: 'This branch is 1 commit behind AinhoaLeger:main.' To the right of this message are links for 'Pull request' and 'Compare'. Below the status bar is a list of files and commits:

File/Commit	Commit Hash	Time Ago	Commits
gonzalofichero add comments to all ex1 ...	637bc34	5 days ago	48 commits
Basic Mathematics E120	.	6 months ago	
Computer Programming E140	.	6 months ago	
Sequence Analysis	add comments to all ex1	5 days ago	
Stable Populations	.	22 days ago	
.gitignore	Adding .gitignore to project	6 months ago	
EDSD202021.Rproj	First commit.	6 months ago	
README.md	Initial commit	6 months ago	

At the bottom, there is a 'README.md' section with an edit icon. On the right side of the repository page, there are sections for 'About' (No description, website, or topics provided), 'Releases' (No releases published, Create a new release), 'Packages' (No packages published, Publish your first package), and 'Languages' (R 73.1%, TeX 26.9%).

8. Refresh forked repo

gonzalofichero / EDSD202021
forked from AinhoaLeger/EDSD202021

Watch 0 Star 0 Fork 4

Code Pull requests Actions Projects Wiki Security Insights Settings

main 1 branch 0 tags

Go to file Add file Code

This branch is 1 commit behind AinhoaLeger:main.

Pull request Compare

gonzalofichero add comments to all ex1 ... 637bc34 5 days ago 48 commits

Basic Mathematics E120	.	6 months ago
Computer Programming E140	.	6 months ago
Sequence Analysis	add comments to all ex1	5 days ago
Stable Populations	.	22 days ago
.gitignore	Adding .gitignore to project	6 months ago
EDSD202021.Rproj	First commit.	6 months ago
README.md	Initial commit	6 months ago

README.md

About

No description, website, or topics provided.

Readme

Releases

No releases published
[Create a new release](#)

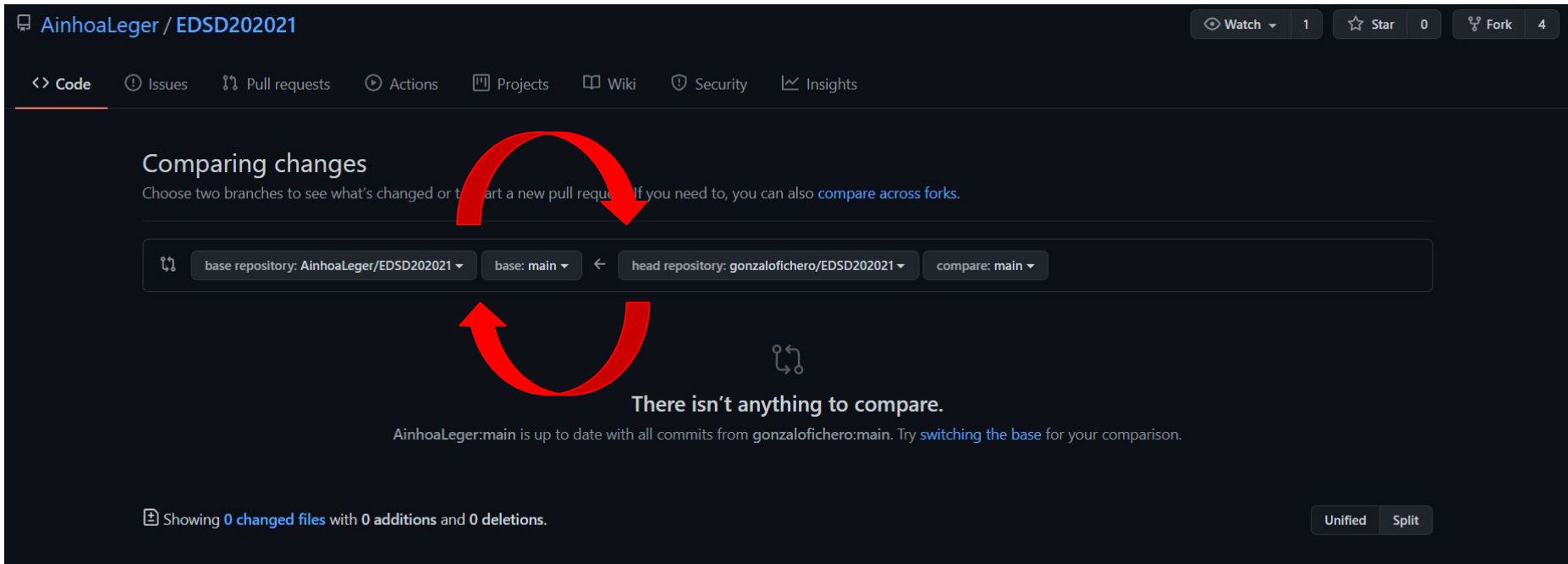
Packages

No packages published
[Publish your first package](#)

Languages

R 73.1% TeX 26.9%

8. Refresh forked repo



The screenshot shows the GitHub interface for the repository `AinhoaLeger / EDSD202021`. The top navigation bar includes links for `<> Code`, `Issues`, `Pull requests`, `Actions`, `Projects`, `Wiki`, `Security`, and `Insights`. On the right, there are buttons for `Watch` (1), `Star` (0), and `Fork` (4).

The main section is titled `Comparing changes` with the subtitle `Choose two branches to see what's changed or to start a new pull request. If you need to, you can also compare across forks.`

Below the subtitle, there is a comparison bar with the following settings:

- `base repository: AinhoaLeger/EDSD202021`
- `base: main`
- `head repository: gonzalofichero/EDSD202021`
- `compare: main`

Two red curved arrows are drawn on the screen. One arrow starts at the `base repository` dropdown and points to the `head repository` dropdown. The other arrow starts at the `head repository` dropdown and points back to the `base repository` dropdown, indicating a cycle or refresh action.

Below the comparison bar, there is a message: `There isn't anything to compare.` with a subtext: `AinhoaLeger:main is up to date with all commits from gonzalofichero:main. Try switching the base for your comparison.`


At the bottom left, it says: `Showing 0 changed files with 0 additions and 0 deletions.`

At the bottom right, there are buttons for `Unified` and `Split`.

8. Refresh forked repo

Comparing changes

Choose two branches to see what's changed or to start a new pull request. If you need to, you can also [compare across forks](#).

 base repository: AinhoaLeger/EDSD202021 ▾ base: main ▾ ← head repository: gonzalofichero/EDSD202021 ▾ compare: main ▾

Choose a Base Repository


✓ AinhoaLeger/EDSD202021

gonzalofichero/EDSD202021

PinkyPao/EDSD202021

liliana-calderon/EDSD202021

ozgelifozer/EDSD202021



There isn't anything to compare.

There are no commits from gonzalofichero:main. Try [switching the base](#) for your comparison.

Show options.

Unified Split

8. Refresh forked repo

Comparing changes

Choose two branches to see what's changed or to start a new pull request. If you need to, you can also [compare across forks](#).



base repository: gonzalofichero/EDSD202021 ▾

base: main ▾



head repository: AinhoaLeger/EDSD202021 ▾

compare: main ▾

✓ **Able to merge.** These branches can be automatically merged.

Discuss and review the changes in this comparison with others. [Learn about pull requests](#)

Create pull request

1 commit

0 files changed

0 comments

1 contributor



Commits on Apr 20, 2021



Merge pull request [AinhoaLeger#5](#) from gonzalofichero/main ...

Verified

50b6e2f

Showing 0 changed files with 0 additions and 0 deletions.


Unified

Split

8. Refresh forked repo

Open a pull request

Create a new pull request by comparing changes across two branches. If you need to, you can also [compare across forks](#).

 base repository: gonzalofichero/EDSD202021


base: main

←

head repository: AinhoaLeger/EDSD202021

compare: main










✓ **Able to merge.** These branches can be automatically merged.




Merge pull request #5 from gonzalofichero/main


Write

Preview


H B I         

Adding Comments to Ex1

Attach files by dragging & dropping, selecting or pasting them. 



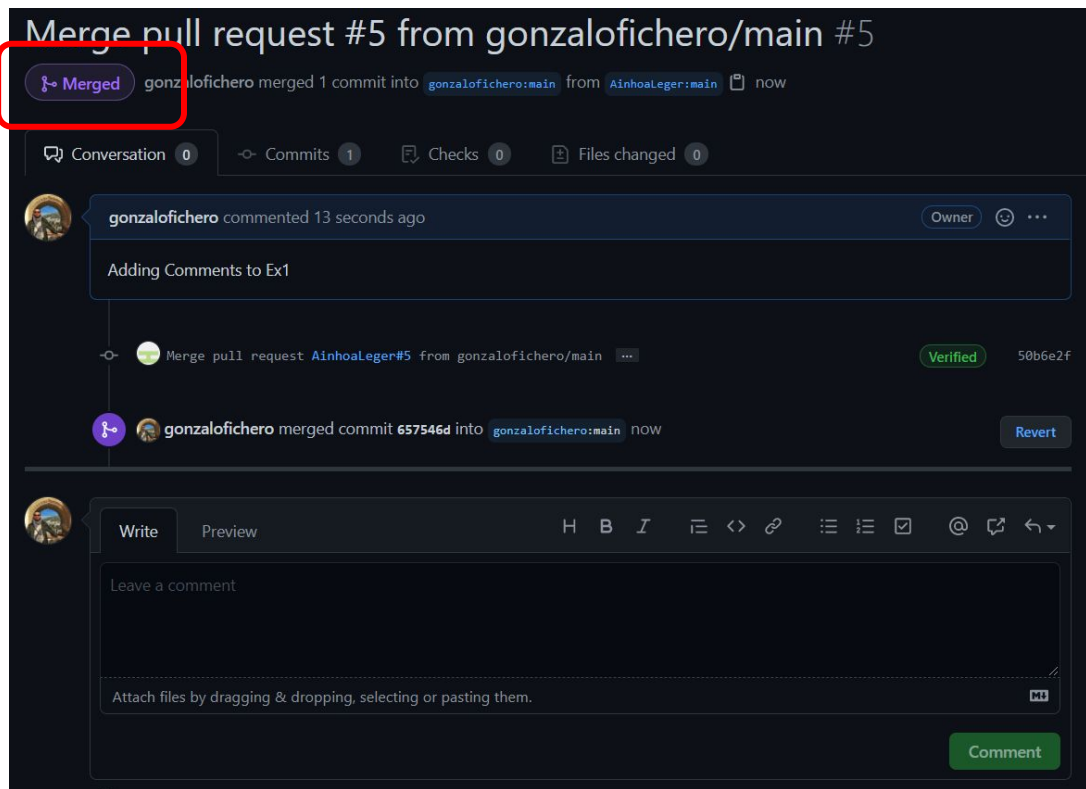
Create pull request

 Remember, contributions to this repository should follow our [GitHub Community Guidelines](#).

8. Refresh forked repo

Same steps as in Merging, until...

8. Refresh forked repo



Merge pull request #5 from gonzalofichero/main #5

Merged gonzalofichero merged 1 commit into gonzalofichero:main from AinhoaLeger:main now

Conversation 0 Commits 1 Checks 0 Files changed 0

gonzalofichero commented 13 seconds ago

Adding Comments to Ex1

Merge pull request AinhoaLeger#5 from gonzalofichero/main Verified 50b6e2f

gonzalofichero merged commit 657546d into gonzalofichero:main now Revert

Write Preview H B I ≡ <> 🔗 ☰ ☷ ✉ @ ↻ ↶

Leave a comment

Attach files by dragging & dropping, selecting or pasting them. 📎

Comment

8. Refresh forked repo

And don't forget to re-pull the repo in your RStudio!

As to basic Git, that's all you need to know

9.



9.a. Git commands

- We did many actions directly in the browser
- But we could use Git's shell for:
 - Fork repo
 - Create pull request
 - Merge changes
 - Refresh forked repo
 - more...
- Why? Faster!
- How?
 - <https://happygitwithr.com/git-commands.html>
 - <https://dzone.com/articles/top-20-git-commands-with-examples>
 - <https://git-scm.com/docs/git>

9.b. Branches

- Branch?
 - Is an independent line of development.
- Example?
 - Different regression models, but pushing just the one that fits best.
- Why?
 - More tidy, less merging problems
 - Working in parallel
- How?
 - <https://happygitwithr.com/git-branches.html>
 - <https://www.atlassian.com/git/tutorials/using-branches>

9.c Gaming with Git

- Interactive game to learn Git commands
- <https://ohmygit.org/>



HANDS ON!

Small exercise to feel comfortable

1. Groups of 2: 1 forker & 1 owner
2. Owner:
 - a. Create Git repo
 - b. Create R project linked to repo
 - c. Add R file (provided) and commit + push to repo
 - d. Wait for it...
3. Forker:
 - a. Fork repo
 - b. Pull from forked repo (should have a file in your folder now)
 - c. Create change in file
 - d. Commit + Push
 - e. Create pull request
4. Owner:
 - a. Merge changes in own repo
5. Forker:
 - a. Refresh merged repo

See you back in 20'

Q&A