Commit, push and create issue*

Tran Thai Hung

Last version: December 05, 2024

In this example, we will create a new simple script file in day1 folder of our project then commit and push it to github. After that we will create an issue related to this script and resolve the issue

Table of contents

*Please do not circulate.

List of Figures

1 Steps to commit and push to Github using Github desktop

1.1 Working on local repository

1.1.1 Create a new R script

- Open Rstudio
- In the menu, `File` -> `New File` -> `R script`

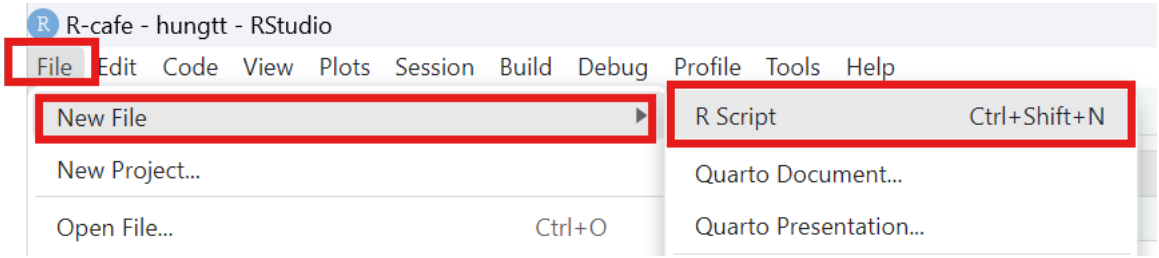

Figure 1: Create New R script

1.1.2 Write down your work in script and save.

- Type your work in `Source code` section -> click on `save` icon or `Ctrl + S`

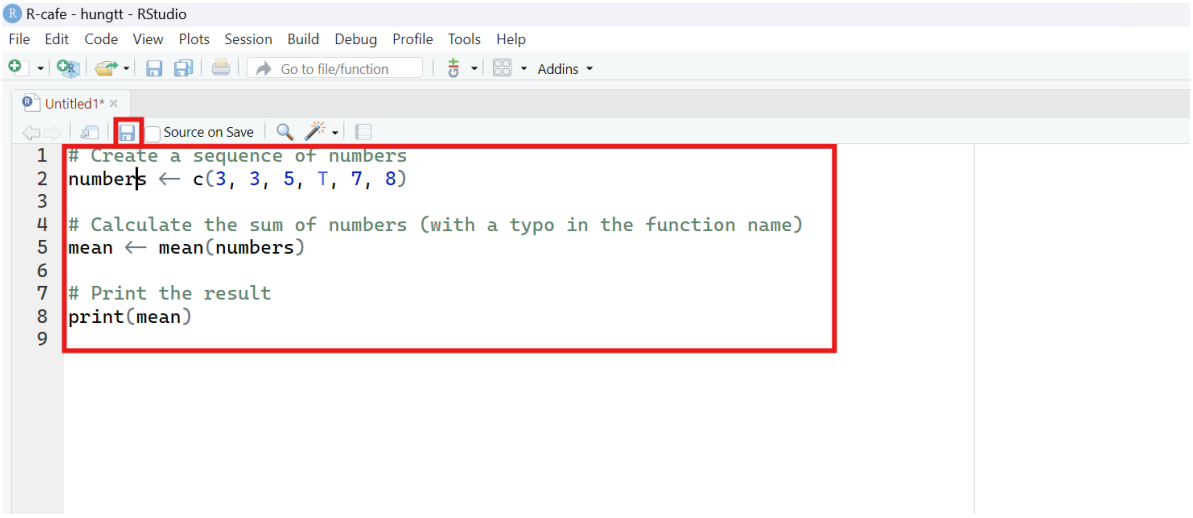

Figure 2: Write down your work

1.1.3 Save in local repository directory and name the file

- Click `Browse` and select your project directory
- Name your file in `File name` -> click `save`

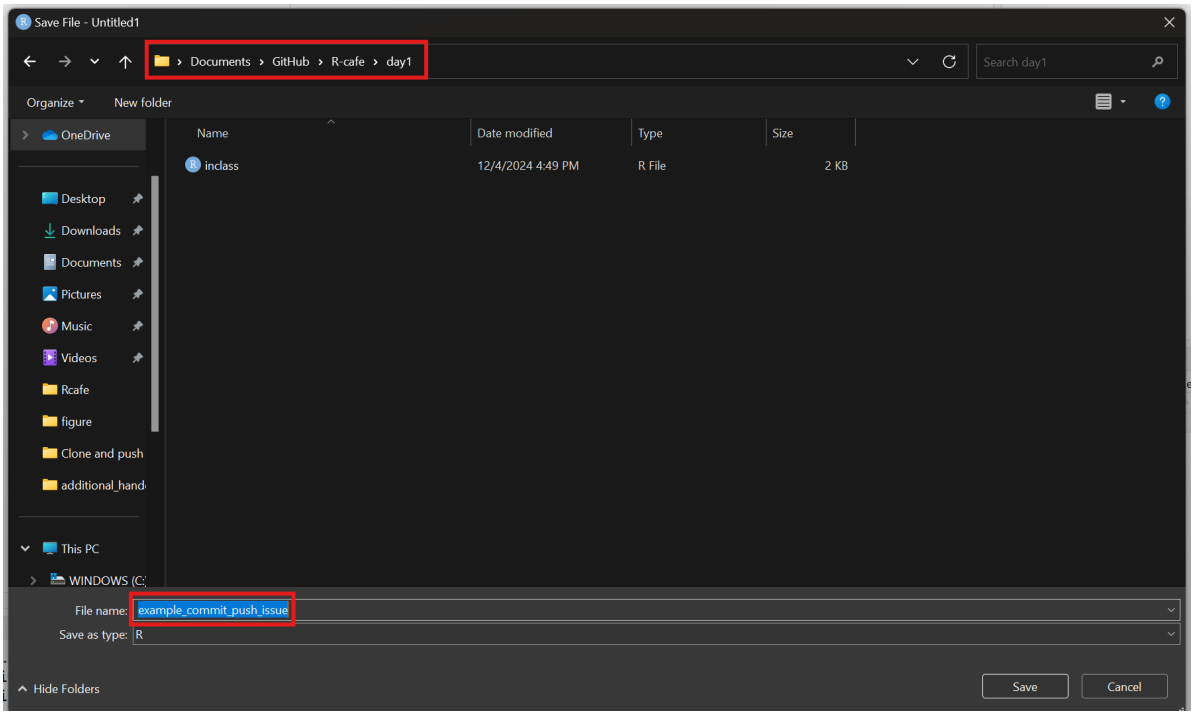

Figure 3: Select the directory

1.2 Commit and push your work

1.2.1 Commit your work

- Open Github desktop.
- Tick files which you would like to commit in the `Changes` section. You could view what changes has been made to any file by clicking on each file. The changes will appear on the right.
- Fill in the summary of your commit in the `Summary required`. Add more details in the `Description`.
- Click on `Commit to [your branch]`. Always check whether you commit to correct branch.

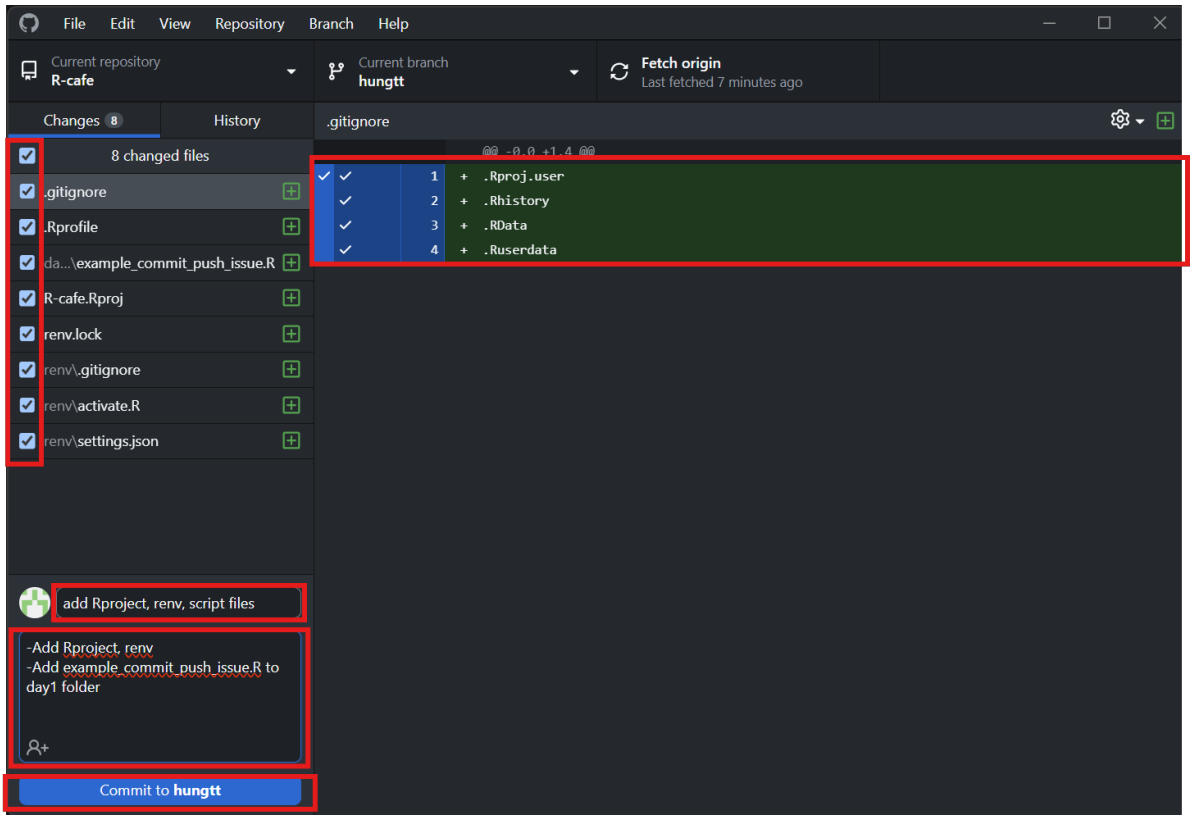

Figure 4: Select files, add summary, description

1.2.2 Push to your branch on Github

- Double check if your `Current branch` is correct. Now, click on `Push origin` to push to your branch.

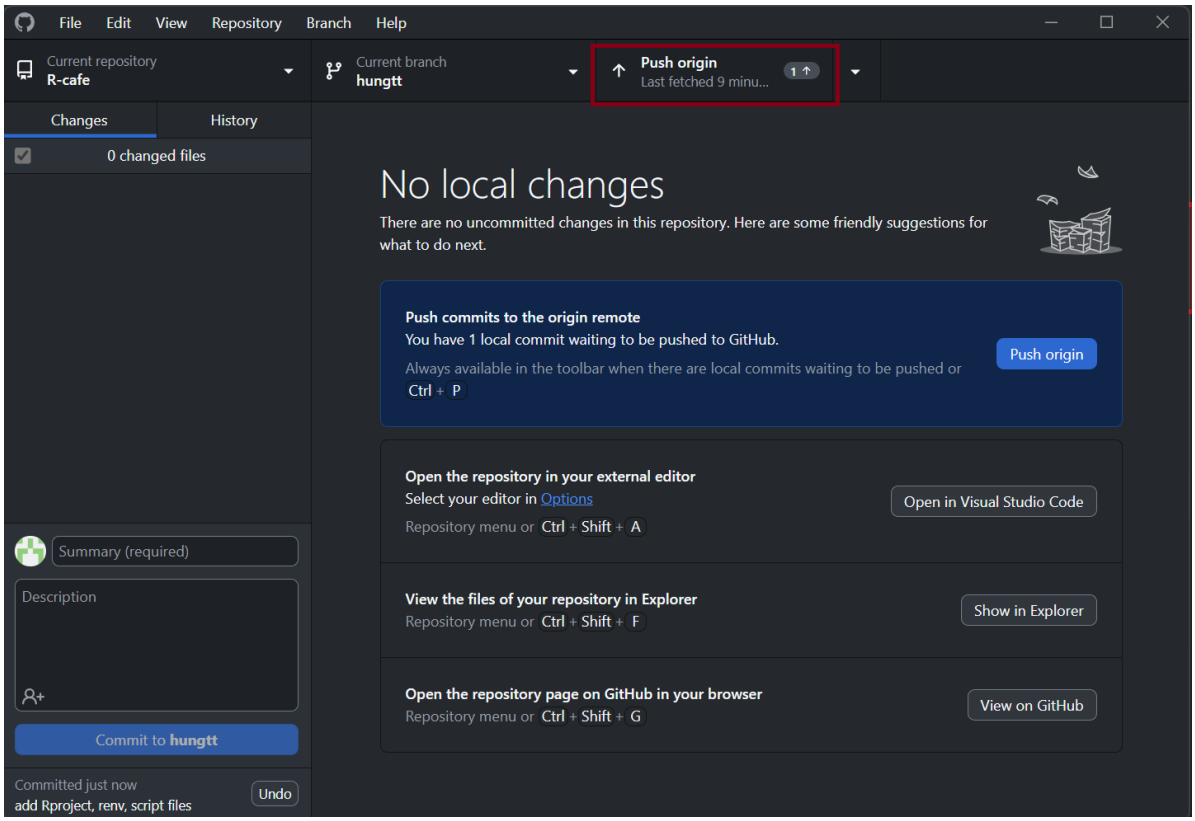

Figure 5: Push to your branch

1.2.3 Check your changes on your Github branch

1. Select your branch on Github
2. You will see a notification of your push from local repository
3. Show you how many commit you ahead of or behind main
4. You can compare your branch with other branch or main and make pull request. You could ignore this for now.
5. Click on day1 to view our example_commit_push_issue.R script

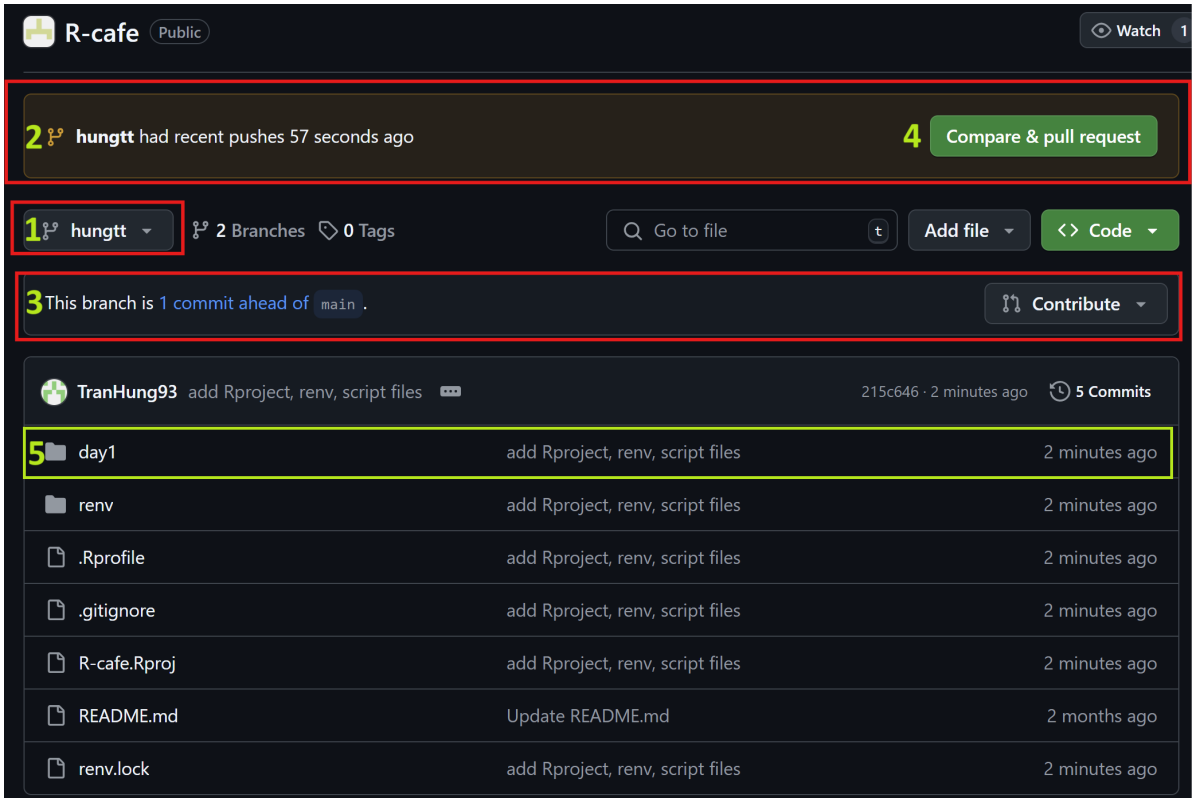

Figure 6: View our push on Github branch

6. View our example_commit_push_issue.R

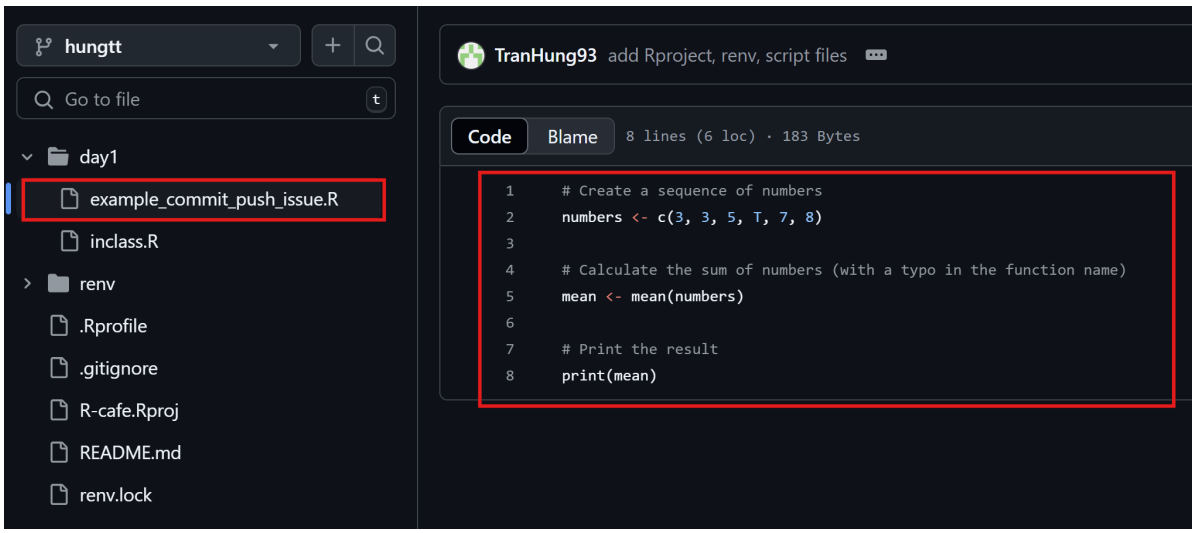

Figure 7: View our script on Github branch

2 Steps to create an issue and resolve issues on Github

2.1 Create an issue on Github

2.1.1 Select the reference in new issue

- Click on the number of the line in the code
- Click `...` -> select `Reference in new issue`

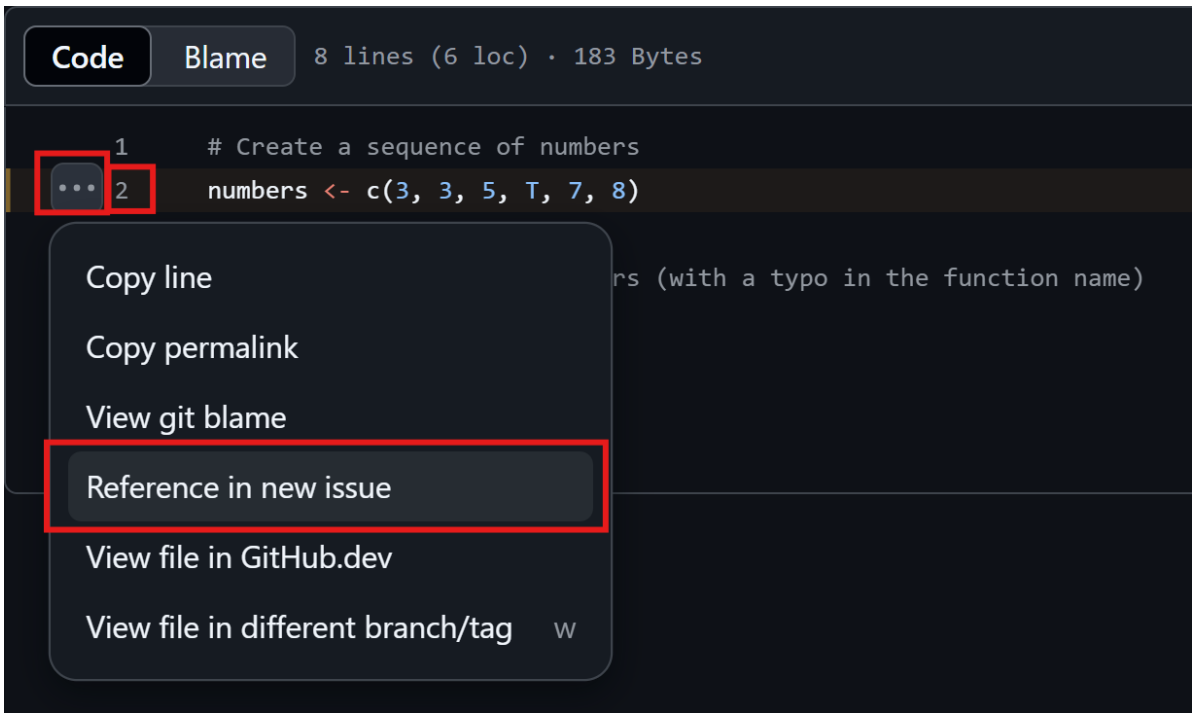

Figure 8: Select the place where you want to refer to

2.1.2 Put more information to your issue and submit issue

- Fill in the title in `Add a title`
- The link to the error place will automatically appear in `Add a description`. You might write more detail about the errors in this section as below
- Select the person you would like assign this issue by select `Assignees` or quickly assign it to yourself by `assign yourself`
- Classify the issue by `Labels`
- Click `Submit new issue`

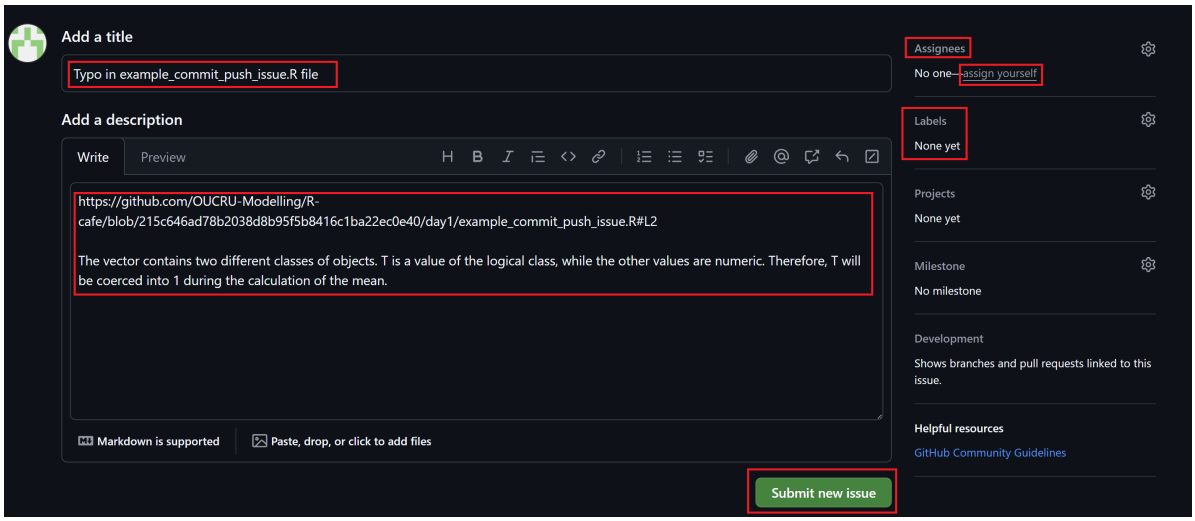

Figure 9: Add information and submit issue

2.1.3 View the open issues

- Select `Issues` -> `Open`. You might see the one who is responsible for this issue in `Assignee` tab.
- Select the issue you would like to view `Typo in example_push_commit_issue.R file`.

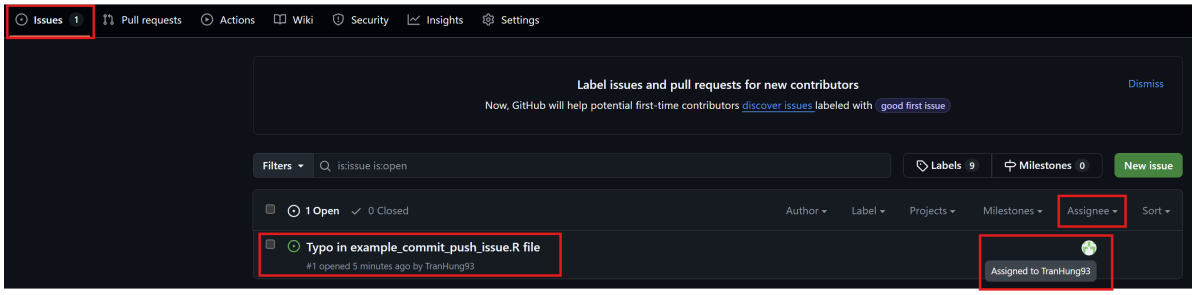

Figure 10: View the open issue

2.1.4 View the detail of an issue

- You might see the title of the issue, the reference links to the errors, and the detail of the issue.
- Navigate the place of `error` by clicking the reference link

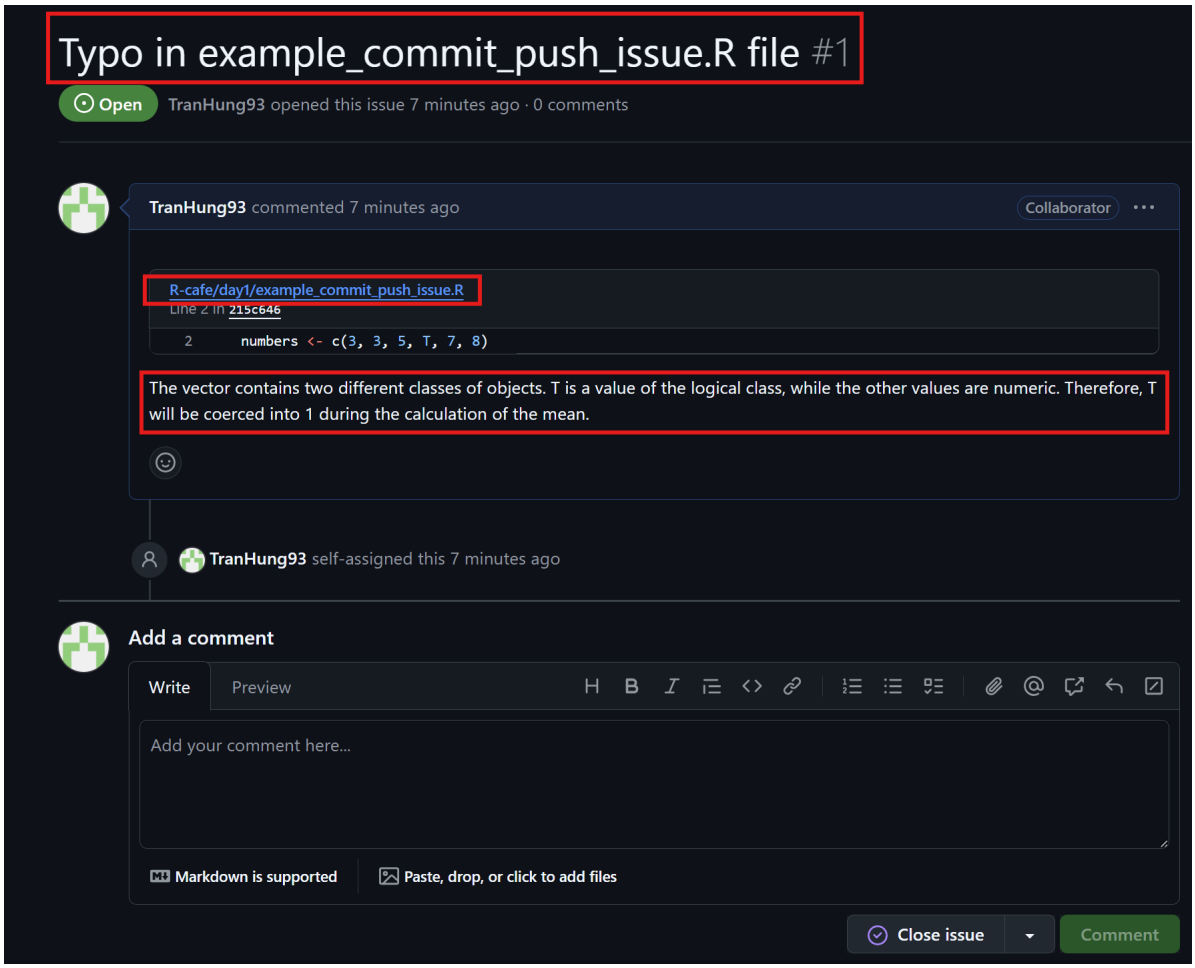

Figure 11: View the details of issue

2.2 Resolve the issue

2.2.1 Clone the branch you would like to fix the issue

- Refer to Steps to Clone a Repository on Github in Github note 1

2.2.2 Fix the code locally

- Open the example_push_commit_issue.R file
- Make correction to the code and make a comment.

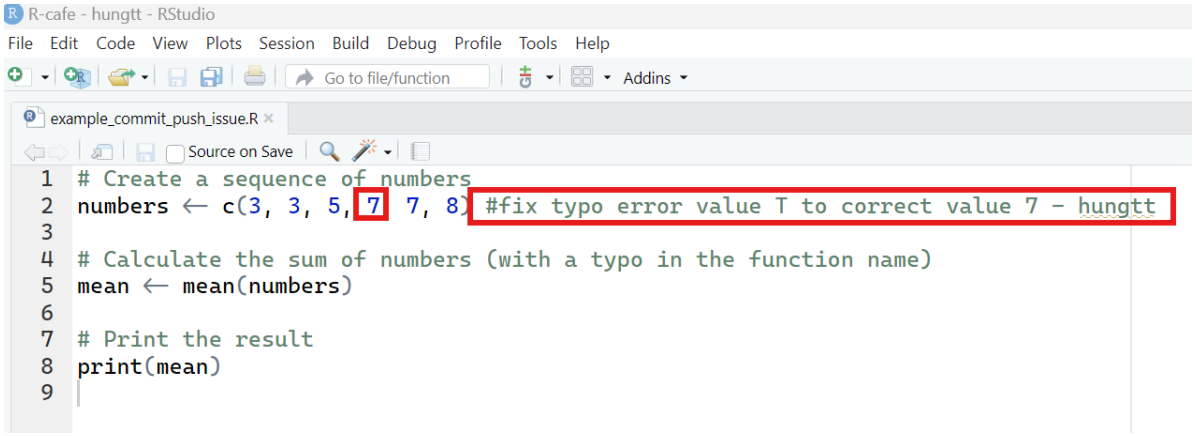

Figure 12: Resolve issue in local cloned repository

2.2.3 Commit the corrections using Github desktop

- Open the Github desktop. You will see the changes now. The red lines are old version. The green lines are current version.
- Commit and push the corrections

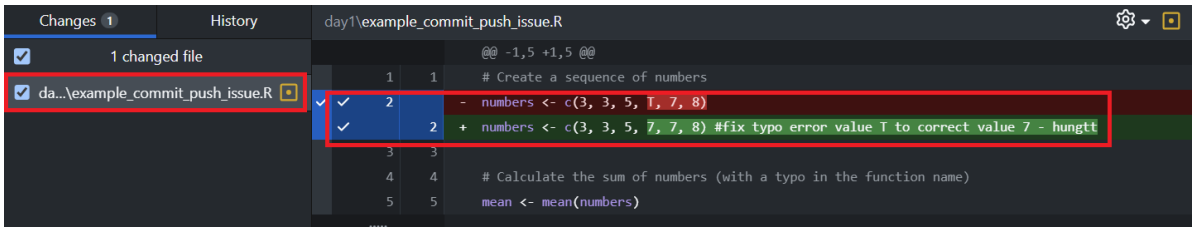

Figure 13: Commit and push the correction

2.2.4 Check the branch again

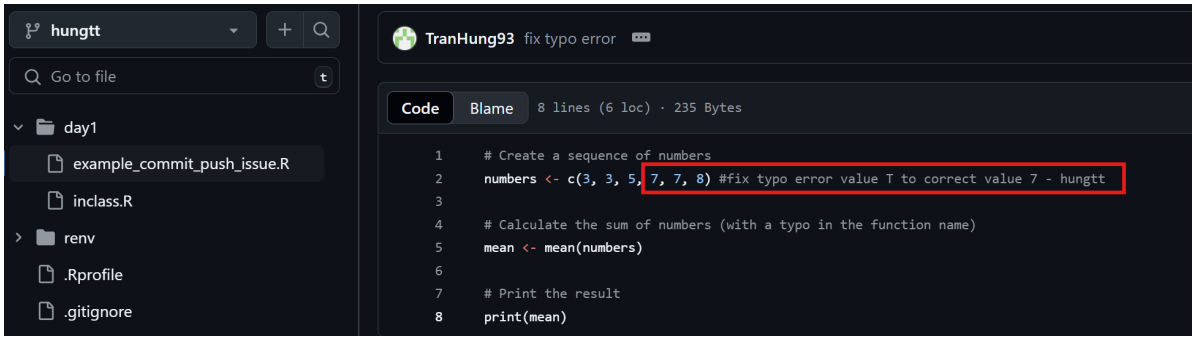

Figure 14: View the code on github

2.2.5 Comment and close the issue

- Go back to the issue. Select `Issues` -> `Open` -> `Typo in example_push_commit_issue.R file`
- Add comments in `Add a comment` -> Click on `Close with comment`. (If the issue has not been resolved, we can click `Comment` to discuss with other collaborators)

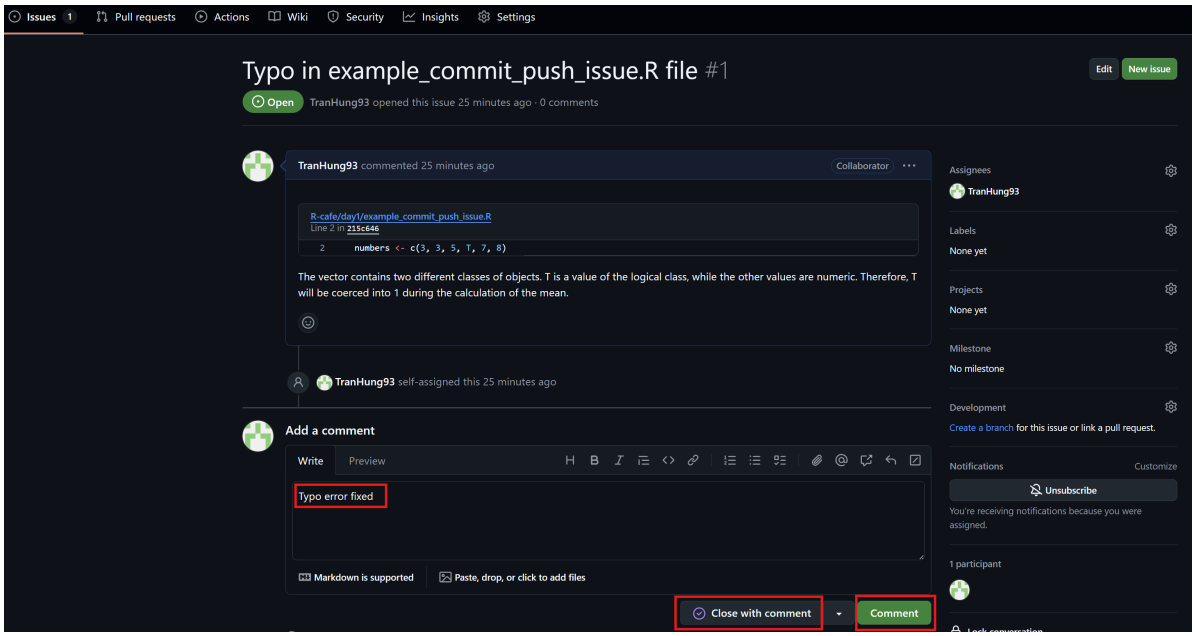

Figure 15: Close and comment issue on Github

2.2.6 View closed issue

- We can always view the old closed issue by clicking `Issues` -> `Closed`

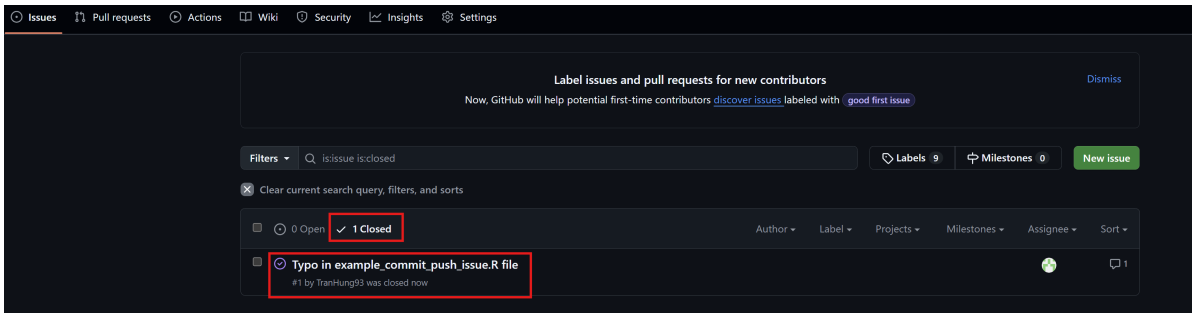

Figure 16: View the closed issues on github