How to Git with RStudio

Part II

Connect RStudio to Git and GitHub

Make a repo on GitHub

On your profile page, click on **Repositories**,

then click the big green **New** button.

Repository Template: **No template**

Repository Name: **secondrepo**

Description:

"This is a test repository for my Git/GitHub setup."

Public.

Initialize this repository with:

Add a README file.

Click the green button that says

Create repository.

Now click the green button that says

<> **Code**.

Copy a clone URL to your clipboard.

Clone the test GitHub repository via RStudio

In RStudio, start a new Project:

- *File* > *New Project* > *Version Control* > *Git*.

Paste the URL of your new GitHub repository.

Take charge of where the Project will be saved locally.

Check *Open in new session*.

Click **Create Project**.

You should find yourself in a new local RStudio Project that represents your test repo on GitHub.

This should download the `README.md` file from GitHub.

Look in RStudio's file browser pane for the `README.md` file.

Make local changes, save, commit

From RStudio, modify the `README.md` file by adding the line "This is a line from RStudio".

Save your changes.

Commit these changes to your local repo.

How?

Click the **Git** tab in upper right pane.

Check **Staged** box for `README.md`.

Click **Commit**.

Type a message in **Commit message**,

such as “Commit from RStudio”.

Click **Commit**.

Push your local changes online to GitHub

Click the green **Push** button

to send your local changes to GitHub.

Confirm the change propagated to GitHub

Go back to the browser.

You should see the new

"This is a line from RStudio" in the README.

If you click on **commits**, you should see one with the message "Commit from RStudio".

Clean up

Local

You can delete the local repo any way you like.

It's just a regular directory on your computer.

GitHub

In the browser, go to your repo's landing page on GitHub.

GitHub

Click on **Settings**.

GitHub

Scroll down,

click on **delete repository**,

and do as it asks.

Workflow: New Project

We create a new Project, with the

"GitHub first, then RStudio" sequence.

Why do we prefer this?

Because this method of copying the Project from GitHub to your computer also sets up the local Git repository for immediate pulling and pushing.

Under the hood, we are doing `git clone`.

Make a repo on GitHub

Repository name:

Approach this similar to a variable name, in code: descriptive but brief, no whitespace.

Letters, digits, -, ., or _ are allowed.

Description: Write this for humans.

New RStudio Project

In RStudio, start a new Project:

File > *New Project* > *Version Control* > *Git*.

Paste the URL of your new GitHub repository.

Be intentional about where you create this Project.

Check “Open in new session”.

Click “Create Project” to create a new directory, which will be:

- a directory or “folder” on your computer
- a Git repository, linked to a remote GitHub repository
- an RStudio Project

This should download the `README.md` file that we created on GitHub.

Look in RStudio's file browser pane for the `README.md` file.

There's a big advantage to the "GitHub first, then RStudio" workflow:

the remote GitHub repo is configured as the `origin` remote for your local repo and your local `main` branch is now tracking the `main` on GitHub.

This is a technical but important point about Git.

The practical implication is that you are now set up to push and pull.

Make local changes, save, commit

Do this every time you finish a valuable chunk of work, probably many times a day.

From RStudio, modify the `README.md` file by adding the line "This is a line from RStudio".

Save your changes.

Commit these changes to your local repo.

How?

Click the **Git** tab in upper right pane.

Check **Staged** box for `README.md`.

Click **Commit**.

Type a message in **Commit message**,

such as “Commit from RStudio”.

Click **Commit**.

Push your local changes online to GitHub

Click the green **Push** button

to send your local changes to GitHub.

Confirm the change propagated to GitHub

Go back to the browser.

You should see the new

"This is a line from RStudio" in the README.

If you click on **commits**, you should see one with the message "Commit from RStudio".

Make a change on GitHub

Click on `README.md` in the file listing on GitHub.

In the upper right corner,

click on the pencil for “Edit this file”.

Add a line to this file,

such as “Line added from GitHub.”

Edit the commit message in “Commit changes” or accept the default.

Click the big green button “Commit changes.”

Pull from GitHub

Back in RStudio:

Inspect your `README.md`.

It should NOT have the line “Line added from GitHub”.

Click the blue Pull button.

Look at `README.md` again.

You should now see the new line there.

FIN

Now just “lather, rinse, repeat”.

Do work somewhere: locally or on GitHub.

Commit it.

Push it or pull it, depending on where you did the work, but get local and remote "synced up".

Repeat.

Note that in general (and especially in future when collaborating with other developers) you will usually need to pull changes from the remote (GitHub) before pushing the local changes you have made.

For this reason, it's a good idea to try and get into the habit of pulling before you attempt to push.

Workflow: Existing Project

Make a repo on GitHub

This slide is intentionally left blank.

New RStudio Project

In RStudio, start a new Project:

File > *New Project* > *Version Control* > *Git*.

Bring your existing project over

Bring your existing project over

Copy the files that constitute your existing project into the directory for this new project.

In RStudio, consult the Git pane and the file browser.

- Are you seeing all the files?

Are they showing up in the Git pane with questions marks?

They should be appearing as new untracked files.

Stage and commit

Commit your files to this repo.

Click the **Git** tab in upper right pane.

Check **Staged** box for `README.md`.

Click **Commit**.

Type a message in **Commit message**,

such as “Init project XYZ”.

Click **Commit**.

Push your local changes to GitHub

Click the green **Push** button

to send your local changes to GitHub.

Confirm the change propagated to GitHub

Go back to the browser.

You should see the new

"This is a line from RStudio" in the README.

If you click on **commits**, you should see one with the message “Init project XYZ”.

FIN

Now just “lather, rinse, repeat”.

Do work somewhere: locally or on GitHub.

Commit it.

Push it or pull it, depending on where you did the work, but get local and remote "synced up".

Learn More

Reference

Happy Git and GitHub for the useR

by Jennifer Bryan