

Introduction to GitHub

Julia Stewart Lowndes Project Scientist, Ocean Health Index
`lowndes@nceas.ucsb.edu`

What is GitHub?

GitHub is an open-source development platform that enables easy collaboration and versioning, which means that all saved versions are archived and attributed to each user. It is possible to revert back to any previous version, which is incredibly useful to not only to document what work has been done, but how it differs from work done in the past, and who is responsible for the changes.

Similar to Dropbox, you have certain folders on your local computer that will be synched online, although you have more control about the synching. You can store, share, track changes and collaboratively edit many filetypes (including this presentation!), using any program to edit, and can use a shared to-do list (called Issues).

Why use GitHub?

So many reasons to use GitHub. Personal organization, backing up, version control, collaboration, sharing. . .

Nicely explained by Hadley Wickham in his [Git and GitHub Tutorial](#)

And also by Karl Broman in his [GitHub Tutorial](#)

Resources

Learn more about GitHub:

- ▶ **Git and GitHub** by Hadley Wickham
- ▶ **Collaboration and Time Travel: Version Control with Git, GitHub and RStudio** video tutorial by Hadley Wickham
- ▶ **Good Resources for Learning Git and GitHub** by GitHub
- ▶ **Learn Git Branching** by Peter Cottle
- ▶ **Git/GitHub Guide** by Karl Broman
- ▶ **[Git & GitHub]**(<http://htmlpreview.github.io/?https://github.com/eco-data-science/ds-git/blob/gh-pages/index.html>) by Ben Best
- ▶ **Hello World GitHub Guide**, a 10-minute tutorial by GitHub

Just Google 'GitHub Tutorial...'

Outline

1. GitHub Structure
2. GitHub Vocabulary
3. Best Practices
4. Cloning and Synching Options
5. Cloning Examples
6. Synching Examples
7. Getting Started
8. Resources

GitHub Structure

GitHub stores files in **repositories**, owned by an **organization**.

Repositories ('repos') are essentially folders containing scripts and small text files. Repositories are version controlled so that any modifications to files, additions or deletions, are tracked and attributed to contributors with the correct permissions.

- ▶ eco-data-science is an **organization**
- ▶ github-intro is a **repository**

Let's navigate through a github repo: eco-data-science/github-intro

GitHub Vocabulary

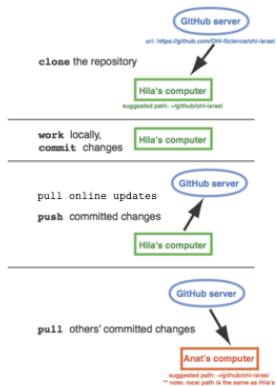
All collaborators will work locally on their own computers, syncing their changes back online so others work from the most current version.

- ▶ **clone**: download to your computer from online version with syncing capabilities enabled
- ▶ **commit**: message associated with your changes (see some best practices)
- ▶ **pull**: sync a repo on your computer with online version
- ▶ **push**: sync the online repo with your version, only possible after committing

sync = pull + commit + push

GitHub Vocabulary

sync = pull + commit + push



Best Practices

Pull often!

Commit frequently

Be mindful of filepaths

We all store GitHub repos in our home directory in a folder called ***'github'*** (all lowercase!), so that you can access it with the filepath `~/github`.

- ▶ **Windows:** `Users\[User]\Documents\github\`
- ▶ **Mac:** `Users/[User]/github/`

Please make a `github` folder in your home directory.

Cloning and Synching Options

You will clone a repository to your computer so you can work offline.

When you work on your computer, any edits you make to any files in your repo, using any program, will be tracked by GitHub. You can then commit and sync your changes back to GitHub. There are many options you can use to first clone and then sync your edits on a repo with the online version:

- ▶ **GitHub App** for Mac and for Windows
- ▶ **RStudio**
- ▶ **shell (Terminal on Mac)**

We will just use RStudio today.

Workflow using RStudio

Demo:

1. **clone a repo** from github.com, then RStudio
2. **edit within the repo**: test_script_ohi-uswest.Rmd in RStudio, then create a file in Excel
3. **commit**
4. **pull**
5. **push**

Let's practice

If you haven't already (you can refer to the OHI Manual):

1. Create a GitHub account
2. Download and install *git* software, set up your git identity
3. Email your GitHub username to `lowndes@nceas.ucsb.edu`
4. Create a folder called ***github*** on your computer
5. Clone github-intro repository to your computer

Let's practice

Now:

1. Pull
2. Edit `test_script_ohi-uswest.Rmd` in RStudio
3. Commit changes
4. Pull
5. Push
6. Repeat!