Using Github

Prerequisites

- Create GitHub account.

- Install GitHub Desktop (optional, because terminal is powerful).

WHEN YOU ARE WORKING ALONE

Step 1. Create repository

- Repository: contains all the files of your projects, including revision history.

- Repository can be created

  - On GitHub website.

  - On GitHub Desktop.

    - Add or Create

- A README.md file can be created together with repository.

Step 2. Checkout the repository

- On GitHub desktop, select the repository you wanna checkout (on Clone option)
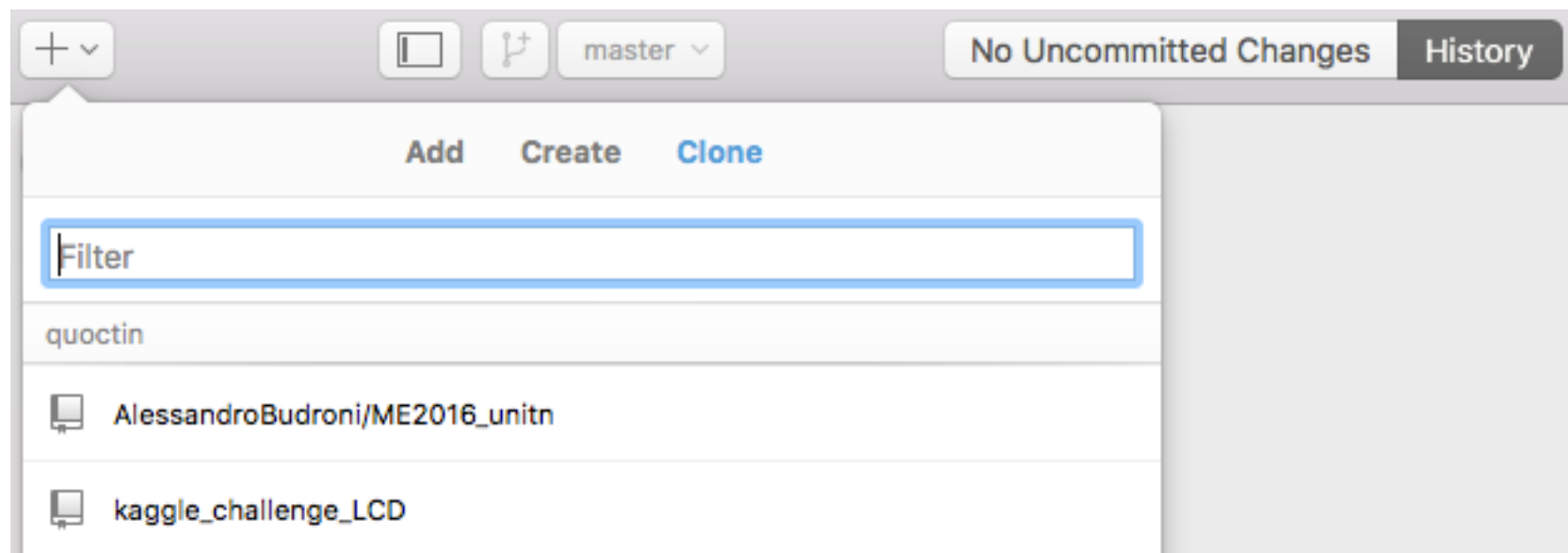
Step 3. Create a Branch

- Branching is a way to work on different versions of a repository.

- By default, it is "master". A branch is a copy or snapshot of master, at the current time.

- A collaborator usually works on his branch.
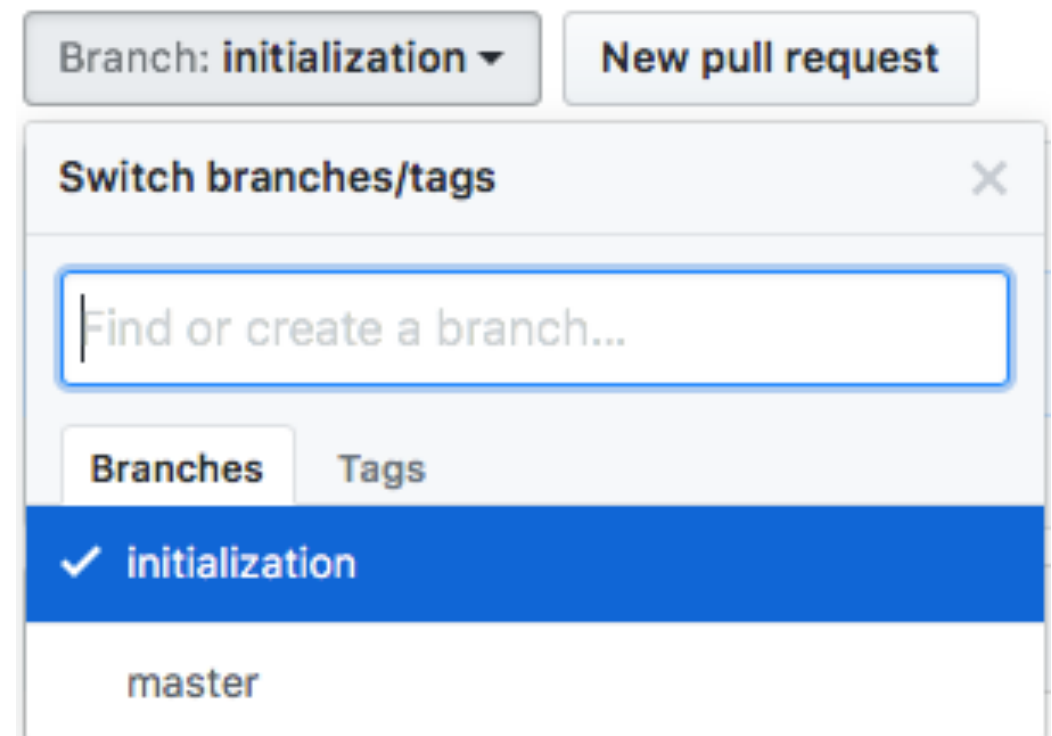
initialization ▾

22 Uncommitted Changes | History

Pull Request

Update from master | View Branch | Sync

master ▾

initialization

☑ **22 Changes**   data/normalized_data/0015ceb851d7251b8f399e39779d1e7d.hdf5

☑ data/normalize...03433ff1e.hdf5

☑ data/normalize...e71799a80.hdf5

☑ data/normalize...884ebe297.hdf5

☑ data/normalize...2012c35df.hdf5

☑ data/normalize...28275e4c.hdf5

☑ data/normalize...6e4cc535.hdf5

☑ data/normalize...86b28c3f2.hdf5

☑ data/stage1_labels.csv   •••••

☑ src/data_loading.py   •••••

Summary

Description

Commit to initialization

Step 3: Make and Commit changes

- Commit means saving changes. Each commit must has an associated commit message.

- It is important to be aware:

  - Which branch you are working on?

  - Which changes you want to commit?

quoctin/kaggle_challenge_LCD

initialization | 22 Uncommitted Changes | History | Pull Request

Update from master | View Branch | Sync

master ▼
initialization

22 Changes — data/normalized_data/0015ceb851d7251b8f399e39779d1e7d.hdf5

- [ ] data/normalize...03433ff1e.hdf5
- [ ] data/normalize...e71799a80.hdf5
- [ ] data/normalize...884ebe297.hdf5
- [ ] data/normalize...2012c35df.hdf5
- [ ] data/normalize...28275e4c.hdf5
- [ ] data/normalize...6e4cc535.hdf5
- [ ] data/normalize...86b28c3f2.hdf5
- [x] data/stage1_labels.csv
- [x] src/data_loading.py

Summary

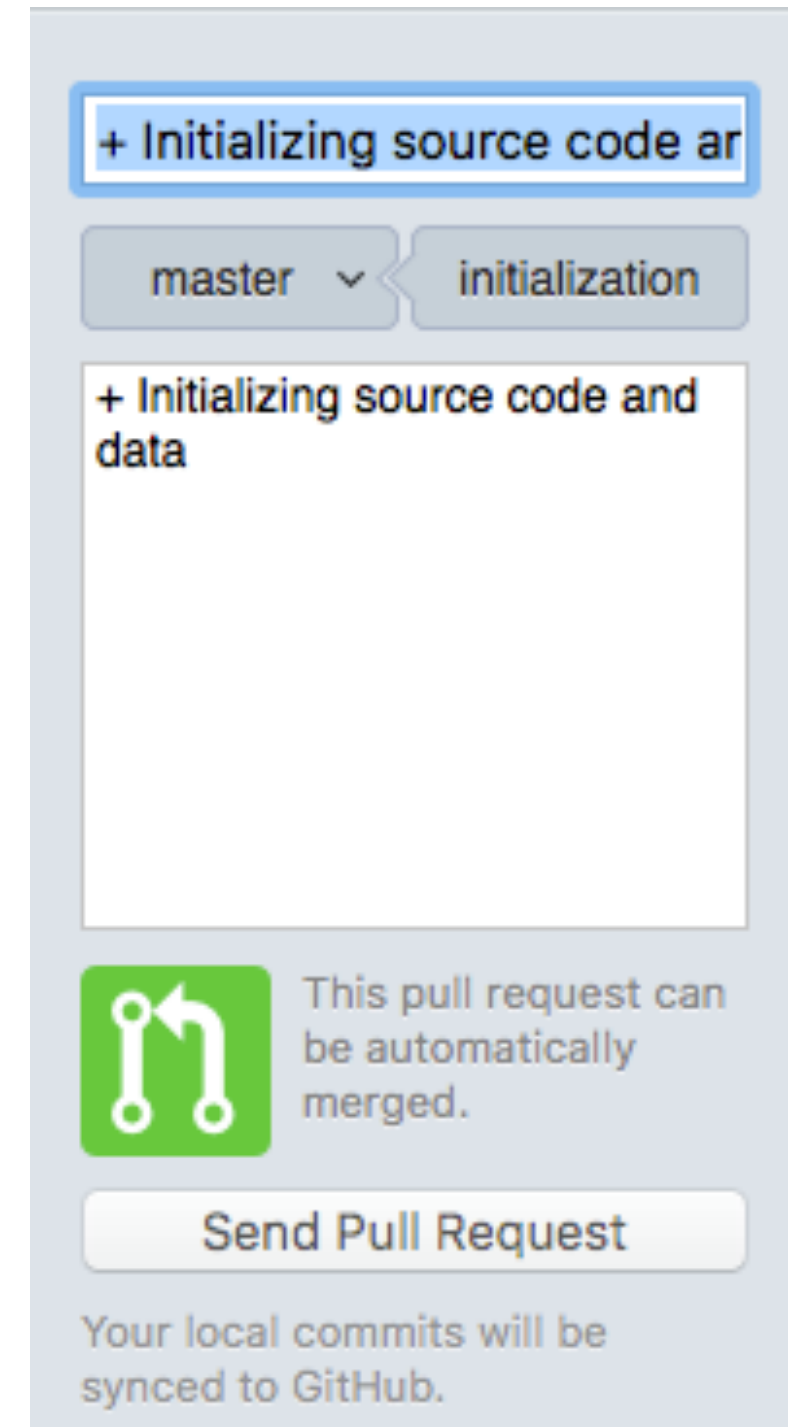+ Initializing source code and data

Commit to initialization

What you gonna save

You are saving changes to…

Step 3: Make and Commit changes

- After saving changes on a branch:

  - You can 'undo' what you have saved

  - Merge those changes to 'master' by 'Pull Request'

Step 4. Review and Merge

- You may review your pull request, comment, and decide to merge to 'master'.

- Don't be so panic, you can revert after merging.

- After merging, you can select to continue working on the branch, or delete it.

+ Initializing source code and data #1

⑂ Open **quoctin** wants to merge 1 commit into `master` from `initialization`

💬 Conversation 0   ⌥ Commits 1   📄 Files changed 2

quoctin commented 21 seconds ago   Owner   +😊   ✏️

- Initializing source code and data

     + Initializing source code and data ⋯   d3faf3e

Add more commits by pushing to the **initialization** branch on **quoctin/kaggle_challenge_LCD**.

✓ **This branch has no conflicts with the base branch**
Merging can be performed automatically.

Merge pull request ▾   You can also open this in GitHub Desktop or view command line instructions.

Write | Preview   AA▾ B *i*   66 <> 🔗   ☰ ☵ ☑   ↩▾ @ 🔖

Leave a comment

Attach files by dragging & dropping or selecting them.

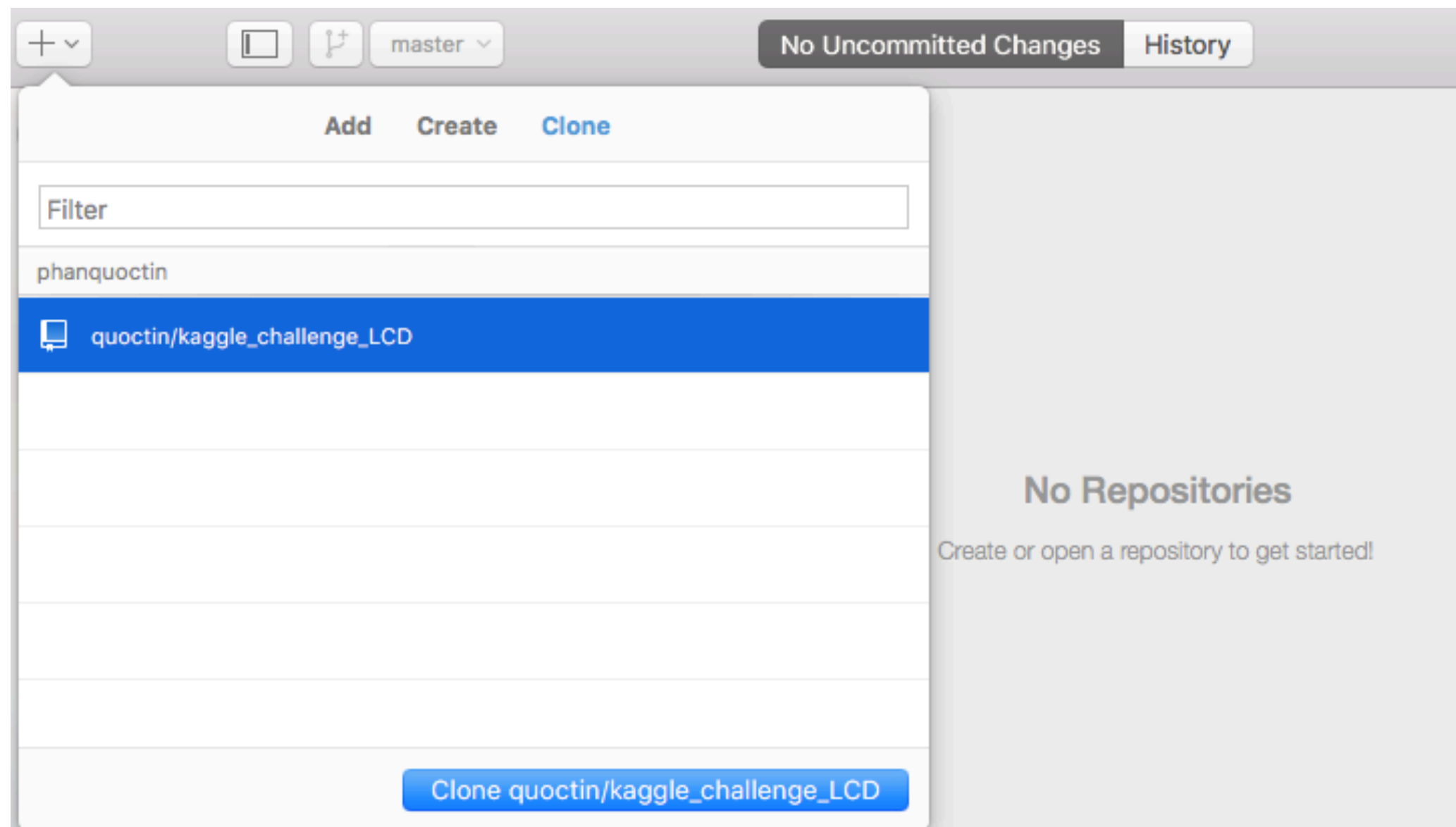# WHEN YOU ARE WORKING IN GROUP

As an owner

- Add collaborators

  - Under Repository, select Settings -> Collaborators

- A collaborator can have access to the repository and perform many actions on that.
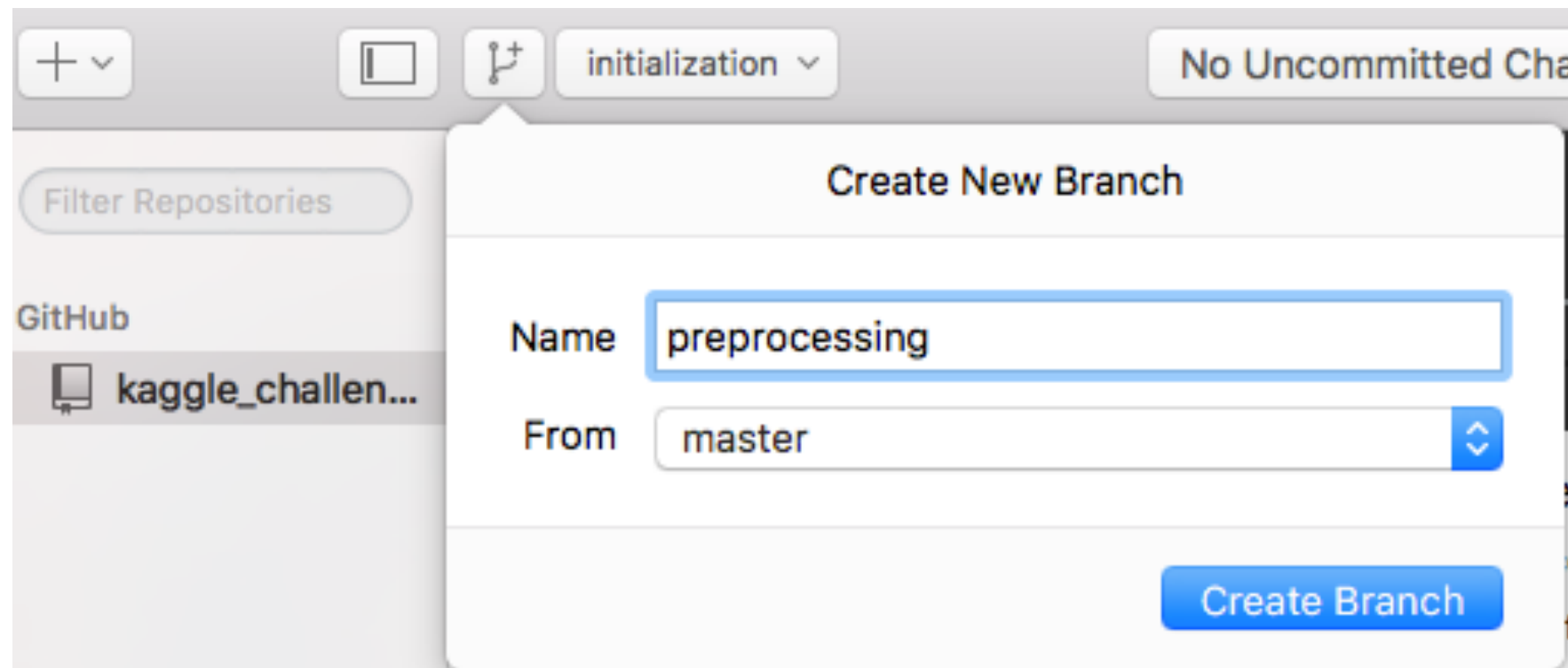
  - More details can be found here

As a collaborator

- Accept email invitation.

- Clone the project to a specific folder.

As a collaborator

- Create a new branch to work on it.

- Follow the procedure as mentioned in previous slides (if create by GitHub Desktop, remember to publish it).

As a collaborator

- After changes in a branch are merged to 'master', you can update those changes to your branch.

- You should compare your branch and other branches frequently.

.gitignore

Annoying?

.gitignore

- Identify files that GitHub Desktop should ignore.

- Each line specifies a pattern. More details can be found <u>here</u>.

- For example, to ignore *.hdf5 in directory data/normalized_data: data/normalized_data/*.hdf5

Markdown (.md)

- To style your text on the web

  - HTML, CSS: so complicated. Developers are making the world better and easier.

  - [Markdown](#) is an lightweight and easy-to-use syntax to describe your project on GitHub platform.

  ```
  It's very easy to make some words **bold** and other words
  *italic* with Markdown. You can even [link to Google!]
  (http://google.com)
  ```

  It's very easy to make some words **bold** and other words *italic* with Markdown. You can even link to Google!

Markdown (.md)

- Markdown can help to highlight programming languages.

- GitHub also support Markdown in user comments.