

# Reproducible research workflows for psychologists

Git & GitHub

Johannes Breuer & Frederik Aust

KU Leuven, 27.-28.04.2022

# Git

## Command line tool

### 1. Version control

- Track changes to *text files*
- Retain version history of snapshot (*commits*)
- Track provenance (who, what, when)



Current Repository: moby-git | Current Branch: main | Fetch origin: Last fetched 24 minutes ago

Changes | History

Select Branch to Compare...

- added separator to main README top section (mcdwayne · May 11, 2021)
- update README on main for readability (mcdwayne · May 11, 2021)
- added repo directions in README (mcdwayne · May 10, 2021)
- Merge pull request #8 from Axosoft/sarabreeding-patch-1 (Dwayne McDaniel · May 7, 2021)**
- Update README.md (Sara (Breeding) Stamas · May 7, 2021)
- edit to next steps main branch (mcdwayne · May 7, 2021)
- updated next steps (mcdwayne · May 7, 2021)
- README fix (mcdwayne · May 6, 2021)
- added next steps to go to git-commit in main README (mcdwayne · May 6, 2021)
- modified README with better markdown for long links (mcdwayne · May 5, 2021)
- Merge pull request #7 from mcdwayne/main (Dwayne McDaniel · May 3, 2021)
- changed : to . line 24 README (mcdwayne · May 3, 2021)
- moved gitkraken dl icon (mcdwayne · May 3, 2021)
- updated link for Git Clone page (mcdwayne · May 3, 2021)
- swapped GitKraken logo for just Keif

### Merge pull request #8 from Axosoft/sarabreeding-patch-1

Dwayne McDaniel · d40cec6 · ± 1 changed file

Update README.md for clarity and fixed description of a field on GK

| File      | Line | Diff   |
|-----------|------|--|
| README.md | 81   | 81   |
| README.md | 82   | 82   |
| README.md | 83   | 83   |
| README.md | 84   | -To continue on to practice making commits you can switch to the 'git-commit' branch.  |
| README.md | 84   | +Keep going! To practice making commits you can switch to the 'git-commit' branch.   |
| README.md | 85   | 85   |
| README.md | 86   | 86   |
| README.md | 87   | 87   |
| README.md | 88   | 88   |
| README.md | 89   | 89   |
| README.md | 90   | -To checkout the branch locally using GitKraken, simply click on the 'git-commit' branch in the Remote menu on the <b>right</b> hand side.                             |
| README.md | 90   | +To checkout the branch locally using GitKraken, simply click on the 'git-commit' branch in the Remote menu on the <b>left</b> hand side.                              |
| README.md | 91   | 91   |
| README.md | 92   | 92   |
| README.md | 93   | 93   |
| README.md |      | ### Working with remotes in GitKraken  |
| README.md |      | @@ -124,4 +124,4 @@ Based on the [Project Gutenberg](http://www.gutenberg.org/ebooks/2701) [Plain Te   |
| README.md | 124  | 124  |
| README.md | 125  | 125  |
| README.md | 126  | 126  |
| README.md | 127  | -[2]: https://www.gitkraken.com/download?utm_source=learn%20git%20practice%20repo&utm_medium=README%20gk%20download%20link&utm_campaign=git%20clone%20practice%20repo  |
| README.md | 127  | + [2]: https://www.gitkraken.com/download?utm_source=learn%20git%20practice%20repo&utm_medium=README%20gk%20download%20link&utm_campaign=git%20clone%20practice%20repo |

# Git

## Command line tool

### 1. Version control

- Track changes to *text files*
- Retain version history of snapshot (*commits*)
- Track provenance (who, what, when)

### 1. Branching

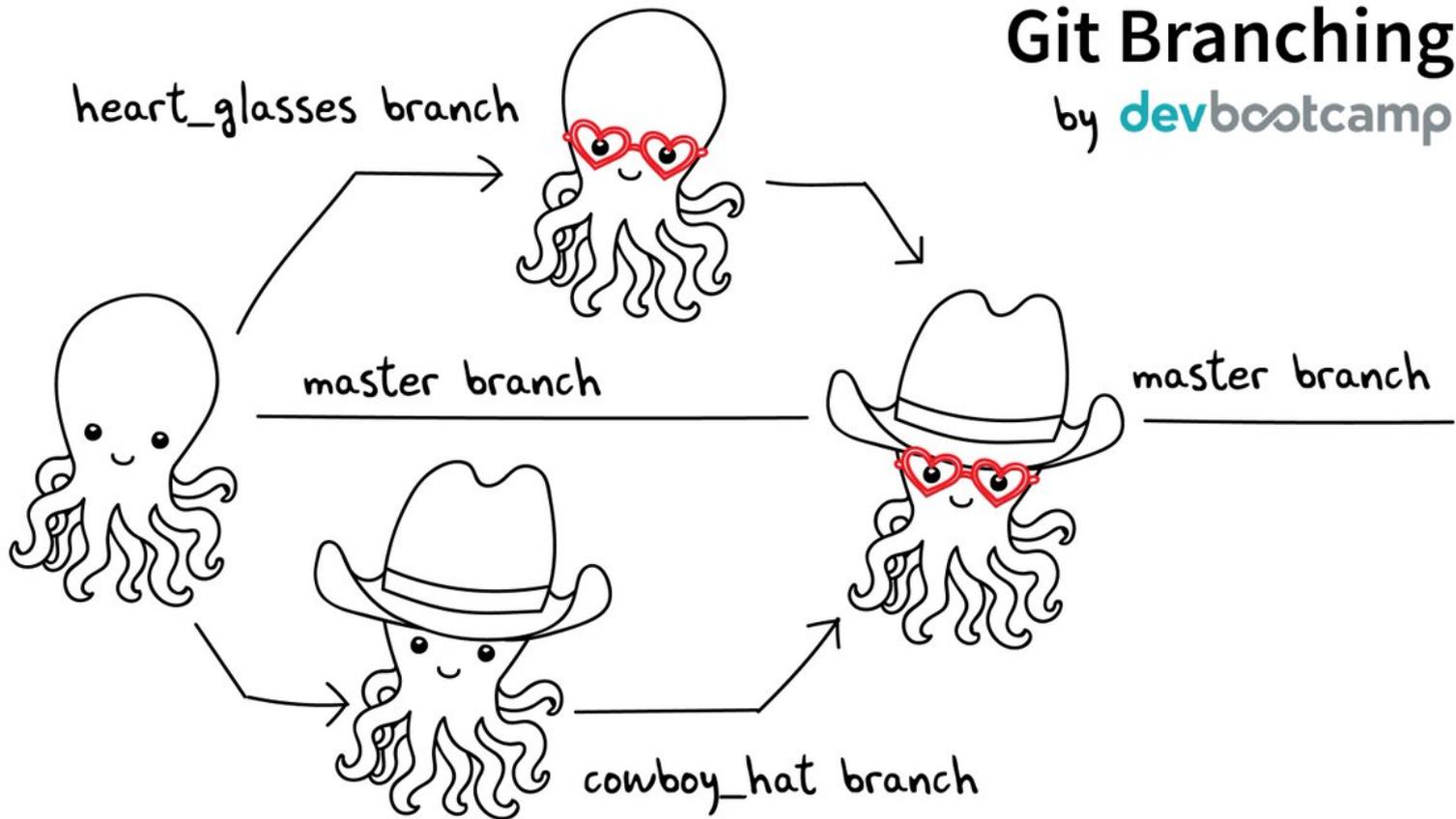
- Alternative versions of files
- Simultaneous asynchronous collaboration

(Vuorre & Curley, 2018)

---

# Git

## Git Branching by [devbootcamp](#)



# Git

## Why use Git?

- Avoid mistakes, e.g.
    - Working on wrong version
    - Inadvertently deleting or editing files
  - Track and merge changes of multiple collaborators across multiple files
  - (Offsite backups)
-

# In case of fire

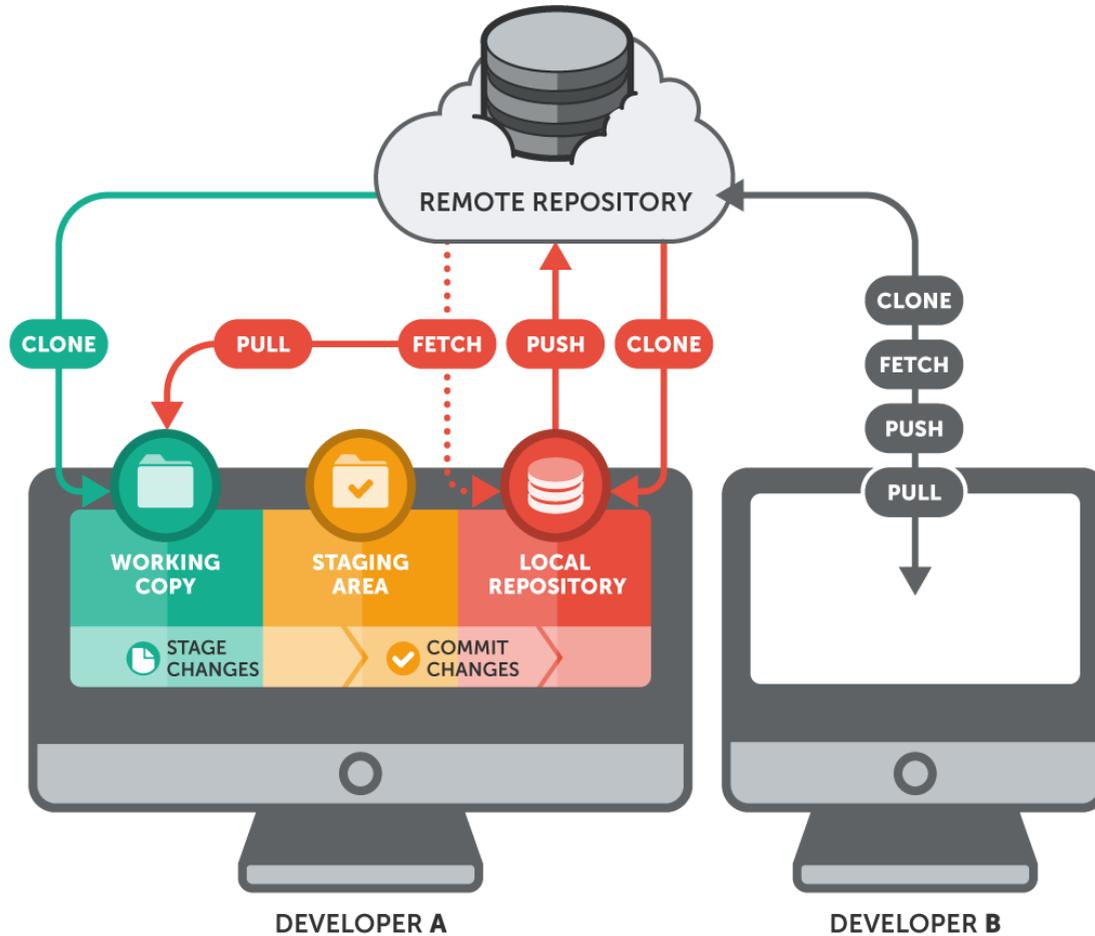


 1. `git commit`

 2. `git push`

 3. `leave building`

# Git

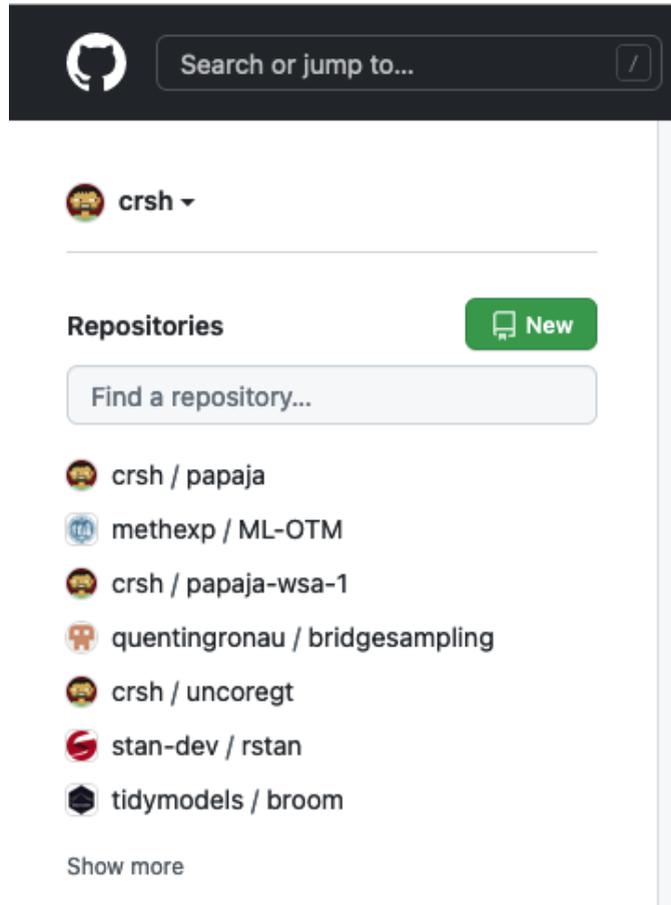


# GitHub



- Proprietary platform
  - Free to use for academics
- Host remote Git repositories
- Provides slick web interface
- Offers project management tools
- Repositories can receive a DOI with [Zenodo](#)
- Servers located in the USA (GDPR)

# GitHub



The screenshot shows the GitHub search interface for the user 'crsh'. At the top, there is a search bar with the placeholder text 'Search or jump to...' and a search icon. Below the search bar, the user's profile is shown with the name 'crsh' and a dropdown arrow. Underneath, the 'Repositories' section is displayed, featuring a 'New' button with a repository icon. A search input field with the placeholder 'Find a repository...' is positioned below the button. A list of repositories follows, each with a repository icon, the user's profile icon, and the repository name: 'crsh / papaja', 'methexp / ML-OTM', 'crsh / papaja-wsa-1', 'quentingronau / bridgesampling', 'crsh / uncoregt', 'stan-dev / rstan', and 'tidymodels / broom'. At the bottom of the list, there is a 'Show more' link.

 Search or jump to... 

 crsh ▾

**Repositories**  **New**

Find a repository...

-  crsh / papaja
-  methexp / ML-OTM
-  crsh / papaja-wsa-1
-  quentingronau / bridgesampling
-  crsh / uncoregt
-  stan-dev / rstan
-  tidymodels / broom

Show more

## Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository](#).

Owner \*



crsh ▾

Repository name \*

my-first-repository



Great repository names are short and memorable. Need inspiration? How about [musical-succotash](#)?

Description (optional)

This is my first repository. Yey!



**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](#).

This will set  `main` as the default branch. Change the default name in your [settings](#).

Create repository

# GitLab



- Open source platform
  - Can be hosted by your institution
- Host remote Git repositories
- Provides slick web interface
- Offers project management tools
- Servers can be located in your country (GDPR)

# GitLab



The image shows the top navigation bar of the GitLab interface. It features a dark blue header with a search bar labeled "Search GitLab" and several utility icons. A dropdown menu is open from the "+" icon, listing "New project/repository", "New group", and "New snippet". A "New project" button is visible to the right of the dropdown. Below the dropdown, there is a "Filter by name..." input field and a "Name" dropdown menu.

Exp

New...

New project/repository

New group

New snippet

Filter by name...

Name

New project

**Project name**

my-first-repository

**Project URL**

https://gitlab.com/crsh/

**Project slug**

my-first-repository

Want to house several dependent projects under the same namespace? [Create a group.](#)

**Project description (optional)**

This is my first repository. Yey!

**Project deployment target (optional)**

Select the deployment target

**Visibility Level** [?](#)

-  Private  
Project access must be granted explicitly to each user. If this project is part of a group, access will be granted to members of the group.
-  Public  
The project can be accessed without any authentication.

**Project Configuration**

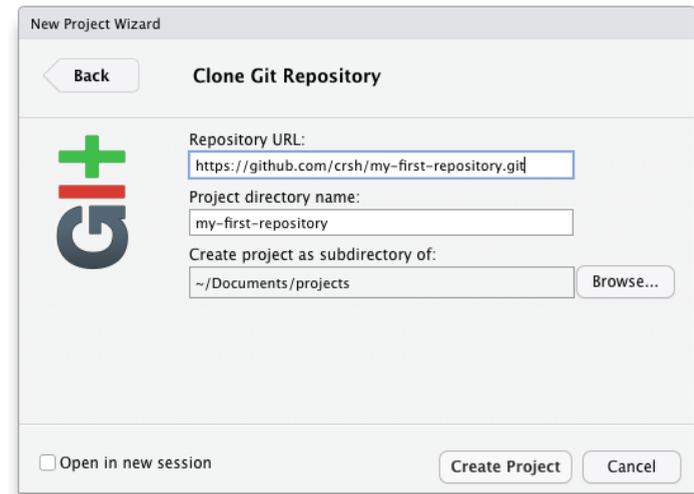
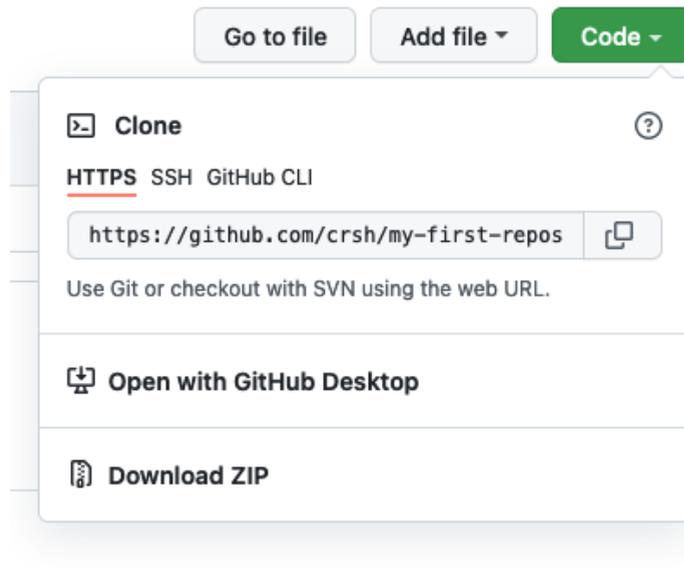
- Initialize repository with a README  
Allows you to immediately clone this project's repository. Skip this if you plan to push up an existing repository.
- Enable Static Application Security Testing (SAST)  
Analyze your source code for known security vulnerabilities. [Learn more.](#)

Create project Cancel



## Clone your GitHub repository to your computer

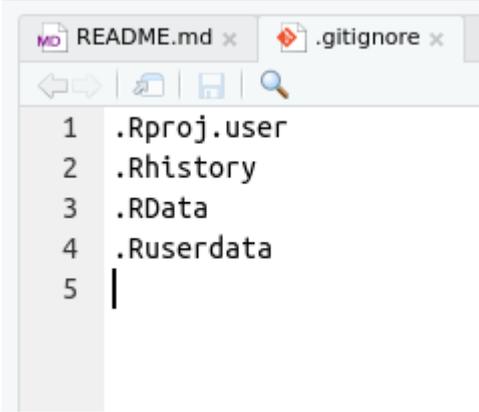
```
git clone git@github.com:crsh/my-first-repository.git`
```



# Git

## .gitignore

- Files to ignore (do not track)
- Each line specifies a **pattern**
- I typically add R Markdown cache files (e.g., `.rdb`, `.rdx`)
  - `usethis:::git_ignore_lines`



```
1 .Rproj.user
2 .Rhistory
3 .RData
4 .Ruserdata
5 |
```

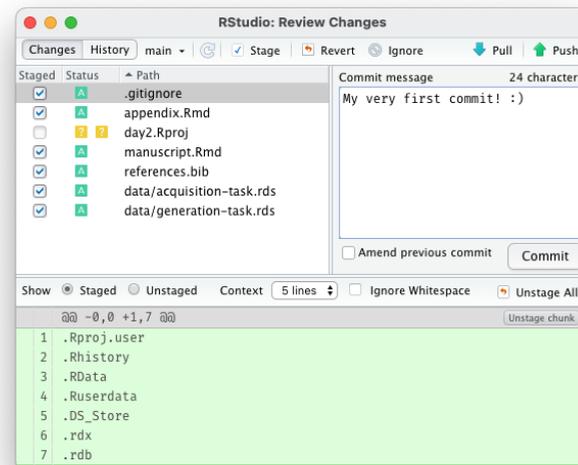
The screenshot shows a code editor window with two tabs: 'README.md' and '.gitignore'. The '.gitignore' tab is active, displaying a list of five lines of text. The first four lines are: '.Rproj.user', '.Rhistory', '.RData', and '.Ruserdata'. The fifth line is a vertical bar '|'. The editor has a standard toolbar with back, forward, home, end, and search icons.

# Git

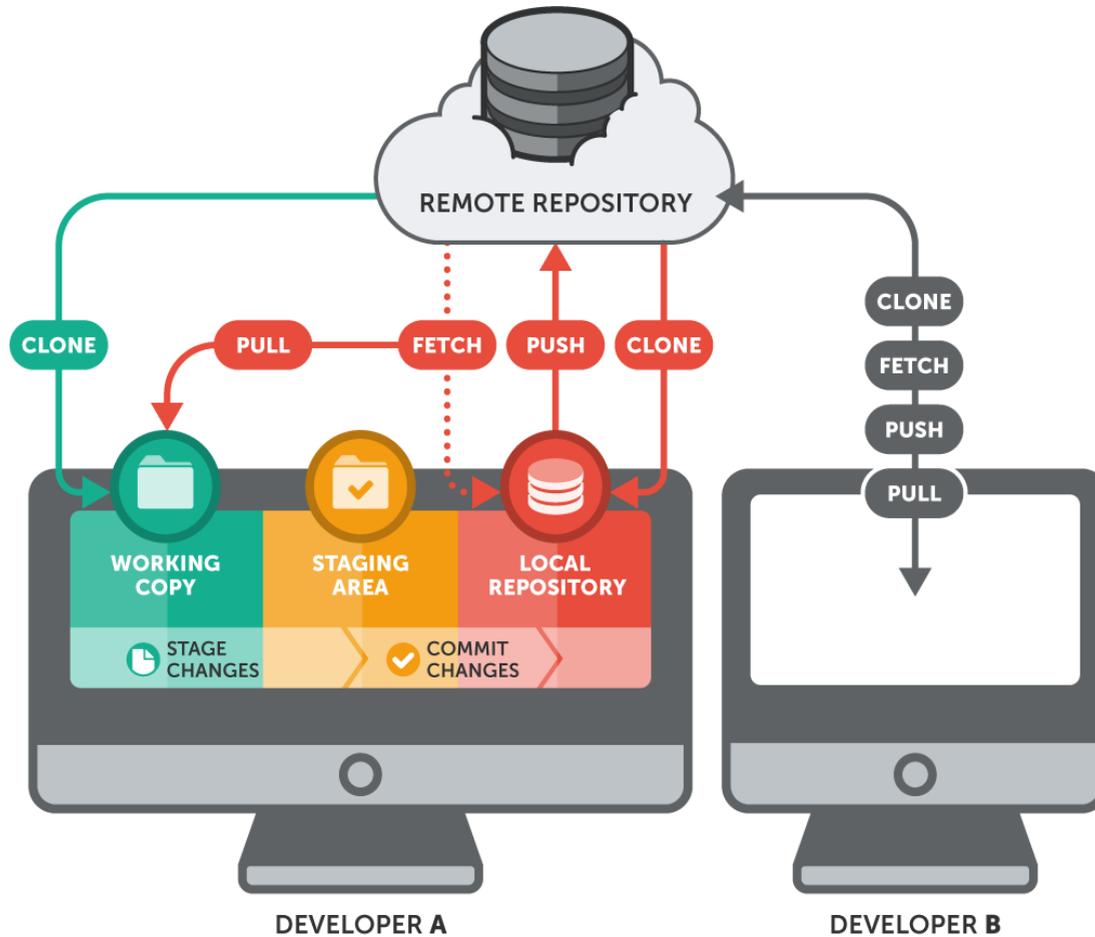
## Adding files

```
git status # What has changed?  
git add . # Stage all changes  
git commit -m "My very first commit! :)"
```

- New files are "untracked", not watched for changes
- To start tracking "stage" and "commit" files



# Git





Now we can push changes to our remote repository

```
git push
```

The screenshot shows an IDE interface with a Git repository. The left pane shows the Git status with a table of staged files:

| Staged                   | Status | Path       |
|--------------------------|--------|------------|
| <input type="checkbox"/> | ??     | day2.Rproj |

The right pane shows the file explorer with a table of files:

| Name           | Size    | Modified               |
|----------------|---------|------------------------|
| ..             |         |                        |
| .gitignore     | 40 B    | Nov 10, 2021, 4:05 AM  |
| day2.Rproj     | 258 B   | Nov 10, 2021, 4:05 AM  |
| README.md      | 56 B    | Nov 10, 2021, 4:05 AM  |
| appendix.Rmd   | 1.6 KB  | Nov 9, 2021, 5:13 AM   |
| data           |         |                        |
| manuscript.Rmd | 63.3 KB | Nov 10, 2021, 12:42 AM |
| references.bib | 49.4 KB | Mar 24, 2021, 9:16 AM  |

```
git pull
```

# Git

```
git log
```

# Git

## Standard Git workflow

```
git pull
git status
git add .
git commit -m "Such exciting changes! :)"
git log
git push
```



GitHub provides a web interface for your repository

- List of files
- Overview of commit history

The screenshot shows a GitHub commit page for a repository. The commit message is "My very first commit! :)" and it is on the "main" branch. The commit was made by "crsh" 14 minutes ago. The commit hash is 817697838e08f54b6ac6dd520d5d90529507e041. The page shows a diff for the .gitignore file, which has 6 changed files with 1,311 additions and 0 deletions. The diff shows the following changes:

```
@@ -0,0 +1,7 @@
1 + .Rproj.user
2 + .Rhistory
3 + .RData
4 + .Ruserdata
5 + .DS_Store
6 + .rdx
7 + .rdb
```

# Exercise time

Exercise

Solutions