

# Git Basics

Models, Algorithms and Data (MAD): Introduction to computing

Prof. Jens H. Walther, Dr. Julija Zavadlav

FS19

Git is a state-of-the-art version control software. It aims to help teams or individual developers maintaining a well organized history of their project and simplify collaboration. In the MAD lecture you will use `git` to submit your exercises and get a first feel for it. In the following we will only introduce you to standard

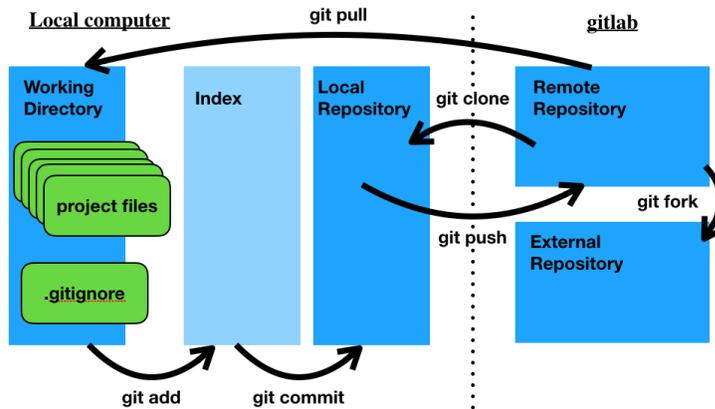


Figure 1: Figure illustrating the basic functionality of git.

use-cases of the basic functionality as illustrated in figure 1. For any further inquiries and many more insight please visit <https://git-scm.com/book/en/v2>, or the <https://git-scm.com/docs>. In the following we will list some of the most frequently used cases in the order they are used when creating a new repository:

**git clone *address*** Copies the file from the remote repository with *address* (which on gitlab can be found when pressing on the `clone` button) onto your local machine. Any changes will be tracked and if someone else worked on the repository prior to you, you can also access these changes.

**git add *file*** This adds your changes on the specified *file* to the index, where the status of the repository is stored. If you want to add all changes, this can be either done using `git add .` adding all files in the current and sub-directories or using `git add -A` adding all changes in the repository (also the ones in higher directories).

**git commit** In order to be able to return to the current status of the repository even after further edits we have to commit the changes indexed. Running the command as given will open a file where we are asked to enter a comment regarding our commit. Another way is to commit using `git commit -m "my comment"`. **Always enter a reasonable comment about what you are committing!**

**git push** After having something ready (this might only happen after many commits to your local repository) which you want to upload to the online version of your repository, you can push it to be uploaded.

**git pull** The command opposed to pushing is pulling and will allow you to update your local repository if someone else added changes to the remote repository.

We hope that with these fundamental commands you are ready to go. For a more elaborate discussion on the fundamentals we refer to <https://swcarpentry.github.io/git-novice/>