|  |  |  |
| --- | --- | --- |
| **Instruction** | **Utilisation** | **Exemples** |
| status | Shows the working tree status  The git status command displays the state of the working directory and the staging area. It lets you see which changes have been staged, which haven’t, and which files aren’t being tracked by Git. Status output does *not* show you any information regarding the committed project history. For this, you need to use git log |  |
| diff | git-diff - Show changes between commits, commit and working tree, etc |  |
| add | git-add - Add file contents to the index |  |
| commit | git-commit - Record changes to the repository |  |
| log | git-log - Show commit logs |  |
| push | git-push - Update remote refs along with associated objects |  |
| pull | git-pull - Fetch from and integrate with another repository or a local branch |  |
| clone | git-clone - Clone a repository into a new directory |  |

**Memo GIT commands : DATA +8**

**Installation**

1. **first install homebrew**

on my mac terminal run this :

/bin/bash -c "$(curl –fsSL <https://raw.githubusercontent.com/Homebrew/install/master/install.sh>)"

Installation successful !

from <https://brew.sh/>

1. **then install git**

run this

$ brew install git

from https://git-scm.com/download/mac

https://git-scm.com/about

**Repository** :

<https://git-scm.com/book/en/v2/Git-Basics-Getting-a-Git-Repository>

**Git vs Github**

Git is a version control system that lets you manage and keep track of your source code history. GitHub is a cloud-based hosting service that lets you manage Git repositories. If you have open-source projects that use Git, then GitHub is designed to help you better manage them.