# Git & GitHub Manual

## Installation of Git Bash

## Create and clone repositories

1. Go to GitHub online (<https://github.com>). If you don’t have an account you need to create one;
2. To create a new **repository**, click on your account area in the right top corner, then click in “Repositories” and then click “New”. If you just want to clone an existing **repository**, jump to step 4;
3. Write the **repository** name and add a simple description of your **repository**. If you want to keep your repository **Private** you can only share it with 3 people;
4. Now you have a repository in GitHub server but you need to **clone** it to your PC (**local repository**) so you be able to work offline. For that, click on “Clone or download” and copy the URL;
5. Then, follow the steps below:
   * Open Git Bash prompt;
   * Travel to the folder where you want your **local repository** to be through the commands cd <folder name> and ls;

Ex: Lamaral ~ $ cd Shared\ with\ me

Lamaral Shared with me $ cd DTOceanPlus

Lamaral GitHub $ cd GitHub

* + Write git clone <URL>;

1. Write cd <repository name> to get into the **master branch** of the **repository**.

## Status, commit, push and pull commands

1. After working on your files and save them as usual, you can write git status on Git Bash to check what changes were made. Basically, status compares your **local repository** files before and after the last **commit** (or the **commit** you are working on);
2. If this modified: <file name>.type appears, means that you have some files that are different from the **repository**. To update your **repository** you need to follow the following steps:
   * git add “<file\_name>” (use status command to check which files can be added, see step 1);
   * create a **commit** (it is essentially a version of your file) using command git commit -m “<commit\_description>”;
   * type git push to upload your files to de **online repository** (GitHub server);
3. To update your local repository and check your co-workers modifications, write git pull.

## Useful commands:

* Discard unstaged changes:
  + git stash save --keep-index
  + git stash drop
* Load older commits (<i> is the commit number, <i>=1 is the most recent commit):
  + git reset --hard HEAD~<i>