

git - Part 2

The lesson images are taken from [git](#) site.

For more information see [git-book](#) and [git-doc](#).

See also [github](#) for details.

Create a new branch and move on it

Go online on [github](#) and, on an existing repository, create a new branch in the [branches](#) page with the button [New branch](#). We will use the label [<branch>](#) for the created branch.

On your terminal, inside the git local repository, to move in the new created branch use:

```
git pull
git checkout <branch>
```

Pull request

Once we finish coding in the branch, you can use the [pull request](#) to **merge** the new code with the existing one in the [main](#) (or [master](#)) branch.

Use the online [github](#) pull request page to create and close a pull request from the branch [<branch>](#) to the [main](#) (or [master](#)) branch.

Once you complete the pull request, delete the branch [<branch>](#) online.

On your local repository, move to the [main](#) (or [master](#)) branch

```
git checkout main
```

download all the new data with

```
git pull
```

update the local branches with

```
git fetch --prune
```

and finally remove the no-more-used branch [<branch>](#) with

```
git branch -D <branch>
```

Conflicts

Conflicts may happen when you work in parallel on the same file.

DON'T PANIC

`git` will warn you about the conflicts in a file using the special characters `<<<`, `===` and `>>>`.

See for an example [here](#).

Choose the right part of the code you want to keep, then commit and push the one selected.