Git good practices

This tutorial assume you want to add a new functionality to master branch and create a temporary branch called task that implement your work. In practice, this branch should have an explicit name related to the task.

git clone repo\_url # clone the repository if you don’t have it yet.

git fetch # pull all branch if things have changed.

git prune # synchronize removed branch.

git checkout –b task # create branch and move on it.

\_\_code here\_\_

Loop until task is completed

git add filename # add your modifications

git commit –m “commit message” # comment your modifications

git push –u origin task # send your modifications on git repo

Once you have finished your development, we recommend you to submit a merge request using the web interface.

If you want to merge locally, you can run:

git rebase master # move to master and task last history intersection

git checkout master # switch to master branch

git merge task –no-ff # merge task and master branch

git push # send master update on git repo

git branch –D task # remove task branch

git push origin:task # inform git repo than task branch is deleted

# Useful commands

git status # show changed files

git diff # show changed lines