

Git Commands

2012-10-27

1 Add <files> to be committed

`git add <files>`

Use 'git add .' to commit everything in the current directory and its subdirectories.

2 Commit <files>, where <description> is the commit message

`git commit -m <description>` Remember to do a pull after committing and before pushing.

Commit → Pull → Push

3 Create a new local <branch>

`git branch <branch>`

4 Create a new local branch <new.branch> from <branch.to.copy>

`git checkout -b <new.branch> <branch.to.copy>`

5 Create a new remote branch <new.branch> from the current branch and tracks it

`git push -u origin <new.branch>`

Tracking is essentially a link between a local and remote branch. When working on a local branch that tracks some other branch, you can git pull and git push without any extra arguments and git will know what to do.

You can also do 'git push origin <branch>', but then every time you want to pull or push you need to specify the remote branch you want to pull from or to push to.

6 Pull <remote.branch> from origin

`git pull origin <remote.branch>`

7 Switch to the local branch <branch>

`git checkout <branch>`

8 Delete local <branch>

`git branch -d <branch>`

9 Delete <branch> at origin

`git push origin --delete <branch>`

10 Merge <branch.to.merge> into the current branch

`git merge --no-ff <branch.to.merge>`

11 Remove <file> from git, and from disk too

`git rm <file>`