

Git : cheatsheet

1 Configuration globale

```
Nom      $ git config --global user.name "Your Name"
Email    $ git config --global user.email "foo@bar.be"
Couleur  $ git config --global color.ui auto
```

2 Création du dépôt

Dans le dossier principal du projet :

```
$ git init
```

Ou bien faire un clone d'un projet existant :

```
$ git clone <url>
```

3 Commandes de base

Voir l'état actuel du dépôt :

```
$ git status
```

Voir l'historique :

```
$ git log [-p]
```

```
$ git lola
```

Voir les modifications en cours :

```
$ git diff
```

4 Création d'un *commit*

Marquer ce qu'il faut mettre dans le *commit*

En fonction du type de modification, il faut utiliser la commande appropriée.

```
Modifier  $ git add <file>1
Ajouter   $ git add <file>
Supprimer $ git rm <file>
Déplacer  $ git mv <old_name> <new_name>
```

¹ Si on veut marquer toutes les modifications (mais pas les nouveaux fichiers), il y a un raccourci : `$ git commit -a`.

Écrire le message du commit et le créer

```
$ git commit
```

L'option `-m` peut être utilisée pour passer le message directement :

```
$ git commit -m "message du commit"
```

5 Branches

Créer une branche

```
$ git branch <name>
```

Changer de branche

```
$ git checkout <branch>
```

Merge une branche

Merge la branche courante avec `other_branch`.

```
$ git merge <other_branch>
```

Rebase une branche

Rebase la branche courante « au dessus » de `other_branch`.

```
$ git checkout <other_branch>
```

Prendre les modifications distantes

```
$ git pull
```

Envoyer ses modifications

```
$ git push
```