

# TP Git - Résumé des étapes

Riyad Derguini

18 janvier 2025

## Table des matières

<b>1</b>	<b>Préparation</b>	<b>1</b>
1.1	Installation de Git . . . . .	1
1.2	Installation de Tig et Gitk (facultatif) . . . . .	1
1.3	Configuration . . . . .	2
<b>2</b>	<b>Solo Linéaire</b>	<b>2</b>
2.1	Création du dépôt et premier commit . . . . .	2
2.2	Deuxième commit et observation . . . . .	2
2.3	Annuler les changements . . . . .	2
<b>3</b>	<b>Solo Branches</b>	<b>2</b>
3.1	Création et manipulation de branches . . . . .	2
3.2	Merge de branches . . . . .	3
<b>4</b>	<b>Collectif</b>	<b>3</b>
4.1	Cloner un dépôt commun . . . . .	3
4.2	Travail en parallèle sur des fichiers disjoints . . . . .	3
4.3	Cadavre exquis . . . . .	3
4.4	Migrer vers GitLab . . . . .	3

## 1 Préparation

### 1.1 Installation de Git

Installez Git (version  $\geq 2.23$ ) et vérifiez la version avec la commande :

```
1 $ git --version
```

### 1.2 Installation de Tig et Gitk (facultatif)

Installez les paquets `tig` et `gitk` pour une visualisation graphique de l'historique Git.

## 1.3 Configuration

Configurez Git avec votre nom et adresse mail :

```
1 $ git config --global user.name "<Pr nom> <Nom>"
2 $ git config --global user.email "<adresse_mail>"
```

Pour définir la branche par défaut sur main :

```
1 $ git config --global init.defaultBranch main
```

## 2 Solo Linéaire

### 2.1 Création du dépôt et premier commit

Créez un répertoire `tp-git/` et initialisez un dépôt Git :

```
1 $ mkdir tp-git
2 $ cd tp-git
3 $ git init
```

Créez un fichier `plop` avec le contenu `hop`, ajoutez-le à la staging area et commitez :

```
1 $ echo "hop" > plop
2 $ git add plop
3 $ git commit -m "mon premier commit"
```

### 2.2 Deuxième commit et observation

Modifiez le fichier `plop` et observez les changements avec `git status` et `git diff`.

### 2.3 Annuler les changements

Utilisez `git restore` pour annuler les modifications non commitées.

## 3 Solo Branches

### 3.1 Création et manipulation de branches

Créez une branche `essai` et basculez dessus :

```
1 $ git branch essai
2 $ git switch essai
```

Créez un tag `milieu-test` sur le commit courant :

```
1 $ git tag milieu-test
```

## 3.2 Merge de branches

Mergez la branche `essai` dans `main` :

```
1 $ git switch main
2 $ git merge essai
```

## 4 Collectif

### 4.1 Cloner un dépôt commun

Clonez le dépôt distant avec :

```
1 $ git clone ssh://gituser@yologit.netlib.re:2271/~/<
    nom_du_depot_distant>
```

### 4.2 Travail en parallèle sur des fichiers disjoints

Chaque membre de l'équipe crée un fichier dans le répertoire `perso/` et pousse ses modifications.

### 4.3 Cadavre exquis

Jouez au cadavre exquis en modifiant un fichier commun `jeu.1.txt` sur des branches séparées.

### 4.4 Migrer vers GitLab

Créez un dépôt sur GitLab et ajoutez-le comme remote :

```
1 $ git remote add gitlab https://git.ig-edu.univ-paris13.fr/<
    nom_groupe>/<nom_d p t>.git
```