

## Commandes Git et GitHub Essentielles

### Commandes Générales

1. Cloner un dépôt GitHub :

```
git clone <URL-du-dépôt>
```

2. Ajouter des modifications :

```
git add <fichier-ou-dossier>
```

Pour tout ajouter : `git add .`

3. Créer un commit :

```
git commit -m "Votre message de commit"
```

4. Pousser des changements vers GitHub :

```
git push origin <nom-de-la-branche>
```

5. Récupérer des modifications :

```
git pull
```

6. Initialiser un dépôt Git local :

```
git init
```

7. Vérifier l'état du dépôt :

```
git status
```

8. Voir l'historique des commits :

## Commandes Git et GitHub Essentielles

`git log`

Pour un affichage simplifié : `git log --oneline`

## Commandes Git et GitHub Essentielles

### Commandes pour Gérer les Branches

1. Créer une branche :

```
git branch <nom-de-la-branche>
```

2. Basculer sur une branche :

```
git checkout <nom-de-la-branche>
```

3. Créer et basculer directement sur une nouvelle branche :

```
git checkout -b <nom-de-la-branche>
```

4. Supprimer une branche (localement) :

```
git branch -d <nom-de-la-branche>
```

Pour forcer : 

```
git branch -D <nom-de-la-branche>
```

5. Supprimer une branche distante :

```
git push origin --delete <nom-de-la-branche>
```

6. Fusionner une branche dans une autre :

```
git merge <nom-de-la-branche>
```

## Commandes Git et GitHub Essentielles

### Commandes Avancées

1. Créer un lien avec un dépôt distant :

```
git remote add origin <URL-du-dépôt>
```

2. Réinitialiser des modifications non validées :

```
git reset --hard
```

3. Résoudre un conflit de merge :

- Après un conflit, éditez les fichiers concernés, puis :

```
git add <fichiers-résolus>
```

```
git commit
```

4. Voir les différences entre modifications locales :

```
git diff
```