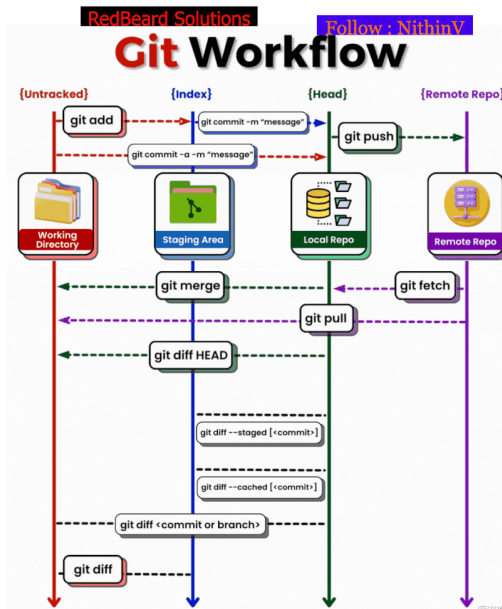


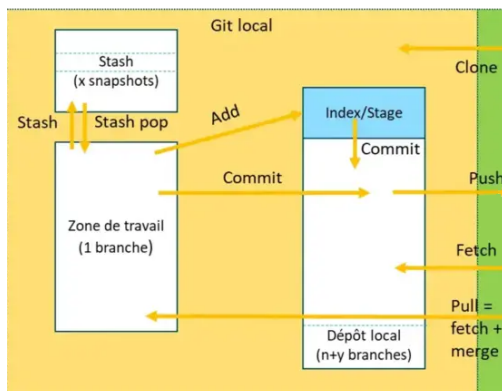
# Git, opérations courantes



## Introduction

Git est un **système de contrôle** de version décentralisé couramment utilisé pour le développement de logiciels. Voici une liste des commandes Git les plus usuelles pour les opérations locales et distantes :

## Commandes Git locales



**1. Initialiser un dépôt Git :**

```
git init
```

Initialise un nouveau dépôt Git dans le répertoire courant.

**2. Cloner un dépôt existant :**

```
git clone <url-du-depot>
```

Clone un dépôt distant vers le répertoire local.

**3. Afficher l'état du dépôt :**

```
git status
```

Affiche l'état des fichiers dans le répertoire de travail.

**4. Ajouter des fichiers au staging area :**

```
git add <fichier>
```

```
git add .
```

Ajoute des fichiers spécifiés ou tous les fichiers modifiés au staging area.

**5. Valider les modifications :**

```
git commit -m "Message de commit"
```

Enregistre les modifications dans l'historique des versions.

**6. Voir l'historique des commits :**

```
git log
```

Affiche l'historique des commits.

**7. Voir les différences entre les commits :**

```
git diff
```

Affiche les différences entre le répertoire de travail et l'index (staging area) ou entre les commits.

**8. Créer une nouvelle branche :**

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

Crée une nouvelle branche.

9. **Changer de branche :**

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

Change de branche.

10. **Fusionner une branche :**

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

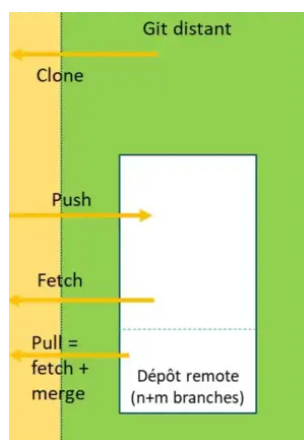
Fusionne la branche spécifiée avec la branche courante.

11. **Supprimer une branche :**

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

Supprime une branche locale.

## Commandes Git distantes



1. **Ajouter un dépôt distant :**

```
git remote add origin <url-du-depot>
```

Ajoute un dépôt distant avec le nom origin.

2. **Afficher les dépôts distants :**

```
git remote -v
```

Affiche les dépôts distants configurés.

3. **Récupérer les modifications du dépôt distant :**

```
git fetch
```

Récupère les branches et leurs commits depuis le dépôt distant.

4. **Récupérer et fusionner les modifications du dépôt distant :**

```
git pull
```

Récupère les modifications et les fusionne avec la branche locale courante.

5. **Envoyer les modifications vers le dépôt distant :**

```
git push
```

Envoie les commits de la branche locale vers la branche correspondante du dépôt distant.

6. **Envoyer une branche spécifique vers le dépôt distant :**

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

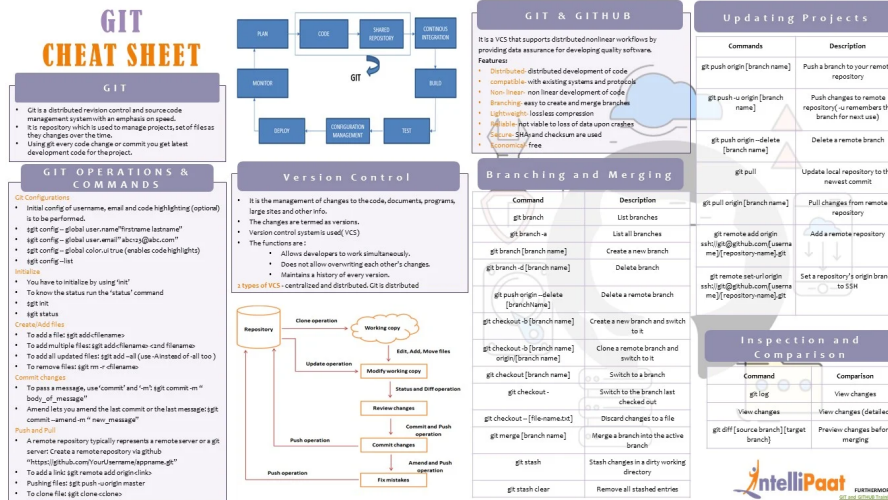
Envoie une branche spécifique vers le dépôt distant.

7. **Supprimer une branche distante :**

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

Supprime une branche du dépôt distant.

## Commandes avancées



### 1. Rebase une branche :

`git rebase <branche-de-base>`

Applique les commits de la branche courante sur la branche spécifiée.

### 2. Réinitialiser une branche à un commit spécifique :

`git reset --hard <commit>`

Réinitialise la branche courante à un commit spécifique, en supprimant les changements locaux.

### 3. Stash des modifications :

`git stash`

Enregistre temporairement les modifications non commit et les retire du répertoire de travail.

### 4. Appliquer un stash :

`git stash apply`

Applique les modifications stashed au répertoire de travail.

### 5. Liste des stashes :

`git stash list`

Affiche la liste des stashes.

## **Conclusion**

Ces commandes couvrent la plupart des opérations courantes avec Git. Pour des informations plus détaillées, consultez la documentation officielle de Git ou utilisez l'option `--help` avec n'importe quelle commande Git (par exemple, `git status --help`).