

Guide Étape par Étape pour Utiliser Git avec GitHub

Introduction

Ce guide explique comment utiliser Git avec GitHub pour vos projets de programmation. Nous aborderons les étapes pour initialiser un dépôt, ajouter des fichiers à la staging area, les committer, et enfin les pousser sur GitHub. Nous expliquerons également comment configurer des tokens d'accès et des clés SSH pour sécuriser vos connexions avec GitHub.

1. Initialiser un Dépôt Git

Créez un nouveau répertoire pour votre projet et initialisez un dépôt Git:

```
mkdir mon_projet
cd mon_projet
git init
```

Explication:

`mkdir mon_projet`: Crée un nouveau répertoire nommé `mon_projet`.

`cd mon_projet`: Change le répertoire courant pour entrer dans le répertoire `mon_projet`.

`git init`: Initialise un nouveau dépôt Git dans le répertoire actuel.

2. Créer des Fichiers et les Ajouter à la Staging Area

Créez des fichiers et ajoutez-les à la staging area:

```
echo "# Mon Projet" > README.md
git add README.md
```

Explication:

`echo "# Mon Projet" > README.md`: Crée un fichier `README.md` avec le contenu `# Mon Projet`.

`git add README.md`: Ajoute le fichier `README.md` à la staging area.

3. Committer les Modifications

Committez les fichiers ajoutés avec un message descriptif:

```
git commit -m "Initial commit"
```

Explication:

`git commit`: Enregistre les modifications dans l'historique du dépôt avec un message descriptif. Le texte après `-m` est le commit message, qui doit décrire les modifications apportées.

4. Ajouter un Dépôt Distant sur GitHub

Ajoutez un dépôt distant pour votre projet GitHub:

```
git remote add origin https://github.com/votre_utilisateur/mon_projet.git
```

Explication:

`git remote add origin`: Lie le dépôt local à un dépôt distant nommé `origin`. Remplacez `https://github.com/votre_utilisateur/mon_projet.git` par l'URL de votre dépôt GitHub.

5. Pousser les Modifications vers GitHub avec un Token

Poussez les modifications locales vers le dépôt distant sur GitHub en utilisant un token d'accès personnel:

```
git push -u origin main
```

Explication:

`git push`: Envoie les commits locaux au dépôt distant sur GitHub. `-u origin main` définit `origin` comme le dépôt distant par défaut pour la branche `main`. Vous serez invité à entrer votre nom d'utilisateur GitHub et votre token d'accès personnel au lieu de votre mot de passe.

6. Générer un Token d'Accès Personnel sur GitHub

Pour créer un token d'accès personnel sur GitHub:

1. Allez sur <https://github.com/settings/tokens>
2. Cliquez sur `Generate new token`

3. Donnez un nom à votre token et sélectionnez les autorisations nécessaires (généralement repo)
4. Cliquez sur Generate token et copiez le token généré

7. Pousser les Modifications vers GitHub avec une Clé SSH

Générer une nouvelle clé SSH:

```
ssh-keygen -t rsa -b 4096 -C "votre_email@example.com"
```

Explication:

ssh-keygen: Génère une nouvelle clé SSH.
-t rsa -b 4096: Spécifie le type de clé RSA avec une longueur de 4096 bits.
-C "votre_email@example.com": Ajoute un label avec votre email.

Ajouter votre clé SSH à l'agent SSH:

```
eval "$(ssh-agent -s)"  
ssh-add ~/.ssh/id_rsa
```

Explication:

eval "\$(ssh-agent -s)": Lance l'agent SSH en arrière-plan.
ssh-add ~/.ssh/id_rsa: Ajoute votre clé privée à l'agent SSH.

Ajouter votre clé SSH à GitHub:

```
cat ~/.ssh/id_rsa.pub
```

Explication:

Utilisez cette commande pour afficher le contenu de votre clé publique, puis copiez-le et ajoutez-le à votre compte GitHub sous "Settings" > "SSH and GPG keys" > "New SSH key".

Exemple Complet

Voici un exemple complet en suivant les étapes ci-dessus:

```
mkdir mon_projet  
cd mon_projet  
git init  
echo "# Mon Projet" > README.md  
git add README.md  
git commit -m "Initial commit"  
git remote add origin https://github.com/votre_utilisateur/mon_projet.git  
git push -u origin main  
ssh-keygen -t rsa -b 4096 -C "votre_email@example.com"  
eval "$(ssh-agent -s)"  
ssh-add ~/.ssh/id_rsa  
cat ~/.ssh/id_rsa.pub
```