

Git et GitHub Dans un Contexte DevOps

Partie 1- Initiation

1. Introduction:

Ce TP a pour objectif de vous initier aux concepts de base de Git et GitHub, essentiels pour les ingénieurs DevOps. Vous allez apprendre à utiliser Git pour gérer efficacement les versions de votre code, ainsi qu'à utiliser GitHub pour héberger et partager votre code avec d'autres développeurs. Ce TP est destiné aux débutants, aucune connaissance préalable de Git ou GitHub n'est requise.

Durée: 3 heures (minimum)

Prérequis:

- Git installé sur votre machine
- Un compte GitHub

2. Exercice 1: Installation et configuration de Git (30 minutes)

1. Téléchargez et installez Git depuis le site officiel (<https://git-scm.com/downloads>) en fonction de votre système d'exploitation.
2. Vérifiez l'installation en exécutant la commande ``git --version`` dans votre terminal.
3. Configurez votre nom d'utilisateur et votre adresse e-mail avec les commandes suivantes :
 - ``git config --global user.name "Votre nom"``
 - ``git config --global user.email "votre@email.com"``

3. Exercice 2: Initialisation d'un repository local (30 minutes)

1. Créez un nouveau répertoire vide sur votre machine.
2. Placez-vous dans ce répertoire et utilisez la commande ``git init`` pour initialiser un repository Git local.
3. Vérifiez l'état de votre repository en utilisant la commande ``git status``.
4. Créez quelques fichiers de votre choix dans ce répertoire.

4. Exercice 3: Commits et gestion des versions (60 minutes)

1. Utilisez la commande ``git add`` pour ajouter les fichiers créés dans l'exercice précédent à la zone de staging.
2. Utilisez la commande ``git commit`` pour créer votre premier commit en ajoutant un message descriptif.
3. Modifiez certains fichiers existants et créez-en quelques-uns de plus.
4. Utilisez à nouveau les commandes ``git add`` et ``git commit`` pour ajouter ces modifications à un nouveau commit.
5. Utilisez la commande ``git log`` pour voir l'historique de vos commits.

5. Exercice 4 : Création d'un repository distant sur GitHub (30 minutes)

1. Rendez-vous sur GitHub (<https://github.com>) et connectez-vous à votre compte (ou créez-en un si vous n'en avez pas).

2. Cliquez sur le bouton "New" pour créer un nouveau repository.
3. Donnez un nom à votre repository, choisissez les options de visibilité souhaitées et créez-le.
4. Suivez les instructions de GitHub pour ajouter un repository distant à votre repository local.
5. Utilisez la commande ``git remote -v`` pour vérifier que votre repository local est bien connecté au repository distant sur GitHub.
6. Utilisez la commande ``git push`` pour envoyer vos commits vers le repository distant.

6. Exercice 5 : Collaboration avec Git et GitHub (30 minutes)

1. Trouvez un partenaire et partagez vos adresses e-mail GitHub.
2. Ajoutez votre partenaire en tant que collaborateur de votre repository distant sur GitHub.
3. Demandez à votre partenaire de cloner votre repository distant sur sa machine en utilisant la commande ``git clone``.
4. Effectuez quelques modifications sur le code et utilisez ``git add``, ``git commit`` et ``git push`` pour envoyer les modifications vers le repository distant.
5. Demandez à votre partenaire d'effectuer des modifications sur le code et utilisez ``git pull`` pour récupérer les changements sur votre machine.

7. Conclusion:

Ce TP vous a permis de découvrir les bases de Git et GitHub. Vous avez appris à initialiser un repository, à gérer les versions de votre code avec des commits, à créer un repository distant sur GitHub et à collaborer avec d'autres développeurs. Ces compétences sont essentielles pour un ingénieur DevOps. Continuez à pratiquer régulièrement pour renforcer vos connaissances.