

Module : Initiation Github, Gitlab
Formation : IA – M1
Durée : 2 heures

CONSIGNES :

Livrable :

Un fichier .doc comportant le script complet de votre travail Git. Ce script doit contenir toutes les commandes que vous avez saisies sur le terminal Git, ainsi que leurs résultats. Assurez-vous d'organiser le script de manière claire et séquentielle, en suivant la chronologie de votre travail.

Le fichier doit inclure des captures d'écran de chaque étape de votre travail réalisé sur GitHub.

EXERCICE 1 : (12 Pts)

1. Affichez les paramètres de configuration de git.
2. Créez un nouveau dépôt Git vide dans un répertoire de votre choix. Appelez ce dépôt Git «Premier_depot_git»
3. Créez un fichier texte appelé monfichier.txt et ajoutez-y quelques lignes de texte.
4. Afficher l'état de votre répertoire de travail.
5. Ajoutez le fichier monfichier.txt au « stage ».
6. Effectuez un commit avec un message descriptif.
7. Créer une nouvelle branche appelée « ma-nouvelle-branche » et y basculer directement.
8. Fusionnez la branche « ma-nouvelle-branche » dans la branche principale master.
9. Créez un conflit en modifiant la même ligne de monfichier.txt à la fois dans la branche « ma-nouvelle-branche » et dans la branche master.
10. Essayez de fusionner les deux branches à nouveau. Expliquez pourquoi vous ne pouvez pas réaliser cette fusion.
11. Résolvez le problème puis utilisez la commande git add pour marquer le fichier comme résolu, enfin, effectuez un commit pour finaliser la fusion.
12. Supprimez la branche "ma-nouvelle-branche" après la fusion.
13. Créez un dépôt distant (repository) sur GitHub et appelez le «Mon_depot_Git_distant».
14. Liez votre dépôt local « Premier_depot_git » à votre dépôt distant «Mon_depot_Git_distant», puis poussez votre dépôt local sur votre dépôt distant.
15. Créer un nouveau répertoire sur votre ordinateur, puis clonez le dépôt distant existant «Mon_depot_Git_distant» depuis la plateforme d'hébergement GitHub vers ce nouveau répertoire.
16. Modifiez le fichier monfichier.txt dans le répertoire cloné, puis effectuez un commit avec un message descriptif.
17. Créez une étiquette (tag) pour marquer le commit que vous avez effectué à la question précédente.
18. Modifiez à nouveau le fichier monfichier.txt puis effectuez un commit avec un message descriptif.
19. Utilisez la commande permettant de voir les différences entre les deux commits que vous avez effectués précédemment.
20. Affichez l'historique des commits de votre dépôt Git, y compris les auteurs, les messages des commits.

EXERCICE 2 : (8 Pts)

1. Créez un nouveau dépôt Git vide dans un répertoire de votre choix. Appelez ce dépôt Git «deuxième_depot_git».
2. Ajoutez un fichier README.md et apportez des modifications.
3. Effectuez un commit avec un message descriptif.
4. Vous avez commis une erreur en effectuant ce commit. Annulez ce dernier commit sans perdre vos modifications locales.
5. Réalisez de nouvelles modifications à votre fichier README.md.
6. Vous devez faire une pause dans votre travail actuel pour répondre à un correctif d'urgence sur un autre projet. Utilisez la commande permettant de mettre de côté vos modifications locales, puis revenez à votre travail précédent lorsque le correctif est terminé.
7. Créez un nouveau dépôt distant sur GitHub et appelez-le «deuxième_depot_git_distant».
8. Poussez vos modifications vers votre dépôt distant «deuxième_depot_git_distant» après avoir créé les liens nécessaires.