**Plan de cours GIT**

**Dure totale du module : 21H00**

Table des matières

[Jour 1 : 3](#_Toc75438249)

[Jour 2 : 4](#_Toc75438250)

[Jour 3 : 6](#_Toc75438251)

# Jour 1 :

-Introduction à GIT (*qu’est-ce que GIT ?, à quoi cela sert-il ?*)

-Installation et configuration de l’outil (*page 4 à 5 du cours*)

Exercice installer et configurer l’outil (ignorer les points optionnels)

-Création d’un dépôt local et initialisation (*page 5 à 6 du cours*)

Exercice création d’un dépôt local et initialisation (*git init*)

-Création d’un compte sur github (*page 7 du cours*)

NB : ! attention au mail de confirmation

-Création d’un dépôt (repository) github (*page 8 à 9 du cours*)

-Clonage du dépôt distant dans le dépôt local (*page 9 du cours*)

-Structure git working directory, stage, commit (*page 10 du cours*)

-Création de fichier local + git add + commit (*page 11 du cours*)

Exercice créer un fichier local + commit du fichier

-Création multiple de fichiers locaux + git add \* + commit (*page 12 du cours*)

Exercice création multiple de fichiers + commit de fichiers

-Modification de fichiers locaux (*édition*) + commit (*page 12 du cours*)

Exercice modification de fichier + commit

-Suppression de fichiers locaux + commit (*page 13 du cours*)

Exercice suppression de fichiers + commit

-Commandes de base linux :

Se positionner dans un dossier : ***cd url\_du\_dossier*** (*ex : c:/users/nom\_utilisateur/desktop/depot\_local*)

Créer un dossier : ***mkdir*** (*il faut un fichier à l’intérieur pour pouvoir commit on add avec git add \**)

Visualiser le contenu d’un dossier : ***ls***

Créer un fichier en ligne de commande : ***touch nomDuFichier.extension***

-interaction avec un dépôt distant : télécharger le contenu d’un dépôt github (*git pull url\_depot\_github*). (*Page 16 du cours*)

Exercice télécharger le contenu de son dépôt distant github(*repository*)

-interaction avec un dépôt distant : uploader le dépôt local vers le contenu d’un dépôt github (*git push url\_depot\_github nom\_branche* ***master*** *par défaut*). (*Page 16-17 du cours*)

Exercice télécharger le contenu de son dépôt distant github(*repository*)

-Création, modification et suppression de fichier sur github

Exercice création, modification et suppression de fichiers sur le repository github

-Gestion des branches (*page 18-19 du cours*)

Exercice création de branches en local

-TP GIT

# Jour 2 :

-Récapitulatif des notions du jour 1.

-Correction du TP GIT

-Les remises Stash (*page 14-15 du cours*)

Exercices créer ou modifier des fichiers en local, puis les stocker dans des remises, basculer les remises sur la branche principale et sur une autre branche.

-Collaboration sur un repository github. TP GIT 02

Créer un repository sur github (*compte github du formateur*).

Ajouter les stagiaires dans settings du repo -> manage access (*on importe les stagiaires avec leurs adresses mail ou login github*)

NB : ! Attention les stagiaires reçoivent un mail de confirmation pour participer.

Lancer le TP GIT 02

-Correction TP GIT 02

-Git reset (*page du cour 23*).

Exercice lancer le TP GIT 03 crash test

-Correction TP 03

Pour les stagiaires qui ont des erreurs au moment du push (*normalement tous sauf le plus rapide !!!*)

Solution :

On stoppe tout !!! **plus aucun stagiaire ne doit effectuer des push vers le dépôt du formateur**.

-le stagiaire doit **relever l’id du commit** correspondant à la création de la branche master (*du formateur*) avec la commande **git log**

-le stagiaire doit utiliser la commande **git reset *id\_commit\_creation\_master\_formateur* --hard**

-le stagiaire doit ré-effectuer un pull de master.

-le stagiaire réapplique sa remise avec **git stash apply**.

-le stagiaire commit sa remise.

-le stagiaire push vers master.

On passe au stagiaire suivant avec la même méthode.

-Débriff du TP 03 crash test.

# Jour 3 :

-Récapitulatif des notions du jour 2.

-Commande amend modification du message du dernier commit (*page 20 du cours*).

Exercice renommer son dernier commit.

- Ajouter un fichier oublié dans le dernier commit (*page 20 du cours*).

Exercice ajouter un fichier oublié dans son dernier commit.

-Restauration de la branche master. (*Optionnel voir page 22*)

- Corriger un mauvais commit à distance. (*Optionnel voir page 22*)

-Gestion des Cas d’erreurs :

Présentation des cas d’erreurs (*page cours 25 à 31 en fonction du temps de cour restant*)

-Merge (*page 32-33 du cours*).

-Présentation des alternatives avec interface graphique (*Github desktop, tree Source etc…*)

-**Evaluation GIT**.

-Correction Evaluation.

-Débriff du cours de git avec les stagiaires.