





Travail à rendre : Git – TERRIER Julien

MainRemote

Séquence d'introduction

Une introduction en douceur à la majorité des commandes Git





1: Introduction aux commits avec Git



Montée en puissance

Le prochain excellent plat de pur Git. J'espère que vous êtes affamés



1: Détacher votre HEAD



Déplacer le travail

Soyez à l'aise pour modifier l'arbre Git





1: Introduction à cherry-pick



Un assortiment

Un assortiment de techniques et astuces pour utiliser Git




1: Choisir seulement 1 commit



Sujets avancés

Pour les plus courageux !

1: Rebaser plus de 1000 fois











MainRemote

Push & Pull -- Dépôts Git distants !

Il est le temps de partager vos 1 et vos 0 les enfants, le code vient de devenir social.









1: Introduction à clone



Vers l'infini et au-delà -- dépôts distants version avancée

Et vous pensiez qu'être un dictateur bienfaisant serait amusant...

1: Maître du push !



- **Git config :**

Cette commande sert à configurer plusieurs choses. Par exemple, il est possible de configurer un nom d'utilisateur ou une adresse email en global ou local:

- **git config --global user.name « Julien »**
- **git config user.email « julien@utbm.fr »**

- **Git init :**

Cette commande permet d'initialiser git. Concrètement, cela permet de créer un dossier cache « .git » qui est un sous-répertoire du répertoire dans lequel on se trouve.

- **Git status :**

Cette commande permet de vérifier le statut du répertoire de travail. C'est la commande qui nous montre quels fichiers ont été modifiés, supprimés, ajoutés, ou renommés. Ça nous dit aussi dans quelle branch on se trouve.

- **Git add :**

Cette commande permet de préparer un futur git commit en ajoutant le contenu du fichier à l'index.

- **Git push :**

Cette commande permet de « pousser » et donc de publier des changements effectués dans des commits locaux, vers le serveur distant.

- **Git merge :**

Cette commande permet de combiner le contenu de deux branches différentes en les fusionnant dans une nouvelle branche créée.

- **git merge <branche>**

- **Git diff :**

Cette commande permet de connaître les différences entre plusieurs commits ou branches :

- **git diff <commit1> <commit2>**

- **Git blame :**

Cette commande permet d'analyser ligne à ligne les modifications d'un fichier :

- `git blame <fichier>`