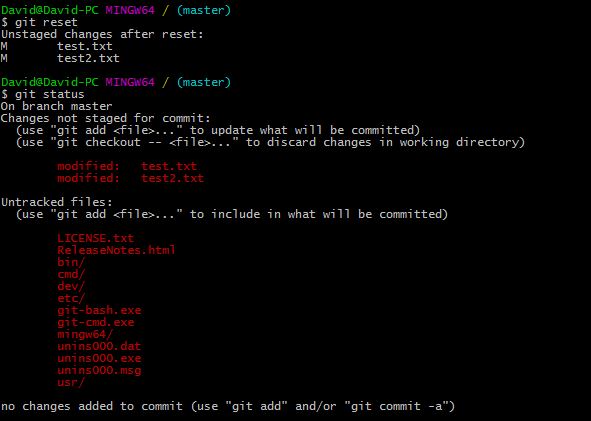
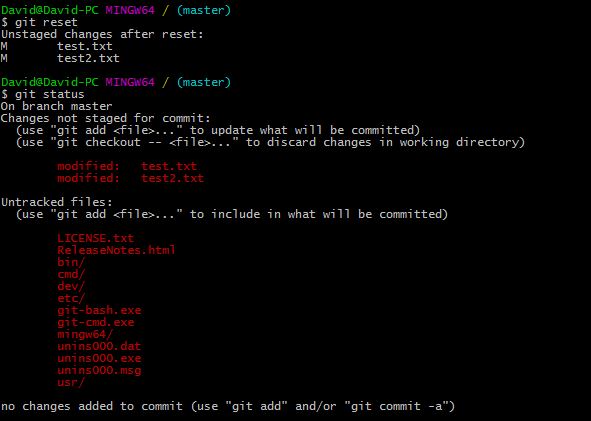
**Git init** « Git init » est la commande qui est utilisée pour la création d’un dépôt vide. La commande va aussi créer des dossiers dans le dépôt qui inclut des dossiers pour les objets, les références et les fichiers template. Si la commande est utilisée à l’intérieur d’un dépôt déjà existant, rien ne sera remplacé. La commande peut être utilisé à l’intérieur d’un dépôt existant pour soit le déplacé vers un autre emplacement ou pour ajouter de nouveaux templates ajoutés.

C:\Users\David\AppData\Local\Microsoft\Windows\INetCache\Content.Word\captureGitInit.jpg

**Git reset**

Git reset peut être vu comme étant l’inverse de « git add ». Donc si on ajoute des fichiers à l’aide de « git add », il est possible d’utiliser « git reset » pour enlever les fichiers en attentes de se faire « commiter ».



**Git stash**

Git stash est une commande très utile pour pouvoir modifié autres choses sans perdre ce qu’il a été fait et pas envoyé. Il suffit d’utiliser la commande « git stash » et tous les modifications faites vont être sauvegardées en mémoire. Même si les modifications sont dans le « stash », il est tout de même possible de voir la liste des modifications avec « git stash list » et même les inspecter avec « git stash show ». Une fois les modifications dans le « stash », il est maintenant possible de faire d’autres modifications et de les « commit » sans envoyer eux du « stash ». Une fois les autres modifications envoyées, il suffit d’utiliser la commande « git stash pop » pour reprendre contrôle des autres modifications « stasher ».

