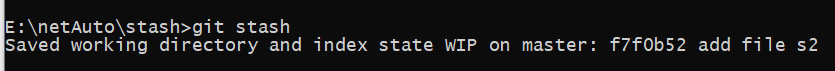
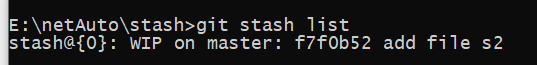
**git-stash - Stocker les modifications dans un répertoire de travail**

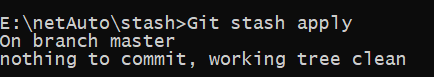
À utiliser git stash lorsque vous souhaitez enregistrer l'état actuel du répertoire de travail et de l'index, mais souhaitez revenir à un répertoire de travail propre.



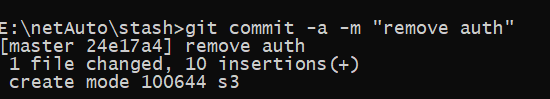
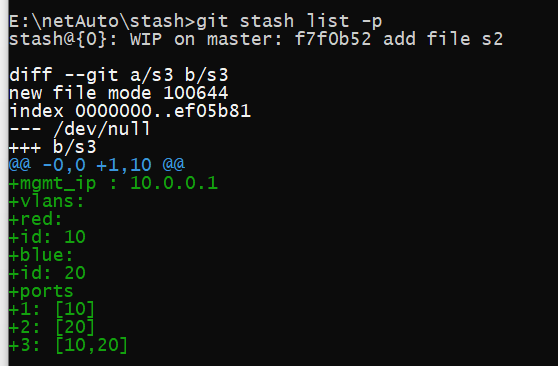
La commande enregistre vos modifications locales et rétablit le répertoire de travail pour qu'il corresponde à la HEAD validation. Les modifications stockées par cette commande peuvent être répertoriées avec git **stash list**, inspectées avec **git stash show** et restaurées (potentiellement au-dessus d'un commit différent) avec git stash apply. Appeler git stash sans aucun argument équivaut à git stash push.

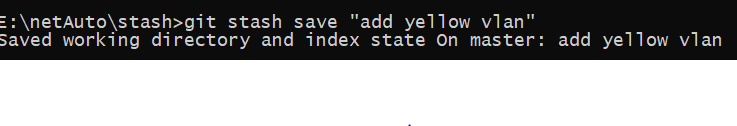


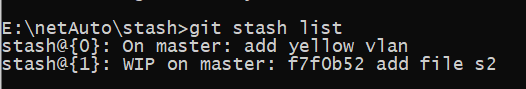
Un stash est par défaut répertorié comme "WIP on branch name …",

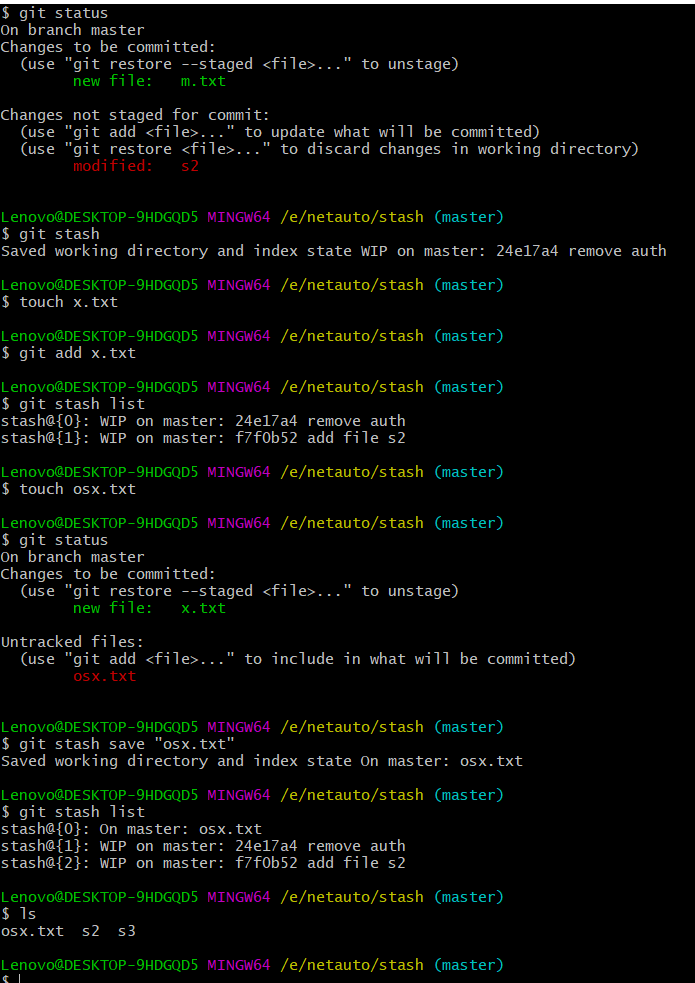


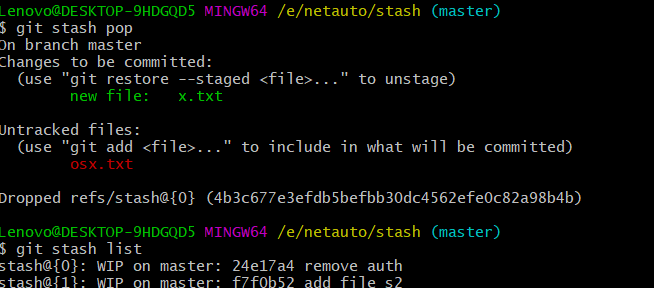
mais vous pouvez donner un message plus descriptif sur la ligne de commande lorsque vous en créez un. Le dernier stash que vous avez créé est stocké dans refs/stash; les caches plus anciennes se trouvent dans le reflog de cette référence et peuvent être nommées en utilisant la syntaxe de reflog habituelle (par exemple stash@{0}, est le cache le plus récemment créé, stash@{1}est celui qui le précède, stash@{2} est également possible). Les cachettes peuvent également être référencées en spécifiant uniquement l'index de cachette (par exemple, l'entier n est équivalent à stash@{n}).

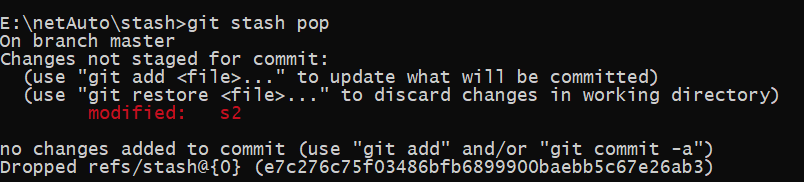


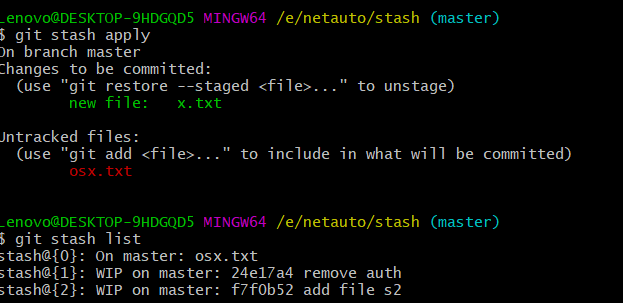


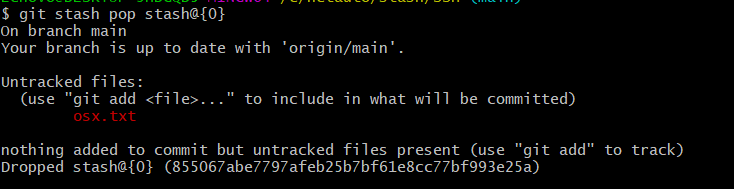


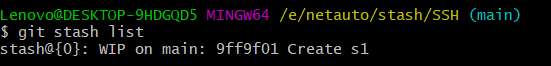












//supprimer le dernier stash



