# Compléments sur Git

Vocabulaire :

Copie de travail : l’arborescence de fichier se trouvant sur l’ordinateur

L’index : les tracked files, destinés à être commités

L’instantané du projet : le commit, destiné à être envoyé sur le serveur

## Signaler les fichiers dont on ne veut pas faire le suivi :

On crée un fichier .gitignore (gitignore est l’extension, le fichier n’a pas de nom. C’est important sinon il n’est pas lu)

A l’intérieur, on écrit les patrons des noms de fichiers à ignorer par git (ils ne seront pas proposé dans la liste des « untracked files », donc ça évite d’avoir des listes entières de fichiers que l’on ne veut pas rajouter 🡪 permet de travailler avec Eclipse et de partager directement les sources).

…

… : On tape ici les types de fichiers dont on ne veut pas (**patrons**). Ex :

\*.[oa] # refus des fichiers finissant pas o ou a

\*~ : #refus des fichiers finissant par ~

\*.a # pas de fichier .a

!lib.a # mais suivre en version lib.a malgré la règle précédente

/TODO # ignorer uniquement le fichier TODO à la racine du projet

build/ # ignorer tous le fichiers dans le répertoire build

doc/\*.txt # ignorer doc/notes.txt, mais pas doc/server/arch.txt

## Commandes principales :

Pour voir les nouveaux fichiers, les fichiers modifiés… :

*git status*

Pour ajouter un fichier à l’index :

*git add [nom fichier]*

Pour ajouter automatiquement les fichiers déjà suivis en version (déjà ajouté à l’index, mais qui ont été modifiés entre temps sur la copie de travail) :

*git add -u*

Pour créer un commit :

git commit –m ‘nom commit’

git commit -a –m ‘nom commit’. Le rajout du ‘-a’ à la commande de commit que nous utilisons déjà permet d’indexer automatiquement les fichiers qui sont déjà en suivi de version. (Donc pas besoin de faire git add)

## Supprimer un fichier de l’index et de la copie de travail

git rm ‘nom fichier’ : supprime le fichier de l’index **et de la copie de travail.**

Si l’on se contente de supprimer un fichier de la copie de travail, le fichier ne sera pas supprimé de l’instantané de l’index.

Pour supprimer des fichiers de l’index (= arrêter le suivi de modification), sans supprimer de la copie de travail (si on a copié tout un projet eclipse et qu’on se dit que ce serait bien de ne copier que les sources par exemple) :

$ git rm --cached nom\_fichier

Nom\_fichier peut être un patron

## 