# GIT - Spis podstawowych komend

**git init** - stwórz puste repozytorium w aktualnym katalogu  
**git add** **<ścieżka>** - dodaje wszystkie zmiany ze *<ścieżka>* do „stagging area”  
**git commit** – tworzy commita ze zmianami aktualnie w "staging area"

**git status** – pokazuje, które pliki się zmieniły od ostatniego commita, a które zostały dodane lub usunięte i czy są lokalnie jakieś commity do wypushowania lub pobrania

**git branch <name>** - tworzy nowy branch o nazwie *<name>***git checkout <name>** - "przełącza się" na brancha *<name>*

**git reset --hard <commit>** - resetuje repozytorium do wersji *<commit>* oraz czyści working directory i staging area (czasami bezpowrotnie - zależnie od scenariuszu)  
**git reset --mixed <commit>** - resetuje repozytorium do wersji *<commit>*, pozostawia zmiany w working directory (jest to domyślny sposób resetowania)  
**git reset --soft <commit>** - resetuje repozytorium do wersji *<commit>*, pozostawia zmiany w staging area  
**git checkout <file>** - usuwa zmiany naniesione na plik *<file>***git clean** **<file>** **-f** - usuwa nie śledzony plik *<file>*, aby usunąć również katalog należy dodać przełącznik *-d*, aby tylko podejrzeć co ma zostać usunięte, przełącznik *-n*

**git rebase <target-commit>** - szuka commita, który był wspólnym przodkiem zarówno aktualnego brancha jak i <target-commit>, nakłada wszystkie commity, których nie ma w <target-commit> na górę <target-commit>  
**git merge <target-branch>** - tworzy commita mergującego, który ma dwóch przodków, ze wspolną historią tych dwóch branchy, gdy <target-branch> ma w historii aktualnego brancha to następuje "Fast Forward", czyli następuję dodanie commitów, ale nie tworzy się commit mergujący.  
**git rebase --interactive <commit>** - pozwala zamieniac ze sobą miejscami, łączyć usuwać, edytować commity od *<commit>* do aktualnego commita *(HEAD)*  
**git revert <commit>** - tworzy nowego commita, który odwraca zmiany z *<commit>*

**git push** – wysyła lokalne commity do zdalnego repozytorium  
**git fetch** - pobiera commity do lokalnego repozytorium (do branchy *remotes/nazwa\_remota/nazwa\_brancha*)  
**git pull** – wykonuje git fetch, następnie git merge  
**git pull** **--rebase** – wykonuje git fetch, następnie rebasuje lokalne commity, nakładając je na wierzch commitów pobranych ze zdalnego repozytorium  
 www.gitwarsztaty.pl