**Ls -la** -> list hidden files.

**Rm -rf .git** -> remove .git folder

**Git rm filename** -> it will **remove file** and automatically **add changes to staging area**

**Git status** – show any changes done on inventory/repository or any **changes done on a file**

**Git add** **filename**– Add files to **staging area** (to check and for confirmation before moving to commit)

**Git add –all** -> add all files to staging area

**Git commit** – take the **snapshot of changes done on file** after adding to staging area.

**git checkout master** – If the head is deattached from master branch, use this to attach again.

**git diff --name-only --cached --diff-filter=AM** -> you get the changes between the index and your current HEAD and Show only names of changed files. **--staged** is also available as an alias for **--cached** above in more recent Git versions.  
NB: This can be combined with **--diff-filter** to only show (filter) staged files that are added (A), modified (M) etc.

**git push -u origin master –** to push code on server

**git rm -f --cached filename** -> to unstage file/files (-f force unstage)

**git reset** **filename -**  to unstage any specific file

**git reset –** to unstage all file from staging area

**ls -la** – show hidden files.

**Nano filename** -> to open file

**Touch filename** -> to create new file

**Git diff** –   
if staged – difference between **working directory** and **staging area**  
if staged not staged - show difference between **working directory** and **last commit**

**Git diff –staged ->** to see the **changes done between staging area file** and **last commit**

**Git mv CurrenFileName NewFileName** -> rename file name and added to staging area.

**Git rm –cached filename** -> to untrack the file in repository (suppose suddenly we have to add the file in .gitignore and don’t wan’t to track further) **or forcefully untrack the file**