**GIT – Commands  
=================**

**1) To Set global user\_name**

Git Config --global user.name "MD\_Sabair"

**2) To set global user\_email**

Git Config --global user.email "MD\_Sabair@yahoo.com"

**3) Initiliaze git**

Git init

**4) To add files to tracking area**

git add file\_name

**5) To add multiple files to tracking area**

git add .

**6) To check Direcotry status**

git status

**7) To Commit files**

git commit -m "Message"

**8) To push changes to central**

git push -u origin master

**9) To add remote origin**

git remote add origin <git\_repo\_url>

**10) To view the remote origin**

git remote -v

**11) To make a copy from central to local**

git clone <git\_repo\_url>

**12) To pull the latest changes to local**

git pull <git\_repo\_url>

**13) list of branches**

git branch --list

**14) To create branch**

git branch <branch\_name>

**15) To checkout branch**

git checkout sabear

**16) To push code to specific branch**

git push -u origin <branch\_name>

**17) To view git logs**

git log

**18) To view last 3 logs**

git log -3

**19) To revert any previous commit**

git revert <commit\_id>

**20) To merge two branches**

git merge <branch\_name>

**21) Clone one particular branch**

git clone -b <branch\_name> <github\_repo\_url>

**22) Git stash is used to push files to stash memory**

git stash

**23) To view the stash list**

git stash list

**24) To get files out of stash memory**

git stash pop

**25) To get a specific file out of stash memory**

git stash pop stash{id}

**26) How to change the previous commit message**

git commit --amend

**27) To merger multiple commits into a sinlge commit.**

git rebase -i HEAD~n

**28) Merge Conflict**

GIt will raise merge conflict if we have identical files in both the branches.

**29) Ignore some files from tracking**

.gitignore

**30) To download the configuration changes.**

git fetch origin

**31)** **git stash**

Temporarily saves changes that are not ready to commit.

**32) git stash pop**

Applies the stashed changes back to the working directory and removes them from the stash.

**33) git cherry-pick**

Applies a specific commit from one branch to another.

**34) git diff**

Shows changes between commits, branches or working directory

**35) git revert**

Creates a new commit that undoes the changes of a specific commit.

git revert <commit-hash>