GIT basics

**git init:** initialize git empty repository in the current directory

**git init <directory path>:** Initialize git empty repository in the mentioned path

**git clone <URL>:** clone the remote repository to the local system

**git config --global user.email "<valid mail>":** add mentioned email in ~/.gitconfig

**git config --global user.name "<name of the user>":** add mentioned name in ~/.gitconfig

**git config --local user.email "<valid mail>":** add mentioned mail in config file in current local repository(.git/config)

**git config --local user.name "<name of the user>":** add mentioned name in config file in current local repository(.git/config)

**git config --system user.email "<valid mail>"**

**:** add mentioned email to system-wide etc/gitconfig

**git config --system user.name "<name of the user>":** add mentioned name to system-wide in etc/gitconfig

**git config –global –edit:** opens .gitconfig file to edit user details

**git config –local –edit:** opens config file to edit

**git config –system –edit:** opens /etc/gitconfig file to edit

**git status:** gives list of modified files which are ready to add to staging area and to commit

**git status –s:** gives list of modified file ready to and commit in one line

**git add <file>:** add mentioned file to staging area to commit

**git add <directory>:** add directory

**git . or git –all:** adds all modified files to staging area

**git log:** gives all commits history

**git log <n>:** gives history of latest n commits

**git log –oneline:** gives history of all commits in oneline

**git log -- <file>:** gives commits for mentioned file

**git log <commit id>:** gives all older commits till mentioned commit id

**git blame <file>:** gives commit id, author, and content in lines with numbers

**git blame –c <file>:** gives lines which are copied from other files, gives commit id, original author info if committed, if not committed it will show not committed yet

**git blame -m <file>:** gives lines which are copied from same file, gives commit id, original author info if committed, if not committed it will show not committed yet

**git blame –e <file>:** same as git blame<file> but instead of author name it will display mail

**git diff:** gives differences b/w staging area and working directory

**git diff –cached:**  difference b/w staging area and local repository

git branch: lists the branches

git branch <name>: creates new branch

git checkout <branch name>: switch to mentioned branch

**git checkout <branch name**>: switch to mentioned branch

**git checkout –b <branchname>:** creates new branch and checkout to it

**git branch –d <branch name>:** deletes mentioned branch but if the branch is not merged to it will not delete the branch

**git branch –D <branch>:** deletes the branch even it has unmerged changes

**git branch –m <new name>:** renames the current branch with <new name>

**git log branch1..branch2:** gives commits which there in branch2 but not in branch1

**git diff branch1..branch2:** gives difference of what is in branch2 but not in branch1

**git push <git url> --delete <branch>** or **git push <git url> :<branch>** deletes branch in the mentioned remote repository

**git remote:** lists the remote connections by variable name

**git remote –v:** lists the remote connections name and its URL as well

**git remote add <variable> <url>:** adds git remote url to a variable. We can use this variable name as a shortcut for the remote repo url

**git remote rm <variable>:** removes the name and corresponding URL

**git remote rename <old> <new>:** renames the old remote name with new name

**git push <url>:** push the local commits to mentioned remote repo

git fetch <url>: fetches remote repo to local repo without changing the local changes

git merge <variable>/<branch>: merges remote branch into current branch to make it up-to-date

git merge <branch>: merges branch history to current branch

git merge <source> <target>: merges source branch into target branch

git merge –abort: stops merge and reverts back to pre-merge state

git mergetool: view the merge conflicts in tool

git pull: fetch and merge commits from the remote to working directory

git pull <url>: fetch and merge the commits from remote to current branch