# GIT

good morning gentlemen my name is scott today i'm gonna say some basics and commands of git lets get started

**GIT**:

simpleterm:A command-line session showing repository creation, addition of a file, and remote synchronization

**repo---->**Git stores this information in a data structure called a repository.

**practical approach**:to understand this we need to first understand difference between version control system and distributed control system

**version control systems**---->Version control systems are a category of software tools that help a software team manage changes to source code over time. Version control software keeps track of every modification to the code in a special kind of database. If a mistake is made, developers can turn back the clock and compare earlier versions of the code to help fix the mistake while minimizing disruption to all team members.

**difference between central version control system and distributed control system**:

**--->** central control system is located in the one place we can access the information and in that particular server like files,repos but we can access the only when only it is online i.e no remote repos and if it is offline we can not access anything

**---->**distributed version control system we can have local remote repos we can have all information about the remote repos so in this condition if the server is offline we can do our work by using local repo every person has copy of that repo i.e reason it was call distributed

**--->**GIT uses distributed version

Ok now lets see some commands of git

Check version---->**git --version**

To config username---->**git config --user.name"name"**

To config email id----> **git config --user.email "email id"**

To get all configuration values---->**git config --list**

help ---->**git config --help**

**-------->** particularly in we should pay attention in the two categories i.e project that has in your local machine or it was remotely to start developing it

**-------->**let's get started to do this first we should initialize the empty repo in the local repo directory hidden file i.e (.git file)

To initialize**----> git init**

To see files and status of local machine projects**---->git status**

To ignore the files we should create the ignore file(i.e .ignore file) and add particular file that we want to ignore

**------>**to work with the git we should know the three stages those are

- **-> working directory**
- **->staging area**
- **->repo(.git directory)**

**------->**so we are now to add files in the working directory that was our project in our machine first we need to add those to staging area and the we create repo

To add file to staging area all files**--->git add -A**
To remove all files in staging area**--->git reset filename**
To remove everything from staging area**---->git reset**
**------->**remember that when you done these things every time you need to see status
**------->**after adding these files we need to update in the form of message that we call git environment is commit in locally
To commit**--->git commit -m "particular message you update"**
To see the commit that we made**--->git log**
To get remote repo i.e from git get files that we want to**--->git clone url .**
To view those repo files after clone **--->git remote -v**
To see the the files that are modified**--->git diff**
**-------->**everything upto we seen locally now lets see how we can get updated that everyone can see those changes publicly to do this we have two commands called **push** and **pull** and also we have some movable pointers called **branch**
To update project or our file**------>git push reponame branchname**
To make modified when modified when push req in git**----->git pull reponame branchname**
**--------->**before we need to do this all above commands we need to create branch
To create new branch**--->git branch branchname**
To see the available branch locally**--->git branch**
**--------->**now we created the branch we need to push
To push**--->git push -u reponame branchname**
u**--->says that connect to the remote and local branches**
To see the available branches**---> git branch -a**(local and remote)
To check local branch and switch to new branch**--->git checkout branchname**
**--------->**so far we know create branch we need to know how to switch to another branch and save those changes to the our present branch and how to delete those branches after working complete.
To see the what we are current branch**--->git branch --merged**
To switch and save those changes to our current branch**--->git merge branchname (**not current branch name)
So now we merged which means that other branch information was stored as in current branch
So after this we need to push
**--------->**so now we created branch for our working purpose so now we need to del branch after finish working locally and remote
To del the branch locally**--->git branch -d branchname**
To del branch remotely**--->git push reponame --delete branchname**
**--------->**now so far we learned that added repo and updated remotely and now what if we wanted to undo the changes that we are made locally we use the **stash** command
To undo the changes beginning of the file**--->git stash save "messgae"**
To list stash**--->git stash list**
To get back changes that we made**--->git stash apply particularstash**(stash{0})
To get apply first stack of the list**--->git stash pop**
If we wanted to undo the particular stash in stash list**--->git drop particularstash**

If wanted to rid of all stashes don't want anything**---->git stash clear**
**Staging overview:**
To unstage all files**--->git reset**
To stage the files in particular directory**--->git add directory/**
If you wanted to add only modified files not new files**--->git add -u**
You can use this command for particular directory also
To add stage all files**--->git add .**(but we need to specify the directory first)