1) Go to desktop and create a folder gitws/demoapp

2) open terminal with the demoapp folder

3) Initialize a local repo => git init

4) git status => to check your git repo if any files yet to be staged

5) git log => to see the sll the updates made since the file was created

6) create a text file master1.txt with a line

7) execute git status it shows this file in red

8) git add master1.txt => will move it to staging area , check using git status

9) git commit -m "First commit to master1.txt"

10) then check git status, git log and to display log in one line use

git log --oneline

11) make changes in master1.txt and say git add/commit. now 2 commits done to the git repo

12) then create a new empty file master2.txt and say git status will show one file remaining to move to staging area

13) create a new file .gitignore and add the file name master2.txt in thwe file .gitignore

14) git status => will show only .gitignore and not master2.txt

15) then git add . => git commit -m and now 3 commits done to the git repo

16) git ls-files will show the list of files currently tracked by git

17) git show <commit\_id> details of a commit

Parallel Development

1) git branch

2) git branch <branchname> will create a new branch : git branch branch1

3) git branch -d branch2 => to delete a branch

4) git branch -m branch1 new\_branch1 => to rename

Remote Repo

1. Git remote => to check if local connected with remote
2. Git remote -v => to check remote url
3. Git remote add origin <git url > to connect local and remote if not already added
4. Git push origin master => to merge local repo changes with remote repo
5. Git pull origin master => to merge remote repo changes with local repo
6. Git clone => to create a local copy of the original
7. Fork => to create a remote copy of the original