**Git** is version control used to track changes in software development project. Each version is a snapshot. Git and Atom are developed by Github company.

Atom is text editor used to manage all the files in a project.

$mkdir myapp

Myapp$touch index.html

Myapp$touch app.js

Myapp$git init ---create new git empty repository environment to manage the project ‘myapp’.

Myapp$git config –global user.name ‘Mike Nguyen’ --take ownership on Git

Myapp$git conifg --global user.email ‘nguyenmikeson@gmail.com’

Goto Atom >open project> select myapp folder. Now, Git will monitor

Every changes in index, app files.

Myapp$git add index.html -- first, need register file to Git repository ‘stage area’, but not commit yet.

Myapp$git rm –cached index.html -- if want rm from Git stage area

Myapp$git status --see all files in stage area or Git repository, ready for commit

Goto Atom> try to change java or html script and see what is happen

Myapp$git status – show scripts have been changed in working directory, not yet move to Git repository. If scripts are OK, I need update it to git repository by

Myapp$git add . – add all my change to stage area.

Myapp$git status -- my modified scripts is add to git repository, ready to commit and upload to Github.com for everyone use.

Myapp$git commit ..

Myapp$git status --every thing is clean now. Project completed

--- change again –

Myapp$git commit –m ‘Changed something in app.js MikeNguyen’ – skip edite ‘vi’ text

---New topic: I want ignore some files in Git stage area

Myapp$touch .ignoremike

$ git status

On branch master

Untracked files:

(use "git add <file>..." to include in what will be committed)

.ignoremike

Myapp$touch log.txt --I don’t want this file is monitored by Git system.

Open .ignoremike> enter file name: log.txt

Myapp$git add . – only .ignoremike add to repository not log.txt. we can ignore whole dir/

--

**Branch:** I create my own branch because there are many guys focus on same project.

Myapp$git branch mybranch

Myapp$git commit –m ‘ok’

Myapp$git checkout mybranch --switch to my branch

Mybranch$touch login.html --in index.html insert this file FORM.

Try to change something more in index.html file of Master branch.

Mybranch$git add .

Mybranch$git commit --everything is committed in my own branch. Nothing affect Master branch

Mybranch$git checkout master –return to Master branch. I see the index.html file is not changed. It keeps the previous version.

---After modifying in my own branch, I want to merge or upload back to Master

Myapp$git merge mybranch

Now, in Master branch, the software is update and available for everyone

**Github.com**

Is public script repository for everyone.

<https://github.com> > create new repository to upload scripts

myapp$git remote --nothing in remote server now

myapp$git remote add origin <https://github.com/mikesonnguyen9/myappsample.git>

myapp$git remote --now I use remote server to store scripts

myapp$**git push** –u origin master --All my *existing local directory* ‘myapp’ will be push to github server. Everyone in world can see ‘myapp’

Counting objects: 17, done.

Delta compression using up to 4 threads.

Compressing objects: 100% (11/11), done.

Writing objects: 100% (17/17), 1.47 KiB | 250.00 KiB/s, done.

Total 17 (delta 3), reused 0 (delta 0)

remote: Resolving deltas: 100% (3/3), done.

To https://github.com/nguyenmikeson9/myappsample.git

\* [new branch] master -> master

Branch 'master' set up to track remote branch 'master' from 'origin'.

On Github server I can see all my directory and files.

I continue working from local machine, but result will be sent to remote Master Github

Owner@Laptop2 MINGW32 ~/Desktop/myapp (master)

$ git status

On branch master

Your branch is up to date with 'origin/master'.

myapp$touch README.md --save

myapp$git status

myapp$git add .

myapp$git commit –m ‘I added readme’

myapp$git push ---upload to github server

My repository on github server is complete. Now, world can download my scripts for free. Ex:

Create on local machine ‘Testapp’ > right click > open Git Bash shell cmd

Testapp$**git clone** https://github.com/nguyenmikeson9/myappsample.git

I can clone/copy any public project from any person around the world in Github server.

Many developers pull the modified project from Github server

Myapp$git pull --from original server pull the project from Github