DEFINITION

1. Source control/version control definition

-The system helps to store all changes of source code.

-Support multiple people working at the same time.

-See who changes the code.

-Revert the changes, bring the code back to an older version, without worrying about losing the code.

2. GIT Definition

-GIT is free and open source distributed version control system designed to handle everything from small to very large projects with speed and efficiency.

3.GITHUB definition.

- GitHub is a project version and code management system that works like a social network for developers. Programmers can clone the source code from a repository and Github is a public repository server service, each person can create accounts on it to create their own repositories to be able to work.

HOW TO USE GIT

To use GIT, you need to remember three important concepts: Repo, commit, branch.

1.Repository

2.Commit

-Note that you have to explicitly tell Git which changes you want to include in a commit before running the "git commit" command. This means that a file won't be automatically included in the next commit just because it was changed. Instead, you need to use the "git add" command to mark the desired changes for inclusion.

- Also note that in Git (not like in Subversion), a commit is not automatically transferred to the remote server. Using the "git commit" command only saves a new commit object in the local Git repository. Exchanging commits has to be performed manually and explicitly (with the "git fetch", "git pull", and "git push" commands).

3.Branch

- Branches are what are used to branch and record the flow of history. Branched branches will not affect other branches so you can make multiple changes simultaneously in the same repository.

-Moreover, branched branches can be edited into a branch by merging with another branch.

How to use Git basic

-Step 1: Create Github account.

* Create new account on GitHub web.

-Step 2: Create a new repository

* Create new repo on GitHub web.

-Step 3: create a file.

* Cd to directory(folder) that was used to git.
* Comment git init to create repo in local.
* Comment git status to check files are add or not.
* Comment git add <file> (add each file) or <.> (add all file).
* Comment git status again to check files are add or not.

-Step 4: make a commit.

* Comment git commit -m “message”.
* Comment git log to see who commit.

-Step 5: Connect your GitHub repo with your computer

* Comment git remote add origin <url link>.
* Comment git push -u origin master to push commit from local to server repo.

How to work with GIT advance.

Branch:

-Step 1: Git branch

* Comment git branch <branchname> to create new branch.
* Comment git branch to see all the branch and \* is a current branch.

-Step 2: Git checkout

* Comment git checkout <Branch> to work with branch was checkout.

-Step 3: Git merge

* Comment git branch again to see all branch and \* is a current branch.
* Comment git checkout <Branch> to work with branch was checkout.
* Comment git merge <another Branch > to merge all branch to branch master.

-Step4: Delete branch:

* Comment git branch -d < branchname > to delete branch.
* Comment git branch to check branch is deleted or not.