Introduction to GIT

GIT:

· Git is the open source distributed version control system that facilitates GitHub activities on your laptop or desktop.

· Version Control system is a platform where the developers work together and maintain their history of the work.

Installing GIT:

· Just go to the link http://git-scm.com/download/win and download will automatically starts.

· Another way to install git is by using git hub includes a command line version of Git as well as GUI. You can download it by using http://windows.github.com link.

· This will install github in your system.

Creating an Account:

· Once git is installed we create a git account by using https://github.com/ link.

Configure Tooling:

· Set the git username:

git config --global user.name "[name]"

· Set the email you want to attach to your commit transactions

git config --global user.email "[email address]"

Create Repositories:

· Create a new local repository with specified name

git init [project-name]

· Downloads a project and its entire version history

git clone [url]

Make Changes:

· List all modified or new files to be commited.

git status

· Shows file differences not yet staged

git diff

· Snapshots the file in preparation for versioning

git add [file]

· Shows the file difference between staging and last file version.

git diff –staged

· Records file snapshots permanently in version history

git commit –m "[descriptive message]"

Group Changes:

· Lists all local branches in the current repository

git branch

· Creates a new branch

git branch [branch-name]

· Switches to the specified branch and updates the working directory

git checkout [branch-name]

· Combines the specified branch's history into the current branch

git merge [branch]

· Deletes the specified branch

git branch –d [branch-name]

Refactor Filenames:

· Deletes the file form the working directory and stages the deletion

git rm [file]

· Removes the file from version control but preserves the file locally

git rm --cached [file]

· Changes the file name and prepare it for commit

git mv [file-original] [file-renamed]

Save Fragments:

· Temporarily stores all modified tracked files.

git stash

· Restores the most recently stashed files

git stash pop

· Lists all stashed changesets

git stash list

· Discards the most recently stashed changeset

git stash drop

Review History:

· Lists version history for the current branch

git log

· Lists a version history for a file, including renames

git log –follow [file]

· Shows content difference between two branches

git diff [first-branch]…[second-branch]

· Outputs metadata and content changes of the specified commit

git show [commit]

Redo commit:

Undoes all commits after [commit],preserving changes locally

git reset [commit]

· Discards all history and changes back to the specified commit

git reset –hard [commit]

Synchronize Changes:

· Downloads all history from the repository bookmark

git fetch [bookmark]

· Combines bookmarks branch into current local branch

git merge [bookmark]/[branch]

· Uploads all local branch commits to github

git push [alias] [branch]

· Downloads bookmark history and incorporates changes

git pull