Aim: Study of any version control tool.

Theory:

* What is version control tool?

Version control (also known as revision control, source control, or source code

management) is a class of systems responsible for managing changes to computer

programs, documents, large web sites, or other collections of information. Version

control is a component of software configuration management.

* Type of version control tool?

1. Local Version Control Systems

It is one of the simplest forms and has a database that kept all the changes to files under revision control. RCS is one of the most common VCS tools. It keeps patch sets (differences between files) in a special format on disk.

1. Centralized Version Control Systems

In a centralized version control system (CVCS), a server acts as the main repository which stores every version of code. Using centralized source control, every user commits directly to the main branch, so this type of version control often works well for small teams, because team members have the ability to communicate quickly so that no two developers want to work on the same piece of code simultaneously.

1. Distributed Version Control Systems

In software development, distributed version control is a form of version control in which the complete codebase, including its full history, is mirrored on every developer's computer. Compared to centralized version control, this enables automatic management branching and merging, speeds up most operations, improves the ability to work offline, and does not rely on a single location for backups. Git, the world's most popular version control system, is a distributed version control system.

* Advantage of version control tool?

1. Track of All the Modifications Made to the Code
2. Managing and Protecting the Source Code
3. Comparing Earlier Versions of the Code
4. Supports the Developers’ Workflow and Not any Rigid Way of Working

Tool:

* What is Git?

Git is software for tracking changes in any set of files, usually used for coordinating work among programmers collaboratively developing source code during software development. Its goals include speed, data integrity, and support for distributed, non-linear workflows (thousands of parallel branches running on different systems).

Git was originally authored by Linus Torvalds in 2005 for development of the Linux kernel, with other kernel developers contributing to its initial development.[11] Since 2005, Junio Hamano has been the core maintainer. As with most other distributed version control systems, and unlike most client–server systems, every Git directory on every computer is a full-fledged repository with complete history and full version-tracking abilities, independent of network access or a central server.[12] Git is free and open-source software distributed under the GPL-2.0-only license.

* Installation process

<https://git-scm.com/download>

* Git commands

1.Configuration

git config --global user.name “yourname”

git config --global user.email “youremailid”

git config --list

2. Navigation to working directory

cd (drag your outer folder)

cd (drag inner folder)

3.Initialization to GIT

git init

4.Create files using git command

touch “filename.exe”

5.Add file to staging area to commit after modification

git add filename.exe

or

git add –A //to stage all files in that folder

6.Commit

git commit –m “message”

or

git commit –a //to commit all files in staging area

7.To un-stage file from staging area

git reset HEAD FILE-TO-UNSTAGE

8.Check Log of each commit

git log

or

git log --oneline //this command display logs with small commit code

9.Display each commit/ changes done to file

git show commit-hash //copy-paste hash function of commit

10.Retrieve deleted file

git checkout HEAD filename.exe

or

git checkout -- .

or

git checkout commit-hash~1 – filename.extension

11.Check Status

git status

12.Rollback to previous version

Reset

git reset --hard commit-hash

Revert

git revert commit-hash

Note: when the revert enter message window is displayed, please press escape tab, then

:wq and enter.

13.Create branch (new feature)

git branch branchname

14.Delete branch

git branch –D branchname

15.To switch between branch

git checkout branchname

16.To merge branches

git merge branchname

17.Show branch list

git branch –list

GIT commands to save file on central repository

1. Create account on GITHUB.com
2. Click on ‘+’ on top right corner in your account and create new repository
   1. Add repository name
   2. Description (optional)
   3. Repository visibility -> public
   4. Check box for initialize this repository with Read Me
   5. Click create repository button
3. Top of your GITHUB repository click on ‘CODE ‘ button to select http code
4. In GIT bash window (type command)
   1. git remote add origin http-URL // to connect to remote github server
   2. git remote –v // to verify connection
   3. git remote remove origin // to remove remote server
5. PUSH Command
   1. git push origin master // will display username and password window.
6. PULL Command
   1. git pull origin master