Experiment No 2

Aim : **To understand Version Control System, Git installation & Github account.**

Lab Outcome :
LO1: To understand the fundamentals of DevOps engineering and be fully proficient with DevOps terminologies, concepts, benefits, and deployment options to meet your busines requirements.
LO2 : To obtain complete knowledge of the "version control system" to effectively track changes augmented with Git and Github.

Theory :

What is Version Control?
Version control, also known as source control, is the practice of tracking and managing changes to software code. Version control systems are software tools that help software teams manage changes to source code over time. As development environments have accelerated, version control systems help software teams work faster and smarter. They are especially useful for DevOps teams since they help them to reduce development time and increase successful deployments.

Version control software keeps track of every modification to the code in a special kind of database. If a mistake is made, developers can turn back the clock and compare earlier versions of the code to help fix the mistake while minimizing disruption to all team members.

What is Git?
Git is a specific open-source version control system created by Linus Torvalds in 2005. Specifically, Git is a distributed version control system, which means that the entire codebase and history is available on every developer's computer, which allows for easy branching and merging.

Instead, version control lets developers safely work through branching and merging. With branching, a developer duplicates part of the source code (called the repository). The developer can then safely make changes to that part of the code without affecting the rest of the project.
Then, once the developer gets his or her part of the code working properly, he or she can merge that code back into the main source code to make it official.

To understand GitHub, you must first have an understanding of Git. Git is an open-source version control system that was started by Linus Torvalds—the same person who created Linux. Git is similar to other version control systems—Subversion, CVS, and Mercurial to name a few.

So, Git is a version control system, but what does that mean? When developers create something (an app, for example), they make constant changes to the code, releasing new versions up to and after the first official (non-beta) release.

Version control systems keep these revisions straight, storing the modifications in a central repository. This allows developers to easily collaborate, as they can download a new version of the software, make changes, and upload the newest revision. Every developer can see these new changes, download them, and contribute.

Similarly, people who have nothing to do with the development of a project can still download the files and use them. Most Linux users should be familiar with this process, as using Git, Subversion, or some other similar method is pretty common for downloading needed files—especially in preparation for compiling a program from source code (a rather common practice for Linux geeks).

Git is the preferred version control system of most developers, since it has multiple advantages over the other systems available. It stores file changes more efficiently and ensures file integrity better.
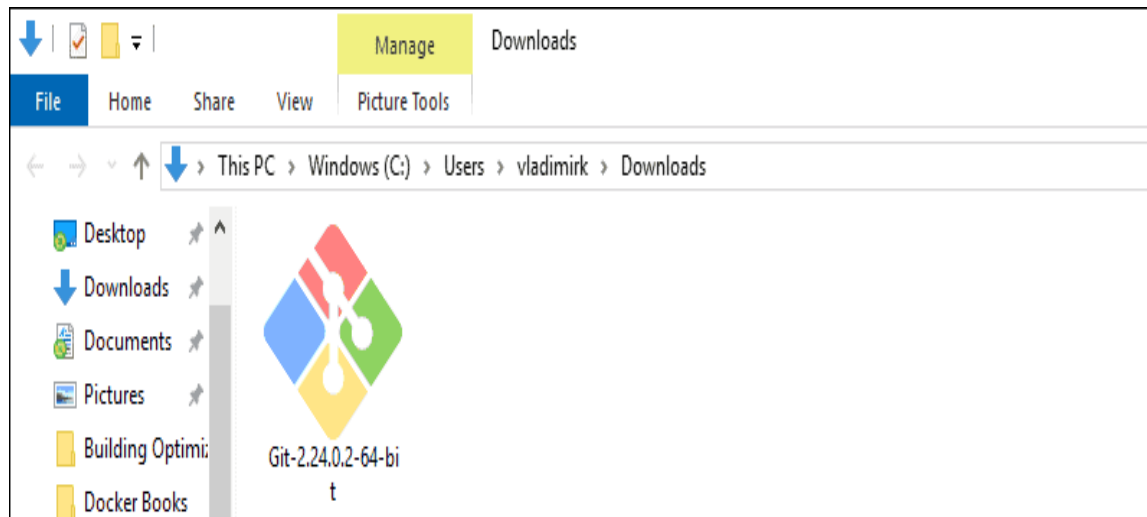
Git Installation in Windows:

1. Browse to the official Git website: https://git-scm.com/downloads
2. Click the download link for Windows and allow the download to complete.
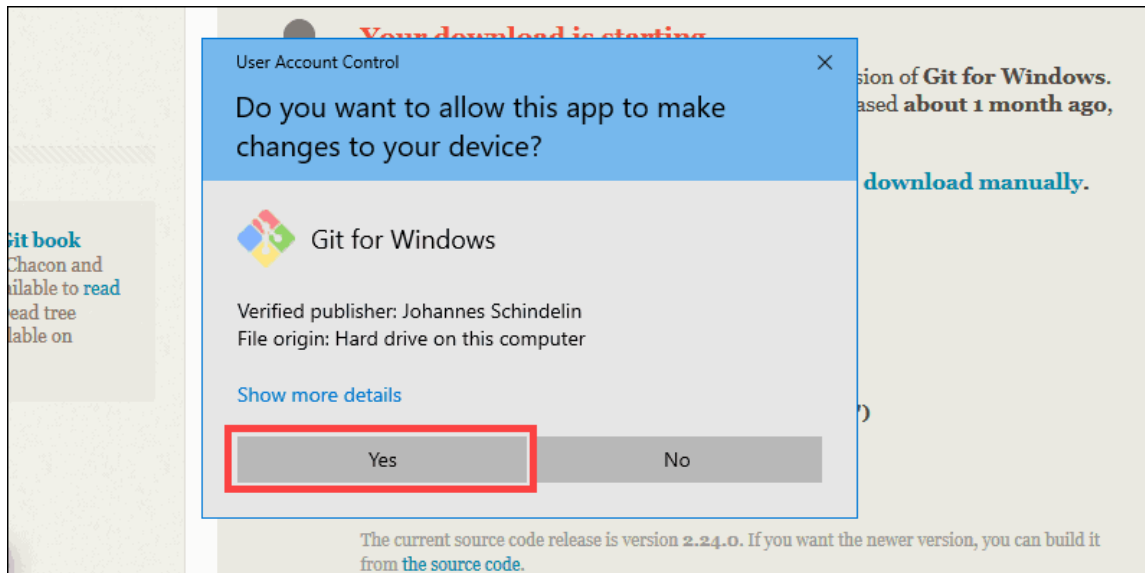
Extract and Launch Git Installer

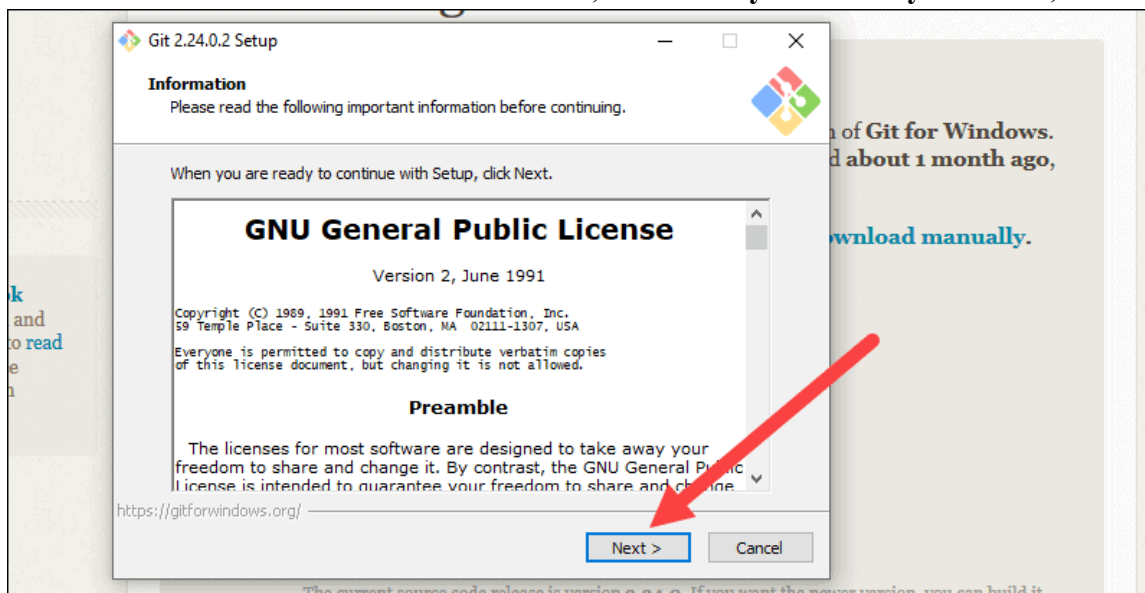**3. Browse to the download location (or use the download shortcut in your browser). Double-click the file to extract and launch the installer.**
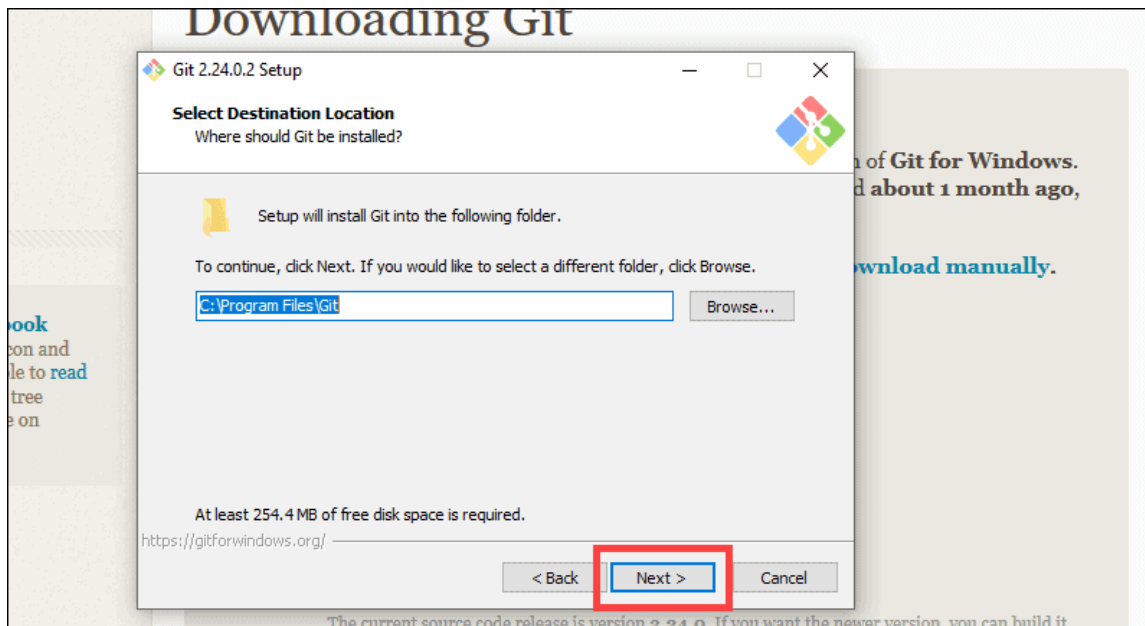

4. Allow the app to make changes to your device by clicking Yes on the User Account Control dialog that opens.
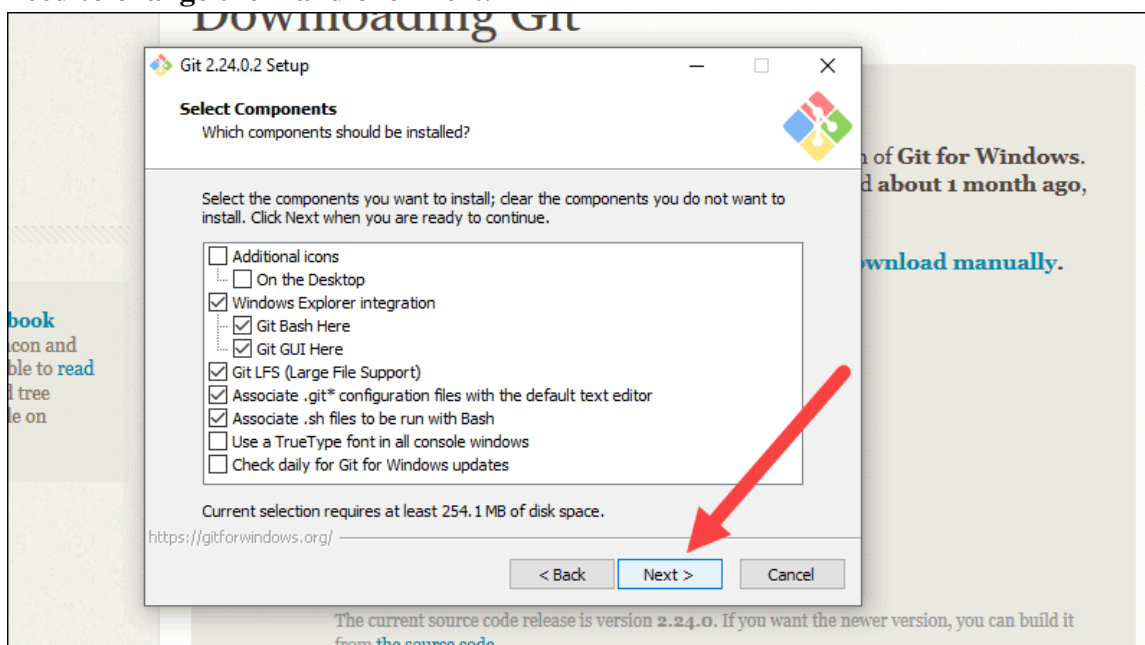
5. Review the GNU General Public License, and when you're ready to install, click Next.
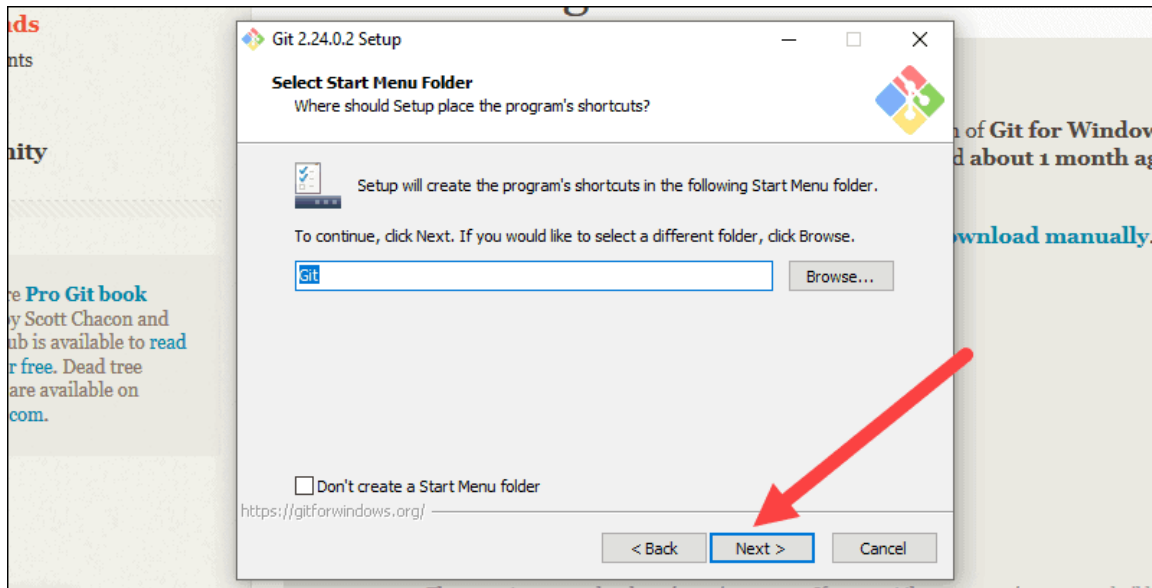

6. The installer will ask you for an installation location. Leave the default, unless you have reason to change it, and click Next.
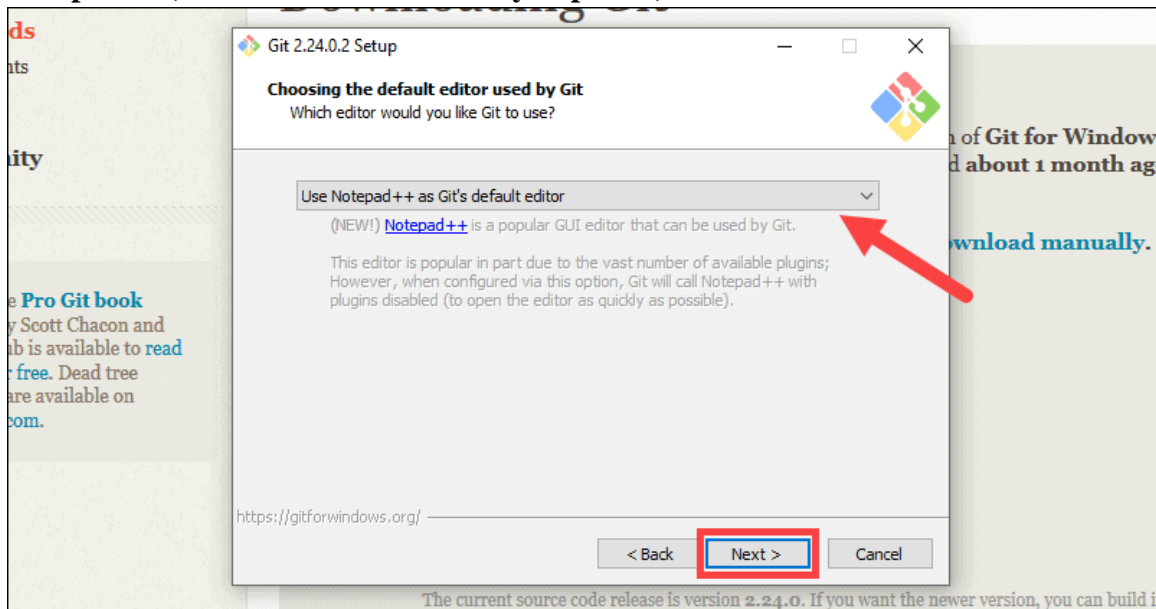
7. A component selection screen will appear. Leave the defaults unless you have a specific need to change them and click next.


**8. The installer will offer to create a start menu folder. Simply click Next.**

9. Select a text editor you'd like to use with Git. Use the drop-down menu to select Notepad++ (or whichever text editor you prefer) and click Next.


10. The next step allows you to choose a different name for your initial branch. The default is 'master.' Unless you're working in a team that requires a different name, leave the default option and click Next.
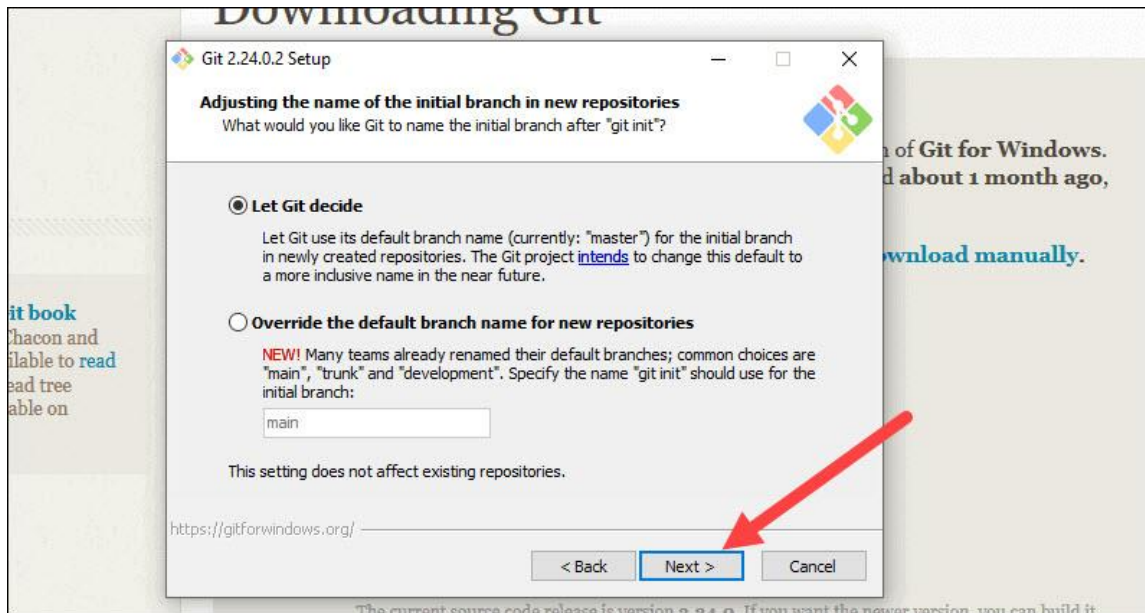
11. This installation step allows you to change the PATH environment. The PATH is the default set of directories included when you run a command from the command line. Leave this on the middle (recommended) selection and click Next.
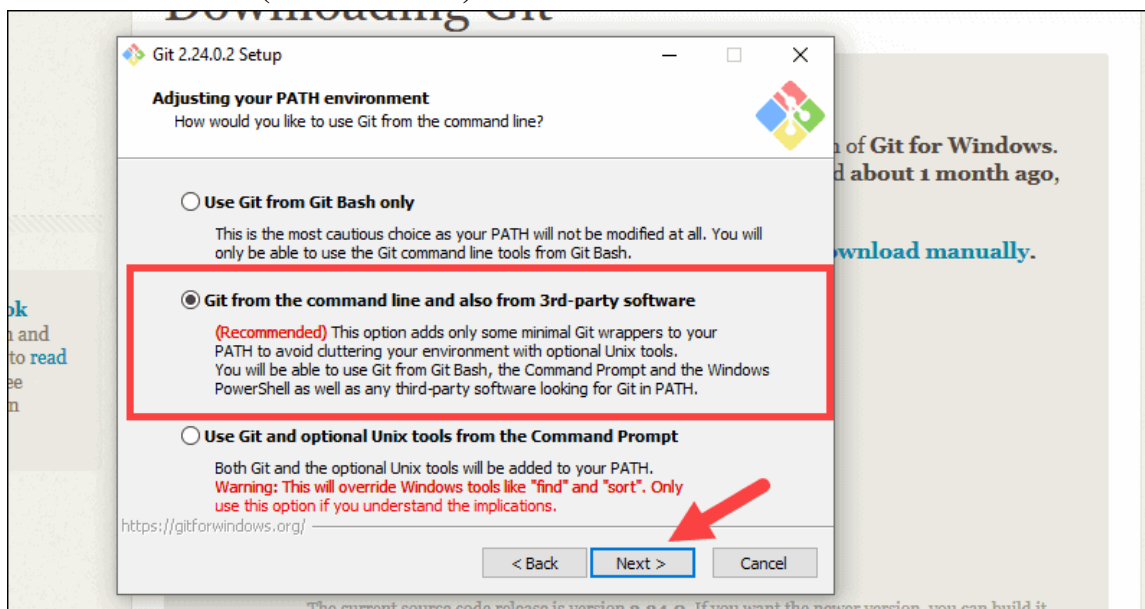

12. The installer now asks which SSH client you want Git to use. Git already comes with its own SSH client, so if you don't need a specific one, leave the default option and click Next.
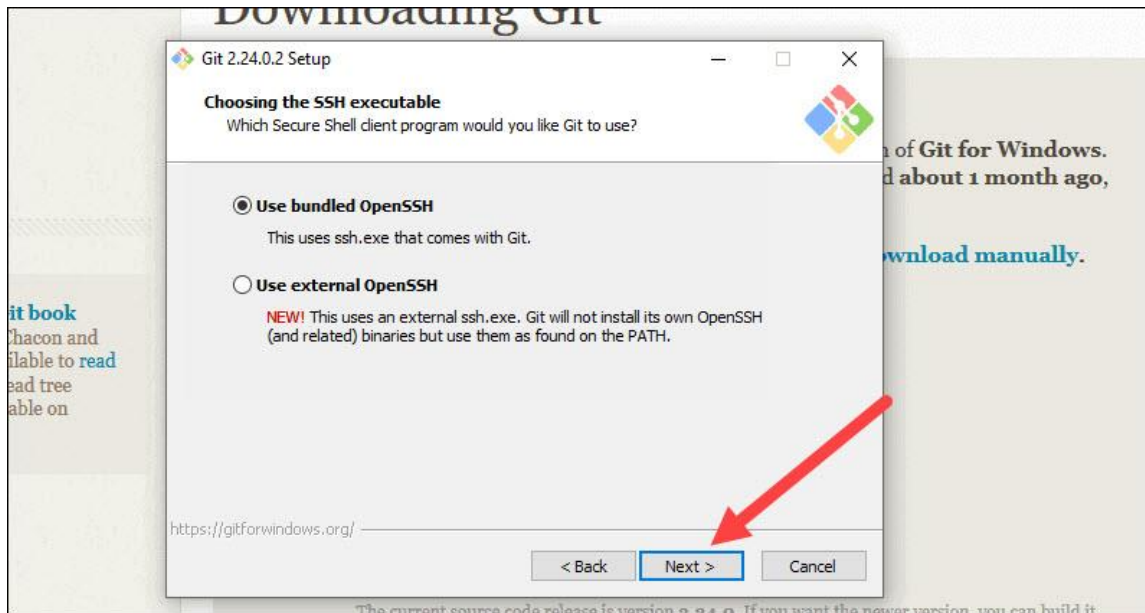
13. The next option relates to server certificates. Most users should use the default. If you're working in an Active Directory environment, you may need to switch to Windows Store certificates. Click Next.
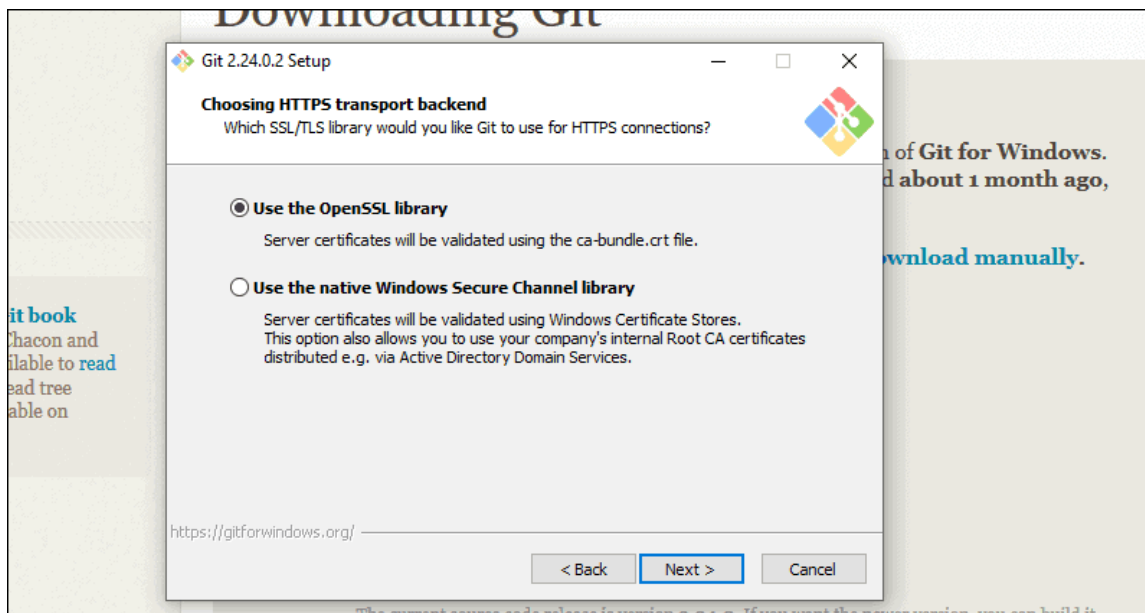
14. The next selection converts line endings. It is recommended that you leave the default selection. This relates to the way data is formatted and changing this option may cause problems. Click Next.
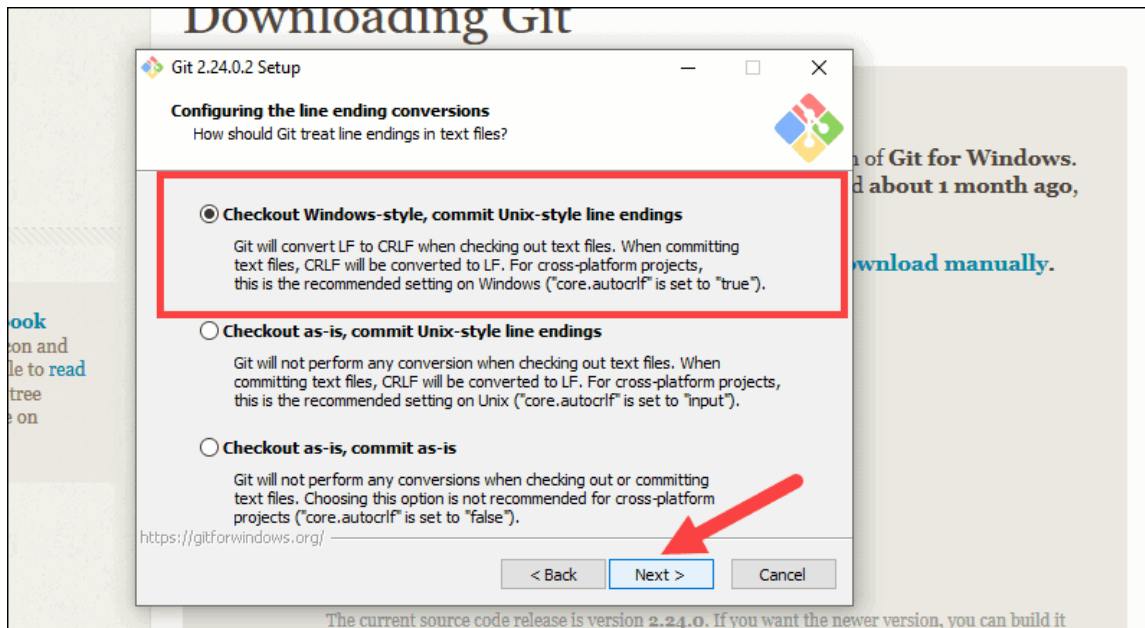
15. Choose the terminal emulator you want to use. The default MinTTY is recommended, for its features. Click Next.
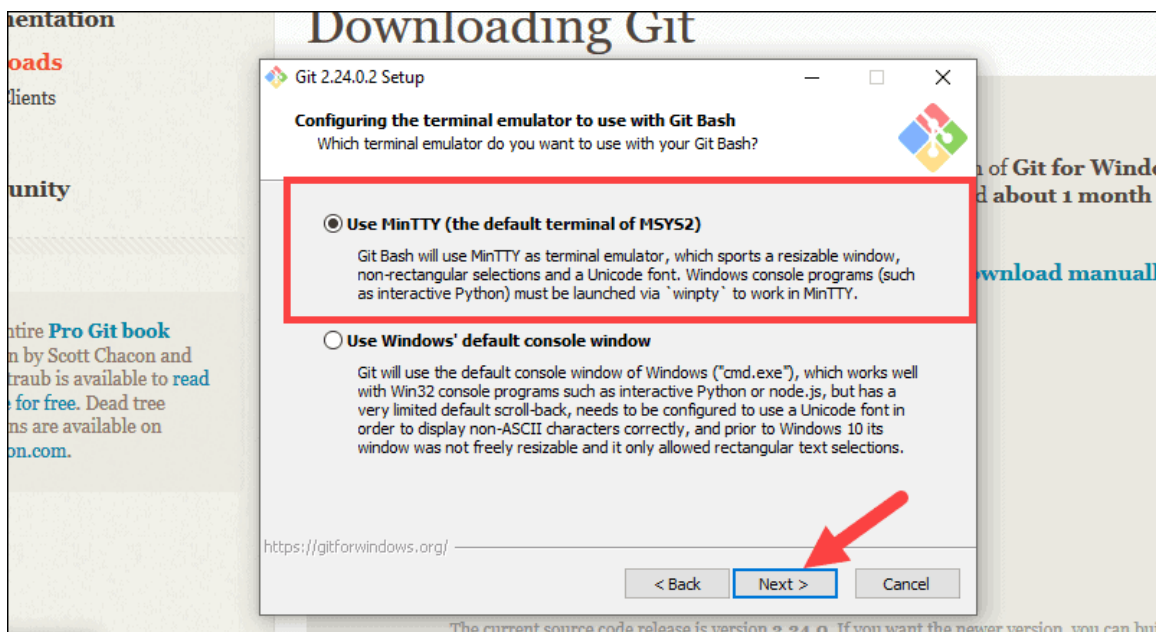

16. The installer now asks what the git pull command should do. The default option is recommended unless you specifically need to change its behavior. Click Next to continue with the installation.
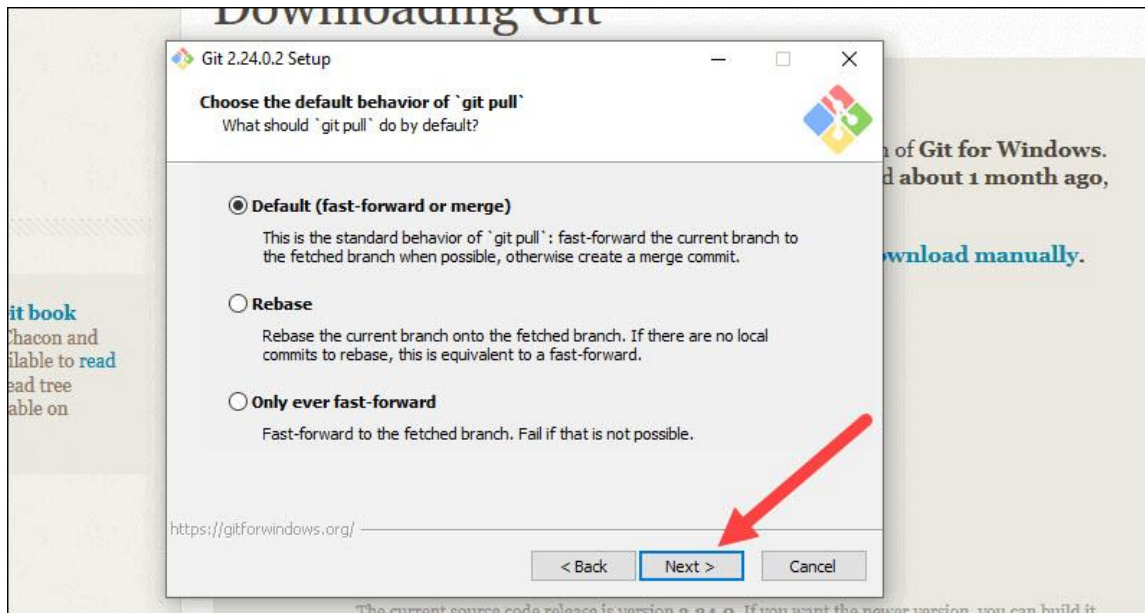
17. Next you should choose which credential helper to use. Git uses credential helpers to fetch or save credentials. Leave the default option as it is the most stable one, and click Next.
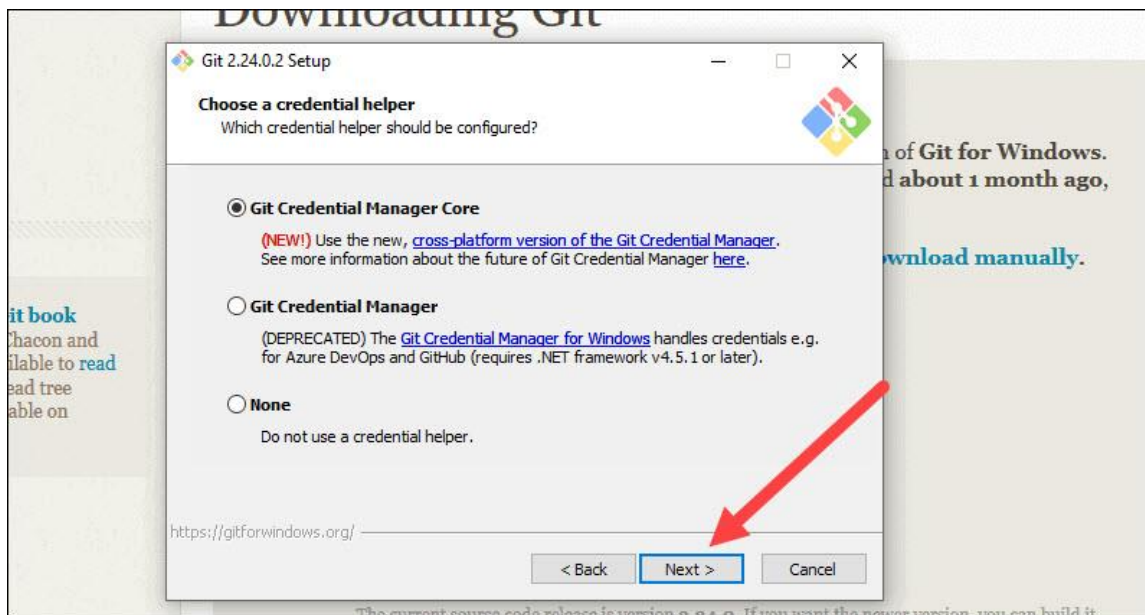
18. The default options are recommended, however this step allows you to decide which extra option you would like to enable. If you use symbolic links, which are like shortcut
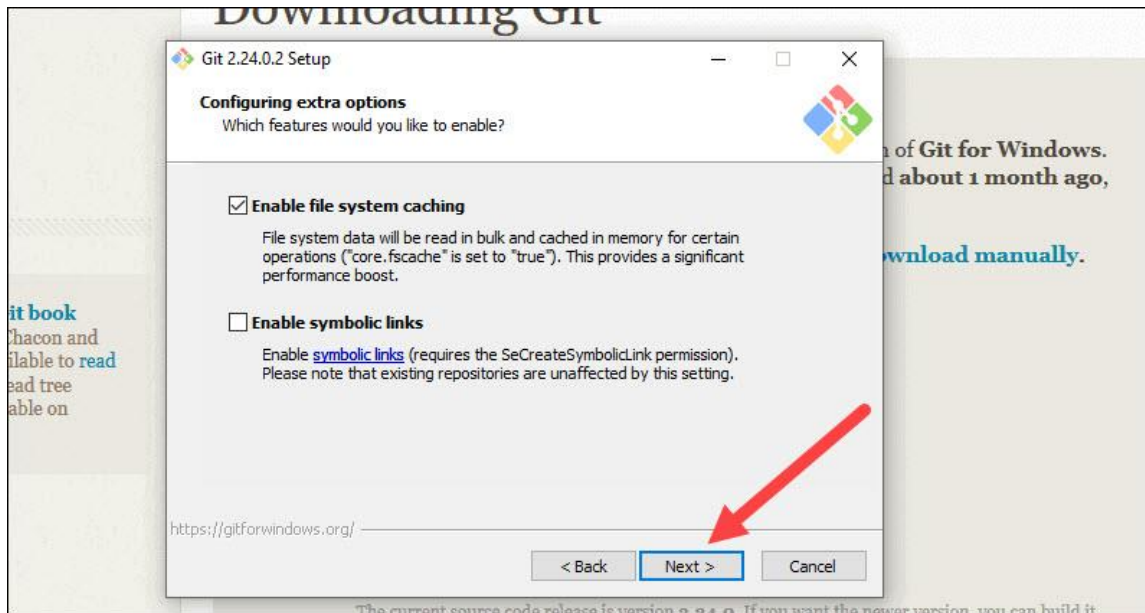
19. Depending on the version of Git you're installing, it may offer to install experimental features. At the time this article was written, the options to include support for pseudo controls and a built-in file system monitor were offered. Unless you are feeling adventurous, leave them unchecked and click Install.
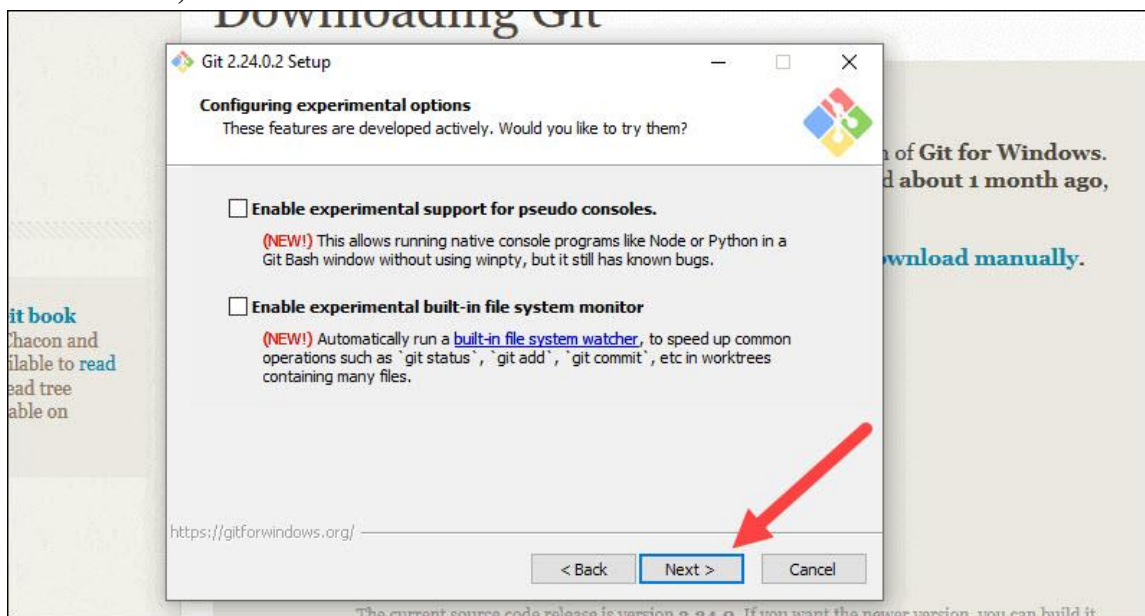

20. Once the installation is complete, tick the boxes to view the Release Notes or Launch Git Bash, then click Finish.

What is Github?

GitHub is a web-based version control and collaboration platform for software developers. Microsoft, the biggest single contributor to GitHub, acquired the platform for $7.5 billion in 2018. GitHub, which is delivered through a software as a service (SaaS) business model, was started in 2008. It was founded on Git, an open source code management system created by Linus Torvalds to make software builds faster.
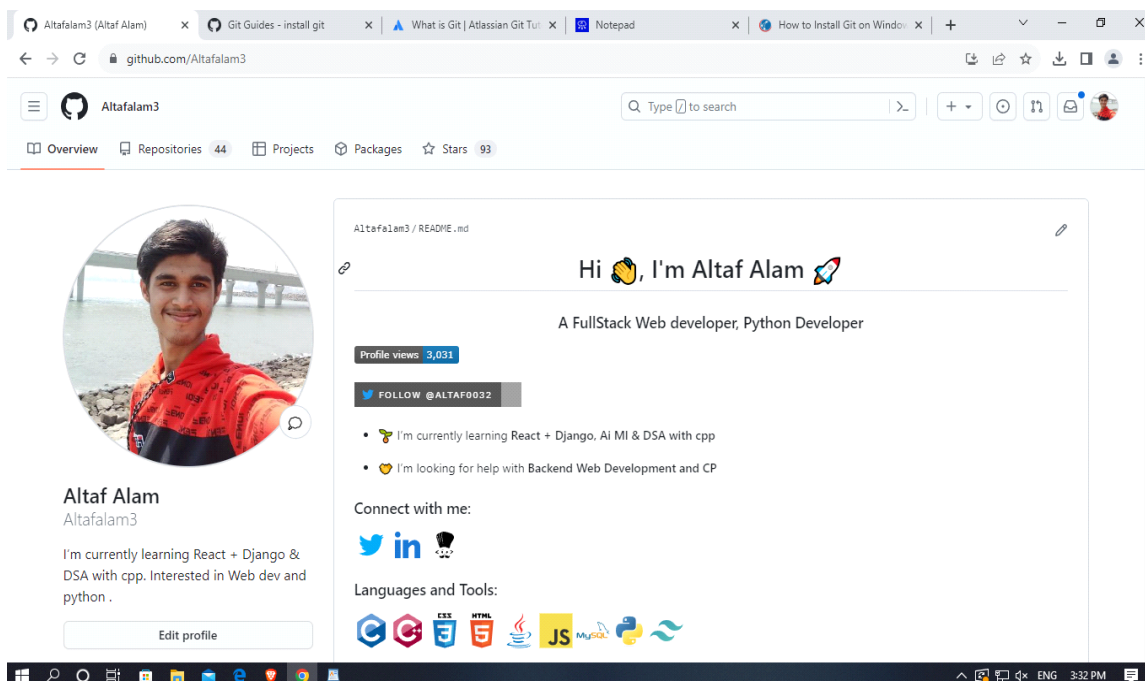
GitHub allows developers to change, adapt and improve software from its public repositories for free, but it charges for private repositories, offering various paid plans. Each public and private repository contains all of a project's files, as well as each file's revision history. Repositories can have multiple collaborators and can be either public or private.

GitHub facilitates social coding by providing a hosting service and web interface for the Git code repository, as well as management tools for collaboration. The developer platform can be thought of as a social networking site for software developers. Members can follow each other, rate each other's work, receive updates for specific open source projects, and communicate publicly or privately.

Conclusion :
Thus we have understood about control version, installation of git and creation of github account.