Experiment No : 02

Aim : To understand Version Control System / Source Code Management, install git and create a GitHub account.

Lab Outcomes : 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 is a system that records changes to a file or set of files over time so that you can recall specific versions later. For the examples in this book, you will use software source code as the files being version controlled, though in reality you can do this with nearly any type of file on a computer.

If you are a graphic or web designer and want to keep every version of an image or layout (which you would most certainly want to), a *Version Control System (VCS) is* a very wise thing to use. It allows you to revert selected files back to a previous state, revert the entire project back to a previous state, compare changes over time, see who last modified something that might be causing a problem, who introduced an issue and when, and more. Using a VCS also generally means that if you screw things up or lose files, you can easily recover. In addition, you get all this for very little overhead.

Local Version Control Systems

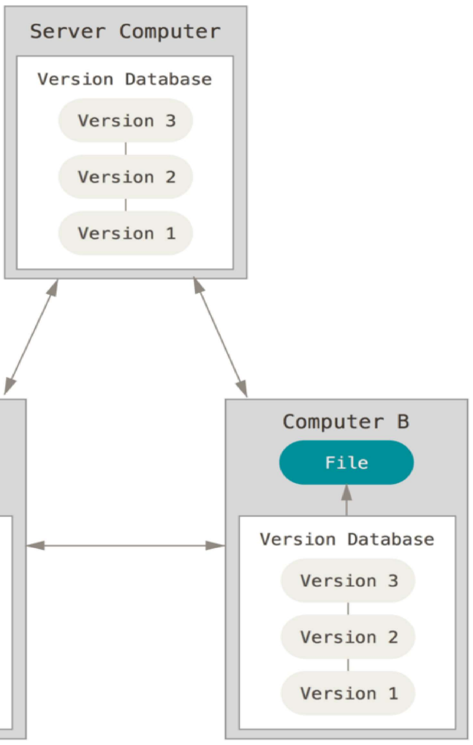
Many people’s version-control method of choice is to copy files into another directory (perhaps a time-stamped directory, if they’re clever). This approach is very common because it is so simple, but it is also incredibly error prone. It is easy to forget which directory you’re in and accidentally write to the wrong file or copy over files you don’t mean to. To deal with this issue, programmers long ago developed local VCSs that had a simple database that kept all the changes to files under revision control.

Centralized Version Control Systems

The next major issue that people encounter is that they need to collaborate with developers on other systems. To deal with this problem, Centralized Version Control Systems (CVCSs) were developed. These systems (such as CVS, Subversion, and Perforce) have a single server that contains all the versioned files, and a number of clients that check out files from that central place. For many years, this has been the standard for version control.

Distributed Version Control Systems (DVCS)

In a DVCS *(such as Git, Mercurial, Bazaar or Darcs)*, clients don't just check out the latest snapshot of the files; rather, they fully mirror the repository, including its full history. Thus, if any server dies, and these systems were collaborating via that server, any of the client repositories can be copied back up to the server to restore it. Every clone is really a full backup of all the data.



Distributed Version Control Systems (DVCS)

Git Installation

Installing on *Linux*

If you want to install the basic Git tools on Linux via a binary installer, you can generally do so through the package management tool that comes with your distribution. If you're on Fedora (or any closely-related RPM-based distribution, such as RHEL or CentOS), you can use dnf:

```
$ sudo dnf install git-all
```

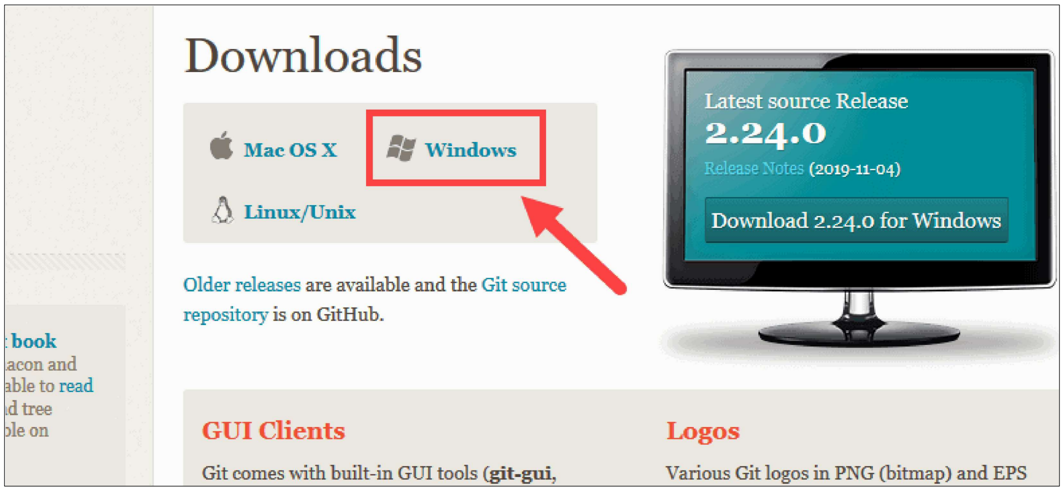
If you're on a Debian-based distribution, such as Ubuntu, try apt:

```
$ sudo apt install git-all
```

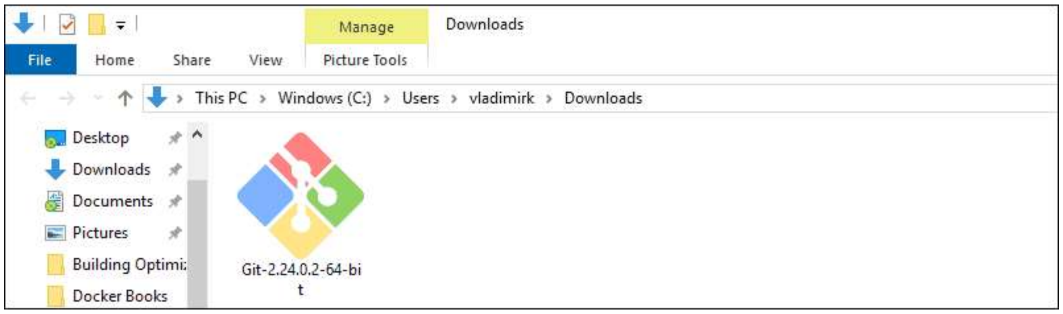
Installing on *Windows*

1. Browse to the official Git website: https://git-scm.com/downloads

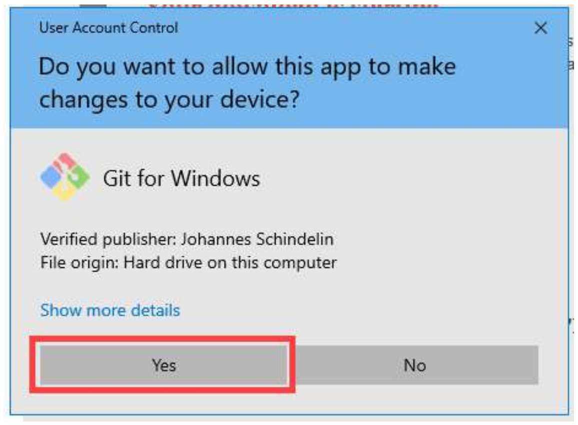
2. Click the download link for Windows and allow the download to complete.


Downloads
Mac OS X
Windows
Linux/Unix
Latest source Release
2.24.0
Release Notes (2019-11-04)
Download 2.24.0 for Windows
Older releases are available and the Git source repository is on GitHub.
GUI Clients
Git comes with built-in GUI tools (git-gui,
Logos
Various Git logos in PNG (bitmap) and EPS

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.



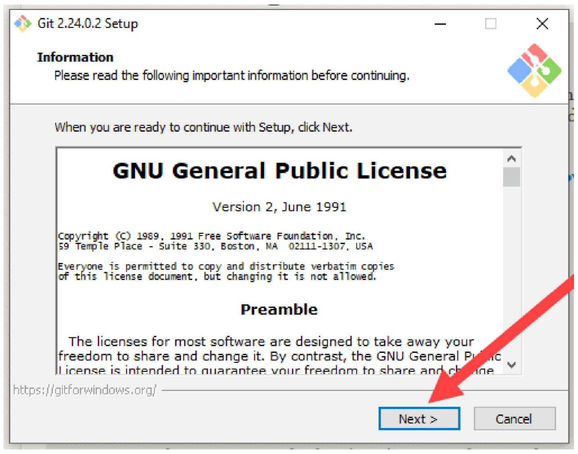
User Account Control

Do you want to allow this app to make changes to your device?

Git for Windows

Verified publisher: Johannes Schindelin
File origin: Hard drive on this computer

5. Review the GNU General Public License, and when you're ready to install, click Next.



Git 2.24.0.2 Setup

Information
Please read the following important information before continuing.

When you are ready to continue with Setup, click Next.

GNU General Public License

Version 2, June 1991

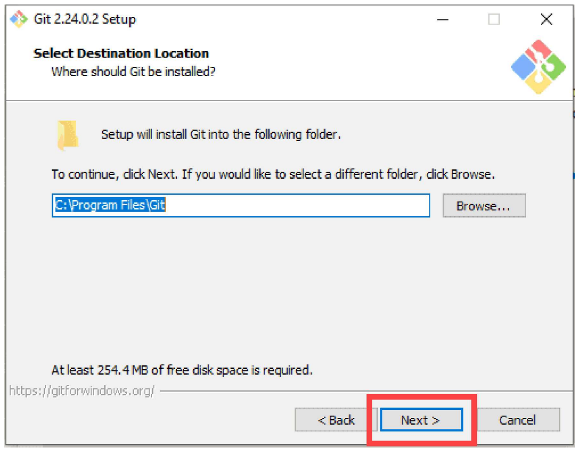
Copyright (C) 1989, 1991 Free Software Foundation, Inc.
59 Temple Place - Suite 330, Boston, MA 02111-1307, USA

Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change

6. The installer will ask you for an installation location. Leave the default, unless you have reason to change it, and click Next.



Git 2.24.0.2 Setup

Select Destination Location
Where should Git be installed?

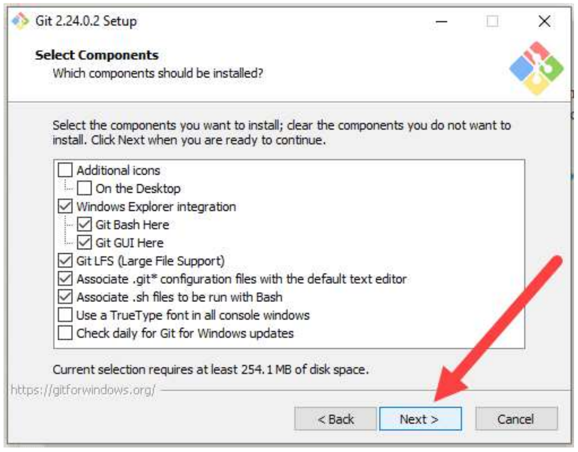
Setup will install Git into the following folder.

To continue, click Next. If you would like to select a different folder, click Browse.

C:\Program Files\Git

At least 254.4 MB of free disk space is required.

7. The installer will offer you options. Simply click Next.



Git 2.24.0.2 Setup

Select Components
Which components should be installed?

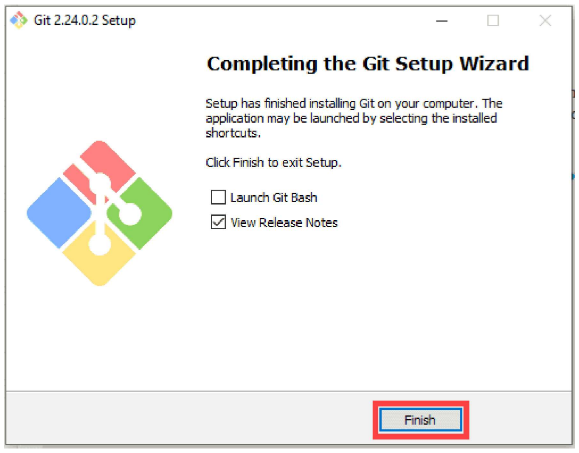
Select the components you want to install; clear the components you do not want to install. Click Next when you are ready to continue.

- [ ] Additional icons
  - [ ] On the Desktop
- [x] Windows Explorer integration
  - [x] Git Bash Here
  - [x] Git GUI Here
- [x] Git LFS (Large File Support)
- [x] Associate .git* configuration files with the default text editor
- [x] Associate .sh files to be run with Bash
- [ ] Use a TrueType font in all console windows
- [ ] Check daily for Git for Windows updates

Current selection requires at least 254.1 MB of disk space.

https://gitforwindows.org/

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



Git 2.24.0.2 Setup

Completing the Git Setup Wizard

Setup has finished installing Git on your computer. The application may be launched by selecting the installed shortcuts.

Click Finish to exit Setup.

- [ ] Launch Git Bash
- [x] View Release Notes

Git Configure

Open a *Command Prompt/Git-Bash/Terminal*.

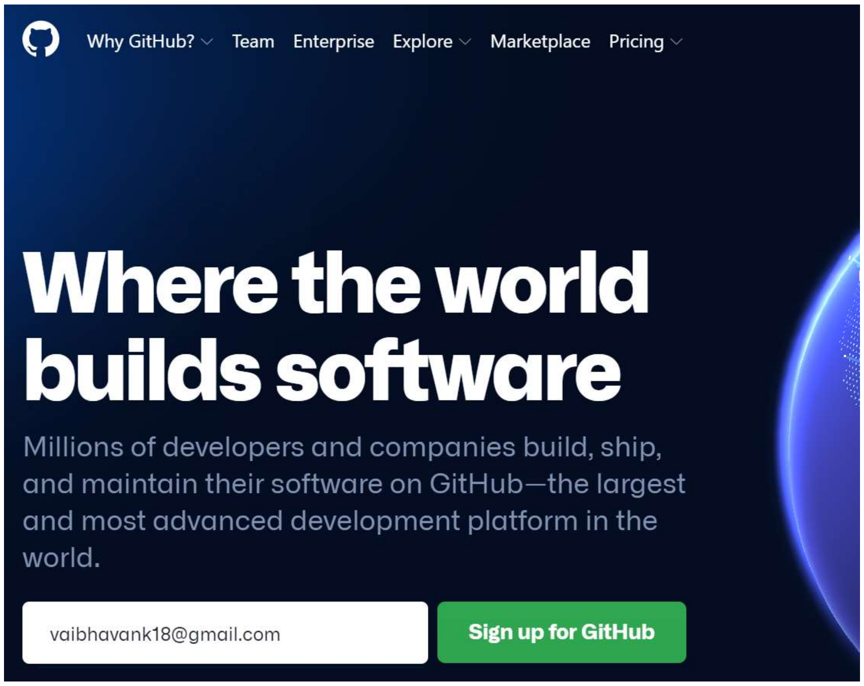
Run the following commands to configure your Git username and email using the following commands. These details will be associated with any commits that you create:

```
$ git config --global user.name "your name"
$ git config --global user.email "your email id"
```

Github Account

1. Go to https://www.github.com

2. Enter your email address in the email field next to sign up button.

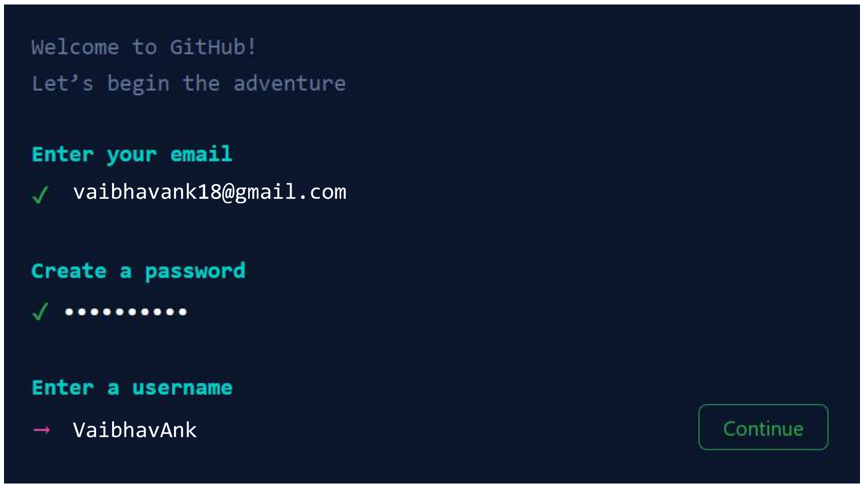


Where the world builds software

Millions of developers and companies build, ship, and maintain their software on GitHub—the largest and most advanced development platform in the world.

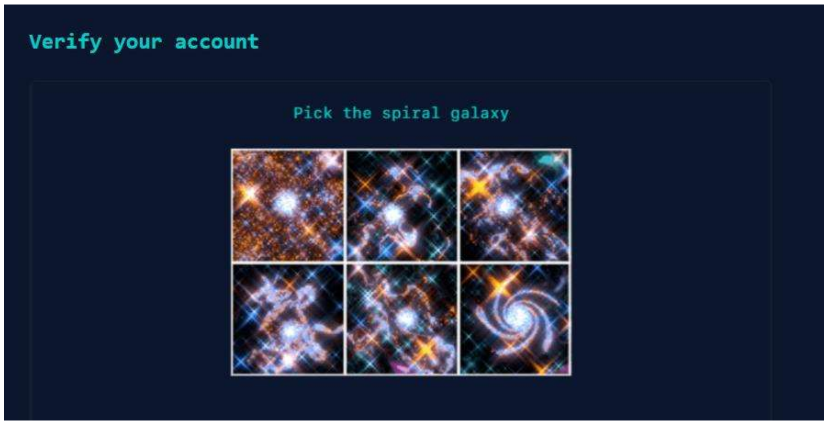
vaibhavank18@gmail.com

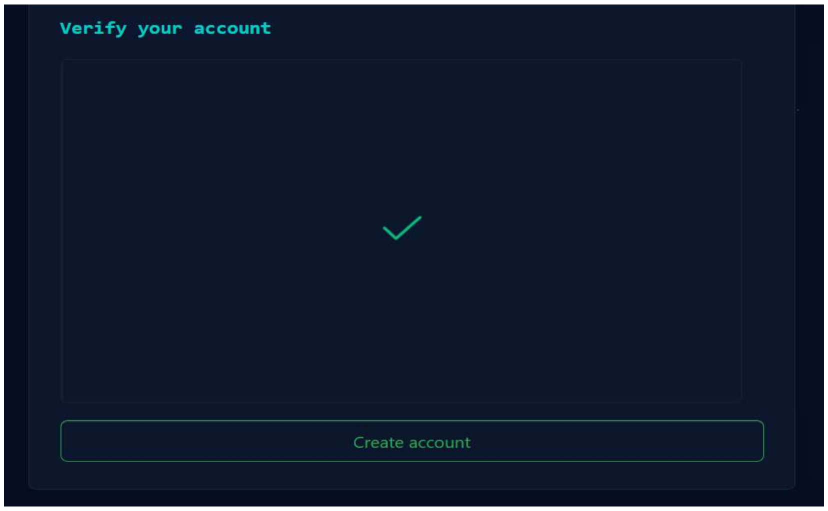
3. Enter a password and a username for your account.



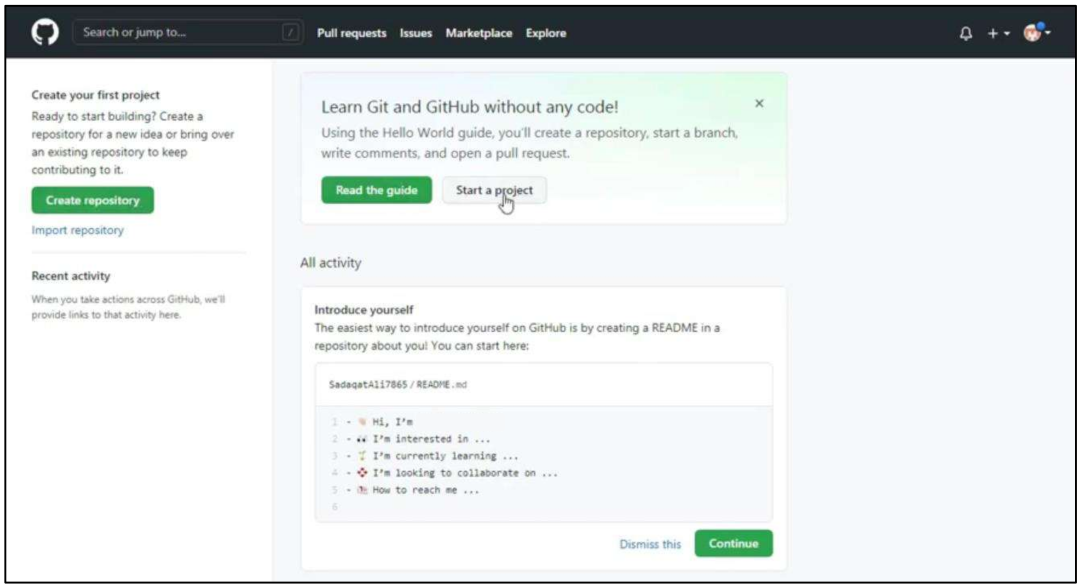
```
Welcome to GitHub!
Let's begin the adventure

Enter your email
✓ vaibhavank18@gmail.com

Create a password
✓ ••••••••••

Enter a username
→ VaibhavAnk
```

4. Solve the two puzzle to get pass the process and click on create account.





5. Your Github account is created.



Create your first project

Ready to start building? Create a repository for a new idea or bring over an existing repository to keep contributing to it.

Recent activity

When you take actions across GitHub, we'll provide links to that activity here.

Learn Git and GitHub without any code!

Using the Hello World guide, you'll create a repository, start a branch, write comments, and open a pull request.

All activity

Introduce yourself

The easiest way to introduce yourself on GitHub is by creating a README in a repository about you! You can start here:

```
SadaqatAli7865 / README.md

- 👋 Hi, I'm
- 👀 I'm interested in ...
- 🌱 I'm currently learning ...
- 💞️ I'm looking to collaborate on ...
- 📫 How to reach me ...
```

Conclusion:

One of the biggest advantages of Git is its branching capabilities. Unlike centralized version control systems, Git branches are cheap and easy to merge. This facilitates the feature branch workflow popular with many Git users. Feature branches provide an isolated environment for every change to your codebase. Github provides us many benefits like storing our project remotely, accessing it anytime and from anywhere. Teams can easily contribute and work on a github projects. Thus, we have successfully installed git and created a github account.