FRONT END DEVELOPMENT FRAMEWORKS

WEEK-1 Experiment:

Aim: To Project Initialization and First Commit

Description:

To begin, install Git from the official website and configure it using git config --global user.name "Your Name" and git config --global user.email "your.email@example.com" to set your identity for commits. Next, create a project folder, open a terminal in that location, and initialize a local Git repository with git init. Inside this folder, create an index.html file containing your name, title, and a brief paragraph about yourself using basic HTML structure. Once the file is ready, track it using git add index.html and commit it with a message using git commit -m "Initial commit with HTML structure". Finally, view your commit history with the git log command to verify that your changes have been recorded successfully.

Tasks:

1. Install Git and configure your name and email.

2. Create a local Git repository using git init.

3. Create a basic index.html file that displays your name, title, and a short paragraph.

4. Track the file using git add and commit it using git commit -m "Initial commit with HTML structure".

5. View the commit history using git log.

Solutions:

1. Install Git and configure your name and email.

1.1Installation of Git and Initial Setup on Windows

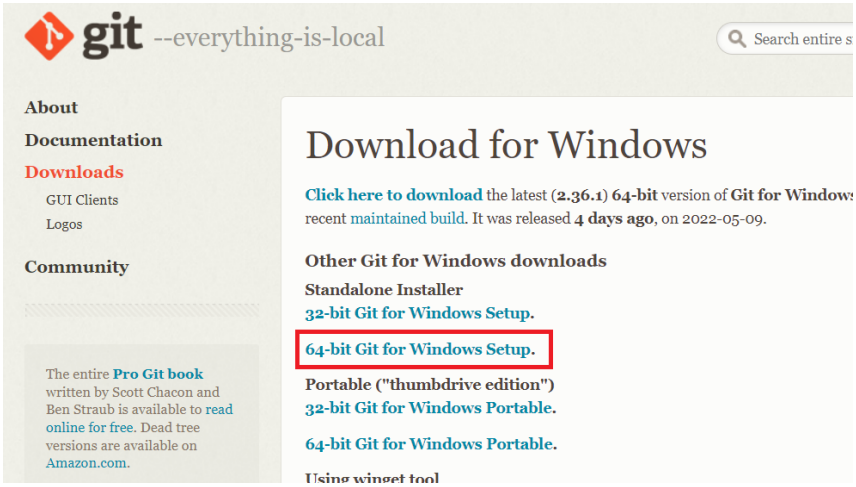
Git is an open-source and free, decentralized version control system designed to handle projects of all sizes with speed and efficiency. Basically, it is a software tracking application that is commonly used to monitor projects across several teams. The best way of downloading and installing Git on the windows command line is to download it from its official site. Let's learn the step for the same.

Steps to download and install Git on Windows:

Download the Git:

Step 1: Go to the official website: https://git-scm.com

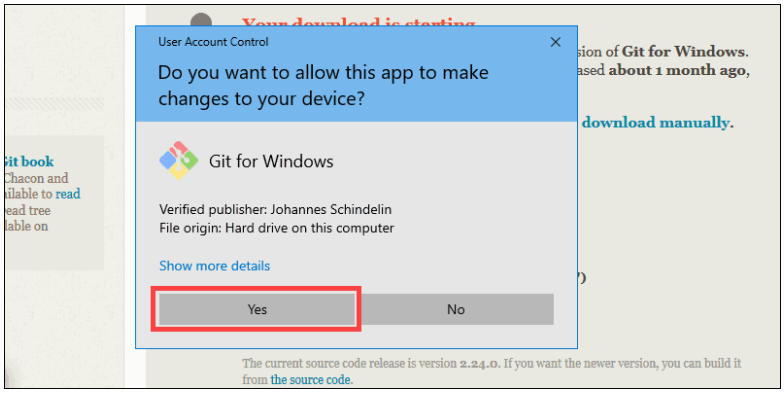
Step 2: Click on 64-bit Git for Windows Setup and allow the download to complete.



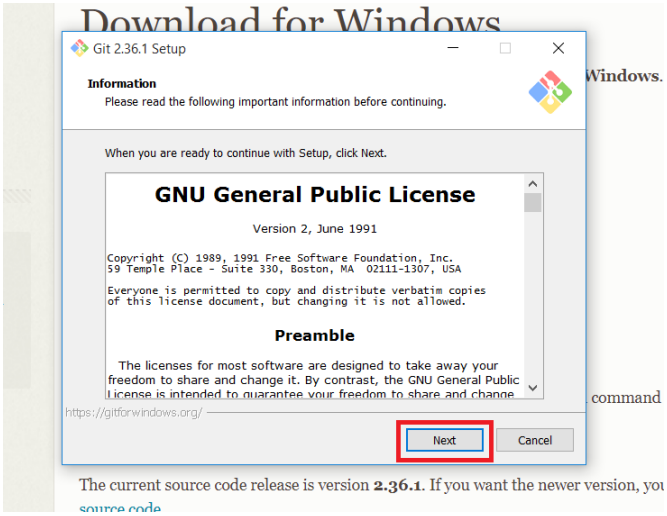
Extract and Launch Git Installer

Step 3: Go to your download location and double-click the file to launch the installer.

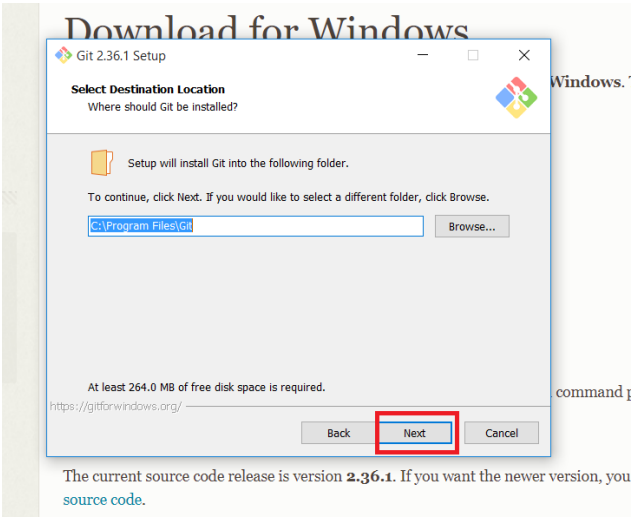
Step 4: Allow the app to modify your device by selecting Yes in the User Account Control window that appears.



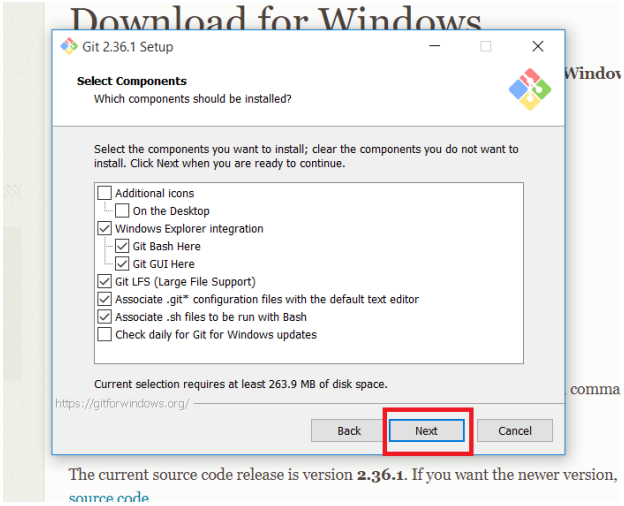
Step 5: Check the GNU General Public License and click Next.



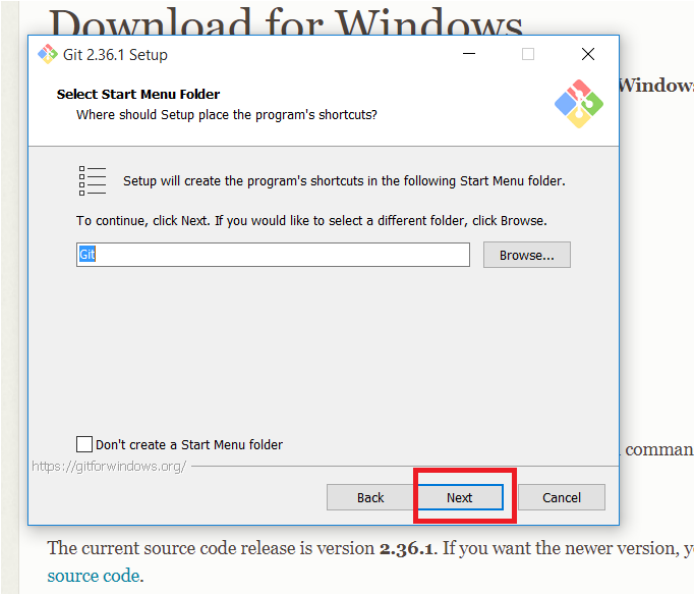
Step 6: Select the install location. If you don't have a reason to modify it, leave it to default and click Next.



Step 7: A screen for component selection will display. Leave the settings as it is and click Next.

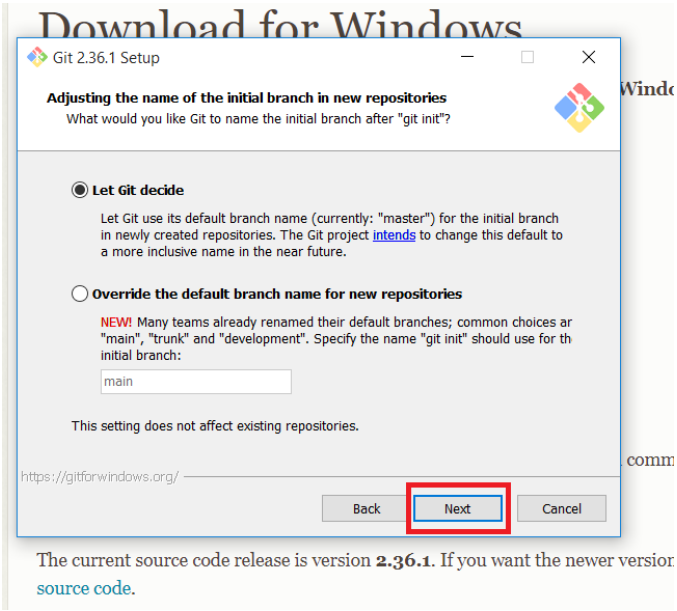


Step 8: The installer asks you to create a start menu folder. Simply click Next.

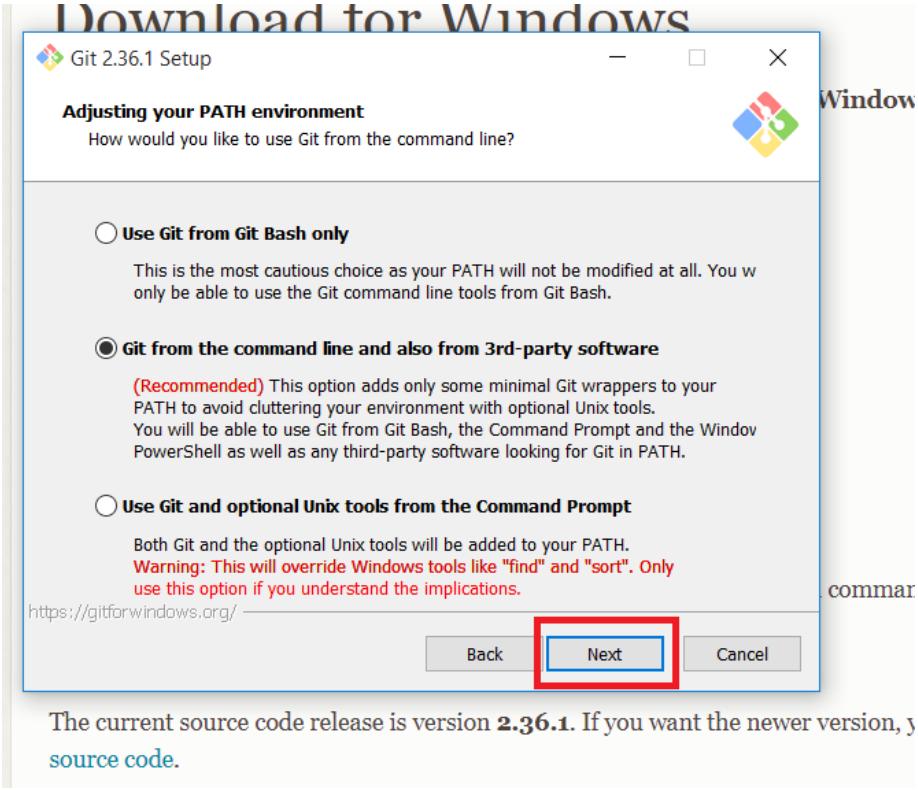


Step 9: Choose the text editor you want to use with Git and click Next.

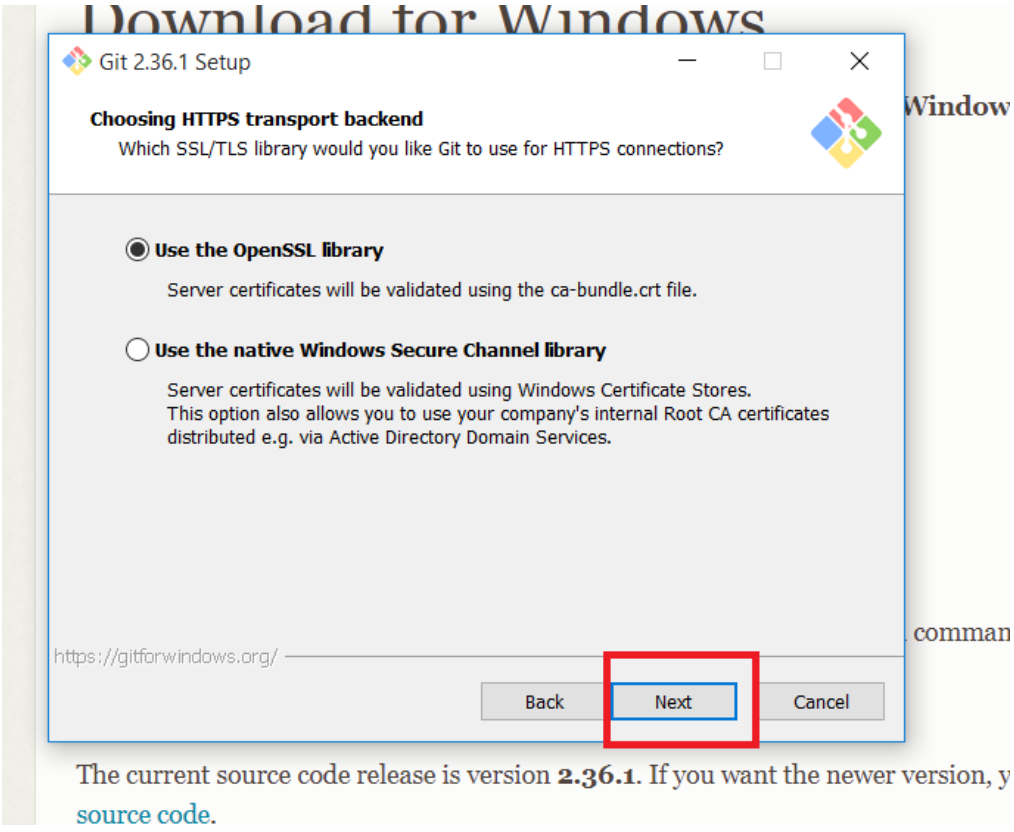
Step 10: The following step allows you to give your original branch a new name. 'Master' is the default. Leave the default choice selected and press the Next button.



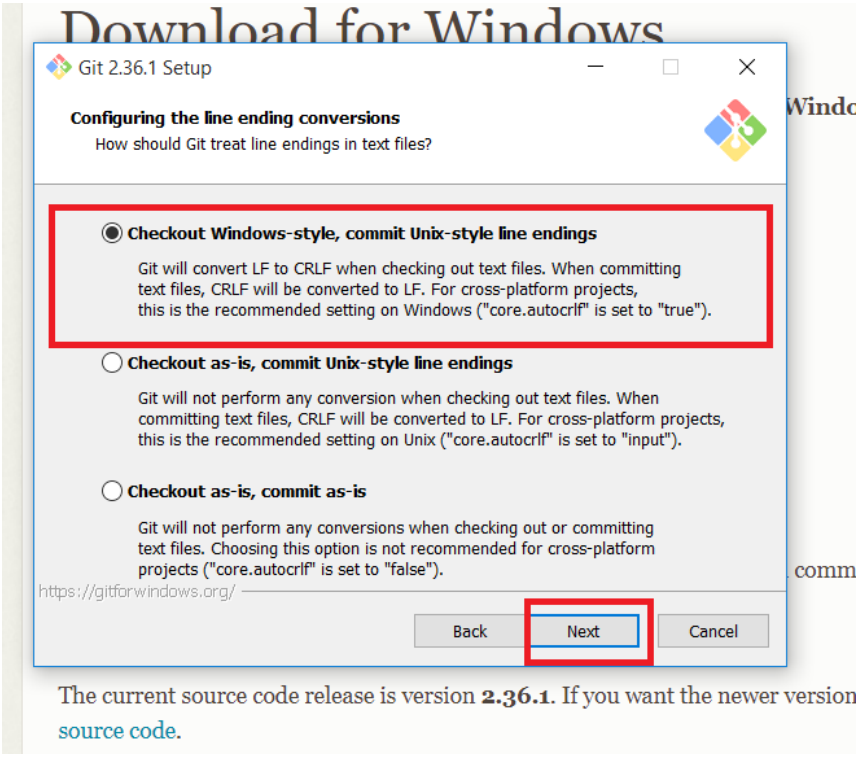
Step 11: You can adjust the PATH environment during this installation phase. When you run a command from the command line, the PATH is the default set of folders that are included. Continue by selecting the middle (recommended) option and clicking Next.



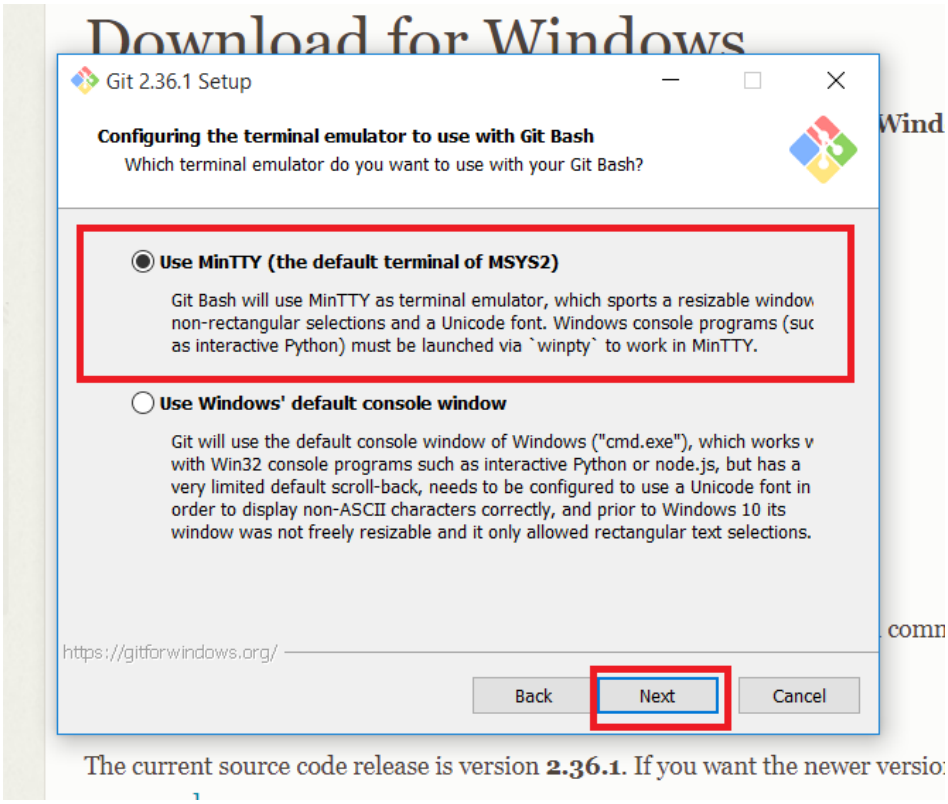
Step 12: The following option concerns server certificates. The default choice is used by the majority of users. Simply click Next.



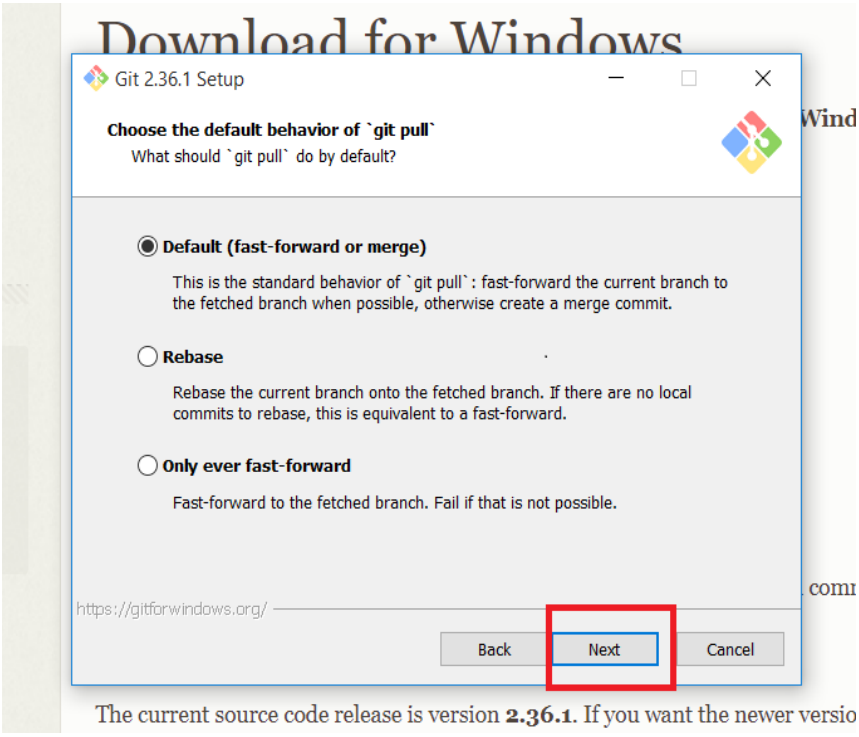
Step 13: This step deals with how data is structured, and altering this option may create issues. So, it is advised to leave the default selection.



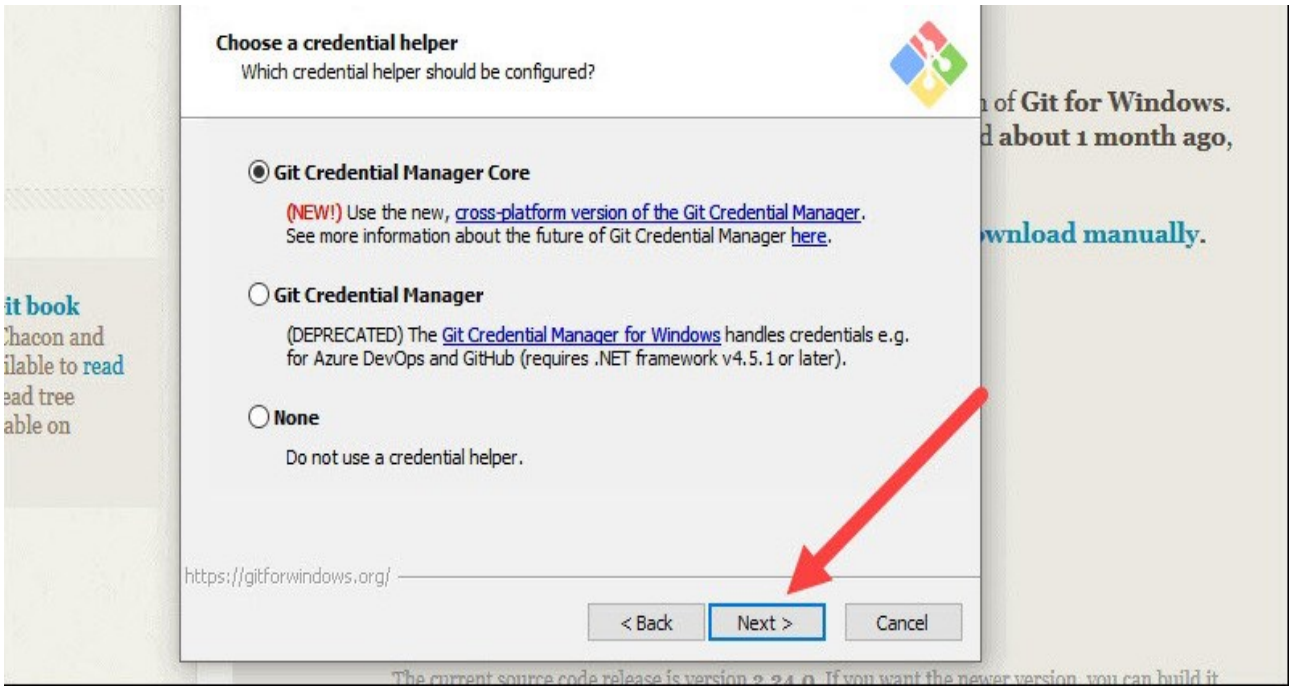
Step 14: Select the terminal emulator that you wish to use. Because of its features, the default MinTTY is suggested. Click Next.



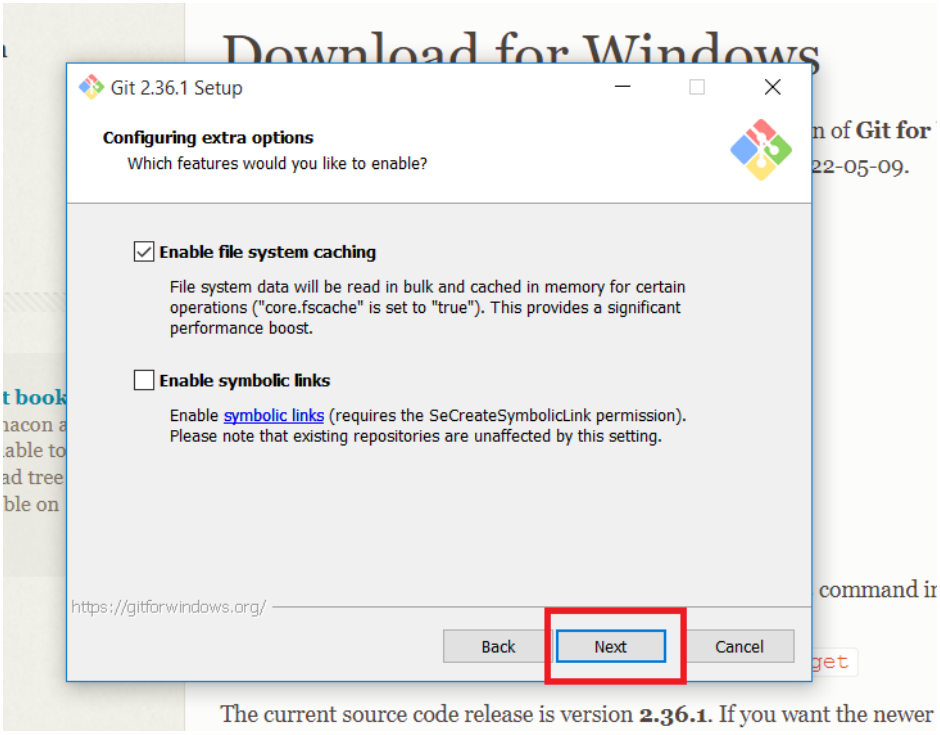
Step 15: The installer now prompts you to specify what the git pull command should perform. Leave the default selected option and click Next.



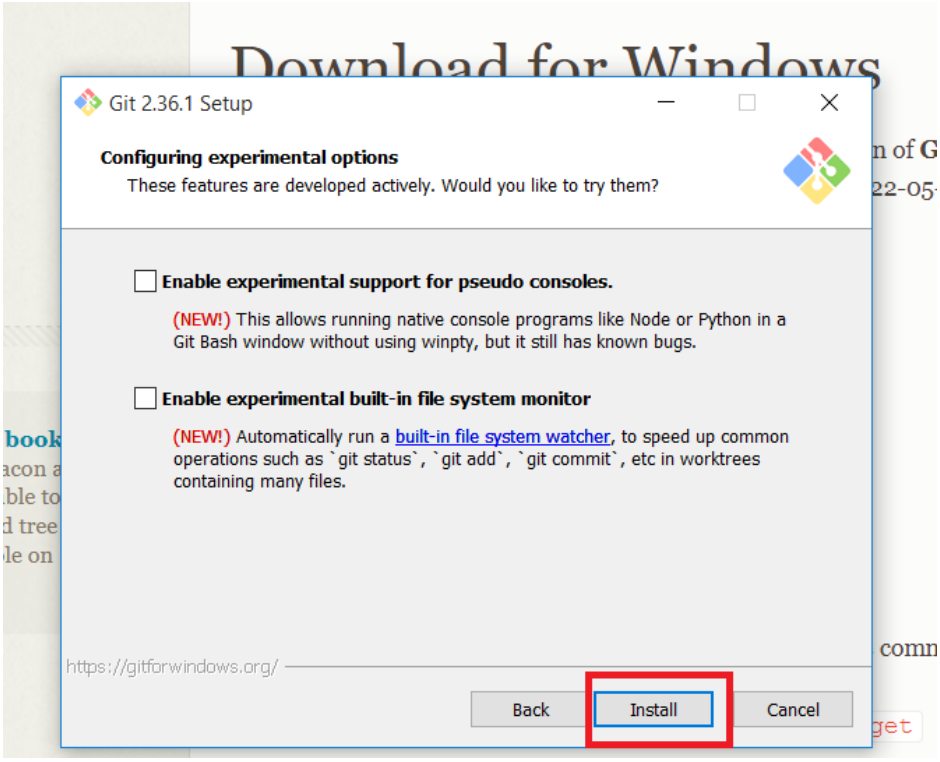
Step 16: The next step is to decide which credential helper to employ. Credential helpers are used by Git to retrieve or save credentials. Leave the default selection and click Next.



Step 17: Although the default choices are suggested, this step allows you to select which additional features to activate.

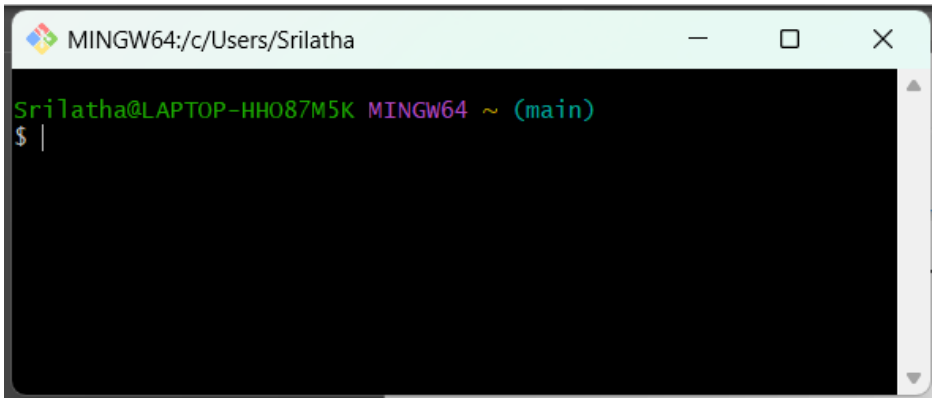


Step 18: Git offers to install some experimental features. Leave them unchecked and click Install.



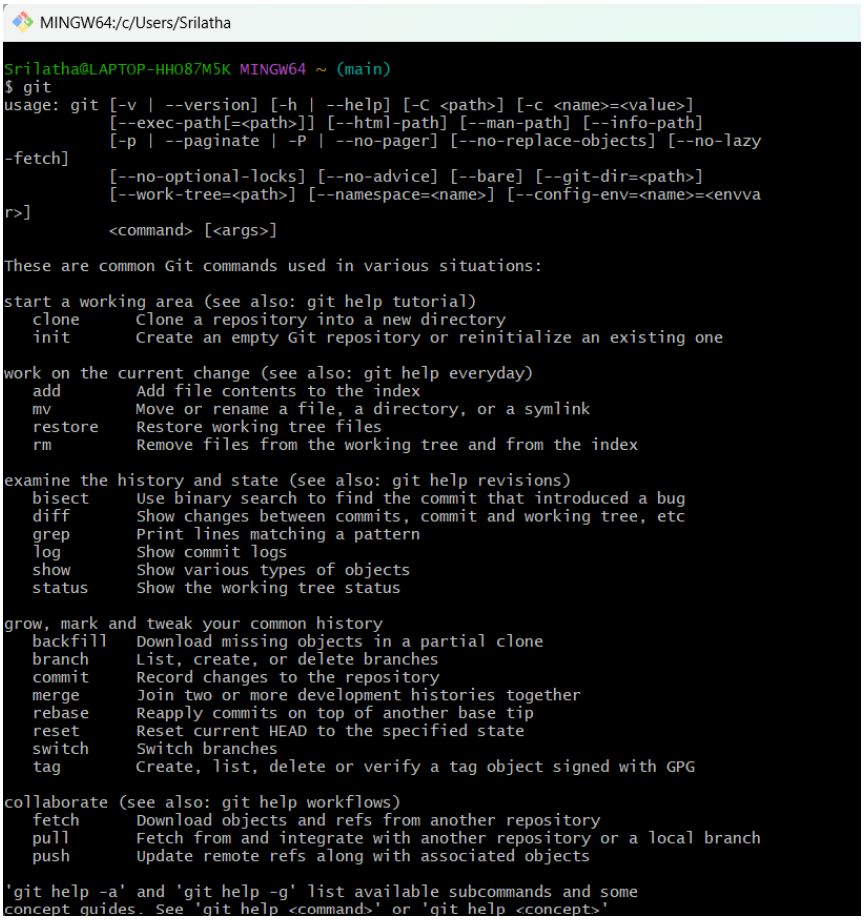
The Current Source Code Version of the Git is 2.50.1

Step 19: Once the installation is complete, launch the Git bash.



Here $ is the command prompt for Git.

Let us type git command in the Git Bash Command Line. Then git displays all the Git commands as shown below.



This indicates git is installed successfully. So, git is ready to use.

Installation of Git on Linux OS(Debian/Ubuntu):

sudo apt update

sudo apt install git

Installation of Git on Mac OS:

brew install git

1.2. Configuration of your name and email.

Git uses a username and email address to associate your commits with your identity. This is especially important in collaborative projects, as it helps track who made which changes.

Why Configure Name and Email in Git?

- To associate your work with your identity in version control.
- Required before making your first commit.
- Helps maintainers know who made what change.

Step-by-Step Configuration

1) Open Git Bash or Terminal
2) Set Your Name using git config command

   git config --global user.name "Your Full Name"

   For Example:

   git config --global user.name "srilathadoddi"

3) Set Your Email using git config command
   git config --global user.email "youremail@example.com"

   For Example:

   git config --global user.email doddisrilatha@gmail.com

4) Verify Your Configuration
   Use the following command to confirm your settings:

   git config --global --list

Sample Output:

```
Srilatha@LAPTOP-HHO87M5K MINGW64 ~ (main)
$ git config --global --list
user.name=srilathadoddi
user.email=doddisrilatha@gmail.com
credential.helper=manager-core
```

Note the following points:

1. Don't forget to use quotes if your name has spaces.
2. Make sure the email address is valid and consistent with GitHub/GitLab if you plan to push commits online.

2.Create a Local Git Repository Using git init

1. Open Git Bash or Terminal.
2. Navigate to or create a project folder:

   mkdir week1

   cd week1

3. Initialize Git repository:

   git init

   This creates a hidden .git folder to track versions.

3. Create a Basic index.html File

Create a file named index.html with the following content:

```
<!DOCTYPE html>
<html lang="en">
<head>
   <title>My Website</title>
</head>
<body>
   <h1>FRONT END DEVELOPMENT FRAMWORKS</h1>
   <h2> FRONT END DEVELOPMENT FRAMWORKS </h2>
   <p>A short paragraph about yourself.</p>
</body>
</html>
```

4. Track the file using git add and commit it using git commit -m "Initial commit with HTML structure".

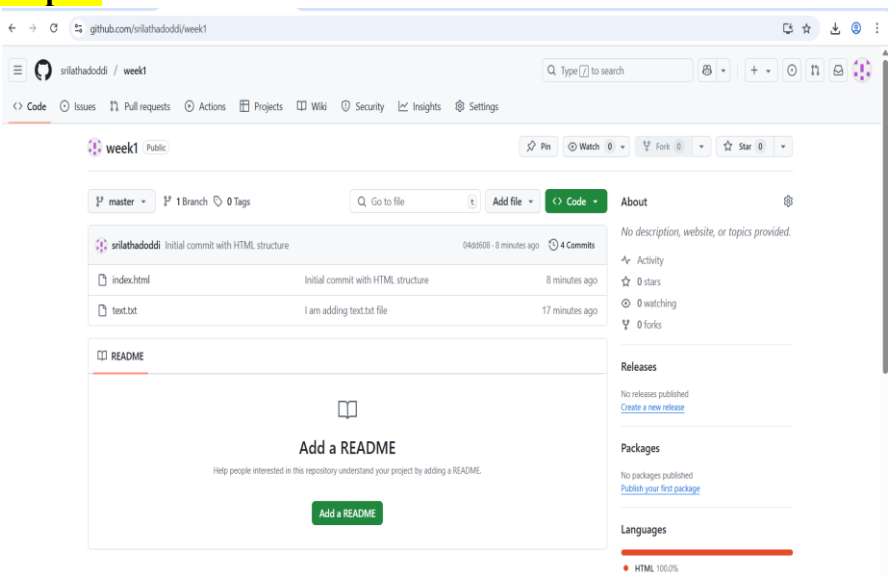
1) **$ git add index.html**
2) **$ git config --global user.name "srilathadoddi"**
   $ git config --global user.email "doddisrilatha@gmail.com"
3) **$ git commit -m "Initial commit with HTML structure"**
4) **$ git remote add origin https://github.com/srilathadoddi/week1.git**

Note: In Git, origin is the default name given to the remote repository when you clone or connect your local Git project to a remote (like GitHub, GitLab, or Bitbucket).

5) **$ git push -u origin master**

Note: The -u flag is used with the git push command. It sets the "upstream" branch, meaning it links your local branch to a remote branch so you don't have to specify the remote and branch every time you push or pull.

Sample output:



5. View the Commit History

Check commit history using log command

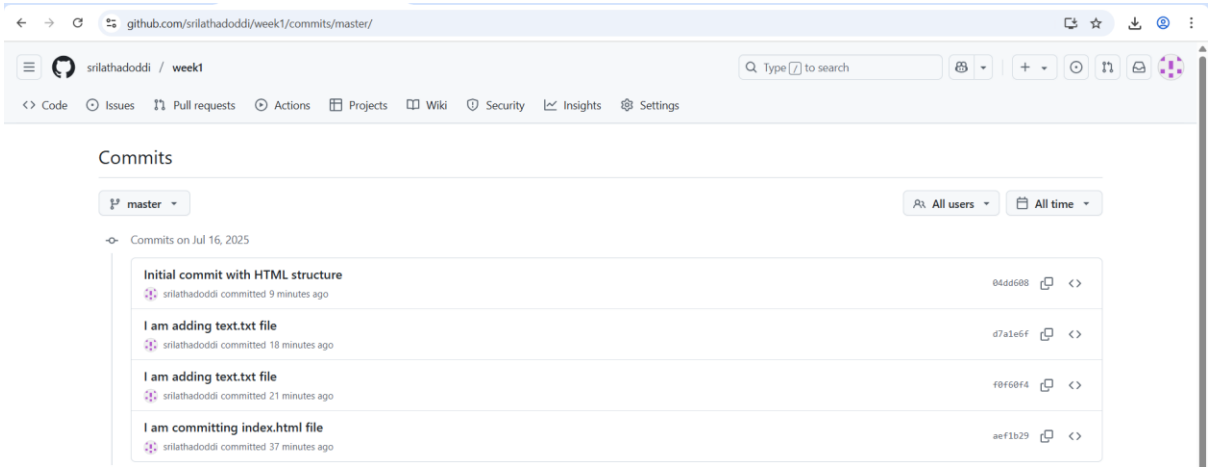
$ git log

Sample Output:

```
Srilatha@LAPTOP-HHO87M5K MINGW64 /d/FEDF/week1 (master)
$ git log
commit 04dd6086de6182d857c729cbec49599148851a55 (HEAD -> master, origin/master)
Author: srilathadoddi <doddisrilatha@gmail.com>
Date:   Wed Jul 16 09:03:09 2025 +0530

    Initial commit with HTML structure

commit d7a1e6fa522e9e037f8193769262374ab85b5637
Author: srilathadoddi <doddisrilatha@gmail.com>
Date:   Wed Jul 16 08:54:18 2025 +0530

    I am adding text.txt file

commit f0f60f4c994e24b3b528909bb3cde178f4c252b8
Author: srilathadoddi <doddisrilatha@gmail.com>
```

In the GitHub also, you can see the commit history.

Sample Output:



Result:

The tasks were successfully executed without any errors. Git was installed and configured correctly with the user's name and email. A new local Git repository was initialized using git init, and the index.html file containing the user's name, title, and a short paragraph was created as expected. The file was successfully tracked using git add, and committed to the repository with the message "Initial commit with HTML structure". Finally, the commit history was viewed using git log, confirming that all actions were completed and recorded properly.