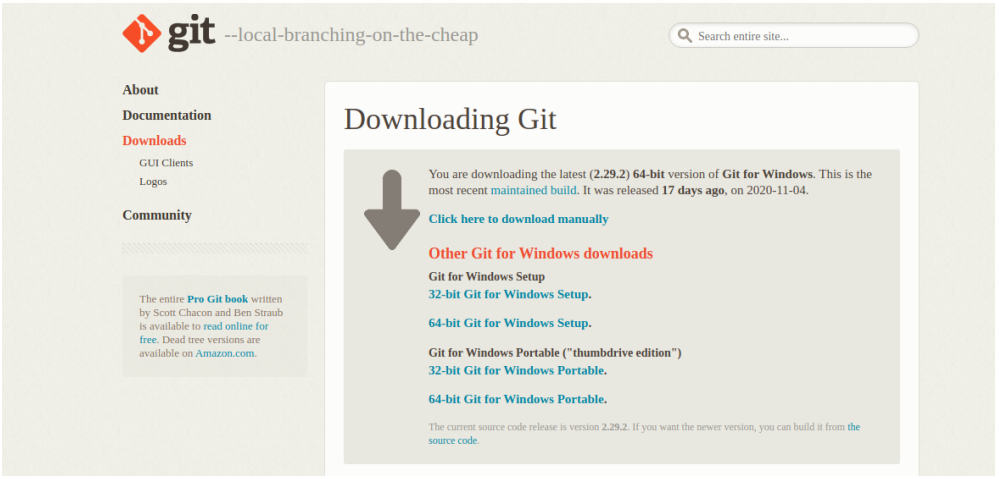
Git Installation:

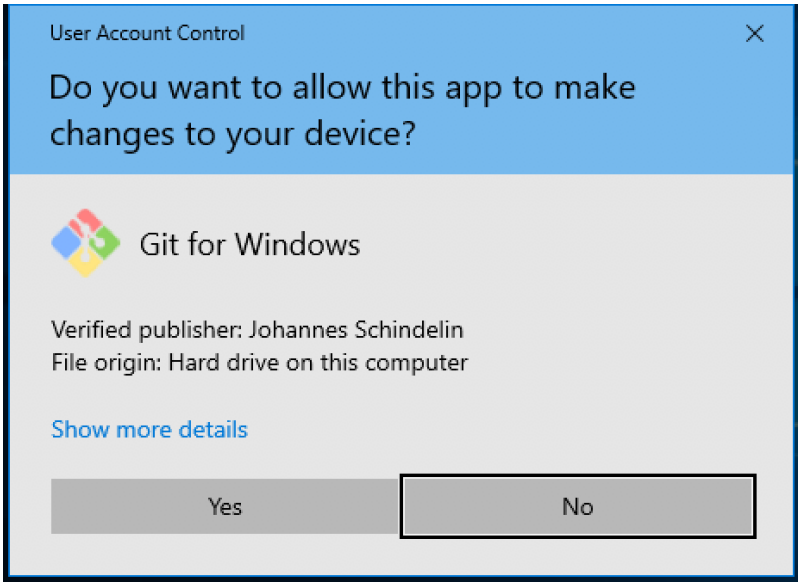
On Windows-:

1. Navigate to the Git website's Download page.

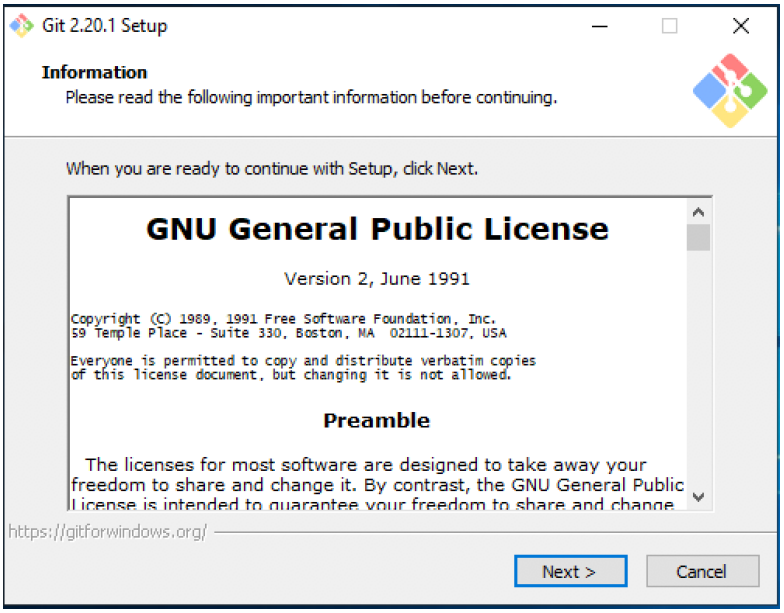


2. Click on a recent version of Git to download it.

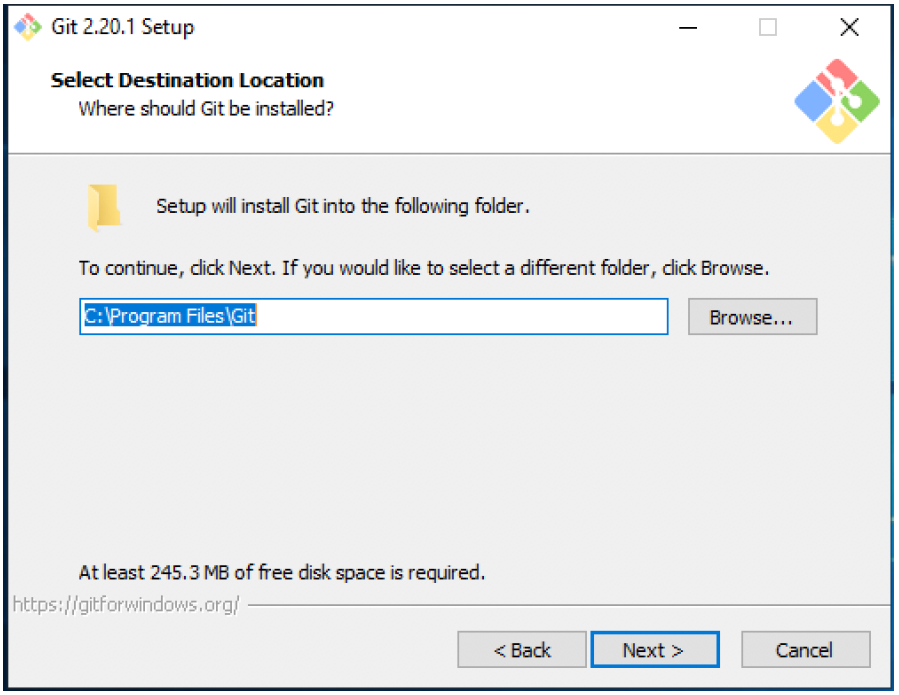
3. When you see an installation prompt, click on **Yes**:



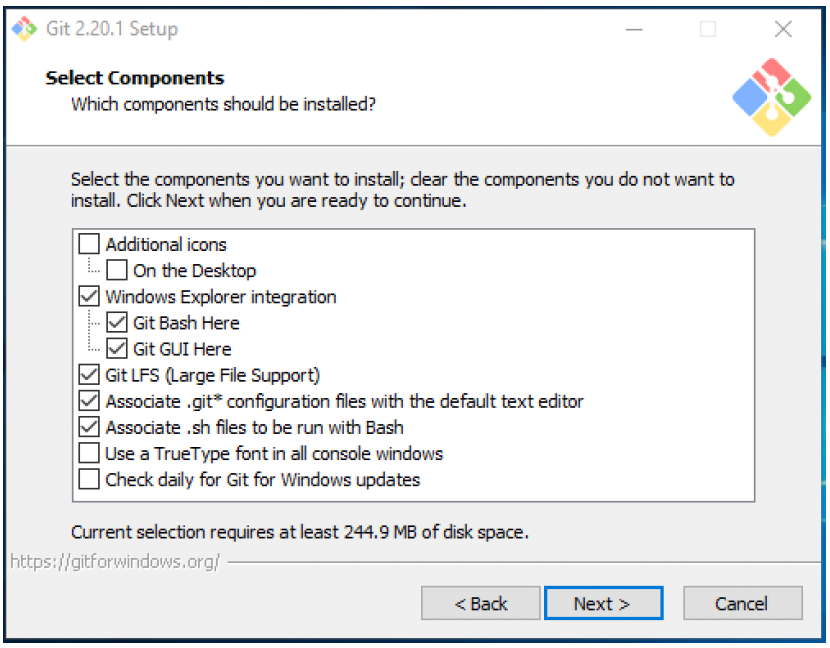
4. Agree to the GNU license terms:



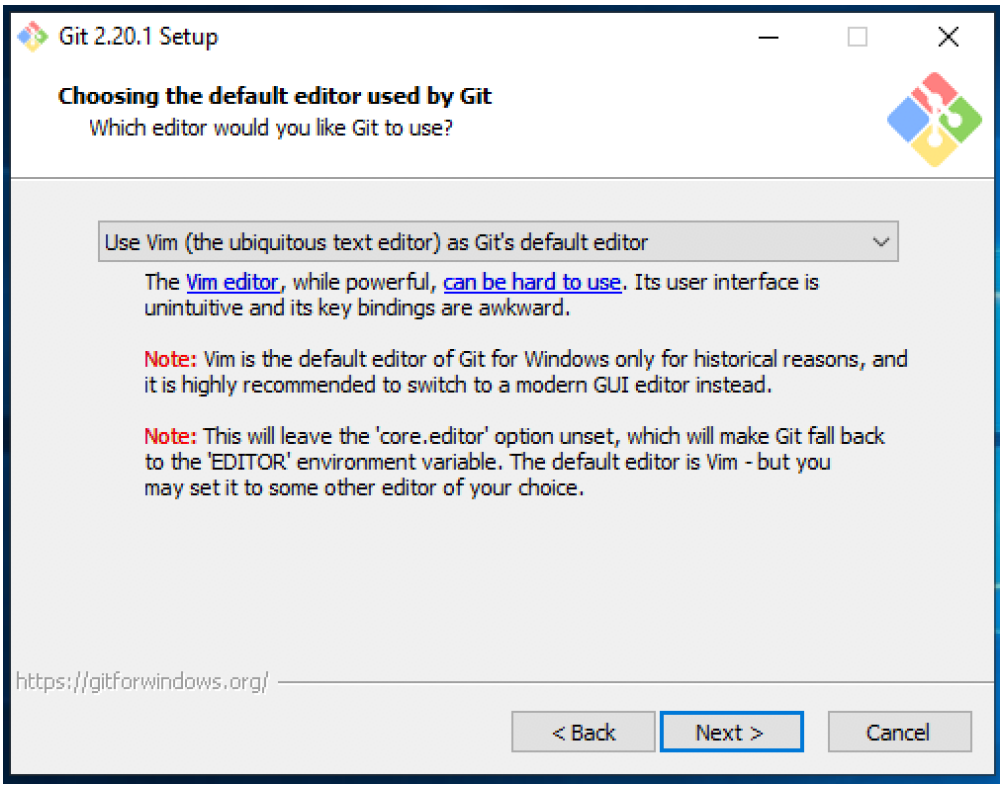
5. Select the directory you want Git to be installed in or use the default location:



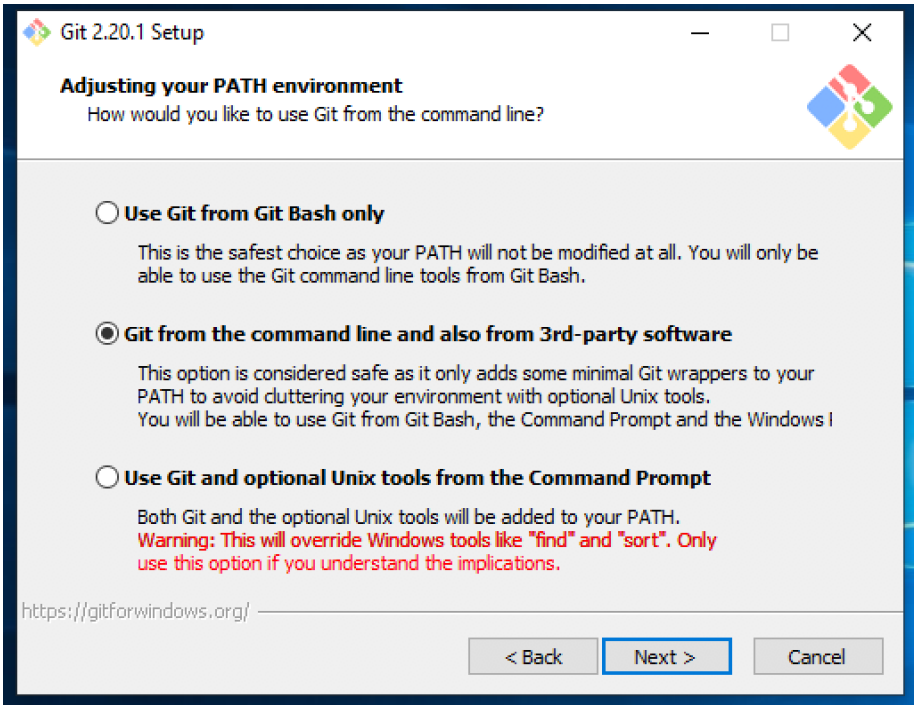
6. Select the components that you want to install. If you are unsure, go ahead with the default selection.



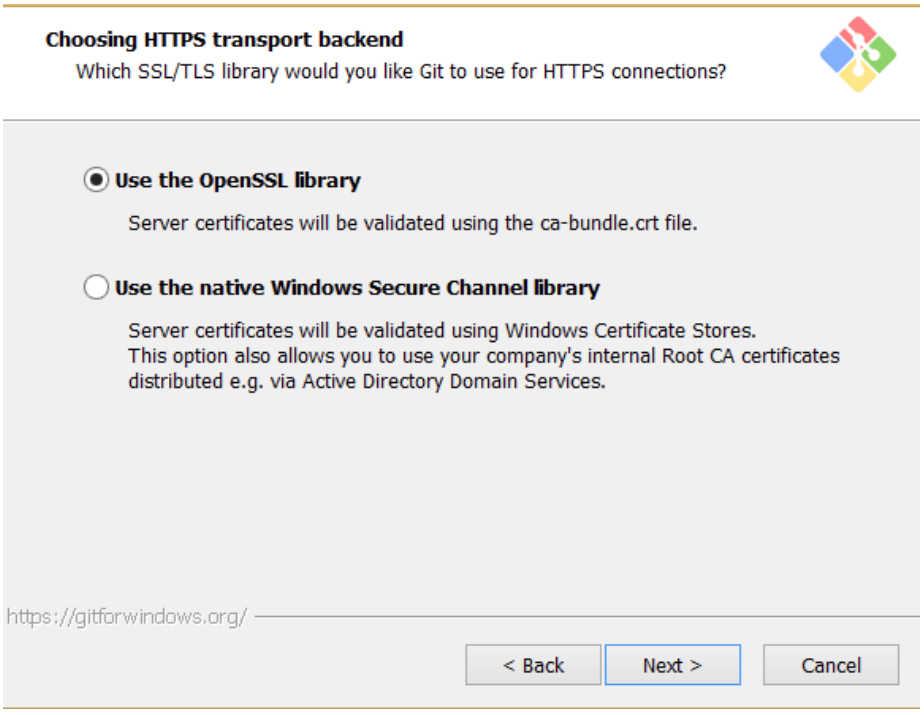
7. Choose the default editor for Git:



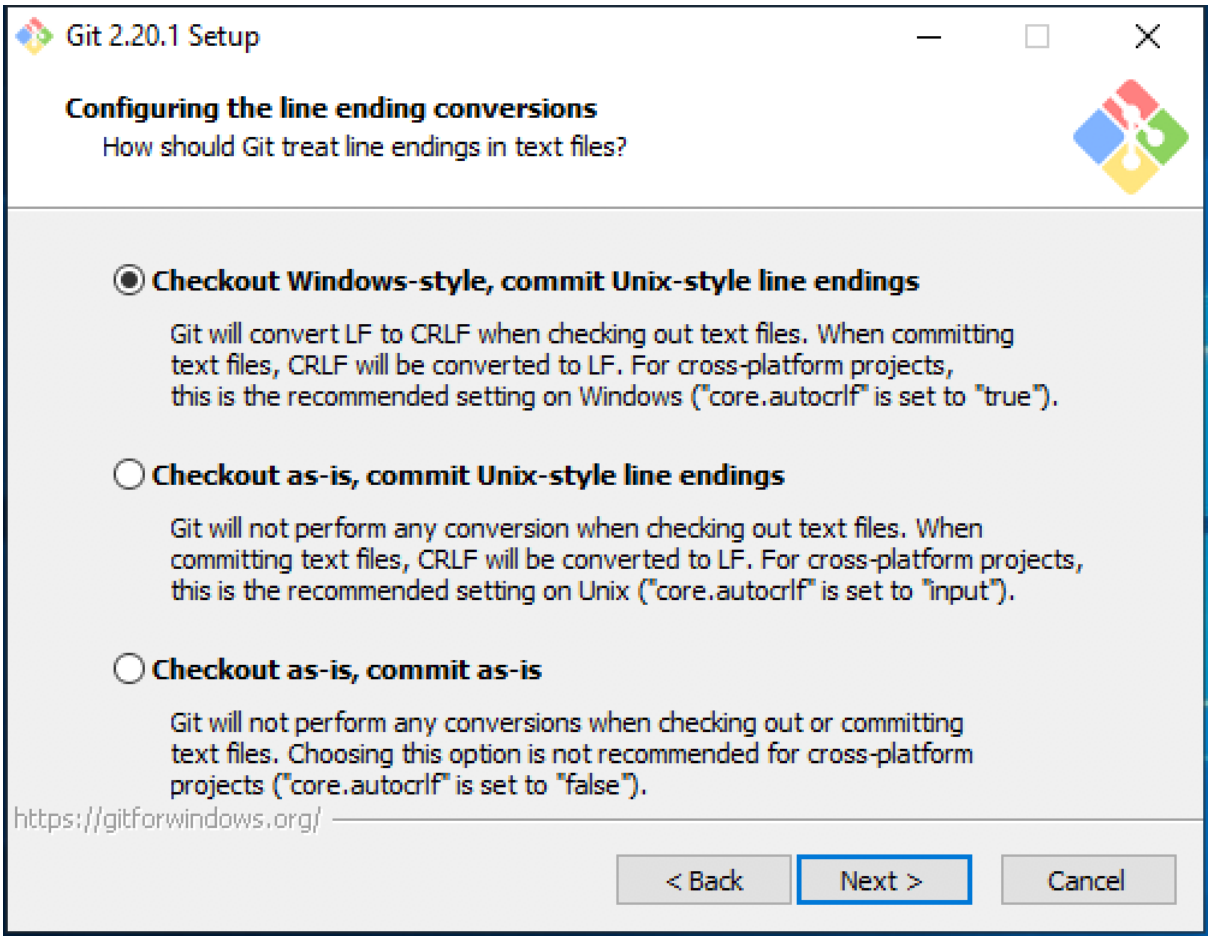
8. Select how you want to use Git from the command line from the options that are presented:



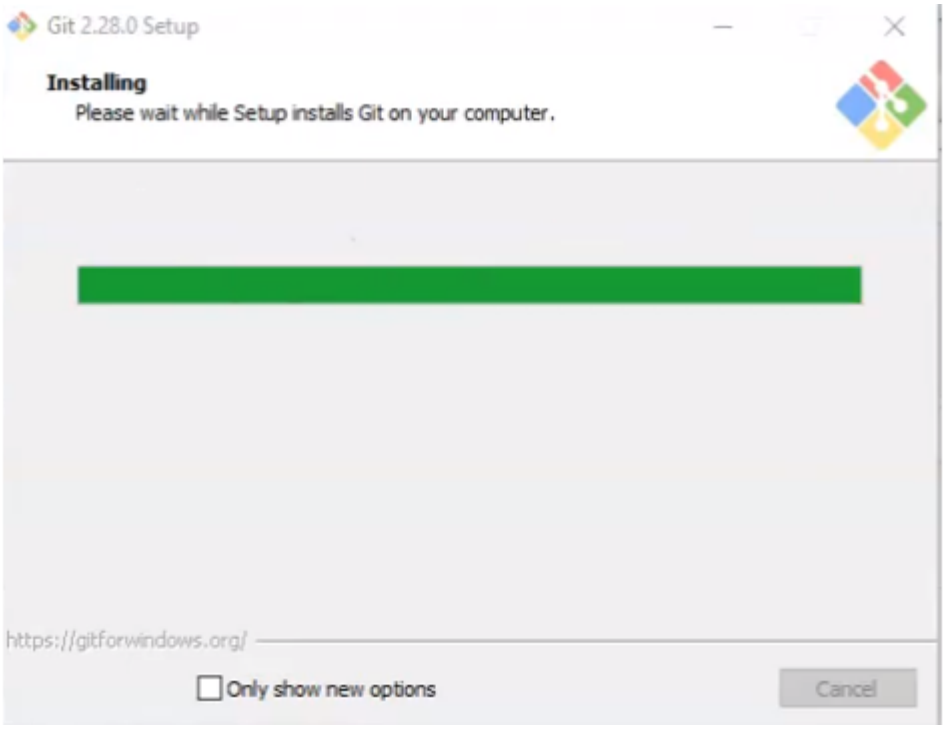
9. Select the SSL/TLS library that you want Git to use for HTTPs connections:



10. Select how Git should treat the line endings in text files:



11. Select your terminal emulator, default behavior of git pull, and some extra configuring options.
    For the simplest installation, keep MinTTY for the terminal emulator, use the default behavior (fast-forward or merge), and enable file system caching in configuring extra options. When you are done selecting your configuration options, click **Install** at the end.



12. Click on **Finish**. You should have a working Git installation on your Windows machine.

On Linux-:

Ubuntu based distros-:

Step 1

Open terminal.

Step 2

sudo apt install git

Step 3

Run git from the terminal.

Fedora / Redhat based distros-:

Step 1

Open terminal
Step 2

sudo dnf install git

Step 3

Run git from the terminal

Arch based distros-:

Step 1

Open terminal

Step 2

sudo pacman install git

Step 3

Run git from the terminal

Git Commands:

git config

Usage: *git config –global user.name "[name]"*

Usage: *git config –global user.email "[email address]"*

This command sets the author name and email address respectively to be used with your commits.

git init

Usage: *git init [repository name]*

This command is used to start a new repository.

git clone

Usage: *git clone [url]*

This command is used to obtain a repository from an existing URL.

git add

Usage: *git add [file]*

This command adds a file to the staging area.

git commit

Usage: *git commit -m "[Type in the commit message]"*

This command records or snapshots the file permanently in the version history.

git diff

Usage: *git diff*

This command shows the file differences which are not yet staged.

Usage: *git diff –staged*

This command shows the differences between the files in the staging area and the latest version present.

git reset

Usage: *git reset [file]*
This command unstages the file, but it preserves the file contents.

Usage: *git reset [commit]*

This command undoes all the commits after a specific commit and preserves the changes locally.

Usage: *git reset –hard [commit]*

This command discards all history and goes back to the specified commit.

git status

Usage: *git status*

This command lists all the files that have to be committed.

git rm

Usage: *git rm [file]*

This command deletes the file from your working directory and stages the deletion.

git log

Usage: *git log*

This command is used to list the version history for the current branch.

git show

Usage: *git show [commit]*

This command shows the metadata and content changes of the specified commit.

git tag

Usage: *git tag [commit]*

This command is used to give tags to the specified commit.

git branch

Usage: *git branch*

This command lists all the local branches in the current repository.

git checkout

Usage: *git checkout [branch name]*

This command is used to switch from one branch to another.

Usage: *git checkout -b [branch name]*

This command creates a new branch and also switches to it.

git merge

Usage: *git merge [branch name]*

This command merges the specified branch's history into the current branch.

git remote

Usage: *git remote add [remote name] [Remote Server Link]*

This command is used to connect your local repository to the remote server.

git push

Usage: *git push*

This command sends the committed changes of your local repository to your remote repository.

git pull

Usage: *git pull [Repository Link]*

This command fetches and merges changes on the remote server to your working directory.

git stash

Usage: *git stash save*

This command temporarily stores all the modified tracked files.

Usage: *git stash list*

This command lists all stashed changesets.

Usage: *git stash pop*

This command restores the most recently stashed files.