

Home > Tutorials > Git

# Git Install Tutorial

Learn about Git initial setup, Git LFS, and user-friendly Git GUI applications in this in-depth tutorial.

Apr 22, 2022 · 9 min read



**Abid Ali Awan**

Certified data scientist, passionate about building ML apps, blogging on data science, and editing.

---

## TOPICS

Git

Data Science

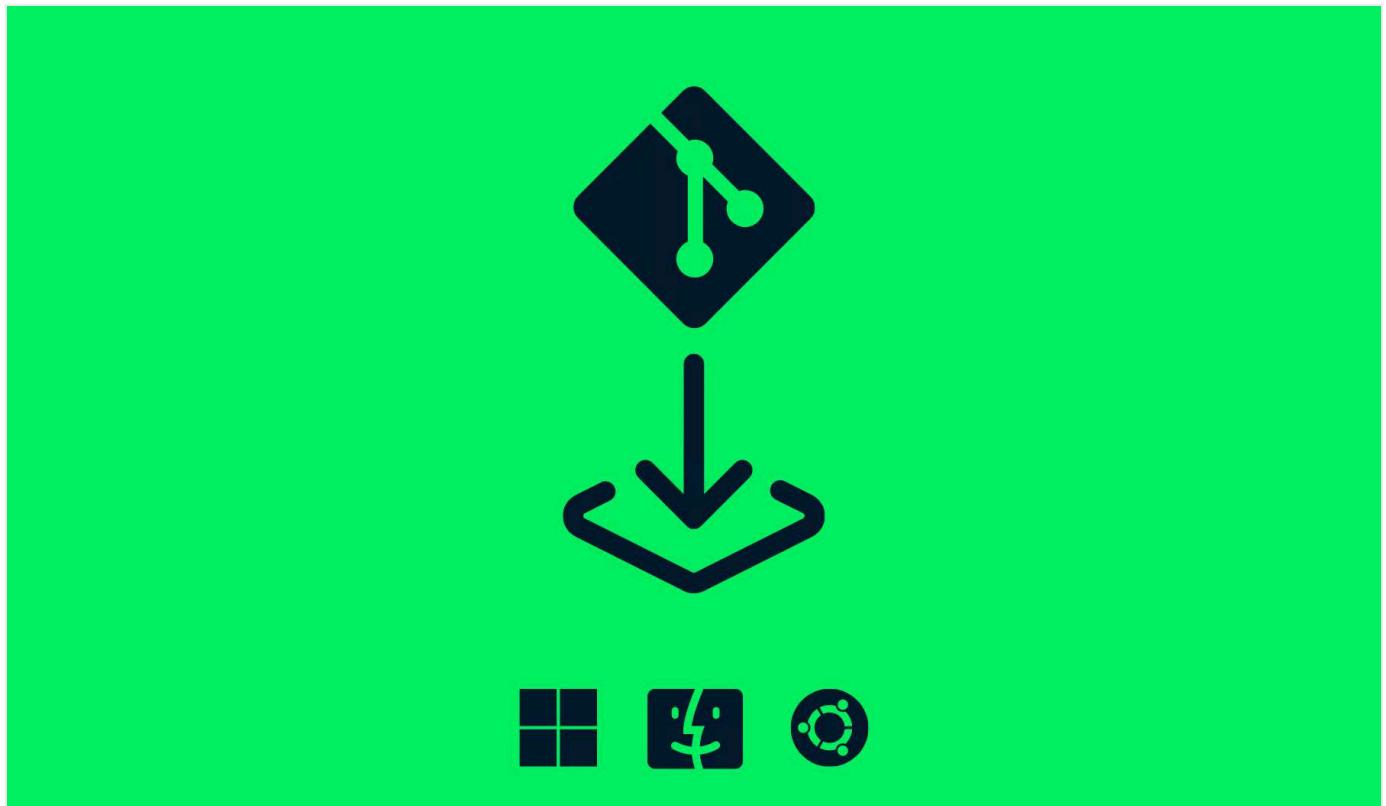


Image by Author

## What is Git

Git is an open-source version control system, which data scientists use to track changes in code, models, datasets, and metadata. Git allows us to merge changes and revert to older versions of code and data. It also lets us sync changes with a remote server. Due to its popularity, Git has become an industry standard, and most development programs (VS Code, JupyterLab, Atom) come with a built-in integration of Git and GitHub. Learn more about Git by reading this [Intro to Git and GitHub Tutorial](#).

In this tutorial, we will learn how to install Git on macOS, Windows, and Linux using command-line tools and a binary installer. We will also learn how to install Git large file storage (Git LFS), and set up Git configurations and Git GUI applications for beginners.

## Install Git on macOS X

macOS [Xcode](#) comes with built-in Git and GitHub features; and if you are not an Xcode user, you can install Git with Homebrew, MacPort, or a binary installer.

### Install Git with Homebrew

[Homebrew](#) is a package manager for macOS. You can install, modify, and uninstall applications using a simple command in the terminal.

First, you need to install Homebrew, then type the command into the terminal.

```
brew install git
```



Explain code

POWERED BY databl

To Install [git-gui](#) and [gitk](#) for a user-friendly experience and browsing interactive commit history, type in the command below.

```
brew install git-gui
```



Explain code

POWERED BY databl

## Install Git with MacPort

MacPorts, a software manager for macOS systems, is similar to Homebrew.

If you have [MacPorts](#) installed, then:

```
sudo port install git
```



Explain code

POWERED BY databl

## Install Git with Mac Installer

You can also install Git using the [installer](#) by Tim Harper. Download and install the latest setup from [SourceForge.net](#).

By following the installation instructions and keeping everything in its default setting, you can install the Git package within a few minutes.

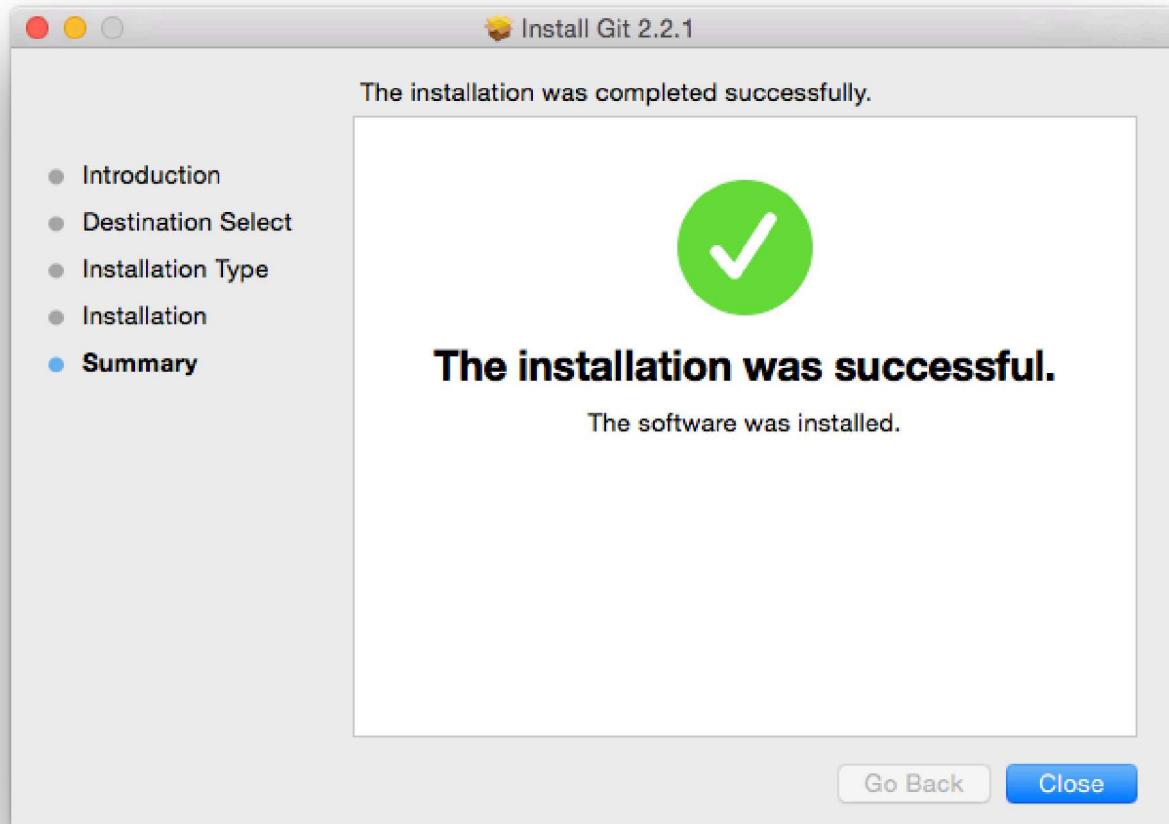


Image from [SourceForge.net](#)

## Install Git on Linux

Linux is the most popular operating system (OS) among developers, and the majority of web servers are running on Linux/Unix-based systems. In this section, we will learn to install Git on various versions of OS using command-line tools.

### Debian and Ubuntu

We will update apt-get and install the latest version of Git. apt-get is a command-line tool that allows you to modify, install, update, and remove the software.

```
sudo apt-get update  
sudo apt-get install git
```



Explain code

POWERED BY datacamp

### Fedora

To install Git on older versions (up to 21) of Fedora use:

```
yum install git
```



Explain code

POWERED BY databricks

For the latest versions (22+) use:

```
dnf install git
```



Explain code

POWERED BY databricks

## Other Linux and Unix based OS

There is a long list of various versions for Linux/Unix-based systems, and you can use the commands below to install Git on almost every OS.

- Gentoo: emerge --ask --verbose dev-vcs/git
- Arch Linux: pacman --ask --verbose dev-vcs/git
- openSUSE: zypper install git
- Mageia: urpmi git
- Nix/NixOS: nix-env -i git
- FreeBSD: pkg install git
- Solaris 11 Express: pkg install developer/versioning/git
- OpenBSD: pkg\_add git
- Alpine: apk add git
- Slitaz: tazpkg get-install git

## Install Git on Windows

Installing Git on Windows is fairly straightforward. In this section, we will learn how to install Git using Winget, Chocolatey, and a binary installer.

### Command line

[Winget](#) package installers come by default with Windows 11, so it is quite easy for users to run the command on PowerShell. You can also install it on older versions of Windows from the [Microsoft store](#).

```
winget install --id Git.Git -e --source winget
```



Explain code

POWERED BY datacamp

## Chocolatey

**Chocolatey** is a package manager for Windows. If you have **installed** it on your system, then simply run the command on PowerShell.

```
choco install git
```



Explain code

POWERED BY datacamp

## Install Git with Windows installer

**Download** the latest (2.35.2) 64-bit version of Git setup. If you want to download a portable or 32-bit version of setup, check out the Git download **page**. After downloading the Windows binary setup, it is time to run it.



## Window Git Install

Click on the install button and follow the instructions to extract the package. The setup will take a few minutes, and it will also install Git LFS, Git Bash, and Git GUI.

Keep options at default. You can remove the installation of Git Bash and other supportive applications by clicking on the checkmark. Make sure the default editor is either VS Code or Atom. You can set the default Git branch later, but it is easy to set it within the installer. In the next step, set Git to be accessed by the command line and other 3rd party software. Use the default option OpenSSL library and Git Credential Manager Core for a better experience.

END OF YEAR  
**SALE** 50% OFF

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

Use Atom as Git's default editor  
Use Visual Studio Code as Git's default editor  
Use Visual Studio Code Insiders as Git's default editor  
Use Sublime Text as Git's default editor  
Use Atom as Git's default editor  
Use VSCode as Git's default editor  
Use Notepad as Git's default editor  
Use Wordpad as Git's default editor  
Select other editor as Git's default editor

Let Git decide  
Let Git use its default branch name (currently: "master") for the initial branch in newly created repositories. The Git project [intends](#) to change this default to a more inclusive name in the near future.

Override the default branch name for new repositories  
NEW! Many teams already renamed their default branches; common choices are "main", "trunk" and "development". Specify the name "git init" should use for the initial branch:  
main

only be able to use the Git command line tools from Git Bash.

**Git from the command line and also from 3rd-party software**  
**(Recommended)** This option adds only some minimal Git wrappers to your PATH to avoid cluttering your environment with optional Unix tools.  
You will be able to use Git from Git Bash, the Command Prompt and the Windows PowerShell as well as any third-party software looking for Git in PATH.

**Use Git and optional Unix tools from the Command Prompt**  
Both Git and the optional Unix tools will be added to your PATH.  
**Warning:** This will override Windows tools like "find" and "sort". Only use this option if you understand the implications.

**Use the OpenSSL library**  
Server certificates will be validated using the ca-bundle.crt file.

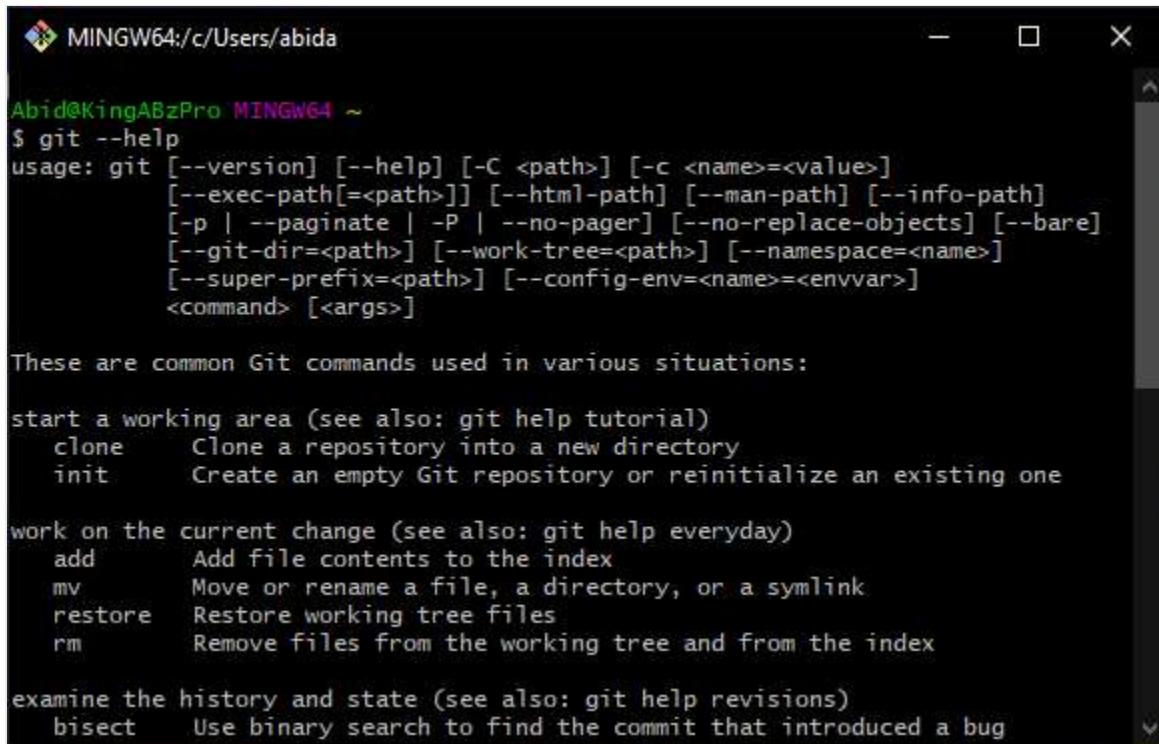
**Use the native Windows Secure Channel library**  
Server certificates will be validated using Windows Certificate Stores.  
This option also allows you to use your company's internal Root CA certificates distributed e.g. via Active Directory Domain Services.

**Git Credential Manager Core**  
**(NEW!)** Use the new, [cross-platform version of the Git Credential Manager](#). See more information about the future of Git Credential Manager [here](#).

**Git Credential Manager**  
**(DEPRECATED)** The [Git Credential Manager for Windows](#) handles credentials e.g. for Azure DevOps and GitHub (requires .NET framework v4.5.1 or later).

## Windows Installer Options

To validate the installation, run the Git Bash and type `git --help`.



```

MINGW64:/c/Users/abida ~
$ git --help
usage: git [--version] [--help] [-C <path>] [-c <name>=<value>]
           [--exec-path[=<path>]] [--html-path] [--man-path] [--info-path]
           [-p | --paginate | -P | --no-pager] [--no-replace-objects] [--bare]
           [--git-dir=<path>] [--work-tree=<path>] [--namespace=<name>]
           [--super-prefix=<path>] [--config-env=<name>=<envvar>]
           <command> [<args>]

These are common Git commands used in various situations:

start a working area (see also: git help tutorial)
  clone      Clone a repository into a new directory
  init       Create an empty Git repository or reinitialize an existing one

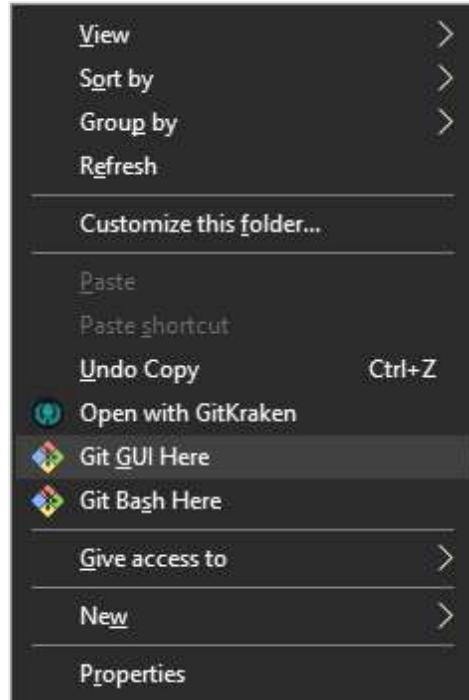
work on the current change (see also: git help everyday)
  add        Add file contents to the index
  mv        Move or rename a file, a directory, or a symlink
  restore    Restore working tree files
  rm        Remove files from the working tree and from the index

examine the history and state (see also: git help revisions)
  bisect    Use binary search to find the commit that introduced a bug

```

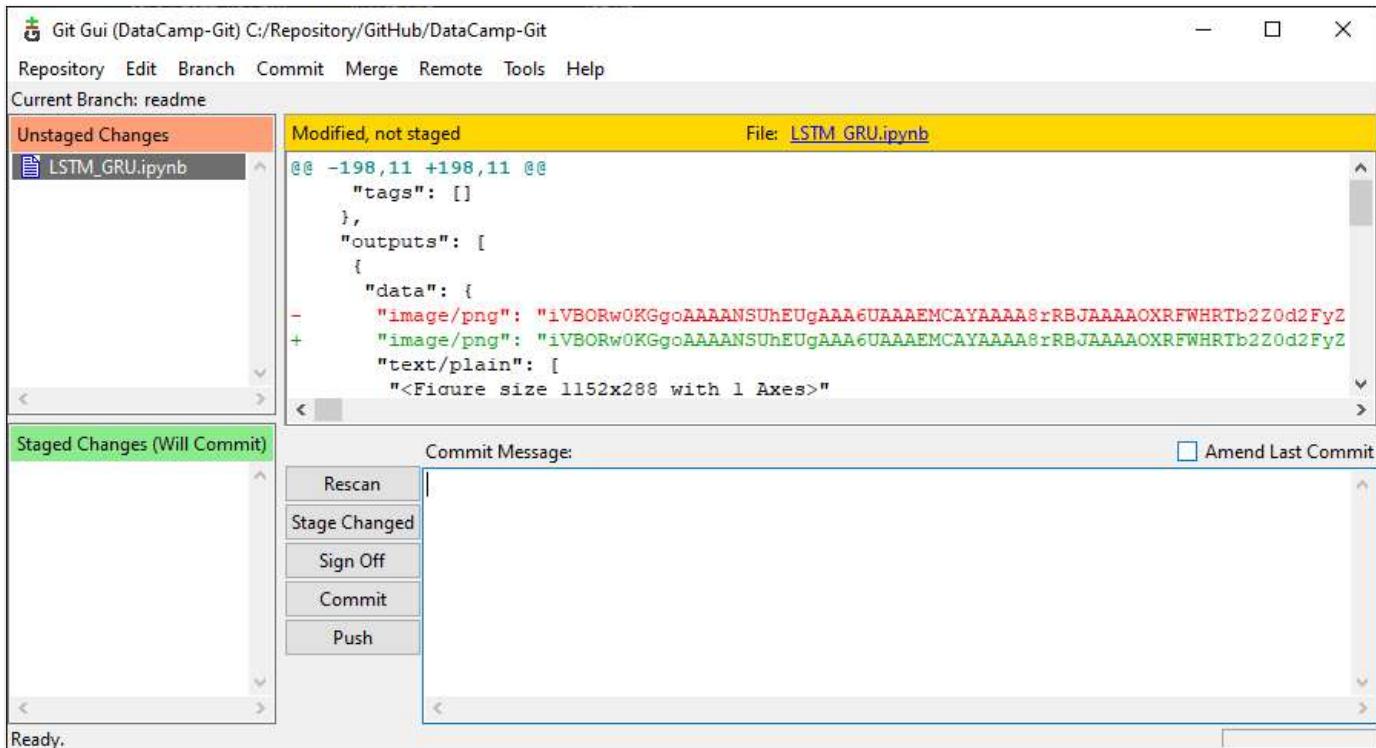
## Git Bash

You will also see Git Bash and Git GUI options when you press the right-click button on your mouse. This is part of the accessibility so that you can open Git Bash in a specific directory containing the Git repository.



## Right-click on Directory

The Git GUI helps visualize branches and historical changes, and instead of typing commands, you will be pressing buttons.



## Git GUI

# Git Initial Setup

It is necessary to configure Git before jumping into creating a repository. Setting up a username and email are necessary as they are yours to sign commits.

```
git config --global user.name "username"
git config --global user.email username@gmail.com
```

★ Explain code

POWERED BY databab

The majority of remote platforms (GitHub) have the main branch as their default, so it is recommended to set the main branch as the default by using:

```
git config --global init.defaultBranch main
```

★ Explain code

POWERED BY databab

Make sure your configurations are successfully added. The command below will show all the configurations – from the remote setup to ssh keys.

```
git config --list
```

 Explain codePOWERED BY  databricks

After early configuration, you are ready to add a file, commit, and push it to GitHub. On your first push, you have to add a username and password. After that, Git will store your credentials so that you don't have to type the password in again.

## Installing Git LFS

**Git Large File Storage (LFS)** is an open-source Git extension for tracking and versioning large files. In data science, it is used for versioning large models, datasets, and metadata. The LSF replaces large datasets with a text pointer inside Git and stores the original files on a remote server. This makes versioning, pulling, and pushing large files faster. It also optimizes the storage.

- Windows: download and install the [package](#).
- Ubuntu: run the command `sudo apt-get install git-lfs`
- macOS: if you have Homebrew installed, then use `brew install git-lfs`
- Other OS: check out GitHub repository [git-lfs/git-lfs](#)

After successfully installing the LFS, run the command below in the terminal to initialize a large file storage system.

```
git lfs install
```



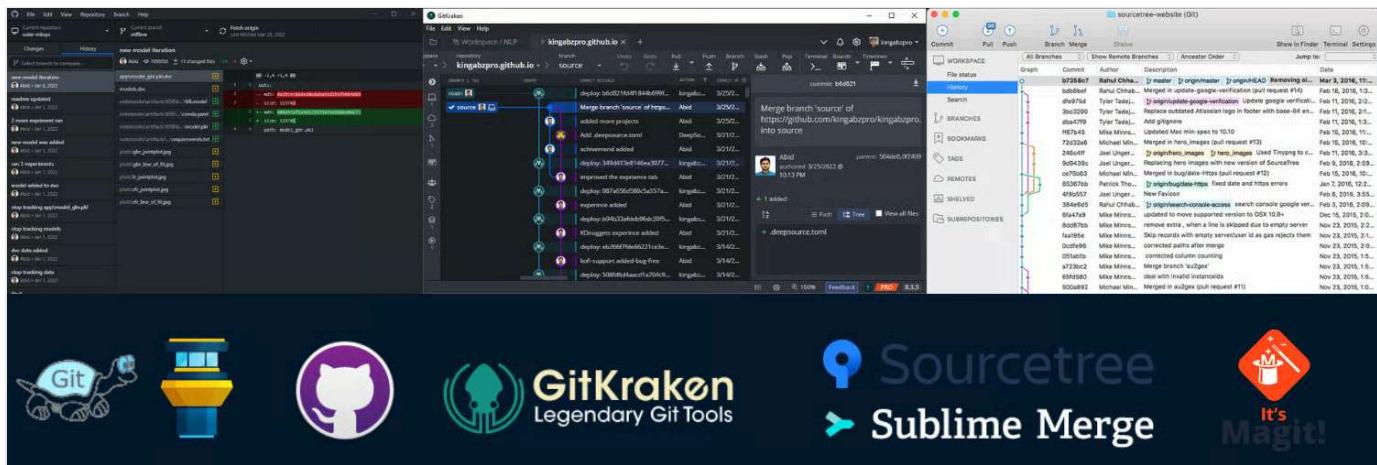
```
>>> Git LFS initialized.
```

 Explain codePOWERED BY  databricks

## Git GUI Clients

Git GUI is for both beginners and experts. Instead of typing commands, you can use a graphical interface to initialize, modify, and push changes to the Git remote server. The visualization features of historical commits are quite useful in tracking the project's progress.

Git GUIs are necessary as they come with additional features such as project management, multiple repository management, team management, visualizing Git branches, tags, submodules, and much more. The best Git GUI client comes with Visual Studio Code as it is fully integrated with the development environment. You can perform all the tasks by clicking on options and avoid making mistakes.



## Git GUI Applications

List of top tier Git GUI clients:

- [GitHub Desktop](#)
- [GitKraken](#)
- [Sourcetree](#)
- [Tower](#)
- [Tortoise Git](#)
- [Sublime Merge](#)
- [Cycligent](#)
- [Cong](#)
- [Aurees](#)
- [Magit](#)
- [GitFiend](#)

## Conclusion

Git has become an essential tool for software developers and data scientists. Even machine learning engineers use it to track the changes in models, datasets, monitor performance metrics, and data pipelines. If you are new to Git and want to understand how it works, then read [All About Git](#). You can also learn by solving 46 coding exercises within DataCamp's [Introduction to Git](#) course.

In this tutorial, we have shown how to install Git on various operating systems. Furthermore, we have discussed how to set up Git configurations, install Git large files storage, and make use of Git graphical user interface.



Git Install | Illustration by [Storyset](#)

## Git Installation FAQ's

### How do I install GitHub Desktop?

Download the installer from the [GitHub Desktop](#) website. GitHub Desktop is an open-source, user-friendly interface for performing Git related tasks. It is highly recommended for beginners.

### How do I install Git on Windows?

### How do I install Git on Mac?

### How do I install Git on Ubuntu?

### How do I install Git on Linux?

### How do I install Git Bash?

### How do I install Git in Visual Studio Code?

How do I install Git inside a Docker container? ▼

How do I install Git LFS? ▼

How do I install Git Lab? ▼

How do I install Git in Fedora? ▼

How do I install the Git command line? ▼

## TOPICS

Git    Data Science



Training more people?

Get your team access to the full DataCamp for business platform.

[For Business](#)

For a bespoke solution [book a demo](#).

## Related

### BLOG

What is Git? - The Complete Guide to Git



### TUTORIAL

Git Setup: The Definitive Guide



### TUTORIAL

GitHub and Git Tutorial for Beginners

[See More →](#)

## Grow your data skills with DataCamp for Mobile

Make progress on the go with our mobile courses and daily 5-minute coding challenges.



### LEARN

[Learn Python](#)

[Learn AI](#)

[Learn Power BI](#)

[Learn Data Engineering](#)

[Assessments](#)

[Career Tracks](#)

[Skill Tracks](#)

[Courses](#)

[Data Science Roadmap](#)

### DATA COURSES

[Python Courses](#)

[R Courses](#)

[SQL Courses](#)

Power BI Courses

Tableau Courses

Alteryx Courses

Azure Courses

AWS Courses

Google Sheets Courses

Excel Courses

AI Courses

Data Analysis Courses

Data Visualization Courses

Machine Learning Courses

Data Engineering Courses

Probability & Statistics Courses

## DATALAB

Get Started

Pricing

Security

Documentation

## CERTIFICATION

Certifications

Data Scientist

Data Analyst

Data Engineer

SQL Associate

Power BI Data Analyst

Tableau Certified Data Analyst

Azure Fundamentals

AI Fundamentals

## RESOURCES

Resource Center

Upcoming Events

Blog

Code-Alongs

Tutorials

Docs

Open Source

RDocumentation

Book a Demo with DataCamp for Business

Data Portfolio

## PLANS

Pricing

For Students

For Business

For Universities

Discounts, Promos & Sales

Expense DataCamp

DataCamp Donates

## FOR BUSINESS

[Business Pricing](#)

[Teams Plan](#)

[Data & AI Unlimited Plan](#)

[Customer Stories](#)

[Partner Program](#)

## ABOUT

[About Us](#)

[Learner Stories](#)

[Careers](#)

[Become an Instructor](#)

[Press](#)

[Leadership](#)

[Contact Us](#)

[DataCamp Español](#)

[DataCamp Português](#)

[DataCamp Deutsch](#)

[DataCamp Français](#)

## SUPPORT

[Help Center](#)

[Become an Affiliate](#)



[Privacy Policy](#)   [Cookie Notice](#)   [Do Not Sell My Personal Information](#)   [Accessibility](#)   [Security](#)  
[Terms of Use](#)

© 2025 DataCamp, Inc. All Rights Reserved.