

INSTALANDO GIT Y GITBASH EN WINDOWS

X

Git 2.26.2 Setup

X

Git 2.26.2 Setup

**Select Components**

Which components should be installed?

**Choosing the default editor used by Git**

Which editor would you like Git to use?

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

• Visitar https://git-scm.com/download/win Otra forma de obtener Git ( • Elegir la versión necesaria, y la descarga empezará automáticamente fácilmente es mediante la instalación de GitHub para

Windows. Puedes descargar este instalador del sitio web de

GitHub para Windows en http://windows.github.com

Use Vim (the ubiquitous text editor) as Git's default editor

The Vim editor, while powerful, can be hard to use. Its user interface is unintuitive and its key bindings are awkward.

Note: Vim is the default editor of Git for Windows only for historical reasons, and it is highly recommended to switch to a modern GUI editor instead.

Additional icons

On the Desktop Windows Explorer integration

Git Bash Here ✓ Git GUI Here V Git LFS Large File Support) ✓ Associate .git\* configuration files with the default text editor ✓ Associate .sh files to be run with Bash ✓ Use a True Type font in all console windows ✓ Check daily for Git for Windows updates

Note: This will leave the 'core.editor' option unset, which will make Git fall back to the 'EDITOR'environment variable. The default editor is Vim - but you may set it to some other editor of your choice.

https://gitforwindows.org/

Current selection requires at least 251.7 MB of disk space. https://gitforwindows.org/ -

<Back

Next >

Cancel

< Back

Next >

Cancel

Git 2.26.2 Setup

х

Git 2.26.2 Setup

**Adjusting your PATH environment**

How would you like to use Git from the command line?

**Choosing HTTPS transport backend**

Which SSL/TLS library would you like Git to use for HTTPS connections?

**Use Git from Git Bash only**

This is the most cautious choice as your PATH will not be modified at all. You will only be able to use the Git command line tools from Git Bash.

**Use the OpenSSL library**

Server certificates will be validated using the ca-bundle.crt file.

**Use the native Windows Secure Channel library**

Una vez culminada la instalación, para verificar

que GIT esta instalado, abrimos en B "GIT BASH" y escribimos el comando git --version para corroborar

la versión instalada.

**O 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.

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.

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. https://gitforwindows.org/

https://gitforwindows.org/

< Back

Next >

Cancel

<Back

Next >

Cancel

Git 2.26.2 Setup

ngeloPaul@Ab7alon MINGW64 -

git Sage: 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] [--barel [--git-dir=<path>] [--work-tree=<path>] [--namespace=<name>] <command> [<args>]

Git 2.26.2 Setup

**Configuring the line ending conversions**

How should Git treat line endings in text files?

**Configuring the terminal emulator to use with Git Bash**

Which terminal emulator do you want to use with your Git Bash?

hese are common Git commands used in various situations :

tart 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 ne

Checkout Windows-style, commit Unix-style line endings Git will convert LF to CRLF when checking out text files. When committing text files, CRLF will be converted to LF. For cross-platform projects, this is the recommended setting on Windows ("core.autocrifis set to "true").

**O Use MinTTY (the default terminal of MSYS2)**

Git Bash will use MinTTY as terminal emulator, which sports a resizable window, non-rectangular selections and a Unicode font. Windows console programs (such as interactive Python) must be launched via winpty to work in MinTTY.

*add*

ork on the current change (see also: git help everyday)

Add file contents to the index Πιν

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 sparse-checkout Initialize and modify the sparse-checkout

Checkout as-is, commit Unix-style line endings Git will not perform any conversion when checking out text files. When committing text files, CRLF will be converted to LF. For cross-platform projects, this is the recommended setting on Unix ("core.autocrlf is set to "input).

kamine the history and state (see also: git help revisions) bisect

Use binary search to find the commit that introduced a bug diff

Show changes between commits, commit and working tree, eta grep

Print lines matching a pattern log

Show commit logs show

Show various types of objects status

Show the working tree status

**Use Windows' default console window**

Git will use the default console window of Windows ("cmd.exe"), which works well with Win32 console programs such as interactive Python or node.js, but has a very limited default scroll-back, needs to be configured to use a Unicode font in order to display non-ASCII characters correctly, and prior to Windows 10 its window was not freely resizable and it only allowed rectangular text selections.

**Checkout as-is, commit as-is**

**Git will not perform any conversions when checking out or committing text files. Choosing this option is not recommended for cross-platform**

projects ("core.autocrlf is set to "false). https://gitforwindows.org/

https://gitforwindows.org/

ow, mark and tweak\_your common history branch

List, create, or delete branches commit

Record changes to the repository merge

Join two or more development histories together rebase

Reapply commits on top of another base tip reset

Reset current HEAD to the specified state switch

Switch branches tag

Create, list, delete or verify a tag object signed with GPG

<Back

Next >

Cancel

<Back

Next >

Cancel

bllaborate (see also: git help workflows) fetch

Download objects and refs from another repository pull

Fetch from and integrate with another repository or a local branch

Update remote refs along with associated objects

Git 2.26.2 Setup

Git 2.26.2 Setup

push

**Installing**

Please wait while Setup installs Git on your computer.

**Configuring extra options**

Which features would you like to enable?

**hit help -a' and 'git help -g' list available subcommands and some oncept guides. See 'git help <command>' or 'git help <concept> bread about a specific subcommand or concept. ze 'git help git' for an overview of the system.**

hing

ngeloPaul@Ab7alon MINGW64 -

git --version it version 2.26.2.windows. 1

Extracting files... C:\Program Files\Gitmingw64\bin Vibgnutis-30.dll

**→ Enable file system caching**

File system data will be read in bulk and cached in memory for certain operations ("core.fscache is set to "true"). This provides a significant performance boost.

ngeloPaul@ab7alon MINGW54 ~

**Enable Git Credential Manager**

The Git Credential Manager for Windows provides secure Git credential storage for Windows, most notably multi-factor authentication support for Visual Studio Team Services and GitHub. (requires .NET Framework v4.5.1 or later).

Enable symbolic links Enable symbolic links (requires the SeCreateSymbolichink permission). Please note that existing repositories are unaffected by this setting.

https://gitforwindows.org/

https://gitforwindows.org/ -

Cancel

Cancel | END

<Back

Install

Cancel

Aunque los programadores utilizan sistemas basados en 'UNIX' (tales como GNU/Linux, Mac OS, etc.), Git también se puede instalar y ejecutar en Windows. Existe un proyecto llamado msysgit, que se puede descargar desde el sitio oficial de GIT e incluye GIT BASH (terminal adaptada para Windows) yotros, que nos van a permitir trabajar y aprovechar

todas las funcionalidades de este sistema.

į

git