



Entornos de Programación

Diplomado en Ciencia de Datos:
Aplicaciones con Machine Learning





Entornos de trabajo para programación

Conjunto de **herramientas y configuraciones** que permiten a un desarrollador, escribir, probar, depurar y ejecutar código. Un entorno de trabajo incluye varios componentes como:

- Editor de texto.
- IDE.
- Intérpretes y compiladores.
- Gestor de paquetes.
- Sistema de control de versiones.
- Librerías especializadas.

A photograph of a server rack filled with server units. In the foreground, a monitor displays a terminal window with several lines of code, likely Python, including imports like numpy and tensorflow. The server units have various ports and lights visible.

Importancia de un entorno bien configurado

Un entorno de programación bien configurado es fundamental para la productividad y la eficiencia en la ciencia de datos por razones tales como:

- Reproducibilidad.
- Gestión de Dependencias.
- Eficiencia en el Desarrollo.
- Colaboración.
- Escalabilidad y Flexibilidad.
- Implementación en contextos de producción.

Algunos conceptos antes de empezar

- **Editor de texto:** Herramienta básica usada para escribir y editar código fuente. Los editores de texto están diseñados para manejar texto sin formato.
- **Compilador / Intérprete:** Herramienta para traducir el código escrito en un lenguaje de programación a un formato que la máquina pueda ejecutar. Un compilador toma el código completo en un archivo antes de ejecutarlo (ej, C++, Java), mientras que un intérprete ejecuta línea por línea en el código y lo ejecuta directamente (ej, Python, R).



Algunos conceptos antes de empezar

- **Gestor de paquetes:** Herramienta que automatiza el proceso de instalación, actualización, configuración y eliminación de bibliotecas y dependencias en un entorno de programación (ej, pip, conda, mamba).
- **Sistema de Control de Versiones:** Herramienta que rastrea y gestiona los cambios realizados en el código fuente a lo largo del tiempo (ej, Git y GitHub).
- **Librerías:** Colecciones de código preescrito que proporcionan funcionalidades específicas que pueden ser reutilizadas en proyectos de ciencia de datos.



Entornos de Desarrollo Integrado (IDE)

Un Entorno de Desarrollo Integrado (IDE, por sus siglas en inglés) es una aplicación de software que proporciona un conjunto completo de herramientas para el desarrollo de software en un único entorno unificado. Los IDEs incluyen características como:

- Editor de código.
- Depurador.
- Compilador o intérprete.
- Gestor de proyectos
- Integración con sistemas de control de versiones.

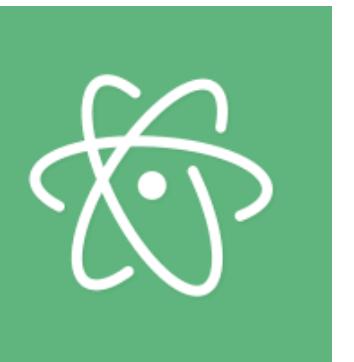
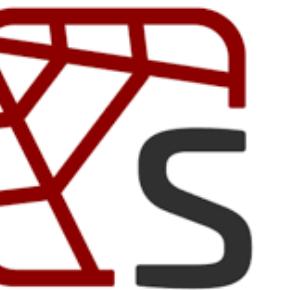


Entornos de Desarrollo Integrado (IDE)

Un Entorno de Desarrollo Integrado (IDE, por sus siglas en inglés) es una aplicación de software que proporciona un conjunto completo de herramientas para el desarrollo de software en un único entorno unificado.



Visual Studio Code



Entornos de Desarrollo en la Nube

Son plataformas de desarrollo en la nube que permite escribir y ejecutar código directamente en un navegador web, generalmente optimizados para trabajar en formatos tipo notebook.



Deepnote

kaggle

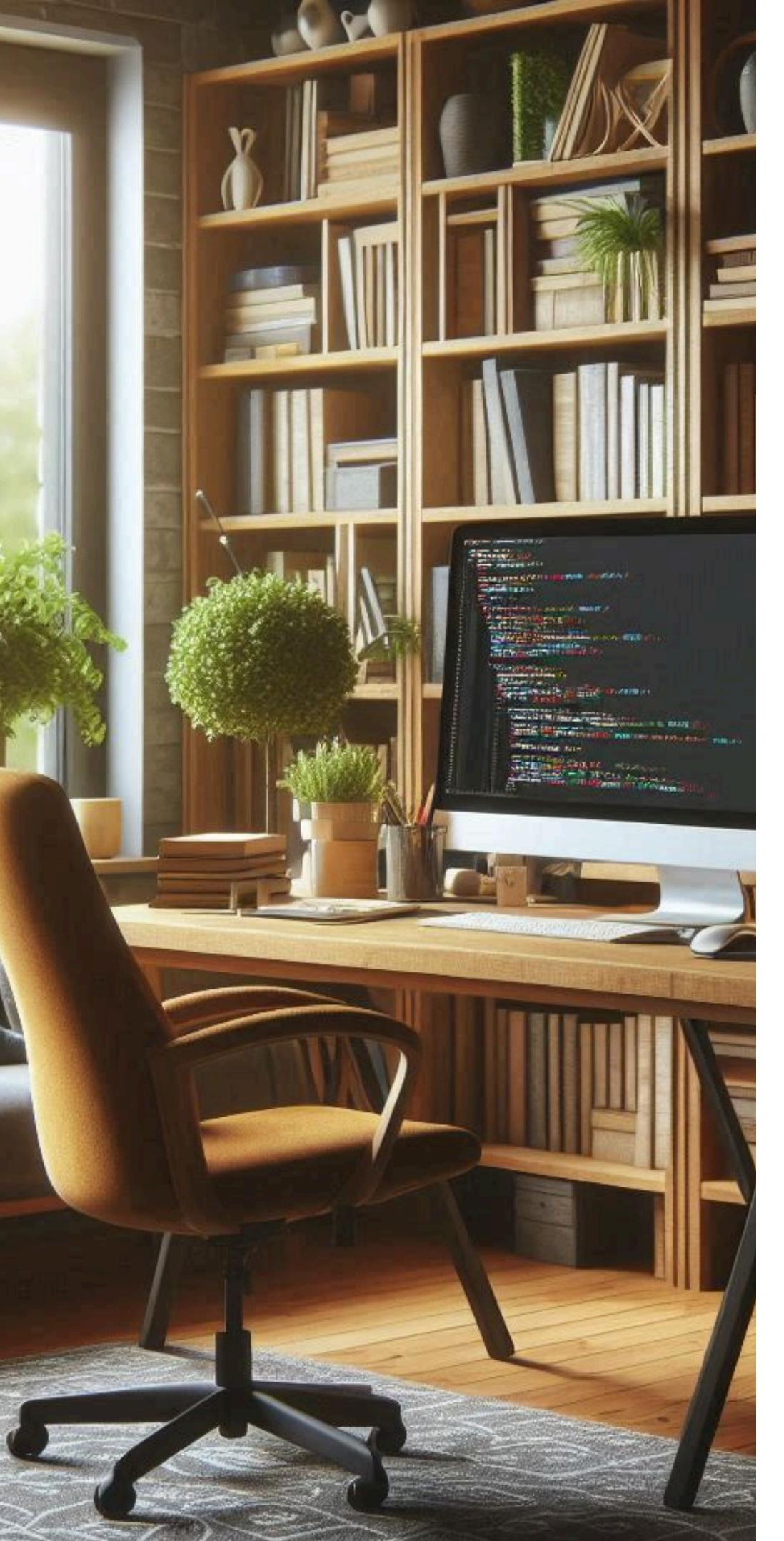


Entornos de Desarrollo en la Nube

Son plataformas de desarrollo en la nube que permite escribir y ejecutar código directamente en un navegador web, generalmente optimizados para trabajar en formatos tipo notebook. Algunos de los beneficios de la programación en la nube son:

- Entornos basados en Jupyter notebooks.
- Acceso a recursos de computación como CPU, GPU y TPU.
- Integración con servicios de almacenamiento.
- Preconfiguraciones de librerías.
- Gestión de paquetes.
- Colaboración en tiempo real.





¿Que herramientas usaremos?



Visual Studio Code



kaggle

 ANACONDA®





Instalación de Anaconda



Distribution

Free Download*

Register to get everything you need to get started on your workstation including Cloud Notebooks, Navigator, AI Assistant, Learning and more.

- ✓ Easily search and install thousands of data science, machine learning, and AI packages
- ✓ Manage packages and environments from a desktop application or work from the command line
- ✓ Deploy across hardware and software platforms
- ✓ Distribution installation on Windows, MacOS, or Linux

Provide email to download Distribution

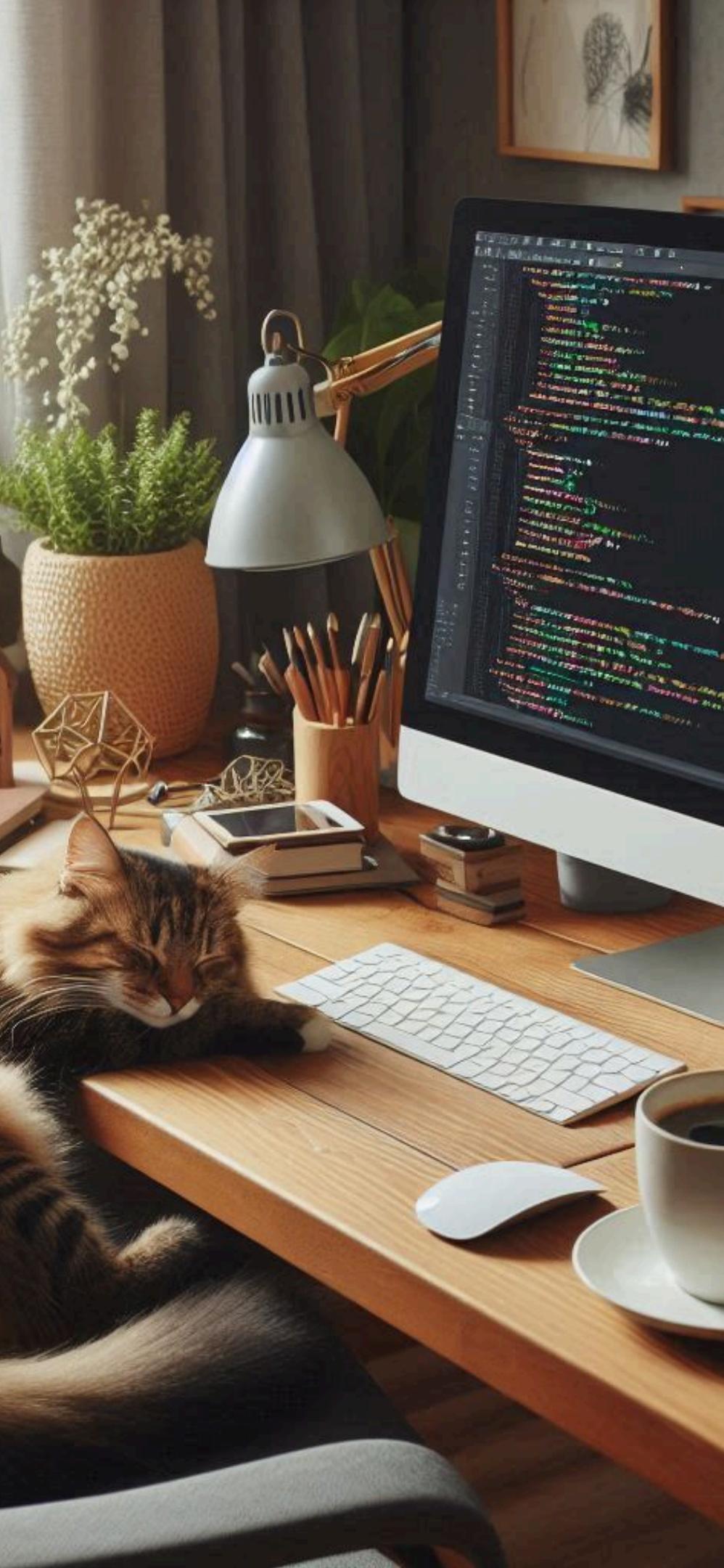
Email Address:

Agree to receive communication from Anaconda regarding relevant content, products, and services. I understand that I can revoke this consent [here](#) at any time.

By continuing, I agree to Anaconda's [Privacy Policy](#) and [Terms of Service](#).

Submit >

Skip registration



Instalación de Anaconda



Anaconda Distribution Download

Thank you for choosing Anaconda and Conda packages.

Continue with your download.

[Download Now](#)

© Copyright 2024 Anaconda, Inc. All Rights Reserved.

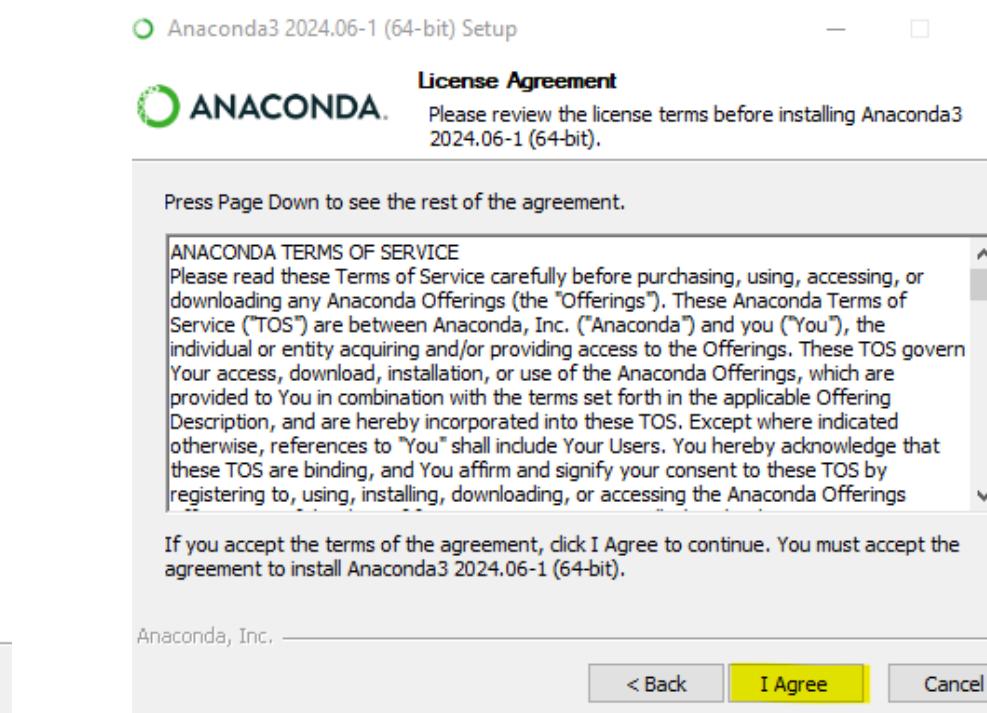


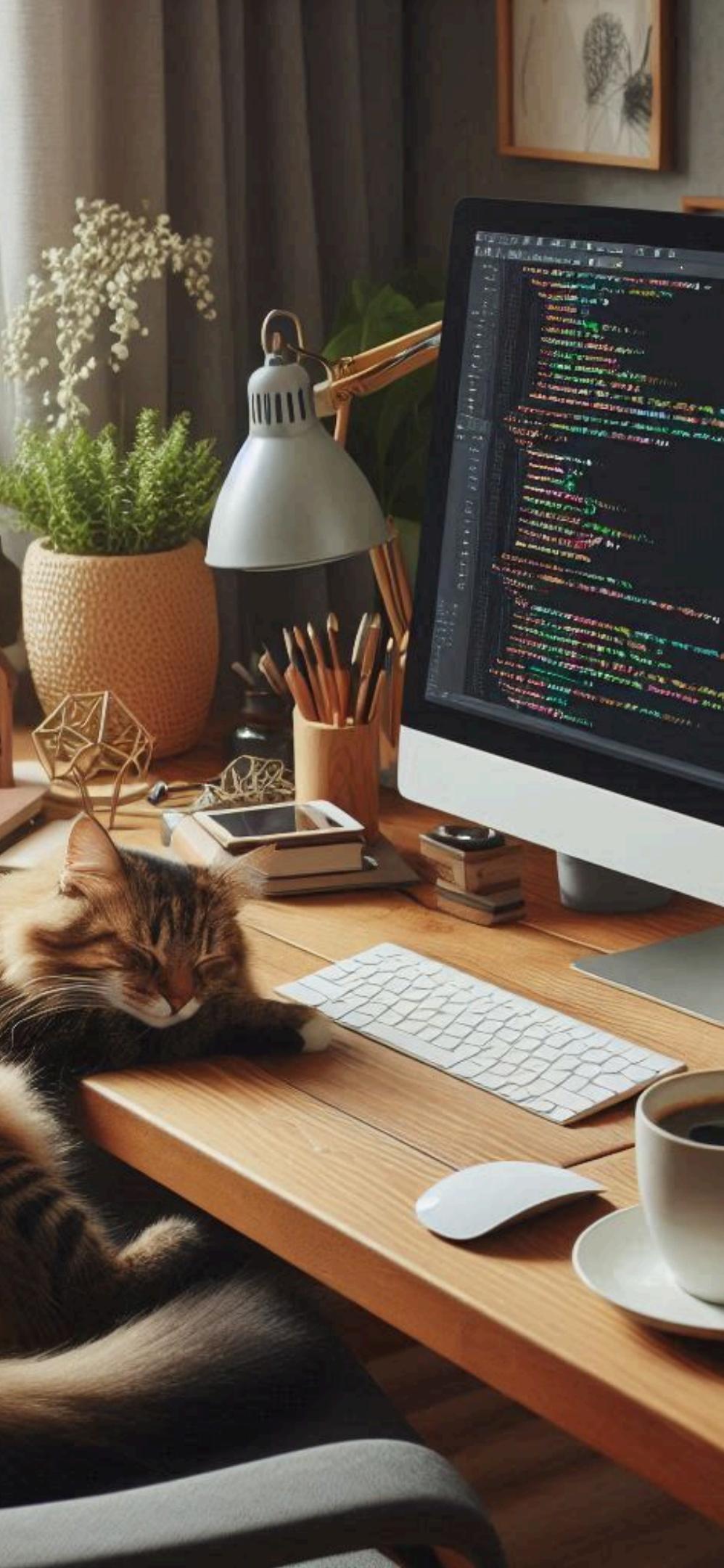
Download Now

For installation assistance, refer to [Troubleshooting](#).

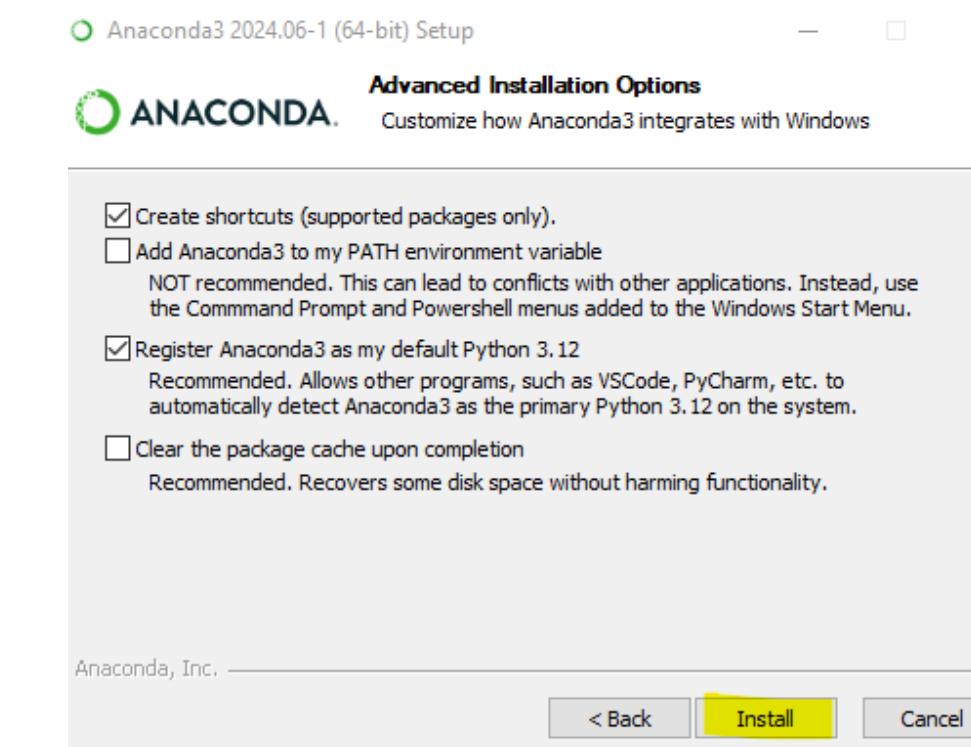
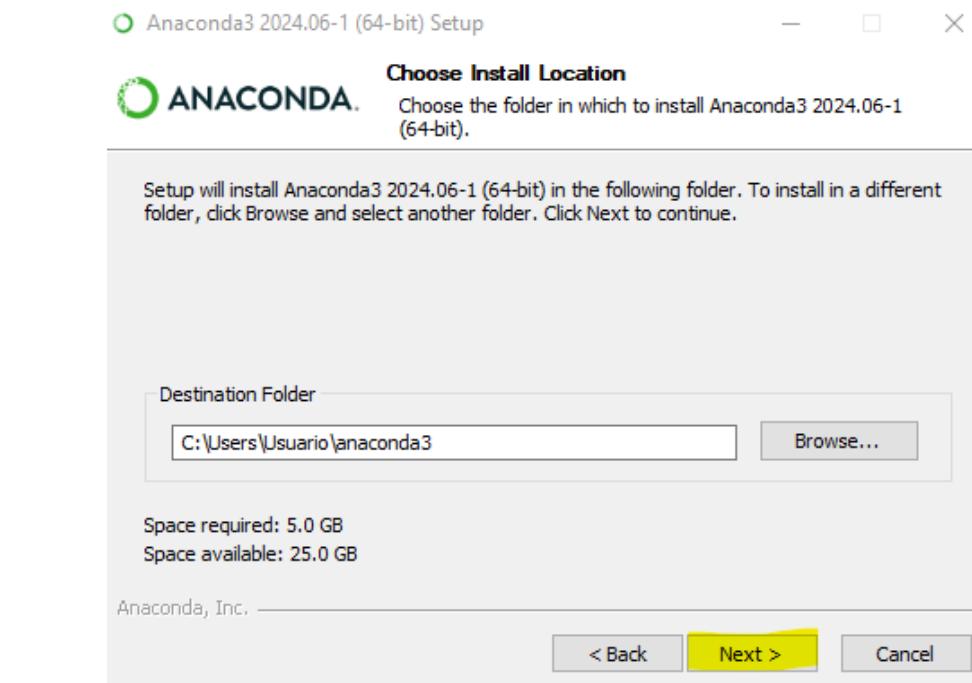
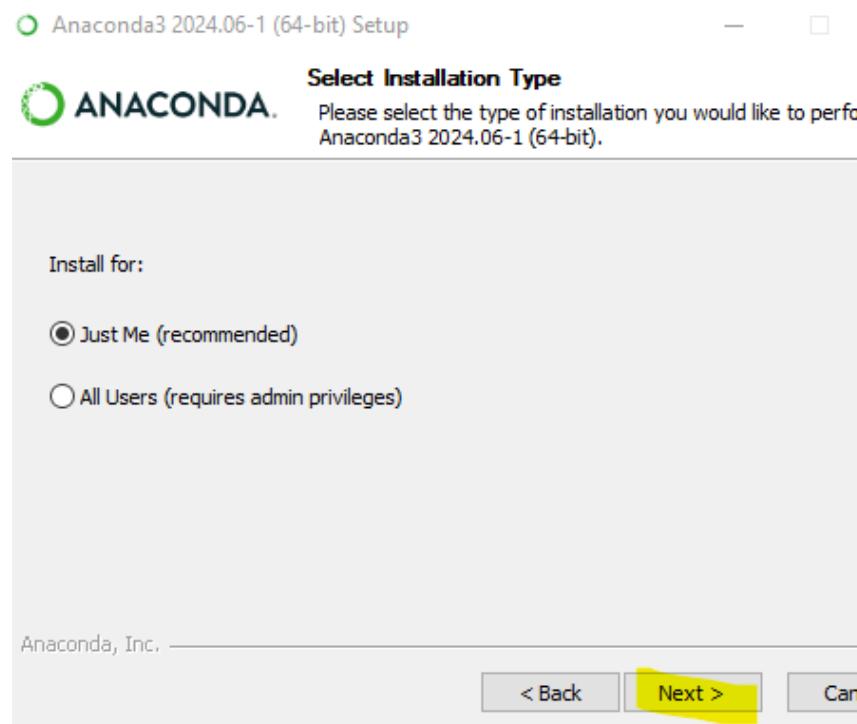
Download Distribution by choosing the proper installer for your machine.

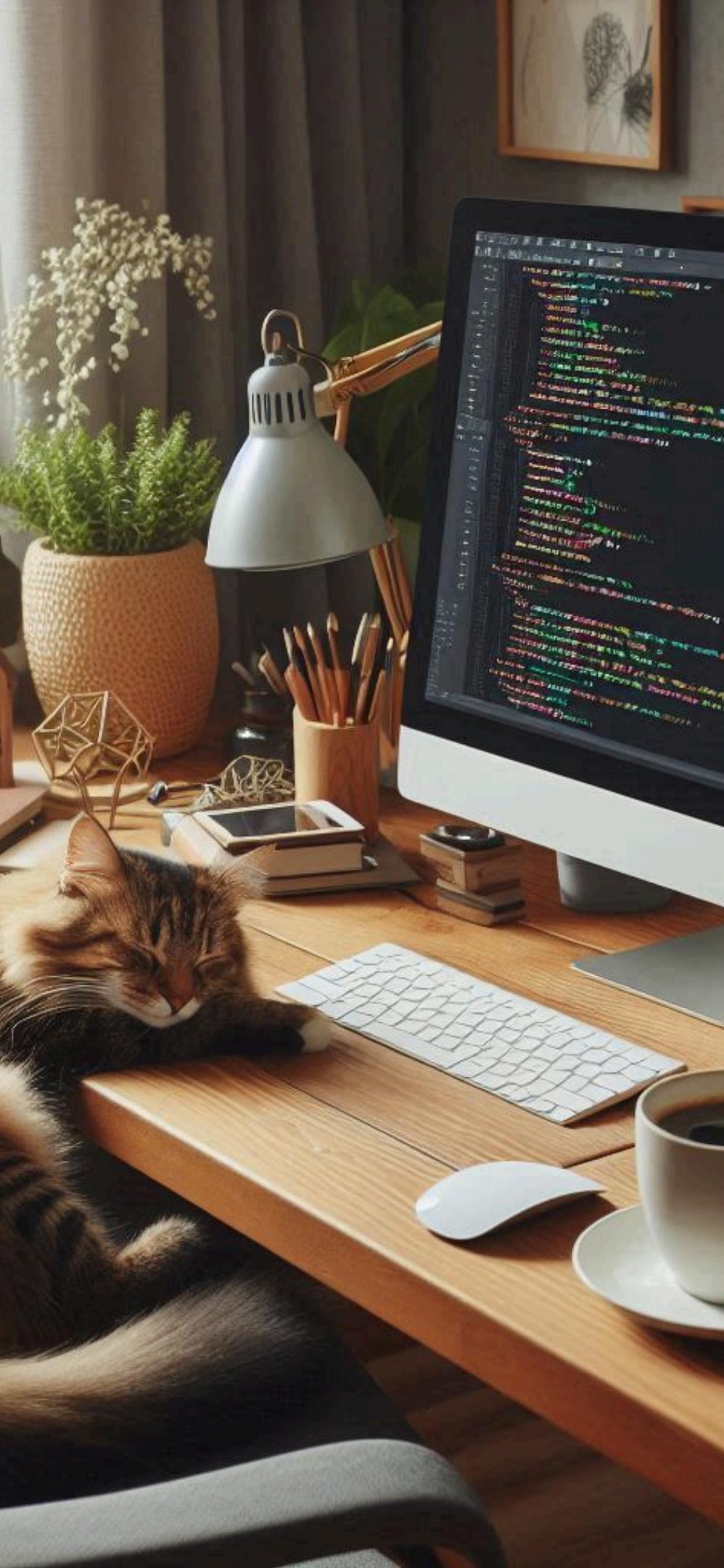
[Download](#)





Instalación de Anaconda





Instalación de Visual SC

Visual Studio Code Docs Updates Blog API Extensions FAQ Search Docs Download

Download Visual Studio Code

Free and built on open source. Integrated Git, debugging and extensions.

Windows Windows 10, 11
User Installer x64 Arm64
System Installer x64 Arm64
zip x64 Arm64
CLI x64 Arm64

.deb Debian, Ubuntu
.rpm Red Hat, Fedora, SUSE
.tar.gz x64 Arm32 Arm64
Snap Snap Store

.zip Intel chip Apple silicon Universal
CLI x64 Arm32 Arm64

Instalar - Microsoft Visual Studio Code (User)

Acuerdo de Licencia

Es importante que lea la siguiente información antes de continuar.

Por favor, lea el siguiente acuerdo de licencia. Debe aceptar las cláusulas de este acuerdo antes de continuar con la instalación.

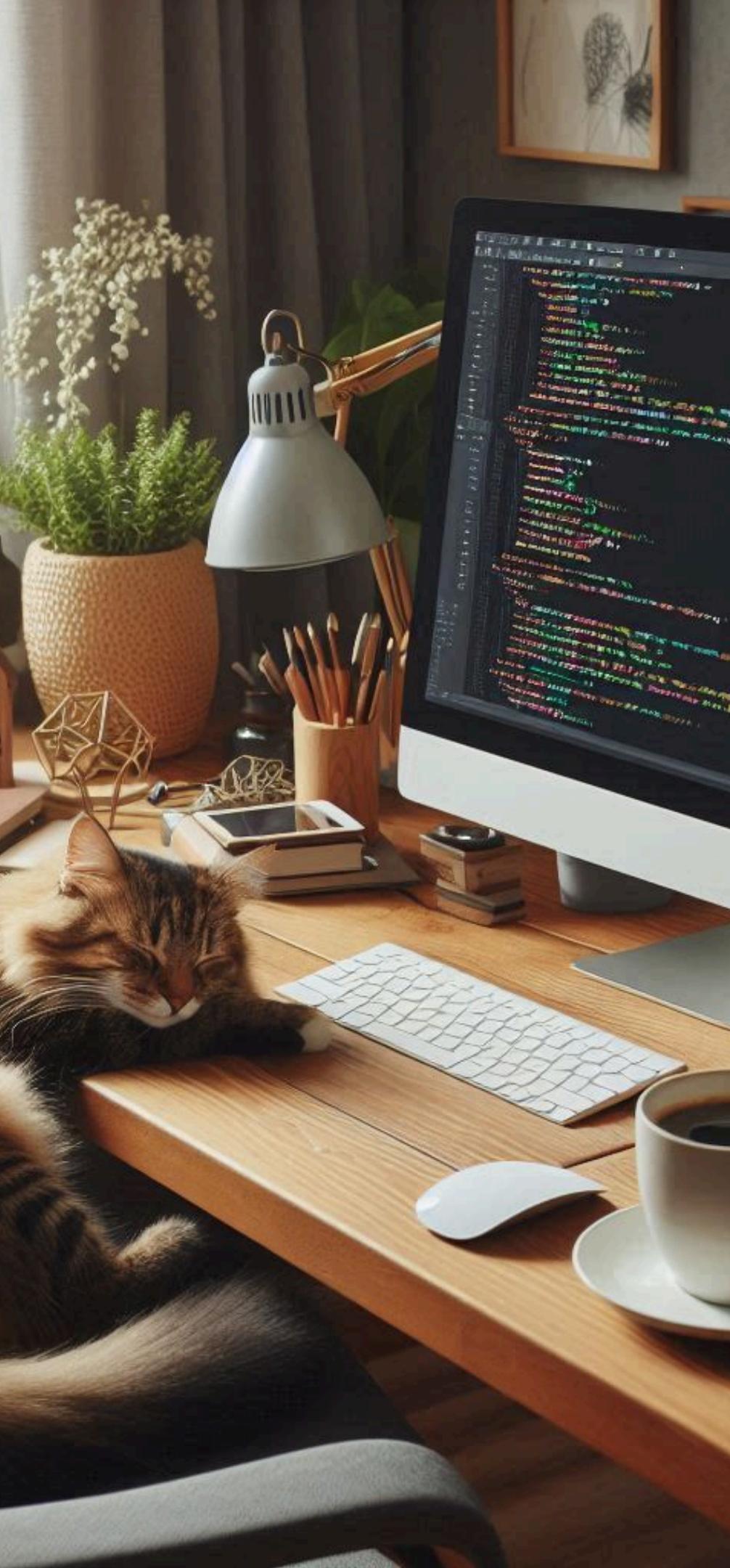
Esta licencia se aplica al producto Visual Studio Code. El código fuente para Visual Studio Code está disponible en <https://github.com/Microsoft/vscode> según el acuerdo de licencia del MIT en <https://github.com/microsoft/vscode/blob/main/LICENSE.txt>. Encontrará información adicional sobre licencias en nuestras preguntas frecuentes en <https://code.visualstudio.com/docs/supporting/faq>.

TÉRMINOS DE LICENCIA DEL SOFTWARE DE MICROSOFT

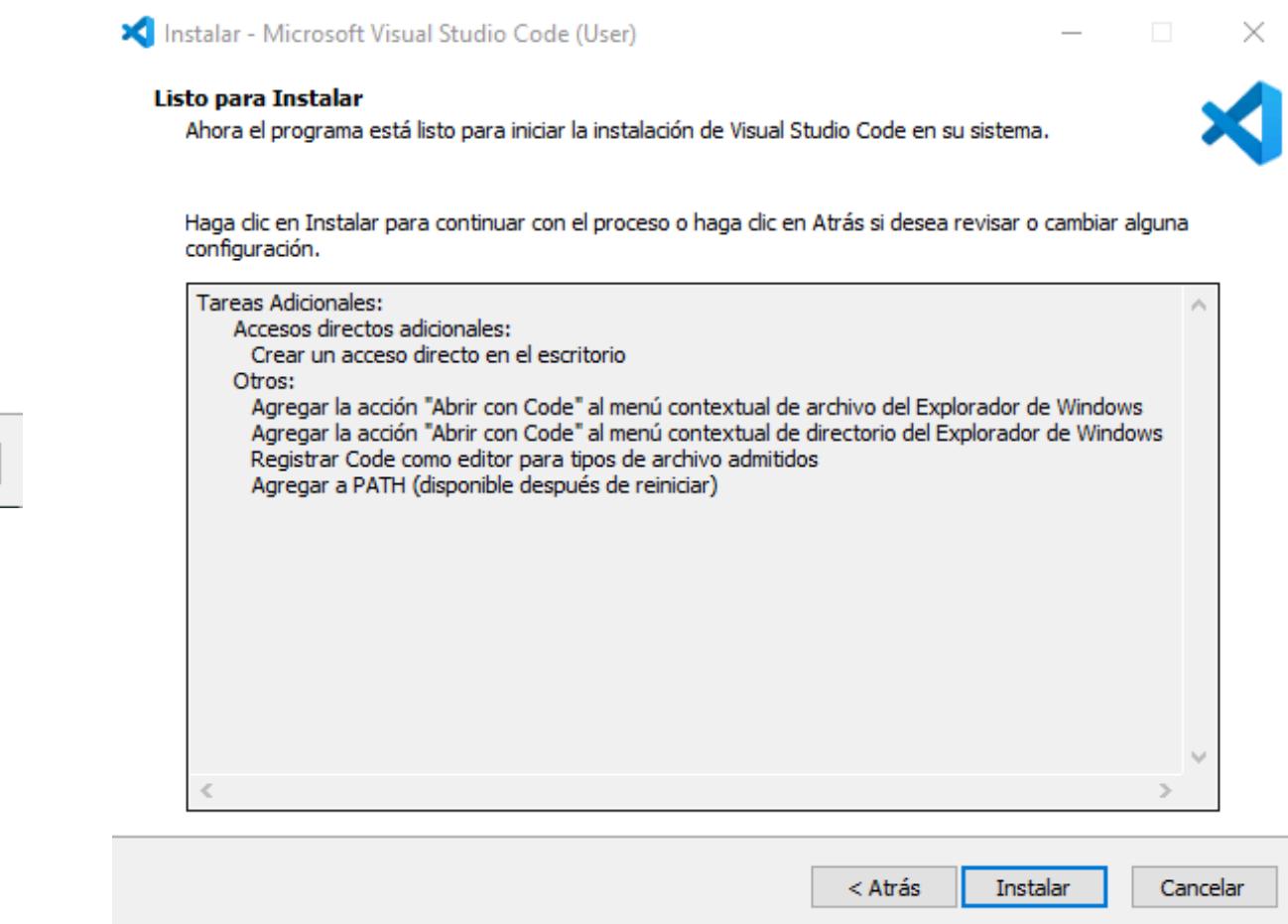
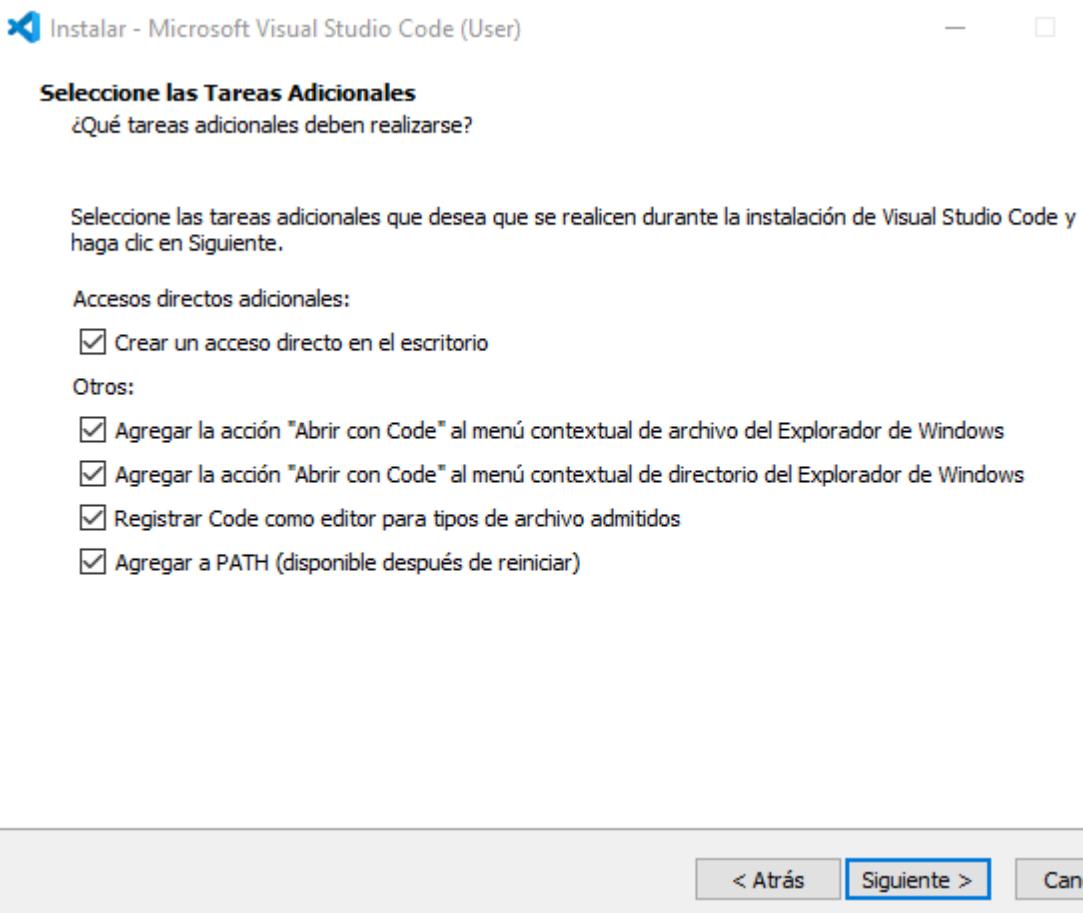
VISUAL STUDIO CODE DE MICROSOFT

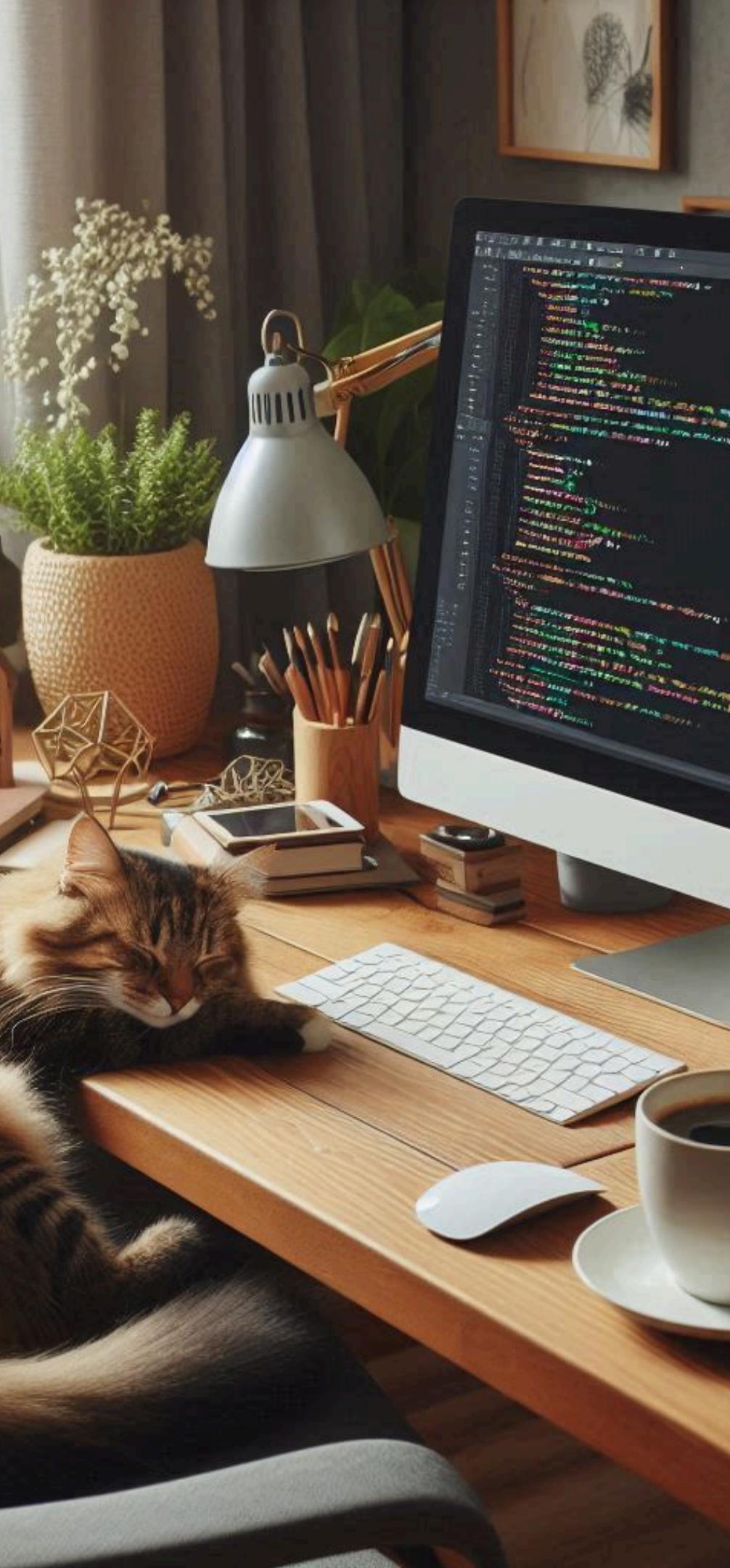
Acepto el acuerdo
 No acepto el acuerdo

Siguiente > Cancelar



Instalación de Visual SC





Extensiones en Visual SC

The screenshot shows the Visual Studio Code interface with the Extensions sidebar open. The search bar in the sidebar is highlighted with a blue box. The main panel displays the details for the "Python" extension, version v2024.12.3, by Microsoft. The extension has 133,192,360 installs and a 5-star rating from 595 reviews. It provides Python language support with IntelliSense (Pylance), debugging (Python Debugger), formatting, linting, code navigation, refactoring, variable explorer, test explorer, and more. The "DETAILS" tab is selected. The sidebar also shows a list of installed extensions, including Python, Python Debugger, Python Environment Manager, Python Extension Pack, Python Indent, Rainbow-Brackets, Remote - SSH, and Remote - SSH: Editing Configuration Files.

EXTENSIONS

Search Extensions in Marketplace

INSTALLED

- Python
- Python Debugger
- Python Environment Manager
- Python Extension Pack
- Python Indent
- Rainbow-Brackets
- Remote - SSH
- Remote - SSH: Editing Configuration Files

RECOMMENDED

Extension: Python X

Python v2024.12.3

Microsoft microsoft.com | 133,192,360 | ★★★★☆(595)

Python language support with extension access points for IntelliSense (Pylanc...

Disable Uninstall Switch to Pre-Release Version Auto Update

DETAILS FEATURES CHANGELOG EXTENSION PACK

Python extension for Visual Studio Code

A Visual Studio Code extension with rich support for the Python language (for all actively supported Python versions), providing access points for extensions to seamlessly integrate and offer support for IntelliSense (Pylance), debugging (Python Debugger), formatting, linting, code navigation, refactoring, variable explorer, test explorer, and more!

Support for vscode.dev

The Python extension does offer some support when running on vscode.dev (which includes github.dev). This includes partial IntelliSense for open files in the editor.

Installed extensions

The Python extension will automatically install the following extensions by

Categories

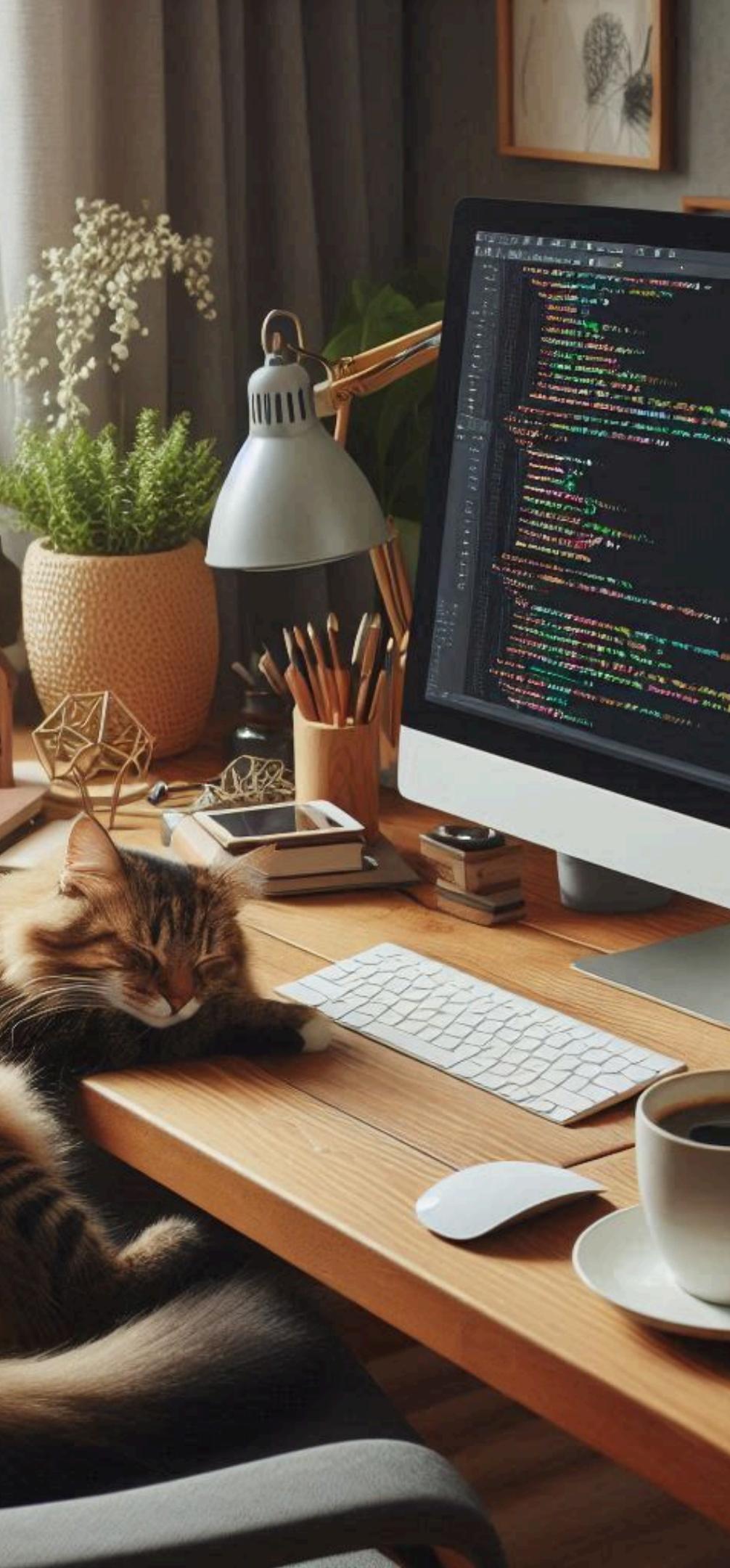
- Programming Languages
- Debuggers
- Data Science
- Machine Learning

Resources

- Marketplace
- Issues
- Repository
- License
- Microsoft

More Info

Published 2016-01-19, 10:03:11



Instalación de Git

 **git** --everything-is-local

Type / to search entire site...

Git is a [free and open source](#) distributed version control system designed to handle everything from small to very large projects with speed and efficiency.

Git is [easy to learn](#) and has a [tiny footprint with lightning fast performance](#). It outclasses SCM tools like Subversion, CVS, Perforce, and ClearCase with features like [cheap local branching](#), convenient staging areas, and [multiple workflows](#).



About
The advantages of Git compared to other source control systems.

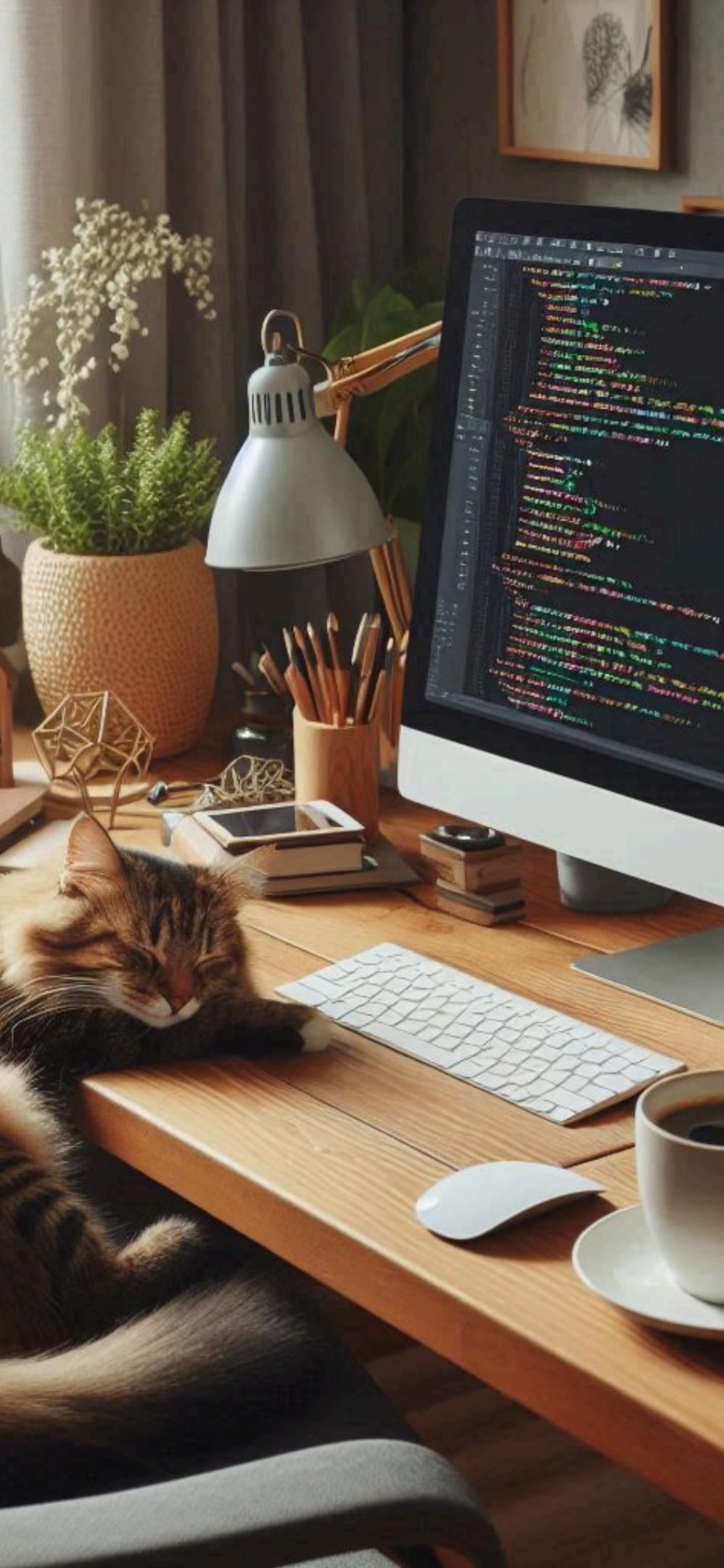
Downloads
GUI clients and binary releases for all major platforms.

Documentation
Command reference pages, Pro Git book content, videos and other material.

Community
Get involved! Bug reporting, mailing list, chat, development and more.

Latest source Release
2.46.0
[Release Notes \(2024-07-29\)](#)

[Download for Windows](#)



Instalación de Git

git --local-branching-on-the-cheap

Type / to search entire site...

About

Documentation

Downloads

- GUI Clients
- Logos

Community

The entire [Pro Git book](#) written by Scott Chacon and Ben Straub is available to [read online for free](#). Dead tree versions are available on [Amazon.com](#).

Download for Windows

[Click here to download](#) the latest (2.46.0) 64-bit version of GIT for Windows. This is the most recent maintained build. It was released 23 days ago, on 2024-07-29.

Other Git for Windows downloads

- Standalone Installer
- [32-bit Git for Windows Setup](#).
- [64-bit Git for Windows Setup](#).
- Portable ("thumbdrive edition")
- [32-bit Git for Windows Portable](#).
- [64-bit Git for Windows Portable](#).

Using winget tool

Install [winget tool](#) if you don't already have it, then type this command in command prompt or Powershell.

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

The current source code release is version 2.46.0. If you want the newer version, you can build it from the [source code](#).

Git 2.46.0 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

<https://gitforwindows.org/>

Next Cancel

Git 2.46.0 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 339,6 MB of free disk space is required.

<https://gitforwindows.org/>

Back Next Cancel

Git 2.46.0 Setup

Select Components

Which components should be installed?

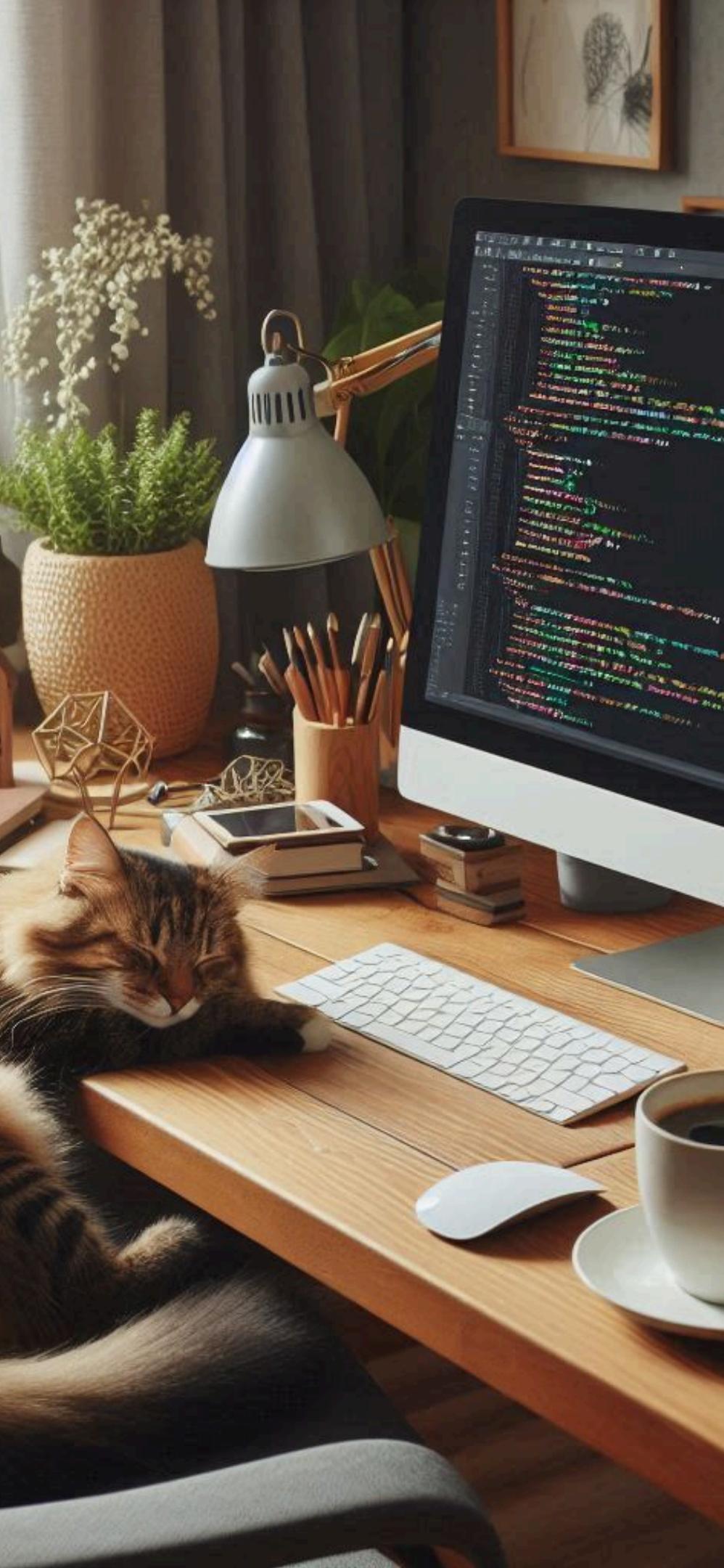
Select the components you want to install; clear the components you do not want to install. Click Next when you are ready to continue.

- Additional icons
 - On the Desktop
- Windows Explorer integration
 - Open Git Bash here
 - Open 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
- Check daily for Git for Windows updates
- (NEW!) Add a Git Bash Profile to Windows Terminal

Current selection requires at least 339,6 MB of disk space.

<https://gitforwindows.org/>

Back Next Cancel



Instalación de Git

Git 2.46.0 Setup

Choosing the default editor used by Git
Which editor would you like Git to use?

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.

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/>

Back Next Cancel

Git 2.46.0 Setup

Adjusting the name of the initial branch in new repositories
What would you like Git to name the initial branch after "git init"?

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

This setting does not affect existing repositories.

<https://gitforwindows.org/>

Back Next Cancel

Git 2.46.0 Setup

Adjusting your PATH environment
How would you like to use Git from the command line?

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.

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.

<https://gitforwindows.org/>

Back Next Cancel

Git 2.46.0 Setup

Choosing the SSH executable
Which Secure Shell client program would you like Git to use?

Use bundled OpenSSH

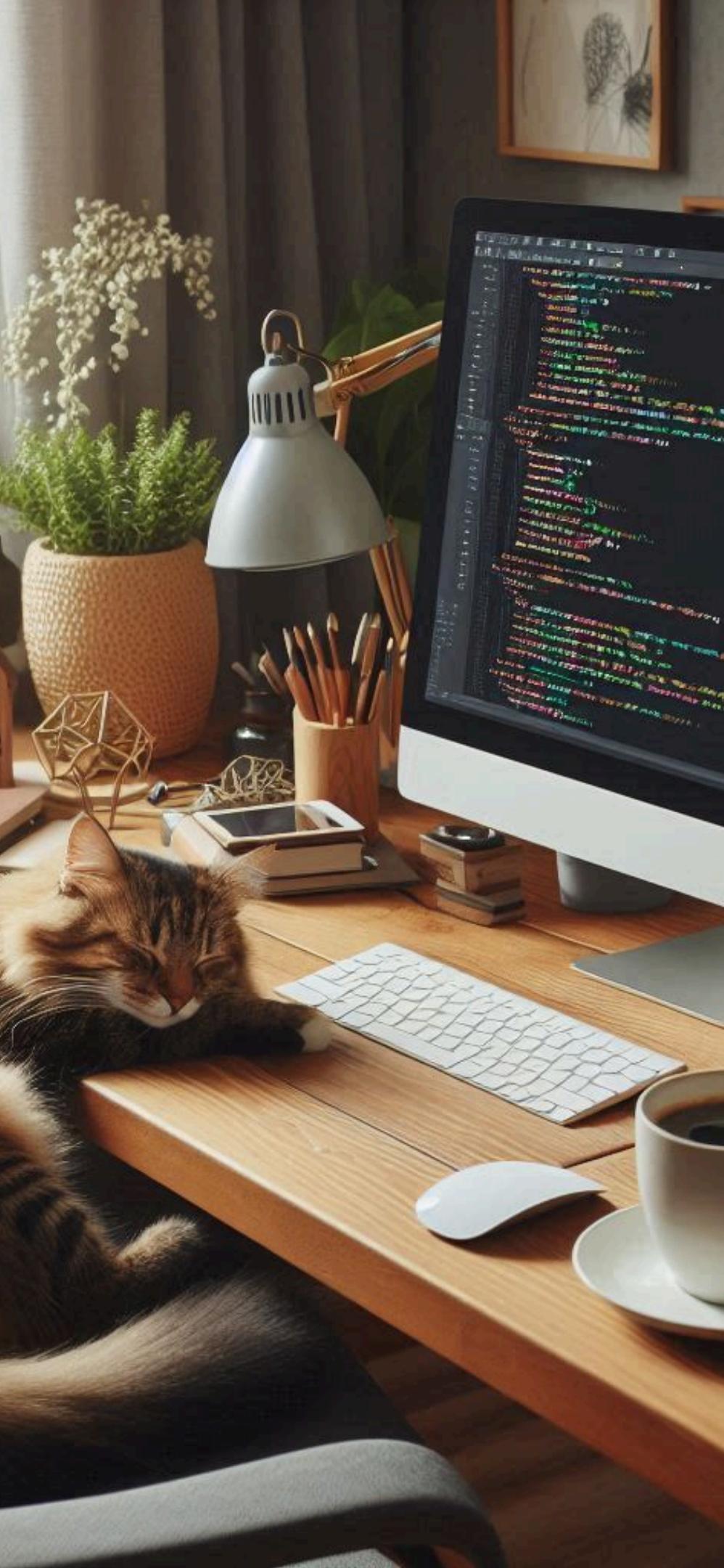
This uses ssh.exe that comes with Git.

Use external OpenSSH

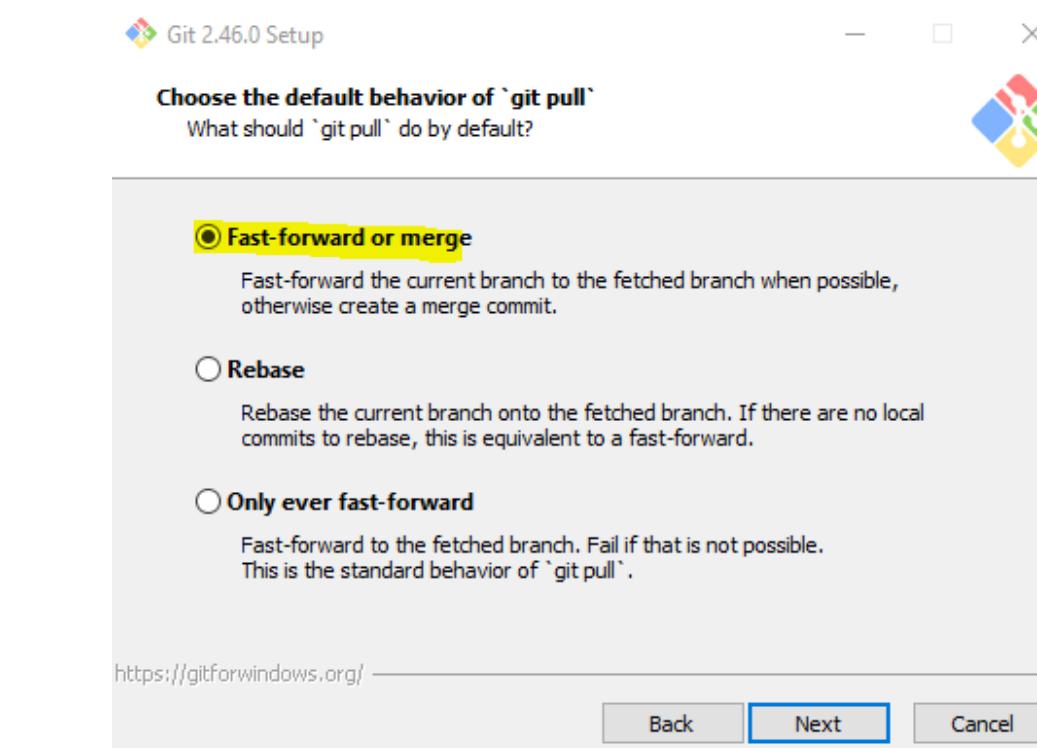
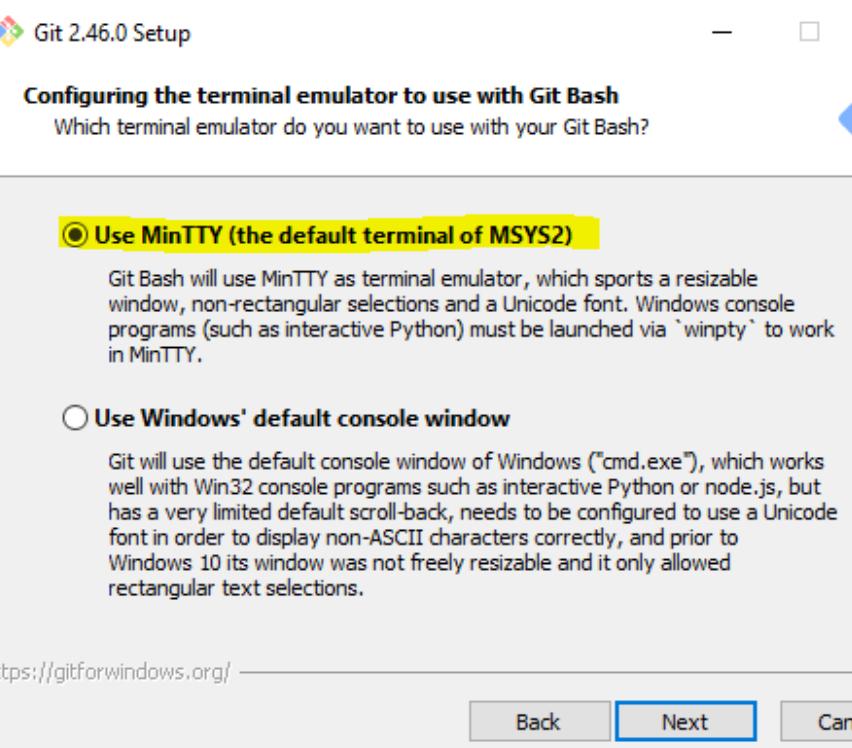
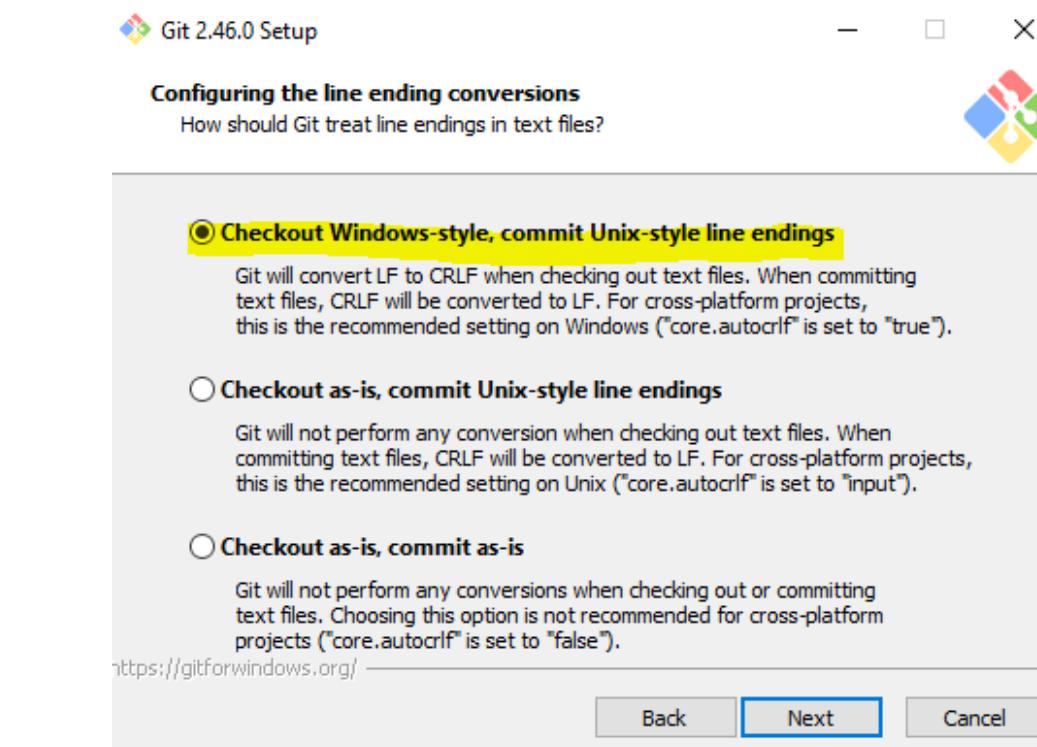
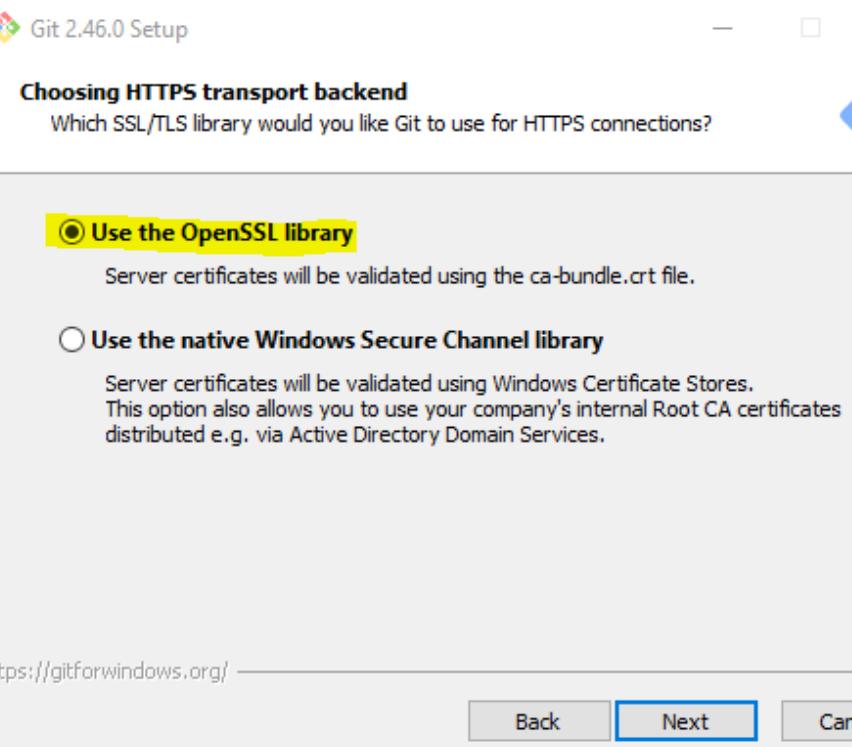
NEW! This uses an external ssh.exe. Git will not install its own OpenSSH (and related) binaries but use them as found on the PATH.

<https://gitforwindows.org/>

Back Next Cancel

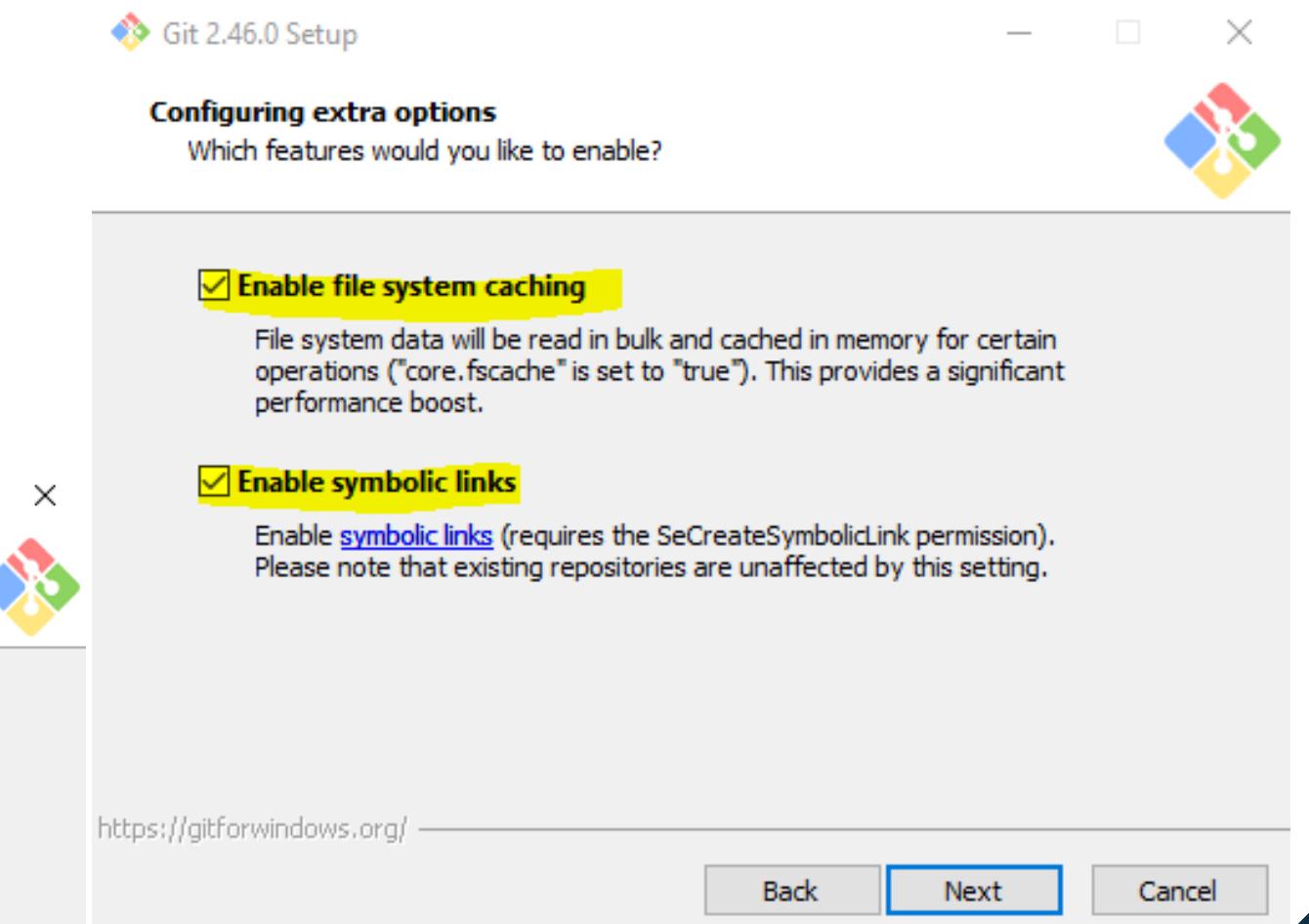
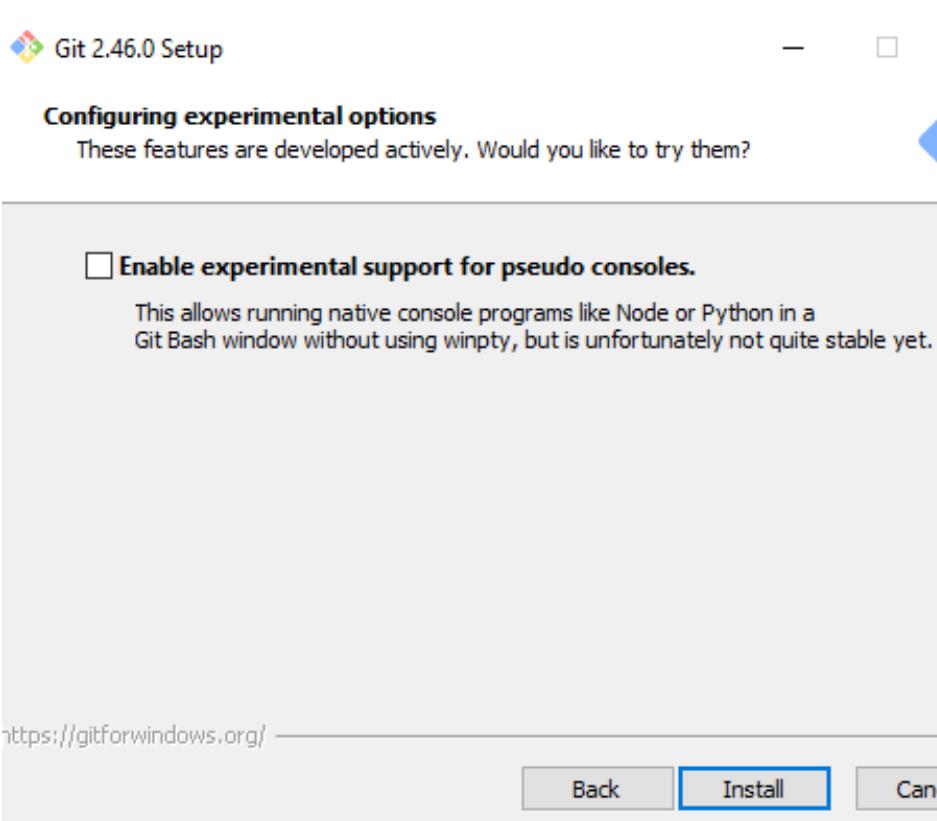
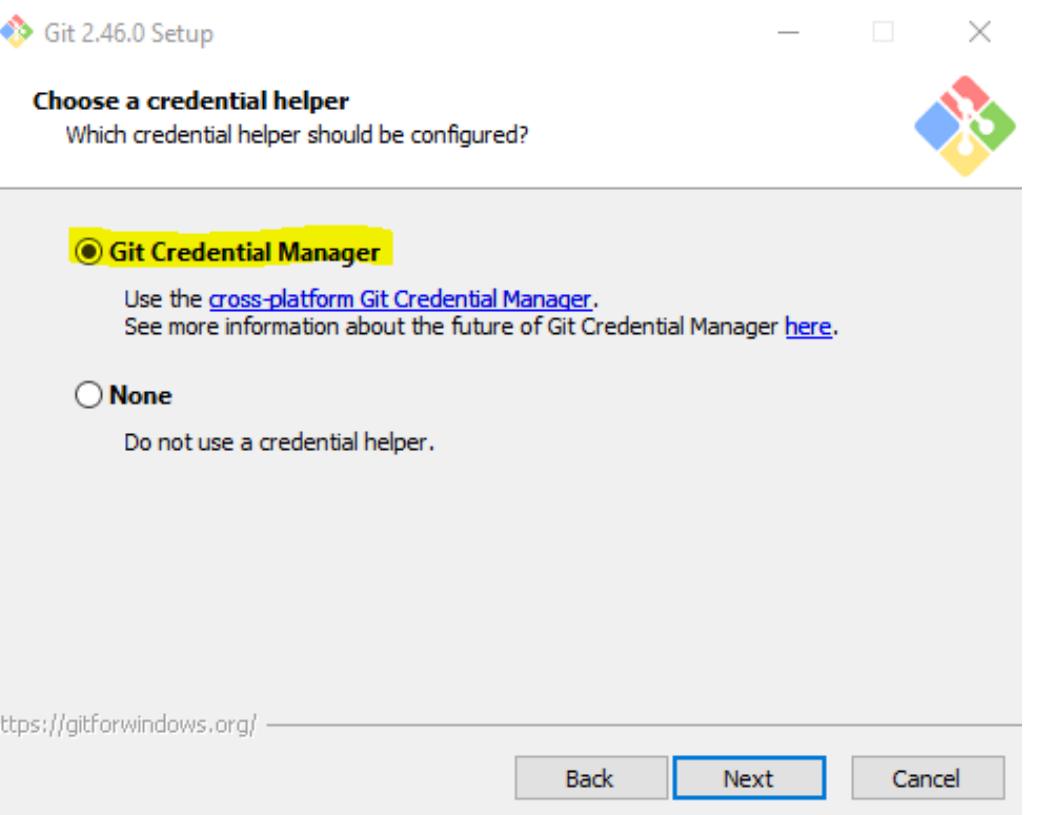


Instalación de Git





Instalación de Git



Entornos Virtuales

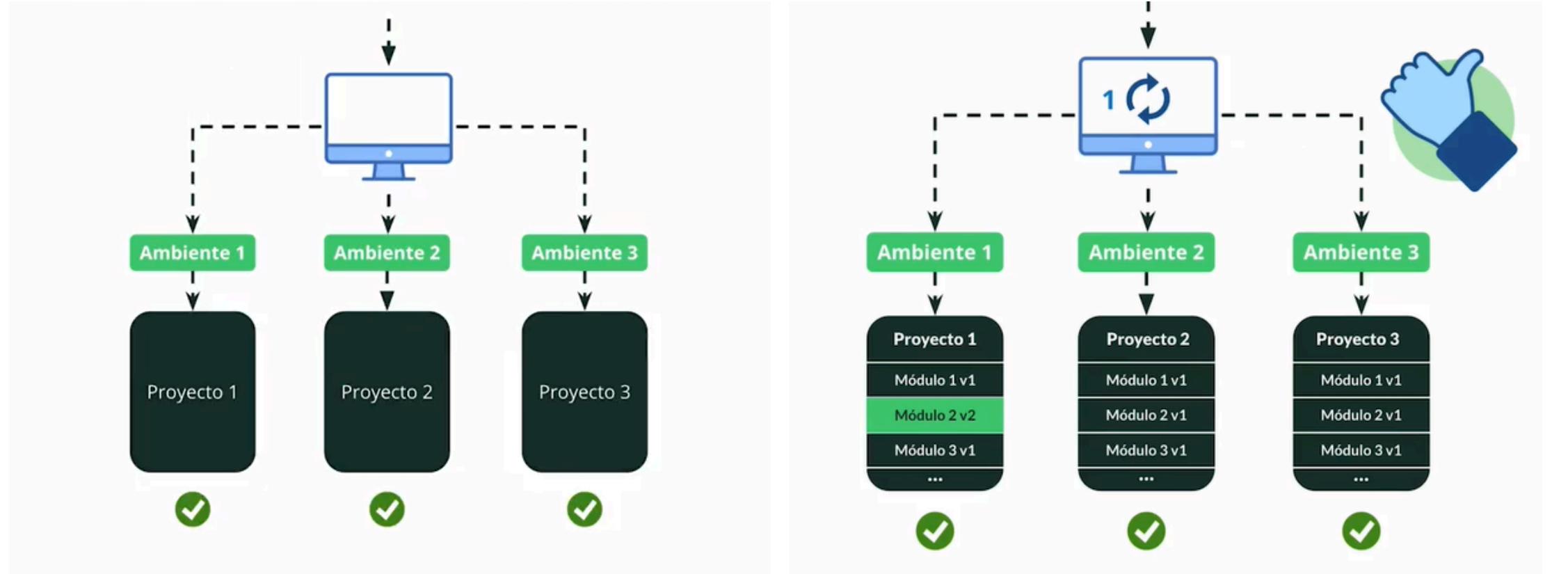
Dentro de todo ambiente de trabajo en programación, debemos configurar los siguientes aspectos en un IDE.

- **Kernel:** Es el núcleo del programa que ejecuta el código. Es el componente encargado de procesar las celdas de código que escribes, calcular los resultados, y devolver las salidas.
- **Ambiente Virtual:** Es un entorno aislado que permite instalar y gestionar paquetes y dependencias de manera independiente para diferentes proyectos. Esto significa que puedes tener diferentes versiones de las mismas librerías instaladas en distintos ambientes, sin que interfieran entre sí.
- **Línea de Comandos:** Es una interfaz que permite a los usuarios interactuar con el sistema operativo mediante comandos de texto.



Entornos Virtuales

Crear y usar entornos virtuales en Python es crucial porque permite aislar las dependencias de cada proyecto. Esto significa que puedes trabajar en múltiples proyectos con diferentes versiones de bibliotecas y herramientas sin que haya conflictos entre ellas. Los entornos virtuales aseguran que cada proyecto tenga exactamente las versiones de paquetes que necesita, lo que facilita la replicabilidad y minimiza problemas al compartir el código con otros.



Entornos Virtuales

- **conda env list:** Lista los ambientes disponibles en conda.
- **conda create --name env_name:** Crea un ambiente virtual.
- **conda list -n env_name:** Lista los paquetes instalados en un ambiente virtual especificado.
- **conda instal pack1 pack2 :** Instala las librerías especificadas dentro del ambiente actual activado.
- **conda instal pack1 = versión :** Instala la librería especificada dentro del ambiente actual activado dada una versión del paquete.
- **conda env create --file environment.yml:** Instala y crea un archivo a partir de un archivo .yml especificado.
- **conda activate env_name:** Activa un ambiente virtual especificado.
- **conda deactivate:** Desactiva un ambiente virtual.
- **conda remove package_name:** Remueve del actual ambiente virtual, un paquete especificado.
- **conda env remove --name env_name:** Elimina un ambiente virtual especificado.





Uptc[®]
Universidad Pedagógica y
Tecnológica de Colombia



CIEC
Facultad de
CIENCIAS

Gracias

