# Sphinx Setup

***Release 0.1***

**Dan Carroll**

**Apr 22, 2023**

## Contents:

This document describes how to get Sphinx-Doc running on a windows machine including the required software for editing, version control and building of the document package files. It also goes into some detail on configuring this software and some basic conventions for RST documents editing.

We are going to describe two methods for building sphinx projects. Firstly, the native way, where we will build the sources with python3 under a virtual environment. Secondly, since building PDFs on Windows this way is difficult, we will also install Rancher-Desktop to build them using docker containers.

A good, quick overview of Sphinx and how to set it up natively (using the Native python instead of a docker package) can be found at the following youtube playlist.

# 1 Introduction

Sphinx is amazingly simple and a versatile documentation helper. For technical documentation it really does shine, giving the user the ability to write concise documentation quickly, and export it in a few useful formats. Coupled with the 'read-the-docs' theme, the resulting html code is very readable.

When it comes to exporting to PDF, this is much more easily achieved when running under a Unix OS, rather than windows. All is not lost however, since Sphinx can also run under docker (which can be very much unix oriented). So, in this document I will describe a few options on how to achive this.

At the end of this procedure you should be able to build great documentation and export it to html, pdf and other formats as well. The 3 ways to do this we will explore are:

- Installing python and git and compiling html docs locally.
- Installing Rancher-Desktop to enable us to build the docs via docker images (html and pdf)
- Using github actions to build the html and pdf docs on github hosted runners, deploying to a github pages site.

At the bare minimum we need to install python and git locally as the integration with VSCode makes it easy to author and quickly compile the sources for review.

**Note:** Rancher-Desktop is opensource software used to administer kubernetes and docker/containerd containers. Unlike Docker-Desktop it can be used commercially for free.

## 1.1 Prerequisites and method

- Python - One of the most popular scripting languages around.
- Sphinx - running under python.
- Visual Studio Code - a great IDE for all sorts of things. (Can also use atom or sublime or notepad - but VS Code is the bomb)
- Git - for saving the sources - used by VSCode.
- A github account.
- Rancher and Docker Engine - For running the Sphinx software in a reliable way.
- Windows Subsystem for Linux ver 2 - Microsoft finally came to the party. Needed for Rancher-Desktop.

Editing is best done in an IDE like VSCode. It provides some formatting context and tips for writing RST documents. It also integrates the building and saving the sources to a local or remote git repository. It is possible to author in another IDE but this set of instructions will focus on VSCode.

To be able to compile the html and pdf files, you need to be able to run the sphinx application. It's a python app so you can install python, create a virtual environment, and build things that way. As mentioned above, I've found that for windows, the PDF part is a bit of a problem. The easiest way to make it work is to run sphinx in a docker container. We will describe both methods, since building the html images is quite quick under native python. And we'll use Rancher-Desktop to build the PDF version under docker.

The docker engine, with docker desktop is popular. However, the docker licensing requires payment for Docker Desktop in commercial environments, so as an alternative, these instructions are for using Rancher with the docker engine.

**Note:** With Rancher and Docker Engine installed, you can also experiment with the millions of available docker images available online. You can run web servers, Java applications, database servers, MQ systems etc... There really is a lot available and there's no concern that you contaminate your own desktop with software that you tested but didn't like. Just delete the container and move on... A good set of documentation for Docker can be found here.

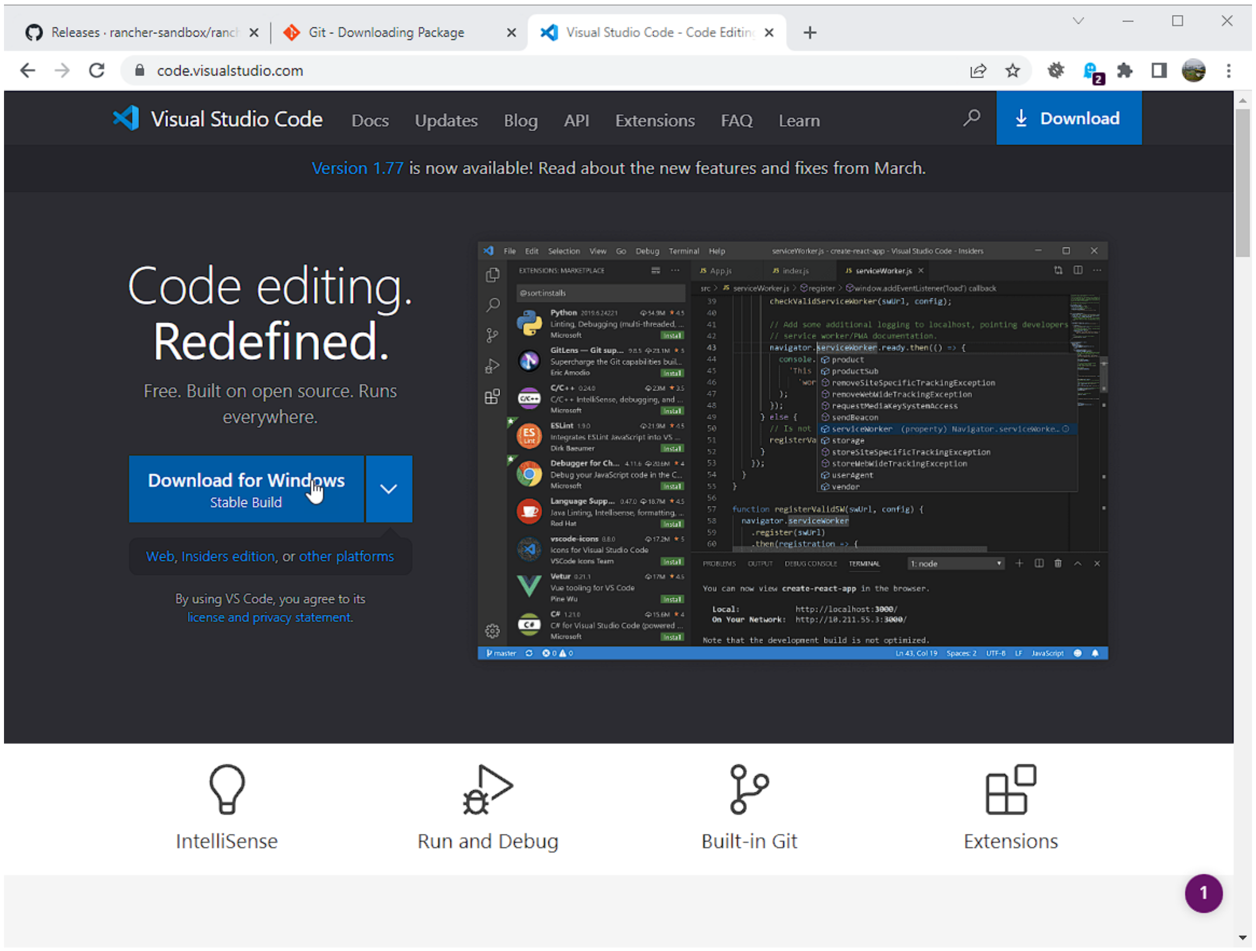
For now, though, let just get sphinx running...

**Warning:** The docker engine runs in a virtual machine under Windows Services for Linux. WSL2 requires admin privileges so these are needed. It will be installed and configured as part of the rancher installation.

# 2 Install Visual Studio Code

Visual Studio Code is a free editor of text files aimed at development. It's enhanced by plugins to extend it's already full base features. Now that you have git and python installed, you can install VSCode. It's not strictly necessary to do it in this order, but it's handy to configure vscode after python and git are installed, since vscode will see this software and suggest useful extensions.

## 2.1 Download Visual Studio Code



Version 1.77 is now available! Read about the new features and fixes from March.

Code editing. Redefined.

Free. Built on open source. Runs everywhere.

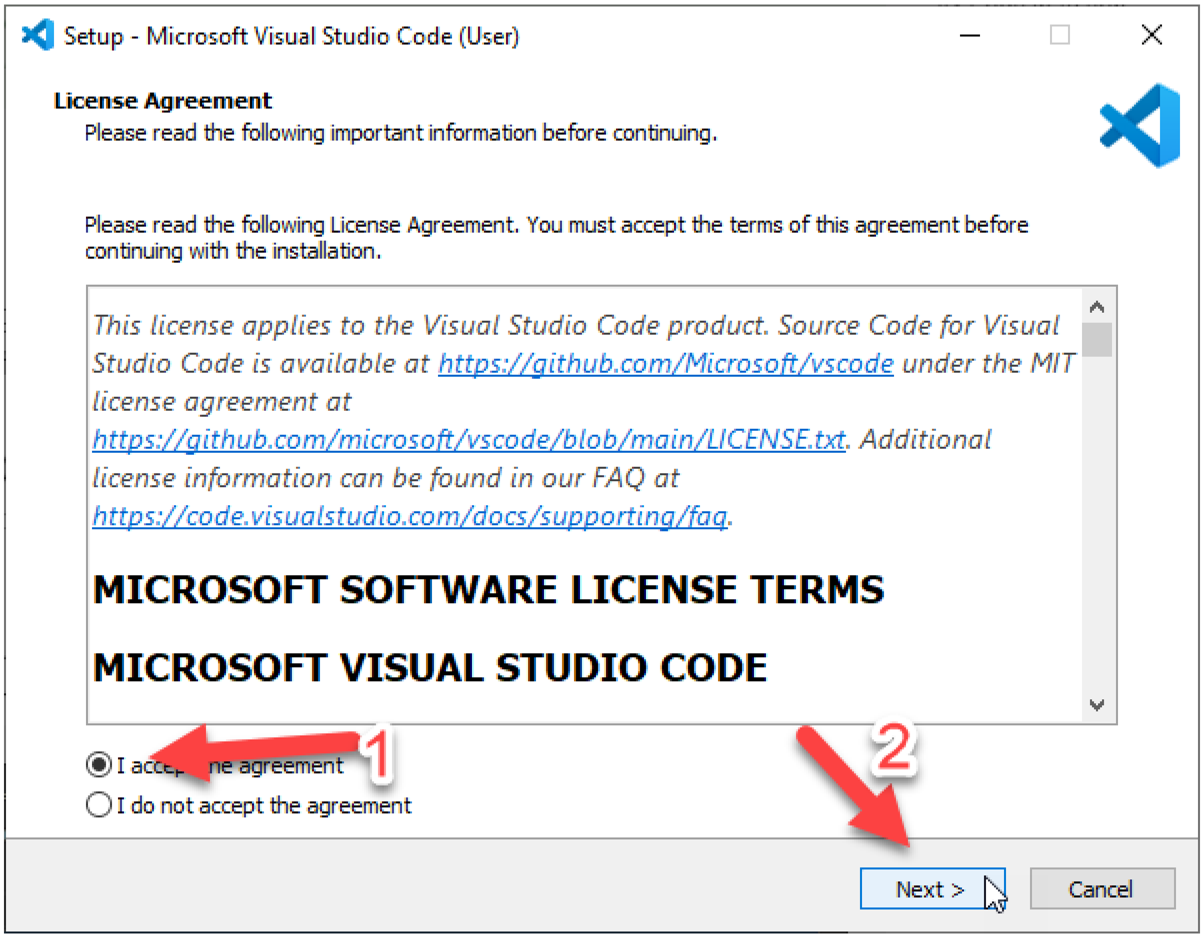
Web, Insiders edition, or other platforms

By using VS Code, you agree to its license and privacy statement.

1. Navigate to https://code.visualstudio.com
2. Click on the ‘Download for Windows’ link to get the latest version of git for windows.

## 2.2 Continue installation

Most options are default.


Setup - Microsoft Visual Studio Code (User)

**License Agreement**

Please read the following important information before continuing.

Please read the following License Agreement. You must accept the terms of this agreement before continuing with the installation.

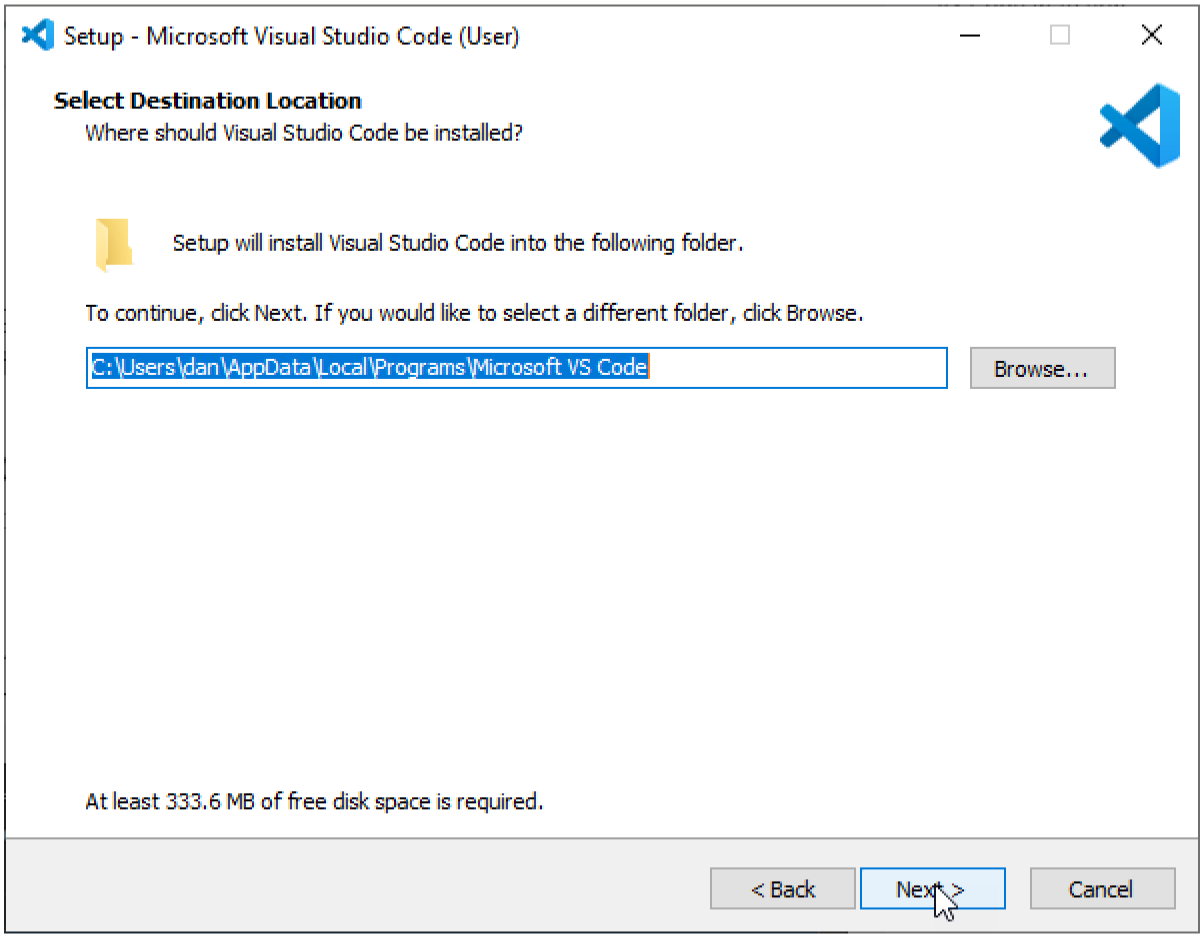
*This license applies to the Visual Studio Code product. Source Code for Visual Studio Code is available at https://github.com/Microsoft/vscode under the MIT license agreement at https://github.com/microsoft/vscode/blob/main/LICENSE.txt. Additional license information can be found in our FAQ at https://code.visualstudio.com/docs/supporting/faq.*

**MICROSOFT SOFTWARE LICENSE TERMS**

**MICROSOFT VISUAL STUDIO CODE**

[✓ I accept the agreement]
I do not accept the agreement

Next


Setup - Microsoft Visual Studio Code (User)

**Select Destination Location**

Where should Visual Studio Code be installed?

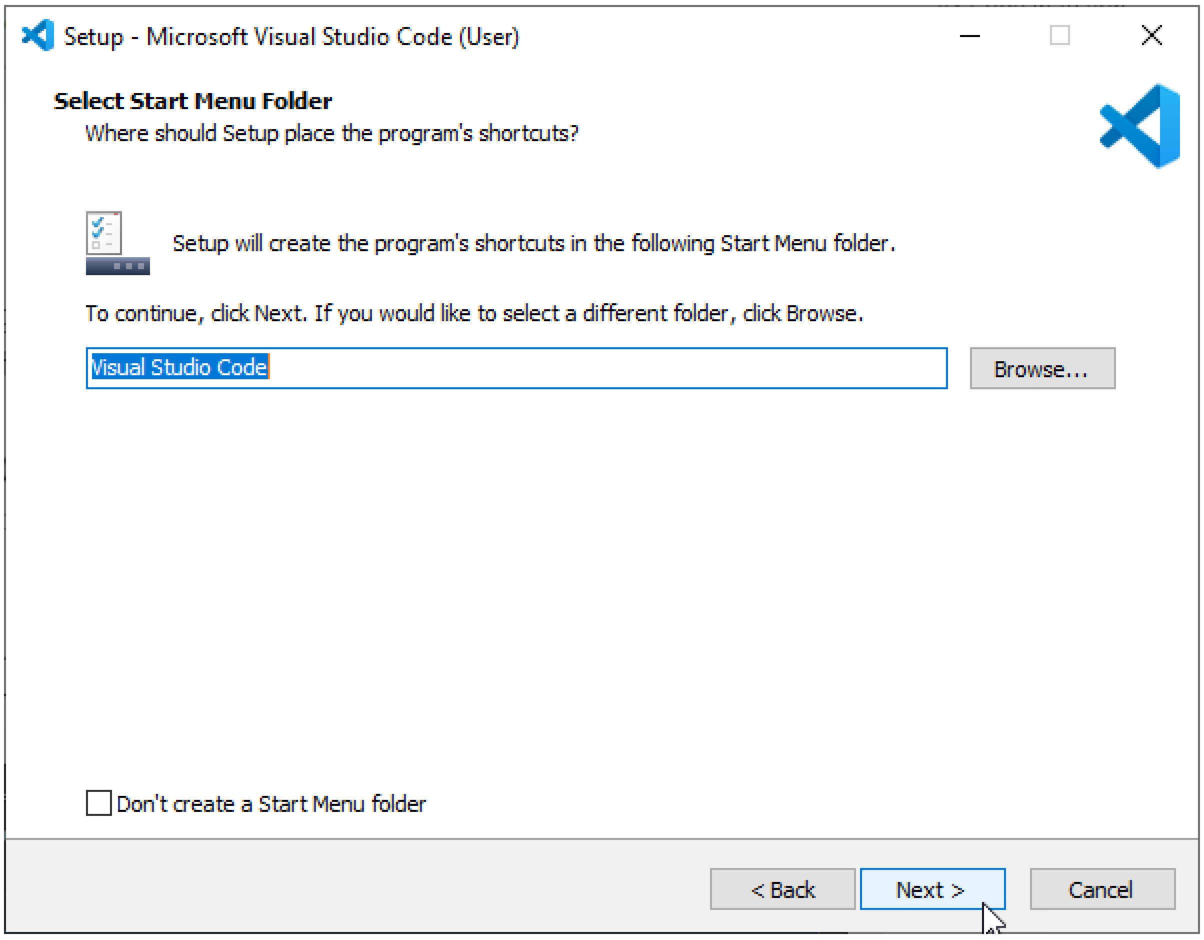
Setup will install Visual Studio Code into the following folder.

To continue, click Next. If you would like to select a different folder, click Browse.

C:\Users\dan\AppData\Local\Programs\Microsoft VS Code

At least 333.6 MB of free disk space is required.

Next


Setup - Microsoft Visual Studio Code (User)

**Select Start Menu Folder**

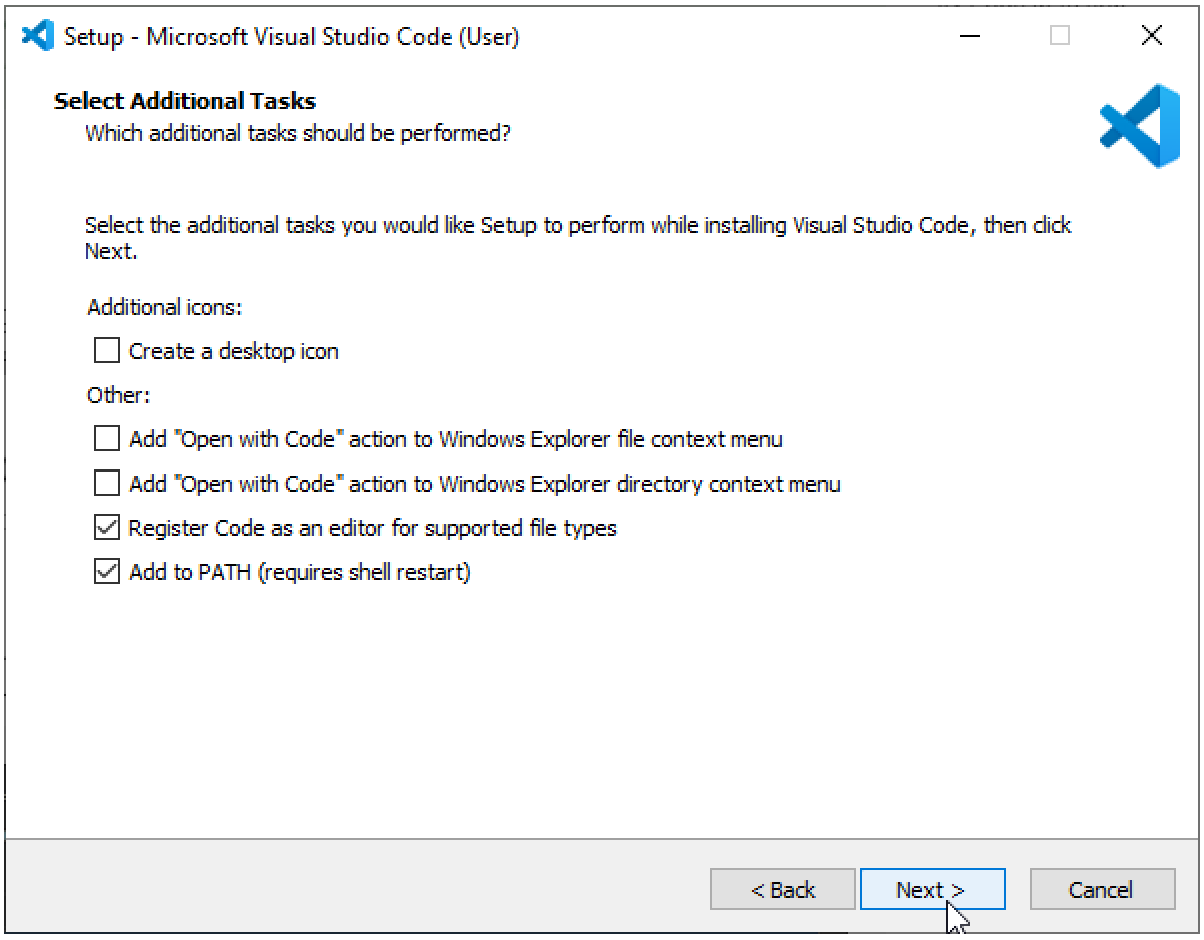
Where should Setup place the program's shortcuts?

Setup will create the program's shortcuts in the following Start Menu folder.

To continue, click Next. If you would like to select a different folder, click Browse.

Visual Studio Code

Next


Setup - Microsoft Visual Studio Code (User)

**Select Additional Tasks**

Which additional tasks should be performed?

Select the additional tasks you would like Setup to perform while installing Visual Studio Code, then click Next.

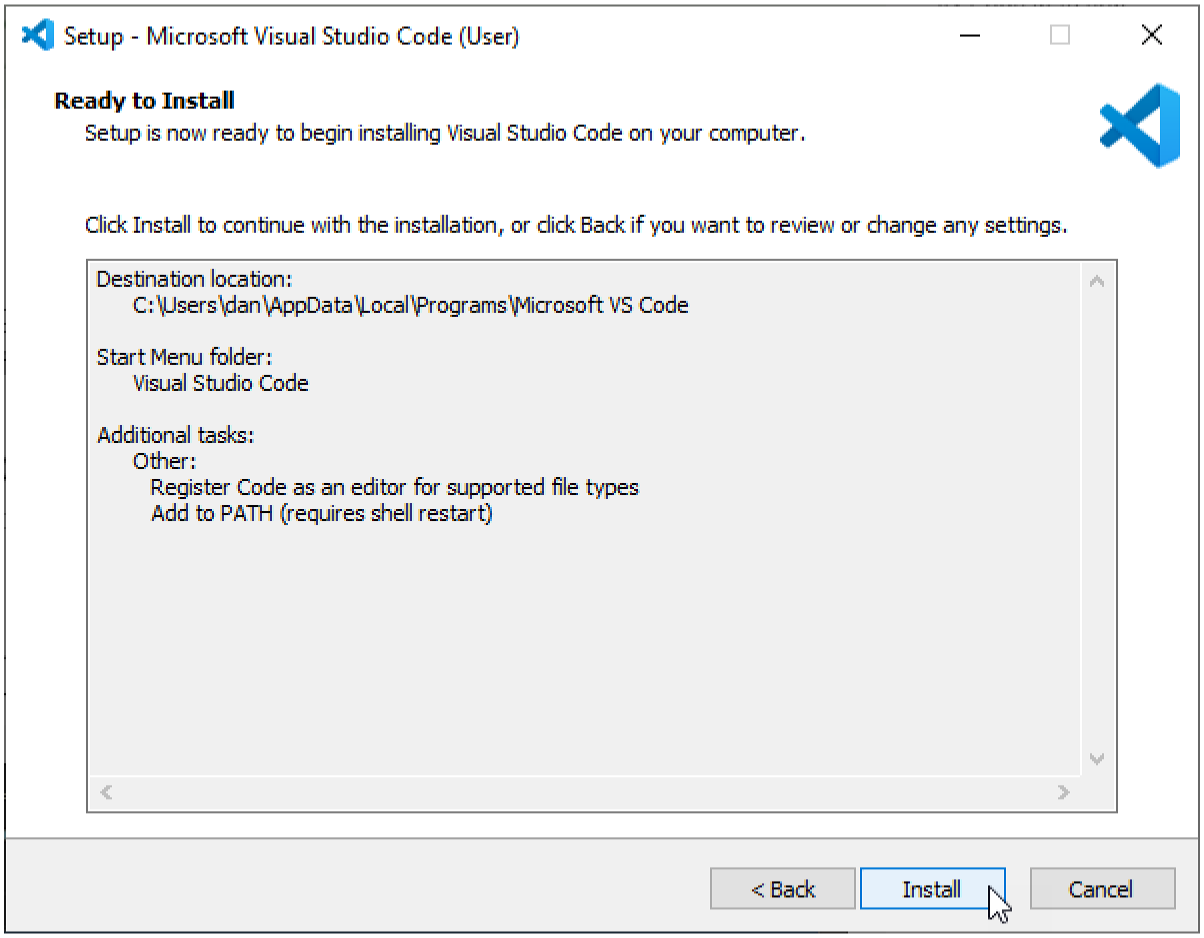
Additional icons:

- [ ] Create a desktop icon

Other:

- [ ] Add "Open with Code" action to Windows Explorer file context menu
- [ ] Add "Open with Code" action to Windows Explorer directory context menu
- [✓] Register Code as an editor for supported file types
- [✓] Add to PATH (requires shell restart)

Next



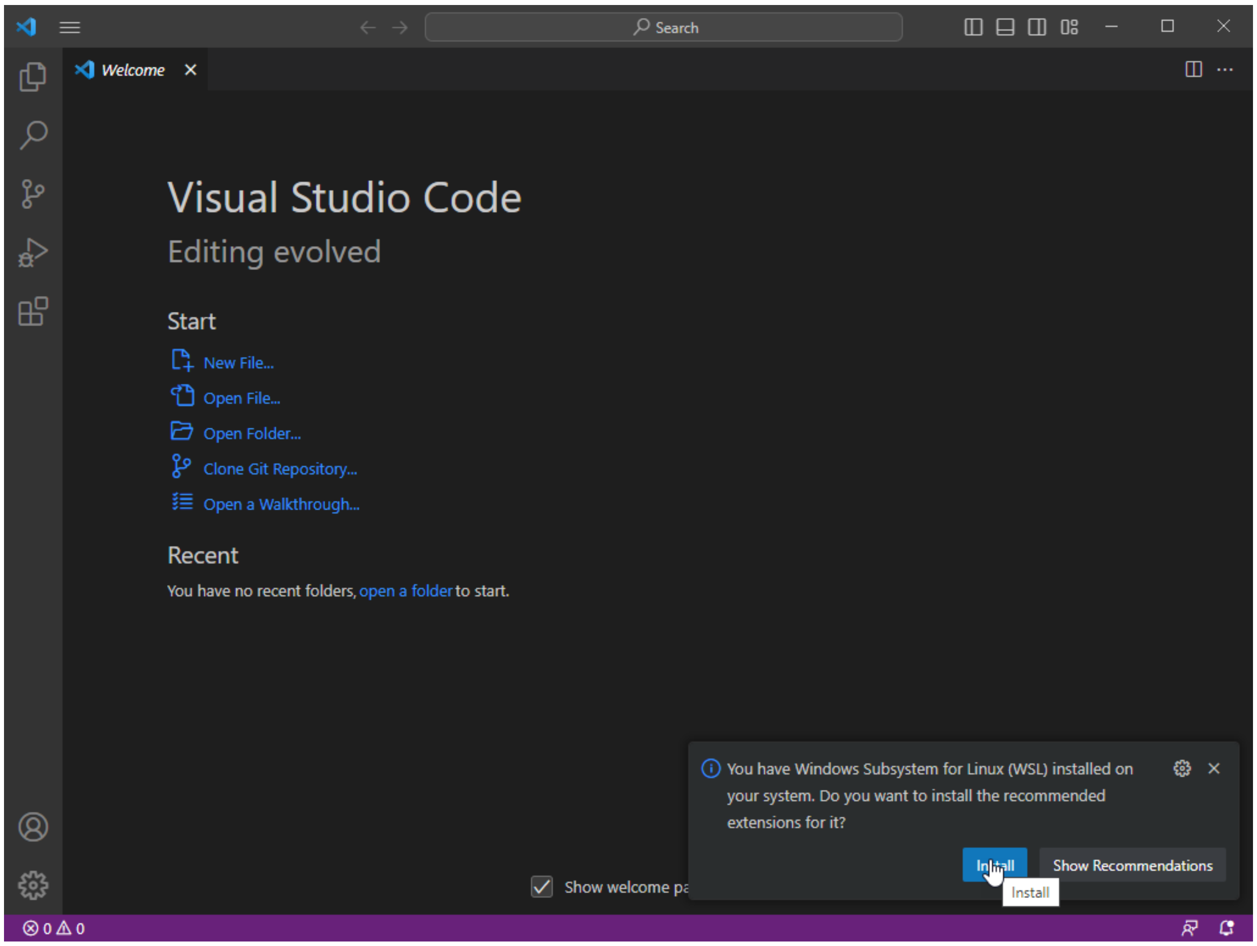
**Ready to Install**

Setup is now ready to begin installing Visual Studio Code on your computer.

Click Install to continue with the installation, or click Back if you want to review or change any settings.

```
Destination location:
    C:\Users\dan\AppData\Local\Programs\Microsoft VS Code

Start Menu folder:
    Visual Studio Code

Additional tasks:
    Other:
      Register Code as an editor for supported file types
      Add to PATH (requires shell restart)
```

Next



You have no recent folders, open a folder to start.

You have Windows Subsystem for Linux (WSL) installed on your system. Do you want to install the recommended extensions for it?

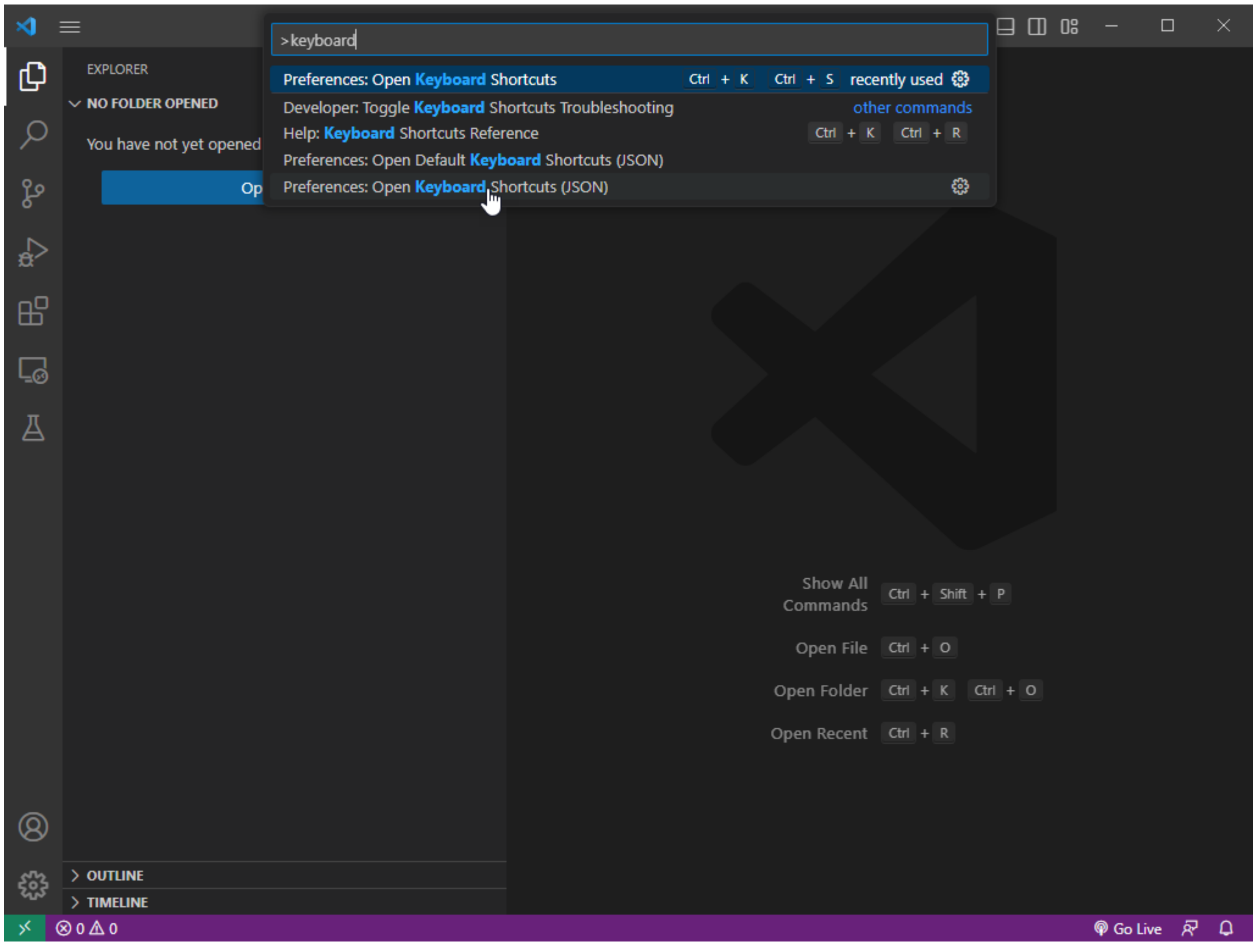
When the installation is complete, you will be asked to install the WSL extension. While not strictly necessary, you can install it anyway.

## 2.3 Final Confguration of Visual Studio Code

VS Code is installed now. But there's one or two things that you can do to make things easier on you. These are largely optional but I find them useful.

### 2.3.1 Setup some key bindings

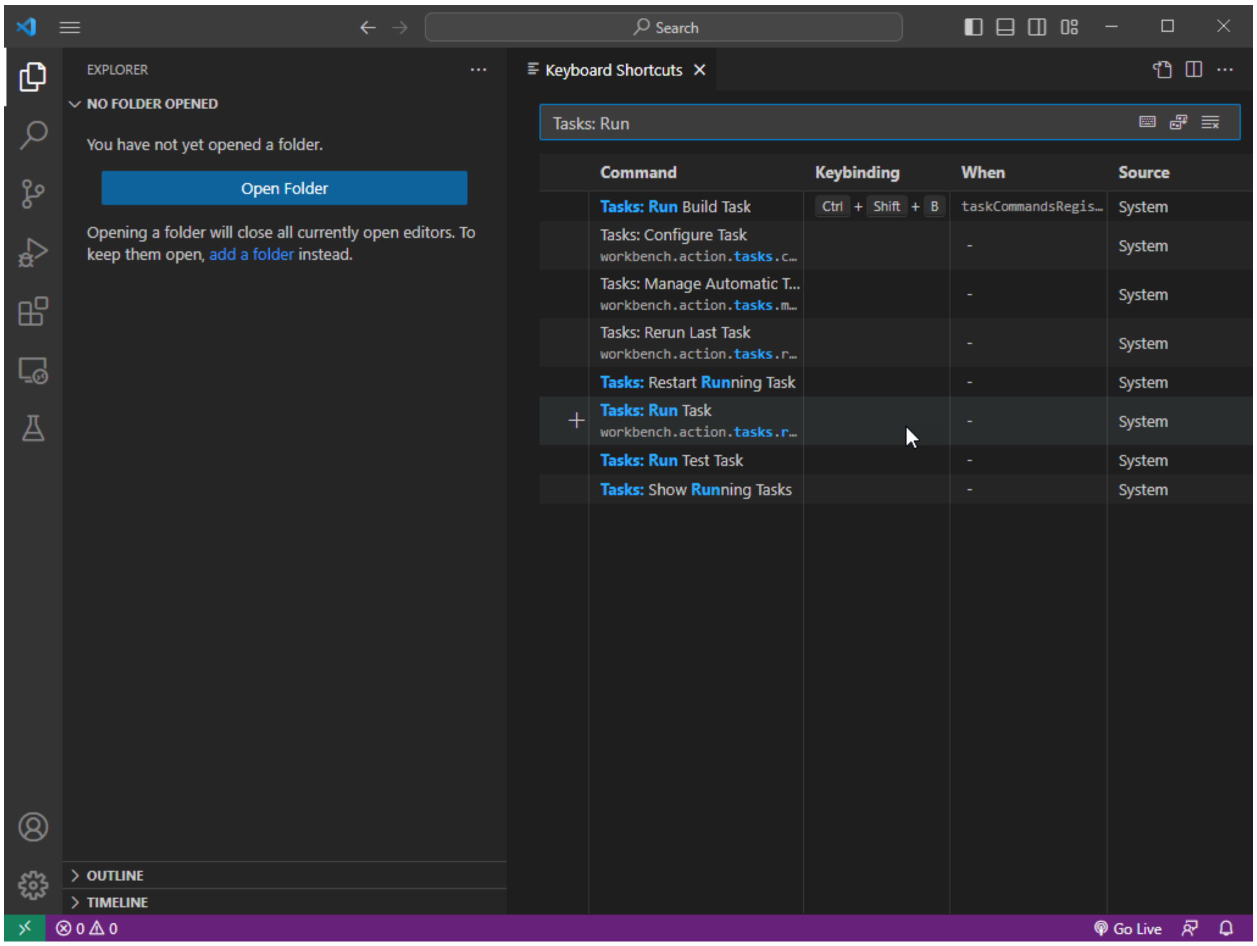
Open VS Code and hit Ctrl-Shift-P to open the “Show Commands” search bar. You can also access this by typing “>” in the standard search bar. Type “Keyboard” and you will see some options.



>keyboard

Preferences: Open Keyboard Shortcuts Ctrl + K Ctrl + S recently used
Developer: Toggle Keyboard Shortcuts Troubleshooting other commands
Help: Keyboard Shortcuts Reference Ctrl + K Ctrl + R
Preferences: Open Default Keyboard Shortcuts (JSON)
Preferences: Open Keyboard Shortcuts (JSON)

We’re looking for the “Preferences: Open Keyboard Shortcuts” one. Click on this (**Not the JSON version**, that is if you wish to edit the config file directly).


EXPLORER

NO FOLDER OPENED

You have not yet opened a folder.

Open Folder

Opening a folder will close all currently open editors. To keep them open, add a folder instead.

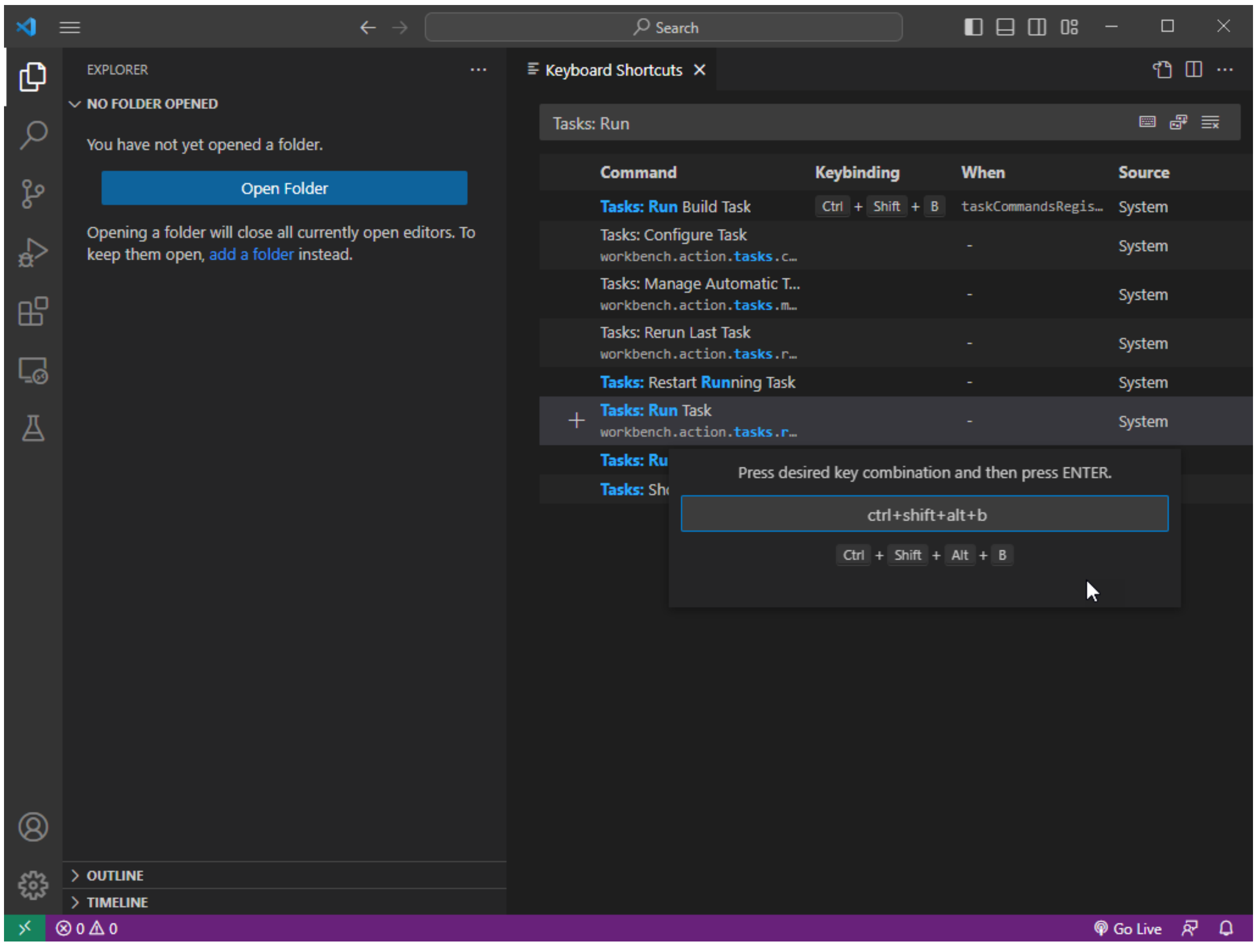
Keyboard Shortcuts

Tasks: Run

| Command | Keybinding | When | Source |
|---|---|---|---|
| Tasks: Run Build Task | Ctrl + Shift + B | taskCommandsRegis… | System |
| Tasks: Configure Task<br>workbench.action.tasks.c… | | - | System |
| Tasks: Manage Automatic T…<br>workbench.action.tasks.m… | | - | System |
| Tasks: Rerun Last Task<br>workbench.action.tasks.r… | | - | System |
| Tasks: Restart Running Task | | - | System |
| Tasks: Run Task<br>workbench.action.tasks.r… | | - | System |
| Tasks: Run Test Task | | - | System |
| Tasks: Show Running Tasks | | - | System |

If you type in “Tasks: Run” then you will see a list of commands you can bind a shortcut key to. Notice that “Ctrl + Shift + B” is already bound to the “Tasks: Run Build Task”? We will use that to build our html documentation.

For now, lets bind a keyboard shortcut to “Tasks: Run” which will allow us to run any defined task. Move your mouse over the “Tasks: Run” line and either click the + on the left or double click the line.


Keyboard Shortcuts

Tasks: Run

| Command | Keybinding | When | Source |
|---|---|---|---|
| Tasks: Run Build Task | Ctrl + Shift + B | taskCommandsRegis… | System |
| Tasks: Configure Task<br>workbench.action.tasks.c… |  | - | System |
| Tasks: Manage Automatic T…<br>workbench.action.tasks.m… |  | - | System |
| Tasks: Rerun Last Task<br>workbench.action.tasks.r… |  | - | System |
| Tasks: Restart Running Task |  | - | System |
| Tasks: Run Task<br>workbench.action.tasks.r… |  | - | System |
| Tasks: Ru |  |  |  |
| Tasks: Sh |  |  |  |

Press desired key combination and then press ENTER.

ctrl+shift+alt+b

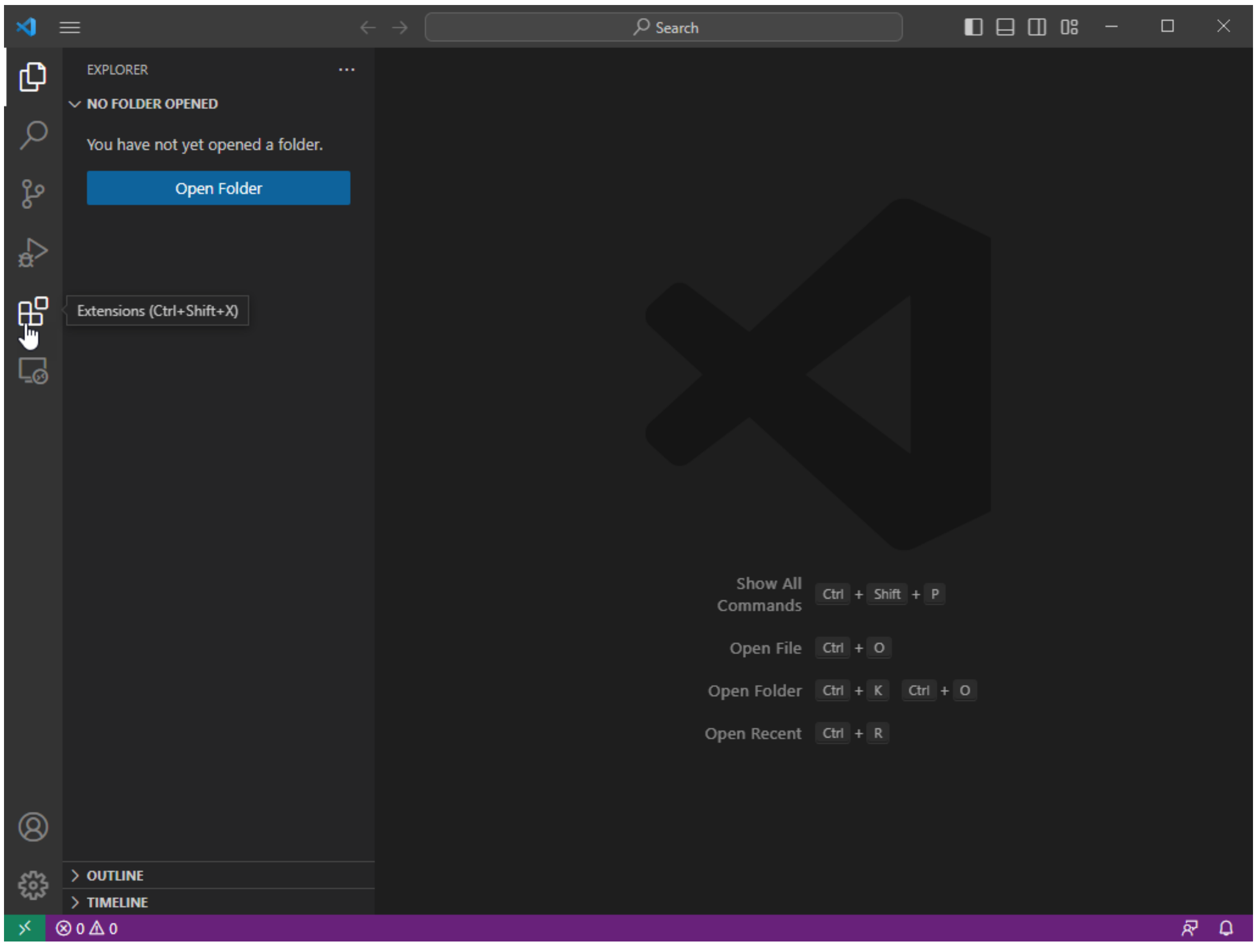
Ctrl + Shift + Alt + B

EXPLORER
NO FOLDER OPENED
You have not yet opened a folder.
Open Folder
Opening a folder will close all currently open editors. To keep them open, add a folder instead.
OUTLINE
TIMELINE
Go Live

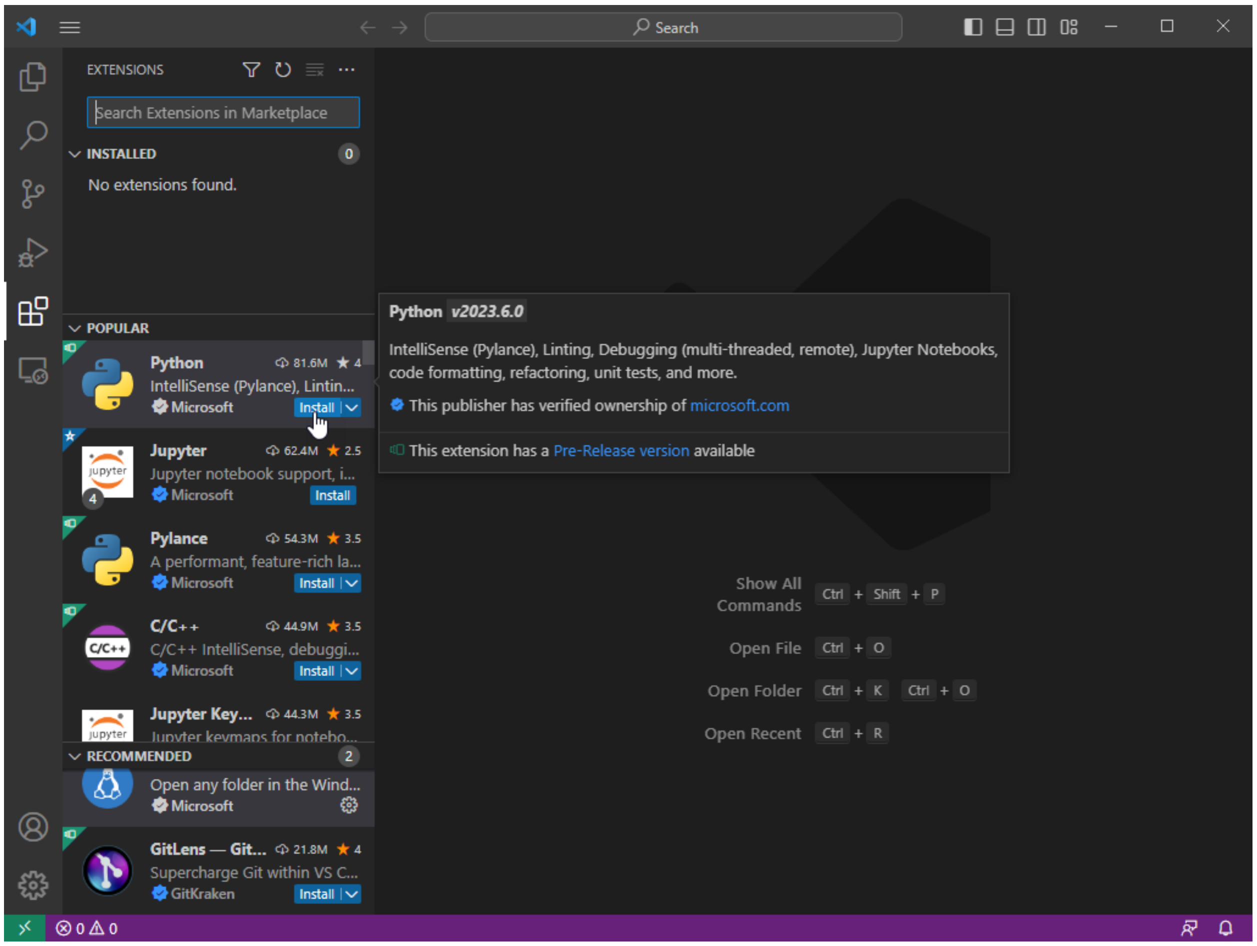
Hold down the keys "Ctrl, Shift, Alt and B". If this keyboard shortcut were already used, it would give us a warning. It is not so just hit Enter to confirm.

### 2.3.2 Install more extensions

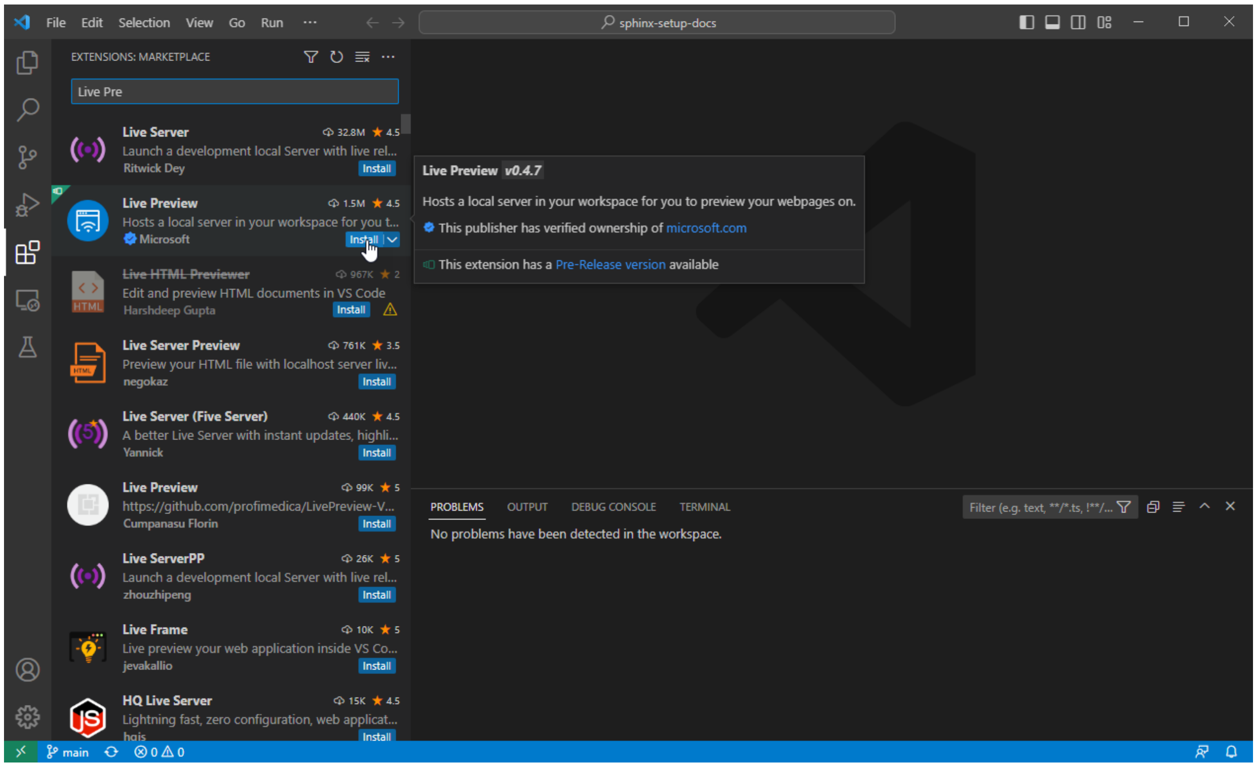
We already installed the WSL extension again, but we can add a few more interesting ones.


Search
EXPLORER
NO FOLDER OPENED
You have not yet opened a folder.
Open Folder
Extensions (Ctrl+Shift+X)
Show All Commands Ctrl + Shift + P
Open File Ctrl + O
Open Folder Ctrl + K Ctrl + O
Open Recent Ctrl + R
OUTLINE
TIMELINE
0 0

Click on the extensions button in the sidebar


EXTENSIONS
Search Extensions in Marketplace
INSTALLED 0
No extensions found.
POPULAR
Python 81.6M ★ 4
IntelliSense (Pylance), Lintin...
Microsoft Install
Jupyter 62.4M ★ 2.5
Jupyter notebook support, i...
Microsoft Install
Pylance 54.3M ★ 3.5
A performant, feature-rich la...
Microsoft Install
C/C++ 44.9M ★ 3.5
C/C++ IntelliSense, debuggi...
Microsoft Install
Jupyter Key... 44.3M ★ 3.5
RECOMMENDED 2
Open any folder in the Wind...
Microsoft
GitLens — Git... 21.8M ★ 4
Supercharge Git within VS C...
GitKraken Install
Python v2023.6.0
IntelliSense (Pylance), Linting, Debugging (multi-threaded, remote), Jupyter Notebooks, code formatting, refactoring, unit tests, and more.
This publisher has verified ownership of microsoft.com
This extension has a Pre-Release version available
Show All Commands Ctrl + Shift + P
Open File Ctrl + O
Open Folder Ctrl + K Ctrl + O
Open Recent Ctrl + R

Search for ‘Python’ and install it.



Live Pre

Live Server 32.8M ★ 4.5
Launch a development local Server with live rel...
Ritwick Dey Install

Live Preview 1.5M ★ 4.5
Hosts a local server in your workspace for you t...
Microsoft Install

~~Live HTML Previewer~~ 967K ★ 2
Edit and preview HTML documents in VS Code
Harshdeep Gupta Install

Live Server Preview 761K ★ 3.5
Preview your HTML file with localhost server liv...
negokaz Install

Live Server (Five Server) 440K ★ 4.5
A better Live Server with instant updates, highli...
Yannick Install

Live Preview 99K ★ 5
https://github.com/profimedica/LivePreview-V...
Cumpanasu Florin Install

Live ServerPP 26K ★ 5
Launch a development local Server with live rel...
zhouzhipeng Install

Live Frame 10K ★ 5
Live preview your web application inside VS Co...
jevakallio Install

HQ Live Server 15K ★ 4.5
Lightning fast, zero configuration, web applicat...
hqjs Install

Live Preview *v0.4.7*

Hosts a local server in your workspace for you to preview your webpages on.

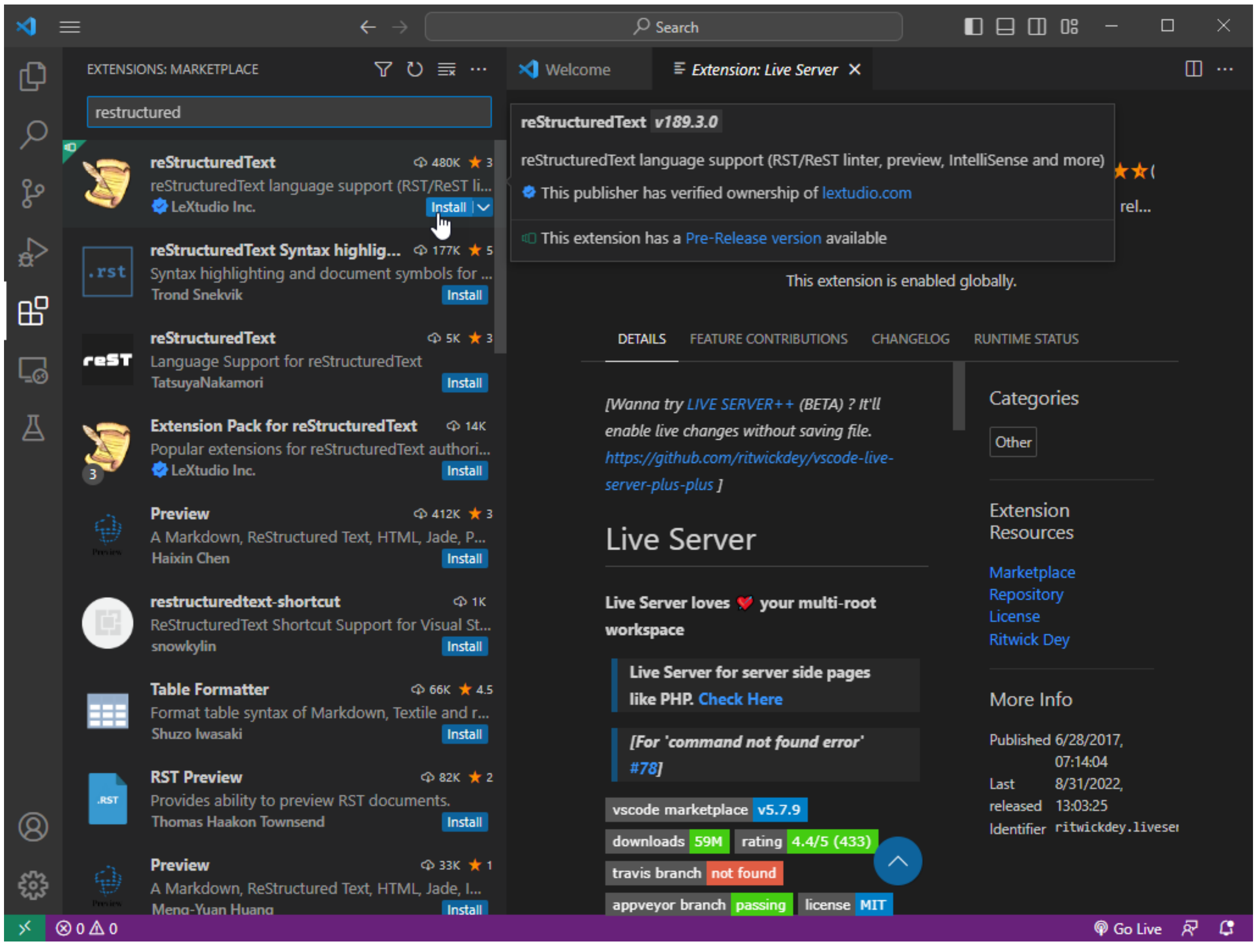
This publisher has verified ownership of microsoft.com

This extension has a Pre-Release version available

Filter (e.g. text, **/*.ts, !**/...

No problems have been detected in the workspace.

Search for ‘Live Preview’ and install it.



reStructuredText v189.3.0

reStructuredText language support (RST/ReST linter, preview, IntelliSense and more)

This publisher has verified ownership of lextudio.com

This extension has a Pre-Release version available

This extension is enabled globally.

| Extension | Description | Publisher | Installs | Rating |
|---|---|---|---|---|
| reStructuredText | reStructuredText language support (RST/ReST li... | LeXtudio Inc. | 480K | 3 |
| reStructuredText Syntax highlig... | Syntax highlighting and document symbols for ... | Trond Snekvik | 177K | 5 |
| reStructuredText | Language Support for reStructuredText | TatsuyaNakamori | 5K | 3 |
| Extension Pack for reStructuredText | Popular extensions for reStructuredText authori... | LeXtudio Inc. | 14K | |
| Preview | A Markdown, ReStructured Text, HTML, Jade, P... | Haixin Chen | 412K | 3 |
| restructuredtext-shortcut | ReStructuredText Shortcut Support for Visual St... | snowkylin | 1K | |
| Table Formatter | Format table syntax of Markdown, Textile and r... | Shuzo Iwasaki | 66K | 4.5 |
| RST Preview | Provides ability to preview RST documents. | Thomas Haakon Townsend | 82K | 2 |
| Preview | A Markdown, ReStructured Text, HTML, Jade, I... | Meng-Yuan Huang | 33K | 1 |

*[Wanna try LIVE SERVER++ (BETA) ? It'll enable live changes without saving file. https://github.com/ritwickdey/vscode-live-server-plus-plus ]*

Live Server

**Live Server loves ❤ your multi-root workspace**

**Live Server for server side pages like PHP. Check Here**

***[For 'command not found error' #78]***

vscode marketplace v5.7.9
downloads 59M rating 4.4/5 (433)
travis branch not found
appveyor branch passing license MIT

Categories

Other

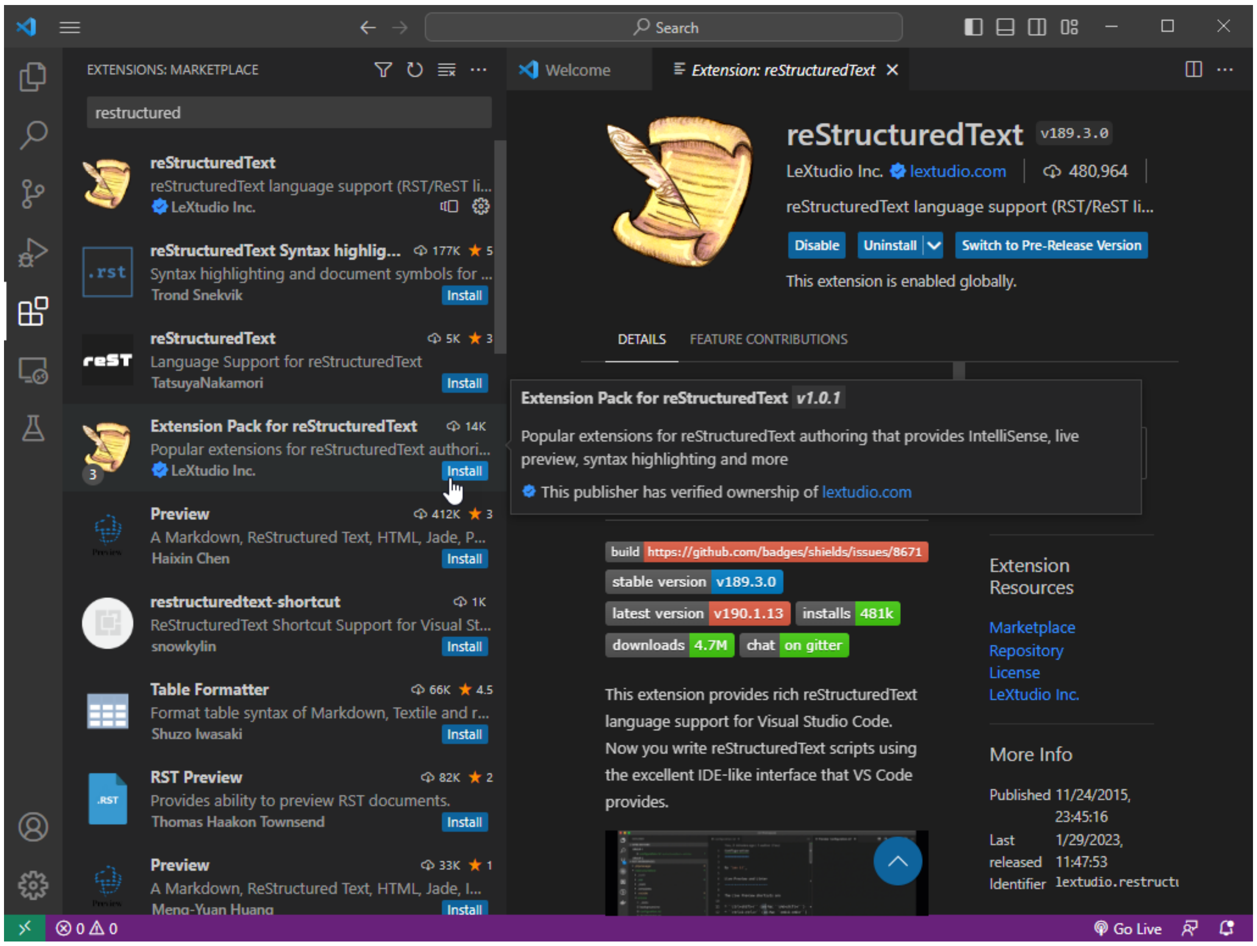
Extension Resources

Marketplace
Repository
License
Ritwick Dey

More Info

| | |
|---|---|
| Published | 6/28/2017, 07:14:04 |
| Last released | 8/31/2022, 13:03:25 |
| Identifier | ritwickdey.liveser |

Search for ‘Restructured’ and install both RestructuredTex…



reStructuredText
reStructuredText language support (RST/ReST li...
LeXtudio Inc.

reStructuredText Syntax highlig... 177K ★ 5
Syntax highlighting and document symbols for ...
Trond Snekvik
Install

reStructuredText 5K ★ 3
Language Support for reStructuredText
TatsuyaNakamori
Install

Extension Pack for reStructuredText 14K
Popular extensions for reStructuredText authori...
LeXtudio Inc.
Install

Preview 412K ★ 3
A Markdown, ReStructured Text, HTML, Jade, P...
Haixin Chen
Install

restructuredtext-shortcut 1K
ReStructuredText Shortcut Support for Visual St...
snowkylin
Install

Table Formatter 66K ★ 4.5
Format table syntax of Markdown, Textile and r...
Shuzo Iwasaki
Install

RST Preview 82K ★ 2
Provides ability to preview RST documents.
Thomas Haakon Townsend
Install

Preview 33K ★ 1
A Markdown, ReStructured Text, HTML, Jade, I...
Meng-Yuan Huang
Install

reStructuredText v189.3.0
LeXtudio Inc. lextudio.com | 480,964
reStructuredText language support (RST/ReST li...
Disable | Uninstall | Switch to Pre-Release Version
This extension is enabled globally.

DETAILS FEATURE CONTRIBUTIONS

Extension Pack for reStructuredText v1.0.1
Popular extensions for reStructuredText authoring that provides IntelliSense, live preview, syntax highlighting and more
This publisher has verified ownership of lextudio.com

build https://github.com/badges/shields/issues/8671
stable version v189.3.0
latest version v190.1.13 installs 481k
downloads 4.7M chat on gitter

This extension provides rich reStructuredText language support for Visual Studio Code. Now you write reStructuredText scripts using the excellent IDE-like interface that VS Code provides.

Extension Resources
Marketplace
Repository
License
LeXtudio Inc.

More Info
Published 11/24/2015, 23:45:16
Last released 1/29/2023, 11:47:53
Identifier lextudio.restructu

⊗ 0 ⚠ 0 Go Live

and the associated extensions pack (both by LeXtudio)

Visual studio code is now installed. Next step, we will download the code to this documentation package and build it.

## 3 Method 1 - Build locally

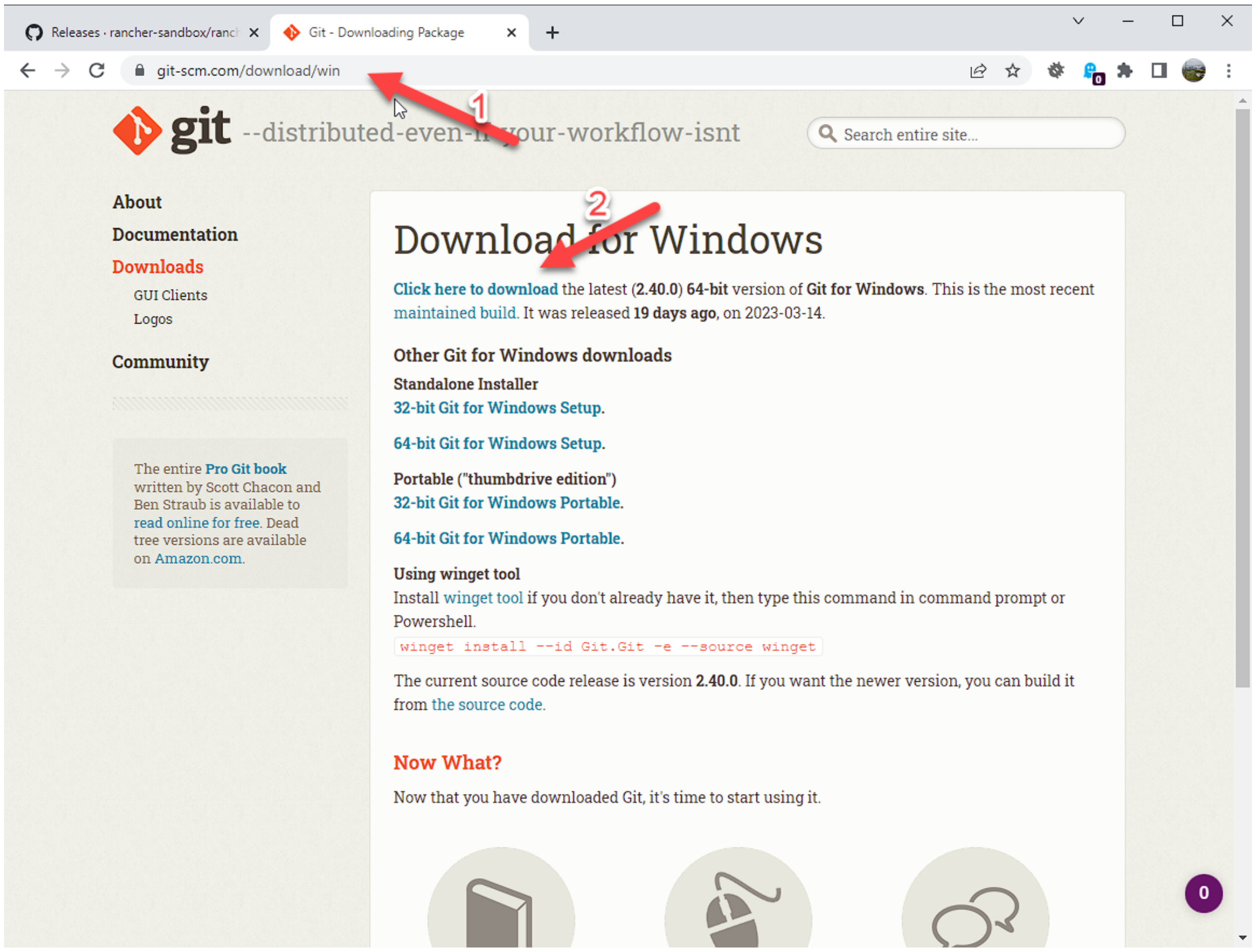
### 3.1 Install Git and Python

Git is the software used to maintain version control for lots of projects. Usually it is for source control for development products, it works equally well for maintaining configuration changes. While it can be used for binary as well as text data, Git's power is shown through best when working with text based documents. Commit changes and version differences can be seen very clearly in these cases. We need it for VS Code to be able to save changes to the git repos.

We'll also install python. We don't strictly need it since we are using python under docker to build the html and pdf package but it's nice to have python installed because VS Code and some of the extensions require python to run.

This part of the document describes the installation process for these two pieces of software on windows.

## 3.2 Download Git


Releases · rancher-sandbox/ranc
Git - Downloading Package
git-scm.com/download/win
1
2
git --distributed-even-if-your-workflow-isnt
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 (**2.40.0**) **64-bit** version of **Git for Windows**. This is the most recent maintained build. It was released **19 days ago**, on 2023-03-14.

**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 **2.40.0**. If you want the newer version, you can build it from the source code.

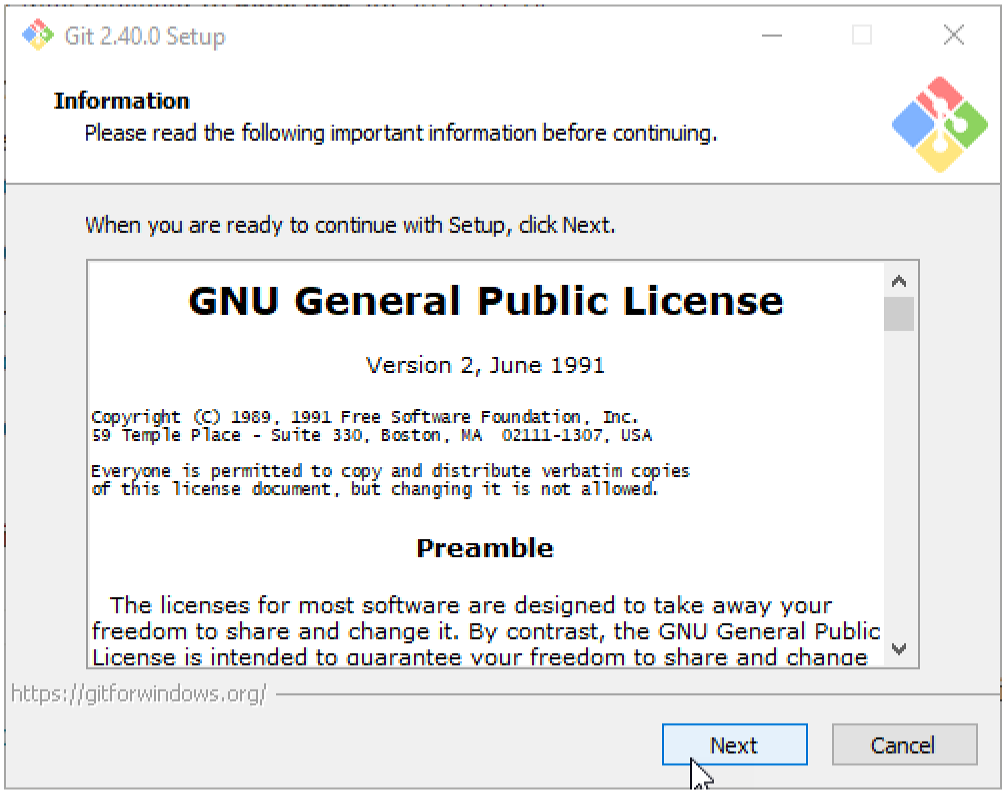
**Now What?**

Now that you have downloaded Git, it's time to start using it.

1. Navigate to https://git-scm.com/download/win
2. Click on the 'Click here to download' link to get the latest version of git for windows.

## 3.3 Continue Installation

Most options are default.



**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

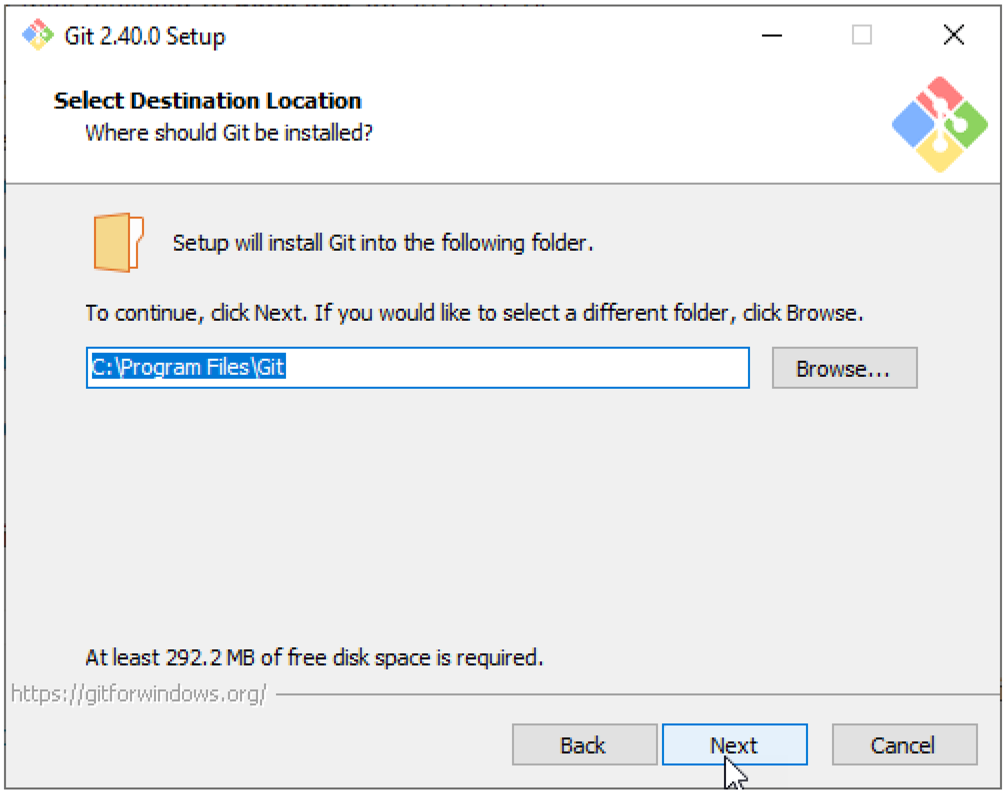
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

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



**Select Destination Location**

Where should Git be installed?

Setup will install Git into the following folder.

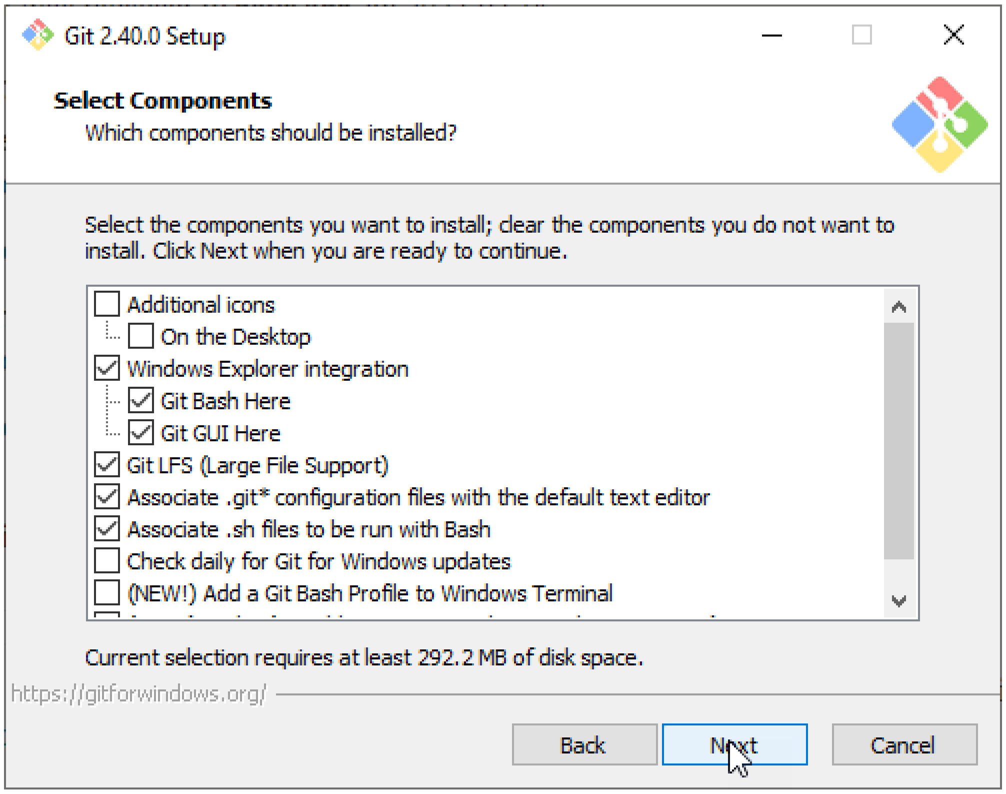
To continue, click Next. If you would like to select a different folder, click Browse.

C:\Program Files\Git

At least 292.2 MB of free disk space is required.

https://gitforwindows.org/

Next



Git 2.40.0 Setup

**Select Components**

Which components should be installed?

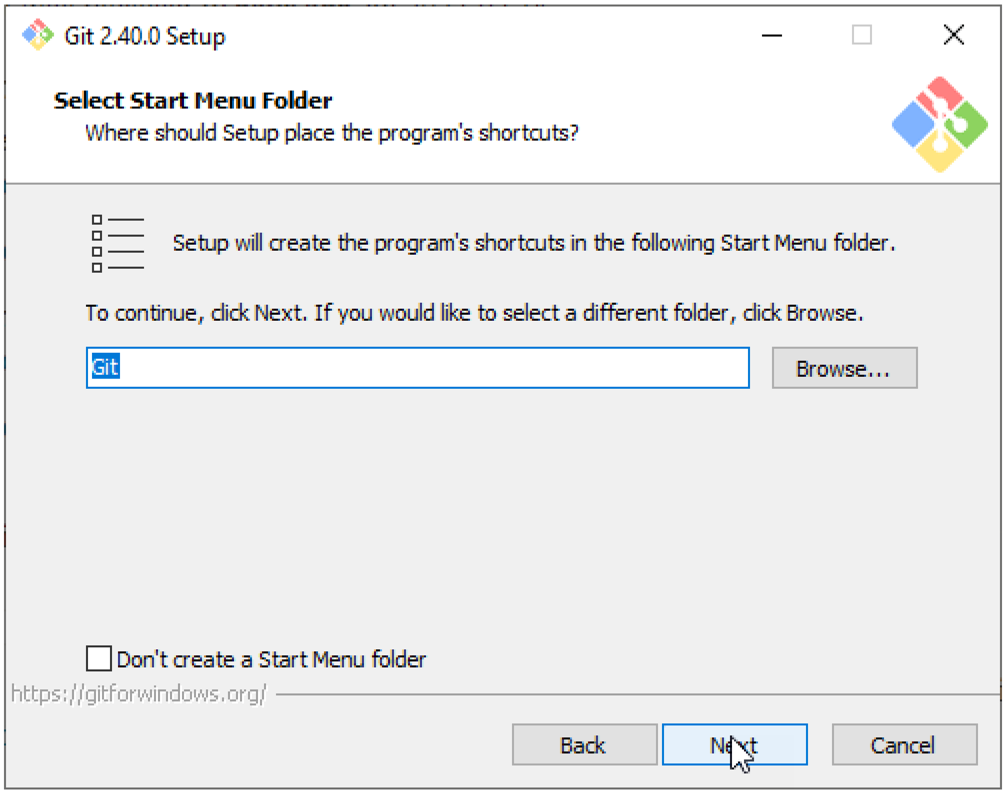
Select the components you want to install; clear the components you do not want to install. Click Next when you are ready to continue.

- [ ] Additional icons
  - [ ] On the Desktop
- [x] Windows Explorer integration
  - [x] Git Bash Here
  - [x] Git GUI Here
- [x] Git LFS (Large File Support)
- [x] Associate .git* configuration files with the default text editor
- [x] 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 292.2 MB of disk space.

https://gitforwindows.org/

Next



Git 2.40.0 Setup

**Select Start Menu Folder**

Where should Setup place the program's shortcuts?

Setup will create the program's shortcuts in the following Start Menu folder.

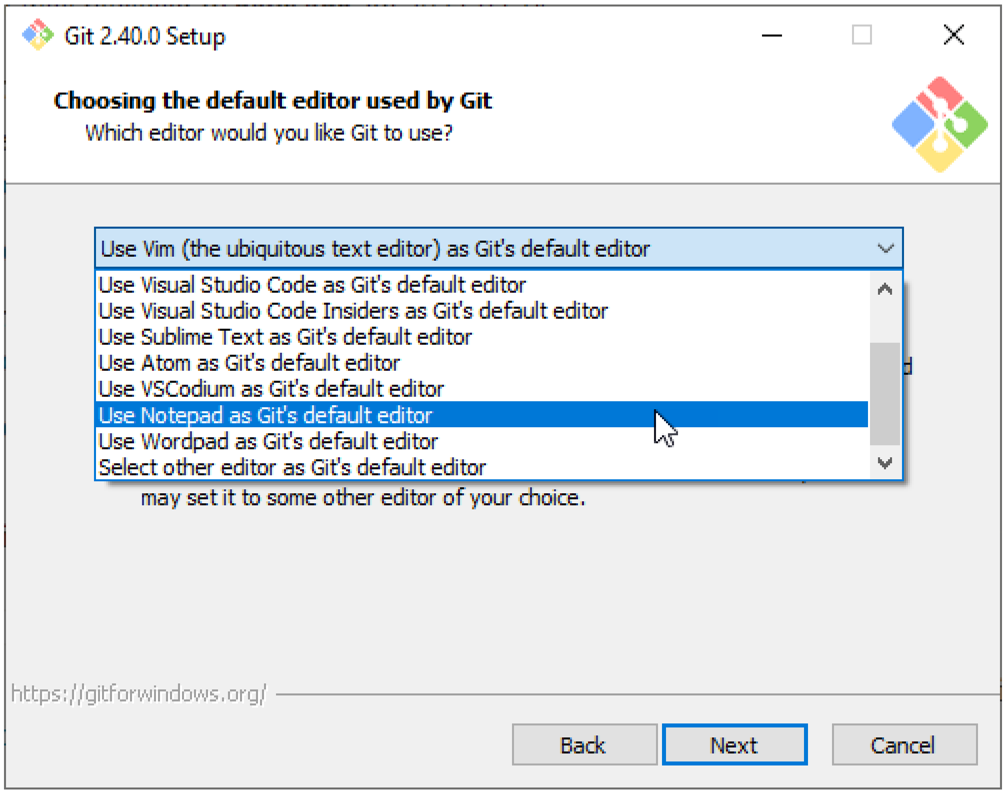
To continue, click Next. If you would like to select a different folder, click Browse.

Git

- [ ] Don't create a Start Menu folder

https://gitforwindows.org/

Next



Git 2.40.0 Setup

**Choosing the default editor used by Git**

Which editor would you like Git to use?

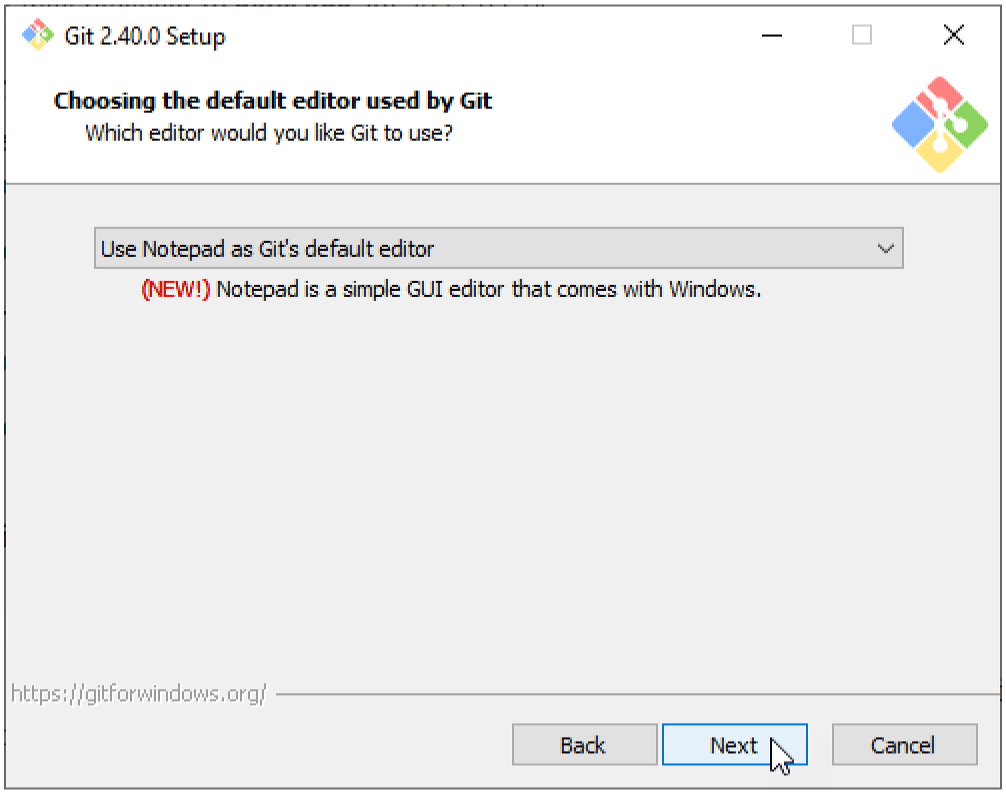
Use Vim (the ubiquitous text editor) as Git's default editor

- 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 VSCodium 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

may set it to some other editor of your choice.

https://gitforwindows.org/

Select Notepad for git's default editor. (It is hardly used).



Git 2.40.0 Setup

**Choosing the default editor used by Git**

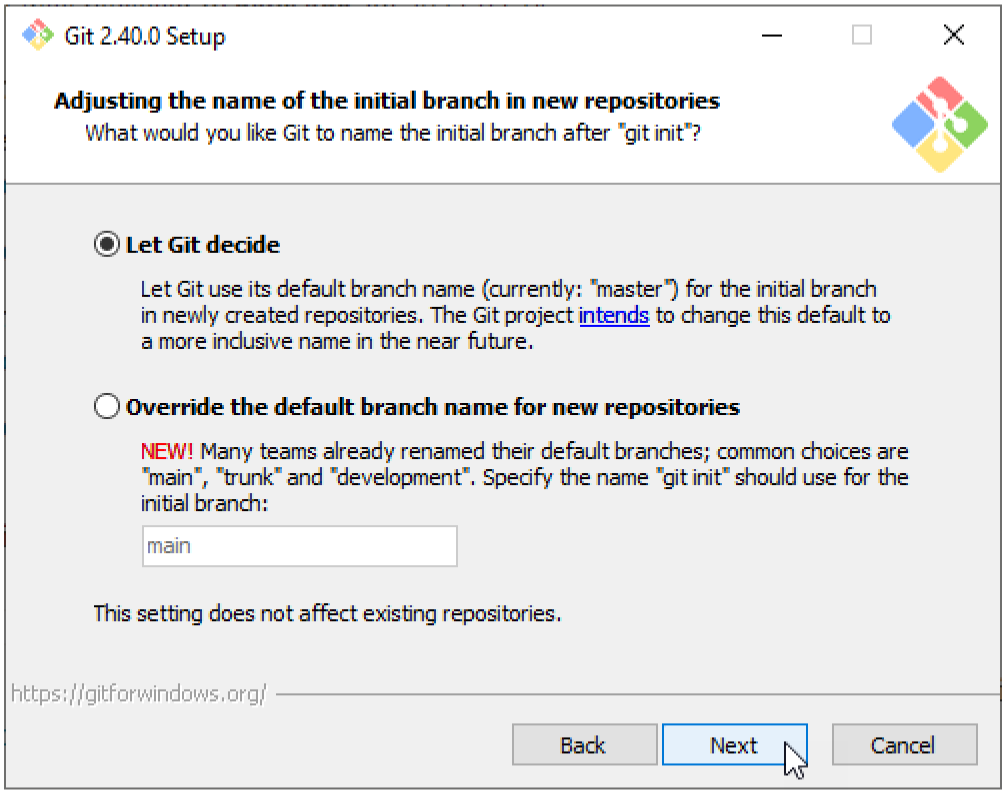
Which editor would you like Git to use?

Use Notepad as Git's default editor

(NEW!) Notepad is a simple GUI editor that comes with Windows.

https://gitforwindows.org/

Next



Git 2.40.0 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.

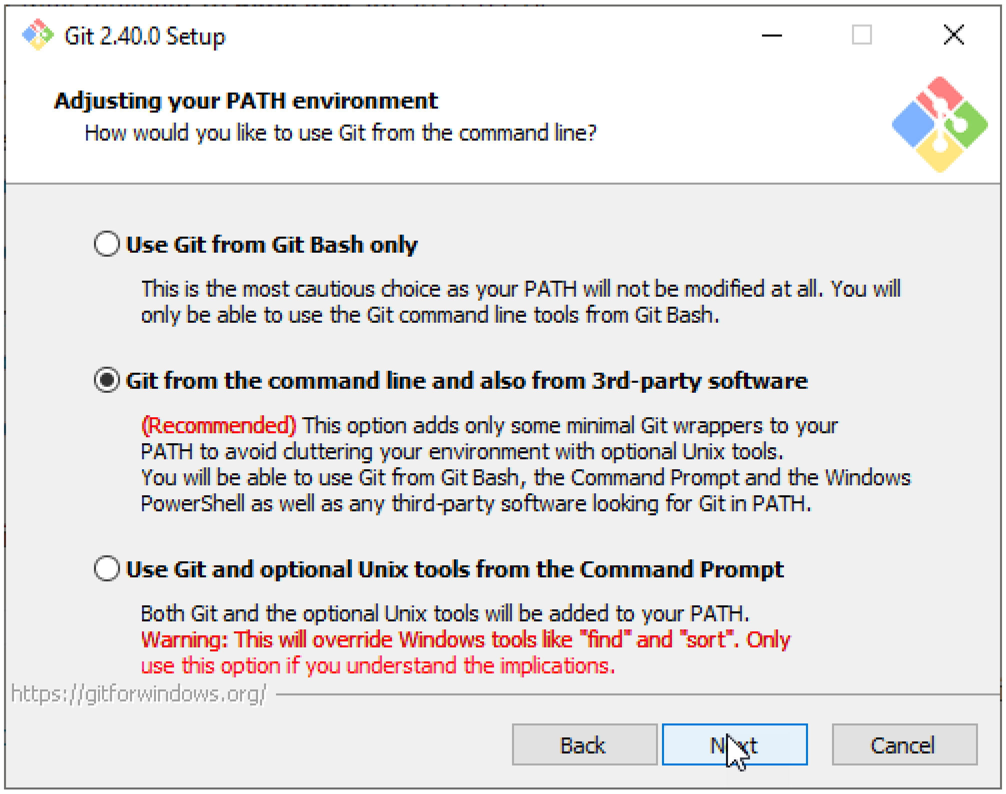
**Override the default branch name for new repositories**

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.

Next



Git 2.40.0 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.

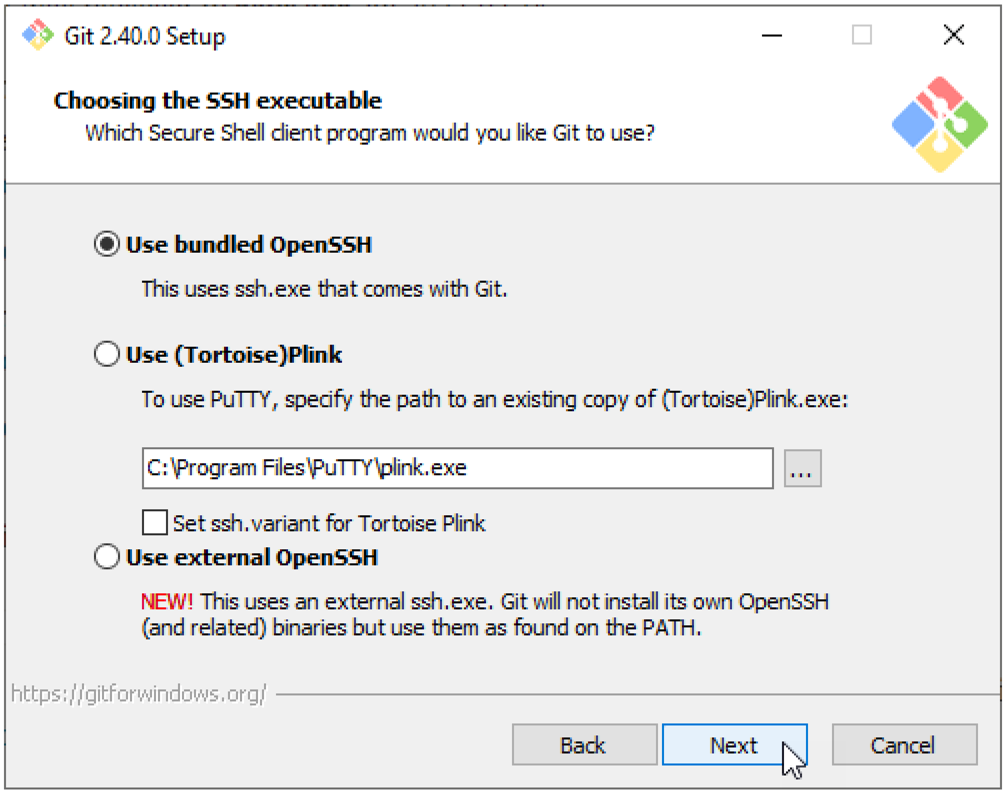
[✓ **Git from the command line and also from 3rd-party software**]

(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.

Next


Git 2.40.0 Setup

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

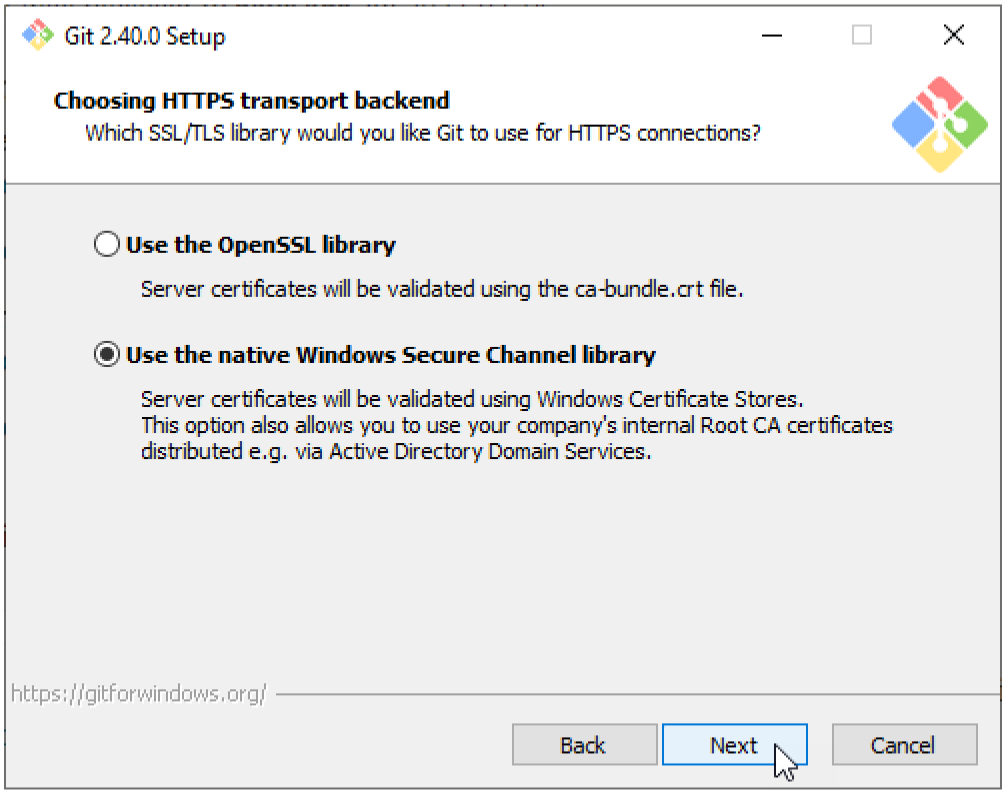
(●) **Use bundled OpenSSH**
This uses ssh.exe that comes with Git.

( ) **Use (Tortoise)Plink**
To use PuTTY, specify the path to an existing copy of (Tortoise)Plink.exe:
C:\Program Files\PuTTY\plink.exe
[ ] Set ssh.variant for Tortoise Plink

( ) **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/

Next


Git 2.40.0 Setup

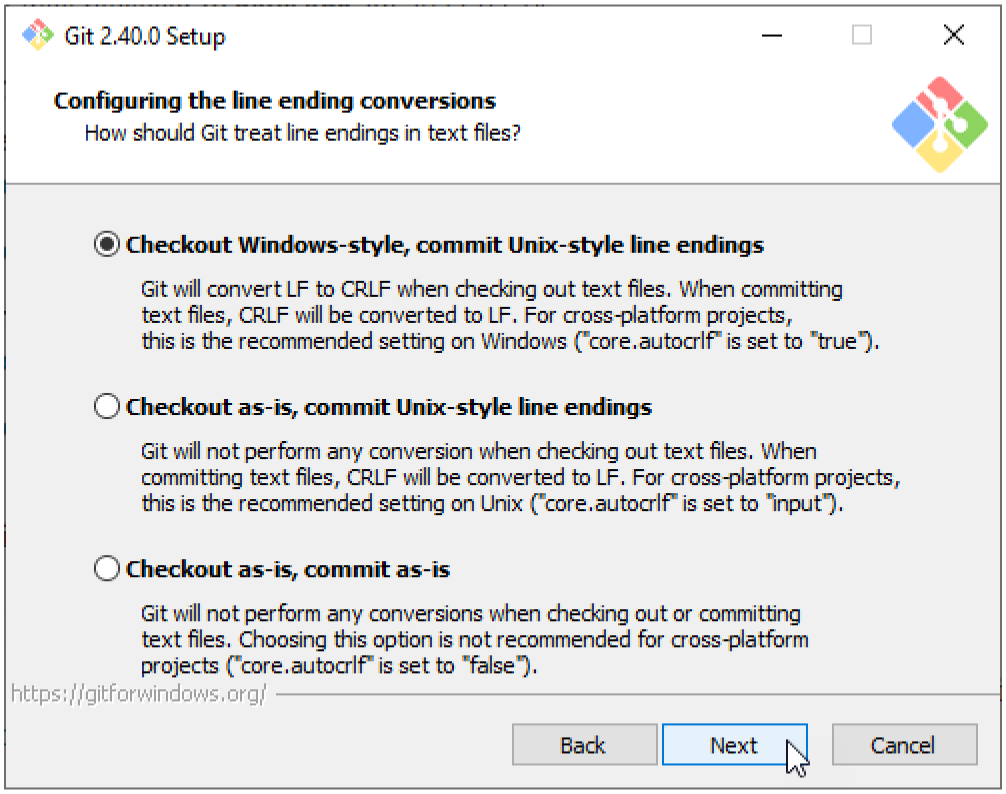
**Choosing HTTPS transport backend**
Which SSL/TLS library would you like Git to use for HTTPS connections?

( ) **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.

https://gitforwindows.org/

Next



Git 2.40.0 Setup

**Configuring the line ending conversions**

How should Git treat line endings in text files?

[✓ **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.autocrlf" is set to "true").

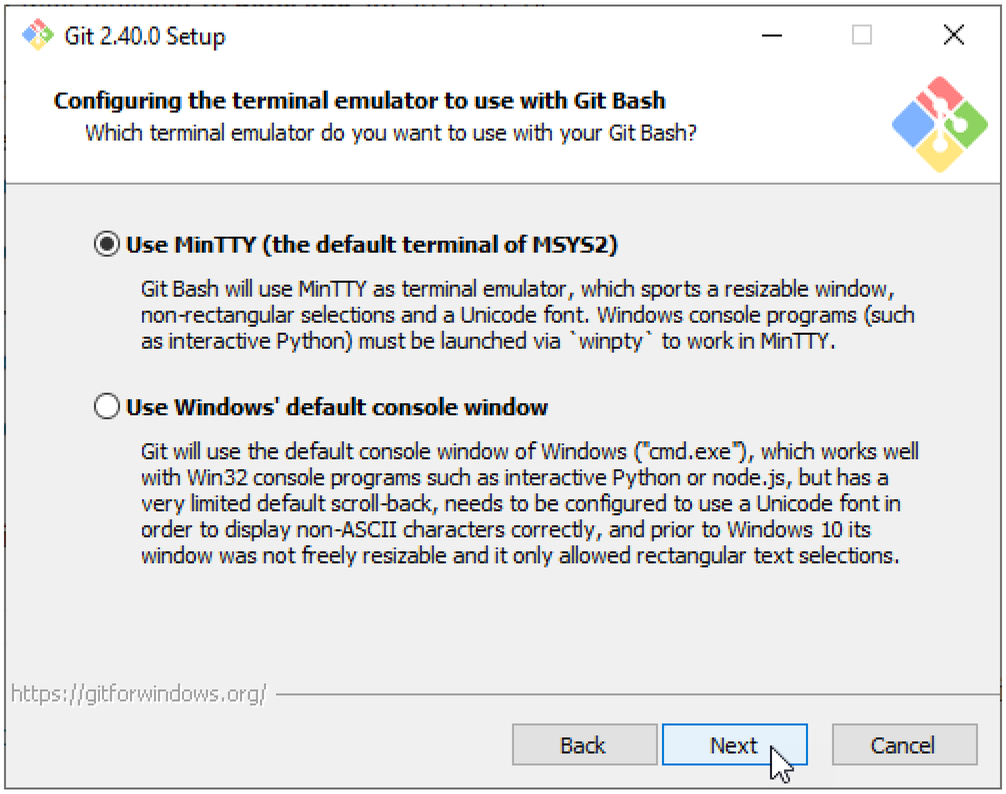
**Checkout as-is, commit Unix-style line endings**

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").

**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").

Next



Git 2.40.0 Setup

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

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

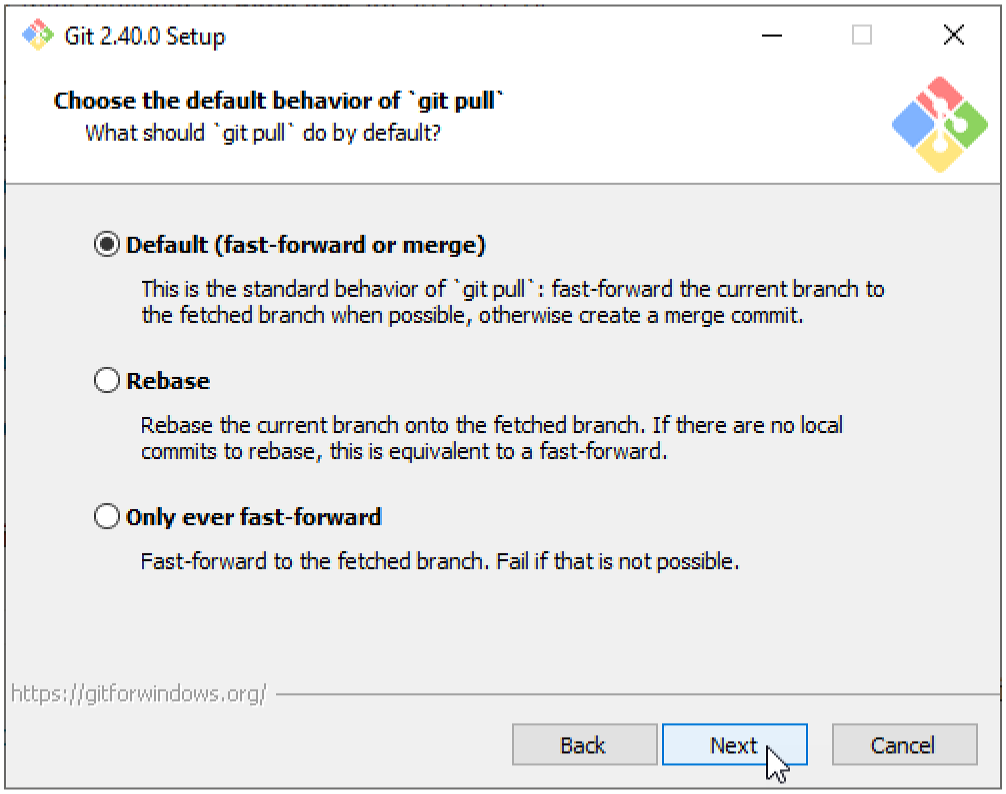
[✓ **Use MinTTY (the default terminal of MSYS2)**]

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.

**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.

Next


Git 2.40.0 Setup

**Choose the default behavior of `git pull`**

What should `git pull` do by default?

[● ] **Default (fast-forward or merge)**

This is the standard behavior of `git pull`: fast-forward the current branch to the fetched branch when possible, otherwise create a merge commit.

○ **Rebase**

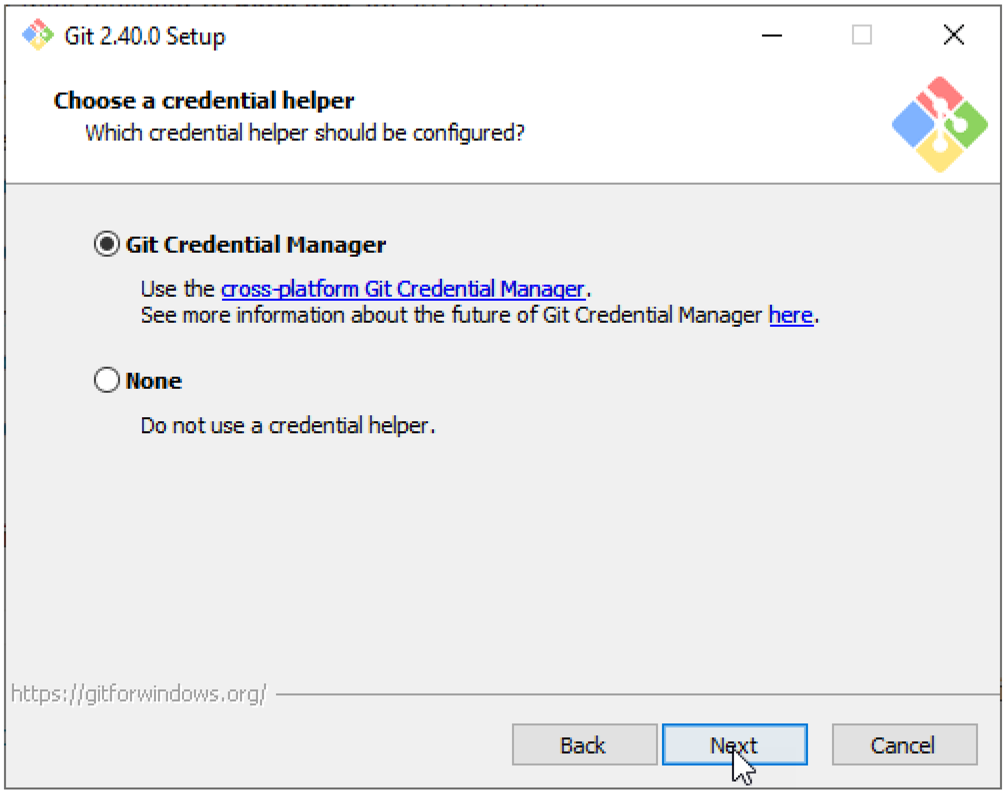
Rebase the current branch onto the fetched branch. If there are no local commits to rebase, this is equivalent to a fast-forward.

○ **Only ever fast-forward**

Fast-forward to the fetched branch. Fail if that is not possible.

https://gitforwindows.org/

Next


Git 2.40.0 Setup

**Choose a credential helper**

Which credential helper should be configured?

[● ] **Git Credential Manager**

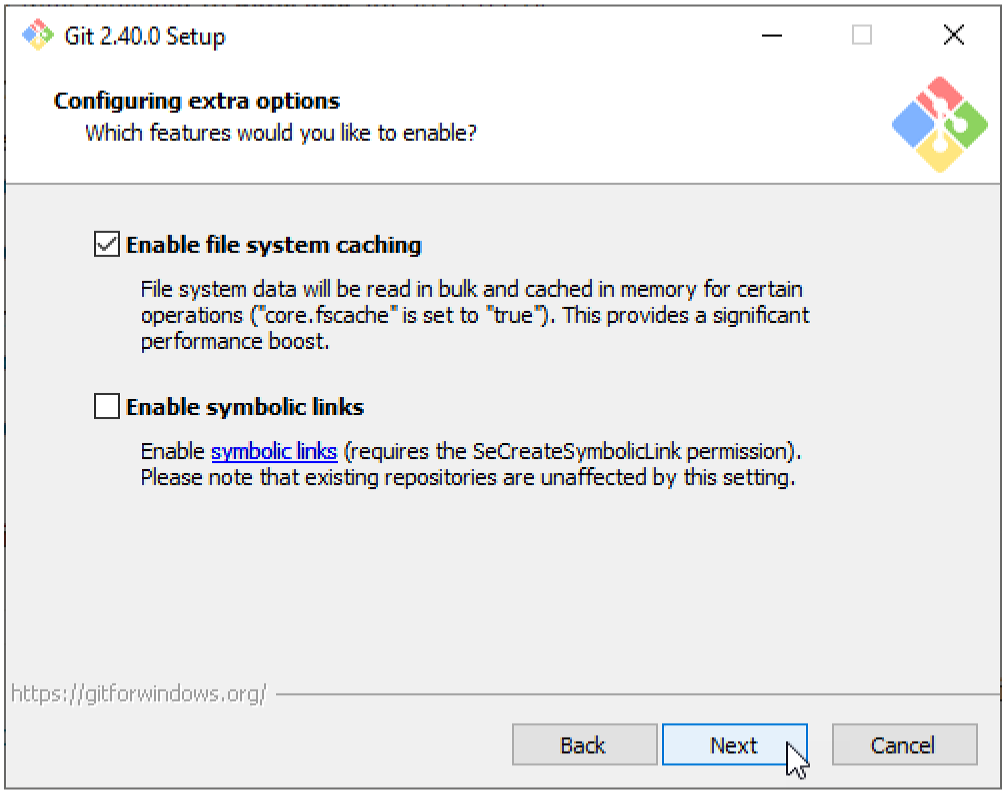
Use the cross-platform Git Credential Manager.
See more information about the future of Git Credential Manager here.

○ **None**

Do not use a credential helper.

https://gitforwindows.org/

Next



**Configuring extra options**

Which features would you like to enable?

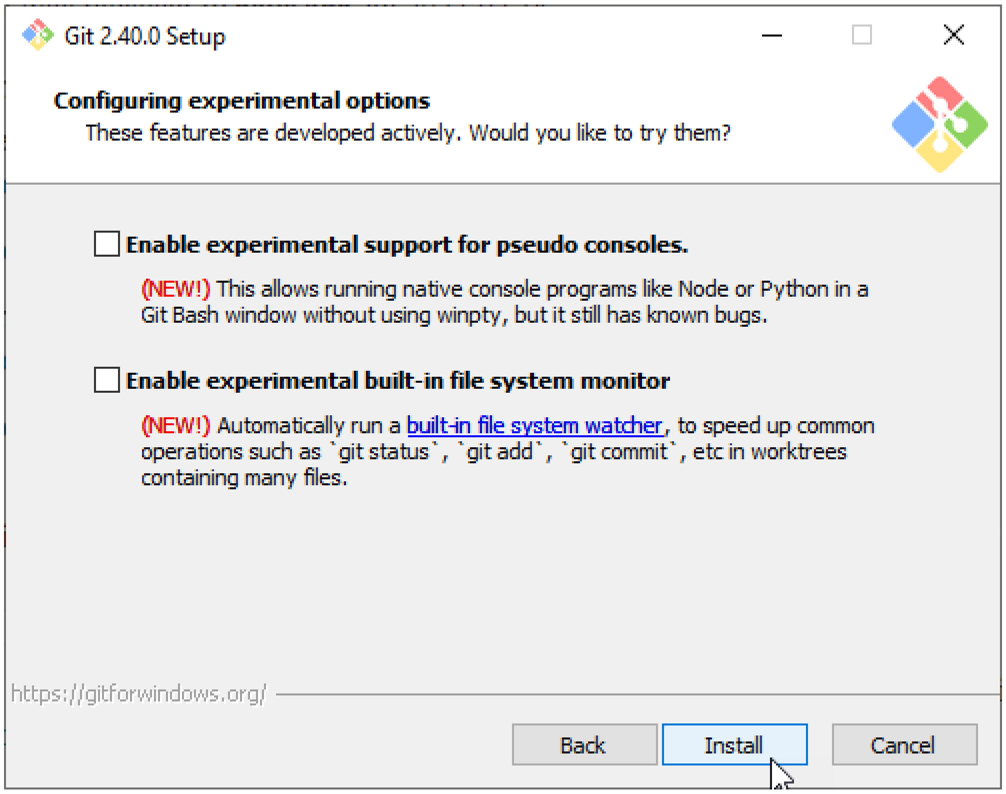
[✓] **Enable file system caching**

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.

[ ] **Enable symbolic links**

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

Next



**Configuring experimental options**

These features are developed actively. Would you like to try them?

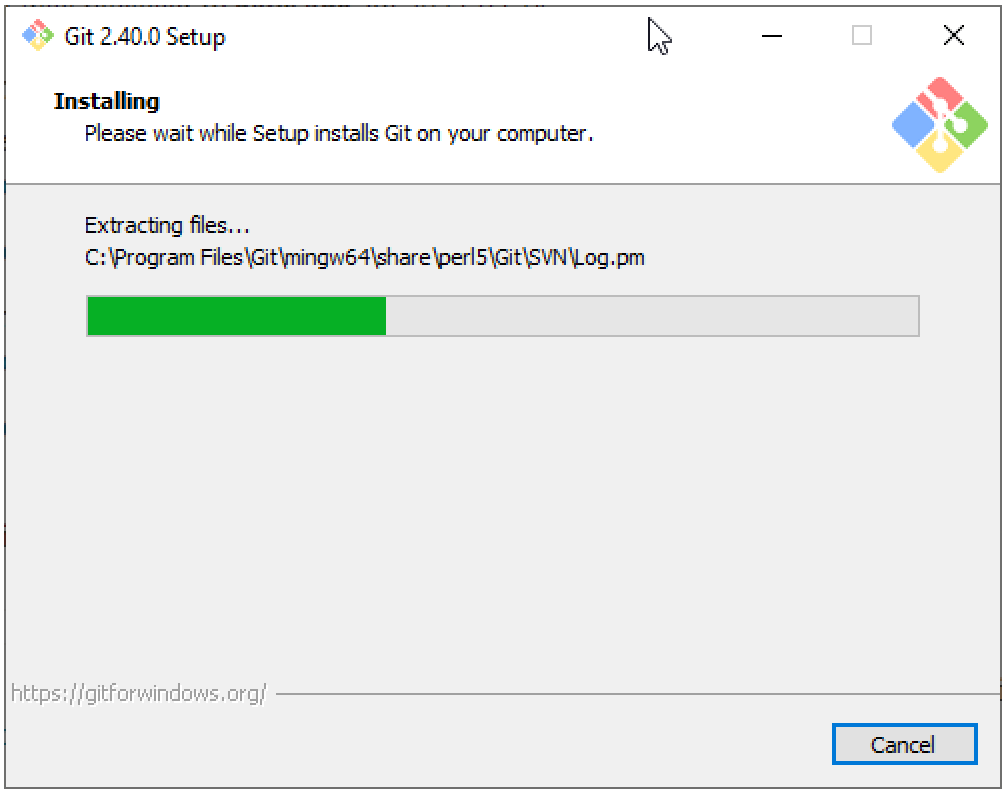
[ ] **Enable experimental support for pseudo consoles.**

(NEW!) This allows running native console programs like Node or Python in a Git Bash window without using winpty, but it still has known bugs.

[ ] **Enable experimental built-in file system monitor**

(NEW!) Automatically run a built-in file system watcher, to speed up common operations such as \`git status\`, \`git add\`, \`git commit\`, etc in worktrees containing many files.

Next


Git 2.40.0 Setup

**Installing**

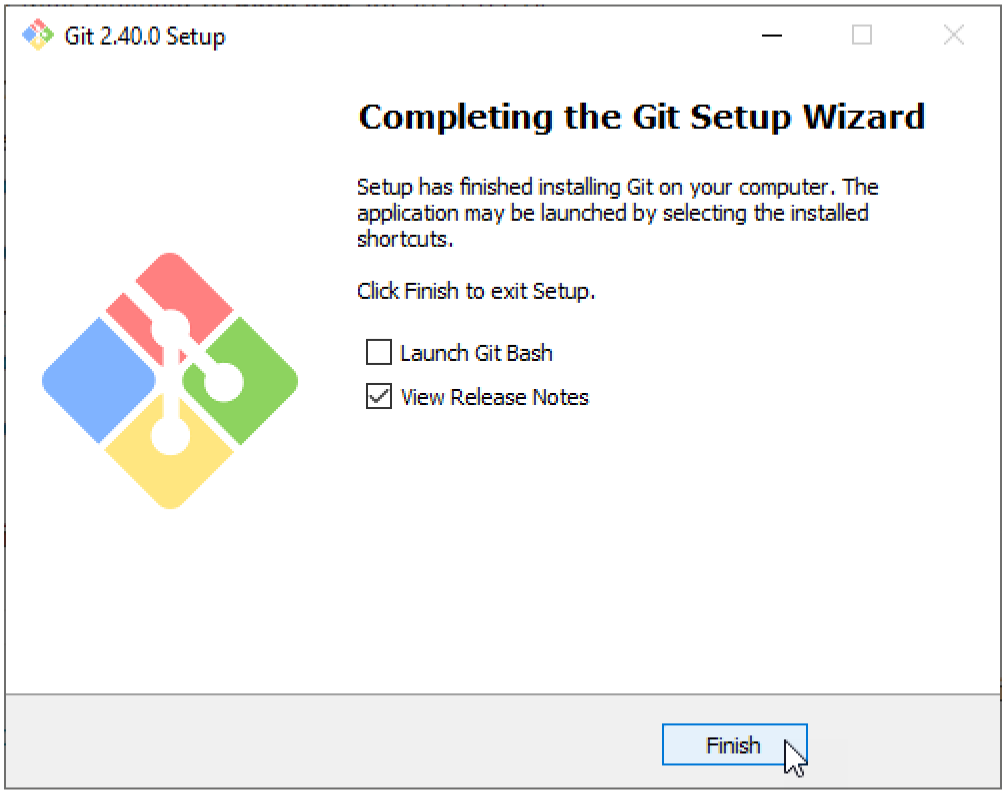
Please wait while Setup installs Git on your computer.

Extracting files...

C:\Program Files\Git\mingw64\share\perl5\Git\SVN\Log.pm

https://gitforwindows.org/

*Wait* for installation to finish


Git 2.40.0 Setup

**Completing the Git Setup Wizard**

Setup has finished installing Git on your computer. The application may be launched by selecting the installed shortcuts.

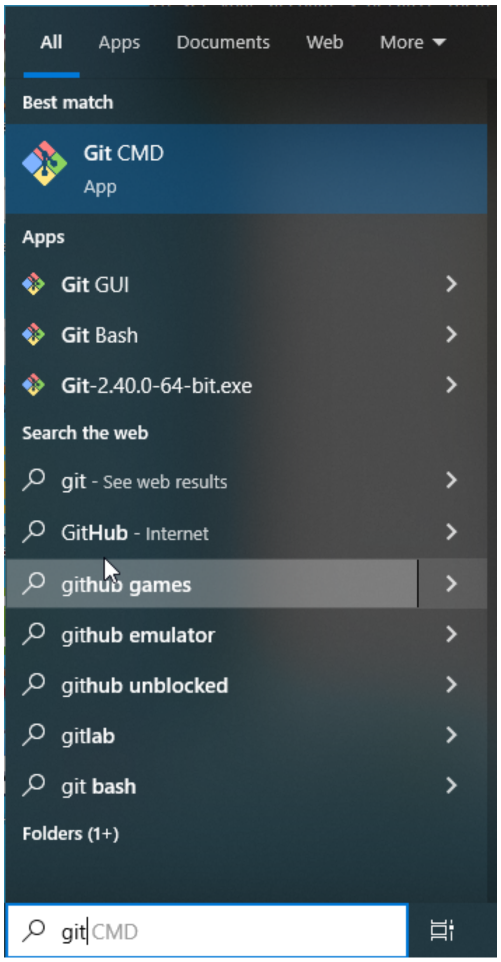
Click Finish to exit Setup.

☐ Launch Git Bash

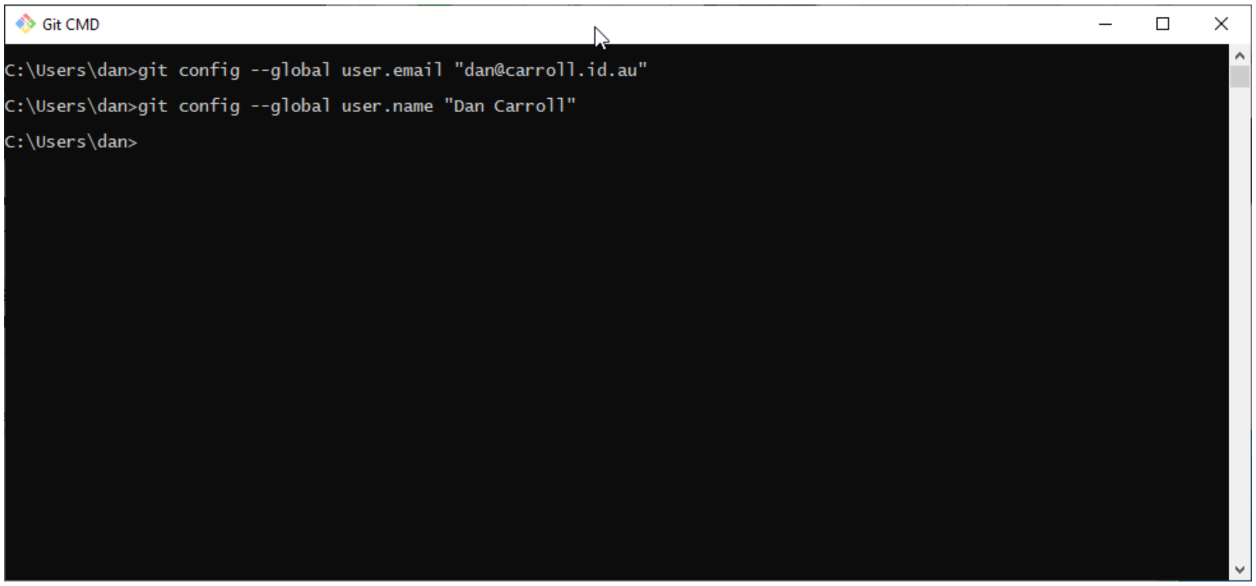
☑ View Release Notes

Click Finish

## 3.4 Setup Git user



Type 'git' into the windows search bar and run Git CMD.

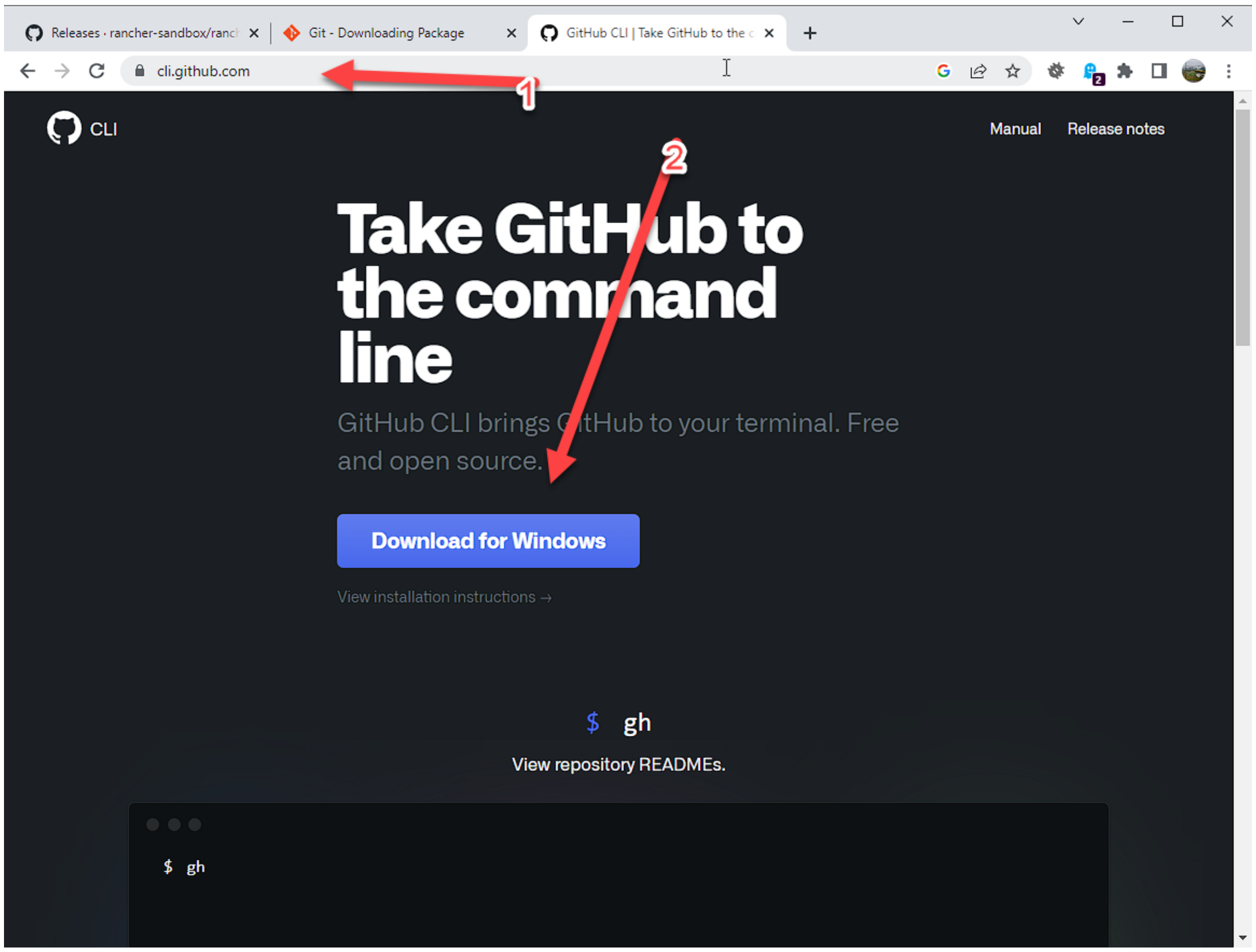


```
C:\Users\dan>git config --global user.email "dan@carroll.id.au"

C:\Users\dan>git config --global user.name "Dan Carroll"

C:\Users\dan>
```

Execute the commands:

```
git config --global user.email "<Email Address>"
git config --global user.name "<Full Name>"
git config --global init.defaultBranch main
```

## 3.5 Download Github CLI

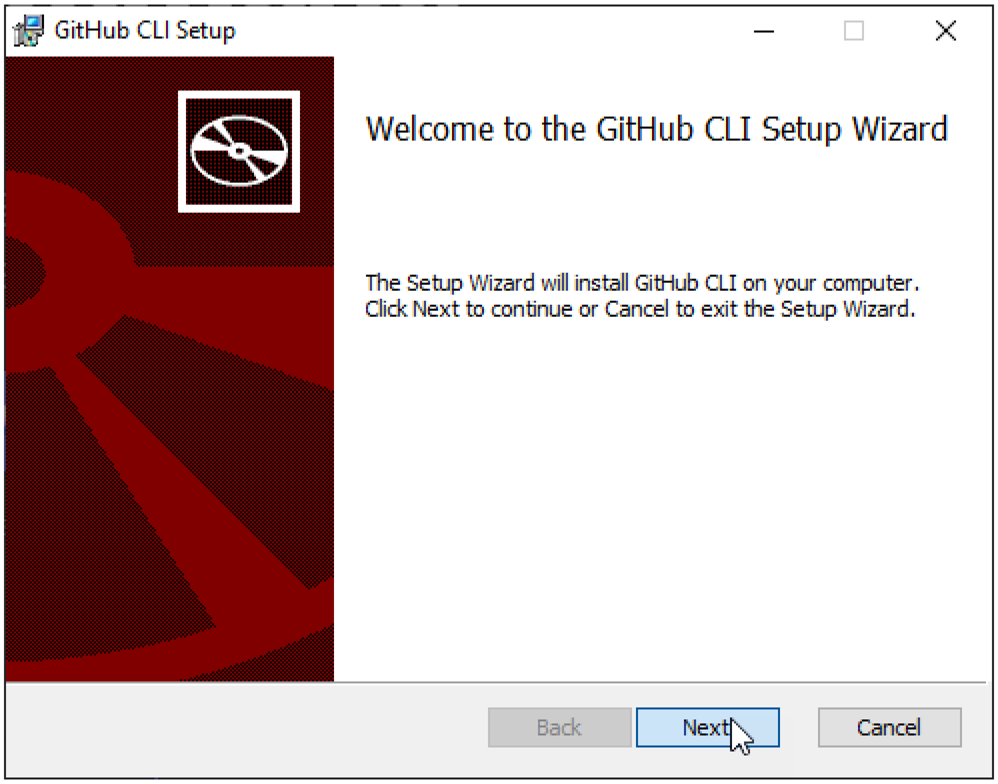

Releases · rancher-sandbox/ranc
Git - Downloading Package
GitHub CLI | Take GitHub to the c
cli.github.com
1
2
CLI
Manual
Release notes
Take GitHub to the command line
GitHub CLI brings GitHub to your terminal. Free and open source.
Download for Windows
View installation instructions →
$ gh
View repository READMEs.
$ gh

The Github CLI package contains some CLI tools to help git authenticate easily to GitHub.

1. Navigate to https://cli.github.com/
2. Click on "Download for Windows" to get the latest windows package.

## 3.6 Continue Installation

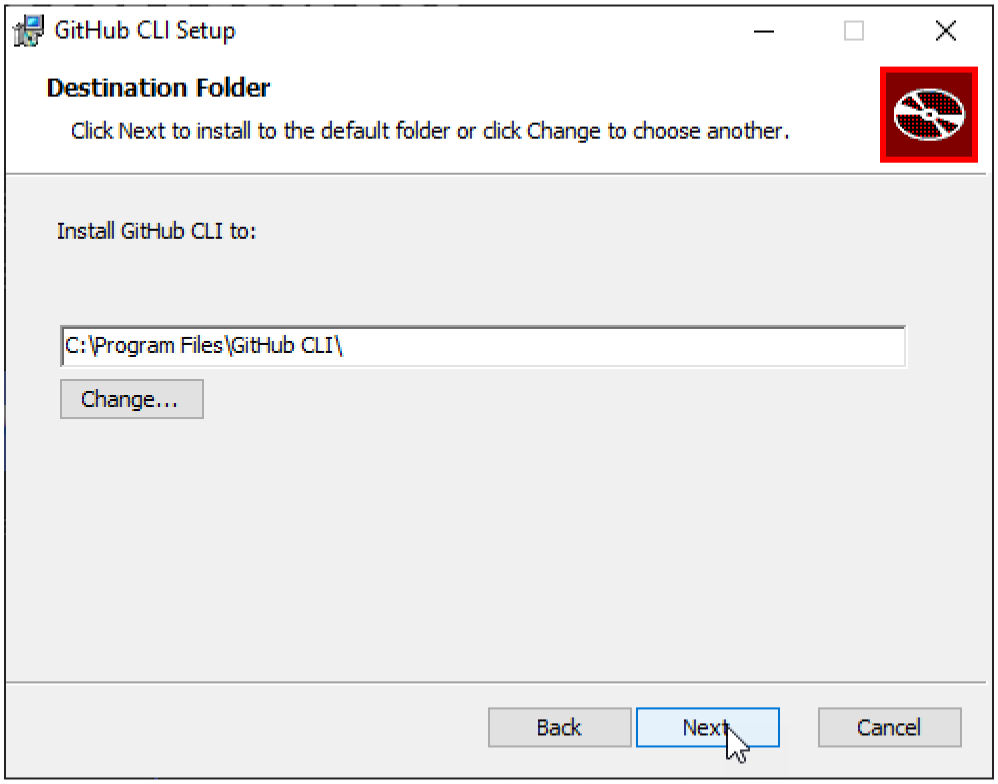
Most options are default.


GitHub CLI Setup

Welcome to the GitHub CLI Setup Wizard

The Setup Wizard will install GitHub CLI on your computer.
Click Next to continue or Cancel to exit the Setup Wizard.

Next


GitHub CLI Setup

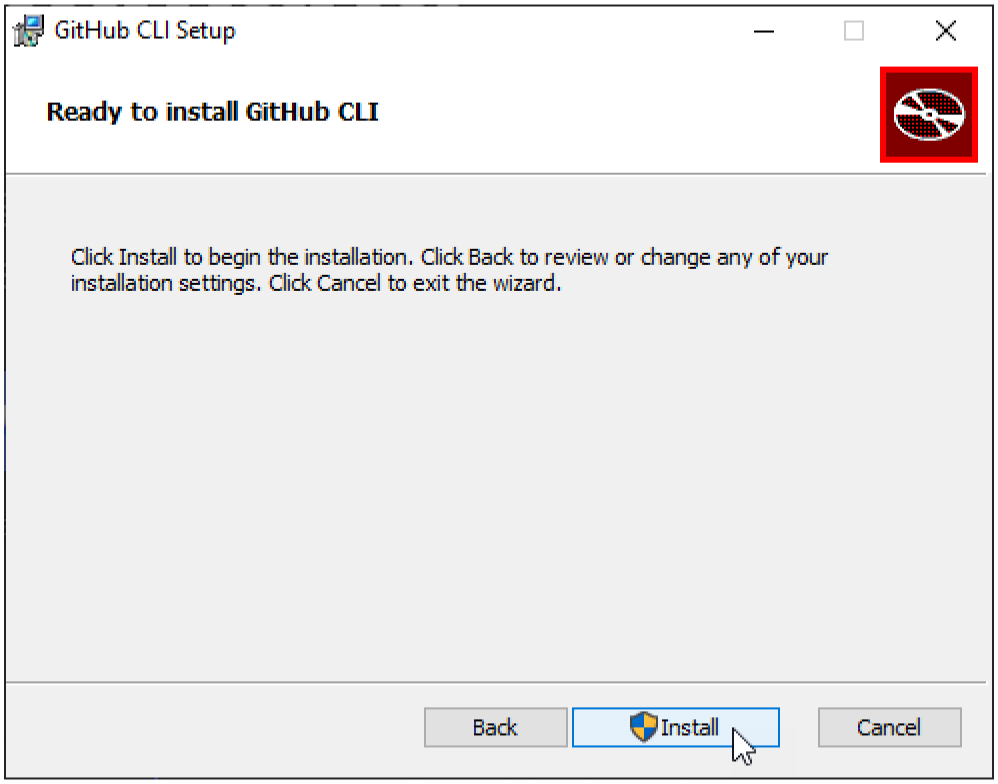
**Destination Folder**

Click Next to install to the default folder or click Change to choose another.

Install GitHub CLI to:

C:\Program Files\GitHub CLI\

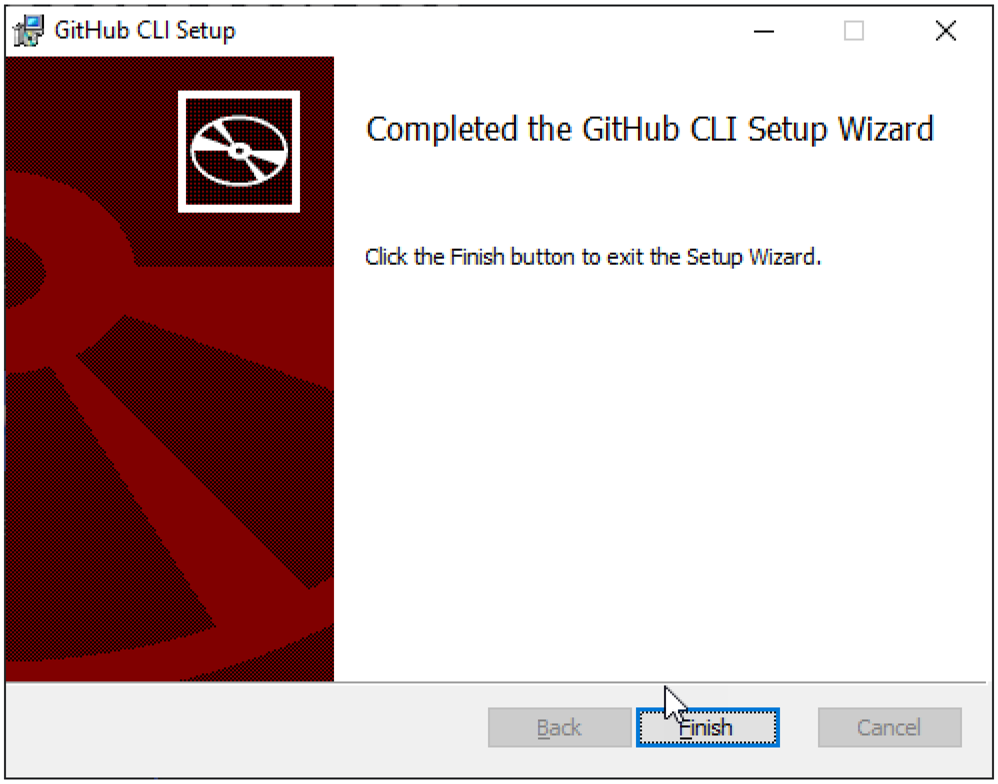
Next


GitHub CLI Setup

**Ready to install GitHub CLI**

Click Install to begin the installation. Click Back to review or change any of your installation settings. Click Cancel to exit the wizard.

Install (You will probably have to confirm changes)

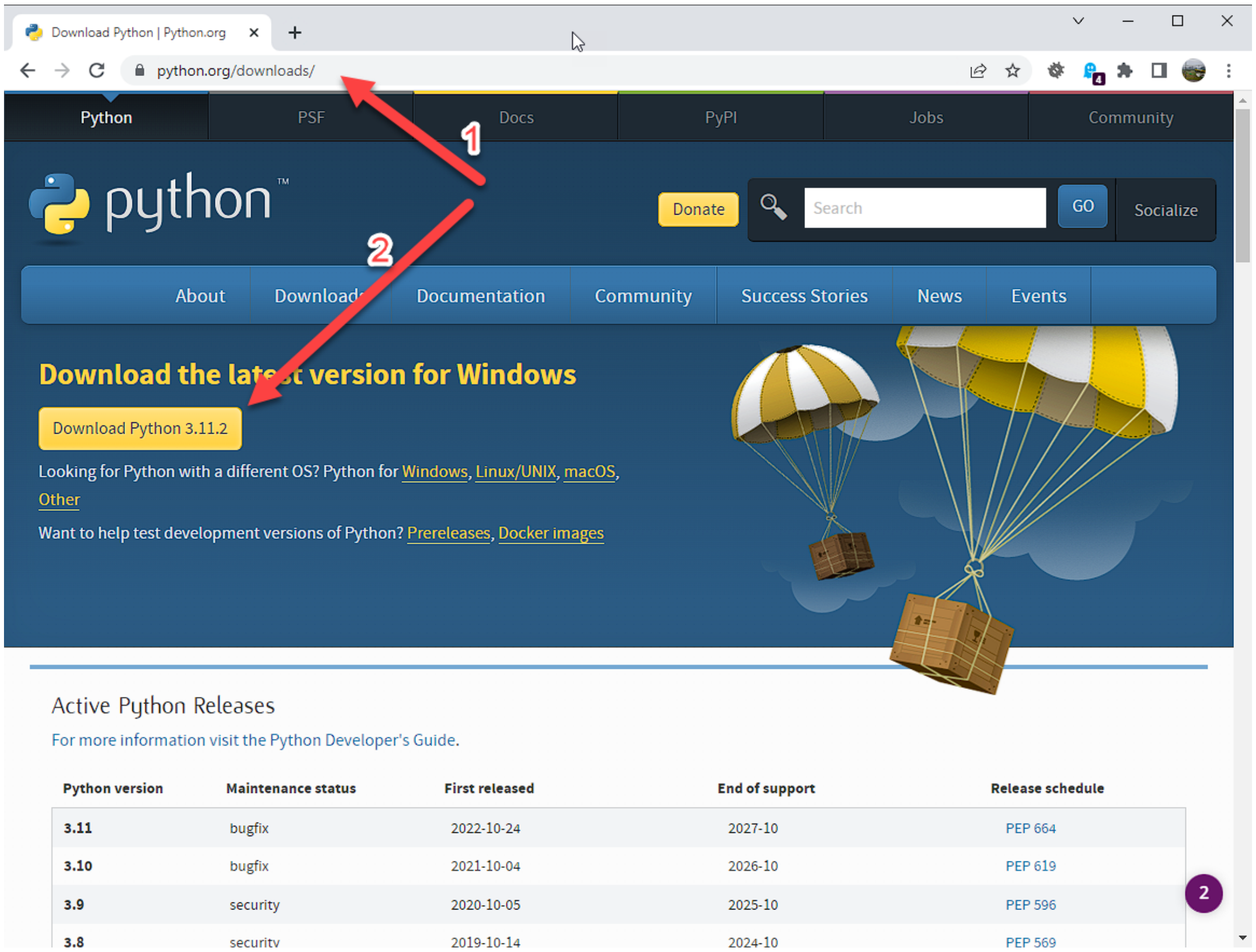


Completed the GitHub CLI Setup Wizard

Click the Finish button to exit the Setup Wizard.

Finish

## 3.7 Download Python



Download the latest version for Windows

Looking for Python with a different OS? Python for Windows, Linux/UNIX, macOS, Other

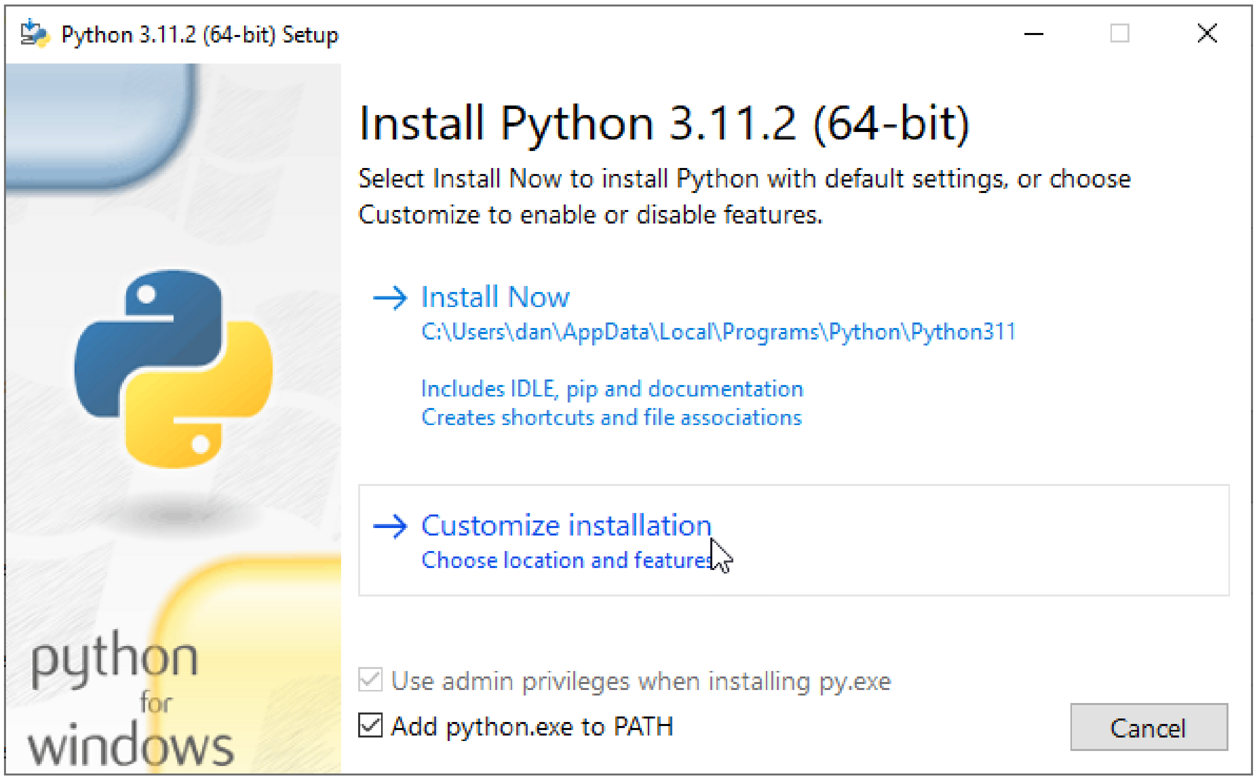
Want to help test development versions of Python? Prereleases, Docker images

Active Python Releases

For more information visit the Python Developer's Guide.

| Python version | Maintenance status | First released | End of support | Release schedule |
|---|---|---|---|---|
| 3.11 | bugfix | 2022-10-24 | 2027-10 | PEP 664 |
| 3.10 | bugfix | 2021-10-04 | 2026-10 | PEP 619 |
| 3.9 | security | 2020-10-05 | 2025-10 | PEP 596 |
| 3.8 | security | 2019-10-14 | 2024-10 | PEP 569 |

1. Navigate to https://python.org/download
2. Click on the ‘Download Python XXX’ link to get the latest version of python for windows.



Python 3.11.2 (64-bit) Setup

Install Python 3.11.2 (64-bit)

Select Install Now to install Python with default settings, or choose Customize to enable or disable features.

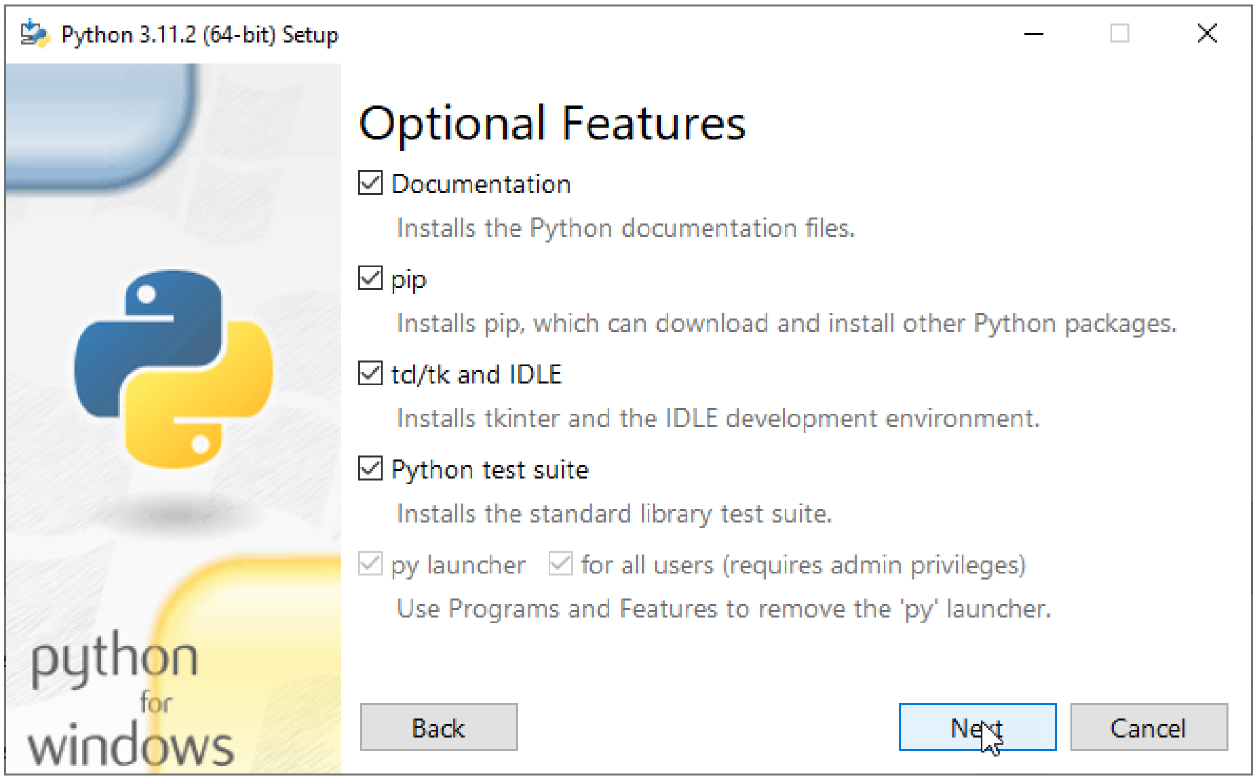
→ Install Now
C:\Users\dan\AppData\Local\Programs\Python\Python311

Includes IDLE, pip and documentation
Creates shortcuts and file associations

→ Customize installation
Choose location and features

☑ Use admin privileges when installing py.exe
☑ Add python.exe to PATH

Make sure to select: “Add python.exe to PATH” and click “Customize installation”.



Python 3.11.2 (64-bit) Setup

Optional Features

☑ Documentation
Installs the Python documentation files.

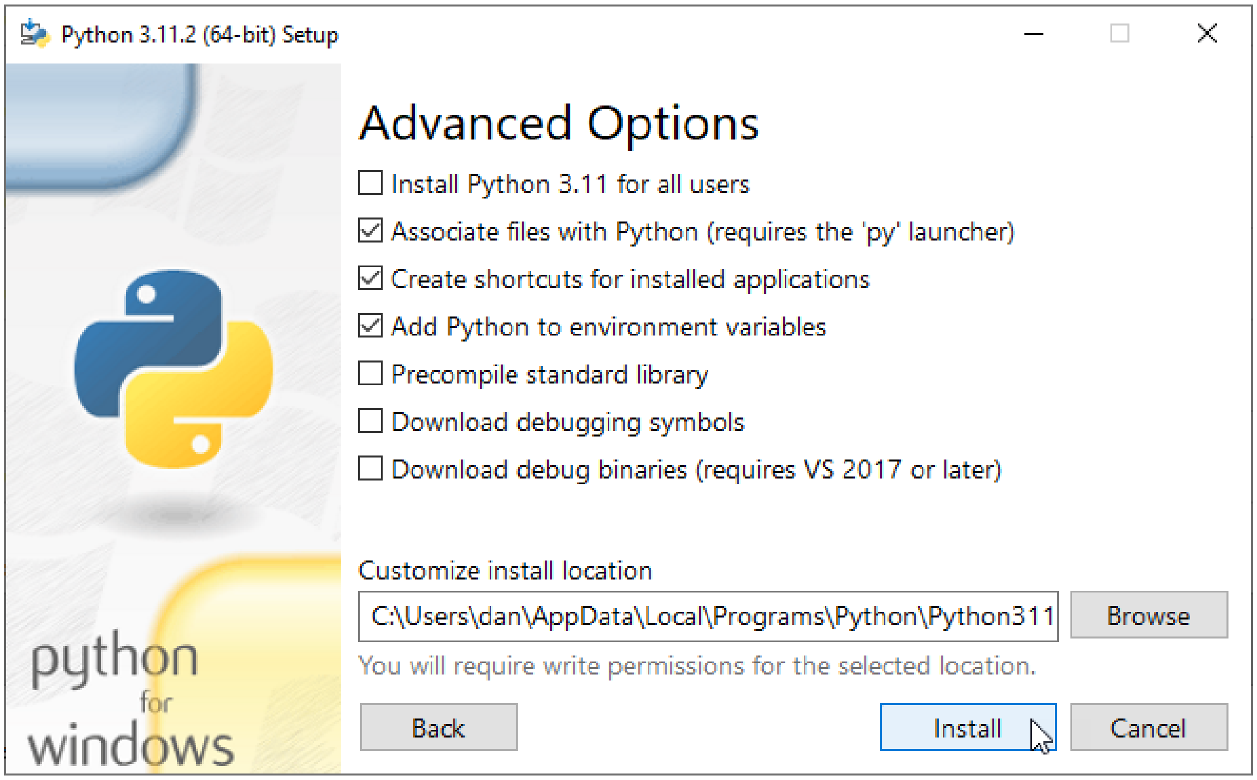
☑ pip
Installs pip, which can download and install other Python packages.

☑ tcl/tk and IDLE
Installs tkinter and the IDLE development environment.

☑ Python test suite
Installs the standard library test suite.

☑ py launcher ☑ for all users (requires admin privileges)
Use Programs and Features to remove the 'py' launcher.

Make sue all ‘Optional Features’ are checked and click “Next”.



Python 3.11.2 (64-bit) Setup

Advanced Options

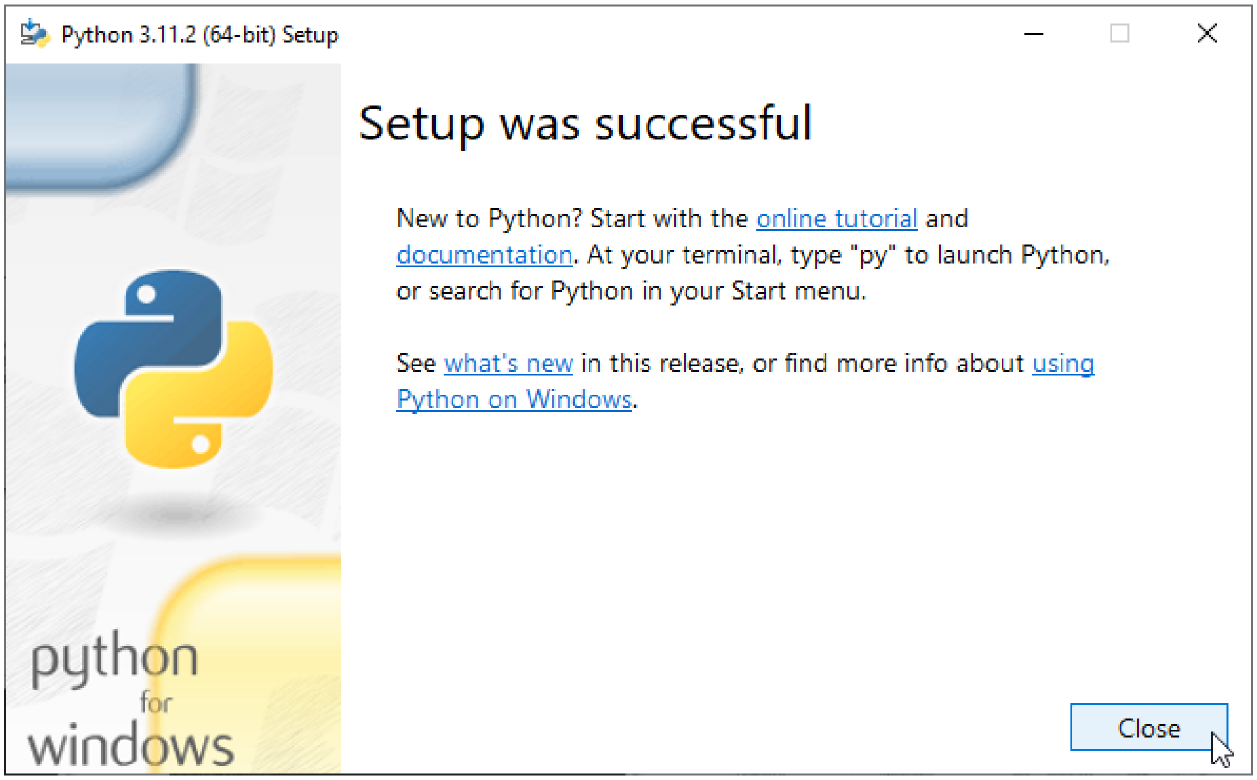
- [ ] Install Python 3.11 for all users
- [x] Associate files with Python (requires the 'py' launcher)
- [x] Create shortcuts for installed applications
- [x] Add Python to environment variables
- [ ] Precompile standard library
- [ ] Download debugging symbols
- [ ] Download debug binaries (requires VS 2017 or later)

Customize install location

C:\Users\dan\AppData\Local\Programs\Python\Python311

You will require write permissions for the selected location.

Make sure “Associate Files”, “Create shortcuts” and “Add Python to environment variables” are all checked and click “Install”.



Python 3.11.2 (64-bit) Setup

Setup was successful

New to Python? Start with the online tutorial and documentation. At your terminal, type "py" to launch Python, or search for Python in your Start menu.

See what's new in this release, or find more info about using Python on Windows.

When the installation is complete, click Close.

# 4 Editing Sphinx documents

By this stage you should have the following:

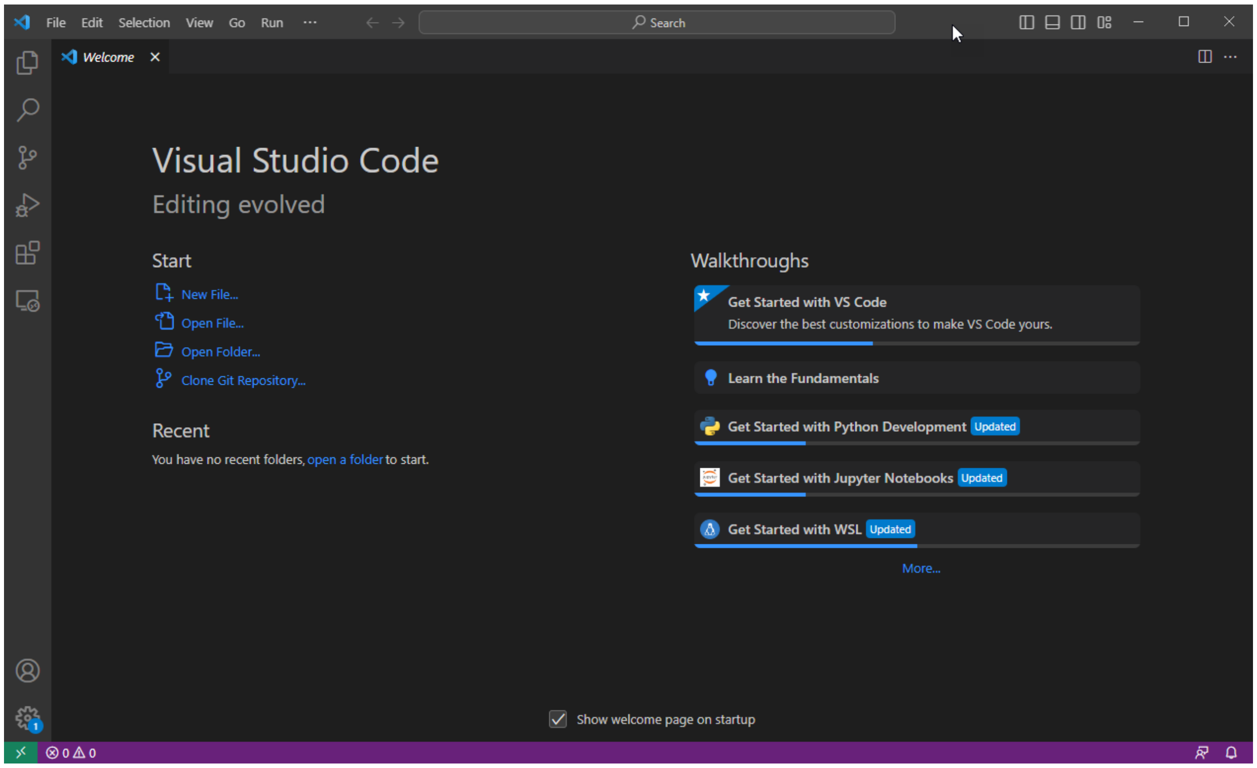
- A working git installation.
- Visual Studio installed with a few useful plugins.
- Python3 installed.

So lets build a basic document, single page with a standard table of contents. And then we'll build it into html with the locally installed python and sphinx.

## 4.1 Create a new working folder

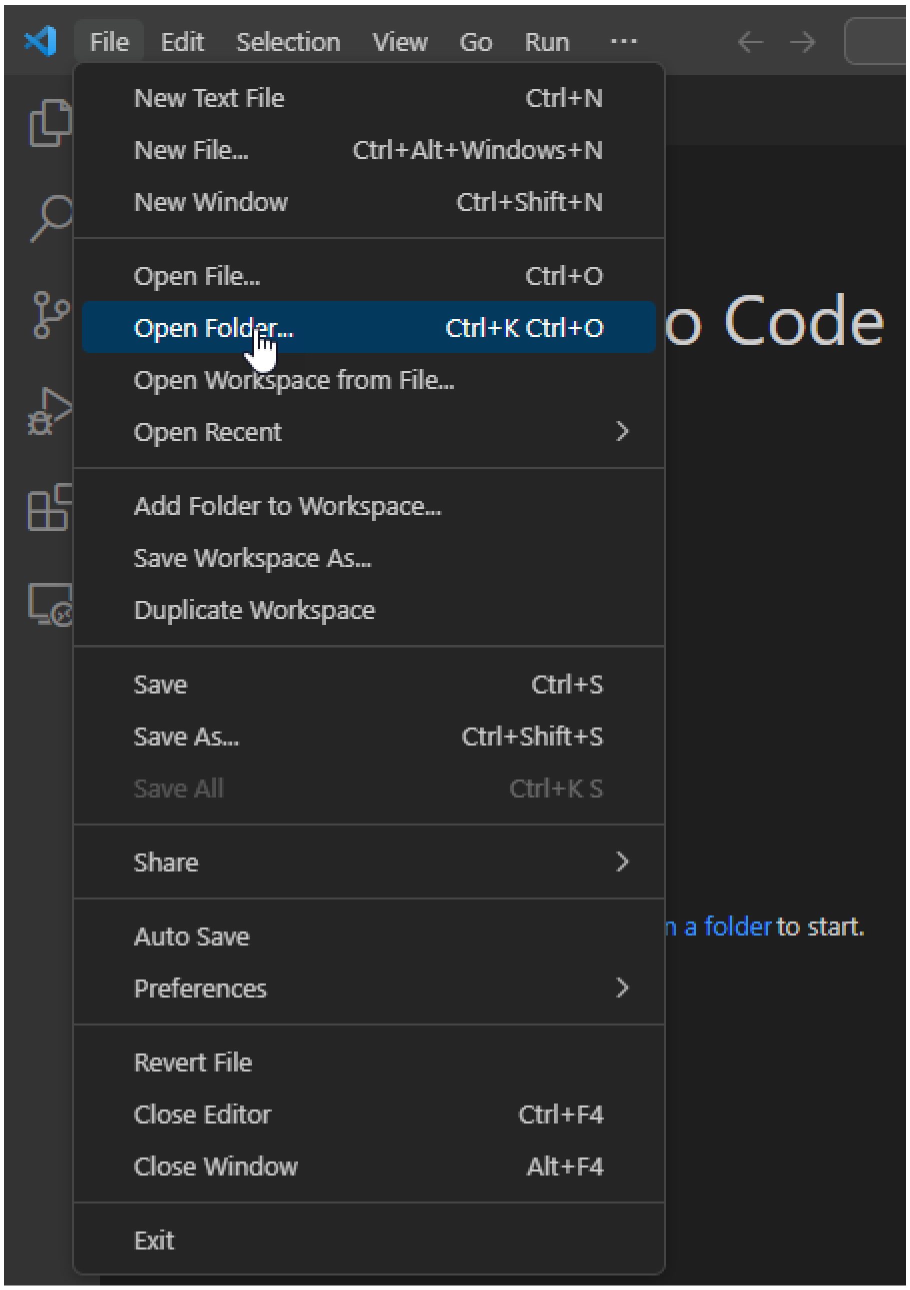
Working folders are a little like projects. You can think of the folder as a new project directory for your documentation.

Open up VS Code. You should be at the welcome screen. You can click on the "Open Folder" link, or go via the menu to the *File* → *Open Folder* selection.

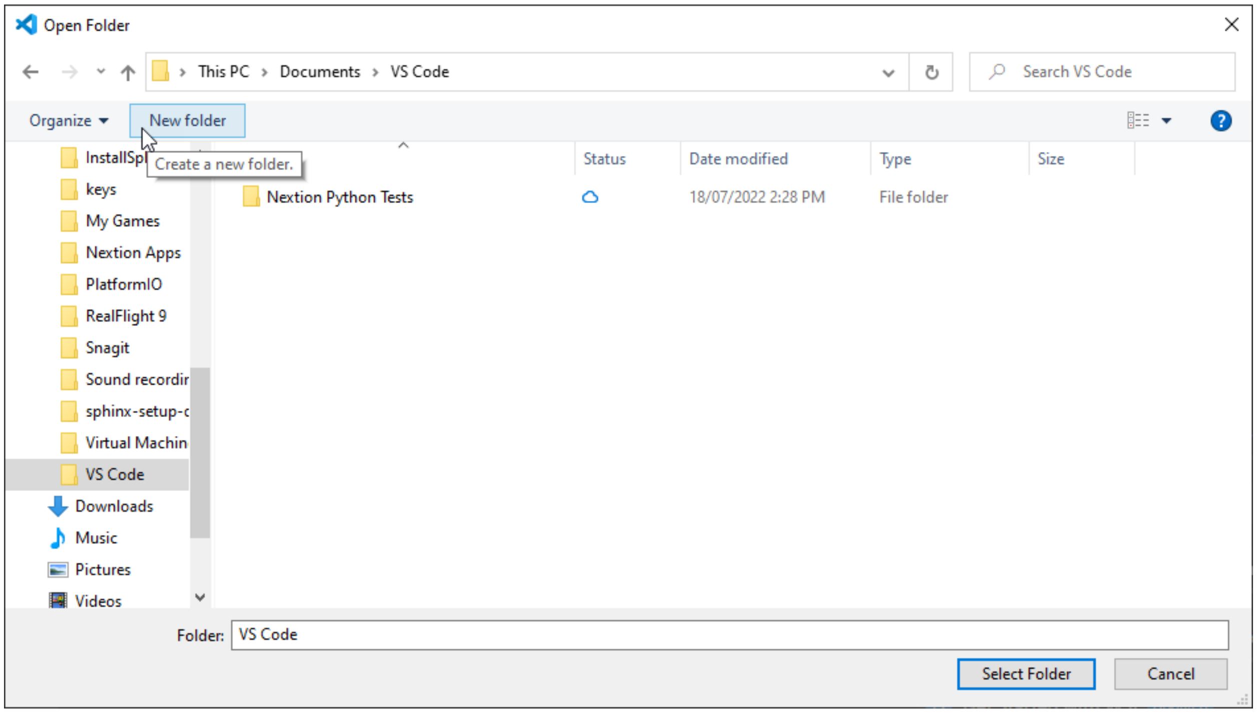


You have no recent folders, open a folder to start.

Discover the best customizations to make VS Code yours.

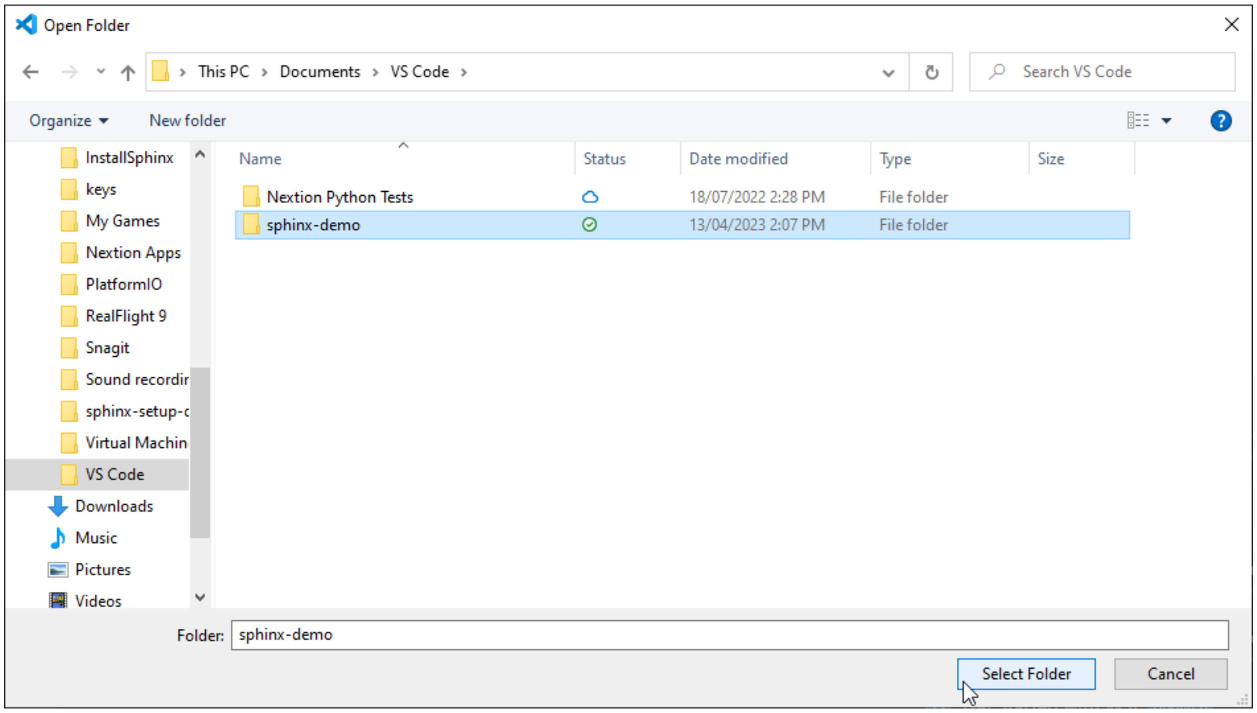


Navigate to a directory where you would like to store your documentation ‘projects’ and click on New Folder. I have chosen Documents/VS Code/ and created a new folder called *sphinx-demo*



| | Status | Date modified | Type | Size |
|---|---|---|---|---|
| Nextion Python Tests | | 18/07/2022 2:28 PM | File folder | |

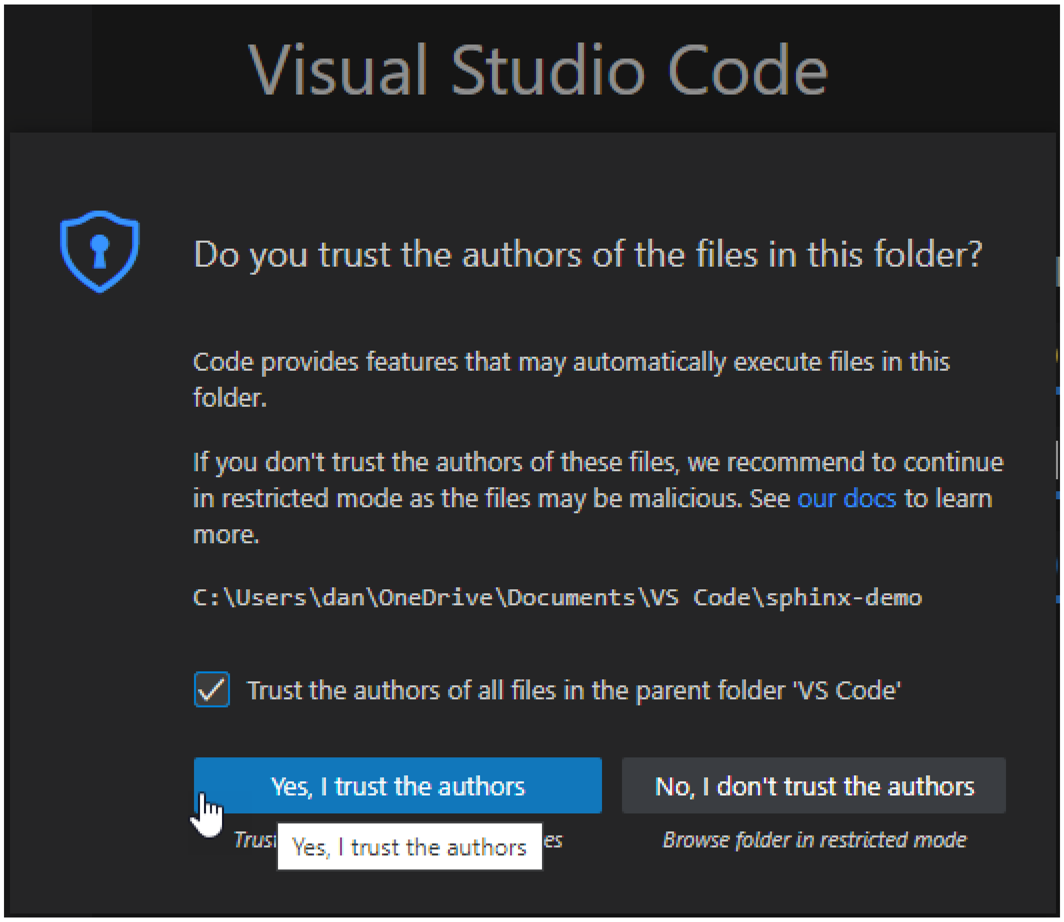
Folder: VS Code



| Name | Status | Date modified | Type | Size |
|---|---|---|---|---|
| Nextion Python Tests | | 18/07/2022 2:28 PM | File folder | |
| sphinx-demo | | 13/04/2023 2:07 PM | File folder | |

Folder: sphinx-demo

Click on the newly created folder and click “Select Folder”.


Visual Studio Code

Do you trust the authors of the files in this folder?

Code provides features that may automatically execute files in this folder.

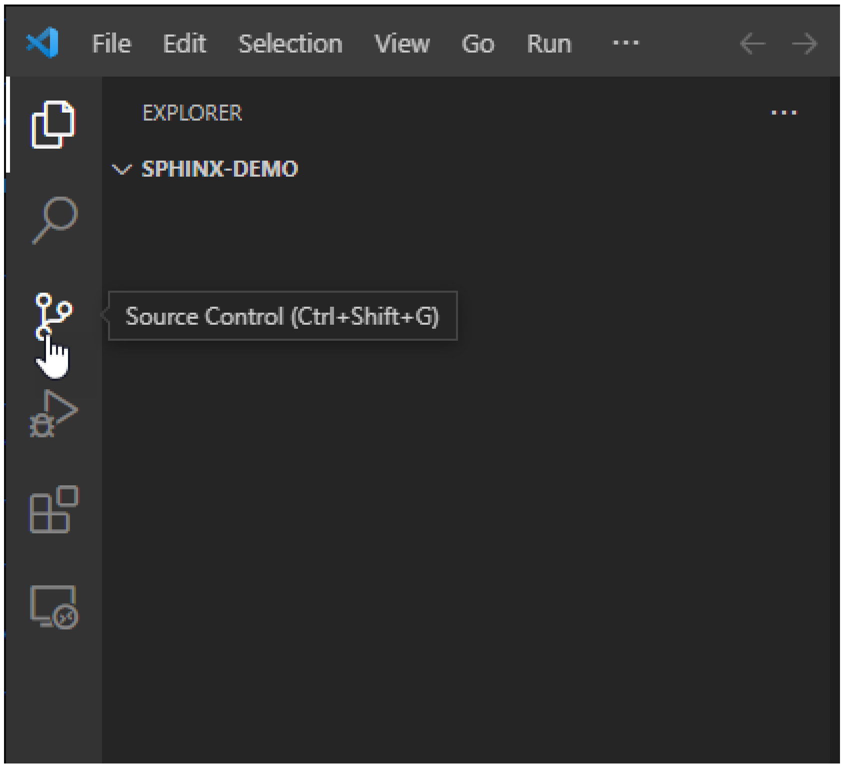
If you don't trust the authors of these files, we recommend to continue in restricted mode as the files may be malicious. See our docs to learn more.

```
C:\Users\dan\OneDrive\Documents\VS Code\sphinx-demo
```

✓ Trust the authors of all files in the parent folder 'VS Code'

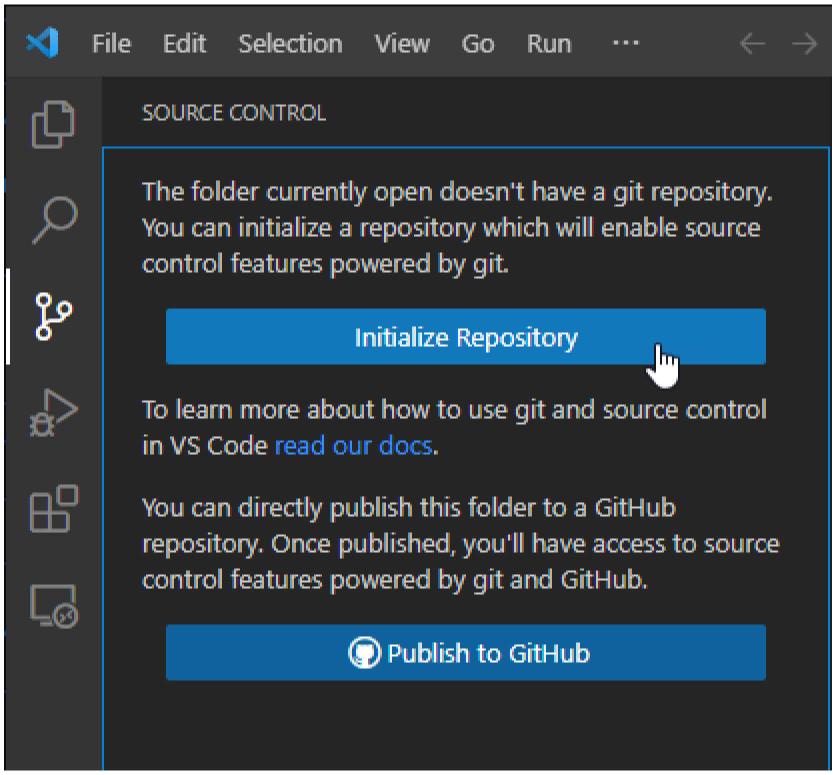
Yes, I trust the authors
No, I don't trust the authors
Trus
Yes, I trust the authors
es
Browse folder in restricted mode

You will be asked to confirm if you trust the authors of this folder. Since we are going to be the authors, you can click Yes.


File Edit Selection View Go Run ···
EXPLORER
SPHINX-DEMO
Source Control (Ctrl+Shift+G)

## 4.2 Set up git for this project

Next we need to intialise the repository for git. Hit *Ctrl* + *Shift* + *G* or click on the "Source Control" button on the left hand side.

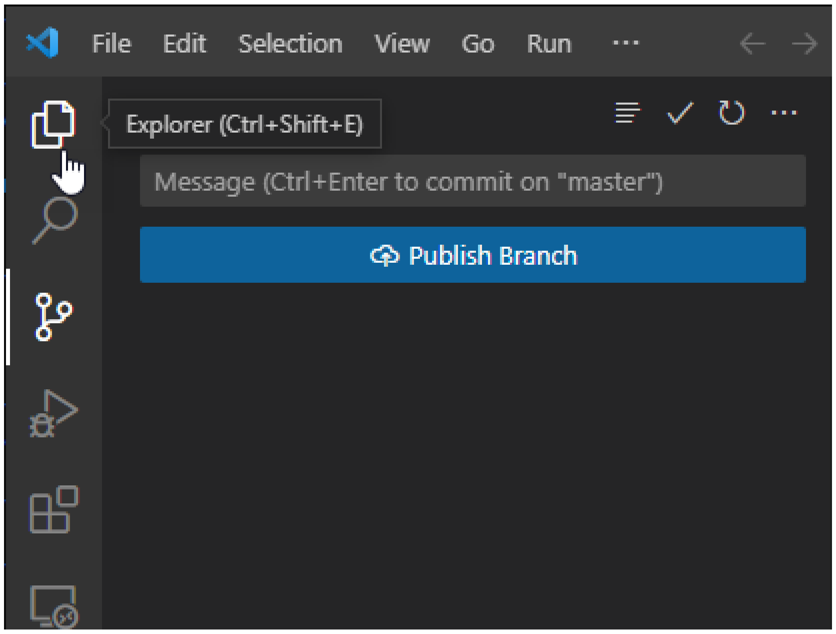


The folder currently open doesn't have a git repository. You can initialize a repository which will enable source control features powered by git.

To learn more about how to use git and source control in VS Code read our docs.

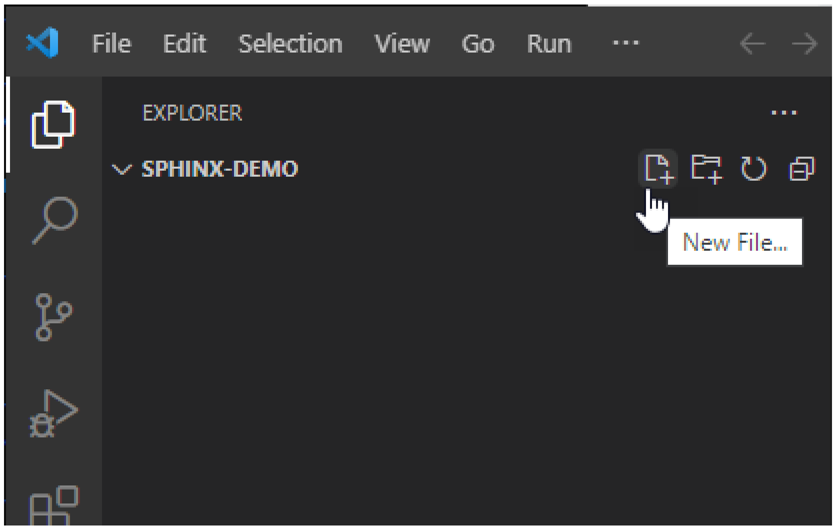
You can directly publish this folder to a GitHub repository. Once published, you'll have access to source control features powered by git and GitHub.

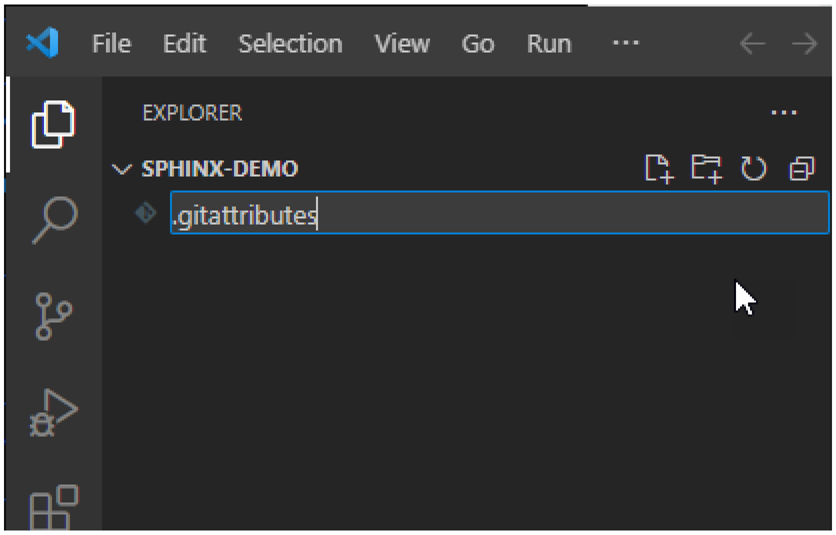
Then click on “Initialize Repository” and the git database will be created. It is in a directory called .git but it is hidden from view.



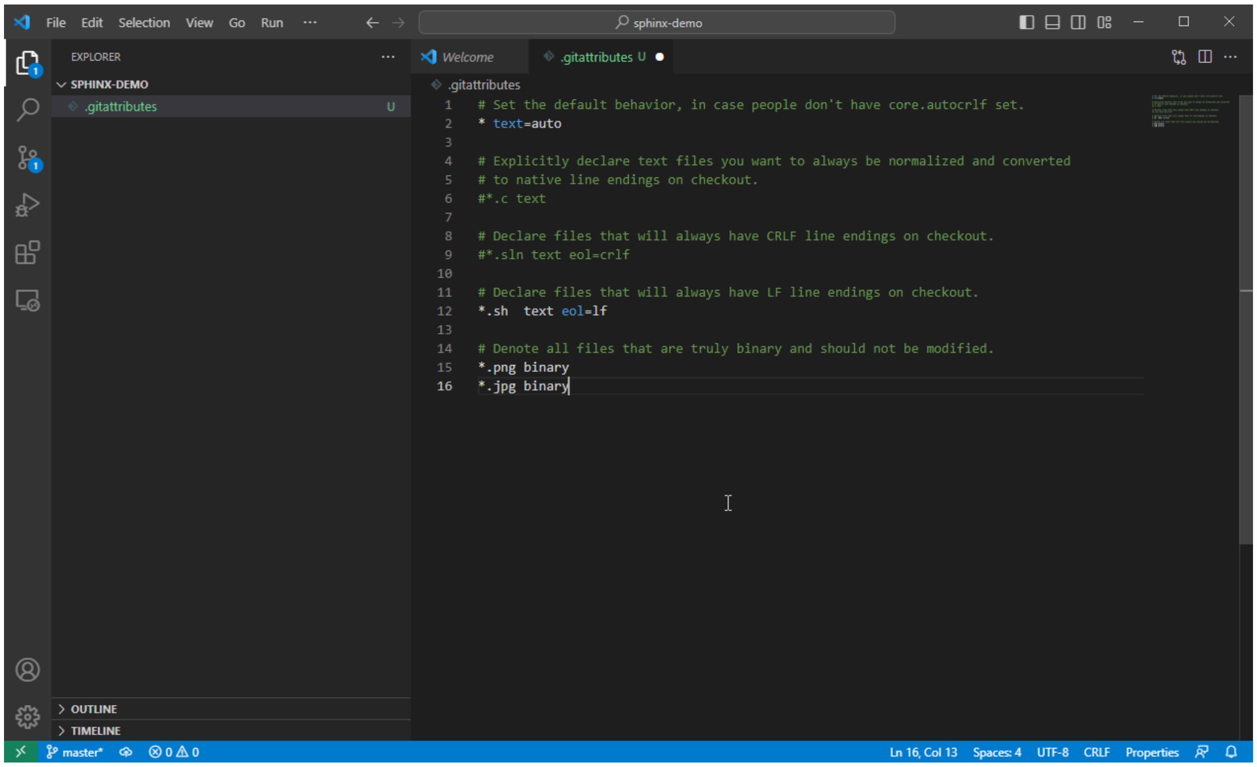
Then click on “Explorer” button to get back to the main source navigation window.

Lets start by creating two files. These are git files for controlling some of git’s functionality.





Click on the new file icon in the Explorer and then type the name of the file: `.gitattributes`



```
# Set the default behavior, in case people don't have core.autocrlf set.
* text=auto

# Explicitly declare text files you want to always be normalized and converted
# to native line endings on checkout.
#*.c text

# Declare files that will always have CRLF line endings on checkout.
#*.sln text eol=crlf

# Declare files that will always have LF line endings on checkout.
*.sh  text eol=lf

# Denote all files that are truly binary and should not be modified.
*.png binary
*.jpg binary
```

Next, paste the following text into the file:

```
# Set the default behavior, in case people don't have core.autocrlf set.
* text=auto

# Explicitly declare text files you want to always be normalized and converted
# to native line endings on checkout.
#*.c text

# Declare files that will always have CRLF line endings on checkout.
```

(continues on next page)

(continued from previous page)

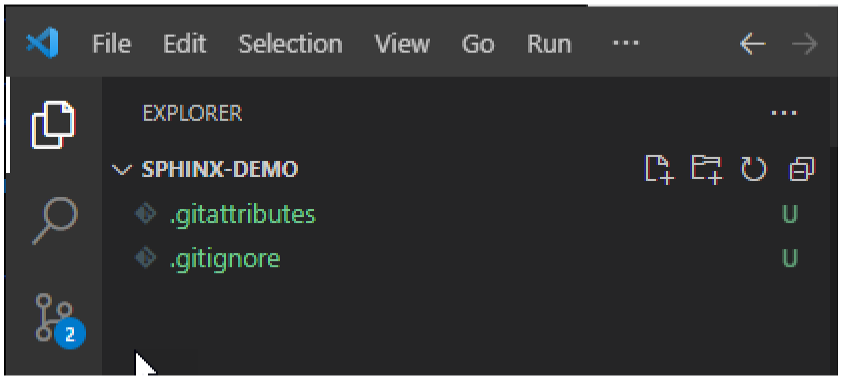
```
#*.sln text eol=crlf

# Declare files that will always have LF line endings on checkout.
*.sh  text eol=lf

# Denote all files that are truly binary and should not be modified.
*.png binary
*.jpg binary
```

This file just makes sure that .sh files are kept as unix LF encoding and .png/.jpg files are always treated as binary.

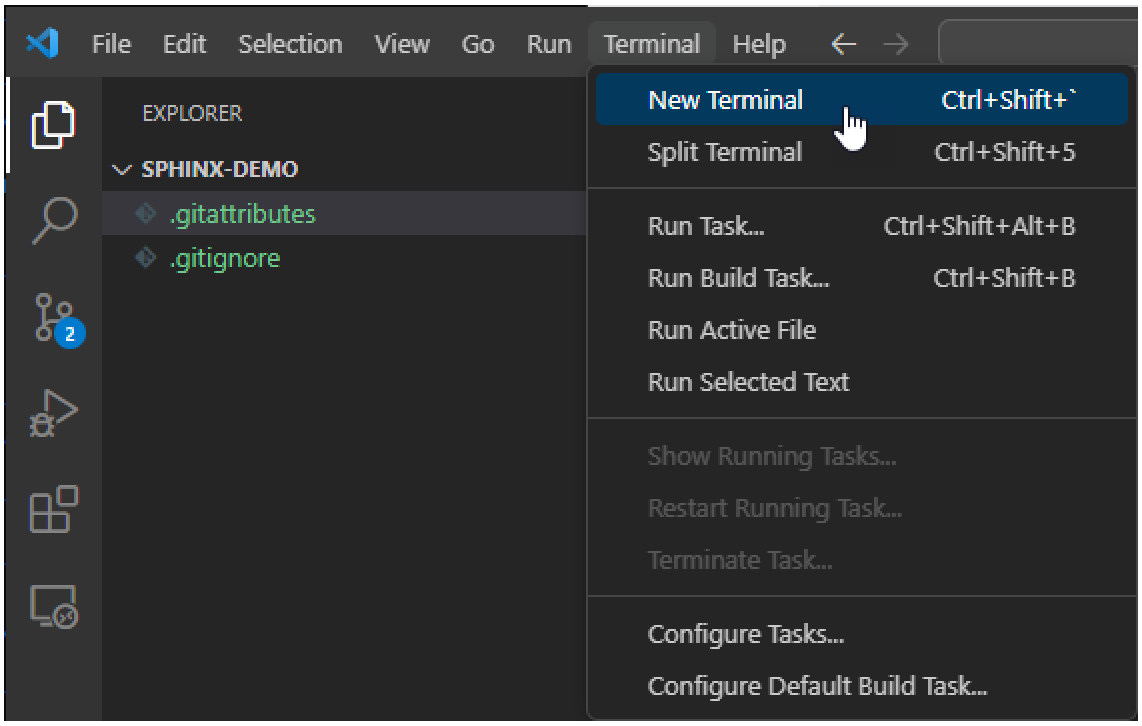
Repeat the process for `.gitignore` which will tell git which files to never consider part of the source code.

```
# VSCode Sphinx Ignore

# For now lets jsut store the sources.
/docs/build

# Put local python envs here.   Only relevant when using local python.
/.env*
```

`/docs/build` is the location of our build docs packages and `/.env*` is the place where we will store our python environments (no need to add these to the source code)

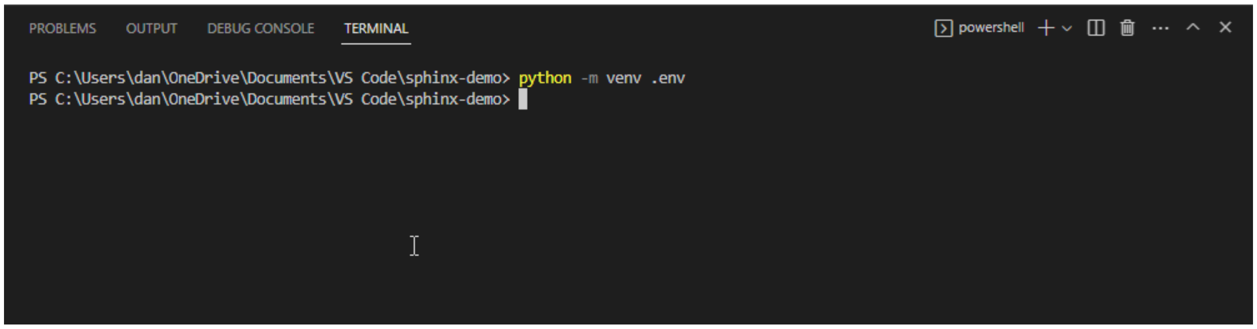


Notice how there is now a small '2' on the source control icon? This tells us that there are two files changed since the last 'commit'. The fact that the filenames are green also tell us they are new files (yellow would indicate a changed file). We will commit the changes later.


File Edit Selection View Go Run Terminal Help
EXPLORER
SPHINX-DEMO
.gitattributes
.gitignore
New Terminal Ctrl+Shift+`
Split Terminal Ctrl+Shift+5
Run Task... Ctrl+Shift+Alt+B
Run Build Task... Ctrl+Shift+B
Run Active File
Run Selected Text
Show Running Tasks...
Restart Running Task...
Terminate Task...
Configure Tasks...
Configure Default Build Task...

## 4.3 Set up python for this project

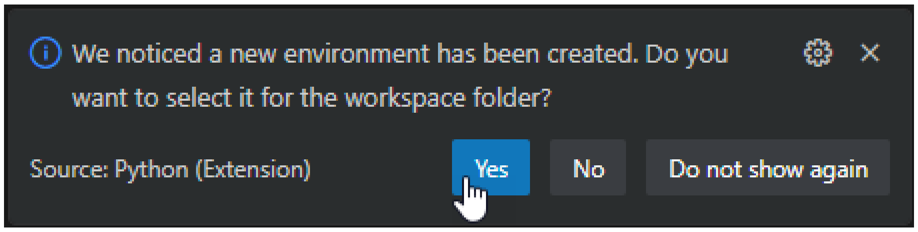
Next we will create our python environment. Open a new 'terminal' window by selecting *Terminal->New Terminal*. You should now see a powershell window in the bottom right of VSCode.



```
PS C:\Users\dan\OneDrive\Documents\VS Code\sphinx-demo> python -m venv .env
PS C:\Users\dan\OneDrive\Documents\VS Code\sphinx-demo> 
```

Type the following command: *python -m venv .env*

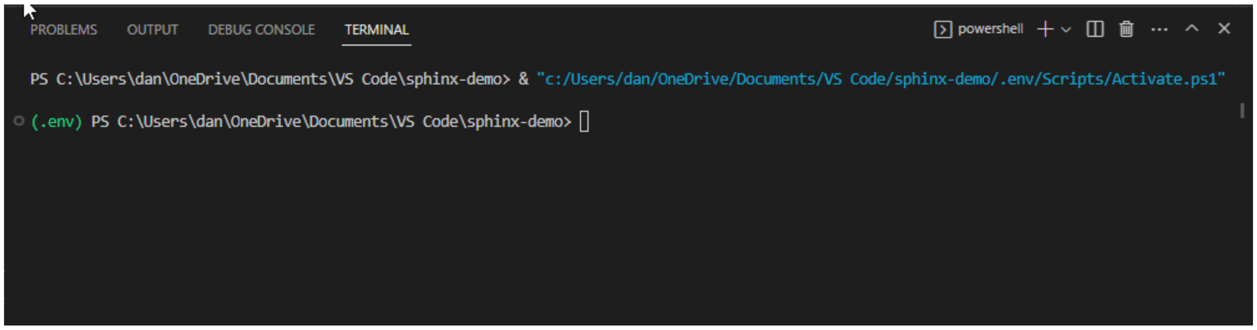
You should notice that the .env folder appears in your explorer. It should be greyed out since we told git to ignore anthing matching */.env**.



We noticed a new environment has been created. Do you want to select it for the workspace folder?

Source: Python (Extension)

Hopefully you will see the popup asking if you want to make it the default for the workspace folder. Click 'Yes'. Starting a new terminal from now on will execute the command to set the python virtual environment to .env. This can be seen by a green “(.env)” at the start of the line in the terminal.

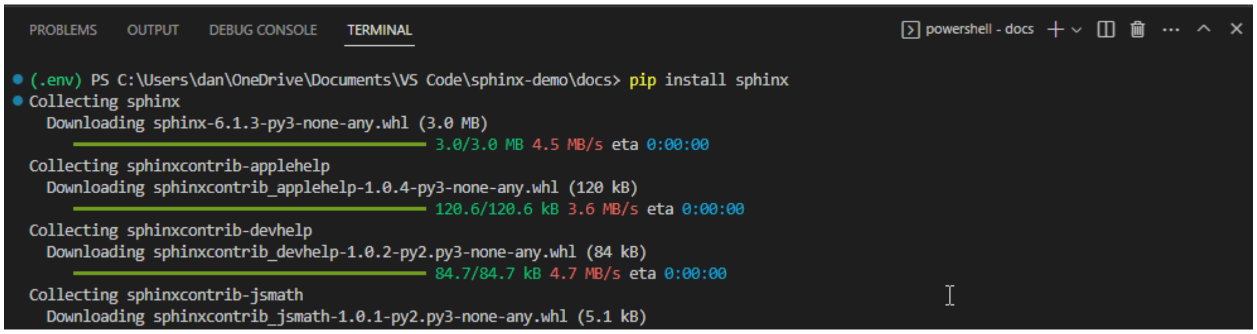


```
PS C:\Users\dan\OneDrive\Documents\VS Code\sphinx-demo> & "c:/Users/dan/OneDrive/Documents/VS Code/sphinx-demo/.env/Scripts/Activate.ps1"
(.env) PS C:\Users\dan\OneDrive\Documents\VS Code\sphinx-demo>
```

## 4.4 Install Sphinx

In a new terminal window, type the following to install Sphinx into our python environment. To keep things neat and tidy, lets put our document stuff in a subdir called docs. In the terminal window, execute the following commands:

```
mkdir docs
cd docs
pip install sphinx
```



```
(.env) PS C:\Users\dan\OneDrive\Documents\VS Code\sphinx-demo\docs> pip install sphinx
Collecting sphinx
  Downloading sphinx-6.1.3-py3-none-any.whl (3.0 MB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 3.0/3.0 MB 4.5 MB/s eta 0:00:00
Collecting sphinxcontrib-applehelp
  Downloading sphinxcontrib_applehelp-1.0.4-py3-none-any.whl (120 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 120.6/120.6 kB 3.6 MB/s eta 0:00:00
Collecting sphinxcontrib-devhelp
  Downloading sphinxcontrib_devhelp-1.0.2-py2.py3-none-any.whl (84 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 84.7/84.7 kB 4.7 MB/s eta 0:00:00
Collecting sphinxcontrib-jsmath
  Downloading sphinxcontrib_jsmath-1.0.1-py2.py3-none-any.whl (5.1 kB)
```

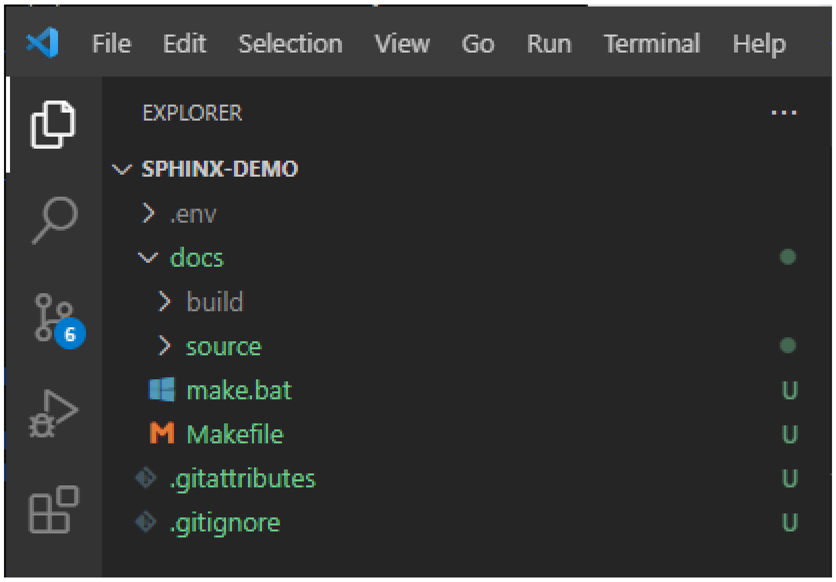
Pip will install the python package and it’s dependencies directly into your virtual python environment (.env). Now we can initialise a new sphinx document.

You can do that by issuing the command sphinx-quickstart in the same terminal window.

It will ask you a number of questions. Here are my suggestions

| Setting | Suggested Value | Notes |
|---|---|---|
| Separate source and build directories | Yes | It makes for a cleaner install. |
| Project name | Sphinx Demo | This text will appear in your docs |
| Author Name(s) | Your name | Can also be team name or dept. |
| Project releases | 1 | Whatever versioning you wish to use |
| Project language | en | Corporate default |

When the quickstart process is finished, it will create some files in the current directory (docs). Namely:



| Directory | Function |
|---|---|
| docs/build | The directory where our packages will be built. It should be greyed out since we elected to ignore it in gitignore. |
| docs/source | The directory containing the RST source files and python config file. |
| docs/make.bat and /docs/Makefile | These are scripts used for building the packages. |

## 4.5 Edit our first document

In the explorer window, navigate to docs/source/index.rst and click on it to open it in an editor tab. As a quick example, lets add some RST text and generate the html pages.

Just above the "Indices and tables" heading, paste the following text…

```
.. note::

    This is note text. If note text runs over a line, make sure the lines wrap
    and are indented to the same level as the note tag. If formatting is
    incorrect, part of the note might not render in the HTML output.

    Notes can have more than one paragraph. Successive paragraphs must indent
    to the same level as the rest of the note.

.. warning::

    Warnings are formatted in the same way as notes. In the same way, lines
    must be broken and indented under the warning tag.

.. error::

    This is an error!

We can also do math: Since Pythagoras, we know that :math:`a^2 + b^2 = c^2`.
```

(continues on next page)

(continued from previous page)

```
For lists, just place an asterisk at
the start of a paragraph and indent properly.  The same goes for numbered
lists; they can also be auto numbered using a ``#`` sign.

* This is a bulleted list.
* It has two items, the second
    item uses two lines.

#. This is a numbered list.
#. It has two items too.

1. This is a different numbered list.
2. It has two items too.

Nested lists are possible, but be aware that they must be separated from the
parent list items by blank lines.

* this is
* a list

    * with a nested list
    * and some subitems

* and here the parent list continues
```

Save the file and lets build the HTML package.

## 4.6 Build html docs

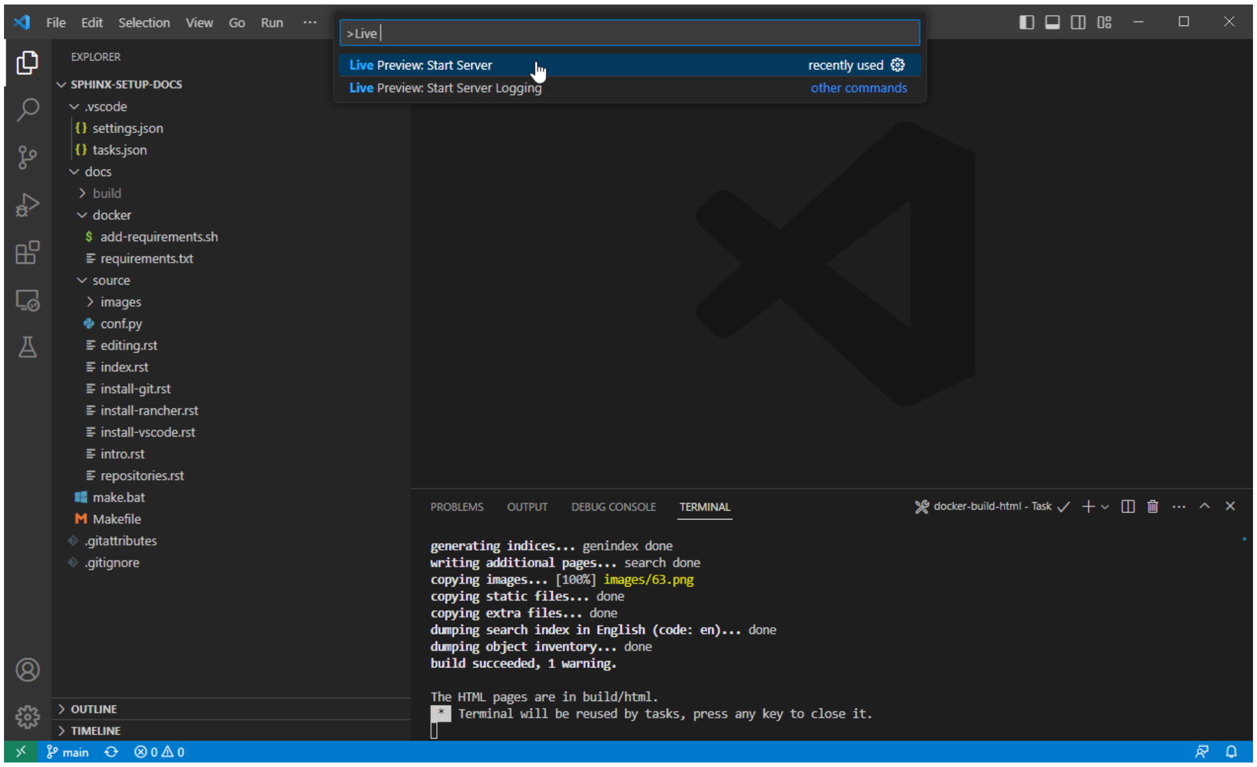
In the terminal window, type the following commands. (Make sure you are still in the docs directory)

```
.\make clean
.\make html
```

The first command makes sure the old html files are removed, and then the second command builds it again. If all went well, you should see under the docs/build directory, a new directory called ‘html’ (and another called doctrees - you can ignore that)

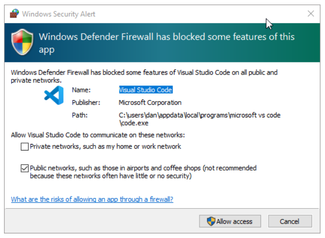
## 4.7 Live Server

If you installed the “Live Preview” plugin, you can start a basic local html server to view your changes whenever you rebuild the documentation in html. This is really handy for seeing your work progress.



```
generating indices... genindex done
writing additional pages... search done
copying images... [100%] images/63.png
copying static files... done
copying extra files... done
dumping search index in English (code: en)... done
dumping object inventory... done
build succeeded, 1 warning.

The HTML pages are in build/html.
 *  Terminal will be reused by tasks, press any key to close it.
```

In the search bar (Ctrl-Shift-P) search for “> Live Preview” and click on “Live Preview: Start Server”. You might get a windows firewall warning to allow the traffic.



Windows Security Alert

Windows Defender Firewall has blocked some features of this app

Windows Defender Firewall has blocked some features of Visual Studio Code on all public and private networks.

| | |
|---|---|
| Name: | Visual Studio Code |
| Publisher: | Microsoft Corporation |
| Path: | C:\users\dan\appdata\local\programs\microsoft vs code \code.exe |

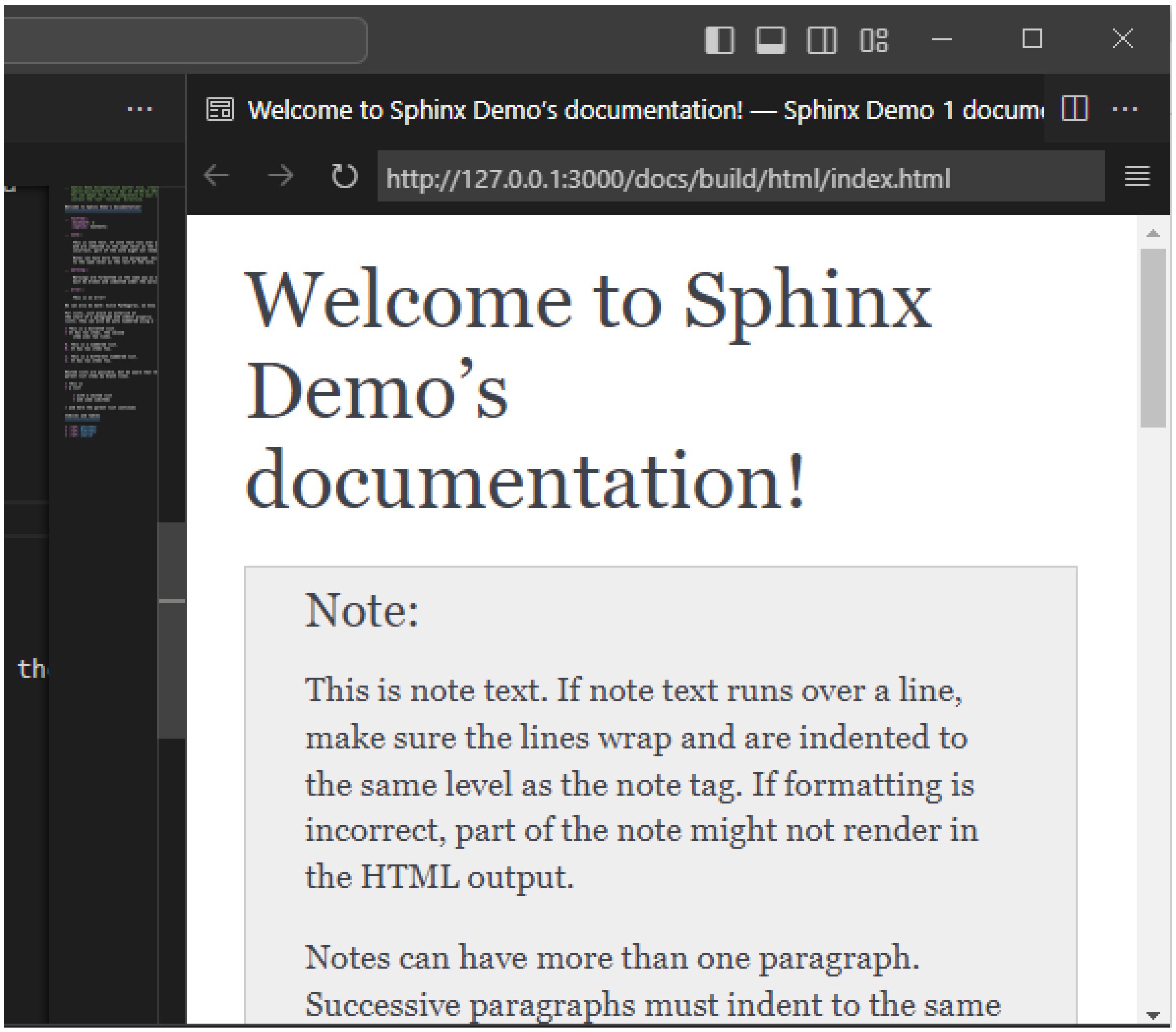
Allow Visual Studio Code to communicate on these networks:

☐ Private networks, such as my home or work network

☑ Public networks, such as those in airports and coffee shops (not recommended because these networks often have little or no security)

What are the risks of allowing an app through a firewall?

It should open a tab to http://127.0.0.1:3000/ but I find it easier opening that in a new web browser. It will refresh the page with every change to the code. If you navigate to http://127.0.0.1/docs/build/html/ you should see your documentation.


Welcome to Sphinx Demo's documentation! — Sphinx Demo 1 docum

http://127.0.0.1:3000/docs/build/html/index.html

Welcome to Sphinx Demo's documentation!

Note:

This is note text. If note text runs over a line, make sure the lines wrap and are indented to the same level as the note tag. If formatting is incorrect, part of the note might not render in the HTML output.

Notes can have more than one paragraph. Successive paragraphs must indent to the same

The page looks nice, but there's a few things we should change. Firstly, remove the "Indices and tables" section at the bottom. It does not really do much for us. Secondly, lets install a nicer html theme (the 'read the docs' theme).

Open the conf.py file in the docs directory. This file is used to define how sphinx itself is configured.

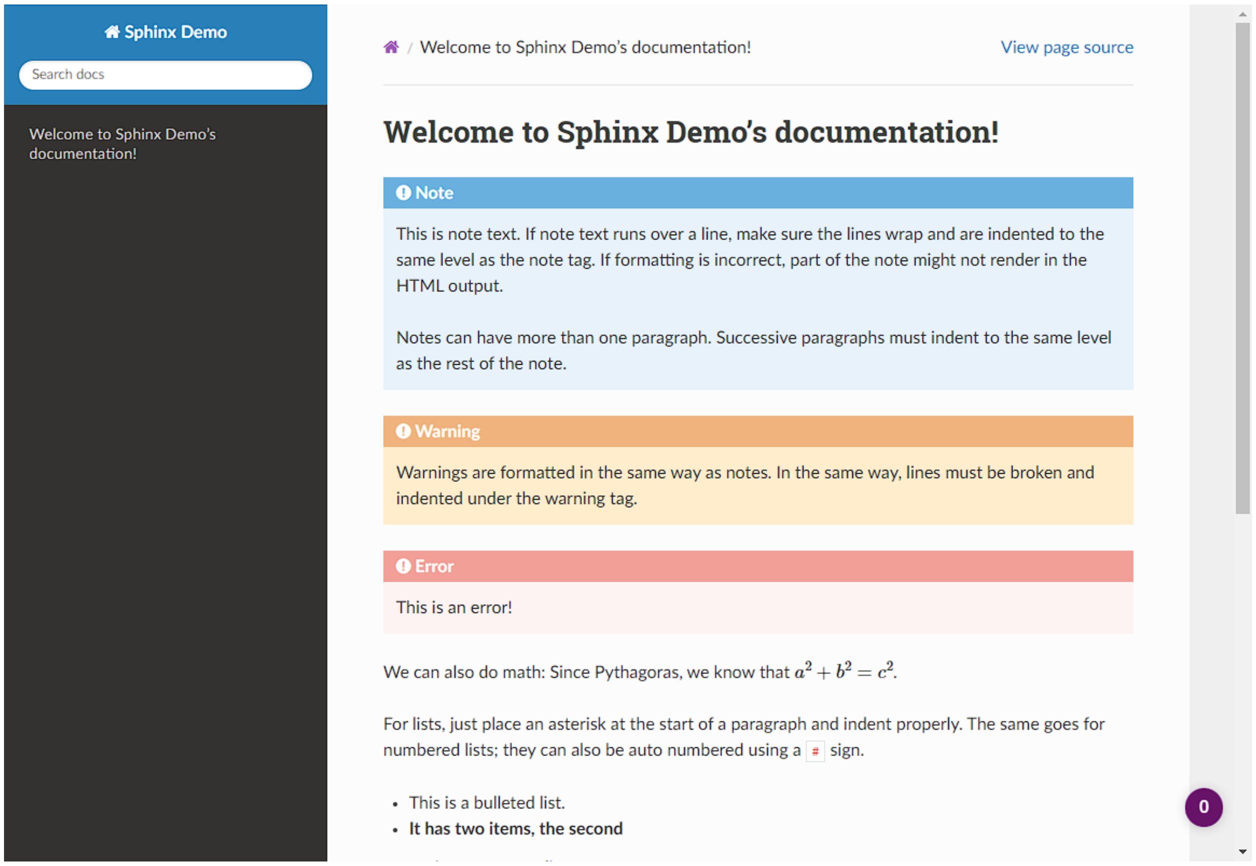
Change the html_theme setting from alabaster to "sphinx_rtd_theme". And above that line, add another line to import the python module for the RTD theme. It should look like this:

```
import sphinx_rtd_theme
html_theme = 'sphinx_rtd_theme'
```

But before we can build that, we have to install the sphinx_rtd_theme module into our python environment. So in a terminal window, type the following.

```
pip install sphinx_rtd_theme
```

Now, run the *make clean ; make html* commands again to rebuild the docs. If you go back to your browser and refresh you should see a big change.


Sphinx Demo
Search docs
Welcome to Sphinx Demo's documentation!
Welcome to Sphinx Demo's documentation!
View page source
0

Welcome to Sphinx Demo's documentation!

Note

This is note text. If note text runs over a line, make sure the lines wrap and are indented to the same level as the note tag. If formatting is incorrect, part of the note might not render in the HTML output.

Notes can have more than one paragraph. Successive paragraphs must indent to the same level as the rest of the note.

Warning

Warnings are formatted in the same way as notes. In the same way, lines must be broken and indented under the warning tag.

Error

This is an error!

We can also do math: Since Pythagoras, we know that $a^2 + b^2 = c^2$.

For lists, just place an asterisk at the start of a paragraph and indent properly. The same goes for numbered lists; they can also be auto numbered using a `#` sign.

- This is a bulleted list.
- **It has two items, the second**

WOW, that looks a whole lot nicer!

## 4.8 Tasks

Next, to make things a little easier. Lets make some tasks for building our html pages.

Hit Ctrl+Shift+P to bring up the VSCode command dialog. It's a way for searching for a command when you don't know how to find it in the menu (or it does not have a key binding). Type 'tasks', choose "Tasks: Configure Task" and select "Create tasks.json from template". Choose 'Others' as the task template.

It will open an editor tab with a blank tasks.json file in it. It looks like this:

```
{
    // See https://go.microsoft.com/fwlink/?LinkId=733558
    // for the documentation about the tasks.json format
    "version": "2.0.0",
    "tasks": [
        {
            "label": "echo",
            "type": "shell",
            "command": "echo Hello"
        }
    ]
}
```

Change that for the following JSON code. Basically, we're deleting the 'echo' task and adding two more, setting one of them as the default build task.

```json
{
    // See https://go.microsoft.com/fwlink/?LinkId=733558
    // for the documentation about the tasks.json format
    "version": "2.0.0",
    "tasks": [
        {
            "label": "local-build-html",
            "type": "shell",
            "dependsOn": ["clean"],
            "command": "source ../.env/bin/activate; make html",
            "options": {
                "cwd": "${workspaceFolder}/docs"
            },
            "windows": {
                "command": "..\\.env\\Scripts\\Activate.ps1; .\\make.bat html",
                "options": {
                    "cwd": "${workspaceFolder}/docs"
                },
            },
            "group": {
                "kind": "build",
                "isDefault": true
            },
            "problemMatcher": []
        },
        {
            "label": "clean",
            "type": "shell",
            "command": "source ../.env/bin/activate; make clean",
            "options": {
                "cwd": "${workspaceFolder}/docs"
            },
            "windows": {
                "command": "..\\.env\\Scripts\\Activate.ps1; .\\make.bat clean",
                "options": {
                    "cwd": "${workspaceFolder}/docs"
                },
            },
            "group": {
                "kind": "build",
                "isDefault": false
            },
            "problemMatcher": []
        }
    ]
}
```

In the code above you can see two tasks defined. 'local-build-html' and 'clean'. These just do what we were doing manually with the `make clean` and `make html` commands. The only difference is we have to tell VS Code where to work from (cwd), and we have to set the python environment first (Activate.ps1). There are also two commands defined in each task, one for windows (`..\\.env\\Scripts\\Activate.ps1; .\\make.bat <something>`). This overrides the other command, which is the unix corresponding unix command (`source ../.env/bin/activate; make <something>`).

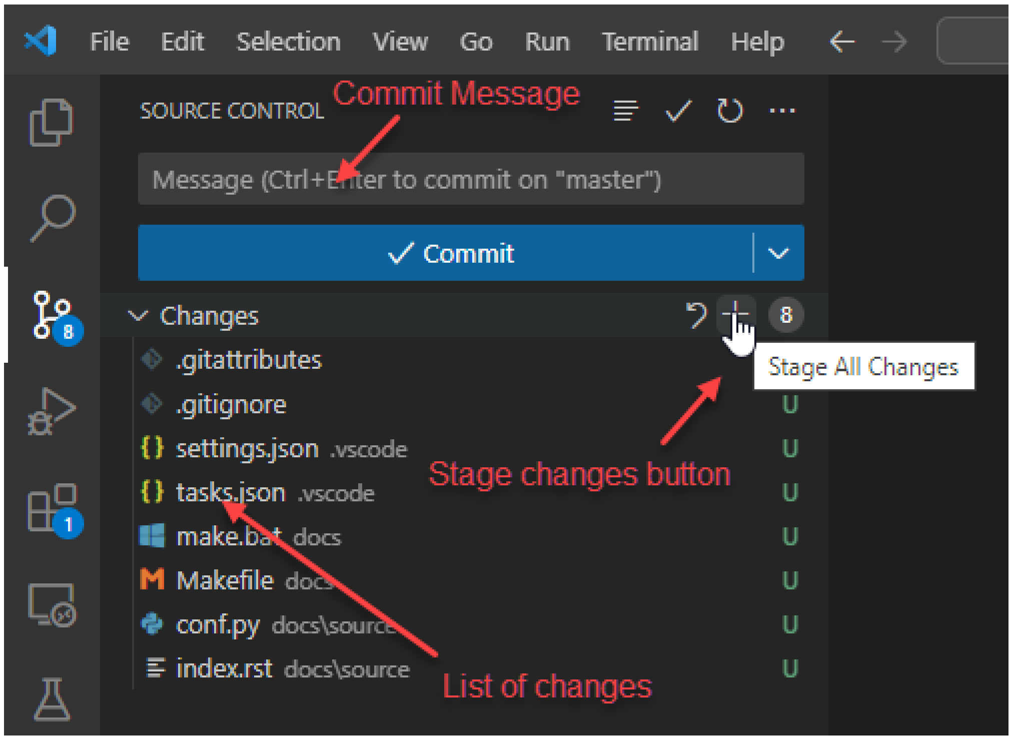
The ‘group’ section of the JSON data just tells VS Code that these are build tasks, and one of them is the default build task. Finally, in the ‘local-build-html’ task, we have a ‘dependsOn’ setting. This makes sure that the ‘clean’ task gets executed always before the ‘local-build-html’ task.

Once you have these saved to tasks.json you should be able to hit Ctrl+Shift+B and the html files will be re-build automatically.

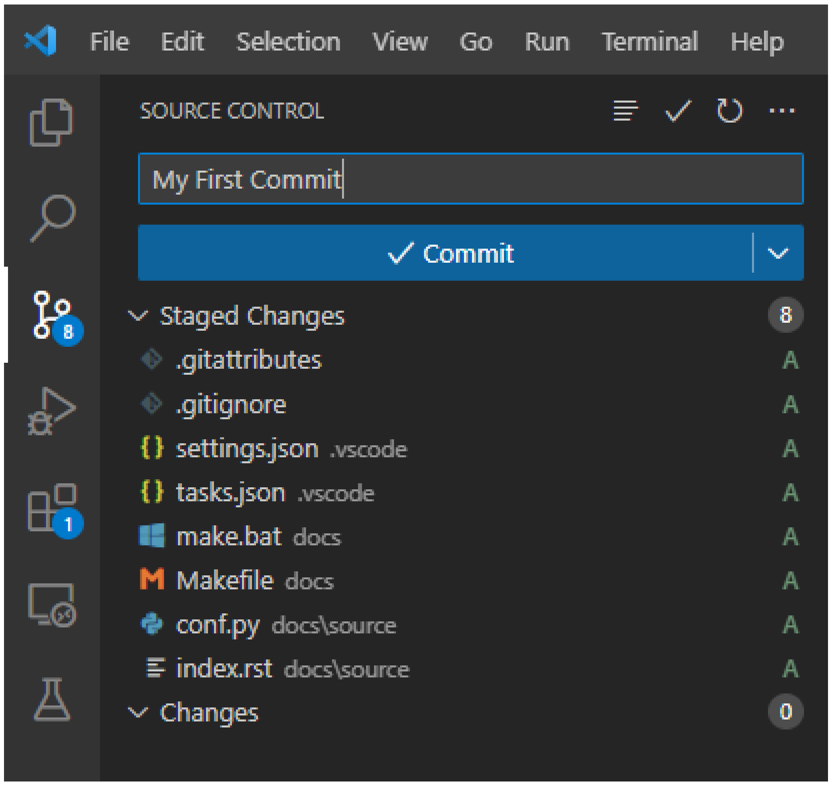
## 4.9 Saving Changes - the git way

So saving the files happens normally of course but when we have git installed we can also save progressive versions. Git is a little complicated but I’ll go over the basics. Your code is stored, not just in the working directory, but also in the git database under the .git directory. For each change to one or more files you can save extra info. Firstly, a small commit message (required) to mention what your changes are, but you can also add tags to a git commit so that it’s clear what stage the full package is… For instance, you might commit the current changes with “My first commit.” and then add a tag that gives the whole package a version number. Tags are optional so we’ll ignore them for now.

Click on the ‘Source Control’ button and we will make our first commit.



In this image you can see there is a text box for the commit message. And below that the list of changes. Now, we need to tell git which changes we wish to commit to the git repo. Clicking on the + on the changes row will let us add them all.

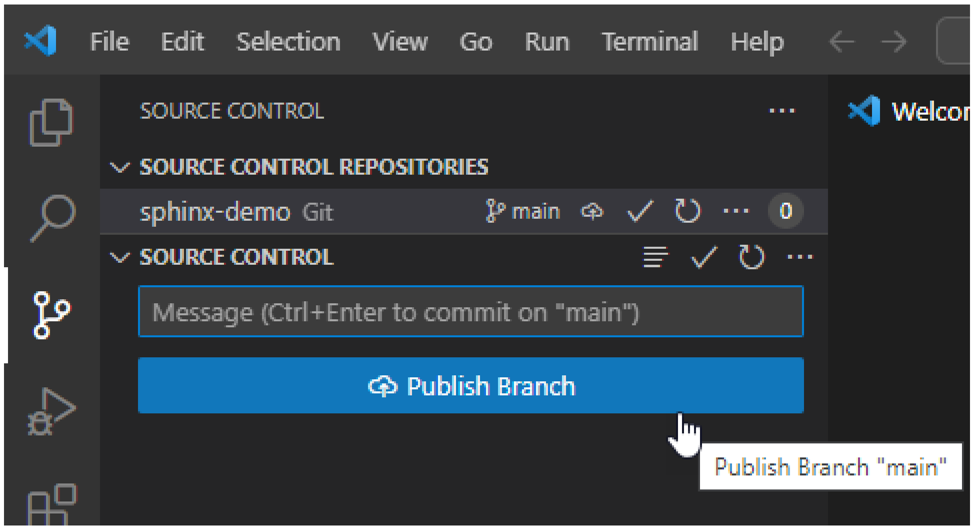


Now we can see all of our changes are staged under the 'Staged Changes' branch and nothing is left in the 'Changes' branch. Enter a commit message and click 'Commit'. You should see all the changes disappear and the change counter on the left 'Source Control' button disappear as well.

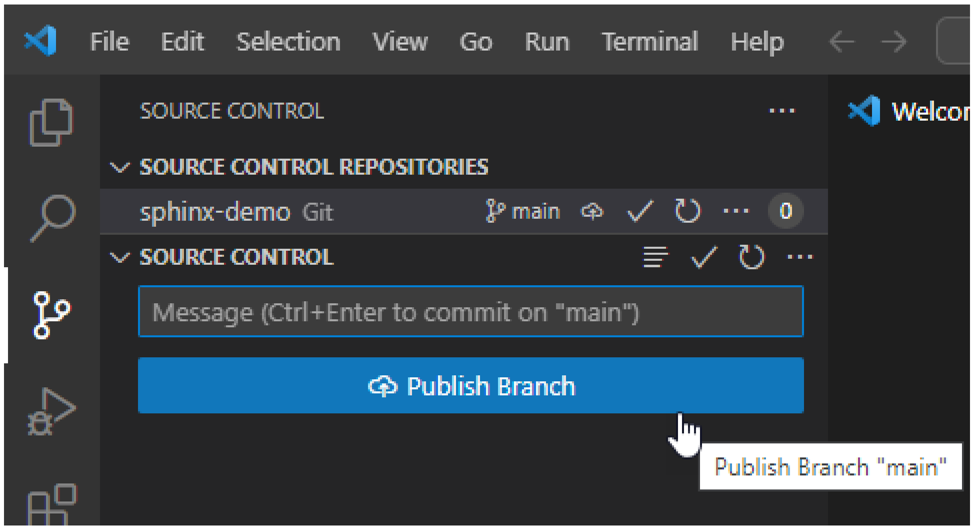
When you navigate back to the Explorer, you will see in your source tree all of the green files are now white.

## 4.10 Saving to github

So we now have our local repository, but the files are all still on our local machine. One of the great things about git is that it's designed to work with remote (shared even) repositories. So let's save our git repo to github. If you don't already have a github account, go and create one.


File Edit Selection View Go Run Terminal Help
SOURCE CONTROL
SOURCE CONTROL REPOSITORIES
sphinx-demo Git main 0
SOURCE CONTROL
Message (Ctrl+Enter to commit on "main")
Publish Branch
Publish Branch "main"
Welcor

Go back to the 'Source Control' section and you should see that the commit button now says "Publish Branch". Click on that and you will be asked to 'allow' the inbuilt github extension to authnticate to github. Click 'Allow'. It will open a web browser when you log in and complete the authentication, an oauth token will be created and stored in the windows credentials store.



You should now see two options… One for creating a public and one for a private repo. Choose one. VS Code will now attempt to create the remote repo on github.com. If it already exisst it will inform you and will not push the code. Otherwise VS code will link your local repo to the newly created remote and publish (push) the code to the new repo.

**Note:** You might be asked to authenticate to github again, this time it's to trust the 'git credentials manager' or the 'git ecosystem'. This is to allow the git command line tools to also interact with your github account.

What to do if the repo exists, and it is the one you want to use? Well normally we would have cloned that repo on our local machine instead of creating a repo locally and pushing it to github, so that situation should not really happen.

In the next section, I describe a way that you can use github infrastructure to build the html and pdf changes, and just save them to a different branch of our same repo.

## 5 Method 2 - Rancher

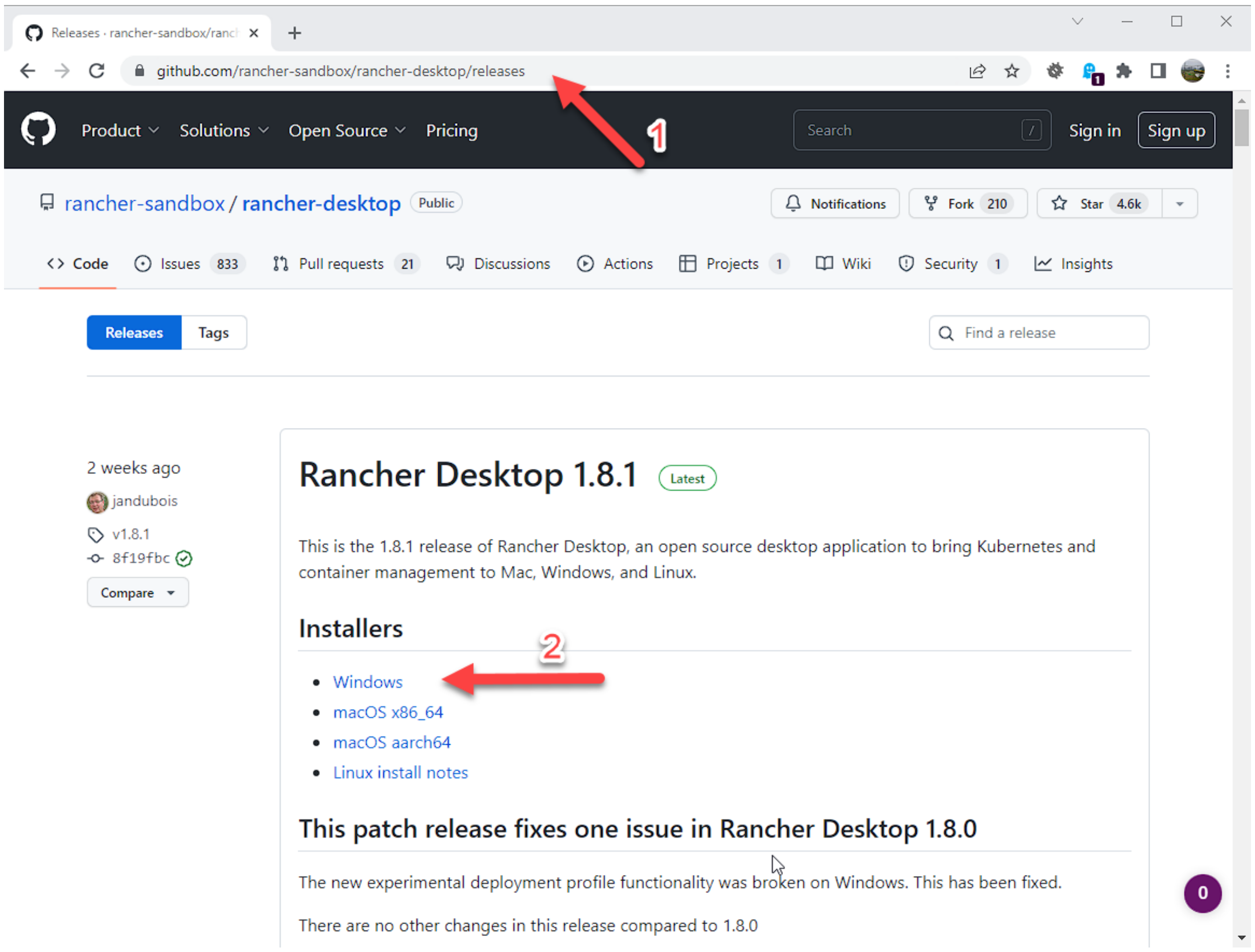
As mentioned earlier, Rancher is the software the controls the docker engine on your desktop. With this method we will run the sphinx pdf build scripts inside docker.

**Warning:** This method is completely optional. If you plan on putting your sources into a github repository then you can have github build the documentation for you, each time you push the sources to github.

What this method does allow for is if you want to build completely separate from github or if you want to build your pdf relatively quickly (it takes a couple of minutes via github workflow)

If you do decide not to install rancher, then the only real downside is that you will not be able to build the PDF images locally on your machine. If that's OK with you, go ahead and skip this entire section.

## 5.1 Download and install Rancher



Rancher Desktop 1.8.1

This is the 1.8.1 release of Rancher Desktop, an open source desktop application to bring Kubernetes and container management to Mac, Windows, and Linux.

Installers

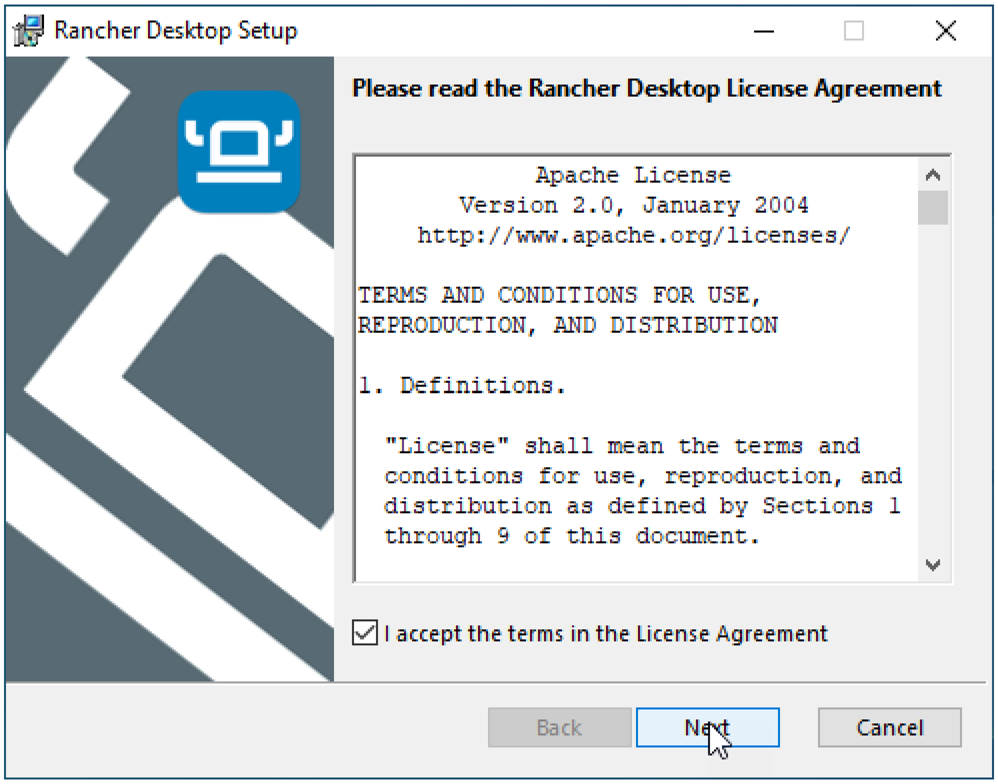
This patch release fixes one issue in Rancher Desktop 1.8.0

The new experimental deployment profile functionality was broken on Windows. This has been fixed.

There are no other changes in this release compared to 1.8.0

1. In your favorite web browser navigate to this page: `https://github.com/rancher-sandbox/rancher-desktop/releases`
2. Click on the "Windows" link to download the latest windows version.

## 5.2 Continue install

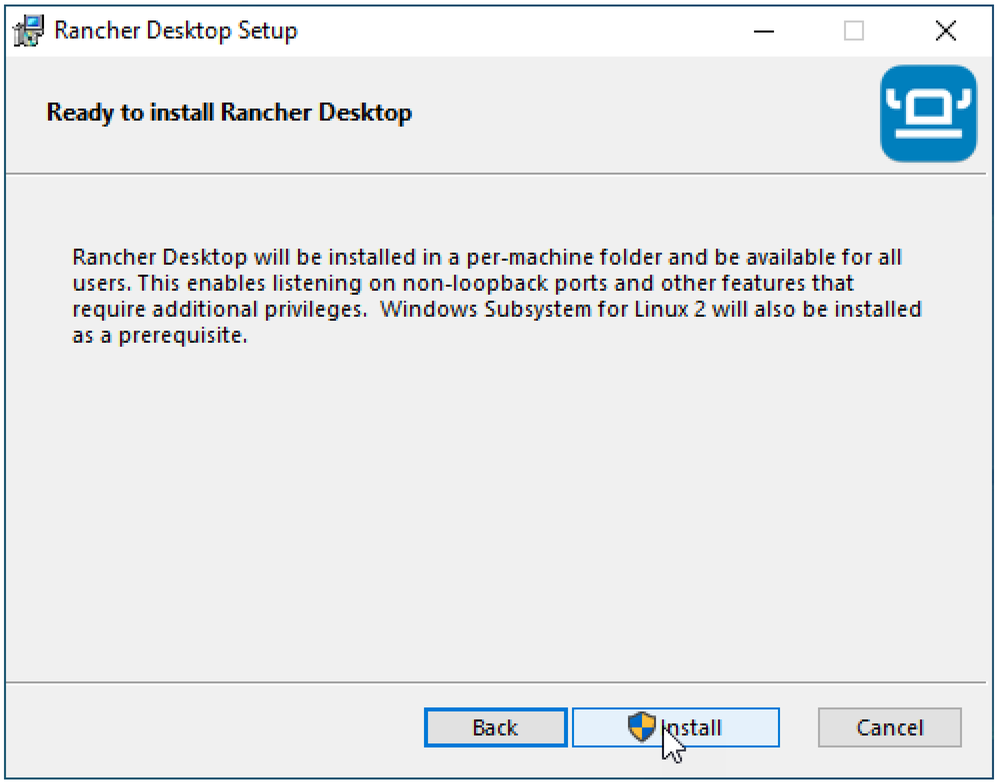


**Please read the Rancher Desktop License Agreement**

```
                Apache License
          Version 2.0, January 2004
       http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE,
REPRODUCTION, AND DISTRIBUTION

1. Definitions.

  "License" shall mean the terms and
  conditions for use, reproduction, and
  distribution as defined by Sections 1
  through 9 of this document.
```

☑ I accept the terms in the License Agreement

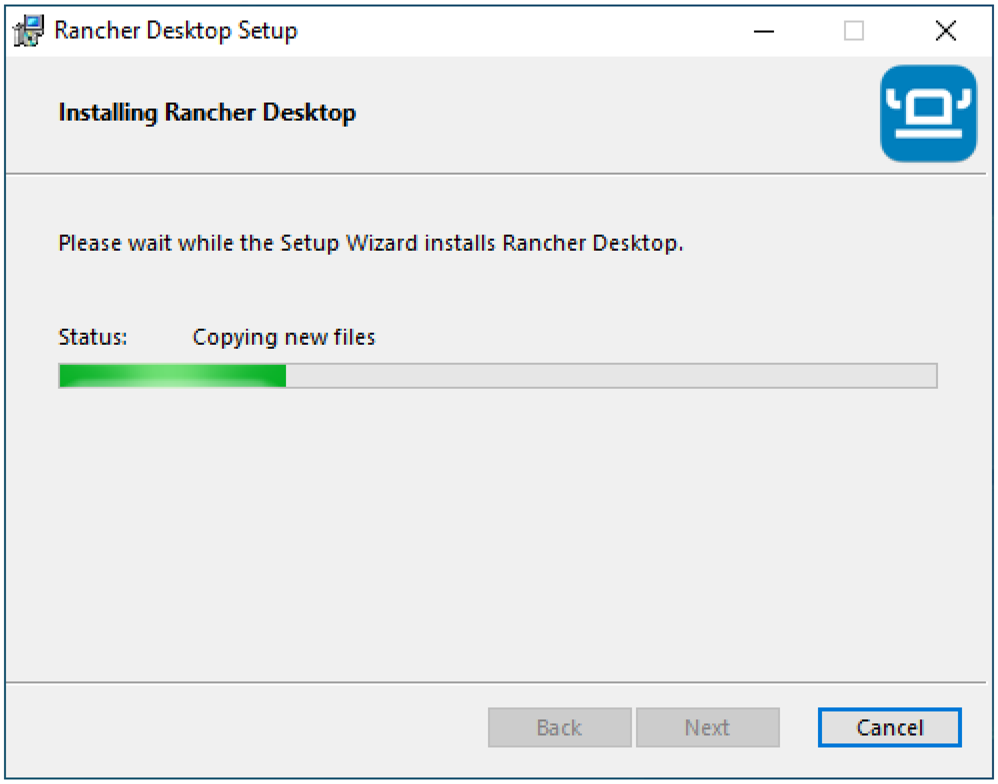
Click Next.



**Ready to install Rancher Desktop**

Rancher Desktop will be installed in a per-machine folder and be available for all users. This enables listening on non-loopback ports and other features that require additional privileges. Windows Subsystem for Linux 2 will also be installed as a prerequisite.

Click Install.

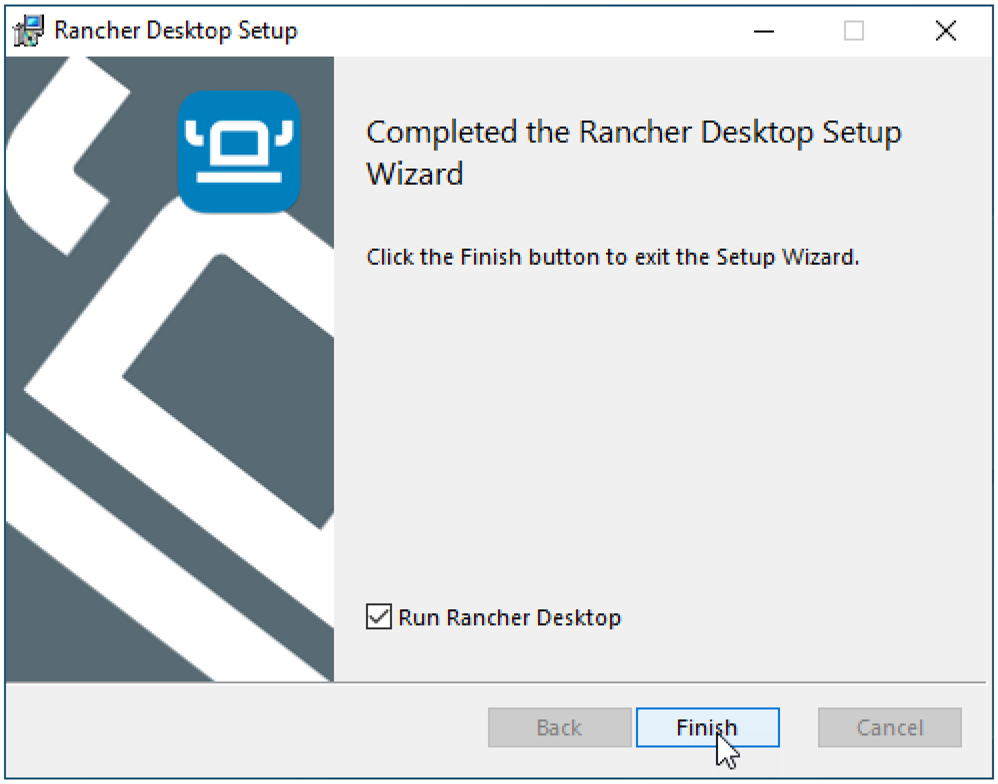


**Installing Rancher Desktop**

Please wait while the Setup Wizard installs Rancher Desktop.

Status: Copying new files

*Wait*



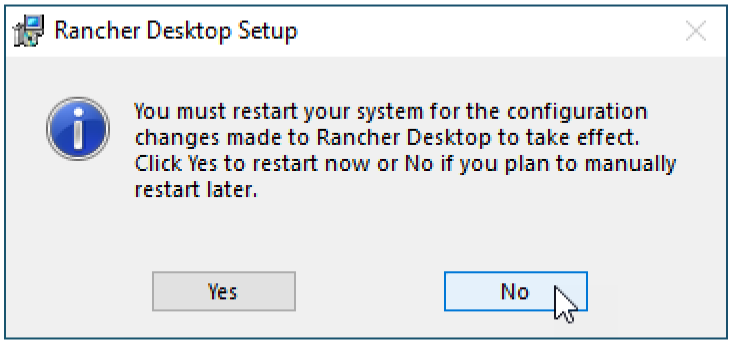
Completed the Rancher Desktop Setup Wizard

Click the Finish button to exit the Setup Wizard.

☑ Run Rancher Desktop

Click Finish.

## 5.3 Don't restart - yet

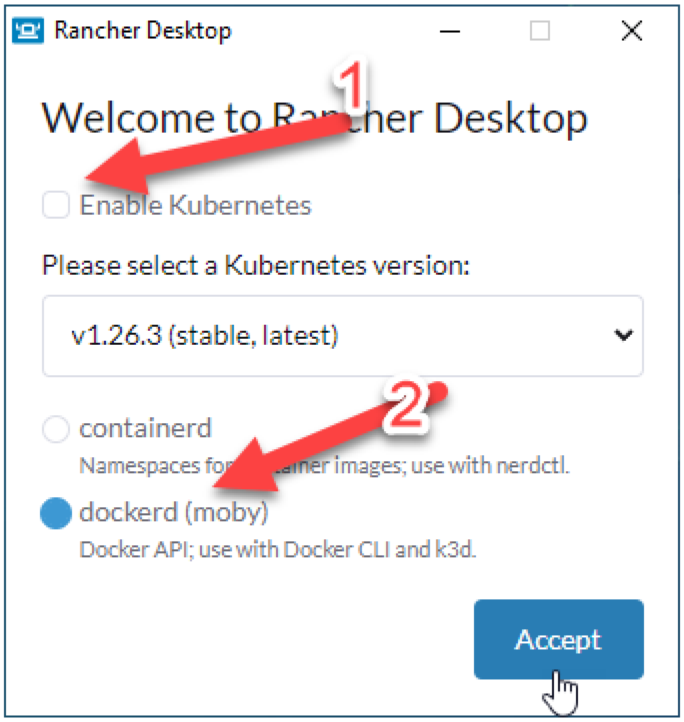

Rancher Desktop Setup

You must restart your system for the configuration changes made to Rancher Desktop to take effect. Click Yes to restart now or No if you plan to manually restart later.

Click No.

**Warning:** Clicking no here allows you to choose the initial setup for rancher (namely chosing the docker engine and not enabling Kubernetes). It does produce an **error** however it is fixed after a reboot.

## 5.4 Configure the engine


Rancher Desktop

Welcome to Rancher Desktop

Enable Kubernetes

Please select a Kubernetes version: v1.26.3 (stable, latest)

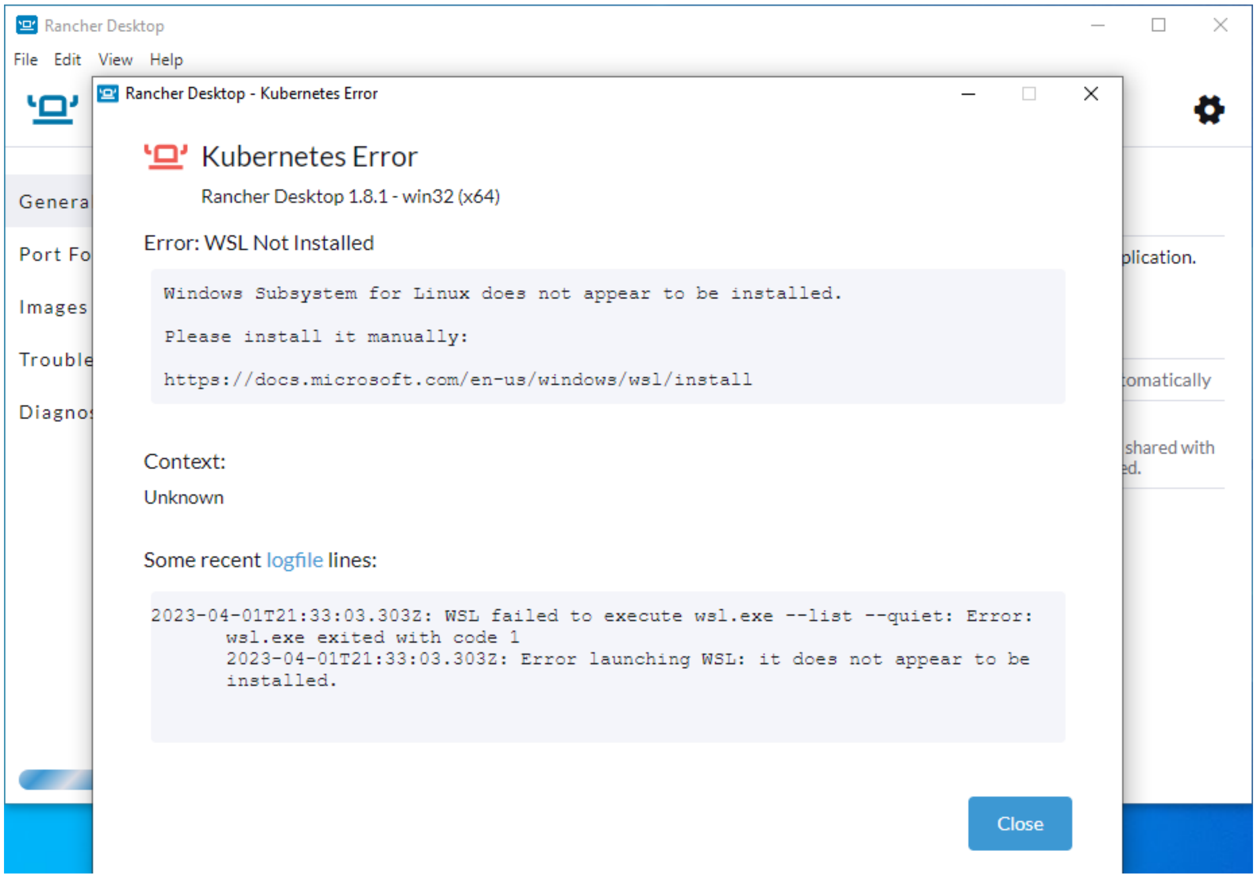
containerd — Namespaces for container images; use with nerdctl.

dockerd (moby) — Docker API; use with Docker CLI and k3d.

1. Disable (uncheck) Kibernetes.
2. Select “dockerd” as the container software.

Click Accept.

## 5.5 Close and restart



Rancher Desktop - Kubernetes Error

Kubernetes Error

Rancher Desktop 1.8.1 - win32 (x64)

Error: WSL Not Installed

```
Windows Subsystem for Linux does not appear to be installed.

Please install it manually:

https://docs.microsoft.com/en-us/windows/wsl/install
```

Context:

Unknown

Some recent logfile lines:

```
2023-04-01T21:33:03.303Z: WSL failed to execute wsl.exe --list --quiet: Error:
      wsl.exe exited with code 1
      2023-04-01T21:33:03.303Z: Error launching WSL: it does not appear to be
      installed.
```

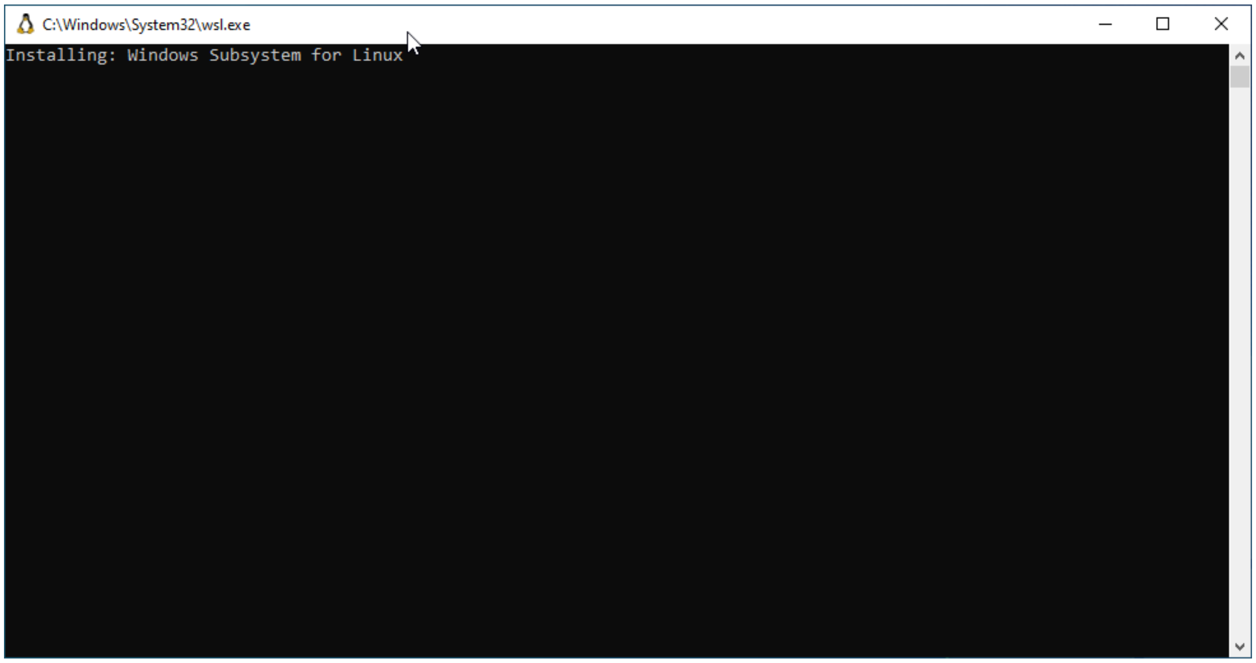
Click Close and restart.

---

**Note:** After some time rancher will throw an error - this is because WLS2 and the docker image are not yet installed. Reboot your PC.

---

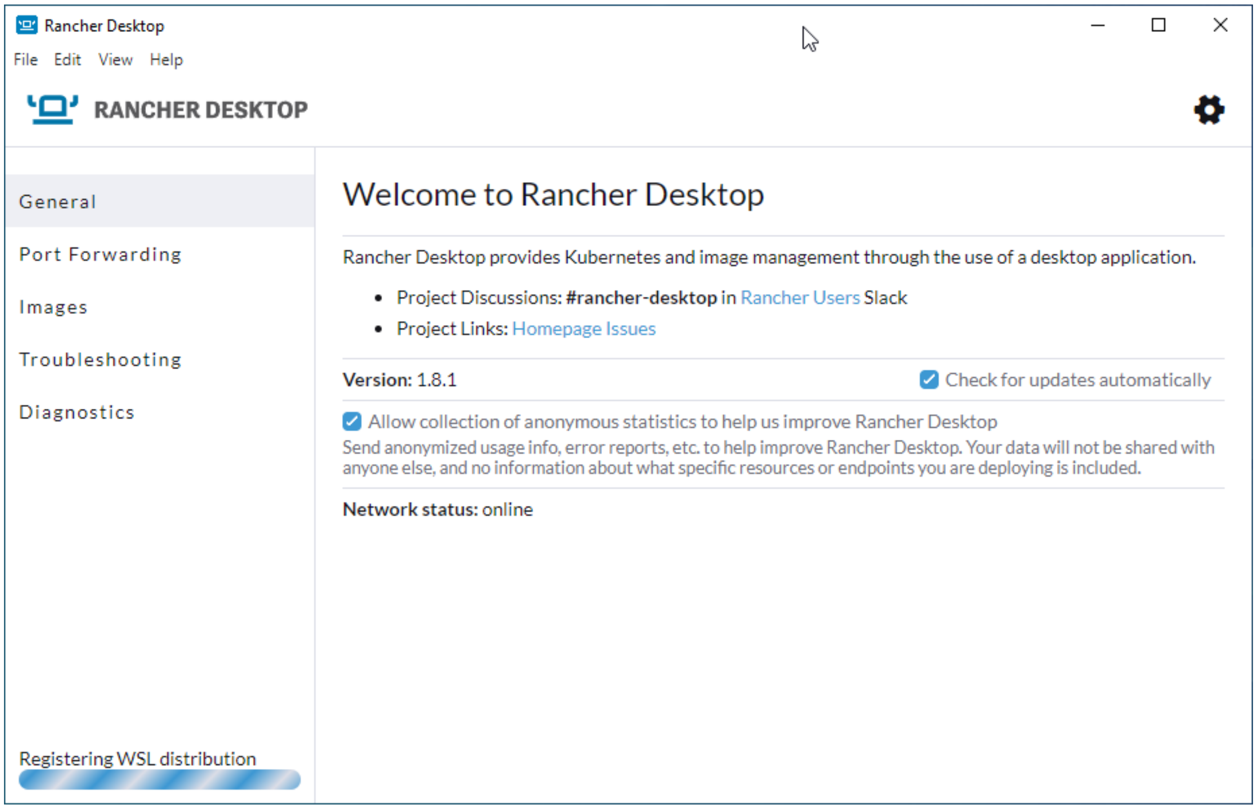
## 5.6 Logging back in



```
Installing: Windows Subsystem for Linux
```

After restart, upon log in WSL2 is installed.

## 5.7 Wait for WSL2 deployment



Welcome to Rancher Desktop

Rancher Desktop provides Kubernetes and image management through the use of a desktop application.

- Project Discussions: **#rancher-desktop** in Rancher Users Slack
- Project Links: Homepage Issues

**Version:** 1.8.1 ✓ Check for updates automatically

✓ Allow collection of anonymous statistics to help us improve Rancher Desktop
Send anonymized usage info, error reports, etc. to help improve Rancher Desktop. Your data will not be shared with anyone else, and no information about what specific resources or endpoints you are deploying is included.

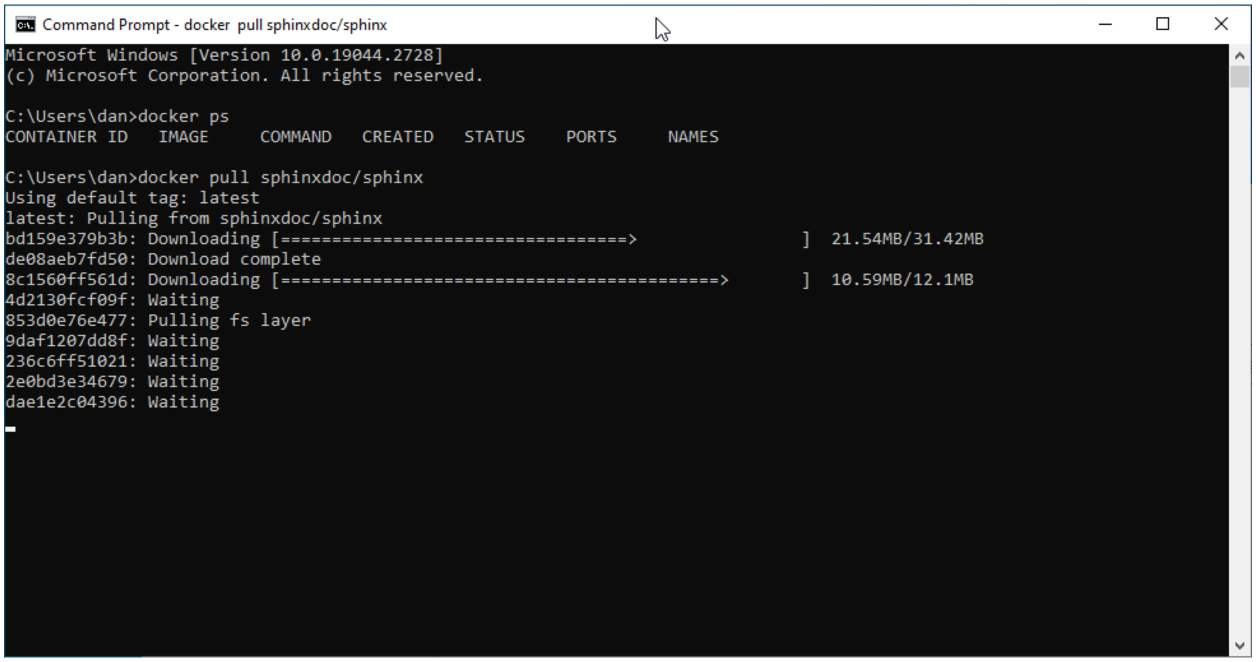
**Network status:** online

Registering WSL distribution

Starting Rancher again shows WSL2 and the docker WSL image are being deployed.

When the messages in the bottom corner are gone, your rancher installation should be complete.

## 5.8 Pull required docker images



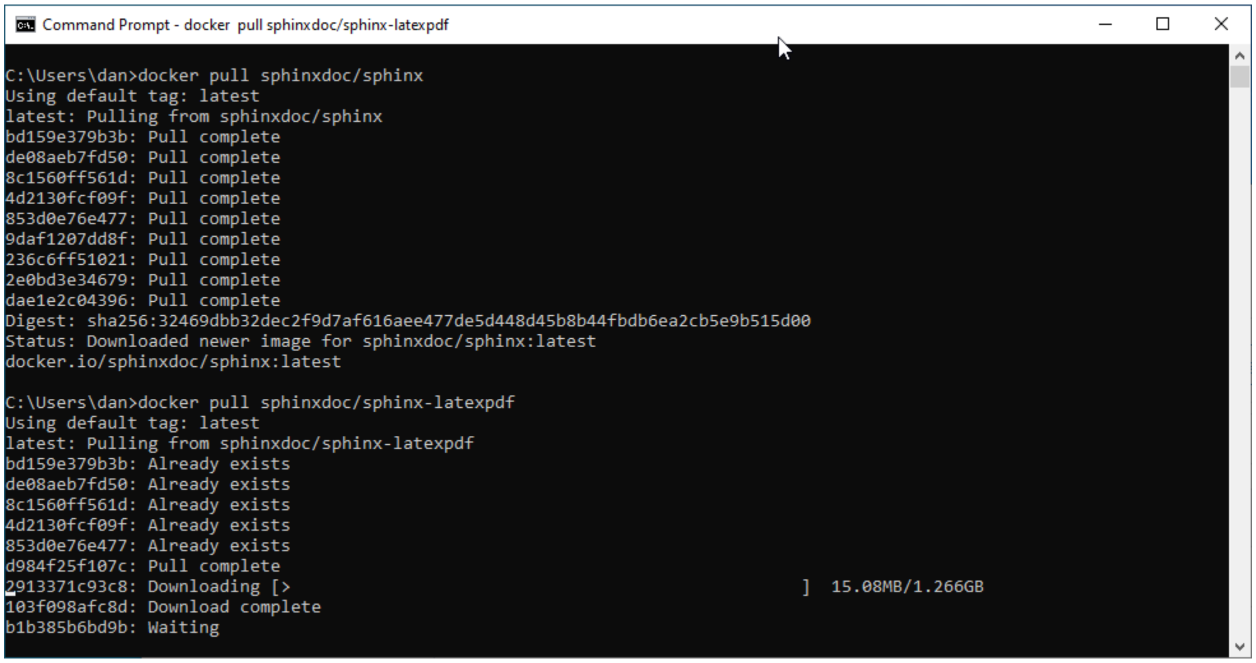
```
Microsoft Windows [Version 10.0.19044.2728]
(c) Microsoft Corporation. All rights reserved.

C:\Users\dan>docker ps
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS     NAMES

C:\Users\dan>docker pull sphinxdoc/sphinx
Using default tag: latest
latest: Pulling from sphinxdoc/sphinx
bd159e379b3b: Downloading [==================================>                ]  21.54MB/31.42MB
de08aeb7fd50: Download complete
8c1560ff561d: Downloading [===========================================>       ]  10.59MB/12.1MB
4d2130fcf09f: Waiting
853d0e76e477: Pulling fs layer
9daf1207dd8f: Waiting
236c6ff51021: Waiting
2e0bd3e34679: Waiting
dae1e2c04396: Waiting
```

You can verify that docker is up by opening a command prompt and typing `docker ps`. It should just return the header row (starting with 'CONTAINER ID'). This tells us that the docker CLI tool can talk to the docker server.

Install the sphinx images. Firstly, the base image.

```
docker pull sphinxdoc/sphinx
```

## 5.9 Install progress



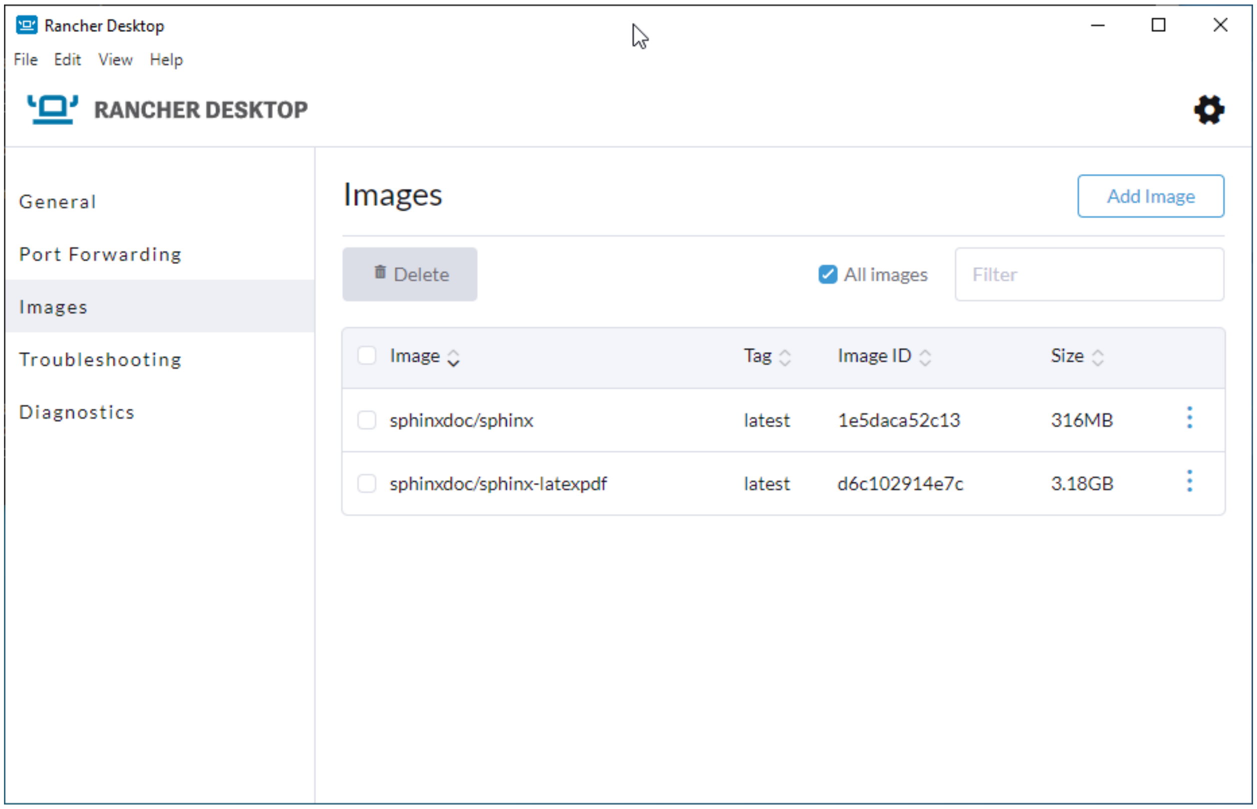
Command Prompt - docker pull sphinxdoc/sphinx-latexpdf

```
C:\Users\dan>docker pull sphinxdoc/sphinx
Using default tag: latest
latest: Pulling from sphinxdoc/sphinx
bd159e379b3b: Pull complete
de08aeb7fd50: Pull complete
8c1560ff561d: Pull complete
4d2130fcf09f: Pull complete
853d0e76e477: Pull complete
9daf1207dd8f: Pull complete
236c6ff51021: Pull complete
2e0bd3e34679: Pull complete
dae1e2c04396: Pull complete
Digest: sha256:32469dbb32dec2f9d7af616aee477de5d448d45b8b44fbdb6ea2cb5e9b515d00
Status: Downloaded newer image for sphinxdoc/sphinx:latest
docker.io/sphinxdoc/sphinx:latest

C:\Users\dan>docker pull sphinxdoc/sphinx-latexpdf
Using default tag: latest
latest: Pulling from sphinxdoc/sphinx-latexpdf
bd159e379b3b: Already exists
de08aeb7fd50: Already exists
8c1560ff561d: Already exists
4d2130fcf09f: Already exists
853d0e76e477: Already exists
d984f25f107c: Pull complete
2913371c93c8: Downloading [>                                                  ]  15.08MB/1.266GB
103f098afc8d: Download complete
b1b385b6bd9b: Waiting
```

Next install the the PDF version. The PDF version of sphinx is over 2Gb, so the download may take some time.

```
docker pull sphinxdoc/sphinx-latexpdf
```

### 5.10 Verify images in Rancher desktop



| Image | Tag | Image ID | Size |
|---|---|---|---|
| sphinxdoc/sphinx | latest | 1e5daca52c13 | 316MB |
| sphinxdoc/sphinx-latexpdf | latest | d6c102914e7c | 3.18GB |

Installed images are listed in Rancher.

In Rancher desktop you can click on the images tab to see installed images.

# 6 Method 3 - Github workflows

One of the nice things we can do with github is to get it to compile our sources on github's own servers. We can create a docker container in the guthub infrastructure and let it do what we want. It can compile our pdf and html documentation each time we push our changes. The actual build process is a little like what is described in Method 2, only it all happens in the cloud.

### 6.1 Configure the github workflow

### 6.2 Create our secrets

### 6.3 Push and check the progress