

How to Install the Typst Compiler Locally on Mint 22.3 (Cinnamon)

By kalwisti

Revised and expanded: 23 Aug. 2026

In a [previous](#) article, I discussed the basic features of Typst, a new typesetting system for the sciences. Although the developers recommend signing up for Typst's [web app](#) for the smoothest writing experience, you can also install Typst locally on your computer — without needing to rely on an internet connection. If you are interested in this use case, I will explain how to install the Typst compiler and configure a code editor (VSCodium) for use with Typst.

Based on my experience, installing the Typst compiler is safe and will not bork your Linux Mint system. I followed the procedure(s) below on three different PCs; I have not experienced any glitches or breakage while I have been using Typst locally.

Although the procedure described here was used on Mint 22.3 Cinnamon, it should work identically on the Xfce and MATE editions.

Contents

1. Method A: Typst Installer Script	1
1.1. Place the Typst Binary in Your \$PATH	3
2. Method B: Typst in Your /home Directory	4
2.1. Download the Typst Binary	4
2.2. Extract the Typst Binary	5
2.3. Place the Typst Binary in Your \$PATH	6
3. Method C, D and E	7
4. Choose a Code Editor to Use with Typst	9
4.1. VSCodium's .DEB Package vs. Flatpak	9
4.2. Install Recommended Extensions for Typst	10
4.3. Error Messages in VSCodium	13
5. Changing the Default Font	14
6. Adding Images to Your Typst Project	15
7. A Better Organized Workflow in VSCodium	16
8. Stay Up to Date with Future Releases	17
9. Additional Resources	17

Note:

Linux allows you the flexibility to install a third-party application such as Typst in multiple ways. I will discuss two installation methods in detail (Section 1 and Section 2), and mention other options you might want to consider in Section 3.

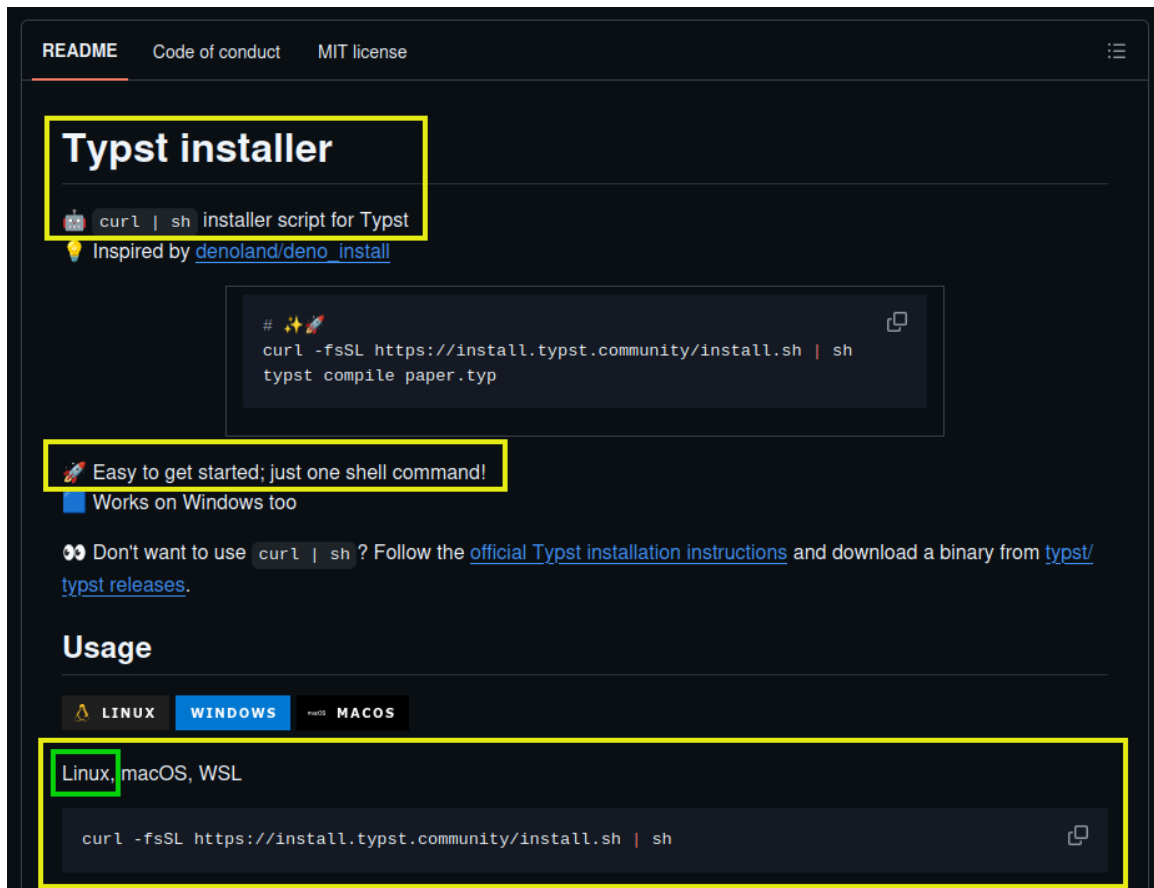
In my opinion, if your computer is a single-user, single-disk machine, then where you install Typst is not a crucial matter. However, if your system has multiple users and/or multiple disks, you should evaluate all these options carefully before deciding.

1. Method A: Typst Installer Script

The easiest, ready-made solution is to use the “[Typst installer](#)” script which is available on GitHub. It

downloads the most recent Typst release from the main branch's root folder and automatically installs it.¹

I should clarify that this resource is unofficial; it is maintained by Typst Community, an organization of volunteer Typst users. Although I see no reason to distrust it, the installation script could theoretically pose a security risk. By using the script, you are extending your trust to the third-party maintainers. So be mindful that there is a tradeoff between convenience and security.



Copy and paste this command into a Terminal. (You can do this as a regular user; it does not require sudo or root privileges.)

```
curl -fsSL https://install.typst.community/install.sh | sh
```

After running the installer script, you should see output similar to the lines below:

```
david@mint-xfce:~$ curl -fsSL https://install.typst.community/install.sh | sh
Downloading Typst from https://github.com/typst/typst/releases/latest/download/typst-x86_64-unknown-linux-musl.tar.xz
Typst installed to /home/david/.typst/bin/typst
Manually add the directory to your $HOME/.bashrc (or similar)
  export TYPST_INSTALL="/home/david/.typst"
  export PATH="$TYPST_INSTALL/bin:$PATH"

Example:
  echo 'export TYPST_INSTALL="/home/david/.typst"' >> ~/.bashrc
  echo 'export PATH="$TYPST_INSTALL/bin:$PATH"' >> ~/.bashrc

Run '/home/david/.typst/bin/typst --help' to get started
Stuck? Open an Issue https://github.com/typst-community/typst-install/issues
```

¹Thanks to member Y.D.X in the Typst Forum for this [tip](#).

1.1. Place the Typst Binary in Your \$PATH

As noted in the Terminal output above, you must now place the typst directory in your \$PATH.

A path — or the search path — is the list of directories that will be searched for anything you type on the command line. After editing your path, when you call/invoke Typst, the system will look for [and find] the typst binary in your /home directory.²

- Open a Terminal, and copy and paste the two commands below separately (as a regular user). They will add the appropriate entries to the .bashrc file in your /home directory.

```
echo 'export TYPST_INSTALL="/home/david/.typst"' >> ~/.bashrc
```

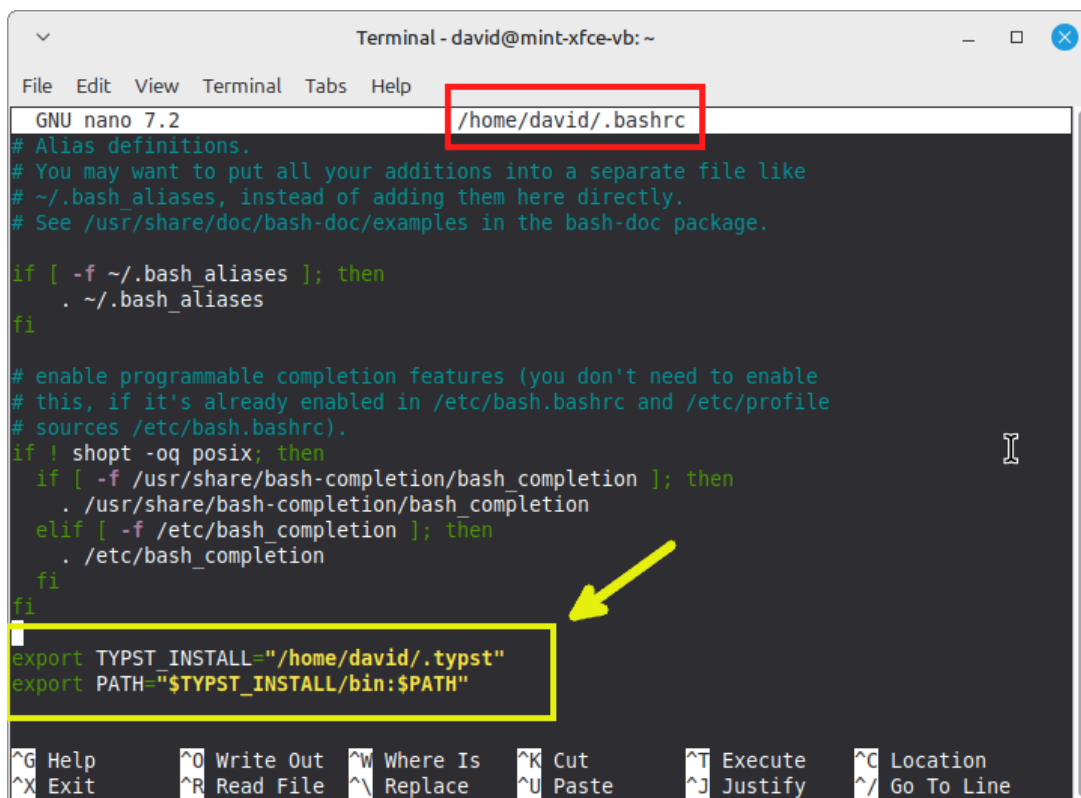
(**Note:** Change “david” to whatever the username is on your system.)

```
echo 'export PATH="$TYPST_INSTALL/bin:$PATH"' >> ~/.bashrc
```

To double-check the result of these commands, open your ~/.bashrc file with your favorite text editor — nano, vim, Emacs, etc. The example below uses nano:

```
$ nano ~/.bashrc
```

You should see something like this screenshot:



To apply this change permanently to your \$PATH environment variable, open a Terminal and issue the command:

```
$ source ~/.bashrc
```

Alternatively, you can log out of your current session, then log back in to make sure the change takes effect.

²If you would like to read additional background on \$PATH, this [article](#) by Dave McKay is helpful and concise.

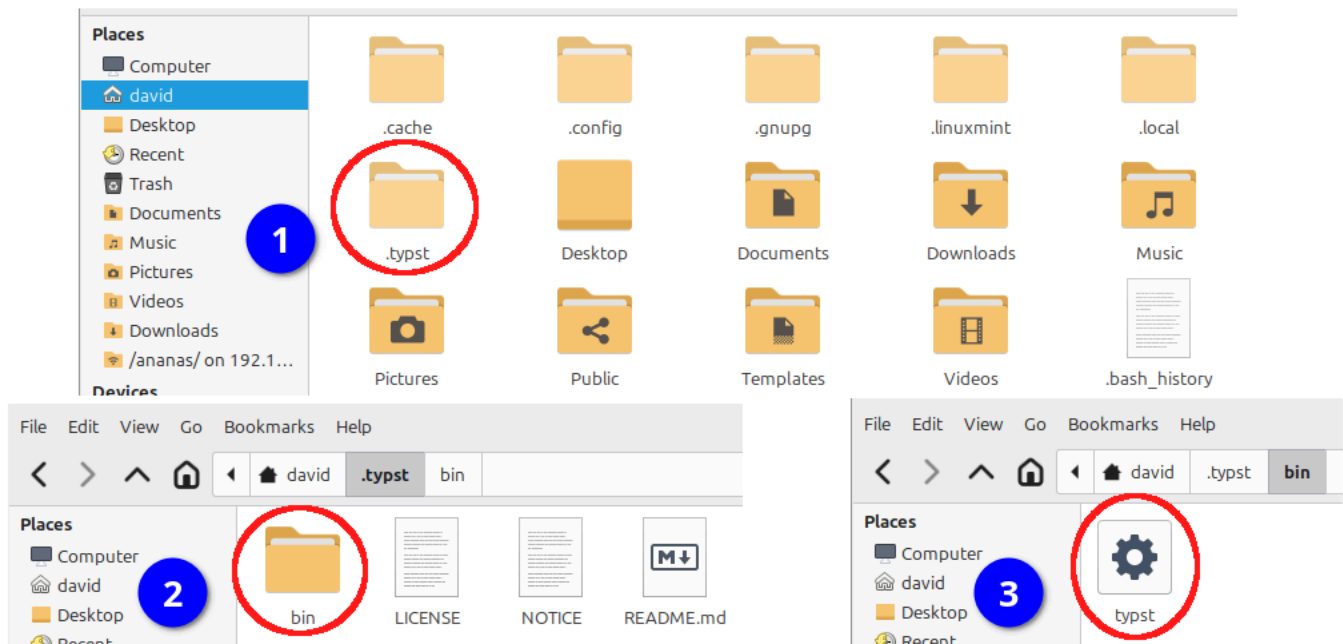
To verify this change, type the command:

```
$ echo $PATH
```

Your results should look something like the screenshot below:

```
File Edit View Terminal Tabs Help
david@mint-xfce-vb:~$ echo $PATH
/home/david/.typst/bin:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games:/snap/bin
david@mint-xfce-vb:~$
```

This screenshot montage shows the directory structure (`~/ .typst/bin/typst`) created by the `typst-installer` script.³



Typst is now ready to use. Unless you want to use Typst via the CLI, jump to Section 4 for information on choosing a GUI-based code editor.

2. Method B: Typst in Your /home Directory

2.1. Download the Typst Binary

You should first download the Typst pre-built binary from <https://github.com/typst/typst/releases>

- Look for Release ver. 0.15.1 (July 17, 2026)
- Download the file: `typst-x86_64-unknown-linux-musl.tar.xz`

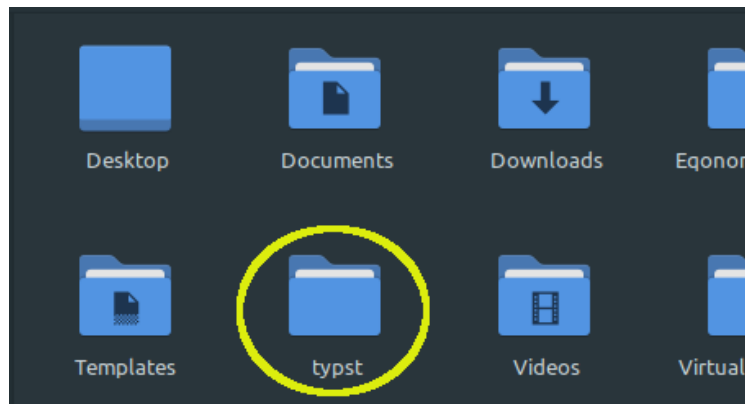
³Hidden files were made visible by opening the Nemo file manager and typing `Ctrl + H`. You can also toggle on the visibility of hidden files by opening `Nemo >` and going to the **View** menu > then choosing the option **Show Hidden Files**.



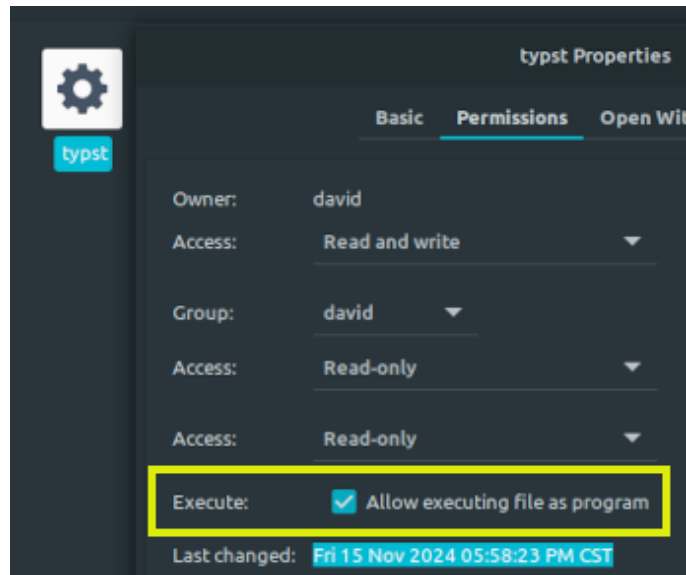
This contains the Typst compiler and its CLI, which is everything you need to compile Typst documents locally.

2.2. Extract the Typst Binary

- Move the tarball from your Downloads folder to your /home folder.
- Extract/unzip the tarball. It is a modest 55.8 MB in size.
- To simplify the naming convention, I renamed the unpacked directory from “typst-x86_64-unknown-linux-musl” to just “typst”.



- Within the typst directory, use your DE’s file manager (Nemo, Thunar, Caja, etc.) to double-check that the typst binary is set as executable:



2.3. Place the Typst Binary in Your \$PATH

- You must now place the typst directory in your \$PATH.

Your default \$PATH in Linux Mint probably looks something like this:

```
File Edit View Search Terminal Help
david@david-ASUSTek:~$ echo $PATH
/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local
/games:/snap/bin
david@david-ASUSTek:~$
```

To (permanently) add the typst directory to your PATH,

- Open a Terminal and issue the command below in your /home directory to edit the .bashrc file:
(I chose the nano editor but you may use whatever your preferred text editor is — vim, Emacs, etc.)
- Type the command:

```
$ nano ~/.bashrc
```

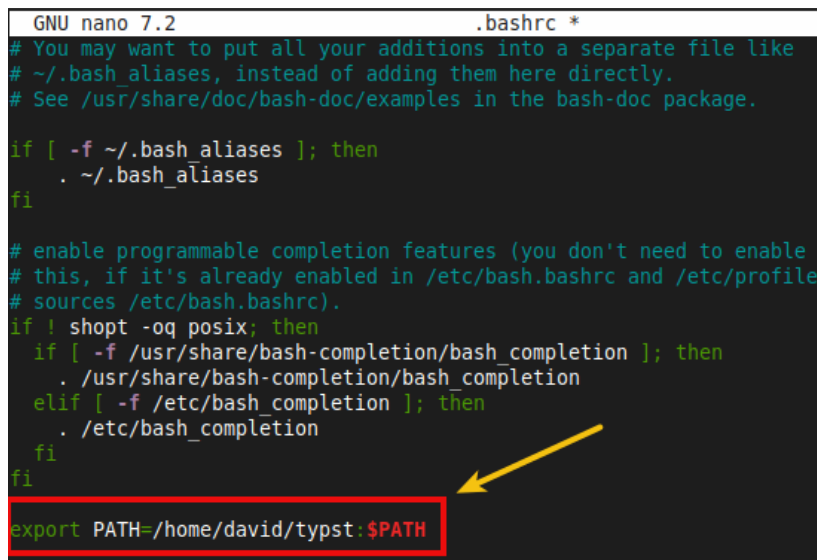
Do not forget to type the initial period (“.”) before bashrc

- Add the line

```
export PATH=/home/david/typst:$PATH
```

at the bottom of the existing .bashrc file.

(Change “david” to whatever the username is on your system.)



```
GNU nano 7.2 .bashrc *
# You may want to put all your additions into a separate file like
# ~/.bash_aliases, instead of adding them here directly.
# See /usr/share/doc/bash-doc/examples in the bash-doc package.

if [ -f ~/.bash_aliases ]; then
    . ~/.bash_aliases
fi

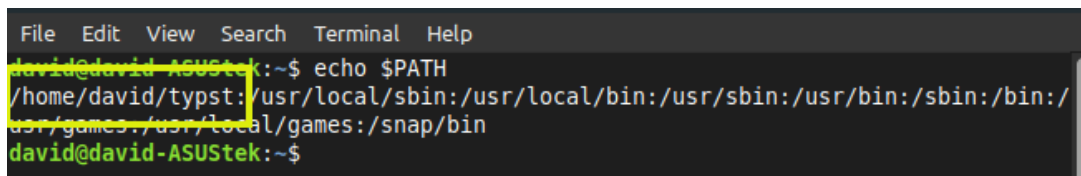
# enable programmable completion features (you don't need to enable
# this, if it's already enabled in /etc/bash.bashrc and /etc/profile
# sources /etc/bash.bashrc).
if ! shopt -oq posix; then
    if [ -f /usr/share/bash-completion/bash_completion ]; then
        . /usr/share/bash-completion/bash_completion
    elif [ -f /etc/bash_completion ]; then
        . /etc/bash_completion
    fi
fi

export PATH=/home/david/typst:$PATH
```

- Save the file and exit your text editor.
- Log out of your session, then log back in to make sure the change takes effect.
- To double-check the results, type:

```
$ echo $PATH
```

Your results should look something like the screenshot below:



```
File Edit View Search Terminal Help
david@david-ASUSTek:~$ echo $PATH
/home/david/typst:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games:/snap/bin
david@david-ASUSTek:~$
```

Typst is now ready to use. Unless you want to use Typst via the CLI, jump to Section 4 for information on choosing a GUI-based code editor.

3. Method C, D and E

As mentioned above, a determining factor will be whether you are the single user on the PC and will be the only person using Typst, or whether your goal is to make Typst available to multiple users.

Method C Place the Typst binary in `~/.local/bin`

If it is a single-user system, this is another viable option. It was suggested by member Y.D.X. in the [Typst forum](#). He recommends issuing the command

```
mkdir -p ~/.local/bin/
```

to create the required bin subdirectory.

(The “ -p “ flag enables the command to create parent directories as necessary.)

Afterwards, you should manually add the Typst directory to your `~/.bashrc` file.

```
export PATH=~/.local/bin/typst:$PATH
```

Log out of your current session, then log back in for this change to take effect.

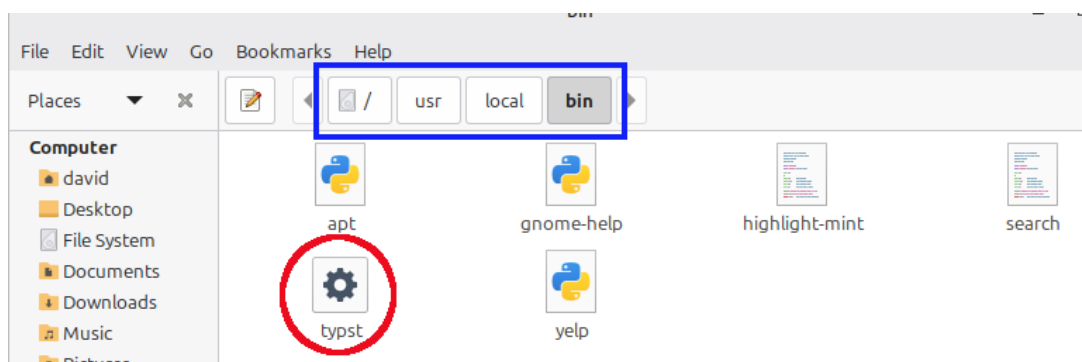
Method D Another option is to place it in `/usr/local/bin`

This approach is sensible if you wish to adhere to the Linux Filesystem Hierarchy Standard (*FHS*) and/or need to make Typst available to multiple users.⁴

“The `~/bin` directory is a good place to put scripts intended for personal use. If we write a script that everyone on a system is allowed to use, the traditional location is `/usr/local/bin`. Scripts intended for use by the system administrator are often located in `/usr/local/sbin`. In most cases, locally supplied software, whether scripts or compiled programs, should be placed in the `/usr/local` hierarchy and not in `/bin` or `/usr/bin`. These directories are specified by the Linux Filesystem Hierarchy Standard to contain only files supplied and maintained by the Linux distributor.”

— William Shotts, *The Linux Command Line*, 2026, p. 386.

If you copy (or move) only the typst executable [55.7 MB] to `/usr/local/bin`, you do **not** have to make any adjustments to your `$PATH` because the `/usr/local/bin` directory is already in Linux Mint’s default `$PATH`. Typst will be available for use without you needing to take further action.



However, if you copy (or move) the entire typst folder and its contents — the typst executable, its License, Notice and `README.md` — you will have to manually add the path to that directory in your `~/.bashrc` file:

```
export PATH=/usr/local/bin/typst:$PATH
```

Log out of your current session, then log back in for this change to take effect.

Method E Place the Typst binary in `/opt`

`/opt` is intended for files of third-party applications which you want to make available to multiple users. This directory can store programs outside of your distribution’s control, i.e., not available in Linux Mint’s repository.

Afterwards, manually add the `$PATH` to the Typst directory in your `~/.bashrc` file:

```
export PATH=/opt/typst:$PATH
```

Log out of your current session, then log back in for this change to take effect.

After applying Method C, D or E, Typst is now ready to use. Unless you want to use Typst via the CLI, jump to Section 4 for information on choosing a GUI-based code editor.

⁴Abhishek Prakash wrote an [introduction](#) to the Filesystem Hierarchy Standard; it is worth reading if you would like an overview of the Linux directory structure.

4. Choose a Code Editor to Use with Typst

Although Typst can be used exclusively from the command line (as described on the [GitHub](#) page), in my experience it is more comfortable to use the app within a code editor.

The two commonly recommended choices are Visual Studio Code (often abbreviated as “VS Code”) and VSCodium.

VS Code is a Microsoft product licensed under a non-FLOSS [license](#) and it has telemetry reporting enabled (although the telemetry can be [disabled](#)). VS Code is a cross-platform code editor that straddles the line between a text editor and an integrated development environment (IDE).

I prefer VSCodium because it is a free/libre binary distribution, licensed under the MIT license (open-source) without Microsoft’s proprietary additions.

Note:

Other editors may be used with Typst, if you wish. Alternatives include a terminal-based text editor such as [Vim](#) (with a special plugin) and GUI-based code editors such as [Geany](#).

If you plan to use only VS Code/VSCodium + the Tinymist extension (see Section 4.2), then you do not have to install the Typst compiler separately because the Typst compiler is already embedded in Tinymist.

4.1. VSCodium’s .DEB Package vs. Flatpak

To save disk space, I chose to install the .DEB package rather than the Flatpak. There are easy-to-follow [instructions](#) on VSCodium’s website; look for the section “*Install on Debian / Ubuntu (deb package)*”:

```

Install on Debian / Ubuntu (deb package):

Add the GPG key of the repository:

wget -qO - https://gitlab.com/paulcarroty/vscodium-deb-rpm-repo/raw/master/pub.gpg \
| gpg --dearmor \
| sudo dd of=/usr/share/keyrings/vscodium-archive-keyring.gpg

Add the repository:

echo 'deb [ signed-by=/usr/share/keyrings/vscodium-archive-keyring.gpg ] https://download.vscodium.c
| sudo tee /etc/apt/sources.list.d/vscodium.list

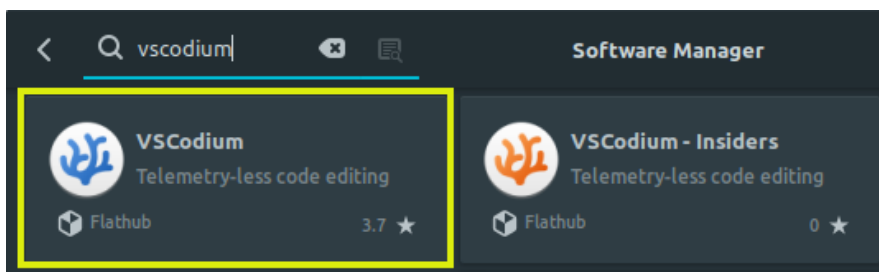
Update then install vscodium (if you want vscodium-insiders, then replace codium by codium-insiders):

sudo apt update && sudo apt install codium

```

The current release of VSCodium is ver. 1.126.04524 (at the time this guide was written).

I use the Flatpak on one of my production machines; it has performed smoothly. The Flatpak is listed in the Mint Software Manager because it is a verified app.



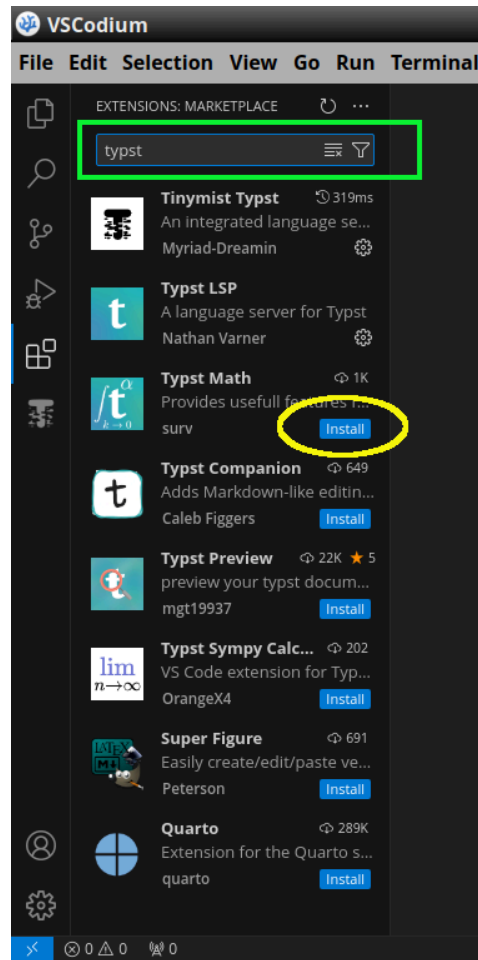
4.2. Install Recommended Extensions for Typst

Extensions (or plugins) are third-party software packages which extend the code editor's functionality and improve your workflow. You download and install these extensions from the so-called "Extension Marketplace." Due to licensing restrictions, Microsoft's official extensions for VS Code are not available in the VSCodium extension marketplace (which instead uses the open-source registry <http://open-vsx.org>). However, that does not pose a problem because the Typst extensions we need are available through open-vsx.org.

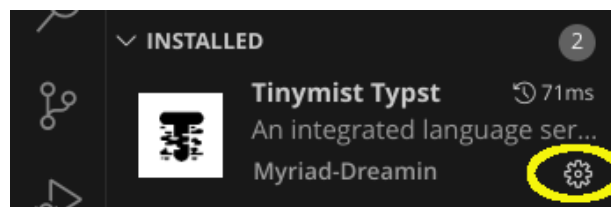
First, find the **Extensions** icon in the Activity Bar of the left-hand panel:



Next, type “ typst “ in the Search box to filter the available extension offerings:

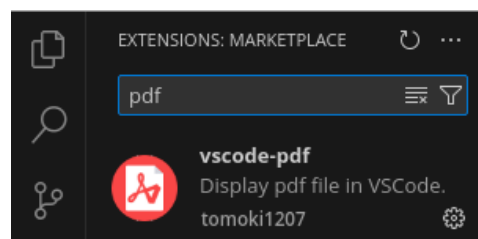


To install an extension, click on the **Install** button beside its description (as shown above). When the installation finishes, the **Install** button will change to the **Manage** button [a gear icon]:



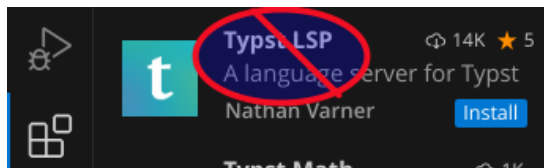
The two extensions I am using with Typst are:

- | | | |
|-----------------------|---------------|--|
| Tinymist Typst | (shown above) | An integrated language service for Typst |
| vscode-pdf | (shown below) | Display PDF in VSCode (or VSCodium) |

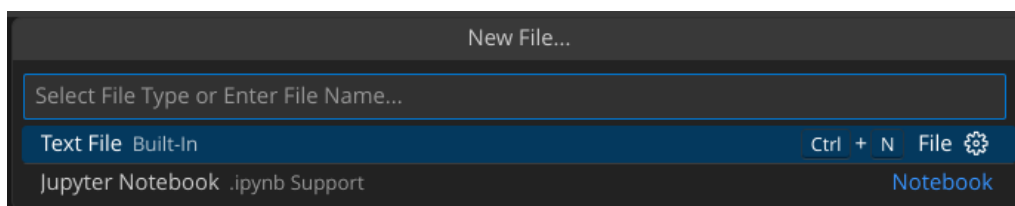


An advisory on the Typst LSP extension: This community-provided extension is no longer recommended. The author did not update it beyond Typst ver. 0.11.0 (Typst's current version is 0.15.1) and he has recently archived its project repository.

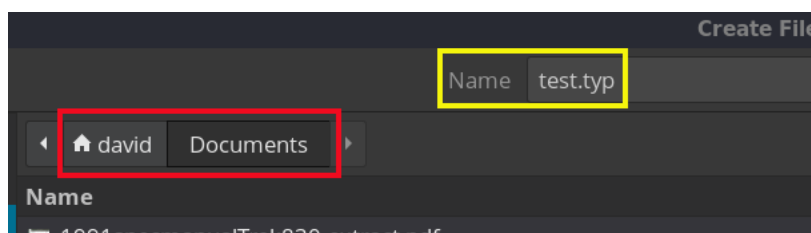
All functionality provided by Typst LSP in the past is available in the Tinyminst Typst extension. (I would like to thank Martin Haug, the co-founder of Typst, for clarifying this point.)



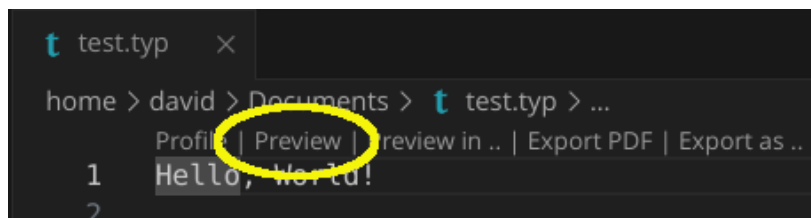
To start a new Typst file, from the **File** menu > select the **New File** option. A dialog box will open in the editor's toolbar:



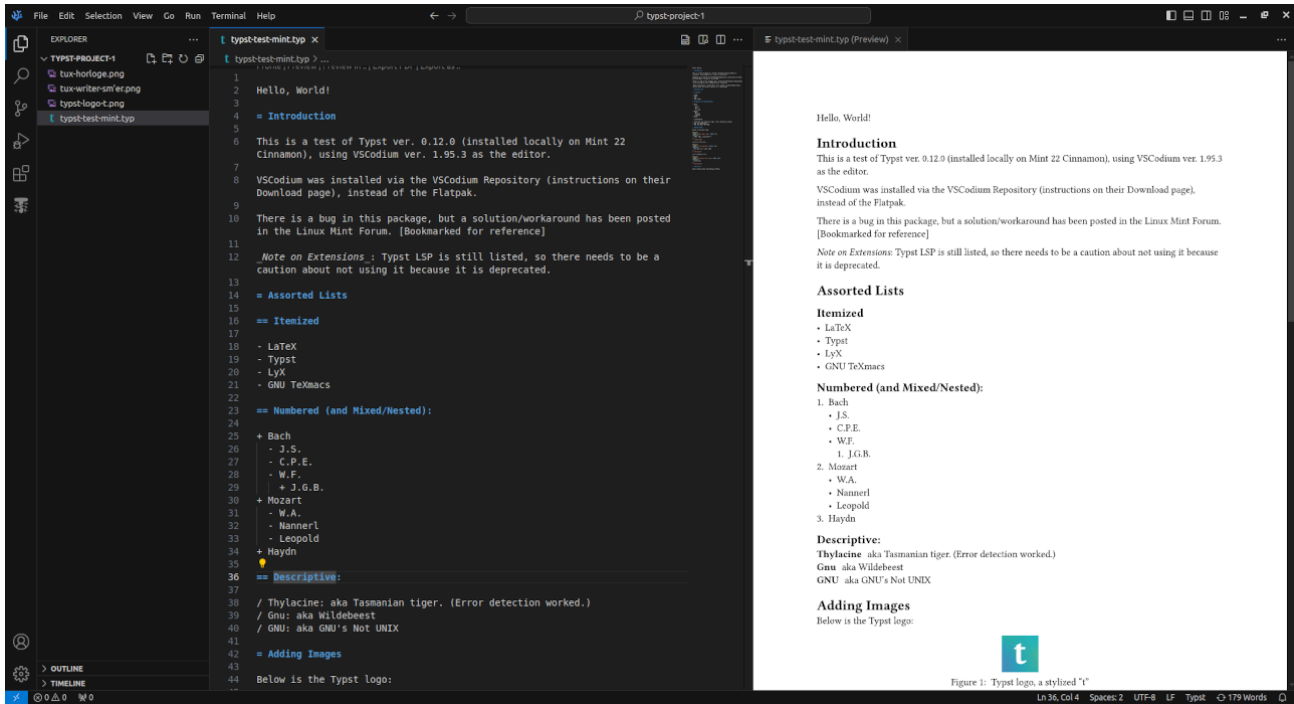
Enter the name of your new file (for this example, I called it “test.typ”), press Enter/Return and choose the directory (e.g., Documents) where you want to save the file.



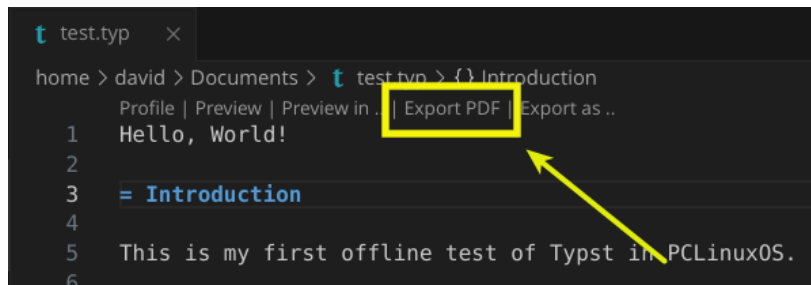
To open a preview of your document, just click on the **Preview** tab in the top portion of the editor pane:



The screenshot below shows a Typst test file, open in the VSCodium editor, with a document Preview displaying in the right-hand pane:

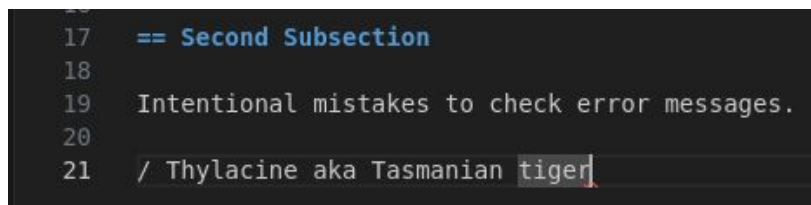


Clicking on the **Export PDF** tab will export your current document as a PDF to your Documents folder, and display the generated PDF in a new tab of VSCode's right-hand pane.

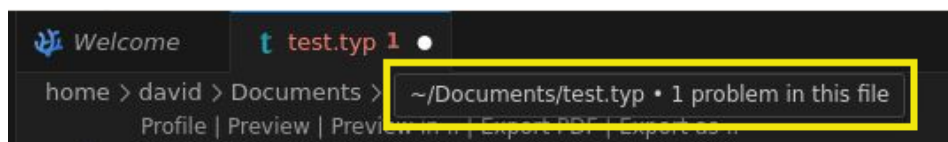


4.3. Error Messages in VSCode

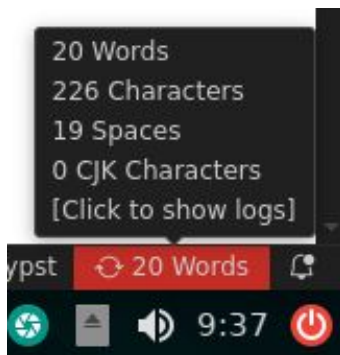
Error messages display differently in the VSCode editor than in Typst's web app. The Tinyminst Typst extension which I installed earlier handles the function of error reporting. In the screenshot below, I was using Typst's “[term list](#)” function and intentionally omitted the mandatory colon following the term. (It should be: “/ Thylacine:”):



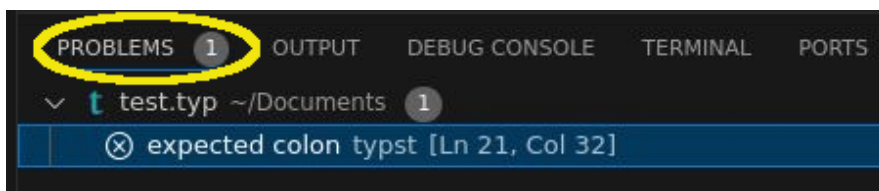
The active file name in the top tab turns red and a brief message appears near the editor's toolbar, notifying me that there is a problem in the file:



To view Typst's logs, click on the Notification panel in the lower right corner of the editor:



Clicking on the **Problems** tab of the log will display the specific error message as well as its location in the file:

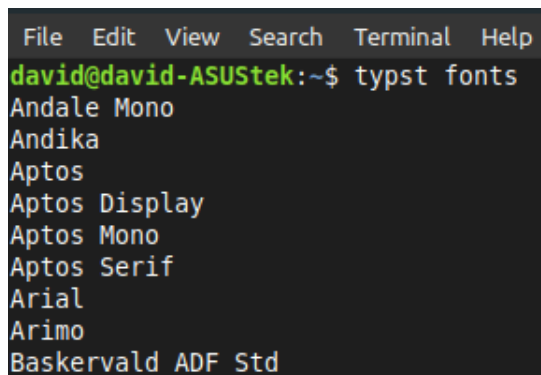


5. Changing the Default Font

By default, Typst uses fonts embedded in the CLI: Libertinus Serif, New Computer Modern, New Computer Modern Math, and DejaVu Sans Mono.

However, the Typst compiler can also use system-wide fonts that you have installed. To display a list of available fonts, open a Terminal and type the command:

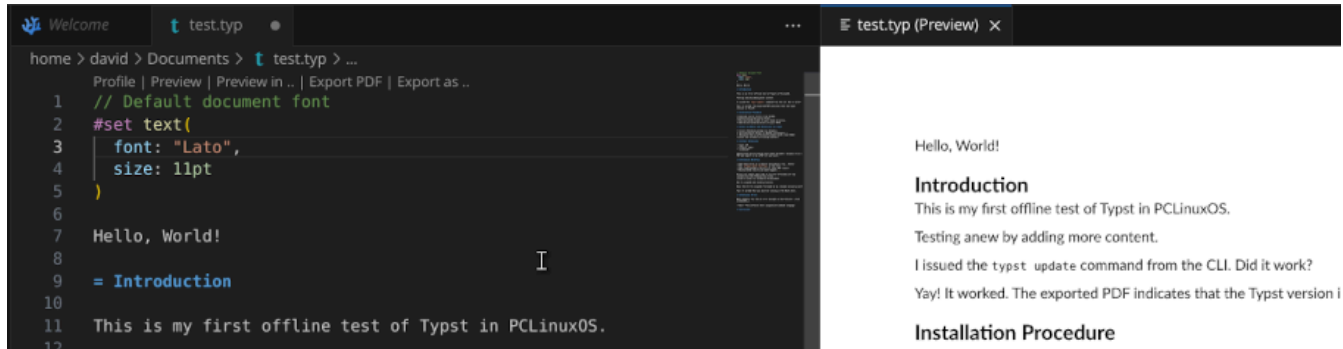
```
$ typst fonts
```



After you have decided which typeface to use, add this `#set` command at the beginning of your document:

```
// Default document font
#set text(
  font: "Lato",
  size: 11pt
)
```

Change the typeface and font size to suit your taste. (Lato, the Polish word for ‘summer’, is an attractive typeface [shown below] which was designed by Łukasz Dziedzić.)



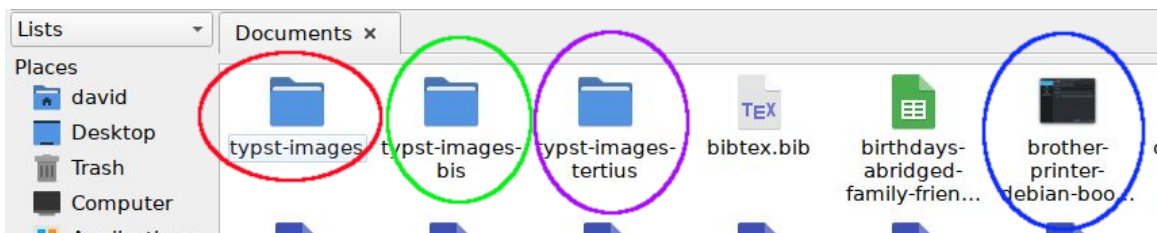
6. Adding Images to Your Typst Project

It was not immediately obvious where to store image files that I wanted to add to my project, so I experimented with a few different methods.

I created my example file (“test.typ” [pictured above]) in my Documents folder. VSCode expects to find both your Typst file and any image files in the same folder.

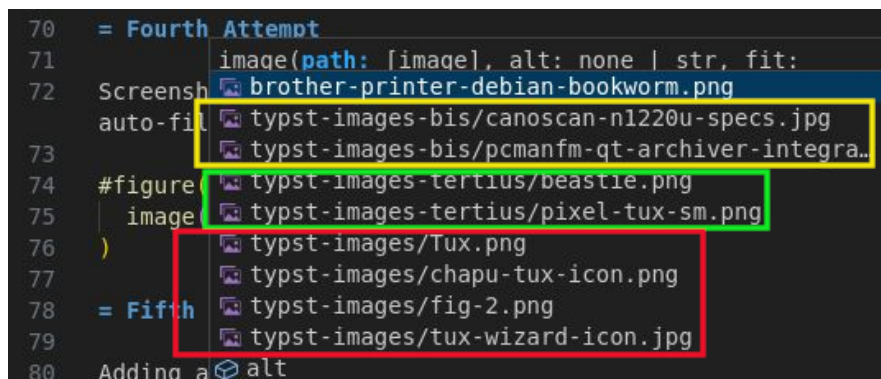
If your Typst file contains numerous images, you will find that your Documents folder quickly becomes cluttered — something you probably want to avoid. I learned that it is possible to create subfolders within the Documents folder, which store your image files.

In the screenshot below, you can see that I have one image file (circled in blue) in my Documents folder, as well as three image test subfolders (circled in red, green and purple):



When you type the code for Typst’s [figure function](#) to insert a figure, the editor will help autofill the image file name(s). If the image is in the same folder as your Typst document, just use the image file name. However, if your image is located in a subfolder, you must include the path, e.g., /typst-images/Tux.png, etc.

The screenshot below illustrates that VSCode will detect image files in the Documents folder, in addition to the typst-images, typst-images-bis, and typst-images-tertius subfolders. The code completion autofill should begin working when you type the “**image(**” portion of the code snippet:



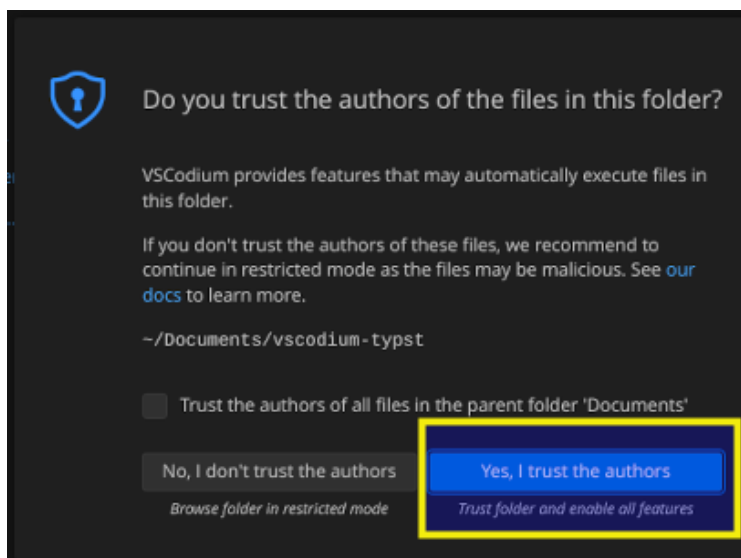
7. A Better Organized Workflow in VSCodium

After skimming through the Visual Studio Code [documentation](#) and watching a couple of video [tutorials](#), I realized that the recommended approach of using a folder (also known as a workspace) results in a cleaner workflow.

Follow the procedure below to benefit from a more organized workflow:

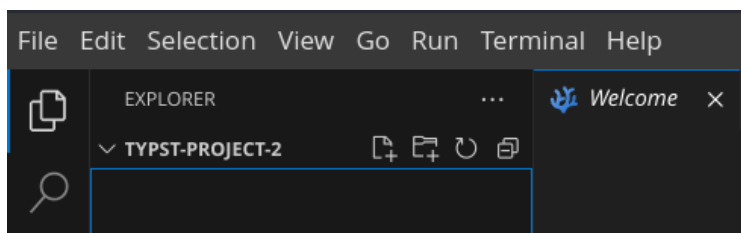
- Create a new (empty) project folder in your Documents folder. (I called this example “typst-project-2”)
- Start the VSCodium editor.
- From the **File** menu > choose **Open Folder**, then select the “typst-project-2” folder.
- The “Workspace Trust” dialog will appear:

This feature lets you decide whether code in your project folder can be executed by VSCodium and extensions without your approval.

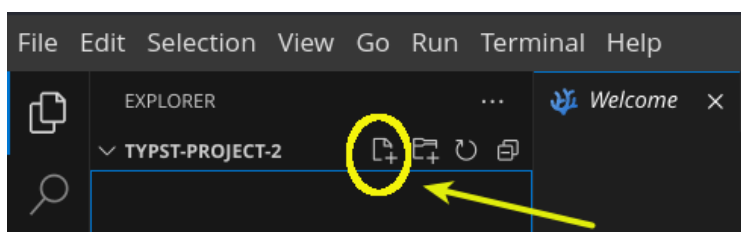


- Click on the blue “Yes, I trust the authors” button.

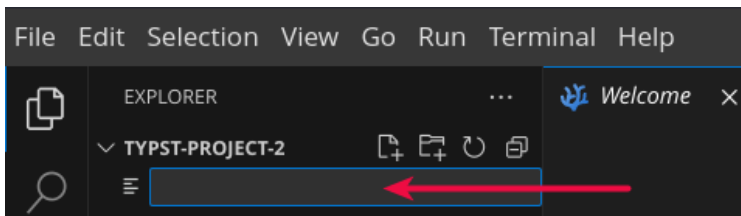
Your new folder (aka workspace) will display in the **Explorer** tab [the stacked sheets of paper icon] of the Activity Bar in the editor. The Explorer view lets you view and manage the files and folders in your workspace:



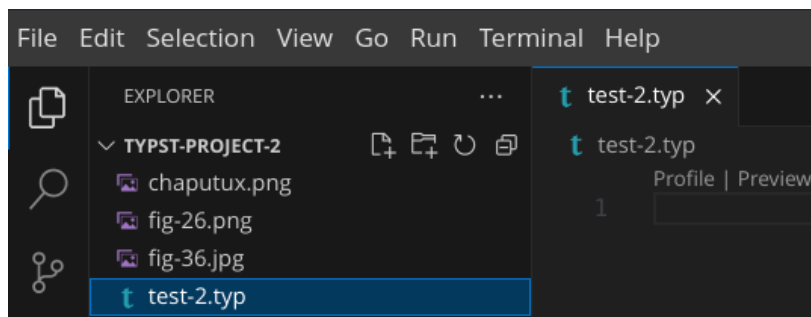
- Click on the **New File** icon:



- In the box, type the name of the Typst file which will be created in your workspace. (I called mine “test-2.typ”):



- Copy any image files you wish to use into your “typst-project-2” folder.
- The image files and the Typst file will be listed under the Explorer tab:



- Add the image files to your Typst document as you normally would, via the “**#figure**” command.

8. Stay Up to Date with Future Releases

Once you have installed Typst locally using this procedure, you can stay up to date with future releases by opening a Terminal and issuing the command:

```
$ typst update
```

I hope these instructions will allow you to successfully install Typst on your local Mint 22.3 system and use it productively with VSCodium.

9. Additional Resources

Although the resources mentioned below focus on Visual Studio Code, they are also helpful explaining the basics of VSCodium because VSCodium maintains feature parity with VS Code.

<https://code.visualstudio.com/docs/>
Visual Studio Code Documentation (official)

The tutorial on [getting started](#) with VS Code provides a succinct overview.

<https://www.youtube.com/watch?v=ORrELERGIHs>

Ruscica, Tim. *Tech with Tim*. “VSCode Tutorial for Beginners: Getting Started with VSCode.” *YouTube*, 13 Feb. 2021. (22 min., 56 sec.)

<https://www.youtube.com/watch?v=yjeHLSrhPao>

Coding with Estefanía. “Learn Visual Studio Code: Course for Beginners.” *YouTube*, 20 Jan. 2022. (43 min., 15 sec.)

This tutorial was created with Typst. The document’s body typeface is Source Serif Pro and the headings use Source Sans Pro.



Good luck and Have fun using Typst!