

# Read Me First

February 8, 2021

## Contents

<a href="#">1</a>	<a href="#">T<sub>E</sub>X, L<sup>A</sup>T<sub>E</sub>X, and TeXShop</a>	<a href="#">1</a>
<a href="#">2</a>	<a href="#">First Steps with T<sub>E</sub>X and TeXShop</a>	<a href="#">2</a>
<a href="#">3</a>	<a href="#">Second Steps with T<sub>E</sub>X</a>	<a href="#">4</a>
<a href="#">4</a>	<a href="#">MacT<sub>E</sub>X for Experts</a>	<a href="#">5</a>

## 1 T<sub>E</sub>X, L<sup>A</sup>T<sub>E</sub>X, and TeXShop

T<sub>E</sub>X is a typesetting program by Donald Knuth, originally designed for mathematical and scientific documents and increasingly used in other fields as well. The MacT<sub>E</sub>X distribution contains the T<sub>E</sub>X program and a large number of support files: fonts, style files, and documentation.

Most of these files remain hidden from the user, who interacts with T<sub>E</sub>X via a front end. MacT<sub>E</sub>X installs the front end TeXShop, which has been available for macOS since the introduction of the operating system.

The ‘smallest’ possible T<sub>E</sub>X is what Knuth called ‘virgin’ T<sub>E</sub>X and knows only primitive input commands. Plain T<sub>E</sub>X is a set of macros (developed by Knuth) which makes T<sub>E</sub>X usable in the everyday life of a typist. This explanation comes from the web site <https://tex.stackexchange.com/questions/97520/what-is-plain-tex>.

Later on, Leslie Lamport developed a more extensive macro set, called L<sup>A</sup>T<sub>E</sub>X, which is often used today, and will be referenced in this document. If you are writing in L<sup>A</sup>T<sub>E</sub>X, you are using T<sub>E</sub>X with a particular set of macros, and you are probably adding many of the style files and packages developed by a multitude of others after Lamport’s work.

## 2 First Steps with T<sub>E</sub>X and TeXShop

Go to `/Applications/TeX` and find TeXShop. Drag its icon to your dock. Click this icon to run the program. You will be presented with a blank window; at the top right of this window you’ll find a pull down menu named “Templates”. From this menu choose “LaTeX Template”. The blank

window will fill with some standard boilerplate required in each  $\text{\TeX}$  source file. The red lines are comments which  $\text{\TeX}$  ignores.

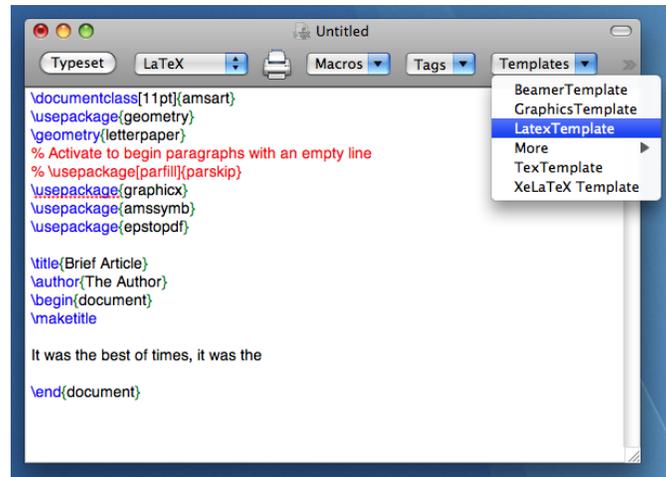


Figure 1: Edit Window

New material goes between the lines

```
\begin{document}
```

and

```
\end{document}
```

Type some sentences there now.  $\text{\TeX}$  will ignore most carriage returns because it knows how to format text, so insert them randomly if you wish. Use a blank line to indicate the start of a new paragraph.

Uncomment the source line with the words “`\usepackage[parfill]{parskip}`” if you prefer to separate paragraphs with an empty line instead of an indent.

When you have some material, hit the “Typeset” button at the top left of the window. A dialog will appear asking you to save the document. It is a standard Save Dialog which has a minimal form and a more extensive form. Both dialogs have a field at the top where you give the document a name, and a field below that showing the folder where the document will be saved. At the right side of this field there is a small arrow. If the arrow points down, click it and the dialog will expand to a form which allows you to navigate to other locations and create new folders.

When  $\text{\TeX}$  typesets, it creates three or four additional files, so it is not a good idea to save directly to a location with many other files. Instead, navigate to a reasonable location, say `~/Documents`, and then click the “New Folder” button at the bottom of the dialog. Accept the default name or choose another and create the folder. Then name the document and save it. The default “Untitled” name will do.

As soon as you save,  $\text{\TeX}$  will typeset the document and open a second window showing the result.

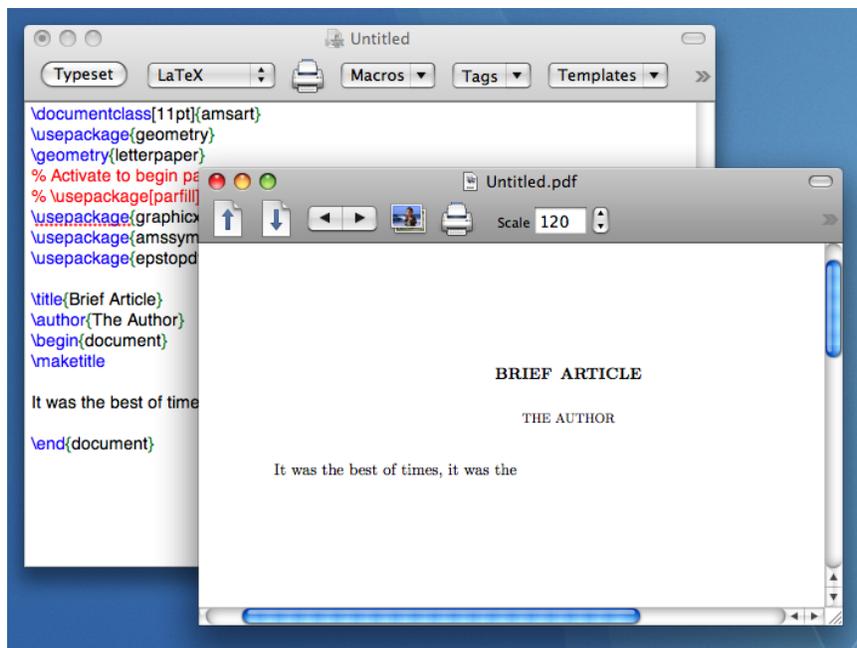


Figure 2: Typeset Window

Go back to the original window and add some additional text. Hit “Typeset” again. This time  $\text{\TeX}$  immediately typesets the new material and updates the output window.

Close the document by hitting the red button at the top of the edit window. There is no need to save the program first because TeXShop automatically saves regularly. Quit TeXShop.

Restart TeXShop by clicking on the icon in the dock.

It is possible to reopen the project by double clicking on the Untitled file or by starting TeXShop and navigating to Untitled in the Open dialog, but there is a much easier way. Start TeXShop and under the File menu choose the item “Open Recent...” In the resulting list, choose “Untitled” and you will immediately return to the document.

If you do not close the document before quitting TeXShop, then it may reappear automatically the next time you run TeXShop. This depends on a setting in Apple’s System Preferences. Run System Preferences and select the General module. In the middle of the resulting dialog, boxes appear labeled “Ask to keep changes when closing documents” and “Close windows when quitting an app”. If both are unchecked, then TeXShop and most other programs will reopen windows automatically if they were open when the program quit.

One configuration step is highly recommended when you first run TeXShop. Open a project which has both source and preview windows. Resize and position these windows as you prefer. Most

users position the source window on the left half of the screen and the preview window on the right half of the screen, with both covering most of their half of the screen, but the choice is up to you. Select the source window and at the bottom of the TeXShop “Source” menu select “Save Source Position”. Select the preview window and at the bottom of the TeXShop “Preview” menu select “Save Preview Position.” After this step is done, all TeXShop windows will open in these positions.

Those are the basics. You will find further information in the TeXShop Help menu. The item “TeXShop Demos” contains two short movies. The first illustrates starting and typesetting a short document, and the second illustrates the configuration just made.

### 3 Second Steps with T<sub>E</sub>X

The next crucial step is to learn T<sub>E</sub>X. The easiest way to do that is to pick one or two short introductions to T<sub>E</sub>X and work through them while trying examples in TeXShop. After a week or so working in this manner, it will be time to start using the program to write lecture notes and articles.

Ask friends where to start. Because it is how I started, I’d recommend Leslie Lamport’s book *L<sup>A</sup>T<sub>E</sub>X, A Document Preparation System*. Be sure to get the second edition, which will say “Updated for L<sup>A</sup>T<sub>E</sub>X 2e”. It would be enough to read chapters 2 and 3, a total of 52 pages.

TeX Live contains a free 139 page book on L<sup>A</sup>T<sub>E</sub>X, *The Not So Short Introduction to L<sup>A</sup>T<sub>E</sub>X 2e*, by Tobias Oetiker, Hubert Partl, Irene Hyna and Elisabeth Schlegl. See <file:///Library/TeX/Root/texmf-dist/doc/latex/lshort-english/lshort.pdf>.

The link <https://learnlatex.org> leads to a series of online L<sup>A</sup>T<sub>E</sub>X lessons, with interactive examples, which were partly funded by the TeX User Group in 2020.

Another source is the first portion of George Grätzer’s book *More Math Into L<sup>A</sup>T<sub>E</sub>X*. Grätzer gave permission to put this section in the TeXShop Help window, so you already have this as well.

After reading one of these short works, I’d recommend starting to use the program for serious writing. This will be frustrating at first, and you’ll probably want to buy one of the excellent books on T<sub>E</sub>X for reference as you work. But don’t ignore Google! A vast amount of information is available about T<sub>E</sub>X and L<sup>A</sup>T<sub>E</sub>X by asking succinct questions of Google. Try “how do I type a url in LaTeX” if you don’t believe me.

### 4 MacT<sub>E</sub>X for Experts

MacT<sub>E</sub>X installs T<sub>E</sub>X Live 2021 in `/usr/local/texlive/2021`. The distribution it installs is exactly the same as the distribution that would be obtained by using T<sub>E</sub>X Live’s standard install script. T<sub>E</sub>X Live runs on almost all architectures: macOS, Windows, GNU/Linux, and other Unix systems. The distribution is the same on all of these platforms; nothing has been added or removed to customize it for macOS.

The actual binary programs are in `/usr/local/texlive/2021/bin/universal-darwin`. Each is a universal binary containing both Arm code and Intel code. The location `/Library/TeX/texbin` is

a symbolic link which indirectly leads to this location. If you use a different front end to  $\text{\TeX}$ , you should configure it to find the  $\text{\TeX}$  binaries at `/Library/TeX/texbin`

The  $\text{\MacTeX}$  installer performs extra tasks which aren't done when you install with  $\text{\TeX}$  Live's standard script.  $\text{\MacTeX}$  extends `PATH` and `MANPATH` so  $\text{\TeX}$  binaries and man pages for the distribution are available within Unix shells.

If you were using an earlier  $\text{\TeX}$  distribution like  $\text{\TeX}$  Live 2020 before you installed  $\text{\MacTeX}$ -2021, that distribution remains unchanged on your machine after installing the new distribution. You can switch back and forth between  $\text{\TeX}$  Live 2021 and your old distribution by using the program  $\text{\TeX}$  Live Utility in `/Applications/TeX`. Run this program and select the menu item "Reconfigure Distributions..." in the Configure menu. A panel will appear listing all  $\text{\TeX}$  distributions on your machine. Click the radio button attached to one of these items to activate that distribution.

When you click that button, `PATH` and `MANPATH` and all front ends and utilities are automatically reconfigured. A consequence of the mechanism used to make the command work is that all of the  $\text{\TeX}$  front ends and utilities provided by  $\text{\MacTeX}$  are automatically configured for  $\text{\TeX}$ ; this applies to most current programs available over the web, not just those we supply with  $\text{\MacTeX}$ .

On other Unix machines it is common to set environment variables like `TEXINPUTS`. While  $\text{\TeX}$  Live recognizes these variables, it is usually not necessary or desirable to set them.

$\text{\MacTeX}$  tries to guess your paper size at installation time. If it makes a mistake, you can set the paper size by running the program  $\text{\TeX}$  Live Utility in `/Applications/TeX`, and selecting the "Change Paper Size..." item in the Action menu.

Use  $\text{\TeX}$  Live Utility to keep your distribution up to date. When the program starts, it lists packages in  $\text{\TeX}$  for which updates are available. Select the "Update All Packages" item in the Action menu to update these packages over the net.

For details about these features, read "What Is Installed" in `/Applications/TeX`.

The full  $\text{\TeX}$  Live is installed in a folder inside `/usr/local/texlive` named for the release date of the distribution. For example, moving the folder 2020 in this location to the trash will erase last year's distribution.

You may want to add style files, fonts, and other files to the  $\text{\TeX}$  distribution. This can be done in two ways. To install files available only for the current user, create a folder named `texmf` in `~/Library`, where this is the Library folder in your home directory rather than the system Library folder. Create a series of subfolders in this location which match the structure of the folder tree in  $\text{\TeX}$  Live. In practice, this means creating `~/Library/texmf/tex/latex`, `~/Library/texmf/bibtex/bib`, and `~/Library/texmf/bibtex/bst`. Put style files for  $\text{\LaTeX}$  in the first location, bib files in the second location, and bst files in the third location.

To install extra files for all users, put the files in `/usr/local/texlive/texmf-local`. This location and an associated folder structure have already been created.

It is easier to use the first location because the user owns this spot and dragging files in and out is easy. Moreover, files placed there will immediately be recognized by  $\text{\TeX}$ . The second location is harder to use because it is owned by root and dragging files in and out requires typing passwords again and again. Moreover, new files will not be recognized there without reindexing, so after adding or subtracting files to `/usr/local/texlive/texmf-local`, you must open Terminal

in `/Applications/Utilities` and execute the command `texhax`. Another name for this command is `mktexlsr`.

Adding new fonts to TeX is more complicated. See the link <http://tug.org/mactex/TeXLive2018+Changes.pdf>.

When  $\text{\LaTeX}$  searches for a style file, it first consults `~/Library/texmf` and then `/usr/local/texlive/texmf-local` before searching inside the TeX distribution. This means that if you insert a common style file in `~/Library/texmf`, it will be used rather than the similar files in the main distribution. If you later update the distribution, the old versions will be found first and the updated versions will not be used. This can cause serious typesetting problems when updating TeX Live. Therefore, the local `texmf` locations should only be used for special files not in TeX Live, like style files for particular journals or local thesis style files.