

# **Local L<sup>A</sup>T<sub>E</sub>X and You!**

A guide to setting up L<sup>A</sup>T<sub>E</sub>X on your device

Kaushik Rajendran

December 25, 2024

# Contents

<b>1</b>	<b>Introduction</b>	<b>2</b>
1.1	So why use a local version if Overleaf is this good? . . . . .	2
1.2	The purpose and scope of this writeup . . . . .	2
<b>2</b>	<b>Installation and Setup</b>	<b>3</b>
2.1	$\text{\LaTeX}$ installation . . . . .	3
2.2	Using $\text{\LaTeX}$ on VSCode . . . . .	3

# 1 Introduction

L<sup>A</sup>T<sub>E</sub>X is a high quality typesetting environment designed for creating scientific and technical documents. It is the standard for any of our publications or official scientific communication. Perhaps the easiest and most common way to use L<sup>A</sup>T<sub>E</sub>X is through Overleaf, an online L<sup>A</sup>T<sub>E</sub>X environment that includes a bunch of useful features such as the visual editor for first-time L<sup>A</sup>T<sub>E</sub>X users, and real-time collaboration.

## 1.1 So why use a local version if Overleaf is this good?

Obviously, Overleaf is amazing and I do not intend to disparage its usage here. But one of the biggest flaws with using Overleaf exclusively is the fact that it is an *online* editor. This means that you need to be constantly connected to the internet to use Overleaf to edit your documents. This means that in cases like power/server outages, or times when you're travelling without an internet connection (say a long flight on the way to a conference), you can't use Overleaf. In such cases, it is good to have a local option to work in L<sup>A</sup>T<sub>E</sub>X offline. Additionally, depending on the code editor you use to write your local L<sup>A</sup>T<sub>E</sub>X documents, you may have some nice integrations/shortcuts to exploit.

## 1.2 The purpose and scope of this writeup

This writeup is mainly intended for L<sup>A</sup>T<sub>E</sub>X users who are comfortable with using Overleaf for preparing documents. As such, this document will assume basic understanding of the L<sup>A</sup>T<sub>E</sub>X environment. The remainder of this writeup is split into two sections, namely

- **Installation and Setup:** Talking about how to install and use L<sup>A</sup>T<sub>E</sub>X locally, and providing some personalization/workflow tips
- **Integrations:** Exploring integrating local L<sup>A</sup>T<sub>E</sub>X with Overleaf and looking into integrating project code with writeups

## 2 Installation and Setup

### 2.1 $\text{\LaTeX}$ installation

The standard distribution of  $\text{\LaTeX}$  is TeX Live, which works on most platforms. Detailed instructions can be found here. Windows users may download TeX Live from <https://www.tug.org/texlive/windows.html#install>, Mac users may do so using the MacTeX installer from <https://www.tug.org/mactex/mactex-download.html>, and Linux users can download TeX Live from <https://www.tug.org/texlive/quickinstall.html>. This is technically the only thing you need to install to start using  $\text{\LaTeX}$  locally, since all of these will also install a front end to edit your  $\text{\LaTeX}$  documents, as well as install most commonly used packages (I have never needed to manually install packages so far).

You can also use your preferred code editing front end to edit your  $\text{\LaTeX}$  code. I prefer using VSCode for this purpose, and recommend it due to its easy to use  $\text{\LaTeX}$  extension and abundance of other useful tools. I will cover this in the following subsection

### 2.2 Using $\text{\LaTeX}$ on VSCode

The first step to this is obviously, downloading VSCode itself. You can do this at <https://code.visualstudio.com/download>. Once VSCode is downloaded, the LaTeX Workshop extension must be installed. This can be done by navigating to the Extensions tab on the sidebar and typing it into the search bar. This is shown below

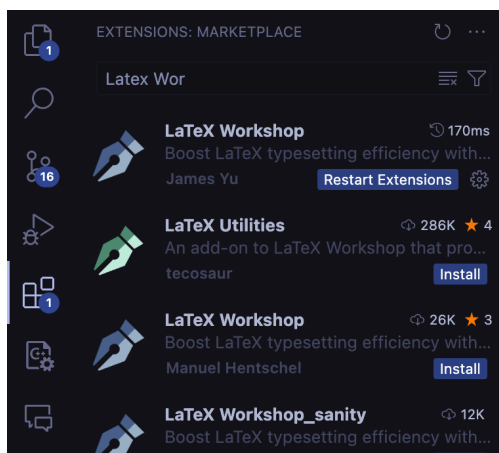


Figure 2.1: The extension you want to install is the topmost one on the search results (the one by James Yu), which I have already installed in this image.

From here, you can simply create TeX files in VSCode by ending a file with the `.tex` extension, and edit and compile them through the build button in the LaTeX Workshop sidebar tab, or the build button on top of the currently open TeX file. You can then use the LaTeX workshop sidebar once more to view the output PDF. Alternatively, you may find the output PDF in your file path and open it to the side in VSCode like you would a secondary tab of code. Nominally, this output PDF is in the same directory as your main TeX file. This is shown in Fig. 2.2

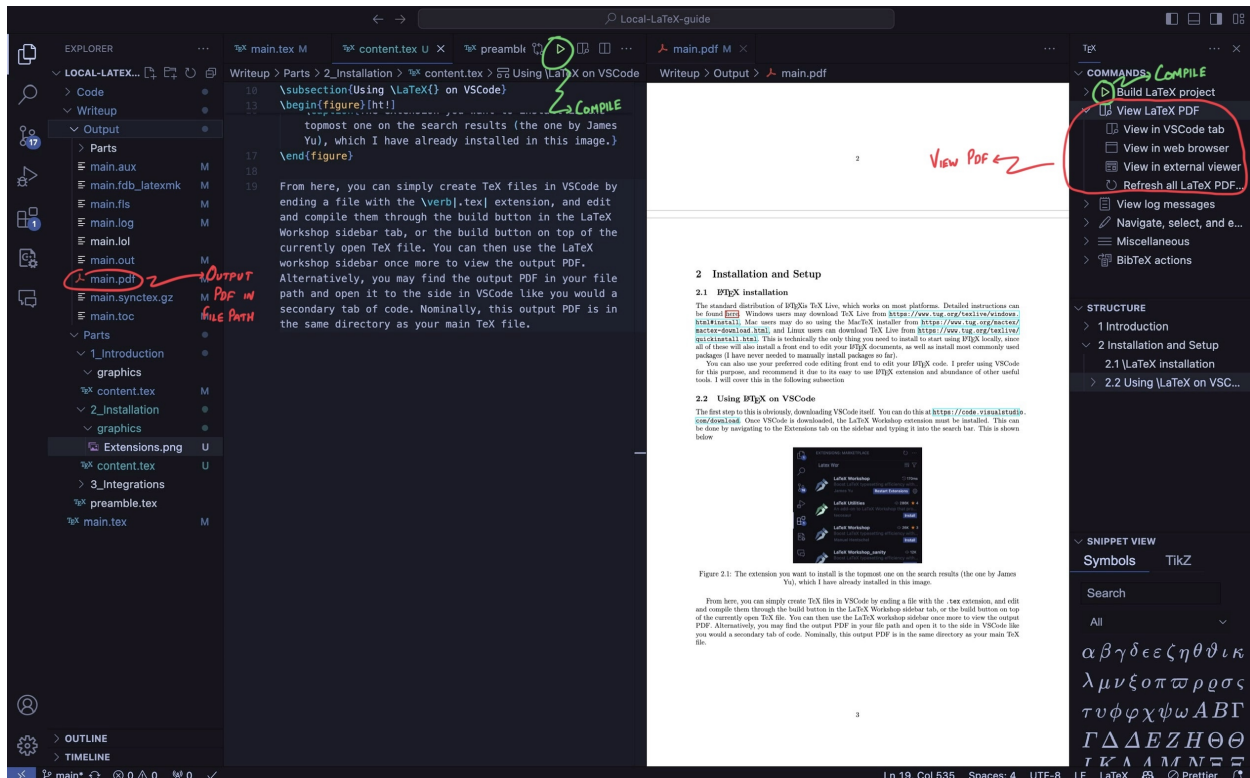


Figure 2.2: Where to find the output pdf and compile button in VSCode. Note that I moved the LaTeX workshop menu to the right sidebar in this picture, and have slightly modified the place where the output files go (more on this in a bit).