

PiE LaTeX Workshop

Notes and Reference Guide

May 2020

1 Introduction

This is a tutorial to go along with the workshop on \LaTeX hosted by the Program in Ecology. The goal of this workshop is three-fold:

1. Show you the benefits of using LaTeX
2. Teach you how to use LaTeX (the basics)
3. Give you a little taste of advanced features

This document will complement the material covered in the Zoom workshop, and hopefully will serve as a reference that you can refer back to in the future.



Fig. 1

LaTeX, basically.

Source: <https://twitter.com/hvyDesert>

Figure 1: A comic about how easy it is to produce nice-looking documents from \LaTeX .

2 Why should you use \LaTeX ?

- The content of the document is separate from the style
 - This means that it looks nice with very little effort from you
 - It is easy to quickly and (relatively) painlessly switch the appearance
 - This point is SUPER useful for submitting manuscripts to journals!
 - Many journals have templates for formatting, so you can just place your manuscript within this framework.
- You can easily generate bibliographies, especially if you already use a reference manager (e.g. Zotero, Mendeley)

- You can easily place images/figures (and they stay where you want them to!), and won't have to change references to those figures by hand when they need to be reordered in the document
- You can create dynamic documents, where figures/tables/numbers are updated when code is re-run, which eliminates copy-paste errors that are common when trying to track down every single instance of a number during revisions
- It is possible to embed PDF/SVG vector files (unlike Microsoft Word), which means that your graphics don't lose their quality
- \LaTeX is way better than anything else for mathematical formulas or symbols

3 How to use \LaTeX ?

While you can draft \LaTeX documents in any text editor, most people use a specific \LaTeX editor program. These can be either programs that you download to your computer, or online editors (or you can use both!), and offer some nice functionality beyond just using a text editor. Most of these \LaTeX editors have multiple panels, with one panel showing the document structure, one with the text and code that you're working with (the actual meat of the document), and then a preview pane to see what the rendered document actually looks like. If you're familiar with working in `Rstudio` and working with `RMarkdown` documents, then \LaTeX editors may feel pretty familiar. One thing to note: in \LaTeX , comments are demarcated with `%`, rather than the `#` common in other programming languages.

Online \LaTeX editors allow real-time collaboration, in the style of Google Docs. This feature can be super useful for working on projects with multiple authors, or projects where you want to get input from multiple people without having to email around a bunch of documents and try to keep versions straight. Most of the online editors also do automatic version control, so your document is backed up and checkpointed (again, ala Google Docs) along the way. One other thing that is great about working in \LaTeX is the ability to comment out chunks of text, which keeps those hard-fought words in the raw document, but removes them from the rendered document. This is incredibly useful when re-writing, as it allows you to play with writing things a different way without having to totally remove the old text.

3.1 Working from Templates

The easiest way to start in \LaTeX is to work from a template, rather than trying to build your own document from scratch. I would recommend checking out the [Overleaf template gallery](#) or [my \$\LaTeX\$ templates](#)—I've created some for general types of useful documents: a general writing document, a thesis/dissertation proposal, a "lab notebook" document, and an academic CV. Once you have a template, you can modify the document in whatever makes sense for what you're doing. Just as with other types of coding, this often involves Googling "latex [insert thing I want to do here]" and searching through StackOverflow to figure out how to do what you want to do.

3.2 Project Structure

Within the main directory for your \LaTeX project, you will need to have a main control file, at a minimum. You also are likely to have a bibliography file (`.bib`) with your citations for the project, any figures that you want to include, and a style file (`.cls`), if you're working with a specific format, such as a journal's own style. You can also create directories within the main project directory, which can be useful for keeping a larger project organized. For example, a manuscript may have a folder for `figures/`, `tables/`, or `code/` (if you are integrating your analyses from R into your document). Organization within a project is generally based on own personal preference (e.g. I prefer to have most things within well-named directories)—however, some journals have a preference on how things are organized, so check their recommendations when you're preparing a manuscript to submit.

3.3 The format of your .tex file

Documents will differ based on what kind of a project you are working on, but the main file will need to have some basic elements. The beginning of your document will likely look something like this:

```
\documentclass{article}
\usepackage[utf8]{inputenc}
\usepackage{natbib}

\title{PiE LaTeX Workshop}
\author{Notes and Tutorials}
\date{May 2020}

\begin{document}

\maketitle
```

These commands tell L^AT_EX what your document should look like, and gives it some metadata about your document. The first line (`\documentclass{}`) dictates what you are writing—in most cases, you’ll likely be working with the document class `article`. After that, there are some packages called that will be used in the document (here, package `inputenc` and `natbib`). Just like R, L^AT_EX has thousands of packages available with different features, and the ones you want to use need to be specified using these `\usepackage{}` commands. Following this, we can specify our title, author, and date, which are then written in the document when we tell it to `\maketitle`. Beyond these commands at the beginning of the document, every document **must** start with `\begin{document}` and end with `\end{document}`.

3.4 Environments within your document

Once you’ve gotten the header information out of the way, you can write your text just like you normally would. In L^AT_EX, anything that needs to be something other than “normal” text will be contained within an environment, and these environments are specified using the “\” incantations. As some examples:

```
\textbf{} -- bolds the text inside the {}
\textit{} -- italicizes the text
\href{}{} -- specifies a link (when using the hyperref package)
\section{} \subsection{} \subsubsection{} etc. -- specify headings
\cite{} \citep{} \citealt{} etc. -- adds a citation
\ref{} -- allows you to refer to an element in your document (e.g. figure or section)

$ math $ -- specifies an equation environment, where the text between the dollar signs
will be written using the equation editor
```

There are many more, but these are just some common examples that you may need. In addition to these shorthand forms, some environments will be specified using the `\begin{}` and `\end{}` commands, with the environment type specified between the brackets. For example, we can create a list environment using:

```
\begin{itemize}
  \item List item 1
  \item List item 2
  \item A third thing on our list
\end{itemize}
```

With this code, we would produce a list that looks like this:

- List item 1
- List item 2

- A third thing on our list

We could also change this to a numbered list by changing our `\begin{itemize}` to `\begin{enumerate}` (and changing the `\end{}` call as well, of course).

3.5 Bibliographies and references

Bibliographies can be imported into your \LaTeX document in `Bibtex` format as `.bib`. To do this, you can export a `.bib` file from your reference manager, and then add that document to your \LaTeX project. Alternatively, if you have a premium account on Overleaf, you can link your project to your Mendeley or Zotero account, and your bibliography will be automatically updated whenever you update entries in your reference manager.

For bibliographies, I like using the `natbib` package (just add `\usepackage{natbib}` at the beginning of your document). If you have a specific citation style that you want to use (e.g. a `.bst` file from a journal), you'll import that into the project, and then specify that format by typing `\bibliographystyle{stylename}` after calling the package. If you don't have a specific format, there are several built-in styles, which you can find in the documentation for both `Bibtex` and the `natbib` package (e.g. see the list on CTAN at <https://ctan.org/topic/bibtex-sty?lang=en>). To create your bibliography, you'll simply add the command `\bibliography{.bib filename}` where you want it to go (likely just before your `\end{document}` command).

To add in-text citations, you will use the commands `\citep{citationkey}` for parenthetical citations (Macarthur, 1965), `\cite{citationkey}` for textual author-year citations like Macarthur (1965), or `\citealt{citationkey}` for a citation with no parentheses like Macarthur 1965. There are also the options `\citeauthor{citationkey}` and `\citeyear{citationkey}` to print only the author or year, respectively. To add two citations, you just add both citation keys within the command: `\citep{citationkey1,citationkey2}`, which will produce both citations inside of parentheses. (Cohen et al., 2016; Danley et al., 2012). Just like if you use a reference manager with Microsoft Word, any citation added in-text will automatically have its reference added to the bibliography.

3.6 Figures and tables

We won't have time to get into all of the intricacies of inserting figures and tables into your \LaTeX document, but here's a bit of an introduction to the topic. \LaTeX (and specifically Overleaf) can handle many different file types, including `.pdf` and `.svg` vector images, which means that your images will be high-quality even when magnified. You'll need to import your images into your \LaTeX project (perhaps into a `figures/` directory, if you're following the advice above), and then you can insert your figure into your document using a figure environment, such as in the following code:

```
\begin{figure}[hbt]
\centering
\includegraphics[width=0.25\textwidth]{latex.png}
\caption{A comic about how easy it is to produce nice-looking documents from
\LaTeX.}\label{fig:latex}
\end{figure}
```

The `[hbt]` tells \LaTeX that I want the figure to be placed *here*, *bottom*, or *top*, in order of preference. I also tell it that I would like the figure to be 25% of the `\textwidth`. After that, I include my caption and what I want the figure to be labeled—this label allows me to refer to this figure in-text using the command `\ref{fig:latex}`. This way, the figure numbers in my document are dynamic (i.e. they will update automatically if I reorder my figures), and therefore I won't have to spend hours re-checking my document to make sure that all of the numbers match up.

Tables are similar, but with slightly more complicated code. A simple, easy table can be created with the following code:

```
\begin{table}[h]
```

```

\centering
  \caption{Your first table.}
  \label{tab:table1}
  \begin{tabular}{l|c|r}
    \textbf{Value 1} & \textbf{Value 2} & \textbf{Value 3} \\
    \hline
    1 & 1110.1 & a \\
    2 & 10.1 & b \\
    3 & 23.113231 & c \\
  \end{tabular}
\end{table}

```

This produces a table that looks like Table 1. Here, the & symbols separate out columns, and the double backslashes indicate the ends of rows. On the same line as `\begin{tabular}`, we specify `{l|c|r}`, which means that we want three columns, with the first left-justified, the second center-justified, and the third right-justified. It is relatively easy to hand-code small tables, but for larger ones, I prefer to use the [LaTeX Table Generator](#), where you can either paste your table or import a `.csv` document with your data, and visually choose what you want your table to look like. When using this, I often choose the "booktabs style" for my tables, which makes them look slightly slicker. As an example, Table 2 was created using the LaTeX Table Generator.

Table 1: Your first table.

Value 1	Value 2	Value 3
1	1110.1	a
2	10.1	b
3	23.113231	c

Table 2: Your second table, in Booktabs style.

Value 1	Value 2	Value 3
1	1110.1	a
2	10.1	b
3	23.113231	c

3.7 Bonus: example R Integration

You can also include R code/calculations/plots within your documents through integration with `knitr`. To do this, you just have to change the `.tex` extension on your document to `.Rtex`, and then you can write R chunks beginning with `<<chunk1>>=` and ending with an `@`. When you do this, your code will appear as in the example below:

```

# Create a sequence of numbers
X = 2:10

# Display basic statistical measures
summary(X)

##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##         2         4         6         6         8         10

```

So, the mean of the data then is 6 (I specified that by typing `$$\Sexpr{mean(x)}$$`, which was then evaluated by `LATEX` and `knitr`. This chunk of code can also be hidden by using the `echo=FALSE` command

in the chunk attributes, just as we would in an RMarkdown document. We can also create plots using R code, such as Fig. 2 (this number was generated using `\ref{fig:plot1}`, rather than hand-coded as "Fig. 2").

```
xdata = read.csv(file="data.txt", head=TRUE, sep=" ")  
  
hist(xdata$data, main="ShareLaTeX histogram", xlab="Data")
```

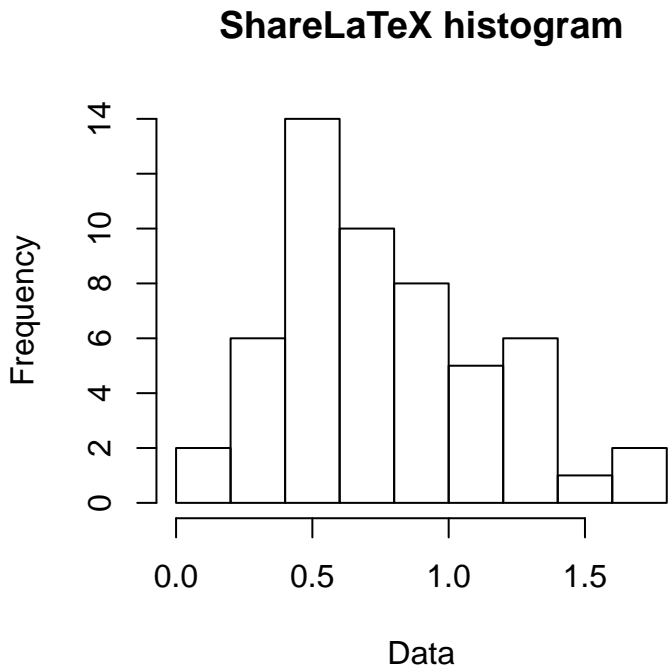


Figure 2: Histogram created from our R code

This is a super simple example, just to show the functionality and give you a little taste of what can be done; if you are interested in exploring this more, there is a more detailed tutorial from Overleaf at <https://www.overleaf.com/learn/latex/Knitr>. For more advanced functionality for combining RMarkdown and Overleaf documents, you can [check out this tutorial on my GitHub](#).

4 Additional sources for further information

Obviously, this is only the tip of the L^AT_EX iceberg. There are lots of further resources out there, depending on the direction you need to go with your L^AT_EX-ing. I have appended some quick guides to this tutorial; here are some more to get you started:

- CTAN (<https://www.ctan.org/>) – the main source for L^AT_EX packages
- L^AT_EX Project (<https://www.latex-project.org/>) – the source for downloading L^AT_EX onto your computer, with additional information about recommended L^AT_EX editors
- Overleaf’s Tutorials and Cheat Sheets (<https://www.overleaf.com/learn/latex/Tutorials>) – a collection of tutorials to get you working in L^AT_EX (and on Overleaf), from beginner to advanced. New guides are being added all the time by the Overleaf community
- detexify (<http://detexify.kirelabs.org>) – a website for looking up the codes for symbols in L^AT_EX

A quick guide to L^AT_EX

What is L^AT_EX?

L^AT_EX (usually pronounced “LAY tek,” sometimes “LAH tek,” and never “LAY tek”) is a *format*, or collection of macro commands, for T_EX, the standard for most professional mathematics and scientific writing. T_EX is a powerful typesetting engine created by Donald Knuth of Stanford University (his first version appeared in 1978). Leslie Lamport was responsible for creating L^AT_EX, a popular set of user commands for T_EX. A team of L^AT_EX programmers created the current version, L^AT_EX 2_ε.

Mathematics

Math vs. text vs. functions

In properly typeset mathematics, the variables appear in italics (for example, $f(x) = x^2 + 2x - 3$). The exception to this rule is predefined functions (for example, $\sin(x)$). Thus it is important to *always* treat text, variables, and functions correctly. See the difference between x and x , -1 and -1 , and $\sin(x)$ and $\sin(x)$. There are two ways to present a mathematical expression – *inline* or as a *display*.

Inline mathematical expressions

Inline math expressions occur as part of the normal flow of text. To produce an inline expression, place the math expression between dollar signs ($\$$). For example, typing $\$90^\circ(\text{circ})\$$ is the same as $\$\frac{\pi}{2}\$$ radians yields 90° is the same as $\frac{\pi}{2}$ radians.

Displayed mathematical expressions

Displays are mathematical expressions that are given their own line and are centered on the page. These are usually used for important equations that deserve to be showcased on their own line or for large equations that cannot fit inline. To produce displayed mathematics, place the mathematical expression between the symbols $[\$ and $\backslash]$. Typing $\backslashx=\backfrac{-b\pm\sqrt{b^2-4ac}}{2a}\backslash$ yields

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

Display style

To get full-size inline math, use \backdisplaystyle . Use this sparingly. Typing this $\backdisplaystyle \backsum_{n=1}^{\infty}\backfrac{1}{n}\$$, and not this $\$\backsum_{n=1}^{\infty}\backfrac{1}{n}\$$ yields this $\sum_{n=1}^{\infty} \frac{1}{n}$, and not this $\sum_{n=1}^{\infty} \frac{1}{n}$.

Images

You can put images (pdf, png, jpg, or gif) in your document. They need to be in the same location as your .tex file when you compile the document. Omit $[\text{width}=.5in]$ if you want the image to be full-sized.

```
\begin{figure}[tbp]
\includegraphics[width=.5in]{imagename.jpg}
\caption{The (optional) caption goes here.}
\end{figure}
```

Text decorations

Your text can be *italic* (\backtextit{italic}), **bold** (\backtextbf{bold}), or underlined ($\backunderline{underlined}$).

Your math can contain bold, \mathbb{R} (blackboard bold, \backmathbb{R}), integers (\mathbb{Z} or \mathbb{Z}), rational numbers (\mathbb{Q} or \mathbb{Q}), and natural numbers (\mathbb{N} or \mathbb{N}).

For text appearing inside a math expression, use \backtext . $(0,1]=\{x\in\mathbb{R}:x>0\}$ and $x\leq 1$ yields $(0,1] = \{x \in \mathbb{R} : x > 0 \text{ and } x \leq 1\}$. (Without the \backtext command it treats “and” as three variables: $(0,1] = \{x \in \mathbb{R} : x > 0 \text{ and } x \leq 1\}$.)

Spaces and new lines

L^AT_EX ignores extra spaces and new lines. For example, This sentence will look fine after it is compiled.

Leave one full empty line between two paragraphs. Place \back at the end of a line to create a new line (but not create a new paragraph).

This compiles

```
like\
this.
```

This compiles like this. Use \backnoindent to prevent a paragraph from indenting.

Comments

Use $\%$ to create a comment. Nothing on the line after the $\%$ will be typeset. $\back(x)=\backsin(x)\%$ this is the sine function yields $f(x) = \sin(x)$.

Delimiters

description	command	output
parentheses	(x)	(x)
square brackets	$[x]$	$[x]$
curly braces	$\{x\}$	$\{x\}$

To automatically make delimiters large enough to fit the content, use them together with \backright and \backleft . For example, $\backleft\backfrac{1}{n}\backright$ produces

$$\left\{ \sin\left(\frac{1}{n}\right) \right\}_n$$

Curly braces are non-printing characters that are used to gather text that has more than one character. Observe the differences between the four expressions x^2 , x^{*2} , x^{*2t} , x^{*2t} when typeset: x^2 , x^{*2} , x^{*2t} , x^{*2t} .

Lists

You can produce ordered and unordered lists.

description	command	output
unordered list	$\backbegin{itemize}$ $\backitem Thing 1$ $\backitem Thing 2$ $\backend{itemize}$	<ul style="list-style-type: none"> Thing 1 Thing 2
ordered list	$\backbegin{enumerate}$ $\backitem Thing 1$ $\backitem Thing 2$ $\backend{enumerate}$	<ol style="list-style-type: none"> Thing 1 Thing 2

Symbols (in math mode)

description	command	output
addition	$+$	$+$
subtraction	$-$	$-$
plus or minus	\pm	\pm
multiplication (times)	\times	\times
multiplication (dot)	\cdot	\cdot
division symbol	\div	\div
division (slash)	$/$	$/$
circle plus	\oplus	\oplus
circle times	\otimes	\otimes
equal	$=$	$=$
not equal	\neq	\neq
less than	$<$	$<$
greater than	$>$	$>$
less than or equal to	\leq	\leq
greater than or equal to	\geq	\geq
approximately equal to	\approx	\approx
infinity	∞	∞
dots	\dots	\dots
dots	\ldots	\dots
fraction	$\frac{a}{b}$	$\frac{a}{b}$
square root	\sqrt{x}	\sqrt{x}
n th root	$\sqrt[n]{x}$	$\sqrt[n]{x}$
exponentiation	a^b	a^b
subscript	a_b	a_b
absolute value	$ x $	$ x $
natural log	$\ln(x)$	$\ln(x)$
logarithms	$\log_a(b)$	$\log_a b$
exponential function	$e^x = \exp(x)$	$e^x = \exp(x)$
degree	$\deg(f)$	$\deg(f)$



Functions

description	command	output
maps to composition	<code>\to</code> <code>\circ</code>	\rightarrow \circ
piecewise function	<code> x =\begin{cases} x & x \ge 0 \\ -x & x < 0 \end{cases}</code> <code>x & x \ge 0</code> <code>-x & x < 0</code> <code>\end{cases}</code>	$ x = \begin{cases} x & x \geq 0 \\ -x & x < 0 \end{cases}$

Greek and Hebrew letters

command	output	command	output
<code>\alpha</code>	α	<code>\tau</code>	τ
<code>\beta</code>	β	<code>\theta</code>	θ
<code>\chi</code>	χ	<code>\upsilon</code>	υ
<code>\delta</code>	δ	<code>\xi</code>	ξ
<code>\epsilon</code>	ϵ	<code>\zeta</code>	ζ
<code>\varepsilon</code>	ε	<code>\Delta</code>	Δ
<code>\eta</code>	η	<code>\Gamma</code>	Γ
<code>\gamma</code>	γ	<code>\Lambda</code>	Λ
<code>\iota</code>	ι	<code>\Omega</code>	Ω
<code>\kappa</code>	κ	<code>\Phi</code>	Φ
<code>\lambda</code>	λ	<code>\Pi</code>	Π
<code>\mu</code>	μ	<code>\Psi</code>	Ψ
<code>\nu</code>	ν	<code>\Sigma</code>	Σ
<code>\omega</code>	ω	<code>\Theta</code>	Θ
<code>\phi</code>	ϕ	<code>\Upsilon</code>	Υ
<code>\varphi</code>	φ	<code>\Xi</code>	Ξ
<code>\pi</code>	π	<code>\N</code>	\mathbb{N}
<code>\psi</code>	ψ	<code>\beth</code>	\beth
<code>\rho</code>	ρ	<code>\daleth</code>	\daleth
<code>\sigma</code>	σ	<code>\gimel</code>	\gimel

Set theory

description	command	output
set brackets	<code>\{1,2,3\}</code>	$\{1,2,3\}$
element of	<code>\in</code>	\in
not an element of	<code>\notin</code>	\notin
subset of	<code>\subset</code>	\subset
subset of	<code>\subseteq</code>	\subseteq
not a subset of	<code>\not\subset</code>	$\not\subset$
contains	<code>\supset</code>	\supset
contains	<code>\supseteq</code>	\supseteq
union	<code>\cup</code>	\cup
intersection	<code>\cap</code>	\cap
big union	<code>\bigcup_{i=1}^{10} A_i</code>	$\bigcup_{i=1}^{10} A_i$
big intersection	<code>\bigcap_{i=1}^{10} A_i</code>	$\bigcap_{i=1}^{10} A_i$
empty set	<code>\emptyset</code>	\emptyset
power set	<code>\mathcal{P}</code>	\mathcal{P}
minimum	<code>\min</code>	\min
maximum	<code>\max</code>	\max
supremum	<code>\sup</code>	\sup
infimum	<code>\inf</code>	\inf
limit superior	<code>\limsup</code>	\limsup
limit inferior	<code>\liminf</code>	\liminf
closure	<code>\overline{A}</code>	\overline{A}

Calculus

description	command	output
derivative	<code>\frac{df}{dx}</code>	$\frac{df}{dx}$
derivative	<code>f'</code>	f'
partial derivative	<code>\frac{\partial f}{\partial x}</code>	$\frac{\partial f}{\partial x}$
integral	<code>\int</code>	\int
double integral	<code>\iint</code>	\iint
triple integral	<code>\iiint</code>	\iiint
limits	<code>\lim_{x \rightarrow \infty}</code>	$\lim_{x \rightarrow \infty}$
summation	<code>\sum_{n=1}^{\infty} a_n</code>	$\sum_{n=1}^{\infty} a_n$
product	<code>\prod_{n=1}^{\infty} a_n</code>	$\prod_{n=1}^{\infty} a_n$

Logic

description	command	output
not	<code>\sim</code>	\sim
and	<code>\wedge</code>	\wedge
or	<code>\vee</code>	\vee
if...then	<code>\to</code>	\rightarrow
if and only if	<code>\leftrightarrow</code>	\leftrightarrow
logical equivalence	<code>\equiv</code>	\equiv
therefore	<code>\therefore</code>	\therefore
there exists	<code>\exists</code>	\exists
for all	<code>\forall</code>	\forall
implies	<code>\Rightarrow</code>	\Rightarrow
equivalent	<code>\Leftrightarrow</code>	\Leftrightarrow

Linear algebra

description	command	output
vector	<code>\vec{v}</code>	\vec{v}
vector	<code>\mathbf{v}</code>	\mathbf{v}
norm	<code>\ \vec{v}\ </code>	$\ \vec{v}\ $
matrix	<code>\begin{array}{ccc} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 0 \end{array}</code>	$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 0 \end{bmatrix}$
determinant	<code>\begin{array}{ccc} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 0 \end{array}</code>	$\begin{vmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 0 \end{vmatrix}$
determinant	<code>\det(A)</code>	$\det(A)$
trace	<code>\operatorname{tr}(A)</code>	$\operatorname{tr}(A)$
dimension	<code>\dim(V)</code>	$\dim(V)$

Number theory

description	command	output
divides	<code>\mid</code>	\mid
does not divide	<code>\nmid</code>	\nmid
div	<code>\operatorname{div}</code>	div
greatest common divisor	<code>\gcd</code>	\gcd
ceiling	<code>\lceil x \rceil</code>	$\lceil x \rceil$
floor	<code>\lfloor x \rfloor</code>	$\lfloor x \rfloor$

Geometry and trigonometry

description	command	output
angle	<code>\angle ABC</code>	$\angle ABC$
degree	<code>90^\circ</code>	90°
triangle	<code>\triangle ABC</code>	$\triangle ABC$
segment	<code>\overline{AB}</code>	\overline{AB}
sine	<code>\sin</code>	\sin
cosine	<code>\cos</code>	\cos
tangent	<code>\tan</code>	\tan
cotangent	<code>\cot</code>	\cot
secant	<code>\sec</code>	\sec
cosecant	<code>\csc</code>	\csc
inverse sine	<code>\arcsin</code>	\arcsin
inverse cosine	<code>\arccos</code>	\arccos
inverse tangent	<code>\arctan</code>	\arctan

Symbols (in text mode)

These symbols do *not* have to be surrounded by dollar signs.

description	command	output
dollar sign	<code>\\$</code>	$\$$
percent	<code>\%</code>	$\%$
ampersand	<code>\&</code>	$\&$
pound	<code>\#</code>	$\#$
backslash	<code>\textbackslash</code>	\backslash
left quote marks	<code>\‘</code>	$\‘$
right quote marks	<code>\’</code>	$\’$
single left quote	<code>\`</code>	$\`$
single right quote	<code>\`</code>	$\`$
hyphen	X-ray	X-ray
en-dash	pp. 5--15	pp. 5--15
em-dash	Yes---or no?	Yes---or no?

Getting started with Overleaf and L^AT_EX

L^AT_EX collaborative authoring online: <https://overleaf.com>
 Overleaf L^AT_EX documentation: <https://overleaf.com/learn>
 Learn L^AT_EX in 30 minutes: https://overleaf.com/Learn/LaTeX_in_30_minutes

Comprehensive T_EX Archive Network: <http://www.ctan.org/>
 T_EX Users Group: <http://www.tug.org/>

Want to work offline? Local install options for Linux or Windows:
 T_EX Live; MacOS: MacT_EX; Windows: MiK_TE_X

Based on the version by Dave Richeson, Dickinson College,
<http://divisbyzero.com/>. Reproduced with permission.

Overleaf Keyboard Shortcuts

Updated March 12, 2020

Some keyboard shortcuts in this list may not be available on non-US keyboards or in Rich Text/vim/emacs modes. For example, some input methods on the Mac may use `⇧` (equivalent of `Alt`) for accented characters input instead.

Mac users may also want to refer to this list of common editor keyboard shortcuts on Mac: <https://support.apple.com/en-us/HT201236#text>

Windows GNU/Linux	Mac	Description
<code>F1</code>	<code>F1</code>	Show list of shortcuts. If a shortcut isn't available on your keyboard or input method, you might trigger it from this list instead.
<code>Ctrl + .</code> <code>Ctrl + S</code> <code>Ctrl + Enter</code> (Vim) :w	<code>ctrl + .</code> <code>⌘ + S</code> <code>⌘ + Enter</code> (Vim) :w	Trigger compilation
<code>Alt + E</code>	<code>Alt + E</code>	Next error/warning
<code>Alt + ⇧ + E</code>	<code>Alt + ⇧ + E</code>	Previous error/warning
<code>Ctrl + /</code> <code>Ctrl + ⇧ + 7</code>	<code>⌘ + /</code> <code>⌘ + ⇧ + 7</code>	Toggle % comment on selected line(s) some non-US keyboard layouts
<code>F2</code>	<code>F2</code>	Toggle fold current environment
<code>Alt + 0</code>	<code>⇧ + ⌘ + 0</code>	Fold other environments
<code>Alt + ⇧ + 0</code>	<code>⇧ + ⌘ + ⇧ + 0</code>	Unfold all
<code>Ctrl + D</code>	<code>⌘ + D</code>	Delete current line
<code>Ctrl + Space</code>	<code>ctrl + Space</code>	Open auto-complete. Use <code>⇧</code> <code>↓</code> <code>Tab</code> to select candidate, and <code>Enter</code> to insert.
<code>Ctrl + I</code> <code>⇧ + Tab</code>	<code>ctrl + I</code> <code>⇧ + Tab</code>	Indent less

(continued next page) ➡

➡ (from previous page)

Windows GNU/Linux	Mac	Description
<code>Ctrl + J</code> <code>Tab</code> (at line start)	<code>ctrl + J</code> <code>Tab</code> (at line start)	Indent more
<code>Ctrl + U</code>	<code>ctrl + U</code>	To uppercase
<code>Ctrl + ⇧ + U</code>	<code>ctrl + ⇧ + U</code>	To lowercase
<code>Ctrl + B</code>	<code>⌘ + B</code>	Bold text (<code>\textbf</code>)
<code>Ctrl + I</code>	<code>⌘ + I</code>	Italicise text (<code>\textit</code>)
<code>Ctrl + X</code>	<code>⌘ + X</code>	Cut selected text
<code>Ctrl + C</code>	<code>⌘ + C</code>	Copy selected text
<code>Ctrl + V</code>	<code>⌘ + V</code>	Paste copied text
<code>Ctrl + ⇧ + D</code>	<code>⌘ + ⇧ + D</code>	Duplicate selected text
<code>Alt + ⇧ + ↑</code>	<code>⇧ + ⌘ + ↑</code>	Copy lines up
<code>Alt + ⇧ + ↓</code>	<code>⇧ + ⌘ + ↓</code>	Copy lines down
<code>Alt + ↑</code>	<code>⇧ + ↑</code>	Move lines up
<code>Alt + ↓</code>	<code>⇧ + ↓</code>	Move lines down
<code>Ctrl + Alt + S</code>	<code>Ctrl + Alt + S</code>	Sort selected lines
<code>Ctrl + Z</code>	<code>⌘ + Z</code>	Undo
<code>Ctrl + ⇧ + Z</code> <code>Ctrl + Y</code>	<code>⌘ + ⇧ + Z</code> <code>⌘ + Y</code>	Redo
<code>Ctrl + P</code>	<code>ctrl + P</code>	Go to nearest matching start/close parenthesis, brace or bracket <code>{}[]</code>
<code>Ctrl + ⇧ + P</code>	<code>ctrl + ⇧ + P</code>	Select to nearest matching close parenthesis, brace or bracket <code>{}[]</code>
<code>Ctrl + ⇧ + M</code>	<code>ctrl + ⇧ + M</code>	Expand selection to current nearest matching start/close parenthesis, brace or bracket <code>{}[]</code>
<code>Alt + ←</code>	<code>ctrl + ←</code>	Go to line start

(continued next page) ➡

🔍 (from previous page)

Windows GNU/Linux	Mac	Description
Alt + →	ctrl + E	Go to line end
Ctrl + Home	⌘ + ↑	Go to document start
Ctrl + End	⌘ + ↓	Go to document end
Ctrl + L	⌘ + L	Go to line number...
Ctrl + F , Ctrl + H (emacs) ⌘ + ⌥ + Alt + 5	⌘ + F ⌘ + ⌥ + 5	Find and replace
Ctrl + K	⌘ + G	Find next (or just Enter again with search dialog open)
Ctrl + ⌈ + K	⌘ + ⌈ + G	Find previous
Ctrl + A	⌘ + A	Select all
Ctrl + click	⌘ + click	Multiple cursors à la Sublime Text
Ctrl + Alt + ↑	ctrl + ⌥ + ↑	Add cursor above (cumulative)
Ctrl + Alt + ⌈ + ↑	ctrl + ⌥ + ⌈ + ↑	Add cursor above (remove previously added)
Ctrl + Alt + ↓	ctrl + ⌥ + ↓	Add cursor below (cumulative)
Ctrl + Alt + ⌋ + ↓	ctrl + ⌥ + ⌋ + ↓	Add cursor below (remove previously added)
Alt + drag	⌥ + drag	Rectangular selection à la Sublime Text
Ctrl + J	⌘ + J	Toggle review panel
Ctrl + ⌈ + A	⌘ + ⌈ + A	Toggle track changes
Ctrl + ⌈ + C	⌘ + ⌈ + C	Add comment in review panel

\leq	<code>\leq</code>	\propto	<code>\propto</code>	$\$$	<code>\\$</code>	α	<code>\alpha</code>	∞	<code>\infty</code>
\geq	<code>\geq</code>	\cdots	<code>\cdots</code>	$\&$	<code>\&</code>	β	<code>\beta</code>	f'	<code>f\prime</code>
\neq	<code>\neq</code>	\dots	<code>\dots</code>	$\%$	<code>\%</code>	ϵ	<code>\epsilon</code>	\int	<code>\int</code>
\nleq	<code>\nleq</code>	\because	<code>\because</code>	\backslash	<code>\backslash</code>	ζ	<code>\zeta</code>	\oint	<code>\oint</code>
\ngeq	<code>\ngeq</code>	\therefore	<code>\therefore</code>	$\#$	<code>\sharp</code>	η	<code>\eta</code>	\mathbb{Z}	<code>\mathbb{Z}</code>
\cong	<code>\cong</code>	\forall	<code>\forall</code>	∂	<code>\partial</code>	κ	<code>\kappa</code>	\mathbb{R}	<code>\mathbb{R}</code>
\equiv	<code>\equiv</code>	\exists	<code>\exists</code>	90°	<code>90^\circ</code>	λ	<code>\lambda</code>	\mathbb{Q}	<code>\mathbb{Q}</code>
\sim	<code>\sim</code>	\in	<code>\in</code>	\parallel	<code>\parallel</code>	μ	<code>\mu</code>	$\sqrt[3]{2}$	<code>\sqrt[3]{2}</code>
\approx	<code>\approx</code>	\subset	<code>\subset</code>	\perp	<code>\perp</code>	ξ	<code>\xi</code>	$\frac{2}{3}$	<code>\frac{2}{3}</code>
\doteqdot	<code>\doteqdot</code>	\subseteq	<code>\subseteq</code>	\triangle	<code>\triangle</code>	ρ	<code>\rho</code>	$\lceil x \rceil$	<code>\lceil x \rceil</code>
\times	<code>\times</code>	\varnothing	<code>\varnothing</code>	∇	<code>\nabla</code>	τ	<code>\tau</code>	$\lfloor x \rfloor$	<code>\lfloor x \rfloor</code>
\cdot	<code>\cdot</code>	\cap	<code>\cap</code>	\square	<code>\square</code>	ϕ	<code>\phi</code>	$\{x\}$	<code>\{x\}</code>
$*$	<code>\ast</code>	\cup	<code>\cup</code>	\angle	<code>\angle</code>	ψ	<code>\psi</code>	\widehat{p}	<code>\widehat{p}</code>
\div	<code>\div</code>	\setminus	<code>\setminus</code>	Π	<code>\Pi</code>	π	<code>\pi</code>	\overline{AB}	<code>\overline{AB}</code>
\pm	<code>\pm</code>	\wedge	<code>\wedge</code>	Θ	<code>\Theta</code>	θ	<code>\theta</code>	\overleftrightarrow{AB}	<code>\overleftrightarrow{AB}</code>
\mp	<code>\mp</code>	\vee	<code>\vee</code>	Γ	<code>\Gamma</code>	γ	<code>\gamma</code>	\overleftarrow{AB}	<code>\overleftarrow{AB}</code>
\bigcirc	<code>\bigcirc</code>	\Rightarrow	<code>\Rightarrow</code>	Δ	<code>\Delta</code>	δ	<code>\delta</code>	\overfrown{AB}	<code>\overfrown{AB}</code>
\oplus	<code>\oplus</code>	\rightarrow	<code>\rightarrow</code>	Ω	<code>\Omega</code>	ω	<code>\omega</code>	\parallel	<code>\parallel</code>
\otimes	<code>\otimes</code>	\mapsto	<code>\mapsto</code>	Σ	<code>\Sigma</code>	σ	<code>\sigma</code>	\overrightarrow{AB}	<code>\overrightarrow{AB}</code>