

B Commands and errors

Most people these days do their L^AT_EXing in a graphical windowing editor with menus, running in a modern operating system that uses windows, icons, fonts, and a mouse which moves a pointer. This probably works fine 95% of the time, when you're dealing with one or two documents at a time, and everything you want to do is accessible through the menus, and you explicitly *don't* want to see L^AT_EX spilling its guts all over the place every time it reformats the document. Click here, move to there, cut, move somewhere else, paste, edit the text, write some more, click `Typeset` and you're done.

However, life isn't always that easy. Sometimes things go wrong, and you need to open up the lid and find out what it was. This appendix is a short description of how to run L^AT_EX manually, via the command-line, instead of through your editor, and it also covers error messages, and a few internal details about viewing and printing.

The editor wasn't always the primary interface to T_EX, except for actually writing and editing the document. Before editors with built-in L^AT_EX controls became available, you had to leave your editor — or at least go to another window — and type a command to process your document, then another to view it or print it. For a small but significant number of people, running L^AT_EX this way is still the order of the day.

- ☐ Maybe they're working on a remote mainframe or super-computer console with no graphics, just a 3270 or VT-100 terminal like those in [Figure B.1](#) on the next page;
- ☐ They might be using a smartphone where the editing facilities are limited and the scope for full menus entirely absent;
- ☐ Perhaps they are simply uninterested in all the bells and whistles of the modern interface, with too many menus doing things they can actually do faster typing instructions by hand;
- ☐ Possibly they're using automation facilities that most \LaTeX editors don't have, like the ability to apply the same edit to thousands of documents while you go and have a coffee or get on with something else;
- ☐ Or perhaps they are writing a system where \LaTeX is the embedded typesetter, so they're actually working in a completely different scripting or programming language which does a lot of other things before calling on \LaTeX behind the scenes to do some typesetting.

Before I go any further I'm going to assume at this stage that you have typed a document (for example [Figure 1.2](#) on [page 13](#)), and that you have saved it as a plaintext file with a filetype of `.tex` and a name of your own choosing, following the rules in the panel 'Picking suitable filenames' on [p. 49](#)[filenames](#).

B.1 \LaTeX from the Terminal

Originally a terminal was a screen and a keyboard, looking very much like a standard desktop computer in the days before flat screens and windowing systems. There are still a surprising number of these around. The important point is that it was (is) a text-only interface to the computer. You got 25 lines of 80 fixed-width white-on-black or green-on-black characters, no fonts, no colours, and no mouse; maybe reverse-video as a form of highlighting (see below).

Nowadays the word usually means a ‘virtual terminal’: a window that behaves like a terminal — 25 lines of 80 fixed-width characters in monochrome (see Figure B.2 on the following page). It’s a window into the heart of your computer. Even though you still have all your other windows visible, it knows nothing about them and can’t interact with them (except for copy and paste). But instead of being a padded cell, most terminals can do things many other windows can’t, like handling files in bulk, or to a schedule, or unattended, even forcing things to happen even when the graphical world outside has got itself jammed solid.

B.1.1 So where is the terminal window?

Apple Macintosh OS X: Click on `Finder` `Applications` `Utilities` `Terminal`;

Microsoft Windows: Click on the Windows or Start button, `All Programs` `Accessories` `Terminal` (in older versions it’s called `Command Prompt`);

Figure B.1 – Text-only display terminals



Images courtesy of Wikipedia. *Left:* IBM 3279 display by Retro-Computing Society of Rhode Island (CC BY-SA 3.0); *Right:* DEC VT100 terminal by Jason Scott (Flickr [IMG_9976](#), CC BY 2.0), at the Living Computer Museum (apparently connected to the museum’s DEC PDP-11/70).

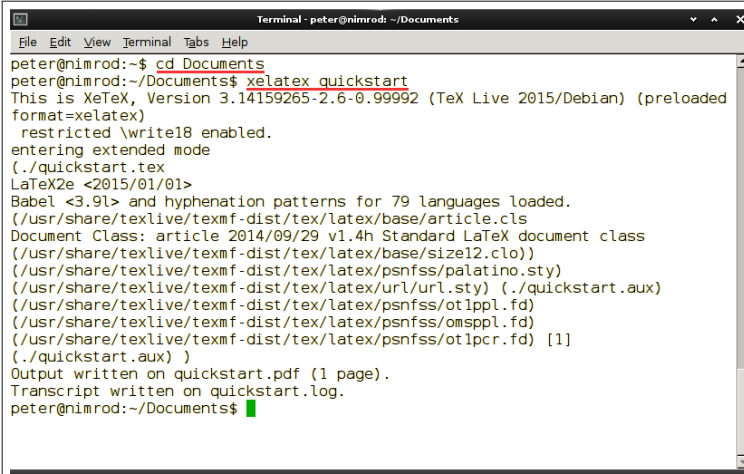
Unix and GNU/Linux: In most graphical interfaces, click on the menu `Applications` `Accessories` `Terminal` (in some systems it's called `Console`).

When you have finished using the terminal, it's good practice to type `exit` (and press `↵`).

B.1.2 Using the terminal window

On a physical terminal you usually have to log in first (very much like today: username and password). In a terminal window this isn't usually necessary (see below) because it's inside your graphical system, and you've already logged in to that.




Figure B.2 – Virtual terminal in a window



```
Terminal - peter@nimrod: ~/Documents
File Edit View Terminal Tabs Help
peter@nimrod:~$ cd Documents
peter@nimrod:~/Documents$ xelatex quickstart
This is XeTeX, Version 3.14159265-2.6-0.99992 (TeX Live 2015/Debian) (preloaded
format=xelatex)
restricted \write18 enabled.
entering extended mode
./quickstart.tex
LaTeX2e <2015/01/01>
Babel <3.9l> and hyphenation patterns for 79 languages loaded.
(/usr/share/texlive/texmf-dist/tex/latex/base/article.cls
(/usr/share/texlive/texmf-dist/tex/latex/base/size12.clo))
(/usr/share/texlive/texmf-dist/tex/latex/psnfss/palatino.sty)
(/usr/share/texlive/texmf-dist/tex/latex/url/url.sty) (. /quickstart.aux)
(/usr/share/texlive/texmf-dist/tex/latex/psnfss/ot1ppl.fd)
(/usr/share/texlive/texmf-dist/tex/latex/psnfss/omsppl.fd)
(/usr/share/texlive/texmf-dist/tex/latex/psnfss/ot1pcr.fd) [1]
(. /quickstart.aux)
Output written on quickstart.pdf (1 page).
Transcript written on quickstart.log.
peter@nimrod:~/Documents$
```

In this example I'm logged into a computer called `nimrod` with my username `peter`. The system prompt is the directory name plus a dollar sign (the tilde indicates that I'm in my home directory system). For visibility, I underlined in red here the commands I typed, one to change to my `Documents` folder, and one to run `XεLaTeX` on the `quickstart.tex` document.

The first thing you see is the prompt (usually a dollar sign or percent sign, or maybe a greater-than pointer `C:\>` like `Microsoft`

Disc Operating System (MS-DOS) used to use). When the prompt appears, you can type an instruction (command) and press the  key at the end of the line to send it off to the computer for processing. Until you press , the computer has no idea you've finished typing: you MUST press  at the end of each line of command for it to take effect.

The results, if any, are displayed on the screen, and the prompt is displayed again ready for your next command. Some commands don't have any output: if you change directory or delete a file, for example, you just get another prompt. There's no message confirming the action, and no check to see if you really meant it. You said to do it, and it's done.

B.2 Typesetting

Which L^AT_EX command you type depends on what output you want and how you want it to be created — see the list on page xxiii. Whichever way you run L^AT_EX, it will process your file and display a log or record of what it's doing (see Figure B.2 on the preceding page: it looks much the same no matter what system you use).

To typeset your document:

1. Make sure you are in the right directory (folder) in your terminal, then type the command `xelatex` followed by your filename; or `latexmk` or one of the other workflow-management commands);
2. If you are using citation and reference commands for a bibliography, you will then need to run `biber` or `bibtex` (followed by a space and the name of your document), whichever you have chosen to use (see section 5.3.2.1 on page 142);
3. Run X_YL^AT_EX again as in item above so that the citations are picked up;

4. If you are creating an index, you will then need to run `makeindex` (followed by a space and the name of your document);
5. Run \LaTeX again as in item above so that the citations and index references are resolved.

\LaTeX and all the ancillary programs write a transcript of what goes, and this will be shown in the window as well as being written into a log file.

If \LaTeX reports any errors — easily identifiable as lines in the log beginning with an exclamation mark (!) — *don't panic!* Turn to [section B.3](#), identify what went wrong, and fix it in your input file. Then re-run \LaTeX .

Exercise 32 – Running \LaTeX in a terminal or console window

- ☐ Open a command window;
- ☐ Type `cd` followed by the name of the folder where you saved your sample document;
- ☐ Type your \LaTeX command followed by the name of your \LaTeX document.

B.3 Errors and warnings

\LaTeX describes what it's typesetting while it does it, and if it encounters something it doesn't understand or can't do, it will display a message saying what's wrong. It may also display warnings for less serious conditions.

Don't panic if you see error messages: it's very common for beginners as well as seasoned users to mistype or mis-spell commands, forget curly braces, type a forward slash instead of a backslash, or use a special character by mistake. Errors are easily spotted and easily corrected in your editor, and you can then run \LaTeX again to check you have fixed everything. Some of the most

common errors are described in below with an explanation of how to fix them.

Some editors show hotlinks in the L^AT_EX log window where you can click on an error message and the cursor will jump to the line in your document where the error was spotted.

There is an extensive guide to how to handle errors in L^AT_EX in (Beeton 2017) (her presentation from TUG 2017) which also has a lot of useful information about how to work with L^AT_EX in general.

B.3.1 Error messages



The format of an error message is always the same. Error messages begin with an exclamation mark at the start of the line, and give a description of the error, followed by another line starting with the number, which refers to the line-number in your document file which L^AT_EX was processing when the error was spotted. Here's an example, showing that the user mistyped the `\tableofcontents` command:

```
! Undefined control sequence.  
1.6 \tableofcotnetns
```

When L^AT_EX finds an error like this, it displays the error message and pauses. You must type one of the following letters to continue:

Key	Meaning
x	Stop immediately and exit the program.
q	Carry on q uietly as best you can and don't bother me with any more error messages.
e	Stop the program but re-position the text in my e ditor at the point where you found the error (this only works if you're using an editor which L ^A T _E X can communicate with).
h	Try to give me more h elp.
i	(followed by a correction) means i nput the correction in place of the error and carry on (this is only a temporary fix to get the file processed. You still have to make that correction in the editor).

Some systems (*Emacs* is one example) run L^AT_EX with a 'non-stop' switch turned on, so it will always process through to the end of the file, regardless of errors, or until a limit is reached.

B.3.2 Warnings

Warnings don't begin with an exclamation mark: they are just comments by L^AT_EX about things you might want to look into, such as overlong or underrun lines (often caused by unusual hyphenations, for example), pages running short or long, and other typographical niceties (most of which you can ignore until later).

Unlike other systems, which try to hide unevennesses in the text — usually unsuccessfully — by interfering with the letter-spacing, L^AT_EX takes the view that the author or editor should be able to contribute. While it is certainly possible to set L^AT_EX's parameters so that the spacing is sufficiently sloppy that you will almost never get a warning about badly-fitting lines or pages, you will almost certainly just be delaying matters until you start to get complaints from your readers or publishers.

B.3.3 Examples

Only a few common error messages are given here: those most likely to be encountered by beginners. If you find another error

message not shown here, and it's not clear what you should do, ask for help.

Most error messages are self-explanatory, but be aware that the place where L^AT_EX spots and reports an error may be later in the file than the place where it actually occurred. For example if you forget to close a curly brace which encloses, say, italics, L^AT_EX won't report this until something else occurs which can't happen until the curly brace is encountered (eg the end of the document!) Some errors can only be righted by humans who can read and understand what the document is supposed to mean or look like.

Newcomers — remember to check the list of **special characters**: many errors when you are learning L^AT_EX are due to accidentally typing a special character when you didn't mean to. This disappears after a few hours as you get used to them.

B.3.3.1 Too many }'s

```
! Too many }'s.  
1.6 \date December 2004}
```

The reason L^AT_EX thinks there are too many }'s here is that the opening curly brace is missing after the `\date` control sequence and before the word `December`, so the closing curly brace is seen as one too many (which it is!).

In fact, there are other things which can follow the `\date` command apart from a date in curly braces, so L^AT_EX cannot possibly guess that you've missed out the opening curly brace — until it finds a closing one!

B.3.3.2 Undefined control sequence

```
! Undefined control sequence.  
1.6 \dtae  
    {December 2004}
```

In this example, L^AT_EX is complaining that it has no such command ('control sequence') as `\dtae`. Obviously it's been mistyped, but

only a human can detect that fact: all \LaTeX knows is that `\dtae` is not a command it knows about — it's undefined.

Mistypings are the commonest source of error. If your editor has drop-down menus to insert common commands and environments, use them!

B.3.3.3 Runaway argument

```
Runaway argument?
{December 2004 \maketitle
! Paragraph ended before \date was complete.
<to be read again>
\par
1.8
```

In this error, the closing curly brace has been omitted from the date. It's the opposite of the error in section B.3.3.1 on the preceding page, and it results in `\maketitle` trying to format the title page while \LaTeX is still expecting more text for the date! As `\maketitle` creates new paragraphs on the title page, this is detected and \LaTeX complains that the previous paragraph has ended but `\date` is not yet finished.

B.3.3.4 Capacity exceeded

```
! TeX capacity exceeded, sorry [parameter stack size=5000].
```

This is rather more serious: it means \TeX has completely run out of memory. This will happen if you try to push the system too far, like getting it to read lines which are unreasonably long, or macros which are too complex to fit in memory (or more likely just badly-written). I had it happen once (admittedly on an older system) with an author who had written a single paragraph over 37 pages long. I suggested this was perhaps a style that was unfair on his readers...but in fact the current version of \LaTeX is capable of handling the longest known footnote at 173 pages without any strain (Flynn 2023b).

B.3.3.5 Underfull hbox

```
Underfull \hbox (badness 1394) in paragraph
at lines 28--30
[ ]\LY1/brm/b/n/10 Bull, RJ: \LY1/brm/m/n/10
Ac-count-ing in Busi-
[94]
```

This is a warning that L^AT_EX cannot stretch the line wide enough to fit, without making the spacing bigger than its currently permitted maximum. The *badness* (0–10,000) indicates how severe this is (here you can probably ignore a badness of 1394). It says what lines of your file it was typesetting when it found this, and the number in square brackets is the number of the page onto which the offending line was printed.

The codes separated by slashes are the typeface and font style and size used in the line according to the definitions of the `fontname` package.

B.3.3.6 Overfull hbox

```
[101]
Overfull \hbox (9.11617pt too wide) in paragraph
at lines 860--861
[ ]\LY1/brm/m/n/10 Windows, \LY1/brm/m/it/10 see
\LY1/brm/m/n/10 X Win-
```

And the opposite warning: this line is too long by a shade over 9pt. The chosen hyphenation point which minimises the error is shown at the end of the line (*Win-*). Line numbers and page numbers are given as before. In this case, 9pt is too much to ignore (over 3mm or more than $\frac{1}{8}$ "), and a manual correction needs making (such as a change to the hyphenation), or the flexibility settings need changing (outside the scope of this book).

B.3.3.7 Missing package

```
! LaTeX Error: File `paralisy.sty' not found.

Type X to quit or <RETURN> to proceed,
```

or enter new name. (Default extension: sty)

Enter file name:

When you use the `\usepackage` command to request L^AT_EX to use a certain package, it will look for a file with the specified name and the filetype `.sty`. In this case the user has mistyped the name of the `paralist` package, so it's easy to fix. However, if you get the name right, but the package is not installed on your machine, you will need to download and install it before continuing (see Chapter 3 starting on page 65).