Word Processing: a Guide for the Perplexed

Word processors can make life easier or more difficult for us, depending on:

- which program we use;
- how much we know about it;
- how efficiently we apply what we know.

Consequently, we need to ask ourselves the following questions:

- What do my documents normally require in terms of layout? Do I use a lot of tables, pictures or footnotes, or do I just want simple text?
- Are my documents going to be used by other programs, or for other purposes (e.g. web pages)?
- Which program best fulfills these needs?
- What do I need to know in order to make it do all this?

Whatever word processor you decide to use, spend some time learning what it can do and the easiest way to make it do it. A few hours spent learning the basics can save you a lot of time later.

1 Text Formatting

Most word processor users spend far too much time on the details of formatting. Following a few simple rules can make life easier.

1. Your document is something to be read, not put on a wall and admired for its artistic qualities. Concentrate on making it readable, not beautiful. On the other hand, a badly-formatted document is harder to read, and may annoy the reader.

2. Unless you have a very good reason for doing otherwise, use one font for the whole of your document. Having a lot of different fonts makes things confusing for the reader, and wastes your time into the bargain. Use a standard font (e.g. Times Roman) in a normal size (e.g. 11 point).

3. For term papers, essays etc., double-space the lines and make sure the paper margins are wide enough for the reader to be able to write comments.

4. In general, you should only use bold for headings. Use italics for emphasis, for titles of books or for foreign words (remember that Turkish counts as a foreign language here!). As you can see from this paragraph, using bold normally makes a word stand out too much.

5. Keep tables simple. Using a lot of colours and patterns only distracts the reader. Similarly, only use borders and boxes if they are really necessary.

6. Leave one space after commas, colons and semi-colons, and two spaces after the end of a sentence (unless you’re using a program which does this automatically, such as LATEX).

7. If you are submitting a paper, find out if the person you are submitting it to has any special requirements. They may, for example, want you to number section headings, or conversely not to use section headings at all. Normally a cover page is required; find out what information needs to go on it and how it should be laid out.
2 File Formats

An irritating feature of word processors is that they all have their own formats for saving files. Not all word processors can read all file formats, so be careful which format you save in if you want other people to be able to read your documents, or if you want to use your own document with a different program. Use the “Save as” or “Export” commands from the “File” menu to see what formats are available. You can normally tell what format a document is in by looking at the part of its name after the dot. Some common formats are as follows:

.doc Microsoft Word Document. Only really suitable if you are only going to use it with MS Word; some other word processors can read it, but they usually lose some formatting in the process (particularly tables and quotation marks). Moreover, there are different versions of the .doc format, so, for example, you may not be able to read Word 97 documents in Word 6.0. Another problem is that it is vulnerable to macro viruses (e.g. the notorious “Love Letter”). Never send or open e-mail attachments in this format.

.txt Plain Text Format (ASCII). No formatting at all really, but very compact. Can be read by any word processor, or inserted into e-mail.

.rtf Rich Text Format. Sort of half way between plain text and Word Document. Has some basic formatting (e.g. italics). Not sophisticated, but can be read easily by many processors.

.html Hypertext Markup Language. This is the standard format for web-pages, but can also be used by and for word processors, so it makes quite a good lingua franca between different programs.

.pdf Portable Document Format. Useful for putting long documents on the web or sending them via e-mail; however, it can’t normally be edited, so you would normally only use this for distributing your final version. It needs a special program (e.g. Adobe Acrobat) to read it.

.ps PostScript. Used mainly for sending files to printers. It can also be viewed using programs such as Ghostview, but can’t be edited unless you are prepared to spend years learning how to do it.

.tex TeX, a typesetting format which is useless in itself, but can be converted easily into other formats such as PostScript, PDF, RTF or HTML.

Other formats, such as .lyx, .kwd, .sdw and so on, are only used by one particular word processor, and therefore need to be converted into something else if you want other programs to be able to read them.

If all this seems unnecessarily complicated, that’s because it is. Fortunately, there are moves to establish a standard document format which will hopefully make all this obsolete in a few years’ time. In the meantime, HTML is probably the best common format to use if you want your documents to be read by different programs. Even here, though, not all programs produce standard HTML (e.g. MS Word produces HTML that looks fine on Internet Explorer, but may come out strangely on other programs).

3 Programs

Contrary to popular opinion, Microsoft Word is not the only word processor in the universe. It is up to you to decide which program is best for your purposes. No program does everything well; all of them have advantages and disadvantages.
Standard Word Processors

These include programs like Microsoft Word, WordPerfect and Star Writer (the word processor part of Star Office). Questions you should ask yourself when choosing one of these are as follows:

- Do I already have it?
- If I haven’t got it, can I get it for free¹?
- What operating systems will it work on²?
- Is my computer powerful enough to use it effectively³?
- Does it have enough functions to do what I want? Does it have so many functions that it is confusing to use?
- What formats can it read and save as?

Remember that there is no point in getting a very sophisticated word processor if you only want to do simple things (especially if you have to pay money for it).

Editors

Editors are not strictly speaking word processors, but they can often do the same job. There are two main types: text editors and HTML editors.

A text editor works like a word processor, except that it does not format text. No bold or italic, no different fonts or font sizes — nothing! This may make it sound rather useless, but in fact, if you don’t need any formatting, a text editor is better to use — it is simpler and quicker, and the output (plain text format) can be read by any program running on any computer. Popular text editors are Notepad and Wordpad for Windows, and Gedit and KWrite for Linux/Unix.

HTML editors are designed for writing web pages, but they can also be used as word processors if you want. You don’t normally have as much control over the fine details of text formatting and layout, but with a little practice they are easy to use, and have the advantage that they save documents in HTML format. They can be divided into “proper” HTML editors, where you write directly in HTML, and WYSIWYG (“What you see is what you get”) editors, which work more like word processors. Examples of the former are EZPad, CoffeeCup and Quanta; WYSIWYG editors include Netscape Composer and Front Page⁴.

Probably in the future there will be no difference between a word processor and an HTML editor; one program will do both jobs.

LyX

LyX is a document processor or high-level typesetting program rather than a word processor — it is actually just a graphic interface to \LaTeX, which is a set of commands for the typesetting language \LaTeX. With LyX you just tell the program what you mean, and let it sort out most of the formatting itself⁵.

¹People generally don’t realise how much software is completely free, or free for limited purposes (e.g. WordPerfect is free for personal non-commercial use).
²For example, MS Word is only available for Windows and the Macintosh, KWord only works on UNIX and Linux, Star Office and AbiWord work on most systems.
³For example, Star Office and MS Office 2000 are big programs which take up a lot of disk space and require a reasonably fast computer (e.g. a Pentium II or III and at least 64MB of RAM). AbiWord, on the other hand, takes up only about 10 MB of disk space and will run on almost any computer.
⁴Older versions of Netscape are less suitable, since they produce non-standard HTML, but version 6.0 is OK. Similarly, if you want to use Front Page, get the full version not Front Page Express, as this also produces bad HTML.
⁵This document was produced with LyX.
\documentclass{article}
\begin{document}

\texttt{LaX} is ideal for academic writing, since that is what it was designed for. It is not quite as flexible as a normal word processor (you wouldn't use it to produce a magazine, for example) but after a little practice, it is much easier to use, and produces more professional-looking output. It can also produce HTML and PDF documents for putting on the web.

\texttt{LaX} can be downloaded free for Linux, UNIX and Windows, though you need to have \TeX\ and \LaX\ installed first (these are standard on Linux systems anyway).

4 Useful tips

1. Find out which things your program can do automatically (lists, styles etc.). For example, this list was produced with a style called "enumerate" – I didn't need to put in any of the numbers or spaces myself. If necessary, find out how to stop it doing things (look for a menu entitled "Options", "Settings" or "Preferences"). Many features are useless or annoying (e.g. changing (c) to a copyright sign).

2. Get to know your keyboard, especially how the Ctrl, Shift, arrow and page keys work (e.g. in most programs hitting Shift-End will block all text to the end of the line). Find out what the Alt or Alt Gr key does in combination with other keys; you may not have realised that your keyboard can produce characters like à or ñ, for example. Learn the most common key-bindings ("hot-keys"), e.g. Ctrl-B for bold, Ctrl-S for Save etc. Remember that when you are typing, your hands are on the keyboard, not on the mouse.

3. Unless your program handles indentation automatically, use the tab key or the rulers to indent text, rather than hitting the space bar repeatedly. Check to see if it has predefined styles for things like blockquotes.

4. When you start writing an essay, it's often a good idea to copy the main items in your outline into your essay as section headings. This will save you time later, and reminds you to keep to your plan. If your program doesn't have a predefined style for section headings (or you don't like the one it has), put them in bold, and make the font size slightly bigger than the rest of your text.

5. While writing, save regularly – don't trust the "Autosave" feature. In most programs, all you need to do is hit Ctrl+S, so do it after each paragraph. Make a backup copy when you finish: "My computer had a virus" is not a very good excuse for not submitting work.

6. Use a spellchecker if your program has one, but do not trust it. Many words you use will not be in the computer's dictionary, and some words may be in the dictionary but have the wrong spelling for that context (e.g. "effect" instead of "affect"). Never use the "Change all" or "Replace all" option – you may end up changing words that are actually correct, or correcting them in the wrong way.

7. Do major formatting, such as page breaks, right at the end – you don't want to have to keep changing things. If your program has a print previewer, use it to make sure your document looks OK before you print it out.

8. After you print your document, read it again. Sometimes you miss mistakes when you're looking at a computer screen, but you notice them when you see them on paper.

\end{document}