

Manejo Práctico de WinEdt: Nociones Básicas

En este documento se proporcionan las instrucciones básicas para la creación y compilación de documentos L^AT_EX con el entorno integrado WinEdt, así como una pequeña introducción al manejo de errores.

Comenzamos por suponer que abrimos un documento nuevo (y lo salvamos con un nombre dado, p.ej. test.tex) con una estructura muy básica:

```
\documentclass{article}

\begin{document}

Esto es una prueba sencilla de documento tipo article

\end{document}
```

Para compilarlo, podemos hacer click en los atajos que se muestran en la figura siguiente:



Cuyo significado es el siguiente:

1 Compilación L^AT_EX estándar del documento; produce como salida un documento test.dvi que, en caso de que se posea visor DVI (*lo cual no es el caso en **TeXLive 2008 instalado en Windows***) puede visualizarse en pantalla. Su equivalente en línea de comandos sería: `latex.exe test.tex`

2 Compilación directa con PDFL^AT_EX del documento; Se traducen los comandos L^AT_EX directamente a código PDF, obteniéndose como salida un documento test.pdf que podemos visualizar con Acrobat Reader (icono n° 6). Su equivalente en línea de comandos sería: `pdflatex.exe test.tex`

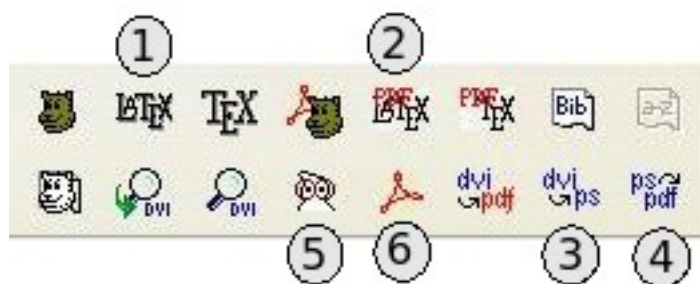
3 Transformación del documento test.dvi a test.ps (postscript) con DVIPS. El documento postscript puede ser visualizado con GSVIEW (icono n° 5) y enviado a imprimir desde ese programa. Su equivalente en línea de comandos sería: `dvips.exe test.dvi`

4 Trasformación del documento test.ps a test.pdf con PS2PDF. El documento PDF resultante puede ser visualizado con Acrobat Reader (icono n° 6) para posteriormente imprimirlo. Su equivalente en línea de comandos sería: `ps2pdf.exe test.ps test.pdf`

5 Visualización con GSVIEW del documento test.ps

6 Visualización con Acrobat Reader del documento test.pdf

Es interesante hacer notar que, a medida que se va compilando el documento, se van activando los iconos para cada tipo de acción posterior:



Se debe tener en cuenta que las dos rutas alternativas de compilación:

- 1) $\text{LATEX} \rightarrow \text{dvips} \rightarrow \text{ps2pdf}$
- 2) PDFLATEX

Son equivalentes en muchos casos, PERO NO EN TODOS. Por ejemplo:

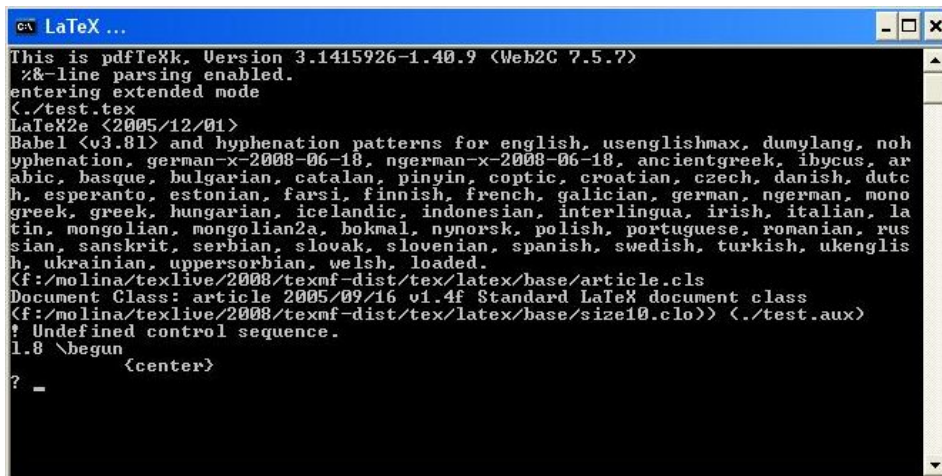
- Compilación de documentos con recursos PSTricks requiere LATEX forzosamente
- Diferentes resultados se obtienen al compilar presentaciones Beamer con LATEX ó PDFLATEX
- Existen algunas diferencias en el manejo de colores; la opción `usenames` del paquete `color` sólo proporciona unos pocos colores al compilar con pdfLATEX
- Documentos que incluyan gráficos vectoriales tipo postscript son compilados con LATEX , mientras que documentos con gráficos de mapa de bits (jpg, gif, bmp, etc...) han de ser compilados con PDFLATEX

Manejo de errores

Imaginemos que modificamos nuestro archivo de prueba para centrar el texto mediante el entorno `center`, de la siguiente manera (los números se incluyen, obviamente, sólo como guía de las líneas):

```
1 \documentclass{article}
2
3
4
5
6 \begin{document}
7
8 \begin{center}
9 Esto es una prueba sencilla de documento tipo article
10 \end{center}
11
12 \end{document}
```

El resultado de la compilación será que se abrirá la siguiente ventana de error:



```
LaTeX ...
This is pdfTeXk, Version 3.1415926-1.40.9 (Web2C 7.5.?)
%&-line parsing enabled.
entering extended mode
<./test.tex
LaTeX2e <2005/12/01>
Babel <v3.81> and hyphenation patterns for english, usenglishmax, dumylang, noh
yphenation, german-x-2008-06-18, ngerman-x-2008-06-18, ancientgreek, ibycus, ar
abic, basque, bulgarian, catalan, pinyin, coptic, croatian, czech, danish, dutc
h, esperanto, estonian, farsi, finnish, french, galician, german, ngerman, mono
greek, greek, hungarian, icelandic, indonesian, interlingua, irish, italian, la
tin, mongolian, mongolian2a, bokmal, nynorsk, polish, portuguese, romanian, rus
sian, sanskrit, serbian, slovak, slovenian, spanish, swedish, turkish, ukenglis
h, ukrainian, uppersorbian, welsh, loaded.
(f:/molina/texlive/2008/texmf-dist/tex/latex/base/article.cls
Document Class: article 2005/09/16 v1.4f Standard LaTeX document class
(f:/molina/texlive/2008/texmf-dist/tex/latex/base/size10.clo)) <./test.aux>
! Undefined control sequence.
l.8 \begin
      {center}
? -
```

donde el compilador indica la línea en la que se ha producido el error (8) y el tipo de error que se encuentra (undefined control sequence — en nuestro caso, significa que hemos escrito mal el comando “begin”).

Al obtener un error, lo más indicado es teclear “Enter” en la ventana, con lo cual se sigue la compilación ignorando ése error. Si no hay más errores, la ventana desaparecerá (indicando el final de la compilación del documento) y se debe pasar entonces a corregir el error en el documento fuente.

Finalmente, la siguiente tabla describe algunos tipos de error corrientes y su posible solución; además de los errores, se mencionan algunos “warnings” (que no detienen la compilación) y que implican diversos problemas no esenciales.

Errores		
Tipo de error	Ejemplo	Motivo y solución
Undefined control sequence	<code>\began{...}</code>	Error en la sintaxis de un comando. Corregir la errata.
Too many }'s.	<code>\textbf{Hola}}</code>	Paréntesis desemparejados. Añadir los que falten ó sobren.
Runaway argument	<code>\hspace{2mm</code>	En un comando que depende de un argumento falta la llave de cierre. Añadirla
Illegal unit of measure (pt inserted)	<code>\parskip=2</code>	En una especificación de longitud falta la unidad de medida (pt, cm, ...). \LaTeX entiende pt por defecto
LaTeX Error: File 'Color.sty' not found	<code>\usepackage{Color}</code>	Errata en el nombre de un paquete. Tanto para paquetes como para comandos, debe tenerse en cuenta que \LaTeX diferencia mayúsculas y minúsculas.
Paragraph ended before \centerline was complete.	<code>\centerline{Hola Pepe}</code>	Intentamos escribir varios párrafos en un comando “corto” cuyo argumento no admite salto de párrafo. Utilizar un entorno: <code>\begin{center} ... \end{center}</code> , por ejemplo.
Warnings		
Underfull hbox		Hay demasiado espacio libre en una “caja” del que le gustaría al compilador (se verá más adelante cómo \LaTeX trabaja situando objetos en cajas y colocando a su vez las cajas a lo largo de la página). No suele ser grave.
Overfull hbox		Lo opuesto al anterior. Un objeto es demasiado largo y se sale de la caja asignada. Sí suele ser grave. Por ejemplo (se usa <code>\fbox{Texto...}</code> para enmarcar texto): Texto muy largo