

Laboratorio Sesión 05: Herramientas de edición de documentación

En esta sesión de laboratorio vamos a trabajar con herramientas de edición ofimática. La gran diferencia de las herramientas que vamos a utilizar con respecto a las herramientas tradicionales para el gran público a las que estáis acostumbrados (tipo LibreOffice Writer) es que las herramientas tradicionales son mucho más sencillas de aprender pero no son tan efectivas para manejar ficheros de gran tamaño (por ejemplo la memoria de un Trabajo de Fin de Grado o una Tesis). Además ya veréis que las herramientas que vamos a ver tienen un acabado más profesional una vez el documento está listo para imprimir. También aprovecharemos el hecho de trabajar con texto para combinar las herramientas de edición con scripts que recaben información automáticamente.

5.1. Sobre tipos de editores de documentos

Una de las grandes diferencias que vais a encontrar es que las herramientas que vamos a ver en esta práctica son WYTIWYG (aunque actualmente hay quienes defienden que hay que transformar estas herramientas bajo el paradigma WYSIWYM) en lugar de WYSIWYG y esto, aunque resulta más complicado de aprender a la larga permite un control mucho más fino y sencillo.

Pregunta 5.1: *¿Qué significan los acrónimos WYTIWYG, WYSIWYG y WYSIWYM?*

Un editor WYSIWYM (por ejemplo **Lyx**) añade un entorno gráfico encima de un sistema WYTIWYG intentando esconder su complejidad. Esto hace que no sea la mejor manera de aprender así que en esta práctica nos limitaremos a ver la base y queda a vuestra decisión decidiros más adelante por un sistema de ese tipo. Otra alternativa de uso común es editar el fichero texto original con un editor especializado que nos ayude con la sintaxis. En el caso de \LaTeX tenemos ejemplos como **Kile**, **TeXMaker** o **Gummi**. Estos editores ayudan a la edición ya que conocen los comandos \LaTeX y pueden autocompletarlos o incluso sugerirlos cuando nos los olvidamos. Además nos ahorran algunos pasos de compilación (sí, los textos \LaTeX se compilan). Si decidís trabajar en \LaTeX os animo a probar este tipo de herramientas.

5.2. Creación de textos: \LaTeX

Cuando hablamos de edición de textos WYTIWYG hablamos de \LaTeX . Evidentemente en esta práctica no vamos a poder ver todas las opciones, pero hay numerosa documentación en Internet que puede os puede resultar muy útil. Os recomiendo por ejemplo el libro sobre \LaTeX de Wikibooks <http://en.wikibooks.org/wiki/LaTeX>. En él podéis buscar toda la información que necesitéis para hacer esta práctica.

5.2.1. Un documento mínimo

Vamos a crear el primer documento. Para empezar con el documento mínimo podéis crear un fichero texto plano (por ejemplo “Miprimerdocumento.tex”) con el siguiente contenido:

```
\documentclass{article}

\begin{document}
Hola mundo.
\end{document}
```

A continuación guardad el fichero. Ahora vamos a compilarlo. Aunque hay varias formas de compilar un fichero \LaTeX , probablemente la más rápida es la siguiente:

```
pdflatex Miprimerdocumento
```

Probablemente os de un error ya que no estan instalados los paquetes de \LaTeX en el sistema, pero si os fijáis el mismo mensaje de error os sugiere la forma de arreglarlo. Hacedlo y compilad.

Veréis que aparecen varias líneas de información y al acabar habrán aparecido varios ficheros auxiliares y el que nos interesa: “Miprimerdocumento.pdf”. Este podéis abrirlo con cualquier lector de PDF que tengáis instalado. Si todo ha ido bien ya sabéis crear documentos en \LaTeX . El resto son detalles (aunque me temo que hay muchos).

5.2.2. Añadiendo detalles: los acentos

Probad ahora a añadir la siguiente línea a vuestro documento y recompilad:

Puedo escribir camión.

Pregunta 5.2: *¿Qué salida genera \LaTeX con la línea anterior?*

Hay dos formas de corregir el problema. La más directa e incómoda es usar la codificación especial de \LaTeX para los acentos: escribir `\’o` en vez de `ó`. Probad con la siguiente línea:

Puedo escribir cami\’oón.

Veréis que ahora la salida es correcta, pero esto es bastante pesado. Lo mejor es que directamente extendáis las capacidades de \LaTeX para que este pueda entender los caracteres escritos en castellano sin problemas. \LaTeX incorpora muchos paquetes que permiten añadirle funcionalidades. El que vosotros queréis se llama `inputenc` y se añade con la siguiente línea que debe ir antes de que empiece el documento:

```
\usepackage[utf8]{inputenc}
```

Pregunta 5.3: *¿Qué salida da ahora la línea Puedo escribir cami\’oón.?*

Más adelante veremos algunos paquetes útiles más, aunque algunos pueden requerir de la instalación de algún programa más.

5.2.3. Errores y otras lindezas

Probablemente de momento no os haya pasado, pero si utilizáis \LaTeX os acabará pasando que el sistema, como cualquier otro compilador, os generará errores. La mayoría de errores son bastante tontos, pero si uno no está acostumbrado generan bastante desconcierto ya que el compilador para a preguntar que hacer (de hecho si usáis mucho \LaTeX en algún momento os sorprenderéis debugando, es uno de sus pocos defectos). Vamos a introducir algún error a posta, me temo que más adelante seguro que veréis más. Cuando el sistema pregunte hay tres soluciones básicas: pulsar `x` para que acabe y volver a editar el texto, pulsar `H` para que nos de algo de ayuda o pulsar “Enter” para ver si funciona (vamos, si cuela, cuela). Probad a quitar un `}` del texto.

Pregunta 5.4: *¿Qué error os da \LaTeX ?*

Otro de los problemas con \LaTeX se da a la hora de introducir caracteres de control. Por ejemplo, la `\` indica un comando. A la hora de escribir estos caracteres tenéis que usar un comando especial. Por ejemplo, para escribir la `\` tenéis que escribir `\textbackslash`.

Pregunta 5.5: *¿Cómo se escriben las llaves (`{}`) en \LaTeX ? ¿Cómo se escribe el símbolo “ \LaTeX ”? ¿Y las comillas?*

Si queréis escribir algo literal y \LaTeX no lo formatee, lo mejor es ponerlo dentro del entorno `verbatim`. Para ello podéis escribir: `\verb"texto"` o `\begin{verbatim} Texto \end{verbatim}`.

Pregunta 5.6: *¿Qué diferencia hay entre usar `\verb` y usar `\begin{verbatim}`?*

5.2.4. Los títulos y las secciones

Ahora que ya podemos escribir texto vamos a ver como organizarlo un poco. Lo primero que deberemos aprender es a escribir títulos y a crear secciones. Para escribir títulos existe el comando `\title`. Podéis probar lo siguiente justo después del `\begin{document}`:

```
\title{Mi Título}

\maketitle
```

Pregunta 5.7: *¿Qué salida genera \LaTeX con el comando anterior?*

Pregunta 5.8: *¿Qué Warning da \LaTeX ? ¿Cómo se inserta un autor?*

Pregunta 5.9: *¿Qué sucede si no ponemos el comando `\maketitle`? ¿Y si insertamos el comando `\date{}` antes del `\maketitle`?*

Pregunta 5.10: *¿Cómo harías que \LaTeX imprimiera siempre la fecha de tu cumpleaños en lugar de la del día que se compila en el título?*

Además para crear secciones (de mayor a menor categoría), \LaTeX tiene los siguientes comandos: `\section`, `\subsection` y `\subsubsection` (luego veremos que también existe `\chapter`). Como `\title` van seguidos de unas llaves con el título dentro, por ejemplo:

```
\section{Primera sección}
```

Pregunta 5.11: *¿Qué diferencia hay entre escribir `\section{}` y `\section*{}`?*

Evidentemente \LaTeX actualiza los índices automáticamente (probad a poner más de una sección o subsección) y las páginas y todo lo que se introduzca en el documento (tablas, figuras, etc.). Eso sí, para poder tener en cuenta tanto referencias hacia delante como hacia atrás, conviene compilar dos veces. Si no es así a veces perderá las numeraciones (en esos casos o bien las numeraciones están mal o bien pone un ?).

Otra de las ventajas importantes de \LaTeX es que aprovecha que todos los elementos del texto están indicados para poder hacer fácilmente los índices. Probad con el comando: `\tableofcontents` y compilad una vez.

Pregunta 5.12: *¿Qué salida genera `\tableofcontents` al compilar una vez? ¿Y al compilar dos veces?*

En este momento si miráis con calma el documento que acabáis de crear veréis que tiene un problema: el título del índice está en inglés. Aunque se pueden cambiar todos los títulos a mano, existe un paquete (`babel`) que lo hace automáticamente. Para usarlo hay que insertar en el preámbulo la línea:

```
\usepackage[spanish]{babel}
```

Como ya habéis visto con el paquete `inputenc`, delante del nombre del paquete, entre corchetes, van las opciones, en este caso el idioma al cual queremos que se adapte el documento (podéis probar con otros idiomas). Si habéis usado la línea anterior y compilado veréis que \LaTeX genera un error debido a que no tiene el paquete de idioma español instalado (generalmente se instala automáticamente pero estos ordenadores como ya comentamos no tienen el soporte de idiomas definido). Si queréis instalarlo tendréis que ejecutar:

```
sudo apt-get install texlive-lang-spanish
```

Finalmente hay tres secciones más que os pueden resultar interesantes:

- El abstract (resumen): en esta sección se suele poner un resumen del artículo. Suele situarse al principio del documento y se define escribiéndola entre los comandos `\begin{abstract}` y `\end{abstract}`.
- Apéndices: el comando `\appendix` crea una nueva numeración de las secciones (o de los capítulos, según el tipo de documento como veremos luego), adecuada al hecho de tener apéndices, claro.
- Bibliografía: En muchos documentos es útil poder introducir bibliografía. La forma más sencilla en \LaTeX de tener bibliografía es añadiendo al final una sección similar al resumen pero con título `thebibliography`. Podéis probar con las siguientes líneas:

```
\begin{thebibliography}{9}
\bibitem{lampport94}
  Leslie Lampport,
  \emph{\LaTeX: A Document Preparation System}.
  Addison Wesley, Massachusetts,
  2nd Edition,
  1994.
\end{thebibliography}
```

Una vez tengáis la bibliografía en medio del documento podéis introducir una referencia usando el comando `\cite{lampport94}`.

Añadid a vuestro documento de prueba un abstract, un par de apéndices y una bibliografía (basta con el ejemplo anterior) con al menos una referencia. Compilad.

Pregunta 5.13: *¿Cuántas entradas aparecen ahora en vuestro índice? ¿Qué texto introduce \LaTeX en lugar de la cita?*

Fijaros que aunque os parezca que está siendo mucho trabajo para elaborar un documento muy sencillo, si este crece (un artículo real puede tener fácilmente 30 páginas de largo) \LaTeX en realidad os facilita mucho la vida ya que no tenéis que preocuparos por introducir una sección nueva (el las vuelve a enumerar y rehace el índice), cambiar de lugar las citas (se acuerda de donde están) o procurar que quede con una apariencia profesional. Todo esto es automático y cuesta mucho trabajo en un editor de documentos convencional conforme el tamaño de dichos documentos crece.

5.2.5. Tipos de documentos y sus opciones

Hasta ahora hemos visto como crear un artículo científico de una forma bastante rápida y efectiva. Y efectivamente esa fue una de las primeras aplicaciones de \LaTeX . Sin embargo actualmente se usa para muchos más tipos de documentos. Por ejemplo: prácticas ;-), libros, informes... bueno, en realidad puede usarse para todo ya que si os fijáis en realidad se pueden usar solo una parte de las opciones de un documento (si a un documento de tipo artículo le quitamos el resumen y la bibliografía ya puede ser casi cualquier texto, como en el caso de estas prácticas), y las partes del documento que queden se pueden modificar tanto como queramos (este sería el caso de las transparencias de clase, pero eso lo veremos luego).

A continuación vamos a ver algunas de las opciones que tenemos para cambiar de forma sencilla la apariencia y las opciones de los documentos. Para ello en primer lugar rellenad con un poco de texto el fichero “prueba” de forma que tenga al menos un par de párrafos y ocupe un par de páginas (prácticamente ya deberíais haber llegado a ese punto).

Ahora vamos a fijarnos en la primera línea:

```
\documentclass{article}
```

Lo primero que debéis de saber es que tenemos varias opciones para incluir delante de la clase de documento. Las opciones en este caso, como en los paquetes, van delante, entre corchetes y

separadas por comas. Las más comunes son: 10pt, 11pt, 12pt; a4paper, letterpaper; onecolumn, twocolumn.

Pregunta 5.14: *¿Qué hacen las opciones anteriores? Probad la opción twocolumn en vuestro documento. ¿Cuántas páginas ocupa ahora?*

Fijaros que es muy fácil modificar el aspecto de TODO el documento con sólo pequeños cambios.

También podéis cambiar el tipo base de documento. Los más comunes son: article, report y book (la verdad es que el tipo letter ya no suele usarse).

Pregunta 5.15: *¿Para qué son adecuados cada uno de los tipos de documento anteriores? ¿Cuales soportan capítulos? ¿Qué comando introduce un capítulo?*

Introducid un par de capítulos en vuestro documento de prueba (cambiando el tipo) y volved a compilar.

Pregunta 5.16: *¿Qué tipo de documento habéis puesto? ¿Cuántas páginas ocupa ahora el documento? ¿En que secciones introduce L^AT_EX un salto de página?*

5.2.6. Tipos de letra, tamaños y espacios

Aunque en realidad L^AT_EX suele encargarse de los detalles del formato en ciertas ocasiones es útil cambiar el tipo de letra por defecto, resaltar algún contenido o añadir espacios o saltos de línea en el documento.

Pregunta 5.17: *¿Cuántos tipos de letra hay en L^AT_EX? ¿Cuales son? ¿Cómo se utilizan?*

Pregunta 5.18: *¿Cuántos tamaños de letra hay en L^AT_EX? ¿Cuál es el más pequeño? ¿Cuántas líneas ocupa la primera línea de ejemplo (Hola mundo. Puedo escribir camión con el tamaño más grande?*

Pregunta 5.19: *¿Cómo se puede poner el texto en negrita? ¿Y en itálica? ¿Y en mayúsculas pequeñas?*

5.2.7. Incluyendo código, listas, imágenes y tablas

Finalmente vamos a ver como incluir cuatro tipos de elementos que probablemente necesitéis en vuestros documentos: código, listas, imágenes y tablas.

Empecemos por el código (en estas prácticas yo lo incluyo usando simplemente verbatim, pero este ejemplo es más elaborado por si tenéis que incluir cantidades significativas de código en un documento). Una buena alternativa es usar el paquete listings. Hacedlo (probablemente necesitéis instalar el paquete texlive-latex-recommended) e incluid el siguiente código en vuestro documento de prueba:

```
\begin{lstlisting}[language=C,numbers=left]
#include <stdio.h>
int main(int argc, char* argv[]) {
  puts("Hola mundo!");
}
\end{lstlisting}
```

Pregunta 5.20: *¿Qué mejoras ofrece el paquete listings sobre el entorno verbatim?*

Incluir listas es tan fácil como usar los entornos itemize y enumerate. Probad con el siguiente código:

```
\begin{itemize}
\item Esto es una lista
\item De dos elementos
\end{itemize}
```

Pregunta 5.21: *¿Qué diferencia hay entre `itemize` y `enumerate`? Fijaros que se pueden usar unos niveles dentro de otros.*

Para incluir figuras basta que uséis el paquete `graphicx`. Este permite incluir varios tipos de imágenes en el documento (JPG, PNG o vectoriales como las que genera el programa Inkscape). En caso de que no dispongáis del formato adecuado recordad que siempre podéis usar la herramienta (de Linux) `convert`. Una vez que tengáis el fichero con la imagen (bajaros alguna de Internet para probar) podéis incluirla con los comandos:

```
\begin{figure}[htb]
\centering
\includegraphics[width=0.8\textwidth]{imagen.jpg}
\caption{Imagen increíble}
\label{fig:imagen_increible}
\end{figure}
```

Pregunta 5.22: *¿Para que sirve el comando `\caption{Imagen increíble}`?*

Fijaros que se puede hacer referencia a la imagen en cualquier parte del documento simplemente escribiendo `\ref{fig:imagen_increible}`. Haced un par de referencias a la imagen en vuestro código.

Pregunta 5.23: *¿Con qué texto referencia \LaTeX la imagen?*

Las tablas son el elemento más complicado. A continuación tenéis un ejemplo que podéis copiar.

```
\begin{table}[tbp]
\centering
\begin{tabular}{||c|c||}
\hline\hline
celda 1.1 & celda 1.2 \\
\hline
celda 2.1 & celda 2.2 \\
\hline\hline
\end{tabular}
\caption{Esta tabla muestra algunos datos}
\label{tab:miprimeratabla}
\end{table}
```

Pregunta 5.24: *¿Para qué sirve el comando `\hline`?*

De todas formas hay herramientas (como `calc2latex`) que permiten exportar tablas desde LibreOffice Calc u otros programas. Como se puede deducir por el código, \LaTeX referencia igual las tablas que las imágenes (llevando la numeración en cada caso). Si queréis, además, puede generar un índice de figuras o de tablas simplemente escribiendo `\listoffigures` o `\listoftables`.

Pregunta 5.25: *¿Qué texto introduce \LaTeX en el documento al escribir `\listoftables`? ¿Qué texto pone en la cabecera de la tabla? ¿Cómo se puede cambiar dicho texto a, por ejemplo, “Tablita” para que cambie en todas las tablas del documento?*