

L^AT_EX para la tarea de mañana

Clase #2: Primer documento

Amado Alberto Cabrera Estrada

amadocab.268@gmail.com*

Junio 2022

Resumen

En esta clase explicaremos lo necesario para poder hacer nuestro primer documento en L^AT_EX, tratando los títulos, los diferentes elementos para dar estructura, etc. De esta manera el alumno debería ser capaz de elaborar su primer documento sin nada más que la referencia de esta hoja.

Índice

0. Introducción	2
0.1. Preámbulo base	2
1. Título del documento	2
2. Dar estructura a un documento	3
2.1. <code>part</code>	3
2.2. <code>section</code>	3
2.3. <code>subsection</code>	3
2.4. <code>subsubsection</code>	4
2.5. Todos los comandos	4
2.6. Índice	4
3. Escribir texto	4
3.1. Espacios	4
3.2. Saltos de línea	5
4. Tipografía	5
4.1. Tipos de letra	5
4.2. Tamaños de letra	6
5. Listas	6
5.1. Listas simples	6
5.2. Listas anidadas	7
5.3. Cambiando etiquetas para toda la lista	8
5.4. Cambiando etiquetas para un elemento	9

*Para sugerencias, correcciones o comentarios

0. Introducción

En esta clase vamos a crear nuestro primer documento de L^AT_EX para aprender cómo se compone generalmente este tipo de documentos.

0.1. Preámbulo base

Para el inicio de nuestra clase utilizaremos el preámbulo que puede verse a continuación, este tiene ligeras diferencias con el que vimos la clase anterior; sin embargo, el significado de ambos es exactamente el mismo.

```
\documentclass[spanish]{article}
\usepackage[utf8]{inputenc}
\usepackage[T1]{fontenc}
\usepackage{babel}
\decimalpoint
```

1. Título del documento

Para hacer el título de nuestros documentos de L^AT_EX necesitaremos únicamente de cuatro comandos. Uno para definir el título de nuestro documento, `\title{...}`; uno para el autor, `\author{...}`; otro para la fecha, `\date{...}` y el último para escribirlo en el documento, `\maketitle`. Esto puede verse así:

```
\title{¡Hola Mundo!}
\author{John Doe}
\date{29 de mayo 2022}

\begin{document}
\maketitle
\end{document}
```

¡Hola Mundo!

John Doe

27 de diciembre 2000

La falta de el comando para el título y el autor llevan a un error y una advertencia respectivamente, pero curiosamente sí puede faltar el comando de la fecha. En los casos en los que el comando es omitido, este es sustituido en posición por el comando `\today`, el cual coloca la fecha del día actual¹.

```
\title{¡Hola Mundo!}
\author{John Doe}

\begin{document}
\maketitle
\end{document}
```

¡Hola Mundo!

John Doe

30 de mayo de 2022

Por último, si queremos evitar que se coloque algún campo del título; la mejor manera es dejar escritos los comandos, pero sin nada dentro de ellos. Un ejemplo con la fecha sería:

¹Cabe aclarar que cuando se hace esto desde Overleaf tendremos como inconveniente que L^AT_EX usará la fecha de la computadora en la que corre (en la nube), así que hay que configurarlo para que coloque la fecha de la zona horaria que queremos.

```
\title{¡Hola Mundo!}
\author{John Doe}
\date{}
```

```
\begin{document}
\maketitle
\end{document}
```

¡Hola Mundo!

John Doe

2. Dar estructura a un documento

Para escribir un documento bien estructurado tenemos que poder separar las distintas partes, secciones, etc; que componen un documento (`article`). Para esto \LaTeX cuenta con una variedad de comandos, iniciando de más grande a más pequeño tenemos.

2.1. `part`

```
\part{Numerado}
Texto para colocar debajo
```

```
\part*{No numerado}
Texto para colocar debajo
```

Parte I

Numerado

Texto para colocar debajo

No numerado

Texto para colocar debajo

2.2. `section`

```
\section{Numerado}
Texto para colocar debajo
```

```
\section*{No numerado}
Texto para colocar debajo
```

1. Numerado

Texto para colocar debajo

No numerado

Texto para colocar debajo

2.3. `subsection`

```
\subsection{Numerado}
Texto para colocar debajo
```

```
\subsection*{No numerado}
Texto para colocar debajo
```

1.1. Numerado

Texto para colocar debajo

No numerado

Texto para colocar debajo

2.4. subsubsection

```
\subsection{Numerado}
Texto para colocar debajo

\subsection*{No numerado}
Texto para colocar debajo
```

1.1. Numerado

Texto para colocar debajo

No numerado

Texto para colocar debajo

2.5. Todos los comandos

Cuando nos referimos a los comandos que nos ayudan a darle estructura al contenido de nuestro documento además de los antes vistos tenemos otros, que no aplican directamente a la clase `article`. Para referencia ver la tabla 1.

Comando	Nivel	Comentario
<code>\part {...}</code>	-1	No está en <code>letter</code>
<code>\chapter {...}</code>	0	Solo en <code>book</code> y <code>report</code>
<code>\section {...}</code>	1	No está en <code>letter</code>
<code>\subsection {...}</code>	2	No está en <code>letter</code>
<code>\subsubsection {...}</code>	3	No está en <code>letter</code>

Tabla 1: Partes para dividir un documento^[3]

2.6. Índice

Hacer un índice en \LaTeX es bastante sencillo, basta con escribir el comando `\tableofcontents` y todas las divisiones numeradas en nuestro texto se incluirán directamente en el documento. Las divisiones que no estén numeradas (versiones estrelladas de nuestros comandos) no se incluyen en el índice de manera automática.

Para añadir una entrada al índice sin necesidad de una sección numerada usaremos:

```
\addcontentsline{toc}{sec_unit}{entry}
```

dónde `sec_unit` es una de las divisiones antes vistas y `entry` es el nombre que queremos que tenga nuestra entrada en el índice.

3. Escribir texto

Uno de los problemas que se tiene al iniciar con \LaTeX se encuentra principalmente en la manera en la que este lidia con los espacios y saltos de línea.

3.1. Espacios

Lo más importante que hay que saber respecto a los espacios es que, a diferencia de los procesadores de texto, estos en \LaTeX no son acumulativos. Esto quiere decir que para \LaTeX no existe diferencia entre un espacio o muchos.

```
Lorem ipsum dolor
```

Lorem ipsum dolor

```
Lorem                ipsum dolor
```

Lorem ipsum dolor

tal como se puede ver en los ejemplos anteriores.

3.2. Saltos de línea

Los saltos de línea por su parte tienen los mismos problemas que los espacios (no son acumulativos), pero además de eso pueden causar problemas dependiendo del estilo de párrafo que se quiere utilizar². Nosotros para facilitar la escritura del texto utilizaremos el paquete `parskip`, donde los párrafos están en formato moderno (o alemán).

Agregaremos a nuestro preámbulo entonces:

```
\usepackage{parskip}
```

Tenemos ahora que, un solo *enter* no cambia de párrafo (se escribe en la misma línea).

```
Primera línea.  
Segunda línea.
```

Primera línea. Segunda línea.

Dos *enter*, por otro lado sí que lo harán.

```
Primera línea.  
  
Segunda línea.
```

Primera línea.
Segunda línea.

Si quisiéramos forzar un salto de línea en algún lugar, podemos hacer uso del comando `\\` simplemente o escribir seguido de él un argumento opcional para especificar el tamaño del salto que queremos hacer.

```
Primera línea.\\[0.5cm] Segunda línea.
```

Primera línea.
Segunda línea.

4. Tipografía

Para resaltar detalles específicos dentro de un texto es fundamental el -buen- uso de detalles tipográficos que nos ayuden a expresar lo que queremos. Para esto veremos como variar tipos y tamaños de letra.

4.1. Tipos de letra

En este caso utilizaremos el tipo por defecto de L^AT_EX, *Computer Modern Roman* e iremos variando la «familia», «serie» y «forma»^[1]. Para no profundizar en estos aspectos los resumiremos en la tabla 2.

Comando	Resultado	Comando	Resultado
<code>\rmfamily</code> Lorem	Lorem	<code>\textrm {Lorem}</code>	Lorem
<code>\sffamily</code> Lorem	Lorem	<code>\textsf {Lorem}</code>	Lorem
<code>\ttfamily</code> Lorem	Lorem	<code>\texttt {Lorem}</code>	Lorem
<code>\mdseries</code> Lorem	Lorem	<code>\textmd {Lorem}</code>	Lorem
<code>\bfseries</code> Lorem	Lorem	<code>\textbf {Lorem}</code>	Lorem
<code>\upshape</code> Lorem	Lorem	<code>\textup {Lorem}</code>	Lorem
<code>\itshape</code> Lorem	<i>Lorem</i>	<code>\textit {Lorem}</code>	<i>Lorem</i>
<code>\slshape</code> Lorem	<i>Lorem</i>	<code>\textsl {Lorem}</code>	<i>Lorem</i>
<code>\scshape</code> Lorem	LOREM	<code>\textsc {Lorem}</code>	LOREM

Tabla 2: Estilos de letra en L^AT_EX

²Para más información leer [este artículo](#) de Wikilengua

Cómo se puede ver existen dos maneras de producir los mismos resultados, la de la derecha es la manera en la que habitualmente se escriben los comandos y tiene la forma:

<code>\textbf{Lorem}</code> ipsum sit		 Lorem ipsum sit
---------------------------------------	--	-------------------------

En este caso se puede ver como el comando aplica únicamente a lo que se le pasa como argumento.

Por otro lado la segunda manera de aplicar este tipo de efectos es utilizando comandos que no toman argumentos y que aplican a todo lo que los rodea, a menos que sea delimitado entre un par de llaves. Este tiene la forma:

<code>{\bfseries Lorem ipsum}</code> sit		 Lorem ipsum sit
--	--	-------------------------

este tipo de comando se suelen utilizar casi de manera exclusiva como configuración de otros comandos o si se quiere aplicar el mismo efecto a todo el texto.

4.2. Tamaños de letra

Para cambiar el tamaño de letra en L^AT_EX disponemos de 10 comandos del tipo que no reciben argumentos. Estos pueden verse en la tabla 3.

Comando	Resultado
<code>\tiny</code> Lorem	<small> Lorem</small>
<code>\scriptsize</code> Lorem	<small> Lorem</small>
<code>\footnotesize</code> Lorem	<small> Lorem</small>
<code>\small</code> Lorem	<small> Lorem</small>
<code>\normalsize</code> Lorem	<small> Lorem</small>
<code>\large</code> Lorem	<big> Lorem</big>
<code>\Large</code> Lorem	<big> Lorem</big>
<code>\LARGE</code> Lorem	<big> Lorem</big>
<code>\huge</code> Lorem	<big> Lorem</big>
<code>\Huge</code> Lorem	<big> Lorem</big>

Tabla 3: Tamaños de letra en L^AT_EX

5. Listas

Es común que cuando estamos escribiendo algo querramos enumerar de algún modo, en L^AT_EX poseemos 3 modos básicos de enumerar cosas. Para esto disponemos de 3 *environments* distintos; las listas ordenadas, **enumerate**; las listas desordenadas, **itemize** y las listas descriptivas, **description**.

Para facilitar el uso de las listas utilizaremos el paquete `enumitem`^[2] con argumento opcional `shortlabels`, agregaremos entonces:

```
\usepackage[shortlabels]{enumitem}
```

5.1. Listas simples

Primero vamos a ver como escribir una lista sencilla dentro de L^AT_EX. Para ello haremos uso de cualquiera de los *environments* antes vistos (`enumerate`, `itemize` y `description`) y precederemos cada elemento de la lista por el comando `\item`.

Para una lista ordenada y no ordenada solo cambiamos el nombre del *environment*, es decir:

```
\begin{enumerate}
  \item Primer elemento

  \item Segundo elemento

  \item Tercer elemento
\end{enumerate}
```

1. Primer elemento
2. Segundo elemento
3. Tercer elemento

```
\begin{itemize}
  \item Primer elemento

  \item Segundo elemento

  \item Tercer elemento
\end{itemize}
```

- Primer elemento
- Segundo elemento
- Tercer elemento

Para la lista descriptiva el único cambio que tenemos es que en esta las etiquetas de los elementos deben especificarse manualmente como argumento opcional al comando `\item`.

```
\begin{description}
  \item[Primer] elemento

  \item[Segundo] elemento

  \item[Tercer] elemento
\end{description}
```

- Primer** elemento
- Segundo** elemento
- Tercer** elemento

5.2. Listas anidadas

Además de hacer listas normales, también podemos hacer listas anidadas (listas dentro de otras listas) esto -normalmente- solo puede hacerse 4 veces seguidas (del mismo tipo de lista).

En las listas ordenadas puede verse como cambian los formatos usados para enumerar; siendo estos: números con punto, letras con paréntesis, números con paréntesis y finalmente letra con apóstrofo.

```
\begin{enumerate}
  \item Nivel 1
  \begin{enumerate}
    \item Nivel 2
    \begin{enumerate}
      \item Nivel 3
      \begin{enumerate}
        \item Nivel 4
      \end{enumerate}
    \end{enumerate}
  \end{enumerate}
  \item Nivel 1
\end{enumerate}
```

1. Nivel 1
 - a) Nivel 2
 - 1) Nivel 3
 - a' Nivel 4
2. Nivel 1

Para las listas no ordenadas también observamos un cambio en las etiquetas usadas para cada elemento; siendo: cuadrado lleno, viñeta llena, viñeta vacía y al final diamante vacío.

```

\begin{itemize}
  \item Nivel 1
  \begin{itemize}
    \item Nivel 2
    \begin{itemize}
      \item Nivel 3
      \begin{itemize}
        \item Nivel 4
      \end{itemize}
    \end{itemize}
  \end{itemize}
\end{itemize}
\item Nivel 1
\end{itemize}

```

- Nivel 1
 - Nivel 2
 - Nivel 3
 - ◇ Nivel 4
- Nivel 1

Para las listas descriptivas no hay ningún tipo de cambio en la manera en la que se muestran las etiquetas.

```

\begin{description}
  \item[lvl1] Nivel 1
  \begin{description}
    \item[lvl2] Nivel 2
    \begin{description}
      \item[lvl3] Nivel 3
      \begin{description}
        \item[lvl4] Nivel 4
      \end{description}
    \end{description}
  \end{description}
\end{description}
\item[lvl1] Nivel 1
\end{description}

```

```

lvl1 Nivel 1
  lvl2 Nivel 2
    lvl3 Nivel 3
      lvl4 Nivel 4

lvl1 Nivel 1

```

5.3. Cambiando etiquetas para toda la lista

Gracias al paquete `enumitem` y la opción que hemos incluido en el (`shortlabels`) ahora tenemos la posibilidad de cambiar las etiquetas que tienen por defecto nuestras listas. Para eso solo tenemos que colocar como argumento opcional de una lista ordenada el primer elemento de patrón que queremos colocar, este identificará lo mejor que pueda los elementos de la etiqueta y continuará la serie.

```

\begin{enumerate}[(1)]
  \item Primer elemento

  \item Segundo elemento

  \item Tercer elemento
\end{enumerate}

```

- (1) Primer elemento
- (2) Segundo elemento
- (3) Tercer elemento

Los elementos que las listas ordenadas utilizan para identificar patrones son: «1», para las listas con números arábigos; «I» e «i», para las listas con números romanos en mayúscula y minúscula; «A» y «a», para las listas numeradas con letras en mayúscula y minúscula. El resto de elementos se colocarán sin ninguna modificación y de no existir ningún elemento que evoque algún patrón solo se repetirán los elementos que se usaron como argumento opcional.

Para una lista no ordenada también podremos utilizar un argumento opcional para especificar la etiqueta, pero en lugar de continuar un patrón, solo repetirá el argumento que se le da.


```

\begin{itemize}[$\star$]
  \item Primer elemento

  \item Segundo elemento

  \item Tercer elemento
\end{itemize}

```

- ★ Primer elemento
- ★ Segundo elemento
- ★ Tercer elemento

Ya que a las listas descriptivas siempre se les especifica la etiqueta para cada elemento, no se pueden utilizar argumentos opcionales de este estilo en ellas.

5.4. Cambiando etiquetas para un elemento

Además de poder cambiar las etiquetas para toda una lista, como vimos en la subsección anterior, podemos cambiar etiquetas para un elemento particular sin afectar el resto. Esto se consigue con la misma sintaxis que al colocar etiquetas a las listas descriptivas, con el argumento opcional del comando `\item`.

Esto es igual para cualquier tipo de lista

```

\begin{enumerate}
  \item Primer elemento

  \item[$\star$] Segundo elemento

  \item Tercer elemento
\end{enumerate}

```

1. Primer elemento
- ★ Segundo elemento
2. Tercer elemento

Como se puede ver, los elementos de las listas enumeradas no avanzarán de número si se usa una etiqueta personalizada.

Referencias

- [1] Tobi. *\bfseries is to \textbf as WHAT is to \textsf*. 2013. URL: <https://tex.stackexchange.com/a/139592/234538>.
- [2] Javier Bezos López. *enumitem – Control layout of itemize, enumerate, description*. URL: <https://ctan.org/pkg/enumitem>.
- [3] Wikibooks, ed. *L^AT_EX*. 2022. URL: <https://en.wikibooks.org/wiki/LaTeX>.