

# L<sup>A</sup>T<sub>E</sub>X en 10 pasos

## 1. No es un editor WYSIWYG

Latex o L<sup>A</sup>T<sub>E</sub>X (se suele escribir con este baile de letras) es un procesador de textos en el que según se va escribiendo no se va viendo como va a quedar lo escrito, es decir, no es un editor WYSIWYG (*What you see is what you get*). A diferencia de ese tipo de procesadores de texto, como *Word* u *OpenOffice.org Writer*, es necesario explicarle mediante diferentes comandos lo que queremos que haga con el texto, consiguiendo una gran precisión, un ahorro considerable en el tamaño del archivo y algún que otro dolor de cabeza buscando el comando adecuado.

Para poder trabajar con LaTeX es necesario instalar dos programas. Uno es la propia distribución de LaTeX, que en Windows se llama MiKTeX y otro es el editor en el que escribir los textos y los envíe a procesar al MiKTeX. Existen varios y todos buenos: WinEdt, Texmaker o TeXnicCenter. Tú escribes el texto en el editor, le dices que lo envíe al LaTeX y éste compone el texto y te lo devuelve en PDF u otro formato.

## 2. Algunos caracteres son especiales

Un inconveniente de tener que explicarle mediante comandos al programa lo que quieres hacer es que existen algunos caracteres que están reservados para esas tareas y si los quieres escribir es necesario ponerlos mediante un comando. Estos caracteres son:

Signo	Comando	Uso
\	<code>\textbackslash</code>	Introduce a un comando
{	<code>\{</code>	Delimita las propiedades del comando
}	<code>\}</code>	Delimita las propiedades del comando
#	<code>\#</code>	Numera los argumentos de un comando
&	<code>\&amp;</code>	Separa las columnas de una tabla
%	<code>\%</code>	Para comentar en el código fuente
~	<code>\~{}</code>	Evita salto de línea entre dos palabras
\$	<code>\\$</code>	Delimita una ecuación
_	<code>\_</code>	En ecuaciones, subíndice
^	<code>\^{}</code>	En ecuaciones, superíndice

Existen otros caracteres que hay que nombrar mediante un comando bien porque no están en el teclado o bien porque LaTeX necesita que los especifiquemos. Esta es una pequeña lista de ellos:

Signo	Comando	Uso	Signo	Comando	Uso
« y »	<code>&lt;&lt; y &gt;&gt;</code>	Comillas francesas	†	<code>\dag</code>	Daga u obelisco
‘ y ’	<code>\`y ’</code>	Comillas inglesas simples	‡	<code>\ddag</code>	Daga doble
“ y ”	<code>\`y ’ ’</code>	Comillas inglesas dobles	§	<code>\S</code>	Signo de sección
-	<code>-</code>	Guión corto	¶	<code>\P</code>	Calderón
—	<code>---</code>	Raya	®	<code>\textregistered</code>	Marca registrada
—	<code>-\$-\$</code>	Resta	©	<code>\copyright</code>	Copyright
...	<code>\dots</code>	Puntos suspensivos	™	<code>\texttrademark</code>	Trademark
°	<code>\textdegree</code>	Grado	£	<code>\pounds</code>	Libra

### 3. Lo fundamental se define al principio

Escribir un documento LaTeX es muy sencillo. Al principio se tiene que definir las características fundamentales del documento, como el tipo de hoja, el tamaño de la letra por defecto o su estructura. Todo se indica en el siguiente comando:

```
\documentclass[opciones]{estructura}
```

Cuando escribimos un documento largo la estructura es importante. LaTeX permite dividir nuestro trabajo en (de mayor a menor importancia): partes (`\part`), capítulos (`\chapter`), secciones (`\section`), subsecciones (`\subsection`), párrafos (`\paragraph`) y subpárrafos (`\subparagraph`). Para darlas un título tan solo es necesario ponerlo entre llaves a continuación del código de la división:

```
\section{Título}
```

Dependiendo del tipo de documento que queramos escribir, necesitaremos más o menos de estas divisiones por lo que LaTeX contempla cuatro tipos de estructuras diferentes:

- **article**: Es el más usado para artículos cortos, la división más grande que permite es la sección.
- **report**: Además de permitir todas las divisiones, permite añadir un resumen (`\begin{abstract}`) calificar capítulos como apéndices (`\appendix`).
- **book**: Muy parecido al **report** excepto que no deja añadir el resumen y además maqueta el documento como un libro, comenzando los capítulos siempre por página impar.
- **slides**: Para hacer presentaciones de diapositivas.

Las opciones, separadas por comas, que permite añadir LaTeX para caracterizar nuestro documento son:

- **Tamaño de letra**: 10pt, 11pt, 12pt... Si no se dice nada, el tamaño por defecto es 10 puntos.
- **Tamaño del papel**: a4paper, a5paper, b5paper, executivepaper, legalpaper o letterpaper.
- **Número de columnas**: Se puede indicar que ordene el texto en una columna (`onecolumn`) o dos (`twocolumn`).
- **Número de caras**: LaTeX maqueta el texto de la forma más cómoda de leer según queramos usar una (`oneside`) o las dos caras del papel (`twoside`).
- **Orientación**: LaTeX pondrá la hoja en horizontal si así se lo indicamos con (`landscape`).
- **Alineación de las ecuaciones**: Por defecto LaTeX centra las ecuaciones y alinea a la derecha el número de ecuación. Si escribimos (`fleqn`) las ecuaciones se alinearán a la izquierda y (`leqno`) los números de ecuación se alinearán a la izquierda.
- **Comienzo de capítulo**: Se puede forzar, dando igual la estructura que hayamos puesto al principio, que los capítulos comiencen en página impar (`openright`) o en cualquier página (`openany`).
- **Página de título**: Excepto en la estructura de artículo, se le puede indicar que reserve una página para el título (`titlepage`).

Después de definir el tipo de documento hay que cargar las extensiones que necesitemos mediante el comando:

```
\usepackage{nombre}
```

Las extensiones son complementos o modificaciones que le hace la gente a LaTeX. Existen de todo tipo y todas las suele instalar MiKTeX de forma automática cuando las necesita. En mi opinión son fundamentales estas cuatro:

```
\usepackage[latin1]{inputenc}
\usepackage[spanish]{babel}
\usepackage{amsmath}
\usepackage{graphicx}
```

La opción `latin1` de `inputenc` permite a LaTeX interpretar caracteres con acentos, diéresis, eñes, ce cedillas... haciendo la escritura de textos mucho más cómoda porque si no tendríamos que indicarle cada caracter raro mediante un código (por ejemplo: á como `\`a`).

El paquete `babel spanish` le da las normas de silabación del castellano a LaTeX para que divida correctamente las palabras al acabarse el renglón y pasar al siguiente.

`amsmath` carga en la memoria de LaTeX un montón de símbolos matemáticos necesarios para escribir ecuaciones como Dios manda.

`graphicx` aumenta la cantidad de cosas que se pueden hacer con las imágenes.

Otro paquete útil es `eurosym`, que nos permite introducir el símbolo del euro cada vez que escribimos el comando `\euro`.

Si queremos darle un título al documento, indicar el autor y cosas así este es el momento. Se utilizan los comandos:

```
\title{nombre del documento}
\author{nombre del autor}
\date{fecha de creación}
\thanks{agradecimientos}
```

Y ya sólo nos queda lo más difícil, que es escribir el documento. LaTeX entiende que todo el texto que compone el documento está entre los comandos:

```
\begin{document}
bla bla bla documento bla bla bla
\end{document}
```

Así que un poquito de por favor y escribe todo dentro ya que lo que se escriba fuera de esos dos comandos no saldrá en el documento final.

Una cosa que siempre queda muy bien es dejar a LaTeX que componga el título del documento a partir de los datos que le hemos dado en el preámbulo. Para ello, nada más empezar el entorno del documento hay que escribir el comando:

```
\maketitle
```

## 4. Unidades de medida en LaTeX

Muchos comandos requieren que se les indique tamaños para poder trabajar. LaTeX admite varias unidades de medida. Algunas son absolutas:

Símbolo	Nombre	Descripción
sp	Scaled point	Es la unidad más pequeña de TeX (65536 sp = 1 pt)
pt	Punto	Equivale a 0,351 mm o 1/(72,27) pulgadas
bp	Punto Postscript o grande	Equivale a 0,353 mm o 1/72 pulgadas
dd	Punto Didôt	Equivale a 0,376 mm o 1/72 de la pulgada francesa
mm	Milímetro	Equivale a 2,845 puntos
pc	Pica	Equivale a 4,218 mm o 12 puntos
cc	Cicero	Equivale a 4,531 mm o 12 puntos Didôt
cm	Centímetro	Equivale a 10 mm o 2,371 picas
in	Pulgada	Equivale a 25,4 mm o 72,27 puntos o 6,022 picas

Y otras son relativas al tipo de letra en uso:

Símbolo	Nombre	Descripción
ex	Equis	Altura de la letra x
em	Eme	Ancho de la letra M
mu	Unidad matemática	18 mu equivalen a una eme

Para usarlas es tan sencillo como poner el número seguido de la unidad sin espacio entre ellos. Por ejemplo, 14 milímetros sería `14mm`.

## 5. LaTeX se encarga de maquetar los párrafos

Cuando enviamos un documento a LaTeX para que lo componga, él se encarga de maquetarlo siguiendo una complicada lista de normas para hacer el documento final lo más atractivo y armónico posible. Por defecto, LaTeX justifica los párrafos, separando palabras en dos renglones si es necesario. Si queremos cambiar la alineación del texto, podemos empezar y acabar un texto de la siguiente forma:

<code>\begin{flushleft}</code>	<code>\begin{center}</code>	<code>\begin{flushright}</code>
Texto alineado a la izquierda	Texto centrado	Texto alineado a la derecha
<code>\end{flushleft}</code>	<code>\end{center}</code>	<code>\end{flushright}</code>

LaTeX también distingue entre un salto de línea y un salto de párrafo. La diferencia entre ambos es que un salto de párrafo sólo se da cuando se cambia de concepto en una exposición, mientras que el salto de línea solo se da cuando se quiere expresar una pausa más larga que la de un punto. En LaTeX, el salto de línea se hace poniendo dos barras invertidas seguidas (`\\`) mientras que el salto de párrafo se hace dejando un renglón en blanco (pulsando dos veces la tecla enter o la tecla intro). Si pulsamos sólo una vez la tecla enter LaTeX lo interpreta como un espacio. Si la pulsamos más de dos veces, LaTeX lo interpreta como un simple salto de párrafo, da igual que hayas pulsado la tecla mil veces.

Si queremos aumentar el espacio entre dos párrafos de manera puntual se puede usar el comando

`\vspace{tamaño}`

que separa los párrafos el espacio que le digamos.

Si lo que queremos es modificar el espacio entre todos los párrafos del documento, en el preámbulo (antes de `\begin{document}`) se puede poner el siguiente comando:

`\parskip=tamaño`

De igual forma pasa con los espacios entre palabras, LaTeX sólo pone un espacio entre dos palabras por mucho que pulsemos la tecla espaciadora. Si queremos poner un espacio más amplio podemos usar el comando:

`\hspace{longitud}`

Existe otro salto que es el salto de página. Latex contempla dos comandos para tal menester.

`\newpage`

para un salto normal y corriente a la siguiente página y

`\cleardoublepage`

para un salto hacia la siguiente página impar.

Las normas de estilo recomiendan que el comienzo de un párrafo esté sangrado. La longitud del sangrado que añade LaTeX por defecto al comienzo de un párrafo se puede modificar escribiendo en el preámbulo del documento el comando:

`\parindent=tamaño`

Incluso se puede eliminar en un momento determinado escribiendo al comienzo del párrafo:

`\noindent`

También se puede cambiar el interlineado del texto que por defecto es 1. Para ello, en el preámbulo es necesario redefinir el valor mediante el código:

`\renewcommand*{\baselinestretch}{cantidad}`

donde `cantidad` puede ser cualquier valor decimal.

Por último, LaTeX elige los márgenes adecuados para cada tamaño de papel y letra para hacer la lectura del texto lo más cómoda posible. Esos tamaños pueden ser modificados en el preámbulo mediante el comando:

`\addtolength{propiedad}{tamaño}`

donde las propiedades que se pueden modificar una cantidad positiva si quieres aumentarla y negativa si quieres reducirla. Todas ellas están recogidas en la siguiente tabla:

Propiedad	Significado
<code>\textwidth</code>	Ancho del texto
<code>\textheight</code>	Altura del texto
<code>\hoffset</code>	Posición inicial del margen lateral izquierdo
<code>\voffset</code>	Posición inicial del margen superior
<code>\evensidemargin</code>	Posición inicial del margen exterior en páginas pares
<code>\oddsidemargin</code>	Posición inicial del margen exterior en páginas impares

## 6. Jugar con los tipos es preciso

LaTeX fue creado para ser una imprenta profesional personal. Es por eso que tiene todos las modificaciones que se le pueden hacer a una letra:

Comando	Nombre
<code>\textbf{}</code>	<b>Negrita</b>
<code>\textit{}</code>	<i>Cursiva</i>
<code>\textsl{}</code>	<i>Inclinada</i>
<code>\underline{}</code>	Subrayada
<code>\emph{}</code>	<i>Enfatizada</i>
<code>\textsc{}</code>	VERSALITA
<code>\$^\text{}</code>	Superíndice
<code>\$_\text{}</code>	Subíndice
<code>\texttt{}</code>	Máquina de escribir
<code>\textsf{}</code>	Palo seco

También se puede cambiar el tamaño de la letra mediante los siguientes comandos que están ordenados de menor a mayor tamaño, pasando por el tamaño normal que corresponde al que declaramos en el preámbulo:

`\tiny < \scriptsize < \footnotesize < \small < \normalsize < \large < \Large < \LARGE < \huge < \Huge`

La forma de insertar estos comandos en el documento LaTeX es diferente para que se puedan combinar con otros códigos como, por ejemplo, los de cambiar la forma de las letras. Estos comandos se ponen entre llaves, al comienzo de la frase que se quiere cambiar:

`{\Large bla bla bla texto}`

## 7. Uso de imágenes

Poner una imagen en LaTeX es tan sencillo como escribir el comando:

```
\includegraphics{opciones}{archivo}
```

Donde en `archivo` hay que poner la dirección de la imagen en el ordenador excepto si se encuentra en la misma carpeta en la que se aloja el propio archivo LaTeX.

Muchas de las opciones están relacionadas con los tamaños de la imagen:

- `width=tamaño`: Usando este comando podemos indicar el ancho que queremos que tenga la imagen, que se escalará para cumplirlo.
- `height=tamaño`: De igual forma se puede hacer con la altura de la imagen.
- `keepaspectratio`: Si se pone que es `=false`, cuando cambiemos el tamaño de una imagen no respetará su proporción entre el alto y el ancho.
- `scale=número`: Muestra la imagen en un tamaño proporcional a la cantidad que le pongamos. Por ejemplo, si ponemos 2, la imagen tendrá el doble de su tamaño original.
- `clip`: Si se declara como `=true`, la imagen será recortada según las dimensiones declaradas en el resto de opciones.
- `draft`: Otro parámetro lógico. Si decimos que es `=true`, en lugar de la imagen aparece un cuadrado con su mismo tamaño donde pone el nombre del archivo. Muy útil para cuando imprimas borradores, que es lo que significa `draft` en inglés.

Poner una imagen es así de fácil, aunque también tenemos la opción de incluir una imagen como hacen los profesionales, dentro de un objeto flotante.

Un objeto flotante es un espacio que reserva el programa en la hoja para incluir una imagen o una tabla con su leyenda. LaTeX siempre ubica el objeto flotante en el lugar en el que es más cómodo para la vista, por lo que hay que tener cuidado al referirnos a la imagen porque puede que no esté donde nos esperamos. El código para incluir una imagen dentro de un objeto flotante es:

```
\begin{figure}[Posición]
\includegraphics{opciones}{archivo}
\caption{Leyenda}
\end{figure}
```

Como hemos dicho, LaTeX coloca la imagen en el lugar más cómodo para la vista. Sin embargo, nosotros podemos sugerirle la posición donde queremos que vaya en lo que el código he llamado `Posición`. Si entre esos corchetes se pone `h` (de *here*), LaTeX intentará colocar la imagen en el mismo lugar que ocupa en el código fuente. Si se pone `t` o `b` (de *top* y *bottom*), la figura se colocará, respectivamente, en la parte superior o inferior de la página que quede más cerca. Por último también podemos indicarle `p` (de *page*) para que la figura vaya a una página nueva en la que sólo habrá imágenes. Si por algo tenemos mucho interés en colocar una figura en lugar determinado y LaTeX ignora nuestra sugerencia, podemos obligarle a que nos haga caso añadiendo a la letra de posición una exclamación, por ejemplo `h!`.

Los objetos flotantes también permiten añadir una leyenda para que el autor explique lo que contiene. Tan sólo hay que añadir el comando `\caption` y, entre llaves, la leyenda que queramos poner.

## 8. Confección de tablas

Escribir una tabla en LaTeX es muy intuitivo ya que sólo utiliza dos símbolos como separadores: `&` para separar columnas y `\\` para separar filas. Así, se declara mediante el comando `tabular` que comienza una tabla y se va escribiendo fila por fila, como muestra el código:

```

\begin{tabular}{número y tipo de columnas}
  Celda & Celda & Celda \\
  Celda & Celda & Celda \\
\end{tabular}

```

El comando `tabular` exige que se indiquen el número de columnas y el alineamiento de todas las celdas de cada columna, y eso es tan fácil como poner una letra por cada columna que queremos que haya. ¿Qué letra?. Depende del formato de la columna: `l` si queremos que todas las celdas de la columna estén alineadas a la izquierda, `c` si queremos que estén centradas, `r` si queremos que estén alineadas a la derecha y `p{ancho}` si queremos forzar a LaTeX a componer una la columna con un ancho determinado (si no el ancho sería el de la celda con el texto más largo).

Si un queremos que una celda ocupe varias columnas existe el siguiente código:

```

\multicolumn{número}{tipo}{texto de la celda}

```

que se pone cuando toque escribir la celda determinada. `número` es el número de columnas que va ocupar y `tipo` es el alineamiento de la celda.

Respecto a dibujar los bordes de las celdas con rayas, LaTeX te da total flexibilidad para ponerlas donde y cuando quieras. Para las rayas verticales tan sólo es necesario añadir junto a la letra que representa la columna en la declaración inicial una pleca (`|`). Si se la ponemos a la izquierda, la columna tendrá una raya vertical a la izquierda y si la ponemos entre dos letras, aparecerá una raya vertical entre las dos columnas representadas por las dos letras. Por ejemplo, una tabla con tres columnas alineadas a la izquierda con rayas a ambos lados para cada columna: `{|l|l|l|}`. Si queremos que la raya sea doble se puede poner de igual modo dos placas (`||`).

Para las rayas horizontales se utiliza el comando `\hline`, que se coloca al final de cada fila (después del separador `\\`) que queremos que tenga un borde inferior. Para la raya superior a la primera fila se puede poner el comando `\hline` después de la declaración del número y el tipo de columnas. Si queremos una doble raya horizontal se puede utilizar dos veces el comando `\hline`.

Las tablas también se pueden componer en forma de un objeto flotante. El código es:

```

\begin{table}[Posición]
\begin{tabular}{número y tipo de columnas}
  celdas celdas y más celdas
\end{tabular}
\caption{Leyenda}
\end{table}

```

donde todo es igual que en los objetos flotantes de las figuras.

## 9. Ecuaciones. La gran ventaja de LaTeX

Dado que era su objetivo inicial, LaTeX sirve para escribir ecuaciones matemáticas de forma sencilla y rápida de forma que queden realmente bellas. Existen tres modos diferentes de escribir una ecuación en LaTeX:

<code>\$</code> Ecuación entre texto <code>\$</code>	<code>\begin{equation}</code> Ecuación numerada <code>\end{equation}</code>	<code>\begin{equation*}</code> Ecuación sin numerar <code>\end{equation*}</code>
--	---	--

- Entre símbolos de dólar: Se utiliza cuando hay que poner una ecuación puntual entre el texto.
- `{equation}`: Usando este código, LaTeX coloca la ecuación en un párrafo a parte, centrada y numerada a su derecha.
- `{equation*}`: Exactamente igual que la anterior, excepto que la ecuación no está numerada y no se tiene en cuenta al numerar el resto de ecuaciones

Como mandan los cánones tipográficos, LaTeX pone en itálica todo el texto que esté dentro de una ecuación. Si queremos introducir texto que no debería estar en cursiva en una ecuación, como, por ejemplo, unidades de medida, es necesario utilizar el comando `\text{texto}`.

A la hora de escribir símbolos, recuerda que es muy importante instalar la extensión `amsmath` para asegurarnos que están todos los símbolos importantes. Cada símbolo tiene su propio comando que no es necesario memorizar puesto que la mayoría de los programas de edición de LaTeX incluyen tablas para insertarlos. De todas formas, existen en internet multitud de listas de símbolos muy útiles (la más completa: *Comprehensive LaTeX symbol list*).

Aun así, existen unos cuantos comandos especiales que vamos a resumir en la siguiente tabla:

Nombre	Código	Ejemplo
Fracciones	<code>\frac{numerador}{denominador}</code>	<code>\frac{a+b}{c}</code> $\frac{a+b}{c}$
Subíndice	<code>-</code>	<code>a_n</code> $a_n$
Superíndice	<code>^</code>	<code>a^2</code> $a^2$
Raíz cuadrada	<code>\sqrt{radicando}</code>	<code>\sqrt{2}</code> $\sqrt{2}$
Raíces en general	<code>\sqrt[índice]{radicando}</code>	<code>\sqrt[3]{2}</code> $\sqrt[3]{2}$
Coficiente binomial	<code>\binom{superior}{inferior}</code>	<code>\binom{1}{6}</code> $\binom{1}{6}$
Sumatorio	<code>\sum_{desde}^{hasta}</code>	<code>\sum_{i=1}^n</code> $\sum_{i=1}^n$
Productorio	<code>\prod_{desde}^{hasta}</code>	<code>\prod_{i=1}^n</code> $\prod_{i=1}^n$
Integral definida	<code>\int_{desde}^{hasta}</code>	<code>\int_0^{\infty}</code> $\int_0^{\infty}$
Int. doble definida	<code>\iint_{desde}^{hasta}</code>	<code>\iint_0^{\infty}</code> $\iint_0^{\infty}$
Int. triple definida	<code>\iiint_{desde}^{hasta}</code>	<code>\iiint_0^{\infty}</code> $\iiint_0^{\infty}$
Límite	<code>\lim_{x}</code>	<code>\lim_{x \rightarrow \infty}</code> $\lim_{x \rightarrow \infty}$

Cuando en una ecuación en la que hay una fracción u otro elemento que tenga una altura especial se utiliza un paréntesis, un corchete, unas llaves, etc. estos no cambiarán su tamaño quedando muy pequeños en comparación con el elemento. Para que ajusten su altura a la de la fracción o el elemento es necesario utilizar los comandos:

`\left§` `\right§`

Donde § puede ser: (, {, [, <, sus opuestos ), ], }, ) o cualquier otro de las decenas de delimitadores que existen en las tablas de LaTeX.

Para escribir una matriz o un determinante también necesitamos un comando especial. Dentro del entorno matemático se escribe el siguiente código, muy similar al de las tablas:

```
\begin{array}{número y tipo de columnas}
  Elemento & Elemento & Elemento \\
  Elemento & Elemento & Elemento \\
\end{array}
```

Luego tan sólo es necesario declarar con un delimitador si se trata de una matriz ( ) o de un determinante | |. El comando `array` es muy útil también para escribir varias ecuaciones en diferentes filas, como, por ejemplo, en un sistema de ecuaciones. Estas ecuaciones, dentro del ámbito `\begin{equation} \end{equation}` están numeradas como si fuesen una sola. Si queremos escribir varias ecuaciones dentro del mismo entorno y que cada una de ellas tenga su propia numeración es necesario recurrir al código:

```
\begin{eqnarray}
  Ecuación 1 \\
  Ecuación 2 \\
\end{eqnarray}
```

## 10. LaTeX te lo ordena todo

Al componer un documento, LaTeX va contando y almacenando en su memoria todos los capítulos, referencias, figuras, tablas, etc. que hemos escrito permitiéndonos sacar partido de ello. Lo más habitual es crear un índice o tabla de contenidos con la lista de partes, capítulos, secciones y subsecciones (dependiendo de la clase, porque por ejemplo, en `article` no hay capítulos) de las que está compuesto el documento. Para ello, sólo hay que escribir el código:

```
{\tableofcontents}
```

en el lugar donde queramos insertar el índice.

LaTeX pone el título de la división en el índice. Si por algo queremos que ponga otra cosa, cuando se nombra la división se puede poner el nombre que queramos que figure en el índice entre corchetes:

```
\section[Título para el índice]{Título de la división}
```

De igual modo, se puede crear índices de las figuras y las tablas del documento. Su comando respectivo es:

```
{\listoffigures}  
{\listoftables}
```

En estos índices aparece el número de figura o tabla que LaTeX da por defecto y la leyenda o descripción que habíamos puesto en `\caption`. Como antes, esto se puede cambiar utilizando el siguiente comando dentro del objeto flotante:

```
\caption[Título para el índice]{Leyenda}
```

Introduciendo etiquetas mediante el comando `\label{nombre de la etiqueta}` al comienzo de un capítulo, sección u otra división, en un objeto flotante o en una ecuación podemos luego en el texto referirnos a la estructura etiquetada mediante el comando `\ref{nombre de la etiqueta}` que será sustituido por el número que LaTeX asignó al objeto citado. Otra opción es el comando `\pageref{nombre de la etiqueta}` que nos permite referirnos a la página en la que se encuentra.

Las citas bibliográficas es otra de las cosas por la que no te tienes que preocupar trabajando con LaTeX. La forma tradicional de escribir la bibliografía es utilizando el siguiente comando cuando queramos que aparezca en nuestro trabajo:

```
\begin{thebibliography}{número}  
\bibitem[nombre alternativo]{nombre de la etiqueta} Cita bibliográfica  
Bibliografía  
y más bibliografía  
\end{thebibliography}
```

Es necesario que pongamos un `número` entre llaves después de abrir el entorno `thebibliography`. El número de cifras que tenga ese número es el espacio que LaTeX va a reservar en su memoria para la numeración de la bibliografía, por lo que tiene que ser igual o más largo que el de el número o nombre alternativo más largo de nuestra bibliografía.

Las citas se introducen dentro del entorno por medio del comando `\bibitem`. Nosotros les damos el formato que queramos y también las ponemos una etiqueta entre llaves para citarla cuando queramos en el texto por medio del comando `\cite{nombre de la etiqueta}`. LaTeX numerará las citas bibliográficas según vayan apareciendo en el texto. Si queremos que en lugar de un número aparezca otra cosa (no será tenido en cuenta en la numeración), tan sólo la ponemos entre corchetes después del comando como muestran los códigos de arriba.

Existe otra forma de escribir la bibliografía mucho más cómoda cuando estamos trabajando con una gran cantidad de citas. Consiste en crear un archivo `.bib` separado del de nuestro documento y guardado en la misma carpeta en que estarán introducidas todas las citas bibliográficas con su

etiqueta y con la información desgranada en `author`, `title`, `year`, etc. En la práctica todo el mundo recurre a programas independientes que se encarguen de escribir y de gestionar ese archivo como, por ejemplo, *JabRef*. Una vez que tenemos toda la bibliografía en ese archivo, tan sólo tenemos que ir citando las referencias con el comando `\cite{nombre de la etiqueta}` y dónde queramos poner la lista de la bibliografía introducir el código:

```
\bibliography{nombre del archivo}
\bibliographystyle{estilo}
```

Escribir de esta forma la bibliografía tiene la ventaja de que se puede aplicar un estilo determinado de forma automática a todas las citas del documento mediante el comando `\bibliographystyle{estilo}`. Existen multitud de estilos, algunos de ellos oficiales, para escribir las citas con un formato determinado para cada uno de sus apartados. Utilizar diferentes estilos también te permite modificar la ordenación de las citas (de forma alfabética, por fecha, etc.) o la forma de referirte a ellas en el texto (nombre del autor y año, un código determinado, número con otro formato, etc.). Existen cuatro estilos estándar:

- `plain`: Citas numeradas alfabéticamente.
- `unsrt`: Citas numeradas según van apareciendo en el texto.
- `alpha`: Citas identificadas por el nombre del autor y el año de publicación.
- `abbrv`: Citas numeradas alfabéticamente pero con los nombres de los autores y de las revistas abreviados.

#### **Mucha más información y fuentes:**

Estos son los libros que he consultado para escribir esta introducción:

El libro de LaTeX, Bernardo Cascales, Ed. Pearson Educación, Madrid, 2003.

The not so short introduction to LaTeX<sub>2 $\epsilon$</sub> , Tobias Oetiker,

<http://tobi.oetiker.ch/lshort/lshort.pdf>

Son muy buenas referencias para saber hacer más cosas con LaTeX. Ante una duda puntual, la principal fuente de información sobre cualquier tema de LaTeX es internet, que está lleno de foros, wikis y tutoriales. Como cualquiera puede contribuir con una extensión al programa también se puede encontrar ahí muchas soluciones a problemas habituales. LaTeX es un programa muy potente y se le puede sacar mucho jugo.