

### 1. Generalidades

Todo documento L<sup>A</sup>T<sub>E</sub>X posee la siguiente estructura:

- **Preámbulo:** Declaraciones de carácter GLOBAL que afectan a la totalidad del documento
  - `\documentclass[opciones]{tipo_de_documento}` → **OBLIGATORIA**; éste debe de ser además el primer comando del documento. Mediante ésta declaración indicamos a L<sup>A</sup>T<sub>E</sub>X que tipo de documento (book, article, report, letter...) queremos escribir
  - `\usepackage[opciones]{paquete}` → carga de paquetes con utilidades (para incluir gráficos, texto en color, presentaciones, etc...)
  - Otras declaraciones: Interlínea, formato de página, fuentes, etc...
- **Cuerpo:** Todo lo comprendido entre `\begin{document}` y `\end{document}`, es decir, el documento propiamente dicho.

En la página siguiente puede verse un ejemplo de documento simple, tomando la forma de la clase `article`.

Existen ciertas reglas generales a la hora de escribir texto:

- **Texto alineado y centrado:** El texto se alinea y justifica automáticamente, según medidas predeterminadas o impuestas por nosotros; IMPORTANTE: Toda medida predeterminada es ajustable
- **Los espacios se ignoran:** Da igual separar las palabras con 1 espacio o varios. Para aumentar el espacio de separación entre palabras se usa: `\` seguido de espacio
- **Punto y aparte:** Dejar una línea en blanco equivale a cambiar de párrafo (punto y aparte). Lo mismo se obtiene con `\par`. **Ojo!** Nótese que dejar *varias* líneas en blanco es igual a dejar una: el efecto es el mismo, i.e., comienzo de un nuevo párrafo.
- **Cambio de línea:** Se puede cambiar de línea, *sin cambiar de párrafo*, usando `\\`

Un texto dado puede ajustarse a cualquiera de los dos lados de la página ó al centro:

- Para centrar un párrafo se utiliza el *entorno* **center**

```
\begin{center}
Texto a centrar
\end{center}
```

Ejemplo:

```
\begin{center}
El ingenioso hidalgo\\
D. Quijote de la Mancha\\[0.3cm]
Miguel de Cervantes Saavedra
\end{center}
```

```
El ingenioso hidalgo
D. Quijote de la Mancha
Miguel de Cervantes Saavedra
```

## Ejemplo 1:

```
\documentclass[a4,11pt]{article}
\usepackage[latin1]{inputenc}
\usepackage[T1]{fontenc}
\usepackage[spanish]{babel}
\renewcommand{\shorthandsspanish}{}

\title{Documento Fuente \LaTeX{}}
\author{Perico de los Palotes}
\date{}

\begin{document}

\maketitle
\tableofcontents

Eejmplo de documento \LaTeX\ de la clase
{\ttfamily article} con una estructura
reducida. Ésta incluye secciones,
subsecciones y una referencia cruzada.

\section{Primera sección}\label{primera}
Una primera sección con una fórmula y una lista.

\subsection{Fórmula}
Una ecuación:  $\frac{\partial f}{\partial x} = \frac{\partial f}{\partial y}$ 

\subsection{Listas}

Una lista de ítems señalados con una marca:
\begin{itemize}\itemsep=0pt
\item Primer ítem
\item Segundo ítem
\item Tercer ítem
\end{itemize}

\section{Segunda sección}
Esta sección complementa a la sección \ref{primera}
incluyendo ejemplos de tablas escritas en \LaTeX.

\end{document}
```

## Documento Fuente L<sup>A</sup>T<sub>E</sub>X

Perico de los Palotes

### Índice

1. Primera sección	1
1.1. Fórmula	1
1.2. Listas	1

2. Segunda sección	1
Eejmplo de documento L <sup>A</sup> T <sub>E</sub> X de la clase <code>article</code> con una estructura reducida. Ésta incluye secciones, subsecciones y una referencia cruzada.	

### 1. Primera sección

Una primera sección con una fórmula y una lista.

#### 1.1. Fórmula

Una ecuación:  $\frac{\partial f}{\partial x} = \frac{\partial f}{\partial y}$

#### 1.2. Listas

Una lista de ítems señalados con una marca:

- Primer ítem
- Segundo ítem
- Tercer ítem

### 2. Segunda sección

Ésta sección complementa a la sección 1 incluyendo ejemplos de tablas escritas en L<sup>A</sup>T<sub>E</sub>X.

- Para alinear a los lados, tenemos los entornos **flushleft** y **flushright**

```
\begin{flushleft}
Texto
\end{flushleft}
```

```
\begin{flushright}
Texto
\end{flushright}
```

Probando  
la forma de alinear  
por la *izquierda*

Probando  
la forma de alinear  
por la *derecha*

- Para textos pequeños, menores que una línea, se pueden utilizar, respectivamente:

`\leftline{Texto}`

`\centerline{Texto}`

`\rightline{Texto}`

Es interesante el hecho de que existen en  $\text{\LaTeX}$  multitud de comandos que poseen ambas versiones, una “corta”, de tipo:

`\comando{argumento}`

donde el comando afecta a un texto pequeño (argumento), y otra “larga”, de tipo:

`\begin{entorno} Texto largo \end{entorno}`

Alternativamente, también existe otra método para éste último procedimiento, incluyendo el comando dentro de un **grupo**:

`{\comando Objeto extenso}`

donde los *delimitadores* `{` y `}` definen el grupo de objetos a los que afectará el comando.

Un elemento muy importante a la hora de contruir un documento son las longitudes que se utilizan para delimitar diferentes distancias entre objetos. Existe una amplia variedad de unidades de longitud que podemos utilizar:

Unidades de longitud:
-----------------------

Medidas absolutas		
pt	punto	1pt $\approx$ 0.35146 mm
pc	pica	1pc = 12 pt
in	pulgada	1in = 72.27 pt = 2.54 cm
cm	centímetro	
mm	milímetro	1 mm = 2.845 pt
dd	didot	1157 dd = 1238 pt
cc	cíbero	1 cc = 12 dd
Medidas relativas		
em	aprox. la anchura de una ‘M’ de la fuente en curso	
ex	aprox. la anchura de una ‘x’ de la fuente en curso	

Las medidas relativas “em” y “ex” dependen del tamaño de letra en curso que estemos utilizando. Existen muchos otros ejemplos de uso de medidas relativas; en general,  $\text{\LaTeX}$  maneja internamente una amplia cantidad de longitudes que en principio desconocemos o que incluso son *elásticas*. Es un procedimiento frecuente, y bastante útil, el manejar longitudes relativas. Por ejemplo, si queremos delimitar la anchura de un objeto cualquiera como la mitad de la anchura del texto en la página, definida a través de la variable `\textwidth`, podemos utilizar la longitud relativa `0.5\textwidth`.

## 2. Manejo de espacios horizontales y verticales

Hay varias formas de añadir espacios verticales entre diversos objetos:

- `\` y `\newline`  $\longrightarrow$  Análogos: salto de línea simple. En este punto es bueno recordar la diferencia con `\par`: éste último *cambia de párrafo*, mientras que los anteriores simplemente terminan bruscamente un línea y pasan a la siguiente, sin comenzar nuevo párrafo.

- `\[Salto]`  $\longrightarrow$  Espacio vertical de longitud *Salto*
- `\vspace{Salto}`  $\longrightarrow$  Análogo al anterior; nótese que no tiene efecto *al comienzo de una página*, en tal circunstancia se puede usar el comando análogo `\vspace*{Salto}`, que evita este problema
- Espacios verticales de longitud predefinida:
  - `\bigskip`  $\longrightarrow$  approx. 1 línea en blanco
  - `\medskip`  $\longrightarrow$  approx. 1/2 del espacio de una línea en blanco
  - `\smallskip`  $\longrightarrow$  approx. 1/4 del espacio de una línea en blanco

Para añadir espacios horizontales podemos utilizar:

- `\hspace{longitud}`  $\longrightarrow$  Espacio horizontal de extensión *longitud*
- `\hspace*{longitud}`  $\longrightarrow$  Igual, válido *al comienzo de una línea*
- Predefinidos:
  - `\_`  $\longrightarrow$  un espacio entre palabras
  - `\enskip`  $\longrightarrow$  medio “em”
  - `\quad`  $\longrightarrow$  un “em”
  - `\qquad`  $\longrightarrow$  dos “em”

Si queremos saltar de página, podemos utilizar tanto `\newpage` como `\clearpage`, que indican a  $\text{\LaTeX}$  que se debe finalizar la página en curso y comanzar una nueva. Nótese que `\clearpage` posee un significado especial: en el caso de que haya *elementos flotantes* (como figuras y tablas, para los cuales  $\text{\LaTeX}$  se encarga por si solo de buscar la ubicación más adecuada) pendientes de ubicar, tales elementos se imprimirán inmediatamente en la página siguiente, formada por sólo tablas y gráficas. El texto se reiniciará entonces otra página después.

Podemos modificar la separación estándar entre párrafos (por defecto igual al valor de la separación entre líneas ó interlínea) manipulando la longitud `\parskip` del siguiente modo:

`\parskip=5mm` (por ejemplo)

lo cual añadirá 5mm a la mencionada separación entre párrafos por defecto. Dependiendo de **dónde** coloquemos éste comando variará su efecto: si se encuentra en el preámbulo, afectará a todo el documento, mientras que dentro del cuerpo del documento sólo afectará a lo que venga a continuación. **OJO!** Dado que el comando es de carácter local, si está incluido en un grupo (o sea, dentro de `{.....}`), sólo actuará dentro de tal grupo.

La distancia entre líneas (interlínea) también es modificable, situando en el preámbulo (**OJO!**, y sólo en el preámbulo, fuera de ahí la instrucción no funciona) la instrucción:

`\renewcommand*{\baselinestretch}{Número}`

lo que **escala** la interlínea por la cantidad Número (un valor de 2.0 equivaldría a doble espaciado). Si se quiere cambiar el valor de la interlínea en distintas partes del documento, se puede utilizar el paquete `setspace`, con la sintaxis:

`\usepackage[espaciado]{setspace}`

lo cual hace el valor de la interlínea en todo el documento igual a **espaciado**. Los posibles valores son: `singlespacing`, `onehalfspacing` y `doublespacing`, que equivalen respectivamente a un valor de `baselinestretch` de 1, 1.5 y 2 (si no se incluye la opción `espaciado`, el valor por defecto es `singlespacing`). Después, en el cuerpo del documento, puede variarse a voluntad la interlínea mediante los comandos: `\singlespacing`, `\onehalfspacing` y `\doublespacing`.

## 2.1. Sangría de párrafos

Para delimitar y hacer más visibles los párrafos, es habitual hacer un sangrado (dejar un pequeño espacio en blanco) al comienzo de cada párrafo. Ésta magnitud es modificable mediante los siguientes comandos:

- `\noindent` → Situado *al comienzo* de un párrafo dado, elimina la sangría de ése párrafo (y sólo de ese).
- `\parindent=longitud` → Redefine la sangría estándar a principio de párrafo como `longitud`. Tiene un comportamiento análogo a `\parskip=longitud` (cambios globales en el preámbulo, y locales dentro del documento).

### Ejercicio 1:

Crear un documento `article`, con la instrucción `\parskip=10pt` en el preámbulo, compilar lo siguiente, y observar el resultado:

Esto es un ejemplo para ver que tanto `vspace` como los separadores con doble backslash introducen separaciones dentro de un párrafo, pero no son capaces de cambiar de párrafo.

```
\\[0.5cm] \vspace{0.5cm} \\ \\
```

Para cambiar de párrafo, es necesario o bien introducir una línea en blanco, o bien utilizar el comando equivalente

```
\par
```

que si cambia de párrafo, lo cual puede verse debido a que este nuevo párrafo si posee sangría.

## 2.2. Silabeo

Puede ocurrir que los algoritmos de silabeo de  $\text{\LaTeX}$  no funcionen correctamente y que al cambiar de línea se rompa una palabra de forma inadecuada. Para evitarlo hay dos alternativas:

- Utilizar la instrucción `\hyphenation{lista de palabras}` en el preámbulo; por ejemplo, `\hyphenation{For-tran fi-che-ro}` sólo permitirá la división de las palabras “fortran” y “fichero” por los lugares indicados. Nótese que no se permiten caracteres con acentos ó símbolos en el argumento, y que no se hacen distinciones entre las letras mayúsculas y minúsculas de las palabras en la orden.
- Fuera del preámbulo, se puede utilizar la instrucción:  $\boxed{\backslash-}$  Ésta es válido utilizarla en palabras con acentos ó símbolos, por ejemplo: `te\lé\fo\no`

### 3. Escribiendo en castellano

El idioma por defecto de  $\text{\LaTeX}$  es el inglés. Esto quiero decir que, de forma estándar, no se reconocen los caracteres especiales como Ñ, letras acentuadas, etc... Además, definiciones por defecto, como títulos para capítulos, fechas, etc, estarán en inglés. Pruébese por ejemplo lo siguiente:

```
\documentclass{article}
\begin{document}
Saludos desde \LaTeX. Haciendo una compilación de prueba, de texto
en español, para ver si todo funciona.

Escrito y compilado el día \today.
\end{document}
```

y se verá que los acentos y ñ desaparecen y que la fecha (comando `\today`) se imprime en inglés. Para solucionar tales problemas se pueden incluir los siguientes paquetes en el preámbulo del documento:

- `\usepackage[latin1]{inputenc}` → Para que  $\text{\LaTeX}$  entienda los símbolos del teclado español. Con este paquete podemos teclear directamente símbolos del teclado que serán reconocidos por el compilador <sup>1</sup>
- `\usepackage[T1]{fontenc}` → Para que utilice nuestros tipos acentuados, en vez de construirlos con METAFONT
- `\usepackage[spanish]{babel}` → Reglas españolas para división de sílabas, traducción de comandos, etc...
- `\renewcommand{\shorthandsspanish}{}`  → Desactiva métodos taquigráficos en español (que pueden molestarnos)

Podemos utilizar las declaraciones anteriores como cabecera estándar para escribir textos en español. Llegado este punto, es bueno destacar que existen alternativas para escribir caracteres acentuados sin utilizar el teclado español y la codificación “latin1” en el paquete `inputnc`. Ésto puede ser útil si, por ejemplo, nos encontramos en el extranjero ó carecemos del teclado adecuado. Para ello, desactivaríamos las declaraciones `\usepackage[latin1]{inputenc}` y `\usepackage[T1]{fontenc}`, haciendo uso de las instrucciones expuestas en la siguiente tabla para conseguir caracteres acentuados:

ò	\‘o	ó	\’o	ô	\^o	õ	\~o
ō	\=o	ô	\.o	ö	\"o	ç	\c c
õ	\u o	õ	\v o	õ	\H o	ø	\d o
o	\b o	oo	\t oo				

**IMPORTANTE:** Las letras *i* y *j* necesitan un tratamiento especial, dado que no deben tener sus puntos antes de ser acentuadas. Su eliminación se consigue con los comandos `\i` y `\j`, respectivamente. Así, para obtener, por ejemplo:

---

<sup>1</sup>a excepción del símbolo del euro (€); véase mas adelante

Él está aquí

se debería escribir `\'{E}l est\{a} aqu\{i}`

Para otros símbolos pertenecientes a diversos idiomas véase la siguiente tabla:

œ	<code>\oe</code>	Œ	<code>\OE</code>	æ	<code>\ae</code>	Æ	<code>\AE</code>
å	<code>\aa</code>	Å	<code>\AA</code>				
ø	<code>\o</code>	Ø	<code>\O</code>	ı	<code>\l</code>	Ł	<code>\L</code>
ß	<code>\ss</code>						
ı	<code>!'</code>	ı	<code>?'</code>				

## 4. Más sobre signos ortográficos

En esta sección describiremos diversos comandos de utilidad a la hora de escribir un texto (comillas, guiones, ordinales, etc...)

### 4.1. Los diez caracteres reservados

En  $\text{\LaTeX}$  existen 10 caracteres especiales que el sistema utiliza para distintos propósitos, a saber:

`\ { } # & % ~ $ _ ^`

los cuales tienen los siguientes usos:

`\` Indicador de comando.

`{ }` Delimitadores de grupos.

`#` Nombra los argumentos de un comando.

`&` Separa columnas de una tabla.

`%` Se utiliza para introducir comentarios: En una línea del fichero fuente dada, todo lo que se encuentre a la derecha de éste signo es ignorado por el compilador y se entiende como comentario.

`~` Se utiliza para evitar la separación de palabras: Es una conocida norma tipográfica el no separar términos complementarios, como por ejemplo Sr. Director ó A. Einstein. Utilizando la tilde como ligadura, se evita que  $\text{\LaTeX}$  rompa éstas palabras en dos al cambiar de línea: Sr.~Director, A.~Einstein <sup>2</sup>

`$ _ ^` Se utilizan en fórmulas matemáticas.

---

<sup>2</sup>Otro método es encerrar la frase dentro de un caja: `\mbox{Texto}`

El carácter reservado significa que no podemos incluirlos en el texto normal simplemente escribiéndolos. En la siguiente tabla se muestra la sintaxis que se debe utilizar para escribirlos dentro de un documento:

<code>~</code>	<code>\~</code>	<code>&amp;</code>	<code>\&amp;</code>
<code>#</code>	<code>\#</code>	<code>_</code>	<code>\_</code>
<code>\$</code>	<code>\\$</code>	<code>\</code>	<code>\textbackslash</code>
<code>%</code>	<code>\%</code>	<code>{</code>	<code>\{</code>
<code>^</code>	<code>\^</code>	<code>}</code>	<code>\}</code>

## 4.2. Comillas, guiones, puntos suspensivos, grados, etc..

### Comillas:

Tecleamos: `<< Texto >>` Resulta: « Texto »  
Tecleamos: `‘ ‘ Texto ’ ’` Resulta: “ Texto ”  
Tecleamos: `‘ Texto ’` Resulta: ‘ Texto ’

### Guiones:

Tecleamos: `-` Resulta: -  
Tecleamos: `--` Resulta: –  
Tecleamos: `---` Resulta: —  
Tecleamos: `$-$` Resulta: −  
(signo matemático menos)

### Puntos suspensivos:

Hay varias formas de introducir los puntos suspensivos:

- `...`  $\longrightarrow$  (la más sencilla)
- `\dots`  $\longrightarrow$  (sólo funciona con la opción `spanish` de `babel`)
- `\dots`  $\longrightarrow$  (comando propio de  $\text{\LaTeX}$ , siempre disponible)
- `\ldots`  $\longrightarrow$  Distancia entre puntos suspensivos algo mayor

### Ordinales y grados:

Para obtener ordinales abreviados ( $1^{\text{a}}$  ó  $1^{\text{o}}$ ), podemos hacerlo directamente desde el teclado. Para escribir otros ordinales, puede hacerse con:

`\textsuperscript{Superíndice}`

Por ejemplo, `3\textsuperscript{er}` produce  $3^{\text{er}}$ . Otra versión (sólo disponible en `babel`, versión `spanish`) es:

`\sptext{Superíndice}`

que introduce un punto antes del superíndice y cambia el tamaño del superíndice si es un carácter en mayúsculas:  $2^{\text{A}}$ ,  $3^{\text{er}}$ . Para generar correctamente el signo de grado, se utiliza: `\textdegree`



### Otros signos:

El siguiente cuadro muestra como obtener otros signos diversos:

Comando	Resultado	Comando	Resultado
<code>\dag</code>	†	<code>\ddag</code>	‡
<code>\S</code>	§	<code>\P</code>	¶
<code>\textbullet</code>	•	<code>\textvisiblespace</code>	~
<code>\textregistered</code>	®	<code>\copyright</code>	©
<code>\texttrademark</code>	™	<code>\pounds</code>	£

Los comandos `\copyright` y `\textregistered` son casos particulares de un comando más general: `\textcircled{Character}`, que encierra `Character` dentro de un círculo.

### 4.3. El euro

Debido a la modernidad del símbolo del euro, la opción `latin1` del paquete `inputenc` desgraciadamente aún no reconoce éste signo. La solución para obtener el símbolo del euro está en cargar en el preámbulo el paquete `eurosym`:

```
\usepackage{eurosym}
```

tras lo cual se obtiene en símbolo € con el comando `\euro`. Puede incluso mejorarse la situación insertando la siguiente declaración en el preámbulo (por supuesto, **después** del comando `\usepackage{eurosym}`):

```
\DeclareInputText{128}{\euro}3
```

que asignaría el símbolo € del teclado la instrucción “`\euro`”, lo que ya permite utilizar el símbolo del teclado con normalidad.

## 5. Párrafos especiales: quote, quotation, verse, y más...

Los entornos `quote` y `quotation` permiten introducir citas textuales, en párrafos ligeramente más pequeños que el texto base; por ejemplo:

La inclusión de citas textuales, como la del escritor Bertolt Brecht que viene a continuación, es una tarea sencilla con `\LaTeX`.

```
\begin{quote}
```

Hay personas que luchan un día, y son buenas. Hay otras que luchan un año y son mejores. Hay quienes luchan muchos años, y son muy buenas.

```
\end{quote}
```

produce:

---

<sup>3</sup>En Windows. Para Linux, el código del carácter € es 164

Con quote:

La inclusión de citas textuales, como la del escritor Bertolt Brecht que viene a continuación, es una tarea sencilla con L<sup>A</sup>T<sub>E</sub>X.

Hay personas que luchan un día, y son buenas. Hay otras que luchan un año y son mejores. Hay quienes luchan muchos años, y son muy buenas.

Pero hay algunas que luchan toda la vida: éstas son las imprescindibles

Con quotation:

La inclusión de citas textuales, como la del escritor Bertolt Brecht que viene a continuación, es una tarea sencilla con L<sup>A</sup>T<sub>E</sub>X.

Hay personas que luchan un día, y son buenas. Hay otras que luchan un año y son mejores. Hay quienes luchan muchos años, y son muy buenas.

Pero hay algunas que luchan toda la vida: éstas son las imprescindibles

Como se puede ver, quotation introduce sangrado en los párrafos de la cita, y disminuye la separación entre los mismos.

Otro entorno predefinido es el `verse`, para escribir versos. Tiene la siguiente sintaxis:

```
\begin{verse}
verso1 \\\
verso2 \\\
....
\end{verse}
```

Utilizando el paquete `shapepar` se pueden contruir párrafos con formas muy especiales. Así por ejemplo, empleando: `\heartpar{Texto del párrafo a formatear}` puede obtenerse lo siguiente:

El 30 de marzo de 1977,  
el profesor Donald E. Knuth, de la Universidad de Stanford, recibió las galeras o pruebas de imprenta de la segunda edición del segundo volumen de su famosa obra *The Art of Computer Programming*. La impresión que dichas pruebas causaron al autor fue nefasta; él mismo las calificó de tipográficamente horribles y tan importantes le parecieron los problemas a los que se enfrentaba que decidió resolverlos por sí mismo. A partir de las ideas de Gutenberg y utilizando las computadoras como herramientas, Knuth creó T<sub>E</sub>X, un sistema para escribir textos científicos (especialmente matemáticos), cómodo y transportable entre plataformas, que muchos consideran ahora como la aportación más importante realizada en este campo de la imprenta de Gutenberg.



Lo anterior es una aplicación del comando general `\parshape`, que permite construir párrafos de forma arbitraria. Su sintaxis es la siguiente:

`\parshape= $n$   $i_1$   $l_1$   $i_2$   $l_2$  ...  $i_n$   $l_n$`

$e$  indica que las primeras  $n$  líneas del párrafo tendrán longitudes  $l_1, \dots, l_n$ , respectivamente, y estarán sangradas  $i_1, \dots, i_n$ , respectivamente. Si el párrafo tiene más de  $n$  líneas, las condiciones para la  $n$ -ésima serán repetidas hasta final de párrafo. Para cancelar el comando, basta incluir `\parshape=0`.

## 6. Tipos de letra

### 6.1. Familias

Por defecto,  $\text{\LaTeX}$  utiliza los tipos Computer Modern Fonts, creados por D.E. Knuth para su utilización en  $\text{\TeX}$ .<sup>4</sup> Estos tipos agrupan tres familias diferentes:

- Roman (la opción por defecto)
- Sanserif (sin adornos)
- Typewriter (tipo máquina de escribir)

con las siguientes instrucciones para obtenerlas, respectivamente:

- |                               |              |                              |
|-------------------------------|--------------|------------------------------|
| ▪ <code>\textrm{Texto}</code> | (roman)      | <code>\rmfamily Texto</code> |
| ▪ <code>\textsf{Texto}</code> | (sanserif)   | <code>\sffamily Texto</code> |
| ▪ <code>\texttt{Texto}</code> | (typewriter) | <code>\ttfamily Texto</code> |

Los comandos a la izquierda corresponden al modo *Texto*, es decir, para textos cortos **no más largos que un párrafo**. Los comandos a la izquierda se mantienen hasta que se declare una nueva familia, aunque lo usual suele ser incluirlos dentro de un *grupo*, en la forma siguiente:

Esto es roman, `\sffamily` esto es sanserif},  
`\ttfamily` esto es typewriter}, y esto sigue siendo roman.

que produce:

Esto es roman, esto es sanserif, esto es typewriter, y esto sigue siendo roman.

Completamente análogo a `\sffamily Texto` sería utilizar el siguiente entorno:

`\begin{sffamily} Texto extenso... \end{sffamily}`

---

<sup>4</sup>Pueden cargarse otros muchos tipos, lo cual se verá mas adelante

## 6.2. Perfiles

Para cada familia tenemos cuatro posibles perfiles, recto (opción por defecto, *itálico*, *inclinado* (slanted) y VERSAL (letras mayúsculas pequeñas), cuyos comandos correspondientes son:

- |                               |                      |                             |
|-------------------------------|----------------------|-----------------------------|
| ▪ <code>\textup{Texto}</code> | (recto)              | <code>\upshape</code> Texto |
| ▪ <code>\textit{Texto}</code> | ( <i>itálico</i> )   | <code>\itshape</code> Texto |
| ▪ <code>\textsl{Texto}</code> | ( <i>inclinado</i> ) | <code>\slshape</code> Texto |
| ▪ <code>\textsc{Texto}</code> | (VERSAL)             | <code>\scshape</code> Texto |

## 6.3. Grosor

Finalmente, hay dos grosores (también llamados series) para cada tipo: el normal ó medio (opción por defecto) y el grueso ó negrita. Se activan con:

- |                               |                   |                              |
|-------------------------------|-------------------|------------------------------|
| ▪ <code>\textmd{Texto}</code> | (medio)           | <code>\mdseries</code> Texto |
| ▪ <code>\textbf{Texto}</code> | ( <b>grueso</b> ) | <code>\bfseries</code> Texto |

Todas las características explicadas (familia, perfil y grosor) pueden combinarse (aunque puntualmente alguna opción mixta no esté disponible). Por ejemplo, `\bfseries\itshape` produce ***letra*** *negrita itálica*.

## 6.4. Enfatizar y subrayar

Se puede resaltar texto con el comando `\emph{Texto}` (ó `\em` Texto, en modo extendido) lo cual pone en *itálica* el texto si el ambiente es normal, ó pone normal el texto si el ambiente es *itálico*.

Para subrayar texto, se utiliza `\underline{Texto}`.

## 6.5. Tamaño

Tenemos a nuestra disposición los siguientes tamaños, los cuales son *relativos* a la fuente estándar del documento (que puede cambiarse, ya se verá mas adelante cómo)

- |                                    |       |
|------------------------------------|-------|
| ▪ <code>\tiny</code> Texto         | Texto |
| ▪ <code>\scriptsize</code> Texto   | Texto |
| ▪ <code>\footnotesize</code> Texto | Texto |
| ▪ <code>\small</code> Texto        | Texto |
| ▪ <code>\normalsize</code> Texto   | Texto |
| ▪ <code>\large</code> Texto        | Texto |
| ▪ <code>\Large</code> Texto        | Texto |

- `\LARGE Texto`
- `\huge Texto`
- `\Huge Texto`

Texto  
 Texto  
 Texto

## 6.6. Colores

El paquete `color` permite colorear un texto. Se puede cargar como `\usepackage{color}`, lo cual permite sólo usar unos pocos colores básicos, o mejor con: `\usepackage[usenames]{color}`, que permite usar hasta 68 colores diferentes (ver figura).

Para cambiar el color, se usan los comandos:

- `\textcolor{NombreColor}{Texto}`
- `\color{NombreColor}`

siendo la primera la versión corta, y la segunda la versión larga que tendrá efecto hasta que se cambie de nuevo el color (ó hasta que termine el grupo, si se usa: `{\color{NombreColor} Texto extenso... }`

Todas las características anteriores pueden combinarse:

**Esto es letra grande, negrita y azul**

*y esto letra pequeña, sanserif, itálica y roja*

	GreenYellow		Yellow		Goldenrod		Dandelion		Apricot
	Peach		Melon		YellowOrange		Orange		BurntOrange
	Bittersweet		RedOrange		Mahogany		Maroon		BrickRed
	Red		OrangeRed		RubineRed		WildStrawberry		Salmon
	CarnationPink		Magenta		VioletRed		Rhodamine		Mulberry
	RedViolet		Fuchsia		Lavender		Thistle		Orchid
	DarkOrchid		Purple		Plum		Violet		RoyalPurple
	BlueViolet		Periwinkle		CadetBlue		CornflowerBlue		MidnightBlue
	NavyBlue		RoyalBlue		Blue		Cerulean		Cyan
	ProcessBlue		SkyBlue		Turquoise		TealBlue		Aquamarine
	BlueGreen		Emerald		JungleGreen		SeaGreen		Green
	ForestGreen		PineGreen		LimeGreen		YellowGreen		SpringGreen
	OliveGreen		RawSienna		Sepia		Brown		Tan
	Gray		Black		White				

Figura 1: Nombre de los 68 colores predefinidos