

## Sesión III

# $\text{\LaTeX}$ como editor de texto avanzado

## ① Insertando objetos

Figuras

Tablas

Códigos

## ② Formato avanzado

Secciones e índices

El formato de página

# Sección 1

## Insertando objetos

## Flotantes y su localización

Los objetos con los que vamos a trabajar: figuras, tablas, etc... se conocen como flotantes.

Por defecto  $\text{\LaTeX}$  los coloca donde menos moleste: en el lugar del texto donde hemos colocado el código, al principio de la página o al final de la página.

# Flotantes y su localización

Los objetos con los que vamos a trabajar: figuras, tablas, etc... se conocen como flotantes.

Por defecto  $\text{\LaTeX}$  los coloca donde menos moleste: en el lugar del texto donde hemos colocado el código, al principio de la página o al final de la página.

Podemos especificar dónde colocarlos mediante parámetros optativos:

`\begin{figure} [placement specifier].`

Las opciones son las siguientes

Specifier	Permission
h	Place the float here (approximately at the same point it occurs in the source text)
t	Position at the top of the page.
b	Position at the bottom of the page.
p	Put on a special page for floats only.
!	Override internal parameters $\text{\LaTeX}$ uses for determining “good” float positions.
H	Places the float at precisely the location in the $\text{\LaTeX}$ code. Requires the <code>float</code> package.

También se admiten cadena `htb` significa: intétalo en su sitio, si no ponlo arriba y, si no, abajo.

# Figuras

Para incluir imágenes empleamos el paquete `graphicx`. Si empleamos el compilador `pdflatex`. Podemos emplear imágenes `.pdf`, `.png`.

Tiene los siguientes parámetros:

- ① `<path>` Es la dirección del archivo a incluir. Lo mejor es escribirla relativa al directorio esto es `<path>=archivo.pdf`

## Código

```
\begin{figure}[h!]
  \includegraphics[scale= <scale>] {<path>}
  \caption {<caption>}
\end{figure}
```

# Figuras

Para incluir imágenes empleamos el paquete `graphicx`. Si empleamos el compilador `pdflatex`. Podemos emplear imágenes `.pdf`, `.png`.

Tiene los siguientes parámetros:

- ① `<path>`
- ② `<scale>` Valor entre 0 y 1 al que escalar la figura.

## Código

```
\begin{figure}[h!]
  \includegraphics[scale= <scale>] {<path>}
  \caption {<caption>}
\end{figure}
```

# Figuras

Para incluir imágenes empleamos el paquete `graphicx`. Si empleamos el compilador `pdflatex`. Podemos emplear imágenes `.pdf`, `.png`.

Tiene los siguientes parámetros:

- ① `<path>`
- ② `<scale>`
- ③ `<caption>` El contenido de `<caption>` será el pie de foto, es decir, aparecerá Figura x:  
caption

## Código

```
\begin{figure}[h!]
  \includegraphics[scale= <scale>] {<path>}
  \caption{<caption>}
\end{figure}
```

# Figura

## Ejemplo

### Código [figura.tex]

```
\documentclass{article}

\usepackage{graphicx}

\begin{document}

\begin{figure}

\centering
\includegraphics[scale=0.5]
{gauss.pdf}
\caption{Gauss}

\end{figure}

\end{document}
```



Figure 1: Gauss

Si añadimos el paquete babel,  
el texto “Figure” cambia a  
“Figura”

# Tablas

Las el entorno `table` es equivalente a `figura`, pero al generar el `caption` obtendremos Cuadro (como recomienda la RAE).

En contenido de la tabla se introduce de manera similar a una matriz

## 1 <align>:

- a) l: izquierda
- b) c: centrado
- c) r: derecha

## Código

```
\begin{tabular}{<align>| ... }
```

cuadro1 & cuadro2 & ... \\

```
\hline\\
```

...

```
\end{tabular}
```

# Tablas

En el entorno `table` es equivalente a `figura`, pero al generar el `caption` obtendremos Cuadro (como recomienda la RAE).

En contenido de la tabla se introduce de manera similar a una matriz

- 1 <align>:
- 2 & Separación entre cuadros en la misma fila

## Código

```
\begin{tabular}{<align>| ... }
```

cuadro1 & cuadro2 & ... \\

```
\hline\\
```

...

```
\end{tabular}
```

# Tablas

Las el entorno `table` es equivalente a `figura`, pero al generar el `caption` obtendremos Cuadro (como recomienda la RAE).

En contenido de la tabla se introduce de manera similar a una matriz

- ① `<align>`:
- ② `&`
- ③ | Si se desea línea vertical entre dos columnas

## Código

```
\begin{tabular}{<align>| ... }  
    cuadro1 & cuadro2 & ... \\  
    \hline\\  
    ...  
\end{tabular}
```

# Tablas

Las el entorno `table` es equivalente a `figura`, pero al generar el `caption` obtendremos Cuadro (como recomienda la RAE).

En contenido de la tabla se introduce de manera similar a una matriz

- ① `<align>`:
- ② `&`
- ③ `|`
- ④ `\hline` Si se desea una línea horizontal.

## Código

```
\begin{tabular}{<align>| ... }  
cuadro1 & cuadro2 & ... \\  
\hline\\  
...  
\end{tabular}
```

# Tablas

## Ejemplo

### Código

```
\begin{table}

\begin{tabular}{c|l r|}
1 & 2 & 3 \\
\hline
4 & 5 & \\
6 & & 7
\end{tabular}

\caption{Mi tabla}
\end{table}
```

1	2	3	
4	5		
6		7	

Cuadro: Mi tabla

# Uso avanzado de tablas

<https://www.overleaf.com/learn/latex/Tables>

# Ejercicio de tablas

# Tablas para vagos

Hay múltiples páginas que permiten convertir tablas de excel en latex

- <https://www.tablesgenerator.com/>
- <https://www.latex-tables.com/>
- <https://tableconvert.com/latex-generator>

# Código

Hay diferentes paquetes para incluir código fuente:

- `verbatim`
- `listings`

MathWorks ofrece un paquete (a descargar aquí) para incluir código con formato de matlab

- `minted`. En mi opinión, el más completo. Será el que usemos. Un tutorial más completo en la web de Overleaf.

# Añadir código

El paquete listing nos permite introducir código de diferentes lenguajes

## Código

```
\usepackage{minted}

\begin{document}

\begin{minted}{c++}
#include<stdio.h>
main()
{
    printf("Hello World");
}
\end{minted}

\end{document}
```

```
#include <stdio.h>
main()
{
    printf("Hello World");
}
```

Figura: Resultado de compilar

# Añadir código

También se puede cargar de un fichero externo

## Código [ejemplo-c-2.tex]

```
\usepackage{minted}

\begin{document}
\inputminted[c++]{codigo2.cpp}
\end{document}
```

## Código [codigo2.cpp]

```
#include <stdio.h>
main()
{
    printf("Hello World");
}
```

```
#include <stdio.h>
main()
{
    printf("Hello World");
}
```

Figura: Resultado de compilar ejemplo-c-2.tex

## Sección 2

### Formato avanzado

## Secciones e índices

Naturalmente L<sup>A</sup>T<sub>E</sub>X permite estructurar el documento en secciones.

# Secciones e índices

Naturalmente L<sup>A</sup>T<sub>E</sub>X permite estructurar el documento en secciones.

Para empezar una nueva sección con título basta con indicarlo y dar un título utilizando los siguientes comandos:

-1	<code>\part{titulo}</code>
0	<code>\chapter{titulo}</code>
1	<code>\section{titulo}</code>
2	<code>\subsection{titulo}</code>
3	<code>\subsubsection{titulo}</code>
4	<code>\paragraph{titulo}</code>
5	<code>\ subparagraph{titulo}</code>

# Secciones e índices

Naturalmente L<sup>A</sup>T<sub>E</sub>X permite estructurar el documento en secciones.

Para empezar una nueva sección con título basta con indicarlo y dar un título utilizando los siguientes comandos:

-1	<code>\part{titulo}</code>
0	<code>\chapter{titulo}</code>
1	<code>\section{titulo}</code>
2	<code>\subsection{titulo}</code>
3	<code>\subsubsection{titulo}</code>
4	<code>\paragraph{titulo}</code>
5	<code>\ subparagraph{titulo}</code>

Se puede generar el índice introduciendo `\tableofcontents`. Este comando admite parámetros optativos.

# Secciones e índices

Naturalmente L<sup>A</sup>T<sub>E</sub>X permite estructurar el documento en secciones.

Para empezar una nueva sección con título basta con indicarlo y dar un título utilizando los siguientes comandos:

-1	<code>\part{titulo}</code>
0	<code>\chapter{titulo}</code>
1	<code>\section{titulo}</code>
2	<code>\subsection{titulo}</code>
3	<code>\subsubsection{titulo}</code>
4	<code>\paragraph{titulo}</code>
5	<code>\ subparagraph{titulo}</code>

Se puede generar el índice introduciendo `\tableofcontents`. Este comando admite parámetros optativos.

También se pueden hacer otras tablas de contenidos: `\listoffigures`, `\listoftables`.

# El paquete fancyhdr

Una forma sencilla de darle un buen estilo a nuestro archivo es emplear el paquete `fancyhdr`. Tiene varias opciones, lo más sencillo es emplear un paquete predefinido

## Código

```
\usepackage{fancyhdr}  
\pagestyle{<style>}
```

<style>:

|                         |  |                           |
|-------------------------|--|---------------------------|
|                         | Encabezado                             | Pie de página             |
| <code>empty</code>      | Vacío                                  | Vacío                     |
| <code>plain</code>      | Vacío                                  | Número de página centrado |
| <code>headings</code>   | Nombre del capítulo y número de página | Vacío                     |
| <code>myheadings</code> | Número de página y otros               | Vacío                     |
| <code>fancy</code>      | Capítulo y sección                     | Número de página          |

# Márgenes: el paquete geometry

[https://www.overleaf.com/learn/latex/Page\\_size\\_and\\_margins](https://www.overleaf.com/learn/latex/Page_size_and_margins)