

Sesión I


Bienvenido a \LaTeX

- ① Primeros pasos
- ② ¡Hola, Mundo! e ingredientes básicos
- ③ Estructura del documento y principales elementos
- ④ \LaTeX en local

Sección 1

Primeros pasos

Overleaf

Una opción sencilla es usar Overleaf 

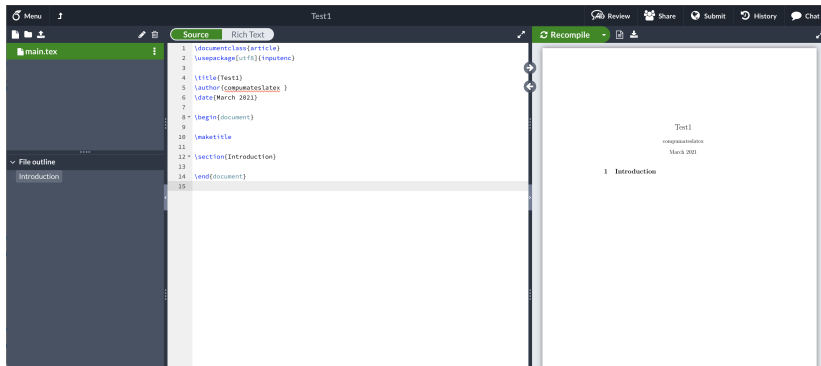


Figura: Interfaz de Overleaf

El archivo mínimo

Los archivos de \LaTeX son archivos de texto (plano) con extension `.tex`.
La estructura es

Código

```
\documentclass{<style>}  
  
% Configuración del archivo  
  
\begin{document}  
  
% El texto  
  
\end{document}
```

Estos documentos se “compilan” para conseguir un documento “con formato”.

El archivo ¡Hola, Mundo!

Creemos un archivo básico

Código [hola-mundo.tex]

```
\documentclass{article}
```

```
\begin{document}
```

```
Hola, Mundo.
```

```
\end{document}
```

Comentarios

Lo que venga después de `%` no se procesará como comandos o contenido, si no como comentarios.

Código [hola-mundo-con-comentario.tex]

```
\documentclass{article}
```

```
\begin{document}
```

```
Hola, Mundo % Cruel.
```

```
\end{document}
```

Errores y avisos

¿Qué pasa si comentemos un error?

Código

```
\documentclass{article}
```

```
\begin{document}
```

Hola, Mundo.

```
\end{document}
```

Al compilar

! LaTeX Error: Missing `\begin{document}`.

Hay tres tipos de mensajes:

- **Error** normalmente no permitirá la compilación total o parcial.
- **Warning** que solo avisan de un posible mal funcionamiento.
- **Info** son cuestiones menores, que conviene revisar.

Sección 3

Estructura del documento y principales elementos

Cabecera y tipo de documento

En la cabecera introduciremos todo lo relativo a configuración

Código

```
\documentclass{<style>}  
  
% Configuración del archivo  
  
\begin{document}  
  
% El texto  
  
\end{document}
```

Tipo de documento

Aquí es donde decidimos que tipo de archivo latex queremos escribir, hay diferentes tipos de documentos

`<style>`:

- ❶ `article` Para artículos cortos. Acepta partes, secciones y subsecciones

Código

```
\documentclass{<style>}
```

```
% Configuración del archivo
```

```
\begin{document}
```

```
% El texto
```

```
\end{document}
```

Tipo de documento

Aquí es donde decidimos que tipo de archivo latex queremos escribir, hay diferentes tipos de documentos

`<style>`:

- ② book Para archivos más extensos. Acepta partes, capítulos, secciones, subsecciones

Código

```
\documentclass{<style>}

% Configuración del archivo

\begin{document}

% El texto

\end{document}
```

El cuerpo

A partir de aquí escribiremos el texto

Todo lo que queramos escribir.

Código

```
\documentclass{<style>

% Configuración del archivo

\begin{document}

% El texto

\end{document}
```

Comandos y variables

Una herramienta fundamental en la escritura con \LaTeX

- `<command>` Nombre del comando
- `<opt>` Argumento optativo.
- `<arg#>` Argumento obligatorio

Código (llamada a comando)

```
\<command>[<opt>]{<arg1>}{<arg2>}
```

Por ejemplo,

Código

```
Hola, \textbf{Mundo}.
```

Entornos

Los entornos funcionan como comandos, pero nos permiten introducir cantidades más largas de texto.

Algunos ejemplos son

- `document`: Es donde introducimos el documento
- `equation`: Para introducir ecuaciones numeradas
- `emph`: Para conseguir textos en cursiva.

Código

```
\begin{<env>}[<opt>]
```

```
\end{<env>}
```

Los paquetes

Por defecto \LaTeX no incluye demasiados comandos ni entornos. Podemos añadir nuevas funcionalidades (comandos y entornos) incluyendo **paquetes**.

Uno de los paquetes más usuales es el paquete matemático de la American Mathematical Society (AMS): `amsmath`.

Código

```
\documentclass{article}
```

```
\usepackage{amsmath}
```

```
\begin{document}
```

```
\begin{equation}
```

```
\sum_{i=1}^3 a_i = 1.
```

```
\end{equation}
```

```
\end{document}
```

$$\sum_{i=1}^3 a_i = 1. \quad (1)$$

Creando comandos

Código

```
\documentclass{standalone}

\newcommand{\deciralgo}[1]
  {Esto es lo que digo: ``#1''.
  Y no me arrepiento.}

\begin{document}
  \deciralgo{Hola}
\end{document}
```

Esto es lo que digo: "Hola". Y no me arrepiento.

El fichero y compatibilidades

El paquete `inputenc`

Para mayor compatibilidad, especialmente entre sistemas operativos es recomendable guardar los archivos de `.tex` en formato UTF8. Esto nos permitirá poner acentos de manera sencilla.

Para indicarle al compilar que hemos hecho eso escribimos.

Código

```
\documentclass{<style>}  
  
\usepackage[utf8]{inputenc}  
  
\begin{document}
```

El paquete babel

Para que \LaTeX ponga todos los textos automáticos en castellano deberemos añadir el paquete `babel`

Para indicarle al compilar que hemos hecho eso escribimos.

Código

```
\documentclass{<style>}  
  
\usepackage[spanish]{babel}  
  
\begin{document}
```

Aspecto de un primer documento

Código [basico.tex]

```
\documentclass{article}

\usepackage[spanish]{babel}

\title{Mi trabajo a \LaTeX}
\author{Yo \&\& Y mi amigo}
\date{\today}

\begin{document}

\maketitle

Este es el principio de mi trabajo.

\end{document}
```

Mi trabajo a L^AT_EX

Yo
Y mi amigo
29 de junio de 2024

Este es el principio de mi trabajo.

1

Ficheros modulares: `input`

Escribir un libro completo en un único archivo no es cómodo. Por eso \LaTeX permite escribir modularmente.

Podemos escribir en diferentes archivos `.tex`, y luego juntarlos en un principal.

Código [`modular.tex`]

```
\documentclass{standalone}
\begin{document}
    \input{modulo1.tex}
    \input{modulo2.tex}
\end{document}
```

Código [`modulo1.tex`]

Un texto.

Código [`modulo2.tex`]

Otro texto.

Un texto. Otro texto.

Figura: Resultado de compilar `modular.tex`

Sección 4

L^AT_EX en local

Descargar e instalar L^AT_EX

Esta experiencia depende del sistema utilices: visita [Latex project](#)

- Windows: MikTeX
- Mac: MacTeX (o via homebrew)
- Linux: a través del gestor software nativo de terminal
 - Debian/Ubuntu: `sudo apt-get install texlive-full`
 - RedHat/Fedora: `yum install texlive-scheme-full`
 - Suse: `zypper install texlive-latex`
 - Arch: `pacman -S texlive-most`
 - Otros: ¿en serio? ¿ninguno de los anteriores?. Te buscas la vida.

Compilando en terminal

Navegar hasta la carpeta y escribir en terminal

```
$ pdflatex hola-mundo.tex
```

En archivos más complicados hay que ejecutar el código varias veces

Diferentes compiladores

Hay diferentes opciones

- ➊ `pdflatex`. El más habitual. Genera un archivo `.pdf`
- ➋ `latex`. El más tradicional. Genera un archivo `.dvi`
- ➌ `XeLaTeX` y `LuaLaTeX` soportan tipografías Truetype and OpenType. Se usan a veces para documentos muy visuales.

Editores locales

Un archivo `.tex` es archivo de texto “plano”. Se puede editar con cualquier editor.

Algunos editores tienen funcionalidades avanzadas (compilación integrada, `synctex`, visor de pdf...), que simplifican el trabajo

- ➊ Visual Studio Code. Con la extensión Latex-workshop.
- ➋ Latex Workshop
- ➌ Texstudio

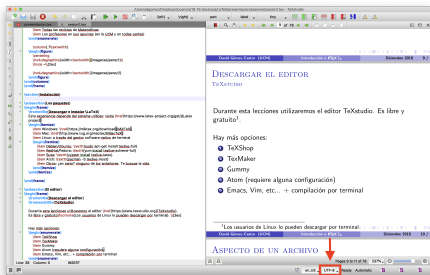
El fichero y compatibilidades

Los ficheros de texto plano en principio soportan muy pocos caracteres (ASCII).

Para aumentar el número de caracteres soportados (por ejemplos añadir letras con acentos) hay diferentes codificaciones:

UTF-8 (por defecto en Linux y Mac), ISO (por defecto en Windows), ...

Para aprovechar esto usamos el paquete `inputenc` con la codificación correcta.



Curiosidades sobre texto plano: *Plain Text* by Dylan Beattie en Youtube