

🔒 [jlmg4589](#) / [Markdown](#) Private

[Code](#) [Issues](#) [Pull requests](#) [Actions](#) [Projects](#) [Security](#) [Insights](#)

[Markdown](#) / [README.md](#) 

...

 [jlmg4589](#) [Update README.md](#) ...

1 minute ago ... 

Update README.md

4e62270

Tutorial de Markdown

407 lines (310 loc) · 10.4 KB

**Preview**

[Code](#)

[Blame](#)

[Raw](#)











# Índice

1. [Taller de introducción a Markdown](#)
  - 1.1 [¿Qué es Markdown?](#)
  - 1.2 [Etiquetas básicas de Markdown](#)
    - 1.2.1 [Encabezados](#)
    - 1.2.2 [Negrita y cursiva](#)
    - 1.2.3 [Resaltar un comando](#)
    - 1.2.4 [Bloques de código](#)
    - 1.2.5 [Enlaces](#)
    - 1.2.6 [Imágenes](#)
    - 1.2.7 [Listas](#)
    - 1.2.8 [Tablas](#)
    - 1.2.9 [Forzar un salto de línea](#)
    - 1.2.10 [Citar textos](#)
    - 1.2.11 [Comentarios](#)
    - 1.2.12 [Diagrama de flujo](#)
    - 1.2.13 [Ecuaciones matemáticas](#)

2. [Referencias](#)

## 1. Taller de introducción a Markdown

## 1.1 ¿Qué es Markdown?

---

**Markdown** es un lenguaje de marcado ligero creado por [John Gruber](#) y [Aaron Swartz](#) que trata de **facilitar la redacción y lectura de documentos en texto plano**, utilizando un conjunto de etiquetas muy sencillas para aplicar estilo al documento.

Actualmente no existe un estándar para **Markdown**, por este motivo existen diferentes versiones o flavors de **Markdown**.

En este curso vamos a trabajar con el flavor que se utiliza en GitHub para redactar los archivos README.md de los repositorios y la documentación técnica de los proyectos. Además *GitHub*, junto a *Discourse* o *reddit* están llevando una iniciativa para crear un estándar moderno de **Markdown** como es **CommonMark**.

En plataformas de repositorios remotos como *GitHub*, la función de este archivo es presentar información del proyecto, como:

- Descripción de su proyecto.
- funcionalidades.
- Cómo pueden usarlo los usuarios.
- Donde los usuarios pueden encontrar ayuda sobre su proyecto.
- Autores del proyecto.

## 1.2 Etiquetas básicas de Markdown

---

### 1.2.1 Encabezados

**Sintaxis Markdown:**

```
# Esto es un encabezado h1
## Esto es un encabezado h2
### Esto es un encabezado h3
#### Esto es un encabezado h4
##### Esto es un encabezado h5
##### Esto es un encabezado h6
```



**HTML renderizado en GitHub:**

# Esto es un encabezado h1

---

## Esto es un encabezado h2

### Esto es un encabezado h3

#### Esto es un encabezado h4

##### Esto es un encabezado h5

##### Esto es un encabezado h6

### 1.2.2 Negrita y cursiva

| Estilo  | Métodos abreviados                             | Sintaxis                              | Ejemplo                                    | Salida                       |
|---------|--|---------------------------------------|--|------------------------------|
| Negrita | Command+B (Mac) o<br>Ctrl+B<br>(Windows/Linux) | <b>** **</b> o <b>__</b><br><b>__</b> | <b>**</b> Texto en<br>negrita <b>**</b>    | <b>Texto en<br/>negrita</b>  |
| Cursiva | Command+I (Mac) o<br>Ctrl+I<br>(Windows/Linux) | <i>* *</i> o <i>_ _</i>               | <i>*</i> Texto en<br>cursiva <i>*</i>      | <i>Texto en<br/>cursiva</i>  |
| Tachado | Ninguno  | <del>~ ~ ~</del>                      | <del>~</del> Texto<br>tachado <del>~</del> | <del>Texto<br/>tachado</del> |

### 1.2.3 Resaltar un comando

#### Sintaxis Markdown:

En esta frase estamos resaltando el comando ``ls -la``.

#### HTML renderizado en GitHub:

En esta frase estamos resaltado el comando `ls -la`.

### 1.2.4 Bloques de código

Al inicio del bloque se puede indicar de forma opcional cuál es el tipo de contenido que contiene el bloque para resaltar las palabras reservadas cuando se renderice.

Por ejemplo: `bash`, `python`, `yaml`, `json`, `html`, `javascript`, etc.

Para omitir las comillas triple de bloque (``), podemos albergar todo el bloque con cuatro comillas (````), o escapar a cada comilla con (\).

#### Sintaxis Markdown:

```
```
sudo systemctl start apache2
```
```



```
```bash
#!/bin/bash
echo "Hola mundo"
```
```



```
```python
celsius = float(input('Introduce una temperatura en grados
Celsius: '))
fahrenheit = (1.8 * celsius) + 32
print(f'La temperatura en grados Fahrenheit es: {fahrenheit}')
```



## HTML renderizado en GitHub:

```
sudo systemctl start apache2
```



```
#!/bin/bash
echo "Hola mundo"
```



```
celsius = float(input('Introduce una temperatura en grados Celsius
fahrenheit = (1.8 * celsius) + 32
print(f'La temperatura en grados Fahrenheit es: {fahrenheit}')
```



## 1.2.5 Enlaces

### Sintaxis Markdown:

```
[Enlace a la página web del IES Doñana](https://iesdonana.org)
```



## HTML renderizado en GitHub:

[Enlace a la página web del IES Doñana](https://iesdonana.org)

### 1.2.5.1 Otra forma de gestionar enlaces

También se pueden crear enlaces con esta sintaxis:

## Sintaxis Markdown:

Enlaces a la página web del [IES Donñana][1] y a [GitHub][2].



[1]: <https://iesdonana.org>

[2]: <https://github.com>

## HTML renderizado en GitHub:

Enlaces a la página web del [IES Donñana](https://iesdonana.org) y a [GitHub](https://github.com).

Esta forma de gestionar los enlaces puede ser útil cuando vamos a utilizar el mismo enlace varias veces en el documento.

## 1.2.6 Imágenes

### Sintaxis Markdown:

Markdown:



```
![imagen1](https://pro.iesdonana.org/assets/logo.png "Leyenda de la
```

HTML:



```
<p align="center">
  
```

El siguiente código es erróneo, pero se ha introducido para comprobar que aparece la identificación indicada entre corchetes [imagen3].

Markdown:



```
![imagen3](/images/instituto.jpeg)
```

El error, está en que el nombre del archivo que hay dentro de la carpeta "images" es "instituto.jpg".

### HTML renderizado en GitHub:



[imagen3](#)

## 1.2.7 Listas

### 1.2.7.1 Listas desordenadas

## Sintaxis Markdown:

- \* Item 1
- \* Item 2
- \* Item 3
- \* Item 4



## HTML renderizado en GitHub:

- Item 1
- Item 2
- Item 3
- Item 4

### 1.2.7.2 Listas desordenadas anidadas

Puedes crear una lista anidada al dejar sangría en uno o más elementos de la lista debajo de otro elemento.

## Sintaxis Markdown:

- \* Item 1
  - \* Item 1.1
  - \* Item 1.2
- \* Item 2
  - \* Item 2.1
- \* Item 3
- \* Item 4



## HTML renderizado en GitHub:

- Item 1
  - Item 1.1
  - Item 1.2
- Item 2
  - Item 2.1
- Item 3
- Item 4

### 1.2.7.3 Listas ordenadas

## Sintaxis Markdown:

1. Item 1
2. Item 2
3. Item 3
4. Item 4



### HTML renderizado en GitHub:

1. Item 1
2. Item 2
3. Item 3
4. Item 4

#### 1.2.7.4 Listas ornenadas anidadas

Puedes crear una lista anidada al dejar sangría en uno o más elementos de la lista debajo de otro elemento.

### Sintaxis Markdown:

1. Item 1
  - 1.1 Item 1.1
  - 1.2 Item 1.2
2. Item 2
  - 2.1 Item 2.1
3. Item 3
4. Item 4



### HTML renderizado en GitHub:

1. Item 1
  - 1.1 Item 1.1
  - 1.2 Item 1.2
2. Item 2
  - 2.1 Item 2.1
3. Item 3
4. Item 4

#### 1.2.7.5 Listas de comprobación

### Sintaxis Markdown:



- [x] Caso 1
- [ ] Caso 2
- [ ] Caso 3
- [x] Caso 4



HTML renderizado en GitHub:

- ☒ Caso 1
- ☐ Caso 2
- ☐ Caso 3
- ☒ Caso 4

1.2.8 Tablas

Sintaxis Markdown:

```
| Encabezado 1 | Encabezado 2 | Encabezado 3 |
| --- | ---: | :---: |
| Fila 1.1 | Fila 1.2 | Fila 1.3 |
| Fila 2.1 | Fila 2.2 | Fila 2.3 |
| Fila 3.1 | Fila 3.2 | Fila 3.3 |
```



HTML renderizado en GitHub:

| Encabezado 1    | <i>Encabezado 2</i> | Encabezado 3    |
|-----------------|---------------------|-----------------|
| <i>Fila 1.1</i> | Fila 1.2            | Fila 1.3        |
| Fila 2.1        | <i>Fila 2.2</i>     | Fila 2.3        |
| Fila 3.1        | Fila 3.2            | <i>Fila 3.3</i> |

1.2.9 Forzar un salto de línea

Para forzar un salto de línea es necesario incluir dos espacios en blanco y un salto de línea.

Sintaxis Markdown:

Por ejemplo, en esta frase  
hemos forzado un salto de línea.



HTML renderizado en GitHub:

Por ejemplo, en esta frase  
hemos forzado un salto de línea.

## 1.2.10 Citar textos

### Sintaxis Markdown:

```
Este texto no es una cita.  
> Este texto daría como resultado una cita.
```



### HTML renderizado en GitHub:

Este texto no es una cita.

Este texto daría como resultado una cita.

## 1.2.11 Comentarios

Para poner un comentario en Markdown y que su contenido no sea renderizado, se utiliza la misma sintaxis que los comentarios de HTML. **Sintaxis Markdown:**

Párrafo 1.



```
<!-- Este texto es un comentario y no será renderizado -->
```

Párrafo 2.

### HTML renderizado en GitHub:

Párrafo 1.

Párrafo 2.

## 1.2.12 Diagrama de flujo

### Sintaxis Markdown:

Simple diagrama de flujo:



```
```mermaid
graph TD;
  A-->B;
  A-->C;
  B-->D;
  C-->D;
```
```

## HTML renderizado en GitHub:

Simple diagrama de flujo:



[Editor de mermaid](#)

## 1.2.13 Ecuaciones matemáticas

### Sintaxis Markdown:

Para habilitar una comunicación clara de las expresiones matemáticas, GitHub admite expresiones matemáticas con formato LaTeX en Markdown.

Como ya se ha comentado, al igual que ocurre con Latex, para poner una ecuación se ha de poner entre \$ la expresión concreta.

- Para insertar una ecuación dentro de una frase, o para utilizar un carácter especial, como letras griegas.

La ecuación,  $x = \frac{-b \pm \sqrt{b^2 - 4 \cdot a \cdot c}}{2 \cdot a}$ , es de segundo grado.



- Como bloque.

La ecuación de segundo grado es:



$$x = \frac{-b \pm \sqrt{b^2 - 4 \cdot a \cdot c}}{2 \cdot a}$$

### HTML renderizado en GitHub:

- Dentro de una frase:

La ecuación,  $x = \frac{-b \pm \sqrt{b^2 - 4 \cdot a \cdot c}}{2 \cdot a}$ , es de segundo grado.

- En un bloque.

La ecuación de segundo grado es:

$$x = \frac{-b \pm \sqrt{b^2 - 4 \cdot a \cdot c}}{2 \cdot a}$$

## 2. Referencias

- [GitHub](#)
- [IAW - Jose Juan Sánchez](#)
- [StackOverFlow](#)
- [Editor de ecuaciones para Latex](#)
- [RADME en tu GitHub](#)