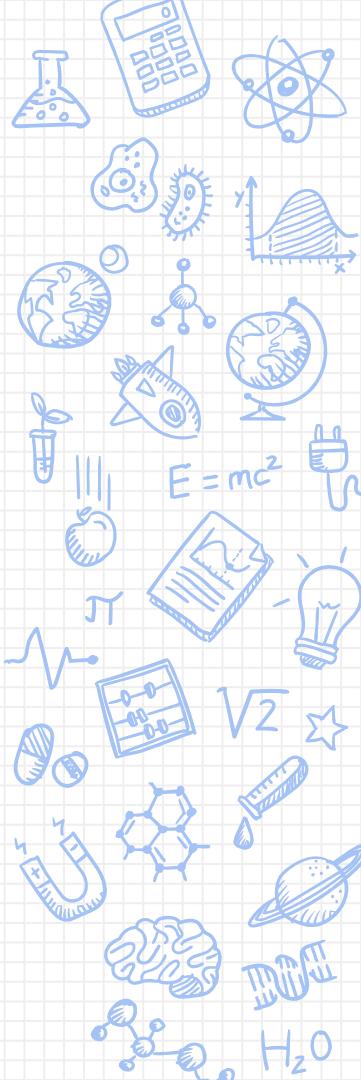


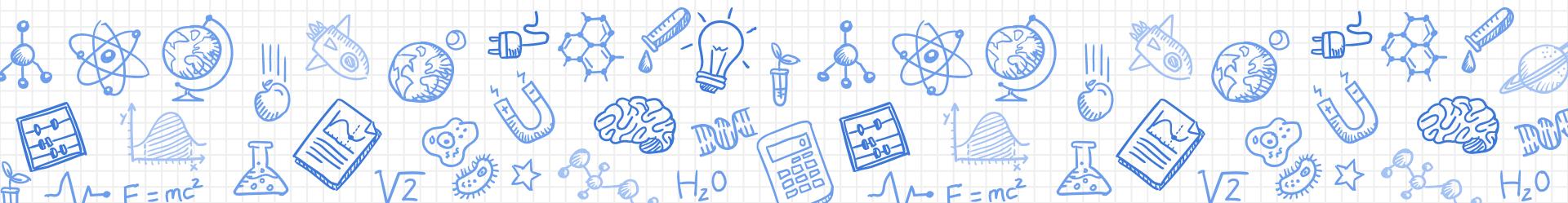
# Clase 03

# En la clase de hoy:

- ## - HTML - Tercera parte



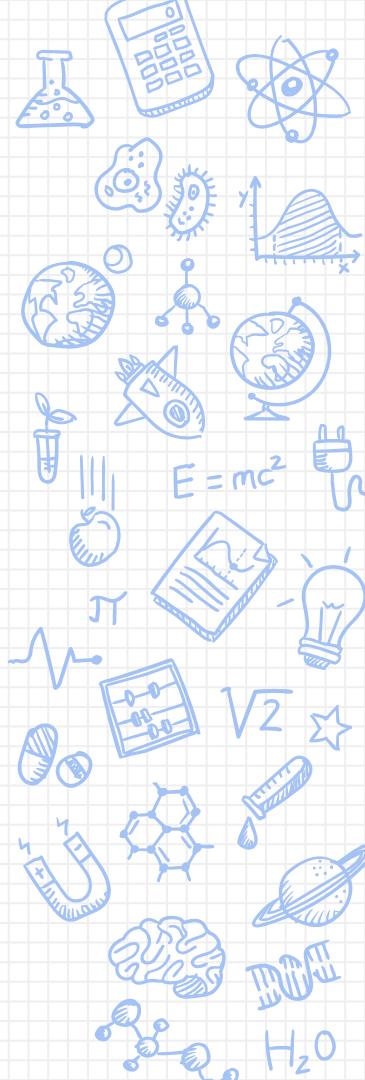
# Audio / Video



# Audio / Video

---

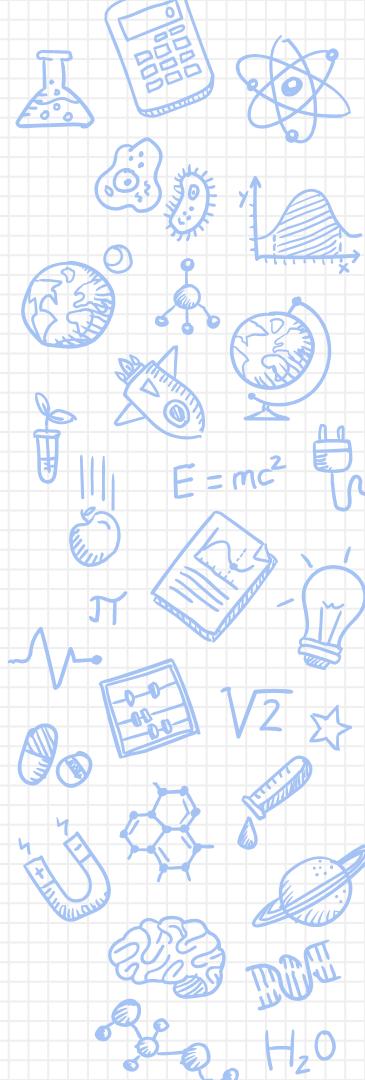
- La forma nativa de HTML5 de insertar Audio o Video es igual en ambos casos.
- Lo único que va a cambiar es el nombre de la etiqueta, pero la estructura es la misma.
- Para insertar audio tenemos la etiqueta  
**<audio></audio>**
- Y para insertar un video tenemos la etiqueta  
**<video></video>**



# Audio / Video

---

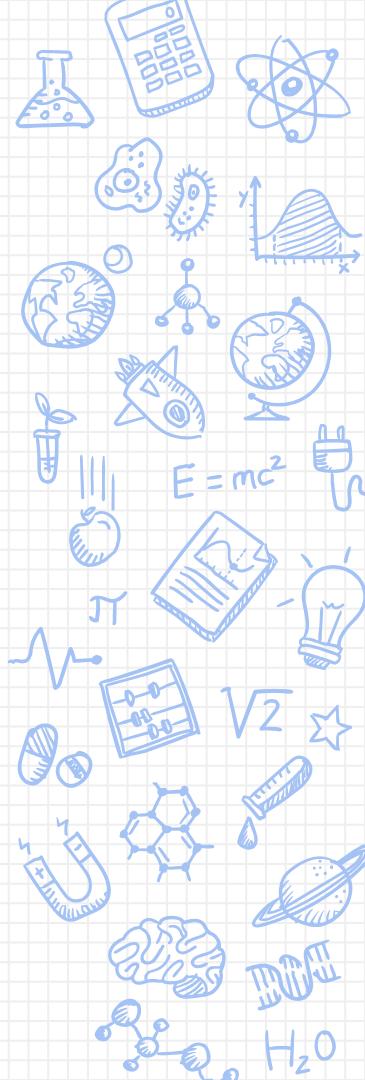
- Es importante saber que estas etiquetas solo funcionan para subir archivos locales. Es decir que tenemos que tener el recurso original.
- Si queremos insertar un video de YouTube, lo vamos a hacer de otra manera.



# Audio / Video

---

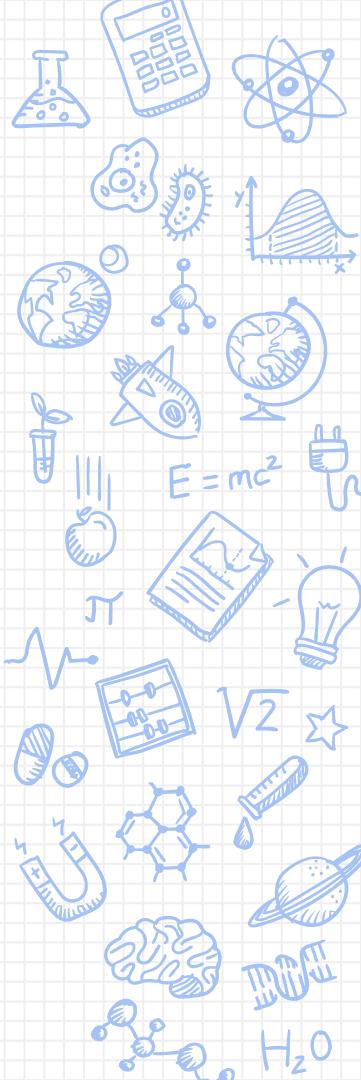
- En el caso de que tengamos un único archivo, es decir, que no contemos con el recurso en múltiples formatos, solamente debemos incluir el atributo **src**:
  - `<audio src="...></audio>`
  - `<video src="...></video>`



# Audio / Video

---

- Si contamos con más de un formato, que sería lo ideal, vamos a dejar de lado el atributo `src` para incluir una etiqueta.
- La etiqueta a incluir es **<source>**. Una etiqueta de autocierre que va a recibir dos atributos:
  - Atributo **src** → la ruta del recurso.
  - Atributo **type** → el formato del recurso



# Audio / Video

---

```
<audio>
```

```
    <source src="audio1.mp3" type="audio/mpeg">
```

```
    <source src="audio1.ogg" type="audio/ogg">
```

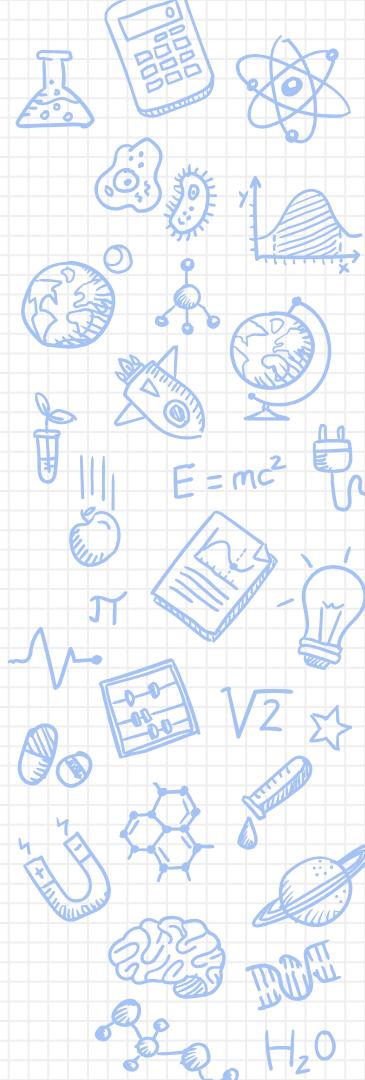
```
</audio>
```

```
<video>
```

```
    <source src="video2.mp4" type="video/mp4">
```

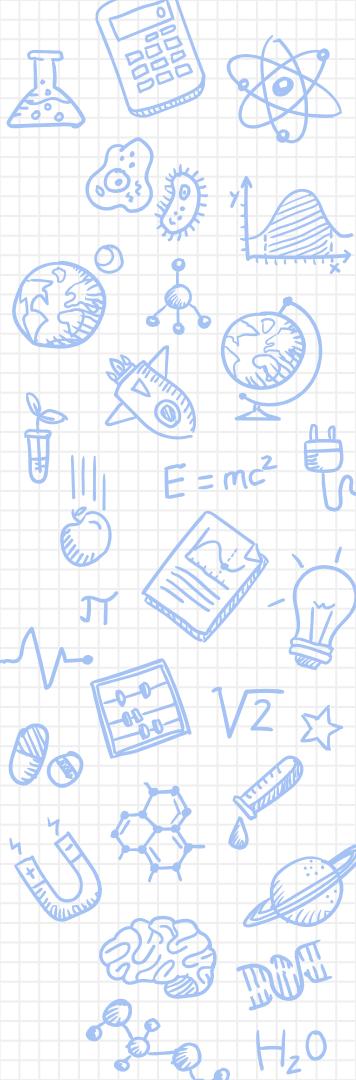
```
    <source src="video2.ogg" type="video/ogg">
```

```
</video>
```



# Audio / Video

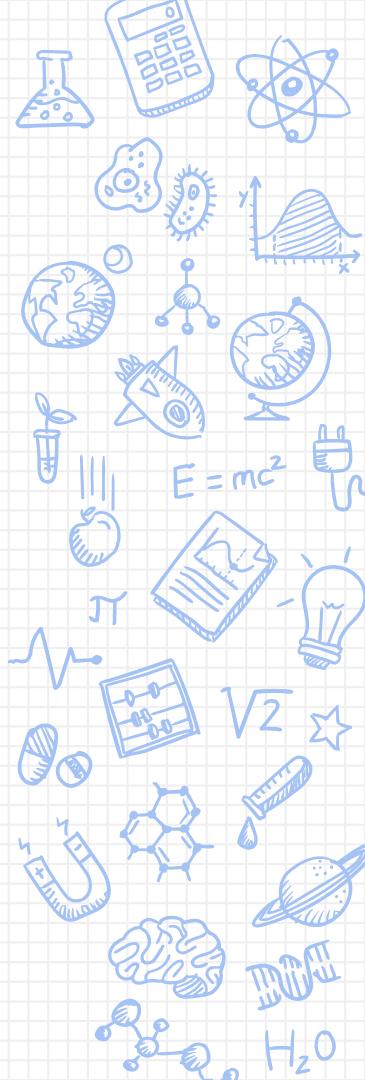
- Una práctica habitual es, luego de las etiquetas source, agregar un párrafo con un mensaje como “tu navegador no soporta la reproducción de ...”
  - Este mensaje sólo se va a mostrar al usuario en caso de que no se pueda visualizar ninguno de los formatos especificados.



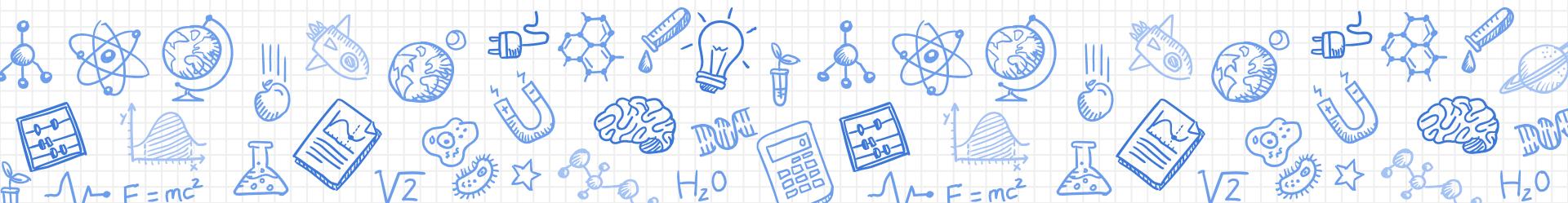
# Audio / Video

---

- Ahora bien, tenemos nuestras etiquetas. Pero no pasa nada.
- Esto se debe a que necesitamos conocer algunos atributos más:
  - Atributo **controls** → activa los controles para el usuario
  - Atributo **autoplay** → reproduce el recurso sin preguntar **X**
  - Atributo **loop** → repite el recurso infinitas veces **X**



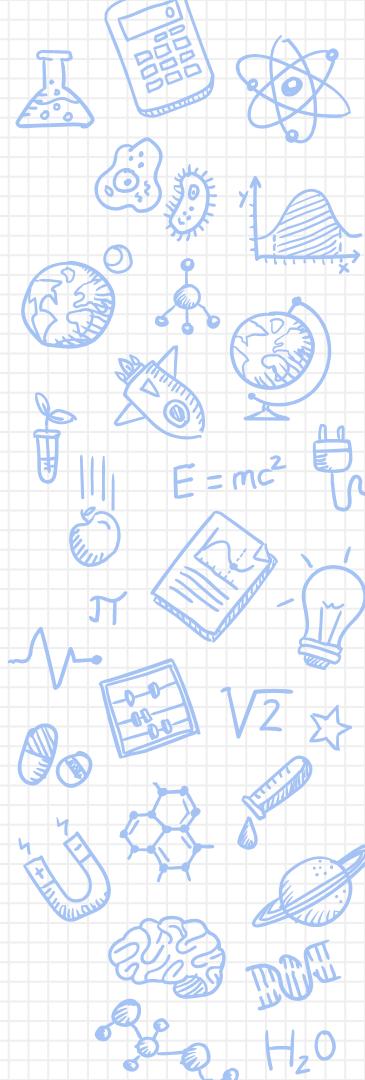
# Atributo ID



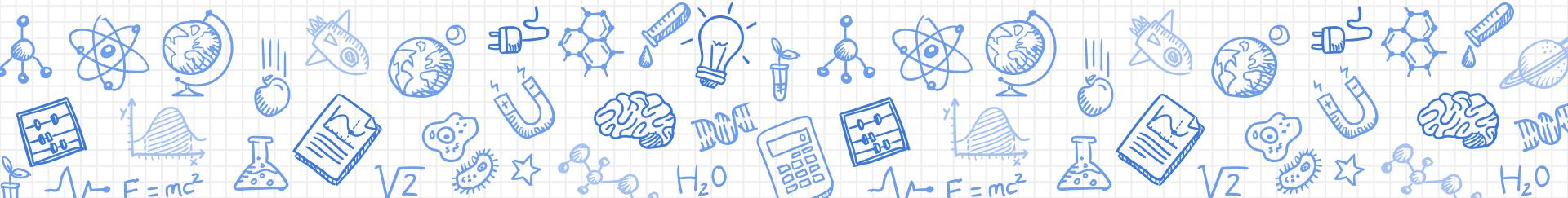
# ID

---

- El **id** (identifier) es un atributo que se puede agregar a cualquier elemento.
- La particularidad es que el valor **no se puede repetir en el mismo archivo html**.
- El id se usa principalmente para tareas de programación. Pero si vinculamos IDs con vínculos, podemos hacer la navegación de una landing page.



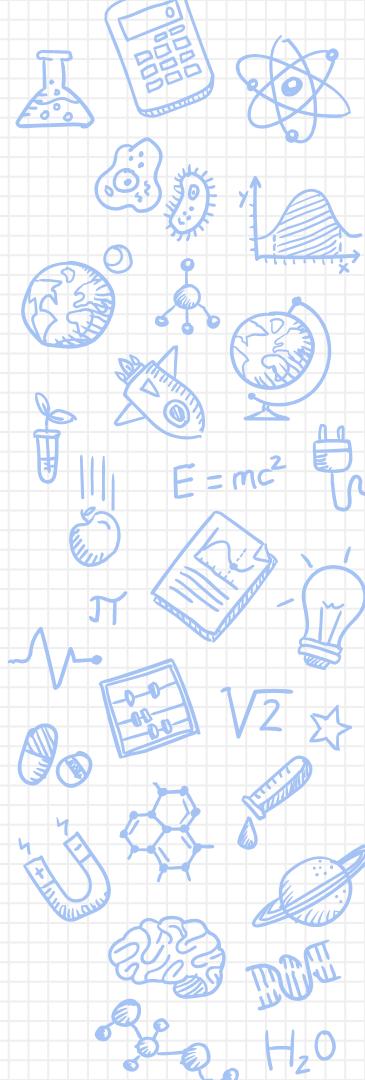
# Atributo CLASS



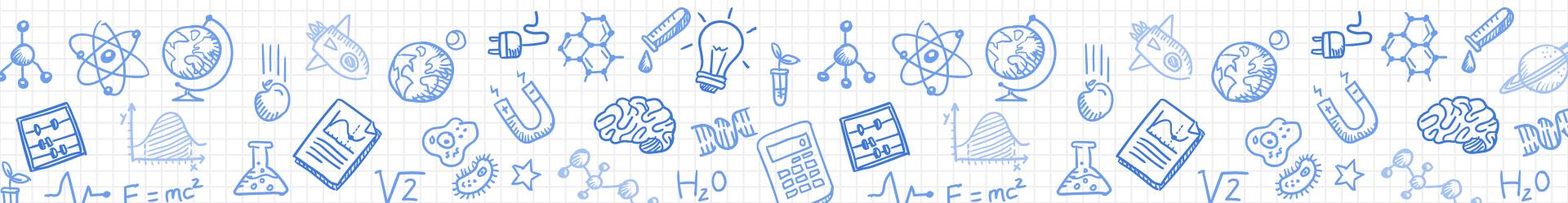
# CLASS

---

- Las clases son bastante similares a los IDs, también lo podemos agregar a cualquier elemento.
- La diferencia está en que, en este caso, podemos repetir los valores en diferentes elementos tantas veces como necesitemos.
- Las clases se usan principalmente para los estilos de la web.



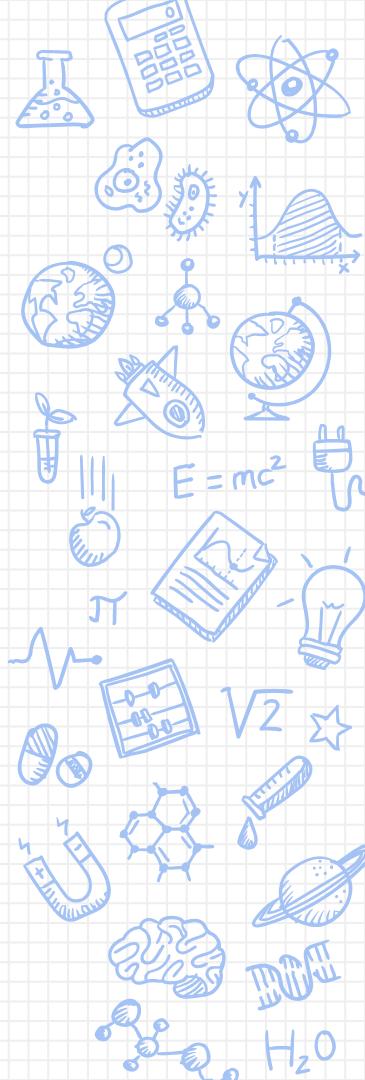
# Listas



# Listas

---

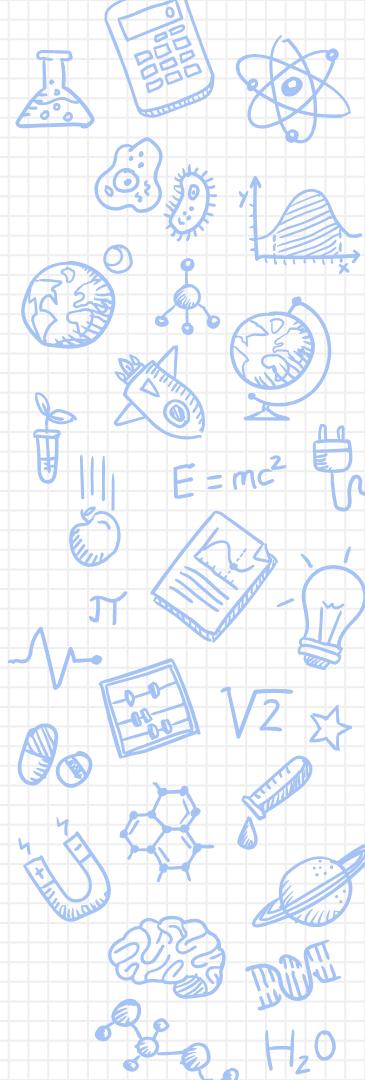
- Las listas son una estructura HTML que vamos a encontrar muy seguido.
- Se utilizan para:
  - Menú principal.
  - Galería de imágenes.
  - Redes sociales.
  - Y mucho más...



# Listas

---

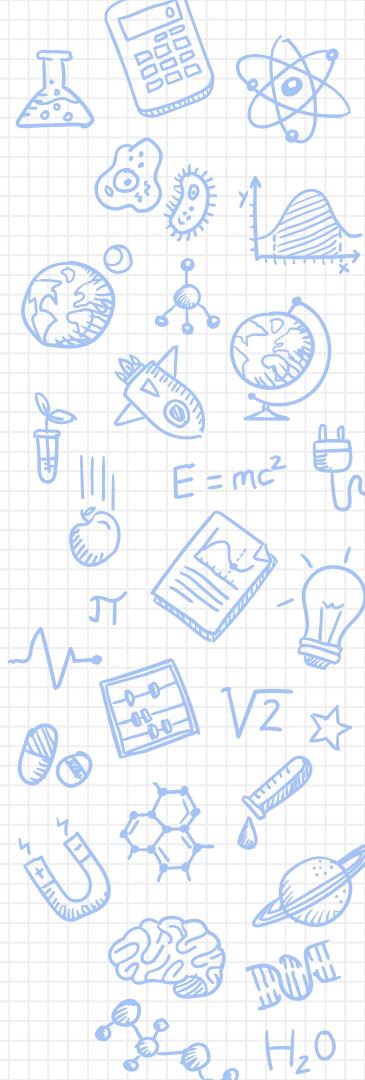
- Existen dos tipos de listas:
  - **Listas ordenadas:** son secuenciales, necesitan ese orden sí o sí. Por ejemplo: receta de cocina.
  - **Lista no ordenadas:** se utilizan para todo el resto de las cosas, que es el 99% de los casos.



# Listas ordenadas

---

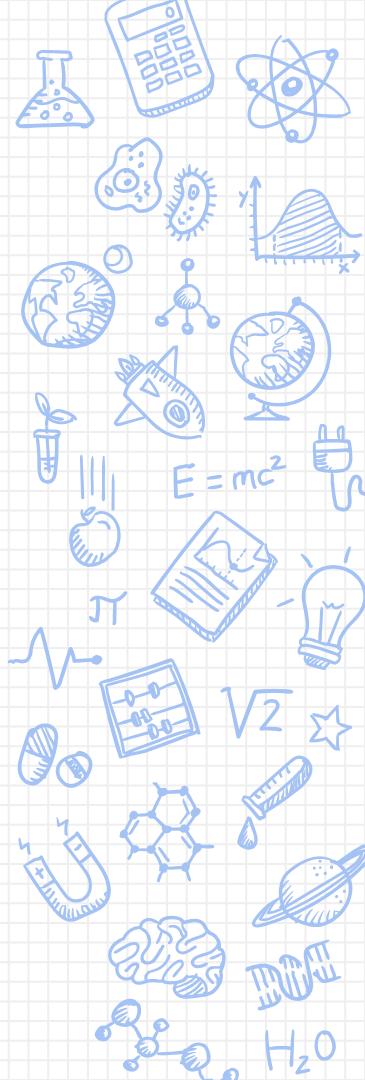
- Para insertar una lista ordenada vamos a utilizar la etiqueta **<ol></ol>**
- Y dentro de ella, tantas etiquetas **<li></li>** como items queramos.



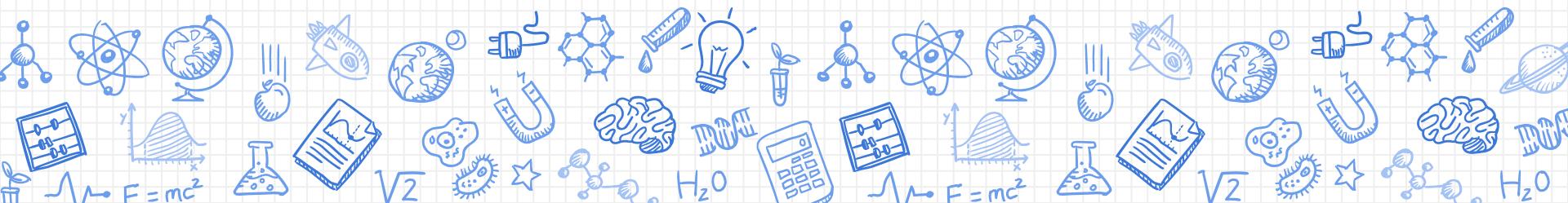
# Listas no ordenadas

---

- En el caso de las no ordenadas la etiqueta a utilizar es **<ul></ul>**
- Y dentro de ella también usamos etiquetas **<li></li>**.



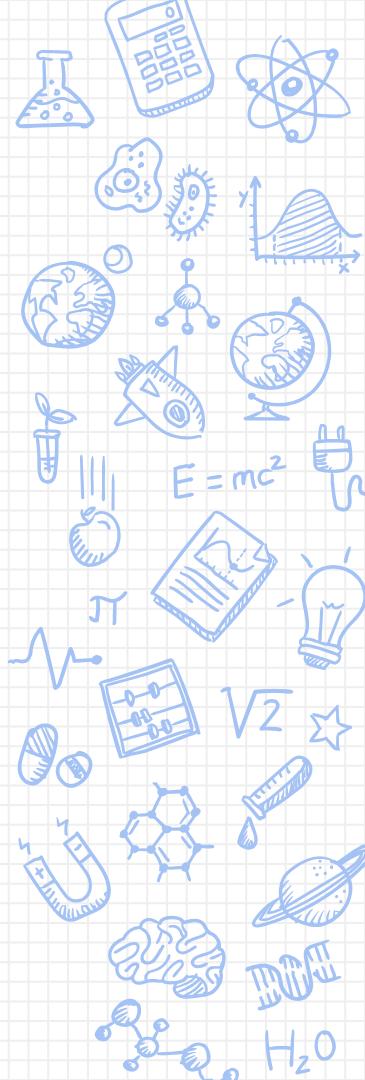
# Tablas



# Tablas

---

- Las tablas son una de las estructuras más viejas que existen en la web.
- De hecho, se usaban para maquetar sitios completos.
- Esto era sumamente costoso, y muy poco práctico. Y de esa manera, las tablas se ganaron el odio de los desarrolladores, que en cuanto pudieron reemplazarlas, las evitaron para siempre.

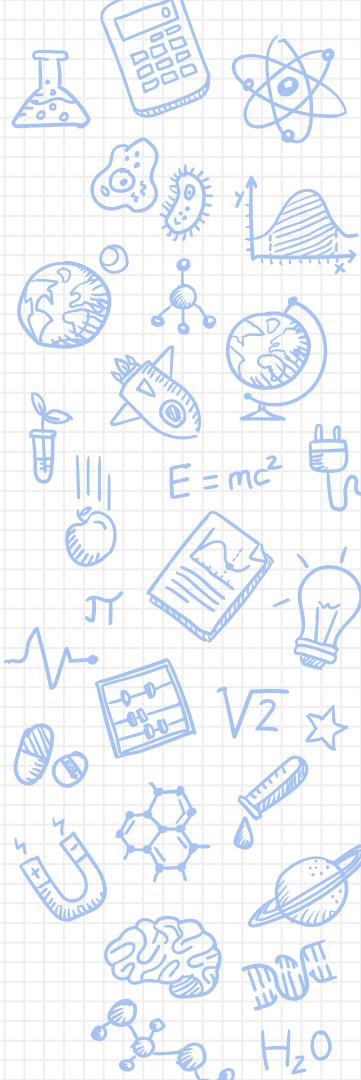


# Tablas

---

- Pero lo cierto es que si tenemos contenido tabular, debemos usar tablas.

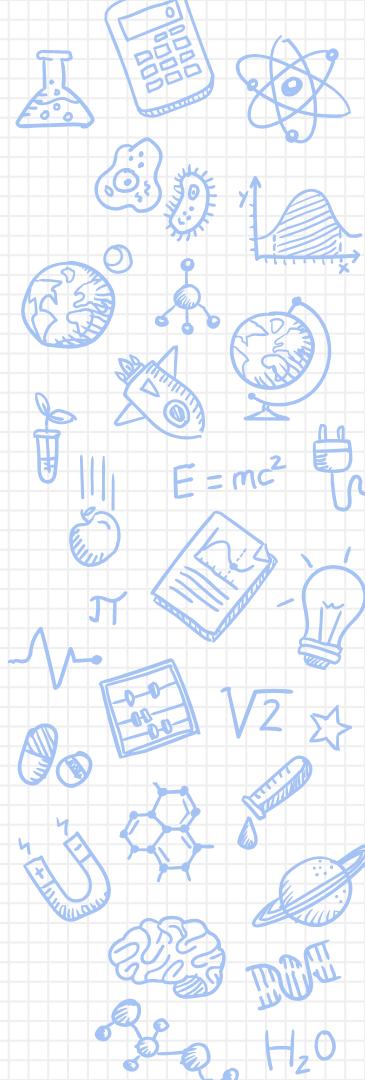
— ( ) —



# Tablas

---

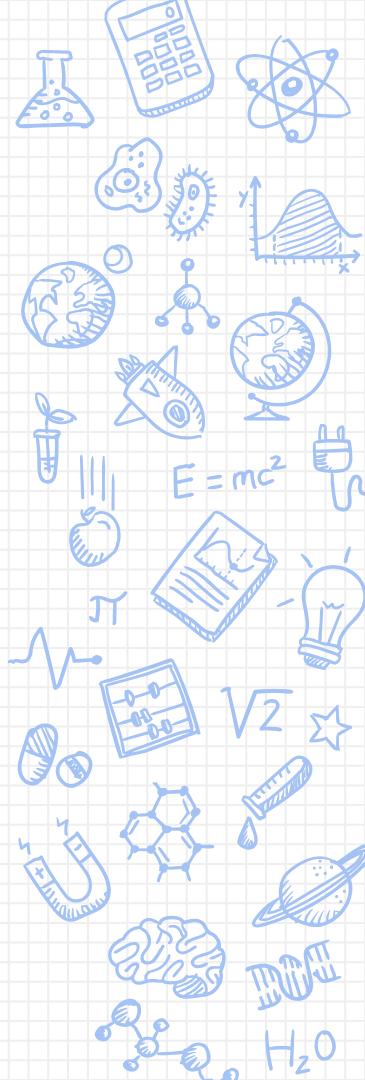
- La estructura de una tabla se compone de tres etiquetas principales:
  - **<table></table>**
  - **<tr></tr>** → table row
  - **<td></td>** → table division
- Y algunas etiquetas adicionales:
  - **<th></th>** → table heading
  - **<caption></caption>**
  - **thead - tbody - tfoot**



# Tablas

---

- Para definir una tabla, debemos utilizar la etiqueta **table** que va a definir desde dónde y hasta donde llega el contenido.
- Luego, debemos empezar a crear filas con la etiqueta **tr**.
- Y por último, generar columnas dentro de esas filas con la etiqueta **td**.



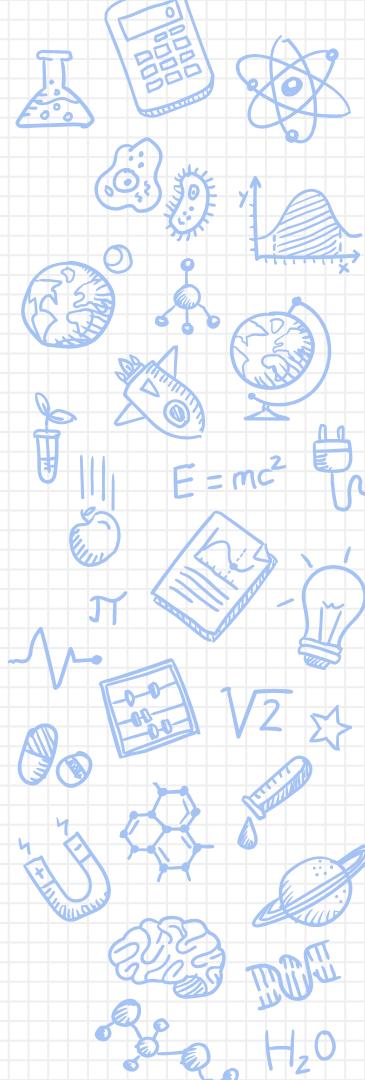
# Tablas

---

<table>

```
<tr> <!-- Empieza la fila 1 -->
    <td>Fila 1 - Columna 1</td>
    <td>Fila 1 - Columna 2</td>
</tr> <!-- Termina la fila 1 -->
```

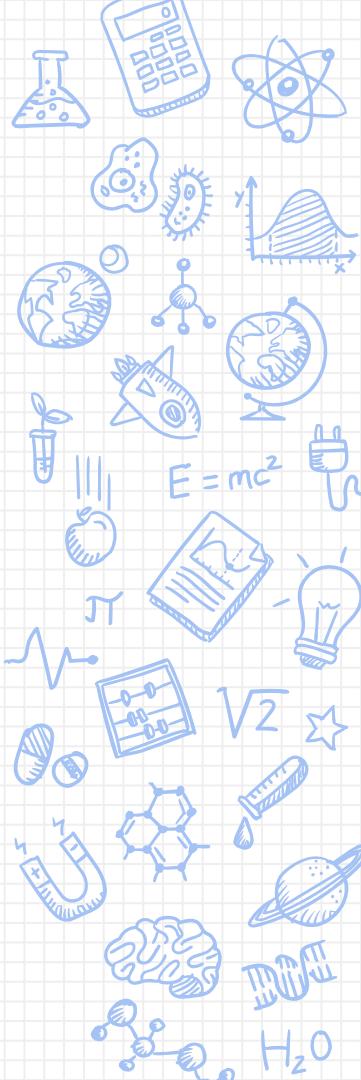
```
<tr> <!-- Empieza la fila 2 -->
    <td>Fila 2 - Columna 1</td>
    <td>Fila 2 - Columna 2</td>
</tr> <!-- Termina la fila 2 -->
</table>
```



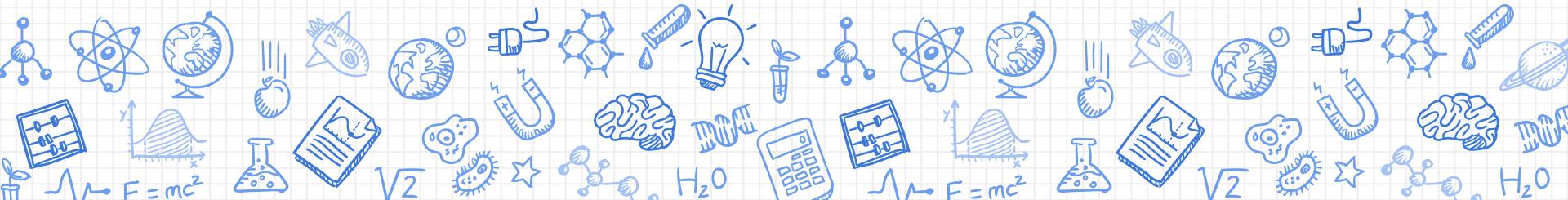
# Tablas

---

- La etiqueta **th**, se utiliza en la primera fila, en lugar de los **td**, para poner los encabezados de cada columna.
- La etiqueta **caption** se usa justo después de **table**, antes de la primera fila, para ponerle un título a la tabla.



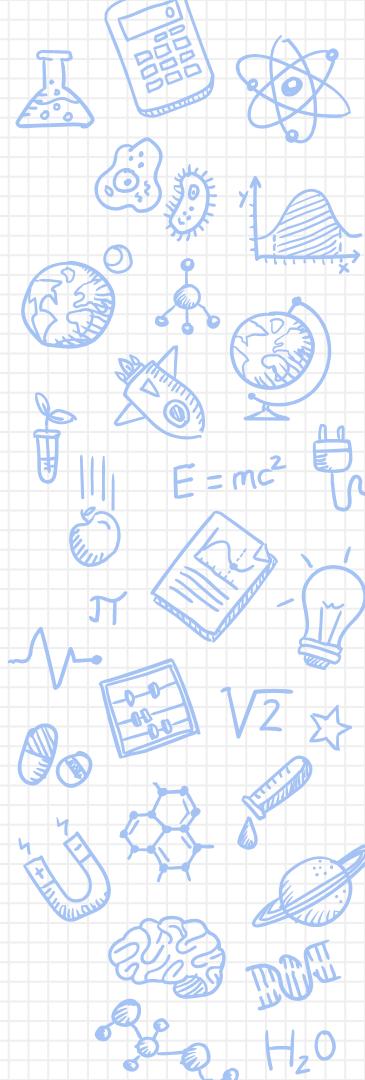
# Formularios



# Formularios

---

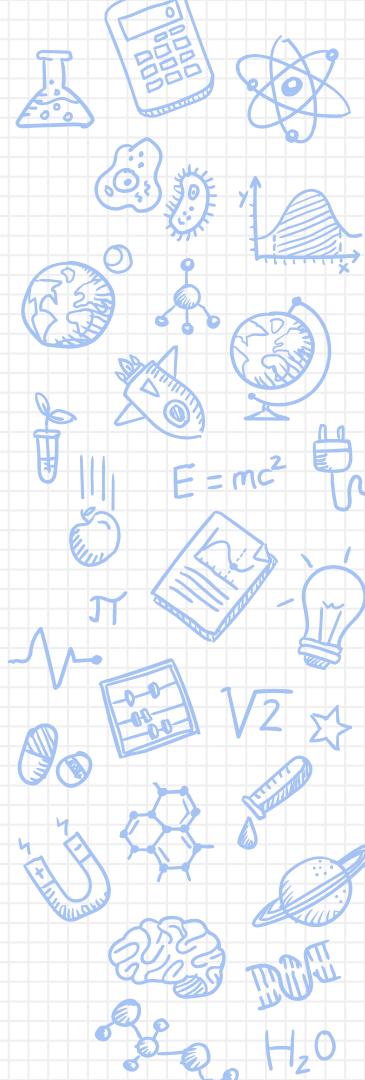
- Cuando pensamos en un formulario, enseguida se nos viene el de contacto. Nombre, email, mensaje y a otra cosa.
- ¿Pero si les pido 5 ejemplos, se les ocurre algo?



# Formularios

---

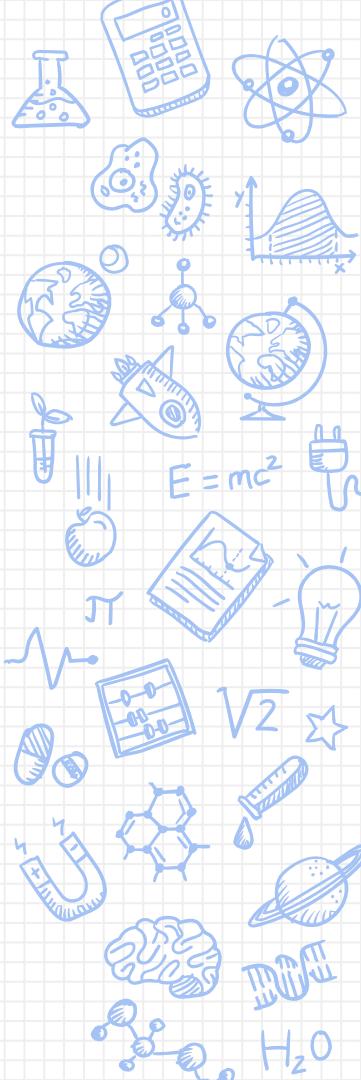
- Buscar en Google.
- Publicar en redes sociales.
- Comentar una publicación en un blog.
- Seleccionar un barrio o una provincia en un buscador de sucursales.
- Aceptar términos y condiciones sin leerlos.
- Iniciar sesión o registrarse en un sistema.



# Formularios

---

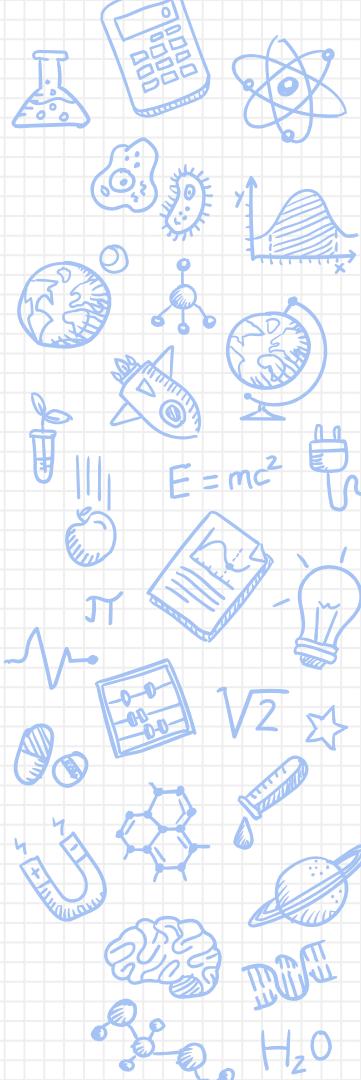
- Los formularios nos permiten hacer toda esa variedad de acciones y por eso son de suma importancia en la web.
- Desde una landing simple, pasando por un sistema interno de una empresa, hasta la red social más importante del mundo, están repletos de formularios.



# Formularios

---

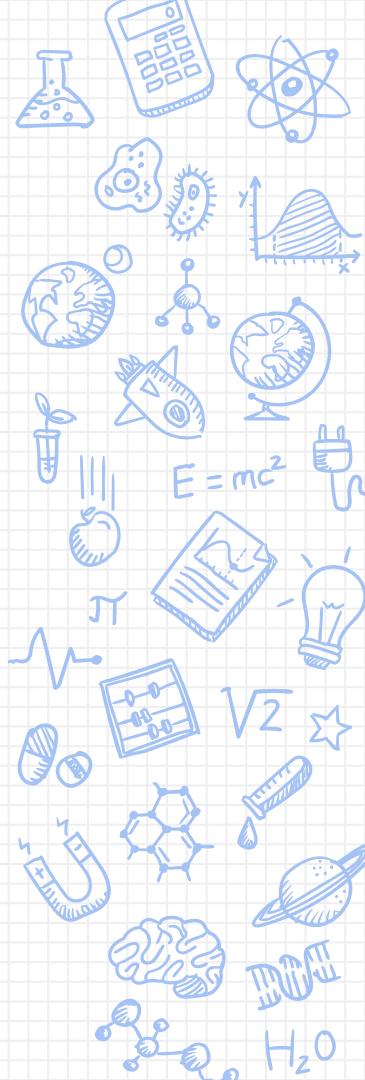
- En este curso vamos a estar usando solamente los formularios de contacto, y tal vez un buscador. PERO....  
Aclaración en la clase 2: ninguno de esos formularios va a funcionar.
- ¿Por qué? Para poder procesar la información que escribimos en un formulario, necesitamos un lenguaje que trabaje del lado del servidor.
- Nosotro solamente haremos la maqueta.



# Formularios

---

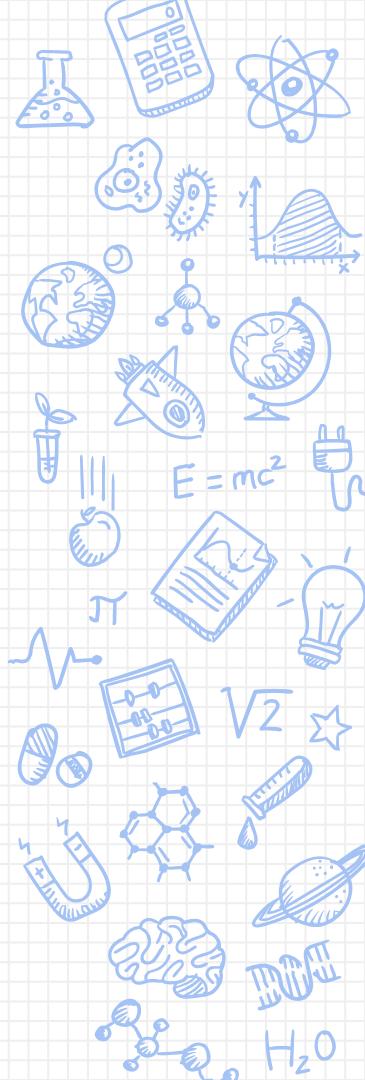
- Para crear un formulario tenemos la etiqueta **<form></form>**
- Esta etiqueta recibe tres atributos. Uno es obligatorio y dos opcionales:
  - **action** (obligatorio).
  - **method**.
  - **enctype**.



# Formularios – Action

---

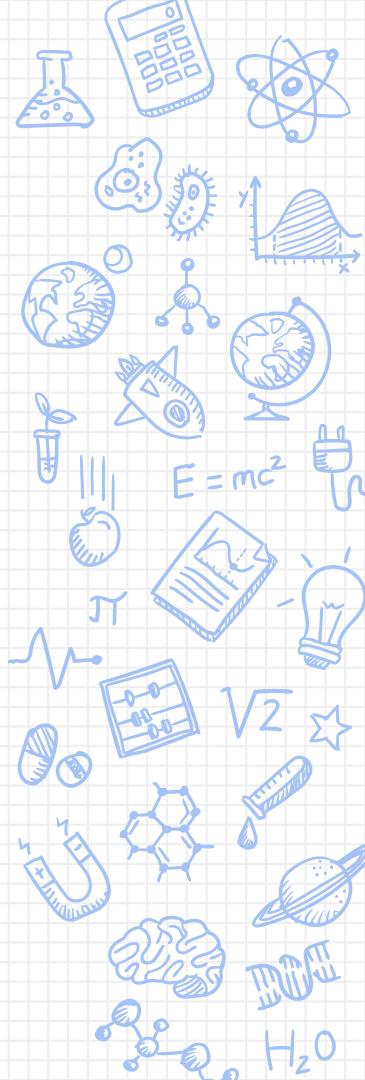
- El **action** es un atributo que, como valor, recibe una ruta que le indica cuál es el archivo que va a procesar la información una vez que se envíe el formulario.
- En nuestro caso vamos a usar un **#**, pero acá iría la ruta a un archivo de PHP, por ejemplo.



# Formularios – Method

---

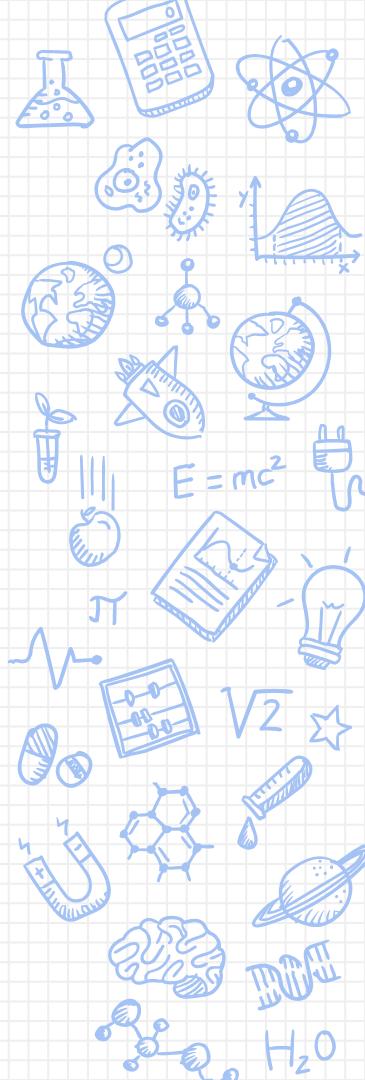
- El **method** es la forma en la que se va a enviar esa información. Tenemos dos formas:
  - **get**: envía la información por medio de la URL (por ejemplo Google). Es el método por defecto.
  - **post**: envía la información de forma oculta.
- No hay uno que sea mejor que el otro, tienen diferentes finalidades.



# Formularios – Method

---

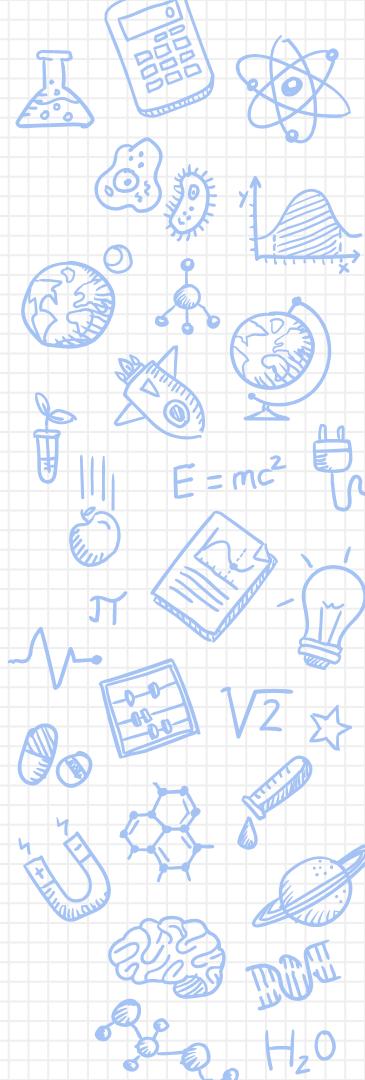
- El **get** por lo general se usa para búsquedas, filtros, etc. Cosas que no manejen información sensible y no tengan mucho volumen.
- El **post** se usa para login, registro, formularios de contacto, envío de imágenes, etc.



# Formularios – Method

---

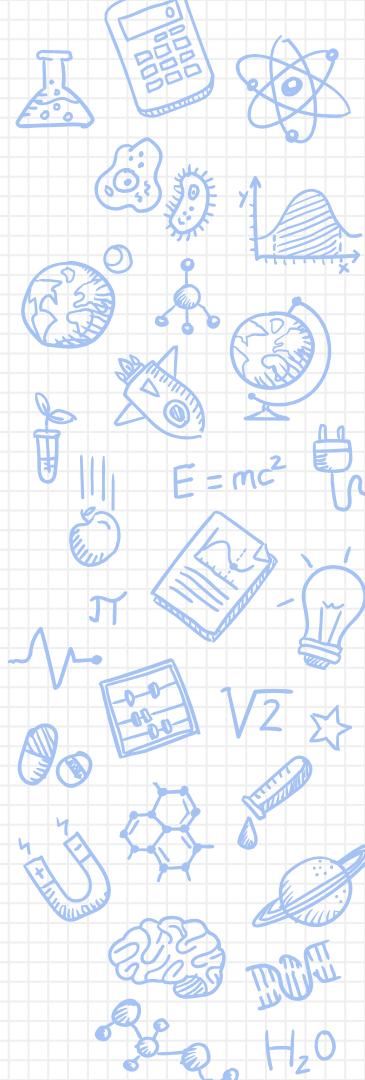
- Estos usos se deben a las características de cada uno. **Get** tiene la cantidad de información limitada al largo de una URL. Y no estaría bueno mandar contraseñas por la URL.
- Post envía de forma oculta y acepta mucha más cantidad de caracteres.



# Formularios – Enctype

---

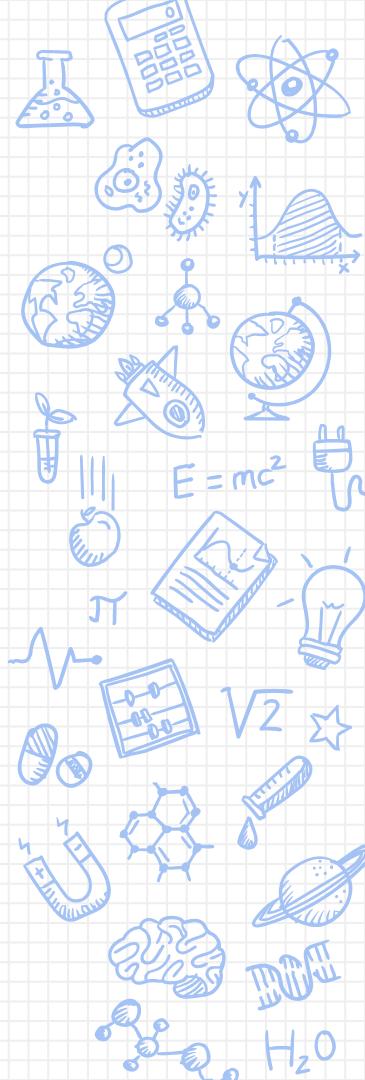
- El **enctype** es un atributo que solo lo vamos a modificar en caso de que necesitemos enviar un archivo adjunto.
- Por defecto, su valor es **application/x-www-form-urlencoded**, y si queremos enviar un archivo lo pasamos a **multipart/form-data**



# Formularios

---

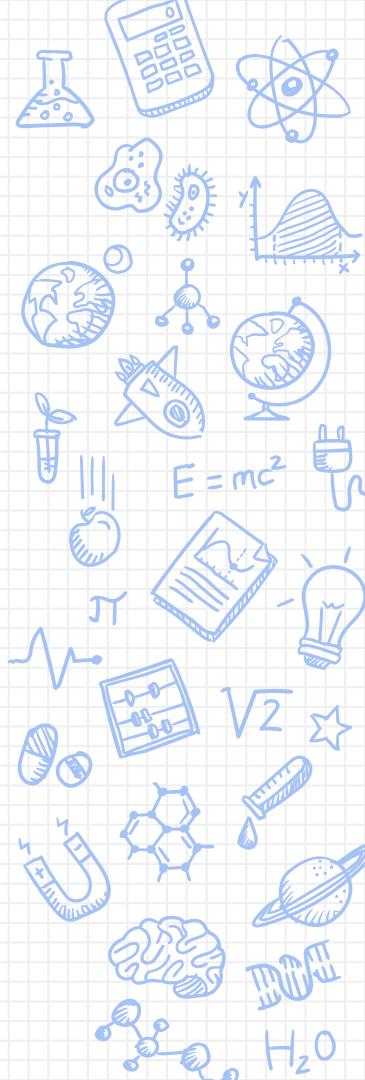
- Una vez dentro del formulario, podemos utilizar las siguientes etiquetas:
  - input.
  - textarea.
  - label.
  - fieldset.
  - legend.
  - select - option.



# Formularios – Input

---

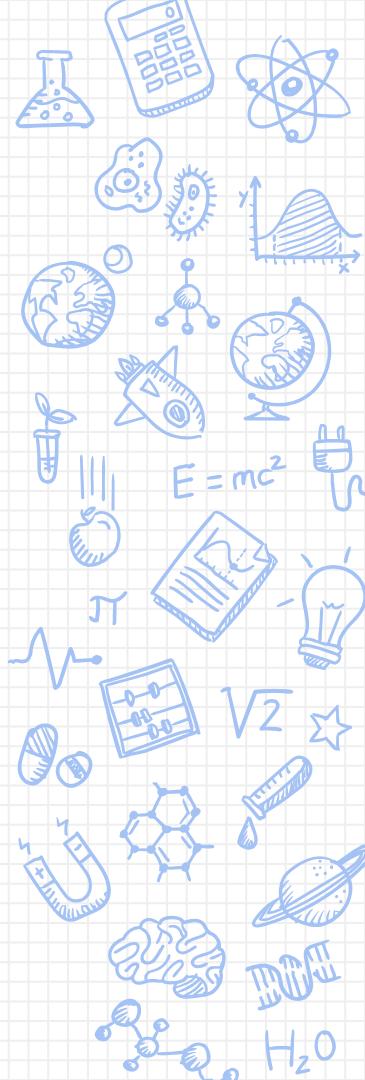
- El **input** es el típico campo de formulario para escribir el nombre.
- Solamente puede recibir textos en una línea.
- Existen inputs de los siguientes tipos:
  - Text → Ingreso de texto.
  - Password → Texto que no se muestra.
  - Number → Sólamente números.
  - Email → Valida que sea un email.
  - Checkbox → Aceptar o no un campo.
  - Radio → Elegir un sólo valor de un grupo.



# Formularios – Input (Atributos)

---

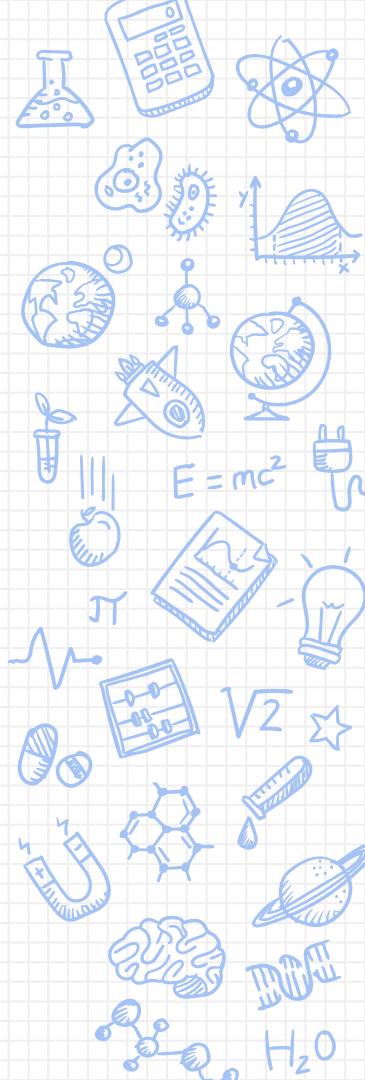
- Como el input es una etiqueta de autocierre, va a manejar su comportamiento por medio de atributos. Los atributos más comunes son:
  - Type → Es obligatorio.
  - Name → Es obligatorio.
  - Placeholder.
  - Required.

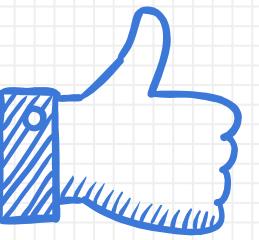


# Formularios – Input

---

- Para enviar el formulario también se utiliza una etiqueta input. Que es del tipo **submit**.
- En este caso no recibe name, pero va a llevar el atributo **value** que se va a encargar de indicar el texto que figura dentro del botón.





# SE ACABÓ!

## Dudas?

Escriban a:

✗ [federico.mirandaa@gmail.com](mailto:federico.mirandaa@gmail.com)