

Template strings

DigitalHouse >
Coding School



**Certified Tech
Developer**
The Ultimate Degree

Sintaxis de un template string

Texto/String

Este valor se mantendrá constante.

Código JavaScript

Este código se ejecutará cada vez que el template string sea utilizado.

```
JS ` Mi variable vale ${miVariable} `
```

Template String

Inician con comillas cruzadas, si intentan utilizar las comillas simples (') o dobles ("), JavaScript solo interpretará que es un texto.

Apertura de código

Con estas llaves indicamos que lo envuelto será interpretado como código JavaScript, es decir, si miVariable tiene un valor de 14, la salida de este template sería: **Mi variable vale 14.**

Ejemplos de uso

```
const nombre= 'Mauro';  
const miTemplate = `Mi nombre es ${nombre}`;  
console.log(miTemplate);
```

El resultado sería:
Mi nombre es mauro

```
const miTemplate = `2 * 3 es ${ 2 * 3 }`;  
console.log(miTemplate);
```

El resultado sería:
2 * 3 es 6

```
function suma(var1, var2) {  
    return var1 + var2  
}  
  
const temp = `El resultado es ${suma(2,2)}`;  
console.log(temp);
```

El resultado sería:
El resultado es 4

Dentro de las llaves se puede ejecutar cualquier código JS, pero es buena costumbre que sea lo más sencillo posible

3 | Modificar HTML

Contenido dinámico

El módulo pasado vimos dos funciones muy útiles para modificar nuestro documento HTML mediante JavaScript: `innerText` e `innerHTML`, ahora podemos probar el siguiente fragmento de código en tu computadora antes de continuar:

```
const template = `

# 

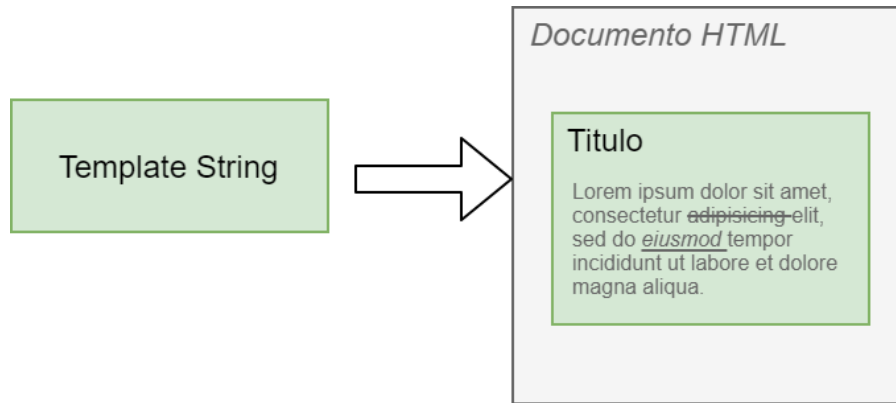

```



¡No olvidemos enlazar nuestro script al HTML!

InnerHTML

A diferencia de `innerText`, `innerHTML` permite incluir etiquetas en nuestro template string y que estas sean interpretadas correctamente por el navegador, esto nos trae la posibilidad de agregar muchísimo dinamismo a nuestras páginas.



¡Empecemos a probarlo!



html

js

```
<html lang="es"> <!-- JS en la siguiente hoja -->
<head>
  <meta charset="UTF-8">
  <script src="scripts.js"></script>
  <title>Ejemplo</title>
</head>
<body id="body">

</body>
</html>
```




html

js

```
function escribirHTML(titulo, texto) {  
  const body = document.getElementById('body');  
  const miTemplate = `  
    <h1>${titulo}</h1>  
    <p>${texto}</p>  
  `;  
  body.innerHTML += miTemplate;  
}  
  
escribirHTML('Hola', 'Esto es un ejemplo de template string en html.');
```

escribirHTML('Es dinámico', 'Podemos insertar elementos HTML mediante JavaScript');

escribirHTML('Facilita la programación', 'Evita escribir mucho código y reutilizar el que si escribamos.');

escribirHTML('En este ejemplo', 'Hemos utilizado una única función para poder escribir 4 veces en HTML, ¿te imaginás lo que sería esto sin esta función?');