Explicación paso a paso del uso de String.trim() en JavaScript

# Código original

// Definir una cadena de texto con espacios al principio y al final  
let texto = " JavaScript es genial ";  
  
// Mostrar la cadena original (con espacios)  
console.log("Cadena original:", `"${texto}"`);  
  
// Usar String.trim() para eliminar los espacios en blanco al principio y al final  
let textoSinEspacios = texto.trim();  
  
// Mostrar el resultado después de usar trim()  
console.log("Cadena sin espacios:", `"${textoSinEspacios}"`);

# Explicación paso a paso

## 1. Definir una cadena de texto con espacios al principio y al final

let texto = " JavaScript es genial ";  
• ¿Qué hace?  
 Crea una variable llamada "texto" que contiene la cadena con espacios extra al principio y al final.  
• ¿Para qué sirve?  
 Es útil para demostrar cómo eliminar espacios sobrantes en cadenas de texto, una tarea común al procesar datos de formularios o archivos.

## 2. Mostrar la cadena original (con espacios)

console.log("Cadena original:", `"${texto}"`);  
• ¿Qué hace?  
 Imprime la cadena tal cual está, incluyendo los espacios al principio y al final. Las comillas permiten ver claramente los espacios.  
• ¿Qué imprime?  
 Cadena original: " JavaScript es genial "

## 3. Usar String.trim() para eliminar los espacios en blanco al principio y al final

let textoSinEspacios = texto.trim();  
• ¿Qué hace?  
 Aplica el método trim() a la cadena, eliminando todos los espacios en blanco al inicio y al final, pero no los del medio.  
 " JavaScript es genial " → "JavaScript es genial"  
• ¿Para qué sirve?  
 Es ideal para limpiar datos que pueden haber sido ingresados con espacios innecesarios.

## 4. Mostrar el resultado después de usar trim()

console.log("Cadena sin espacios:", `"${textoSinEspacios}"`);  
• ¿Qué hace?  
 Imprime la nueva cadena, ya sin los espacios extra.  
• ¿Qué imprime?  
 Cadena sin espacios: "JavaScript es genial"

# Resumen sobre String.trim()

• trim() elimina todos los espacios en blanco al inicio y al final de una cadena.  
• No elimina los espacios dentro de la cadena, solo los de los extremos.  
• Muy útil para limpiar datos de formularios, archivos de texto, etc.  
• No modifica la cadena original, sino que devuelve una nueva cadena "limpia".