

CURSO DE JAVASCRIPT

**Corporación Ecuatoriana para el Desarrollo de la
Investigación y la Academia**

Ing. María Fernanda Granda J., PhD

11-11-2025

**“Conectando ideas,
transformando
sociedades”**

Módulo 2: Funciones

- ¿Qué es una Función en JavaScript?
- Declaración de Funciones
- Definición de Parámetros y Argumentos
- Parámetros de las Funciones
- Valores de Retorno
- Parámetros obligatorios y opcionales
- Formas de Definir una función
- ¿Qué es la escritura Camel Case en JavaScript?
- Funciones predefinidas en JavaScript

¿Qué es la escritura Camel Case en JavaScript?

- Es una forma de separar varias palabras de una variable o función.
- La primera palabra empieza con minúscula y luego se escribe la primera letra de cada palabra en mayúscula y sin espacios. Ejemplos:
- `let primerNombre="Maria";`
- `parseInt();`

Funciones predefinidas en JavaScript

- Son funciones globales, que se llaman globalmente sin hacer referencia a un objeto
- Algunas ya conocidas:
 - `alert()`
 - `prompt()`
 - `confirm()`
- Hacen referencia a un objeto
 - `console.log()`

Funciones predefinidas en JavaScript

- Otras nuevas y muy útiles
 - parseInt()
 - parseFloat()
 - toFixed()
 - isNaN()
 - isFinite()
 - encodeURIComponent()
 - decodeURI()
 - encodeURIComponent()
 - decodeURIComponent()
 - eval()

Funciones predefinidas en JavaScript

parseInt()

- toma un valor e intenta transformarlo en un número entero
- Si falla devuelve NaN
- **Ejemplos en Consola:**

```
> parseInt('123')  
< 123  
> parseInt('abc123')  
< NaN  
> parseInt('1abc23')  
< 1  
> parseInt('123abc')  
< 123
```

Funciones predefinidas en JavaScript

parseInt()

- Admite un segundo argumento opcional que indica la base del número que se le está pasando (decimal, hexadecimal, binario, etc).
- Se recomienda especificar la base para evitar problemas de interpretación.

```
> parseInt(" 0xF", 16);  
< 15  
> parseInt(" F", 16);  
< 15  
> parseInt("17", 8);  
< 15  
> parseInt(021, 8);  
< 15  
> parseInt("015", 10);  
< 15  
> parseInt(15.99, 10);  
< 15  
> parseInt("FXX123", 16);  
< 15  
> parseInt("1111", 2);  
< 15  
> parseInt("15*3", 10);  
< 15  
> parseInt("15e2", 10);  
< 15  
> parseInt("15px", 10);  
< 15  
> parseInt("12", 13);  
< 15
```

Funciones predefinidas en JavaScript

parseFloat()

- toma un valor e intenta transformarlo en un número de coma flotante (con decimales)

- **Ejemplos en Consola:**

```
> parseFloat('123')  
< 123  
  
> parseFloat('1.23')  
< 1.23  
  
> parseFloat('1.23abc.00')  
< 1.23  
  
> parseFloat('a.bc1.23')  
< NaN
```


Funciones predefinidas en JavaScript

toFixed()

- Indica el número de decimales para redondear
- **Ejemplos en Consola:**
- `>let numero= 47.788888;`
- `>numero.toFixed (2);`
- `>47.79`

Funciones predefinidas en JavaScript

isNaN()

- Comprueba si el valor que se le pasa es un número válido.
- Devuelve true en caso de que no lo sea.
- **Ejemplos en Consola:**

```
> isNaN(NaN)
< true
> isNaN(123)
< false
> isNaN(1.23)
< false
> isNaN(parseInt('abc123'))
< true
```

Funciones predefinidas en JavaScript

isFinite()

- Comprueba si el valor que se le pasa es un número no es ni Infinity ni NaN.

- **Ejemplos en Consola:**

```
> isFinite(Infinity)
< false
> isFinite(-Infinity)
< false
> isFinite(12)
< true
> isFinite(1e308)
< true
> isFinite(1e309)
< false
```

Funciones predefinidas en JavaScript

encodeURIComponent()

- Nos permite escapar (codificar) una URL reemplazando algunos caracteres por su correspondiente secuencia de escape UTF-8
- encodeURIComponent nos devuelve una URI usable (sólo codifica algunos caracteres)
- **Ejemplo en Consola:**

```
> var url = 'http://www.packtpub.com/scr ipt.php?q=this and that';  
> encodeURIComponent(url);  
< http://www.packtpub.com/scr%20ipt.php?q=this%20and%20that
```

decodeURI()

- Nos permite decodificar) un string codificado por encodeURIComponent().

Funciones predefinidas en JavaScript

encodeURIComponent() y decodeURIComponent()

- Lo mismo que encodeURI() pero esta función codifica (decodifica) TODOS los caracteres transformables.
- **Ejemplo en Consola:**

```
> encodeURIComponent(url);
```

```
< "http%3A%2F%2Fwww.packtpub.com%2Fscr%20ipt.php%3Fq%3Dthis%20and%20that"
```

Funciones predefinidas en JavaScript

eval()

- Toma una cadena de texto y la ejecuta como código JavaScript.
- No debe usarse básicamente por dos motivos:
- **Rendimiento:** es mucho más lento evaluar código en vivo que tenerlo directamente en el script.
- **Seguridad:** Teniendo en cuenta que ejecuta todo lo que se le pase puede ser un agujero de seguridad.
- **Ejemplo por consola:**

```
> eval('var ii = 2;')
> ii
< 2
```

Funciones predefinidas en JavaScript

- Hay más funciones y muy útiles. Ejemplos:
- matemáticas
 - `Math.sqrt()`
 - `Math.round()`
 - `Math.random()` ...
- Manejo de Strings
 - `length()`
 - `charAt(indice)`
 - `indexOf(substring)`
 - `split(separador)`, etc.

GRACIAS