



JS

Funciones predefinidas

Funciones predefinidas - eval

eval(cadena [,objeto])

Cadena: Una cadena que representa una expresión, sentencia o secuencia de sentencias en JavaScript. La expresión puede incluir variables y propiedades de objetos existentes.

Objeto: Un argumento opcional; si se especifica, la evaluación se restringe al contexto del objeto especificado.

```
eval(new String("2 + 2")); // devuelve un objeto String que contiene "2 + 2"  
eval("2 + 2"); // devuelve 4  
var expresion = new String("2 + 2");  
eval(expresion.toString());
```



Funciones predefinidas - parseInt

parseInt(cadena, base);

Cadena: Una cadena que representa el valor que se desea convertir.

Base: Un entero que representa la base de la mencionada cadena.

Si se encuentra con un carácter que no se corresponde con la base especificada devuelve NaN.

Si no se especifica la base o se especifica como 0, JavaScript asume lo siguiente:

Si el parámetro cadena comienza por "0x", la base es 16 (hexadecimal).

Si el parámetro cadena comienza por "0", la base es de 8 (octal). Esta característica está desaconsejada.

Si el parámetro cadena comienza por cualquier otro valor, la base es 10 (decimal).



Funciones predefinidas - parseInt

parseInt(cadena, base);

Ejemplos:

```
parseInt("F", 16); //Devuelve 15
parseInt("17", 8); ; //Devuelve 15
parseInt("15", 10); ; //Devuelve 15
parseInt(15.99, 10); ; //Devuelve 15
parseInt("Hello", 8); // No es un número en absoluto. Devuelve NaN
parseInt("0x7", 10); // No es de base 10. Devuelve NaN
```



Funciones predefinidas - parseFloat

parseFloat(cadena);

Cadena: Una cadena que representa el valor que se desea convertir.
Si el primer carácter no se puede convertir a número, devuelve NaN.

Ejemplos:

```
parseFloat("3.14"); // Devuelve 3,14  
parseFloat("314e-2"); // Devuelve 3,14  
parseFloat("0.0314E+2"); // Devuelve 3,14  
var cadena = "3.14"; parseFloat(cadena); // Devuelve 3,14  
parseFloat("3.14más caracteres no dígitos"); // Devuelve 3,14  
parseFloat("FF2"); // Devuelve NaN
```



Funciones predefinidas - isNaN

isNaN(valor)

Valor: El valor que se desea evaluar.

isNaN quiere decir: Is Not a Number.

Intenta convertir el argumento a número; si no puede, devuelve true; en caso contrario false.

Ejemplos:

```
isNaN(NaN) // devuelve true  
isNaN("string") // devuelve true  
isNaN("12") // devuelve false  
isNaN(12) // devuelve false
```



Funciones predefinidas - isFinite

isFinite(numero)

Valor: El valor que se desea evaluar.

Si el argumento es NaN, infinito positivo o negativo devuelve false; en caso contrario true.

Ejemplos:

```
isFinite(Infinity); // falso
isFinite(NaN); // falso
isFinite(-Infinity); //falso
isFinite(0); // verdadero
isFinite(2e64); // verdadero
isFinite("0"); // verdadero, hubiera sido falso en el caso de usar Number
// isFinite("0") es mas robusta.
```



Funciones predefinidas – Number / String

Number(objeto) / String(objeto)

Objeto: el objeto que quiere pasarse a Number o String.

Convierten el objeto pasado por parámetro a Number o String.

Ejemplos:

```
var hoy = new Date();  
document.write(String(hoy));
```



END



prof.jduran@iesalixar.org