



Carmelo José Jaén Díaz



Objetos. Métodos

-
- C.F.G.S. DAW
 - 6 horas semanales
 - Mes aprox. de impartición: Dic
 - iPasen - cjaedia071@g.educaand.es

Índice



Objetivo

Glosario

Interacción persona - ordenador

Objetivos

Características. Usable.

Características. Visual.

Características. Educativo y actualizado.

OBJETIVO



- Profundizar en el concepto de objeto.
- Conocer y manejar funciones relativas al lenguaje sobre arrays, strings, números.

GLOSARIO



Backtracking. Estrategia utilizada en algoritmos que resuelven problemas que tienen ciertas restricciones. Este término fue creado por primera vez por el matemático D. H. Lehmer en la década de los cincuenta.

BOM (Browser Object Model). Convención específica implementada por los navegadores para que JavaScript pudiese hacer uso de sus métodos y propiedades de forma uniforme.

Expresión regular. Secuencia de caracteres que forman un patrón determinado, generalmente un patrón de búsqueda.

NaN. Propiedad que indica Not a Number (valor no numérico).

Objeto window. Aquel que soportan todos los navegadores y que representa la ventana del navegador. Se estudiará en profundidad en capítulos posteriores.

URI (Uniform Resource Identifier). Cadena de caracteres que identifica un recurso en una red de forma unívoca. Una URI puede ser una URL, una URN o ambas.

GLOSARIO



URN. Localizador de recursos en la web que funciona de forma parecida a una URL, pero su principal diferencia es que no indica exactamente dónde se encuentra dicho objeto.

INTRODUCCIÓN



En los lenguajes de programación orientados a objetos, además de propiedades éstos tienen lo que se denominan **métodos de un objeto**, que son funciones asociadas o, podría decirse, propiedades en forma de función.

Para definir un método (en ES5) la sintaxis es la siguiente:

```
nombre_método: function () {  
    // Instrucciones  
}
```

Y para acceder a los métodos de un objeto:

```
nombre_objeto.nombre_método();
```

¡Al llamar a un método no olvides poner los paréntesis! De lo contrario en lugar de ejecutarse se mostrará su código.

SINTAXIS DE CREACIÓN DE MÉTODOS DE UN OBJETO



Los métodos son las acciones que podemos realizar sobre un objeto. En JavaScript, podemos decir que los métodos son propiedades que tienen dentro una función.

La sintaxis para definir un método de un objeto es la siguiente:

```
<nombre_método> : function([parametros_si_los_hubiese]) {  
    //Instrucciones  
}
```

SINTAXIS DE ACCESO A MÉTODOS DE UN OBJETO



La sintaxis para acceder a un método de un objeto es la siguiente:

```
<nombre_objeto>.<nombre_método>();
```

No podemos olvidar los paréntesis, ya que su función es la de llamar o invocar al método. Si no los escribimos, lo que hacemos es consultar el contenido de dicho método.

```
let docente = {nombre:"Carmelo",  
               apellido:"Jaén",  
  
               nombreCompleto: function(){  
                   return this.nombre + " " + this.apellido;  
               }  
};
```


SINTAXIS DE ACCESO A MÉTODOS DE UN OBJETO



En el objeto **docente**, hemos creado una propiedad a la que le hemos asignado una función, es decir, hemos definido un método para el objeto **docente** (devuelve la concatenación de las propiedades **nombre** y **apellido**).

En las instrucciones del método empleamos **this** ya que necesitamos acceder a las propiedades del propio objeto en el que nos encontramos.

```
alert (persona1.nombre); //Devuelve Carmelo  
alert (persona1.apellido); //Devuelve Jaén  
alert (persona1.nombreCompleto()); //Devuelve Carmelo Jaén
```

SINTAXIS DE ACCESO A MÉTODOS DE UN OBJETO



En cambio si nuestro esto:

```
alert (persona1.nombreCompleto);  
//Devuelve function (){ return this.nombre + " " + this.apellido;}
```