



Carmelo José Jaén Díaz



Objetos. Propiedades

-
- 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



Los objetos tienen lo que se denominan **propiedades de un objeto** (en otros lenguajes llamados atributos), que no son más que sus características.

Hay varios modos de **acceder a las propiedades de un objeto**:

`objeto.propiedad`

`objeto["propiedad"]`

`objeto[expresión]`

INTRODUCCIÓN



Además, podemos recorrer todas las propiedades que tiene un objeto utilizando un bucle **for in**, estudiado anteriormente.

Además, es posible **añadir y borrar propiedades** de un objeto en concreto de una manera tan sencilla como asignándole un valor en el primer caso, o utilizando la operación **delete** delante del objeto seguido de la propiedad a borrar.

Importante: No es lo mismo asignar o borrar una propiedad de un objeto en concreto que hacerlo del prototipo de un objeto. En este segundo caso se aplicaría a todos los objetos creados a partir de él.

SINTAXIS DE ACCESO A LAS PROPIEDADES DE UN OBJETO



Por simplicidad, vamos a emplear la definición de objetos utilizando un literal y posteriormente accederemos a ellas, aunque no es la manera recomendable de crear un objeto.

Existen tres maneras de acceder a las propiedades de un objeto:

- La primera, escribiendo el nombre de un objeto (.) y el nombre de la propiedad.

`objeto.propiedad`

- La segunda, escribiendo el nombre de un objeto y entre corchetes y entre comillas el nombre de la propiedad

`objeto["propiedad"]`

SINTAXIS DE ACCESO A LAS PROPIEDADES DE UN OBJETO



- Y por último, escribiendo el nombre de un objeto y entre corchetes el nombre de una expresión. Esa expresión hace referencia a la propiedad por lo tanto la igualaríamos a una cadena con el nombre de la propiedad.

`objeto[expresión]`

EJEMPLO

```
let docente = {  
  nombre: "Carmelo",  
  apellido: "Jaen",  
  inicio: 2020  
};
```

SINTAXIS DE ACCESO A LAS PROPIEDADES DE UN OBJETO



EJEMPLO

A continuación, accederemos a las propiedades del objeto mediante los tres métodos citados anteriormente.

1. `alert (docente.nombre);`
2. `alert (docente["apellido"]);`
3. `let expresion = "inicio";`
`alert (docente[expresion]);`

En el ejemplo anterior, se ha accedido a las propiedades conociendo su nombre, si quisiéramos acceder a las mismas sin conocer su nombre, podemos recorrerlas con un bucle `for in`.

RECORRER LAS PROPIEDADES DE UN OBJETO

for in



La sintaxis sería la siguiente:

```
for (<nombre_variable> in <nombre_objeto>){  
    <nombre_objeto>[<nombre_variable>];  
}
```

EJEMPLO

```
for (x in docente){  
    alert(docente[x]);  
}
```

AÑADIR UNA NUEVA PROPIEDAD A UN OBJETO



Si queremos añadir una nueva propiedad a un objeto existente, podemos hacer:

```
<nombre_objeto>.<nueva_propiedad> = <valor_propiedad>;
```

Por ejemplo, para añadir la propiedad **modulo** al objeto **docente**:

```
docente.modulo = "DWECC";  
//para visualizar el cambio podemos ejecutar de nuevo el bucle for in o  
hacer lo siguiente  
alert(docente.modulo);
```

BORRAR UNA NUEVA PROPIEDAD A UN OBJETO



Si queremos eliminar una propiedad existente a un objeto existente, podemos hacer:

```
delete <nombre_objeto>.<nombre_propiedad>;
```

Por ejemplo, para eliminar la propiedad **inicio** al objeto **docente**:

```
delete docente.inicio;  
// mostramos el objeto mediante un bucle con los ultimos cambios  
for (x in docente){  
    alert(docente[x]);  
}
```