



Clases. Métodos estáticos (static)

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

Carmelo José Jaén Díaz

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 esta lección vamos a ver los métodos estáticos, para lo cual utilizamos la palabra reservada **static**.

Al igual que ocurre en otros lenguajes de programación, un método estático se llama directamente sin instanciar la clase (es decir, sin necesidad de crear un objeto). De hecho, si tratáramos de llamar a un método estático a partir de un objeto obtendríamos un error.

Este tipo de métodos se utilizan sobre todo para crear funciones de utilidad en una aplicación.

Recuerda: Los métodos estáticos en programación no pueden ser llamados a partir de un objeto instanciado. Deben llamarse directamente desde la clase.

INTRODUCCIÓN



Para crear un método estático no hay más que utilizar la siguiente sintaxis:

```
static nombreMetodo (parametros) { //código }
```

MÉTODOS ESTÁTICOS



Utilizamos la palabra reservada **static** para definir un método estático en una clase.

Al igual que en otros lenguajes de programación, un método estático se llama directamente sin instanciar la clase (de hecho, no puede hacerse mediante una instancia de clase).

Se suelen utilizar para crear funciones útiles en una aplicación.

No es necesario crear un objeto para llamar a un método estático.

SINTAXIS DE LOS MÉTODOS ESTÁTICOS



A continuación, vamos a crear una clase **Rectangulo**:

```
class Rectangulo {  
    constructor(x, y){  
        this._x = x;  
        this._y = y;  
    }  
}
```

Ahora vamos a implementar un método de clase estático para calcular el área y otro para el perímetro:

SINTAXIS DE LOS MÉTODOS ESTÁTICOS



```
class Rectangulo {  
    constructor(x, y){  
        this._x = x;  
        this._y = y;  
    }  
  
    area ( a, b) {  
        return a * b;  
    }  
    static perimetro (a, b){  
        return a + a + b + b;  
    }  
}
```

SINTAXIS DE LOS MÉTODOS ESTÁTICOS



A continuación, instanciamos un objeto `rectangulo1` de la clase `Rectangulo`:

```
let rectangulo1 = new Rectangulo (2,3);  
console.log(rectangulo1.area(2,3));  
//Devuelve 6  
console.log(rectangulo1.perimetro(2,3));  
//Devuelve "rectangulo1.perimetro is not a function", ya que al ser estático  
no lo reconoce como función
```

Para acceder al método estático, no hay que definir un objeto, basta con invocar directamente el método:

```
console.log(Rectangulo.perimetro(2,3)); //Devuelve 10
```