

DAWE

Arrays. Profundización

Departamento de LSI

Juanan Pereira <juanan.pereira@ehu.es>

Arrays

Como hemos visto, un array puede guardar distintos valores bajo un mismo nombre. Es posible acceder a esos valores referenciándolos con un índice.

```
var marcas = new Array();
```

```
marcas[0]="Saab";
```

```
marcas[1]="Volvo";
```

```
marcas[2]="BMW";
```

Varias formas de crear arrays

❶ Tradicional

```
var equipos = new Array();  
  
equipos[0]="Real Sociedad";  
  
equipos[1]="Athletic Bilbao";  
  
equipos[2]="Osasuna";
```

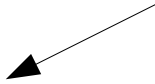
❷ Condensada

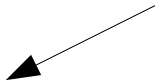
```
var equipos =new Array("Real Sociedad","Athletic Bilbao","Osasuna");
```

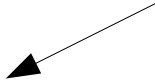
❸ Por medio de literales

```
var equipos=["Real Sociedad","Athletic Bilbao","Osasuna"];
```

Arrays con objetos de distinto tipo

`miArray[0] = Math.random();`  float

`miArray[1] = "una cadena";`  String

`miArray[2] = equipos;`  Array de Strings!

```
> miArray = new Array(Math.random(), "una cadena", equipos);  
[0.9630490869749337, "una cadena", ► Array[3] ]
```

Métodos de los objetos Array

Referencia: http://www.w3schools.com/jsref/jsref_obj_array.asp

```
var unos = ["Cecilie", "Lone"];  
  
var otros = ["Emil", "Tobias", "Linus"];  
  
var todos = unos.concat(otros);
```

Cecilie,Lone,Emil,Tobias,Linus

Otros métodos del objeto Array:

concat, indexOf, join, lastIndexOf, pop, push, reverse, shift, slice, sort, splice, toString, unshift, valueOf

No olvidarse de la **propiedad** length --> lenguajes.length // devuelve 4

Métodos de los objetos Array

```
var equipos = ["Athletic de Bilbao", "Barcelona", "Real Sociedad"];  
  
var ind = equipos.indexOf("Real Sociedad"); // 2
```

```
var frutas = ["Plátano", "Naranja", "Manzana", "Mango"];  
  
var energia = frutas.join(".");  
  
energia  
  
"Plátano,Naranja,Manzana,Mango"
```

Otros métodos del objeto Array:

concat, indexOf, join, lastIndexOf, pop, push, reverse,
shift, slice, sort, splice, toString, unshift, valueOf

Métodos de los objetos Array

Referencia: http://www.w3schools.com/jsref/jsref_obj_array.asp

```
var lenguajes = ["Pascal", "Ada", "C++", "Java", "JavaScript", "Ada"];  
  
lenguajes.lastIndexOf("Ada"); // 5
```

Otros métodos del objeto Array:

concat, indexOf, join, lastIndexOf, pop, push, reverse,
shift, slice, sort, splice, toString, unshift, valueOf

Métodos de los objetos Array

Referencia: http://www.w3schools.com/jsref/jsref_obj_array.asp

```
var palos = ["oros", "copas", "espadas", "bastos"];  
palos.pop();  
"bastos"  
  
palos  
["oros", "copas", "espadas"]
```

Otros métodos del objeto Array:

concat, indexOf, join, lastIndexOf, pop, push, reverse,
shift, slice, sort, splice, toString, unshift, valueOf

Métodos de los objetos Array

Referencia: http://www.w3schools.com/jsref/jsref_obj_array.asp

```
var palos = ["oros", "copas", "espadas"];  
palos.push("bastos");  
4  
palos  
["oros", "copas", "espadas", "bastos"]
```

Otros métodos del objeto Array:

concat, indexOf, join, lastIndexOf, pop, push, reverse,
shift, slice, sort, splice, toString, unshift, valueOf

Métodos de los objetos Array

Referencia: http://www.w3schools.com/jsref/jsref_obj_array.asp

```
var palos = ["oros", "copas", "espadas", "bastos"];  
palos.reverse();  
["bastos", "espadas", "copas", "oros"]  
  
palos  
["bastos", "espadas", "copas", "oros"]
```

Otros métodos del objeto Array:

concat, indexOf, join, lastIndexOf, pop, push, reverse, shift, slice, sort, splice, toString, unshift, valueOf

Estructuras de datos.

Arrays Multidimensionales

Podemos crear arrays de arrays en JavaScript, que pueden ser tratados como arrays multidimensionales.

1	2
3	4
5	6

Estructuras de datos.

Arrays Multidimensionales

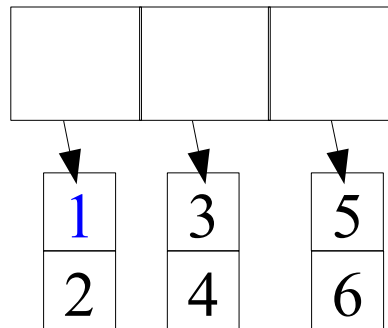
No existe una sintaxis específica para crear arrays multidimensionales en JavaScript, pero pueden simularse

```
var elementos = [[1,2],[3,4],[5,6]];
alert(elementos[0][0]);
```

Lo que entendemos:

1	2
3	4
5	6

Aunque realmente:



Estructuras de datos.

Arrays Multidimensionales

Otro ejemplo con un array de 10x20 elementos:

```
var x = new Array(10);  
  for (var i = 0; i < 10; i++) {  
    x[i] = new Array(20);  
  }  
x[5][12] = 3.0;
```

