



2.A. Estructura del lenguaje JavaScript

5.- Operadores.

5.7.- Operadores misceláneos.

El operador coma ,

Este operador, indica una serie de expresiones que van a ser evaluadas en secuencia, de izquierda a derecha. La mayor parte de las veces, este operador se usa para combinar múltiples declaraciones e inicializaciones de variables en una única línea.

Ejemplo:

```
var nombre, direccion, apellidos, edad;
```

Otra situación en la que podemos usar este operador coma, es dentro de la expresión **loop**. En el siguiente ejemplo inicializamos dos variables de tipo contador, y las incrementamos en diferentes porcentajes. Cuando comienza el bucle, ambas variables se inicializan a 0 y a cada paso del bucle una de ellas se incrementa en 1, mientras que la otra se incrementa en 10.

```
for (var i=0, j=0 ; i < 125; i++, j+10) {  
    // más instrucciones aquí dentro  
}
```

Nota: no confundir la coma, con el delimitador de parámetros “,” en la instrucción **for**.

? : (operador condicional)

Este operador condicional es la forma reducida de la expresión **if else**.

La sintaxis formal para este operador condicional es:

condicion ? expresión si se cumple la condición: expresión si no se cumple;

Si usamos esta expresión con un operador de asignación:

```
var = condicion ? expresión si se cumple la condición: expresión si no se cumple;
```

Ejemplo:

```
var a,b;  
a = 3; b = 5;  
var h = a > b ? a : b; // a h se le asignará el valor 5;
```

typeof (devuelve el tipo de valor de una variable o expresión).

Este operador unario se usa para identificar cuando una variable o expresión es de alguno de los siguientes tipos: **number, string, boolean, object, function** o **undefined**.

Ejemplo:

```
if (typeof miVariable == "number") {  
    miVariable = parseInt(miVariable);  
}
```

Créditos de la imagen

Autoría: alexanderdrachmann.

Licencia: CC BY-SA 2.0.

