

# TEMA 3F

OPERADORES ARITMÉTICOS.

CONVERSIÓN TIPOS.

OPERADORES DE COMPARACIÓN

Y DE ASIGNACION

# 1. OPERADORES ARITMETICOS

- Los operadores aritméticos toman los valores numéricos (literales o variables) como sus operandos y devuelve un solo valor numérico.

Operador	Nombre	Ejemplo	Descripción
+	Suma	5 + 6	Suma dos números
-	Substracción	7 - 9	Resta dos números
*	Multiplicación	6 * 3	Multiplica dos números
/	División	4 / 8	Divide dos números
%	Módulo: el resto después de la división	7 % 2	Devuelve el resto de dividir ambos números, en este ejemplo el resultado es 1
++	Incremento.	a++	Suma 1 al contenido de una variable.
--	Decremento.	a--	Resta 1 al contenido de una variable.
-	Invierte el signo de un operando.	-a	Invierte el signo de un operando.

# 1. OPERADORES ARITMETICOS

## ■ Ejemplo:

```
a = 8;  
b = 3;  
document.write(a + b);  
document.write("<br>");  
document.write(a - b);  
document.write("<br>");  
document.write(a * b);  
document.write("<br>");  
document.write(a / b);  
document.write("<br>");  
a++;  
document.write(a);  
document.write("<br>");  
b--;  
document.write(b);  
document.write("<br>");
```



11  
5  
24  
2.66666666666666666665  
9  
2

## 2. CONVERSIÓN DE TIPOS DE DATOS

### A) CONVERSION DE TEXTO A NUMERO


- **Number()** → convierte el texto a numero. Esta función mantiene la precisión de origen: si es entero convierte a entero y si es decimal, a decimal.

## 2. CONVERSIÓN DE TIPOS DE DATOS

### B) CONVERSION DE NUMERO A TEXTO

- Para esto se utiliza la función **String()**
- Se produce el caso contrario del apartado A). En este caso convertimos a texto lo que se pase como parámetro a dicha función.

```
var telefono=355;  
var telefonoTexto=String(telefono);  
document.write(telefonoTexto+7); //3557
```



3557

# 3. OPERADORES DE ASIGNACION

- Sirven para asignar valores a las variables, ya hemos utilizado en ejemplos anteriores el operación =, pero hay otros operadores de este tipo:

- = Asignación. Asigna la parte de la derecha del igual a la parte de la izquierda. A la derecha se colocan los valores finales y a la izquierda generalmente se coloca una variable donde queremos guardar el dato.
- += Asignación con suma. Realiza la suma de la parte de la derecha con la de la izquierda y guarda el resultado en la parte de la izquierda.
- -= Asignación con resta
- \*= Asignación de la multiplicación
- /= Asignación de la división
- %= Se obtiene el resto y se asigna

### 3. OPERADORES DE ASIGNACION

```
var x=1;  
document.write(x);  
document.write("<br>");  
//x=x+3;  
x+=3;
```

```
x=x+1;  
x++;  
x--; // x=x-1  
document.write(x);
```



1  
5

# 4. OPERADORES DE COMPARACIÓN

- Un operador de la comparación compara sus operando y devuelve un valor lógico basado en si la comparación es verdad o no. Los operandos pueden ser numéricos o cadenas.

Operador	Descripción
==	" Igual a" devuelve true si los operandos son iguales
===	Estrictamente "igual a"
!=	" No igual a" devuelve true si los operandos no son iguales
!==	Estrictamente " No igual a"
>	" Mayor que" devuelve true si el operador de la izquierda es mayor que el de la derecha.
>=	" Mayor o igual que " devuelve true si el operador de la izquierda es mayor o igual que el de la derecha.
<	" Menor que" devuelve true si el operador de la izquierda es menor que el de la derecha.
<=	"Menor o igual que" devuelve true si el operador de la izquierda es menor o igual que el de la derecha.



# 5. OPERADORES LOGICOS

- Los operadores lógicos se usan típicamente con valores Boolean. En tal caso, regresan un valor Boolean

Operador	Uso	Descripción
AND (&&) lógico	<code>expr1</code> && <code>expr2</code>	Regresa <code>expr1</code> si tal puede convertirse a false; de lo contrario, regresa <code>expr2</code> . De esta forma, cuando se usa con valores Boolean, && regresa true si ambos operandos son verdaderos; de lo contrario regresa false.
OR (  ) lógico	<code>expr1</code>    <code>expr2</code>	Regresa <code>expr1</code> si tal puede convertirse a true; de lo contrario, regresa <code>expr2</code> . De esta forma, cuando se usa con valores Boolean,    regresa true si cualquier operando es verdadero; pero si ambos son falsos, regresa "false".
NOT (!) lógico	<code>!expr</code>	Regresa false si su único operando puede convertirse a true; de lo contrario, regresa true.