

Front End

Linguagem JavaScript



Objetivo: Apresentar os diversos tipos de operadores em Javascript

Operadores em Javascript

Operadores são utilizados para realizar operações entre um ou mais valores, ou entre uma ou mais variáveis e retornar apenas um valor final.

Existem três tipos de operadores:

- Operadores unários: operam em apenas um valor. Por exemplo, ! (operador de negação) ou o ++ (operador de incremento).
- Operadores binários: grupo que contém a maioria dos operadores que o PHP suporta.
- Operadores ternários: são elementos de controle de fluxo condicional, que ajudam na tomada de decisões e desvio do fluxo geral do sistema.

Operadores em Javascript

- Operadores Aritméticos
- Operadores de Atribuição
- Operadores de String
- Operadores de Incremento/Decremento
- Operadores de Comparação
- Operadores Lógicos
- Operador ternário

Operadores Aritméticos

...

Operadores aritméticos

São os operadores utilizados para realizar operações aritméticas.

$$x = 2y + z$$

Operadores aritméticos

São os operadores utilizados para realizar operações aritméticas.



The diagram shows the arithmetic expression $x = 2y + z$ centered within a light yellow rectangular box. Three red arrows point downwards to the variables in the expression: one arrow points to the variable x on the left, one arrow points to the variable y in the term $2y$, and one arrow points to the variable z on the right.

$$x = 2y + z$$

Operadores aritméticos

aula4.html

```
<!DOCTYPE html>  
<html lang="pt-br">  
<head>  
  <meta charset="utf-8">  
  <title>Aula4</title>  
</head>  
<body>  
  <script src="script4_01.js"></script>  
</body>  
</html>
```

Operadores aritméticos

script4_01.js

```
let x, y ,z;
```

```
y=3;
```

```
z=5;
```

```
x = 2 * y + z;
```

```
document.write("x = ",x,"<br>");
```

```
document.write("x = "+x+"<br>");
```

```
document.write (`x = ${x}<br>`);
```


Operadores aritméticos

São os operadores utilizados para realizar operações aritméticas.

Para todos os exemplos abaixo, consideramos $a=10$ e $b=2$.

Nome	Símbolo	Exemplo	Resultado
Adição	+	$a + b$	12
Subtração	-	$a - b$	8
Multiplicação	*	$a * b$	20
Divisão	/	a / b	5
Módulo (resto da divisão)	%	$a \% b$	0
Exponenciação	**	$a ** b$	100

Operadores aritméticos

script4_02.js

```
let a,b;  
a=prompt("Digite um número");  
a=parseInt(a);  
b=parseInt(prompt("Digite outro número"));  
document.write (`soma: ${a+b}) <br>`);  
document.write ("subtração: "+ (a-b));
```

Operadores de atribuição

São usados para atribuir valores, numéricos ou strings, a uma variável.

O operador de atribuição em Javascript é símbolo de "=" (igual). Isso significa que a variável a esquerda do símbolo recebe o valor de atribuição à direita.

Tarefa	Significado	Descrição
a = b	a = b	a recebe o conteúdo de b
a += b	a = a + b	a passa a receber a soma de a mais b
a -= b	a = a - b	a passa a receber a diferença de a menos b
a *= b	a = a * b	a passa a receber a resultado da multiplicação de a vezes b
a /= b	a = a / b	a passa a receber a quociente de a dividido b
a %= b	a = a % b	a passa a receber o resto da divisão de a dividido por b
a **=b	a = a ** b	a passa a receber \$a elevado a b (exponenciação)

Operadores de atribuição

script4_03.js

```
let a,b;
```

```
a=parseInt(prompt("Digite um número"));
```

```
b=parseInt(prompt("Digite outro número"));
```

```
a+=b;
```

```
b-=5;
```

```
document.write (`a = ${a}<br>`);
```

```
document.write ("b = "+ b);
```

Operador de string

Há dois tipos operadores de strings em Javascript.

O primeiro é o operador de concatenação +

O segundo é o operador de atribuição de concatenação (+=), que acrescenta o argumento do lado direito no argumento do lado

e

Operador	Nome	Exemplo	Resultado
+	Concatenação	a + b	Concatenação de a e b
+=	Atribuição de Concatenação	a += b	A variável a assume, agora, também o conteúdo da variável b

Operadores de string

script4_04.js

```
let a,b;  
a=prompt("Digite o nome");  
b=prompt("Digite o sobrenome");  
document.write (`${a} ${b}`);  
b+=" ";  
b+=a;  
document.write ("<br>" + b);
```

Operadores de Incremento/Decremento

Os operadores de incremento são utilizados para aumentar o valor de uma variável e o decremento, por consequência, diminuir.

Para todos os exemplos abaixo, consideramos $a=10$.

Operador	Nome	Exemplo	Resultado
++	pré-incremento	++a	11
++	pós-incremento	a++	11
--	pré-decremento	--a	9
--	pós-decremento	a--	9

Operadores de Incremento/Decremento (Pré-incremento)

script4_05.js

```
let a;  
a=parseInt(prompt("Digite um número"));  
document.write ("a =" + (++a));  
document.write ("<br>a =" + a);
```


Operadores de Incremento/Decremento (Pós-incremento)

script4_06.js

```
let a;  
a=parseInt(prompt("Digite um número"));  
document.write ("a =" + (a++));  
document.write ("<br>a =" + a);
```

Operadores de Incremento/Decremento (Pré-decremento)

script4_07.js

```
let a;  
a=parseInt(prompt("Digite um número"));  
document.write ("a =" + (--a));  
document.write ("<br>a =" + a);
```

Operadores de Incremento/Decremento (Pós-decremento)

script4_08.js

```
let a;  
a=parseInt(prompt("Digite um número"));  
document.write ("a =" + (a--));  
document.write ("<br>a =" + a);
```

Operadores de comparação (relacional)

Este tipo de operador permite que você compare dois valores ou duas variáveis, sejam números ou strings.

O resultado de uma comparação sempre será **verdadeiro** ou **falso**, **true** ou **false**.

Operadores de comparação (relacional) - a=10 e b=2.

Operador	Nome	Exemplo	Resultado
==	Igual	a == b	false
===	Idêntico	a === b	false Verifica-se além do conteúdo da variável, o seu tipo (<i>string</i> , <i>number</i> , etc).
!=	Não igual	a != b	true
!==	Não idêntico	a !== b	true
>	Maior	a > b	true
<	Menor	a < b	false
>=	Maior ou igual	a >= b	true
<=	Menor ou igual	a <= b	false

Operadores de comparação (relacional)

script4_09.js

```
let a,b;
```

```
a=10;
```

```
b="10";
```

```
document.write("Igualdade = "+(a==b)+"<br>");
```

```
document.write("Idêntico = "+(a===b)+"<br>");
```

```
document.write("Não igual = "+(a!=b)+"<br>");
```

```
document.write("Não Idêntico = "+(a!==b)+"<br>");
```

Operadores de comparação (relacional)

script4_10.js

```
let a,b,c;  
a=10;  
b=10;  
c=11;  
document.write("Menor: "+(a<b)+"<br>");  
document.write("Menor ou igual: "+(a<=b)+"<br>");  
document.write("Maior: "+(c>b)+"<br>");  
document.write("Maior ou igual: "+(c>=b)+"<br>");
```

Operadores lógicos

São operadores utilizados para comparar duas ou mais condições entre variáveis. A saída retornada em um operador lógico sempre será verdadeira ou falso, true ou false.

Para este exemplo, tomemos as variáveis a= 1, b=2 e c=3.

Operador	Nome	Exemplo	Resultado
!	Não	!(a >= b)	true
&&	E	(a < b) && (b < c)	true Para saída ser verdadeira, as duas comparações necessitam ser verdadeiras
	OU	(a < b) (b > c)	true Para saída ser verdadeira, apenas uma comparação necessita ser verdadeira

Operadores lógicos - Não - Not - !

script4_11.js

```
//Operador ! (not)
```

```
let a,b,res;
```

```
a=9;
```

```
b=10;
```

```
res=(a<b);
```

```
document.write(`a = ${a} , b = ${b} <br>`);
```

```
document.write(`a < b : ${a<b} <br>`);
```

```
document.write(`!( a < b ) : ${!(a<b)}<br>`);
```

```
document.write("res : "+typeof(res));
```

Condição	Resultado
true	false
false	true

Operadores lógicos - E - and - &&

```
script4_12.js
```

```
//Operador && (and)
```

```
let a,b,c,d;
```

```
a=50;
```

```
b=120;
```

```
c=200;
```

```
document.write(`a = ${a}, b = ${b}, c = ${c} <br>`);
```

```
d=(a<=b) && (a>=c);
```

```
document.write("(a<=b) && (a>=c) = "+d);
```

Condição 1	Condição 2	Resultado
true	true	true
true	false	false
false	true	false
false	false	false

Operadores lógicos - Ou - or - ||

```
script4_13.js
```

```
//Operador || (or)
```

```
let a,b,c,d;
```

```
a=50;
```

```
b=120;
```

```
c=200;
```

```
document.write(`a = ${a}, b = ${b}, c = ${c} <br>`);
```

```
d=(a<=b) || (a>=c);
```

```
document.write("(a<=b) || (a>=c) = "+d);
```

Condição 1	Condição 2	Resultado
true	true	true
true	false	true
false	true	true
false	false	false

Operador ternário

Três situações de acordo com a fórmula abaixo

condição ? valor se for verdadeiro : valor se for falso

Essa estrutura também é conhecida como if inline, usada para realizar estruturas condicionais no Excel (planilhas)

Operadores ternário

script4_14.js

```
let a,b,c,d,e;  
a=50;  
b=120;  
c=200;  
d=(a<=b) ? "Verdadeiro" : "Falso";  
e=(a>=c) ? "Verdadeiro" : "Falso";  
document.write(`d = ${d}<br>`);  
document.write(`e = ${e}<br>`);
```

Referências

Referências

MORRISON, M. Use a cabeça JavaScript. 5o Ed. Rio de Janeiro: Alta Books, 2012. 606 p.

OLIVIERO, C. A. J. Faça um site JavaScript orientado por projeto. 6o ed. São Paulo: Érica, 2010. 266 p.

ZAKAS, Nicholas C. JavaScript de alto desempenho. 8o Ed. São Paulo: Novatec, 2010. 245 p.