

Addizione (+)^E_d i t

L'operazione di addizione produce la somma di operandi numerici o la concatenazione di stringhe.

Sintassi

Operator: $x + y$

Esempi

```
// Numero + Numero -> addizione
```

```
1 + 2 // 3
```

```
// Booleano + Numero -> addizione
```

```
true + 1 // 2
```

```
// Booleano + Booleano -> addizione
```

```
false + false // 0
```

```
// Numero + Stringa -> concatenazione
```

```
5 + "foo" // "5foo"
```

```
// Stringa + Booleano -> concatenazione
```

```
"foo" + false // "foofalse"
```

```
// Stringa + Stringa -> concatenazione
```

```
"foo" + "bar" // "foobar"
```

Sottrazione (-)^E_d i t

L'operatore di sottrazione fa la sottrazione dei due operandi e produce la loro differenza.

Sintassi

Operator: $x - y$

Esempi

```
5 - 3 // 2
3 - 5 // -2
"foo" - 3 // NaN (not a number)
```

Divisione (/)^E_d i t

L'operatore di divisione produce il quoziente dei suoi operandi dove l'operando di sinistra è il dividendo e l'operando di destra è il divisore.

Sintassi

Operator: x / y

Esempi

```
1 / 2 // ritorna 0.5 in JavaScript
```

```
1.0 / 2.0 // returns 0.5

2.0 / 0 // returns Infinity in JavaScript
2.0 / 0.0 // returns Infinity too
2.0 / -0.0 // returns -Infinity in JavaScript
```

Moltiplicazione (*) E d i t

Syntax

Operator: $x * y$

Examples

```
2 * 2 // 4
-2 * 2 // -4
Infinity * 0 // NaN
Infinity * Infinity // Infinity
"foo" * 2 // NaN
```

Modulo (%) E d i t

Ritorna il modulo aritmetico, ovvero una divisione priva di resto.

Syntax

Operator: var1 \% var2

Examples

```
12 % 5 // 2
-1 % 2 // -1
NaN % 2 // NaN
1 % 2 // 1
2 % 3 // 2
-4 % 2 // -0
```

Incremento (++)^{Edi} t

Aggiunge una unità all'operatore indicato

- usato postfisso (x++) ritorna il valore e poi incrementa.
- Usato prefisso (++x), incrementa e poi ritorna il valore.

Syntax

Operator: x++ or ++x

Examples

```
// Postfix
var x = 3;
y = x++; // y = 3, x = 4

// Prefix
var a = 2;
```

```
b = ++a; // a = 3, b = 3
```

Decremento (--)

Ed
di
t

Sottrae una unità all'operatore indicato

- usato postfisso (x--) ritorna il valore e poi decrementa.
- Usato prefisso (--x), decrementa e poi ritorna il valore.

Syntax

Operator: x-- or --x

Examples

```
// Postfix
var x = 3;
y = x--; // y = 3, x = 2

// Prefix
var a = 2;
b = --a; // a = 1, b = 1
```

Negazione (-)

Ed
i
t

Syntax

Operator: -x

Examples

```
var x = 3;  
y = -x; // y = -3, x = 3
```