Arithmetic Operators in JavaScript

Arithmetic operators in JavaScript perform mathematical operations on numeric values. They include addition, subtraction, multiplication, division, modulus, and exponentiation. Let's explore each operator with examples:

1. Addition (+)

The addition operator (+) adds two values together.

Example 1: Numeric Addition

```
let sum = 5 + 3;
console.log(sum); // Output: 8
```

Example 2: Adding Negative Values

```
let total = -2 + 7;
console.log(total); // Output: 5
```

Example 3: Concatenating Strings

```
let combined = "Hello " + "World";
console.log(combined); // Output: Hello World
```

The addition operator not only adds numeric values but also concatenates strings.

2. Subtraction (-)

The subtraction operator (-) subtracts the right value from the left value.

Example 1: Numeric Subtraction

```
let difference = 10 - 4; // Result: 6
```

Example 2: Subtracting Negative Values

```
let result = 5 - (-3); // Result: 8
```

Example 3: Invalid Subtraction with Strings

```
let invalid = "Hello" - "World"; // Result: NaN (Not a Number)
```

The subtraction operator performs numeric subtraction and results in `NaN` when used with non-numeric values.

3. Multiplication (*)

The multiplication operator (*) multiplies two values.

Example 1: Numeric Multiplication

```
let product = 3 * 4; // Result: 12
```

Example 2: Multiplying by a Negative Value

```
let negativeProduct = (-2) * 6; // Result: -12
```

Example 3: Invalid Multiplication with Strings

```
let invalidProduct = "Five" * 2; // Result: NaN
```

The multiplication operator performs numeric multiplication and results in `NaN` when used with non-numeric values.

4. Division (/)

The division operator (/) divides the left value by the right value.

Example 1: Numeric Division

```
let quotient = 15 / 3; // Result: 5
```

Example 2: Division with Decimal Result

```
let decimalResult = 10 / 4; // Result: 2.5
```

Example 3: Invalid Division with Strings

```
let invalidDivision = "Ten" / 2; // Result: NaN
```

The division operator performs numeric division and results in `NaN` when used with non-numeric values.

5. Modulus (%)

The modulus operator (%) returns the remainder of the division.

Example 1: Modulus with Positive Numbers

```
let remainder = 17 % 5; // Result: 2
```

Example 2: Modulus with Negative Numbers

```
let negativeRemainder = (-8) % 3; // Result: -2
```

Example 3: Invalid Modulus with Strings

```
let invalidModulus = "Twenty" % 7; // Result: NaN
```

The modulus operator calculates the remainder after division and results in `NaN` when used with non-numeric values.

6. Exponentiation (**)

The exponentiation operator (**) raises the left value to the power of the right value.

Example 1: Exponentiation

```
let powerResult = 2 ** 3; // Result: 8
```

Example 2: Fractional Exponentiation

```
let fractionalPower = 4 ** 0.5; // Result: 2
```

Example 3: Invalid Exponentiation with Strings

```
let invalidExponentiation = "Three" ** 2; // Result: NaN
```

The exponentiation operator performs exponentiation and results in `NaN` when used with non-numeric values.

Why Arthemtic Operators Matter ?

Arithmetic operators are essential for performing mathematical operations on variables, which is a fundamental part of programming.
They allow us to manipulate data and create complex calculations.

These examples illustrate the functionality of each arithmetic operator, providing a detailed understanding of their use cases.