JavaScript Arithmetic

- description : JavaScript Arithmetic
- author : 오션
- email : shlim@repia.com
- lastupdate : 2021-04-20

Source of the article

JavaScript Arithmetic

JavaScript Arithmetic Operators

산술 연산자(Arithmetic Operators)는 숫자(리터럴 또는 변수)에 대한 산술을 수행하는 데 사용됩니다.

Operator	Description
+	adddtion(더하기)
-	subtraction(빼기)
*	Multiplication(곱하기)
**	Exponentiation(거듭제곱)
/	Division(나누기)
%	Modulus(Division Remainder) 나눗셈의 나머지
++	Invcrement (증가)
--	Decrement (감소)

덧셈 할당 연산자(+=)는 변수에 값을 더합니다.

Arithmetic Operations

일반적인 산술 연산은 두 숫자에 대해 작용합니다.

두 숫자는 리터럴이 될 수 있습니다.

예제

```
var x = 100 + 50;
```

또는 두 숫자는 변수가 될 수 있습니다.

예제

```
var x = a + b;
```

또는 두 숫자는 표현식(expressions)이 될 수 있습니다.

예제

```
var x = (100 + 50) * a;
```

Operators and Operands

산술 연산에서 숫자를 피연산자(operands)라고 부릅니다.

두 피연산자 사이에서 실행될 연산은 연산자에 의해 정의됩니다.

Operand (피연산자)	**Operator (연산자)**	**Operand (피연산자)**
100	+	50

Adding

더하기(addition) 연산자 (+)는 숫자를 더합니다.

```
var x = 5;
var y = 2;
var z = x + y;
```

Subtracting

빼기(subtraction) 연산자 (-)는 숫자를 뺍니다.

```
var x = 5;
var y = 2;
var z = x - y;
```

Multiplying

곱셈(Multiplication) 연산자 (*)는 숫자를 곱합니다.

```
var x = 5;
var y = 2;
var z = x * y;
```

Dividing

나눗셈(division) 연산자 (/)는 숫자를 나눕니다.

```
var x = 5;
var y = 2;
var z = x / y;
```

Remainder

나머지(modulus) 연산자 (%)는 나눗셈의 나머지를 반환합니다.

```
var x  = 5;
var y = 2;
var z = x % y;
```

산술에서, 두 정수를 나누면 **몫(quotient)**과 **나머지(remainder)**가 생성됩니다.

수학에서 모듈로 연산의 결과는 산술 분할의 나머지입니다.

Incrementing

증가(increment) 연산자(++)는 숫자를 증가시킵니다.

```
var x = 5;
x++;
var z = z;
```

Decrementing

감소(decrement)는 숫자를 감소시킵니다.

```
var x = 5;
x--;
var z = x;
```

Exponentiation

거듭제곱(Exponentiation) 연산자 (**)는 첫 번째 피연산자를 두 번째 피연산자로 제곱합니다.

```
var x = 5;
var z = x **2;     // result is 25,  z = 5 **2 = 5 X 5
```

x ** y는 Math.pow(x, y)와 동일한 결과를 산출합니다.

```
\\
var x = 5;
var z = Math.pow(x, 2);   // result is 25
```

Operator Precedence

연산자 우선 순위는 산술 표현식에서 연산이 수행되는 순서를 설명합니다.

```
var x = 100 + 50 * 3;     // x = 250
```

위 예제의 결과가 150 * 3과 같습니까? 아니면 100 + 150과 같습니까?

더하기 또는 곱하기가 먼저 실행됩니까?

전통적인 학교 수학에서 처럼, 곱셈이 먼저 이루어집니다.

곱하기 (*) 및 나누기 (/)는 더하기 (+) 및 빼기 (-)보다 우선 순위가 높습니다.

그리고 (학교 수학처럼) 괄호를 사용하여 우선 순위를 변경할 수 있습니다.

```
var x = (100 + 50) * 3;
```

괄호를 사용할 때, 괄호 안의 연산이 먼저 계산됩니다.

많은 연산이 동일한 우선 순위 (예 : 더하기 및 빼기)를 갖는 경우, 왼쪽에서 오른쪽으로 계산됩니다.

```
var x = (100 + 50) - 3;
```

JavaScript Operator Precedence Values

오션,, Javascript, Arithmetic

From:
http://125.132.25.164/dokuwiki/ - **문제를 잘 정의하는 것은 문제를 절반 해결한 것이다. - 2023.12**

Permanent link:
http://125.132.25.164/dokuwiki/doku.php?id=wiki:javascript:javascript_note:js_arithmetic&rev=1618922357

Last update: **2022/03/10 19:52**