Topic/Title: Basic Arithmetic and the Modulo Operator in JavaScript

Increment and Decrement Expressions

Keywords/Questions:

Addition(+) +=

Subtraction(-) -=

Multiplication(*) *=

Division(/) /=

Modulo(%) %=

Pre/Post-Increment ++x x++

Pre/Post Decrement --x x--

BODMAS

parseInt();

//

Notes:

Addition:- made by using the + sign. `var a = 2 + 3; //5`

Subtraction: made by using the - sign. `var b = 10 - 2;`

Multiplication:- made by using the * sign. `var c = 3 * 3;`

Division:- made by the / sign. `var d = 6 / 2;`

Modulo:- made by the % sign. Used to compute reminder. `var e = 9 % 6; //3`

//:- used for commenting out.

JavaScript operations follow the BODMAS rule.

parseInt(string):- used to convert a String to an Integer.

-For example, parseInt("1") converts the string "1" to an Integer.

```
var dogAge=prompt("What is the age of your dog?");
var humanAge=4*(dogAge-2)+21;
alert(humanAge);
```

-For example, if we input 2 in the prompt, the alert will be 21.

Increment Expression:- ++ `var x = 5; x++; //6` -equivalent to x=x+1

Decrement Expression:- -- `var x = 5; x-- ; //4` -equivalent to x=x-1

+= :used to increase the value of our variable. `var x = 5; x += 2 ; //7`

-= :used to decrease the value of our variable.

```
var x = 5;
var y = 3;
x += y ; //8
```

+=

-=

*=

/=

Summary:

JavaScript Operations:-Addition(+) -Subtraction(-) -Multiplication(*) -Division(/) -Modulo(%)

-Post-Increment Expression:x++ Post-Decrement Expression:x--

-Pre-Increment Expression:++x Pre-Decrement Expression:--x

JavaScript Operations follow the BODMAS rule.

x+=5, x-=5, x/=5, x*=5 :-can be used instead of x=x+5, x=x-5, x=x*5, x=x/5 respectively.

parseInt(String); :-converts a string to an integer.

// :-used to comment out.

*=:- applied similarly to +=.

/=:-applied similarly to +=.

What does `y` equal?

```
1  var x = 3;
2  var y = x++;
3  y += 1;
```

Incorrect answer. Please try again.

In this line: var y = x++ the value of x is assigned to y before x is incremented, so y equals 3 on line 2, while x equals 4. There fore on line 3, y now equals 4 instead of 5.

Pre-increment/decrement --> The current value of the variable is used, ***before*** the increment/decrement

```
1  var y = --x + z;  // x is decremented by 1, then the result of the expression
2                    // "x + z" (using this new value of x) is assigned to y.
```

Post-increment/decrement --> The value ***after*** the increment/decrement operation is used.

```
1  var y = x++ + z;  // the expression "x + z" is evaluated (using the current
2                    // value of x) and assigned to y, then x is incremented.
```