Luis Mejia
CIS-14A-48590
Written Assignment 7

Ch 7 Written Assignment

You can submit this in a single word file, PDF, or ZIP archive.

Evaluate. Show your work where the evaluation takes more than one step.

1. **var addit = 6 + “5”;**

   var addit = “6” + “5”;

   Here, because we are adding a number to a string, number 6 gets concatenated to string “5”. Essentially, number 6 gets converted to a string and then concatenated.

   Evaluating to: var addit = “65”;

2. **var addit = “6” + “5”;**

   Here, because both “6” and “5” are strings, they simply get concatenated when using the + operator.

   Evaluating to: var addit = “65”;

3. **var x=5; x += "2";**

   Here we basically have x = x + “2”;

   If x = 5; Then what we are really doing is assigning a new value to x using its old value.

   So, we can say that: x = 5 + “2”;

   x = “5” + “2”;   5 gets converted to a string and finally concatenated with the + operator.

   Evaluating to: x = “52”;

4. **1 == “1”;**

   Here, because we are comparing a number to a string, the string gets converted to a number (if possible), and then compared.

   So, what we really get is 1 == 1; which evaluates to true.

Luis Mejia
CIS-14A-48590
Written Assignment 7

5. **1 === “1”;**

Here, the === operator is the strict equality operator or the identity operator which means both values being compared must be the same type and the same value to evaluate to true.

Therefore, number 1 is not equal to string “1”. This expression evaluates to false.

6. **“42” === “42”;**

Here we are using the strict equality operator again. Because both operands have the same type and value, this expression will evaluate to true.

7. **42 !== “42”;**

!== is the strict version of != just like === is the strict version of ==. Here, !== is the opposite of ===.

Because number 42 is of different type when compared to string “42”, this expression will evaluate to true.

8. **c = Math.sqrt(-9);**

In regular mathematics, the square root of -9 will result in an imaginary number. However, we cannot represent imaginary numbers in computer science, so when attempting to call the .sqrt() method on the Math object with an argument of -9, c = Math.sqrt(-9); will evaluate to c = NaN; because the square root of -9 in computer science results to NaN (Not a number) since imaginary numbers cannot be represented in computer science.

9. **var sub = “1” - “1”;**

Here, because we are using the - operator on two strings, JavaScript will attempt to convert both operands to numbers giving us: var sub = 1 - 1; evaluating to var sub = 0;

Luis Mejia
CIS-14A-48590
Written Assignment 7

10. **if ([ ]) { //code }; //assume that there is code in the braces and the array is defined.**

If assuming that there is code in the braces and that an array is defined, the expression in the if statement header will evaluate to true and the code in the braces will execute.

11. **if (0) { //code }; //assume that there is code in the braces.**

If assuming that there is code in the braces, the expression in the if statement header will evaluate to false and the code in the braces will not execute. This is because 0 is considered false in JavaScript and if an expression in an if statement header evaluates to false, the code following the statement will not execute.

12. **8675309 == "";**

Here, we are comparing a number to a string so JavaScript will attempt to convert the string to a number. Because an empty string is considered false our empty string should convert to false/0, making this expression evaluate to false.

8675309 == false/0; Evaluating to false

13. **false == (5==="5");**

Here we must evaluate the expression within the parentheses first. Because we are using the strict equality operator in the parentheses, number 5 is not of the same type as string "5" making the expression in the parentheses evaluate to false giving us: false == false; which finally will evaluate to true.