Student’s Name

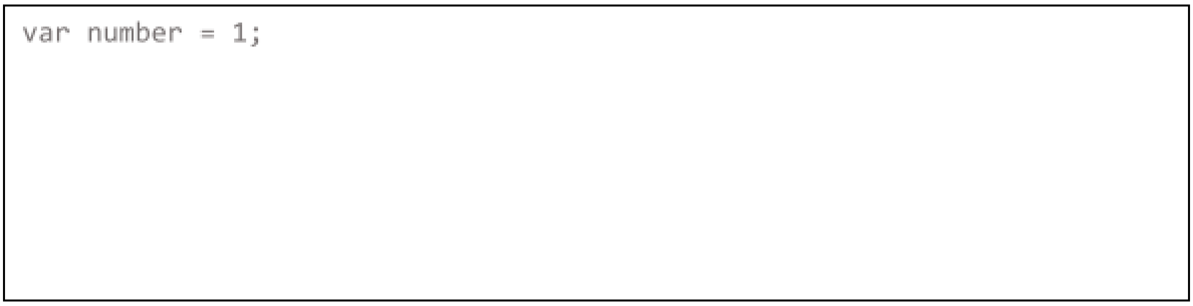
Instructor’s Name

Course Title and Number

Date

The Difference between “=”, “==” and “===” in Javascript

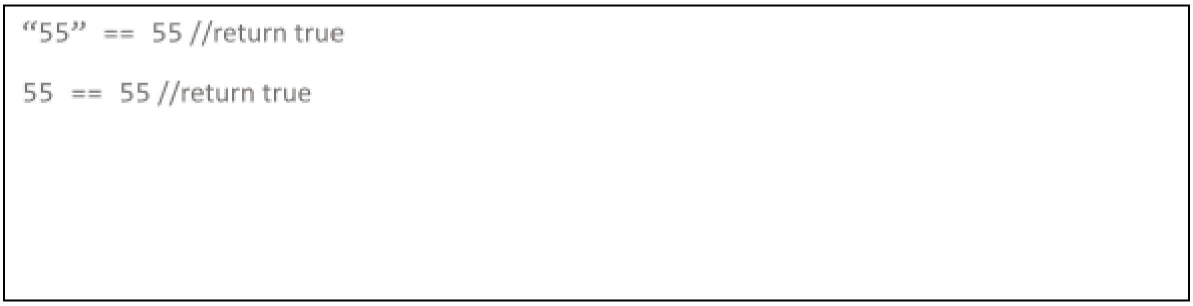
The equal sign in Javascript carries different meanings when it comes in the form of 1, 2, or 3 copies of itself. The single equal (=) is an assignment operator, which sets the value to a variable from right to left (Hartman, 2023). Example:



```
var number = 1;
```

The `number` is the variable, while `1` is the value assigned to it.

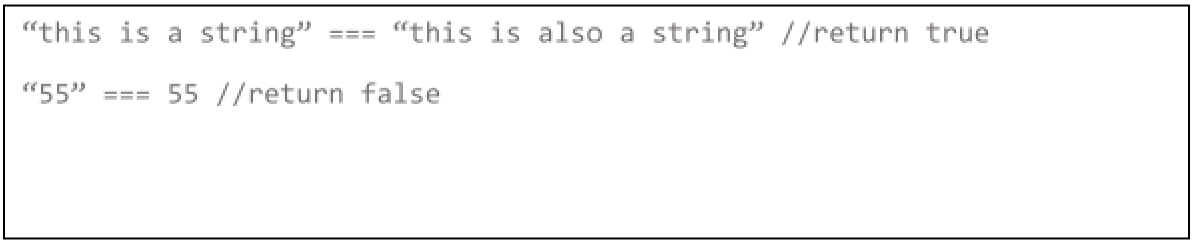
The double equals sign (==) is a comparison operator. This is used to perform a logic test to check if the value is equal to another one (Castiglione, 2021). Javascript doesn’t care if the data type compared between 2 values is different; it will convert it to another type and do a comparison. For example, a String would always get converted into an integer:



```
“55” == 55 //return true

55 == 55 //return true
```

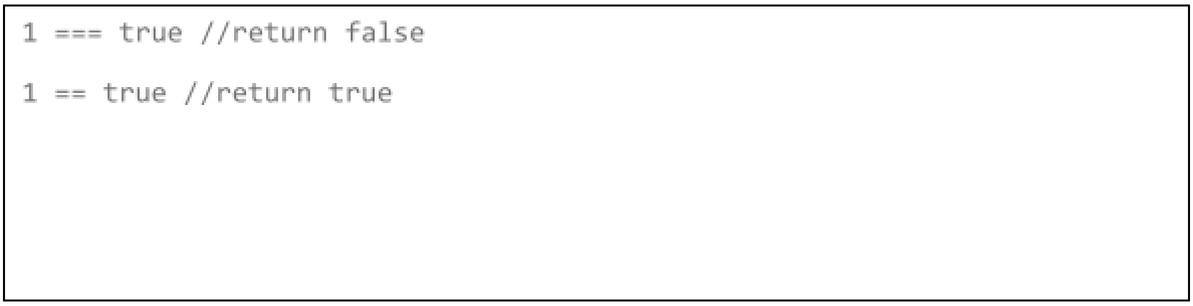
The triple equals sign (===) is a strict comparison operator that checks against the value and the data type. This means that if the value compared is a different data type, it will return false. Example:



```
“this is a string” === “this is also a string” //return true

“55” === 55 //return false
```

Here are some facts about the double and triple equals comparison operators. See the example below:



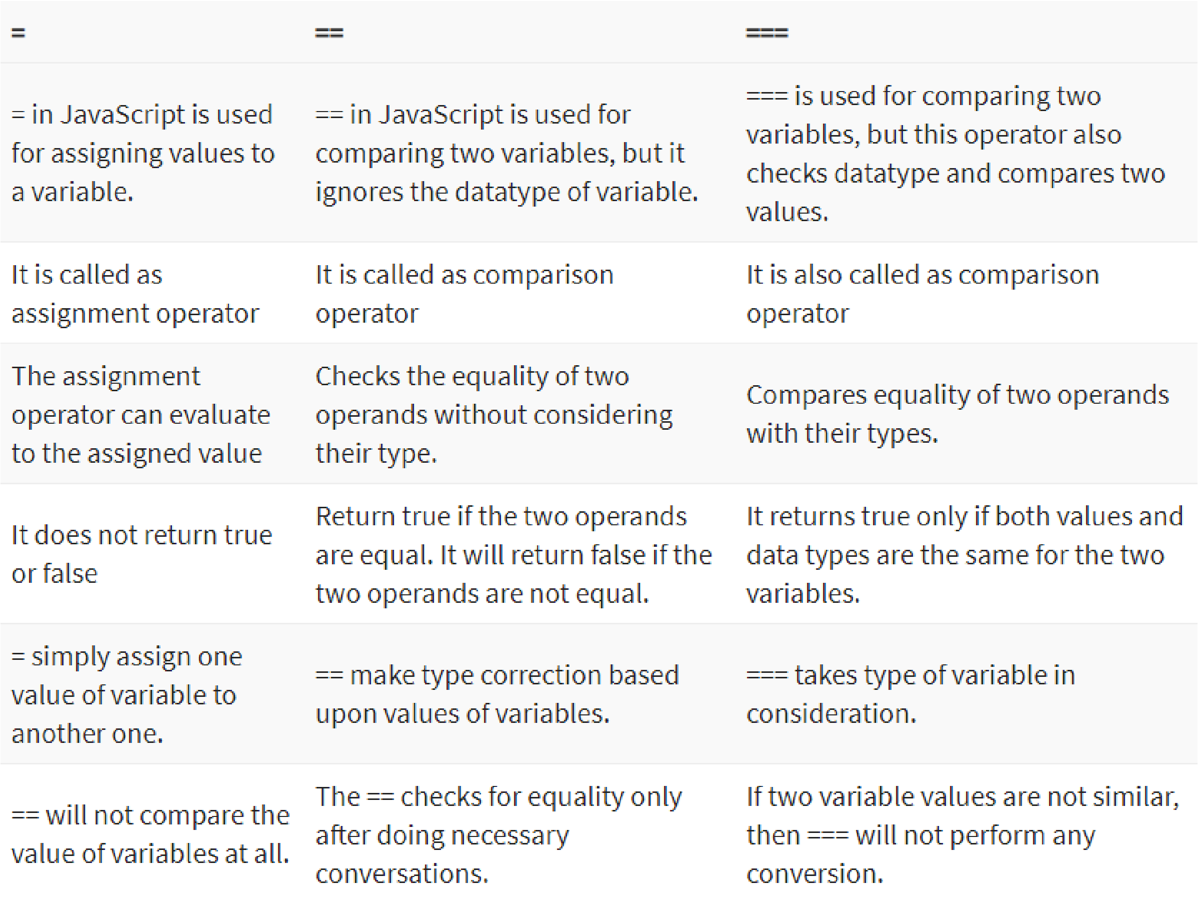
```
1 === true //return false

1 == true //return true
```

“True” is a Boolean type, so when a strict comparison is applied, it will return “false” because 1 is an integer. On the other hand, if double equals is applied, it will return “true” because Javascript will convert the boolean “true” into 1.

So, in most cases, programmers tend to use the strict comparison operator because it will prevent tons of potential bugs, unless the use case requires a normal comparison operator.

In summary, here is a table for a clear comparison between the equal signs.



=	==	===
= in JavaScript is used for assigning values to a variable.	== in JavaScript is used for comparing two variables, but it ignores the datatype of variable.	=== is used for comparing two variables, but this operator also checks datatype and compares two values.
It is called as assignment operator	It is called as comparison operator	It is also called as comparison operator
The assignment operator can evaluate to the assigned value	Checks the equality of two operands without considering their type.	Compares equality of two operands with their types.
It does not return true or false	Return true if the two operands are equal. It will return false if the two operands are not equal.	It returns true only if both values and data types are the same for the two variables.
= simply assign one value of variable to another one.	== make type correction based upon values of variables.	=== takes type of variable in consideration.
== will not compare the value of variables at all.	The == checks for equality only after doing necessary conversations.	If two variable values are not similar, then === will not perform any conversion.

Figure 1: Screenshot taken from

https://www.guru99.com/difference-equality-strict-operator-javascript.html

References

Castiglione, C. (2021, April 30). *The Difference Between == and === In JavaScript*. https://learn.onemonth.com/difference-between-equal-signs-javascript/

Hartman, J. (2023, May 13). *Difference Between =, ==, and === in JavaScript [Examples]*. https://www.guru99.com/difference-equality-strict-operator-javascript.html

Kapoor, S. (2020, March 12). *JavaScript Triple Equals Sign VS Double Equals Sign – Comparison Operators Explained with Examples.* https://www.freecodecamp.org/news/javascript-triple-equals-sign-vs-double-equals-sign-comparison-operators-explained-with-examples/