

## Lesson 02 Demo 01

### Differentiating Between Equal to and Strict Equality Operator

**Objective:** To demonstrate the usage of the comparison operators `==` and `===` in JavaScript

**Tools Required:** Visual Studio Code and Node.js

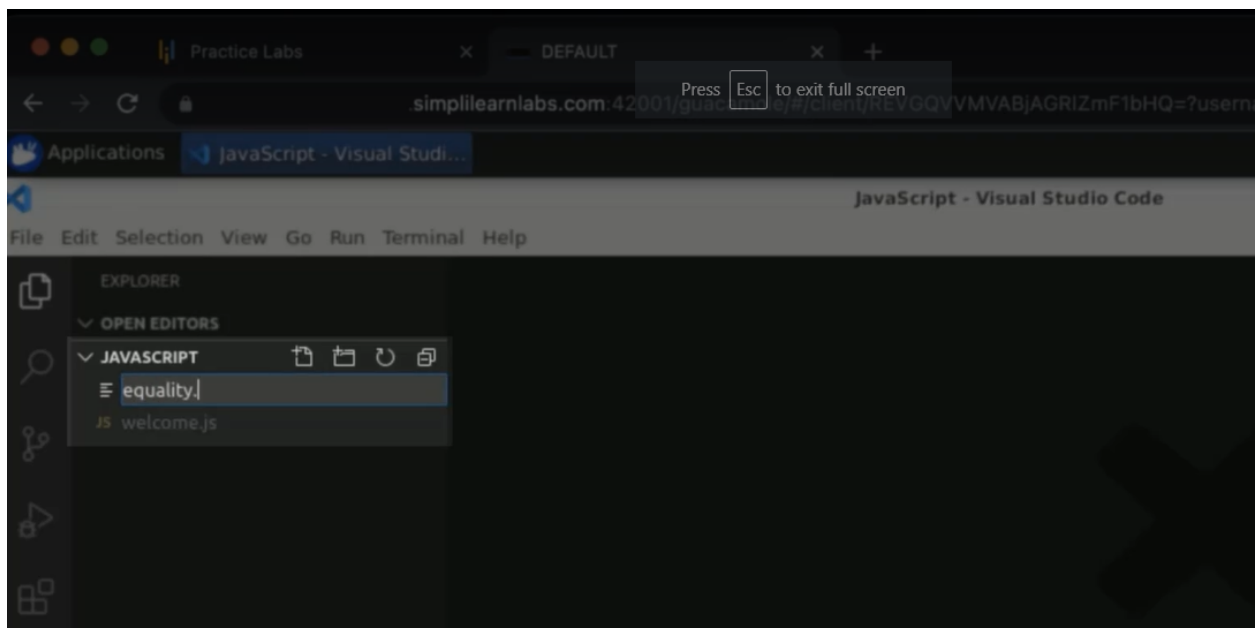
**Prerequisites:** None

#### Steps to be followed:

1. Distinguish between `==` and `===`

#### Step 1: Distinguish between `==` and `===`

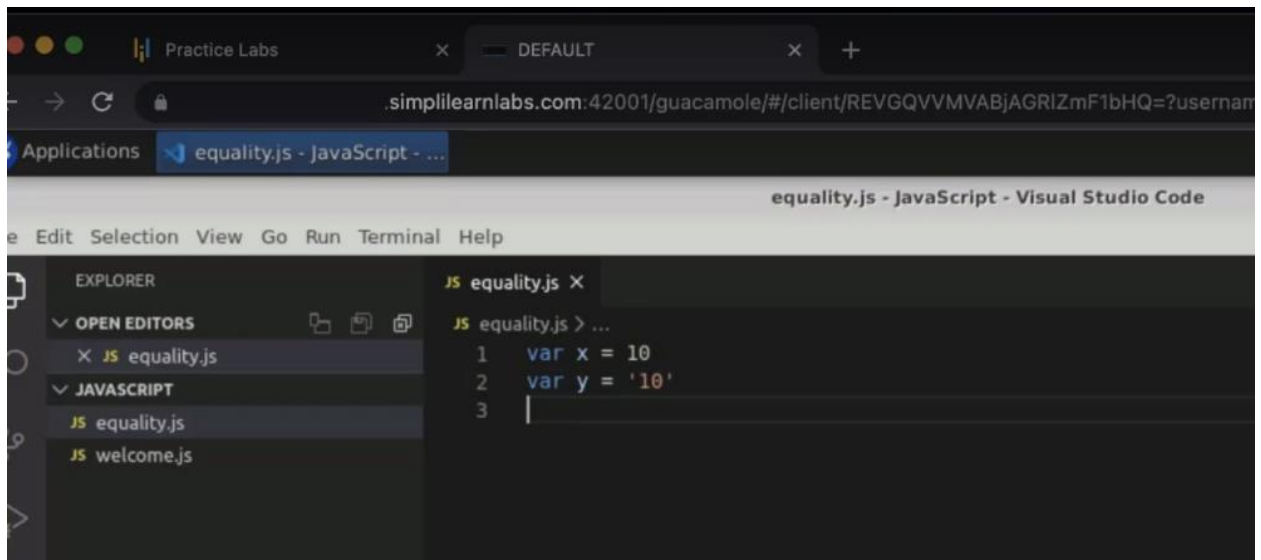
- 1.1 Open Visual Studio Code and create a new file named **equality.js**



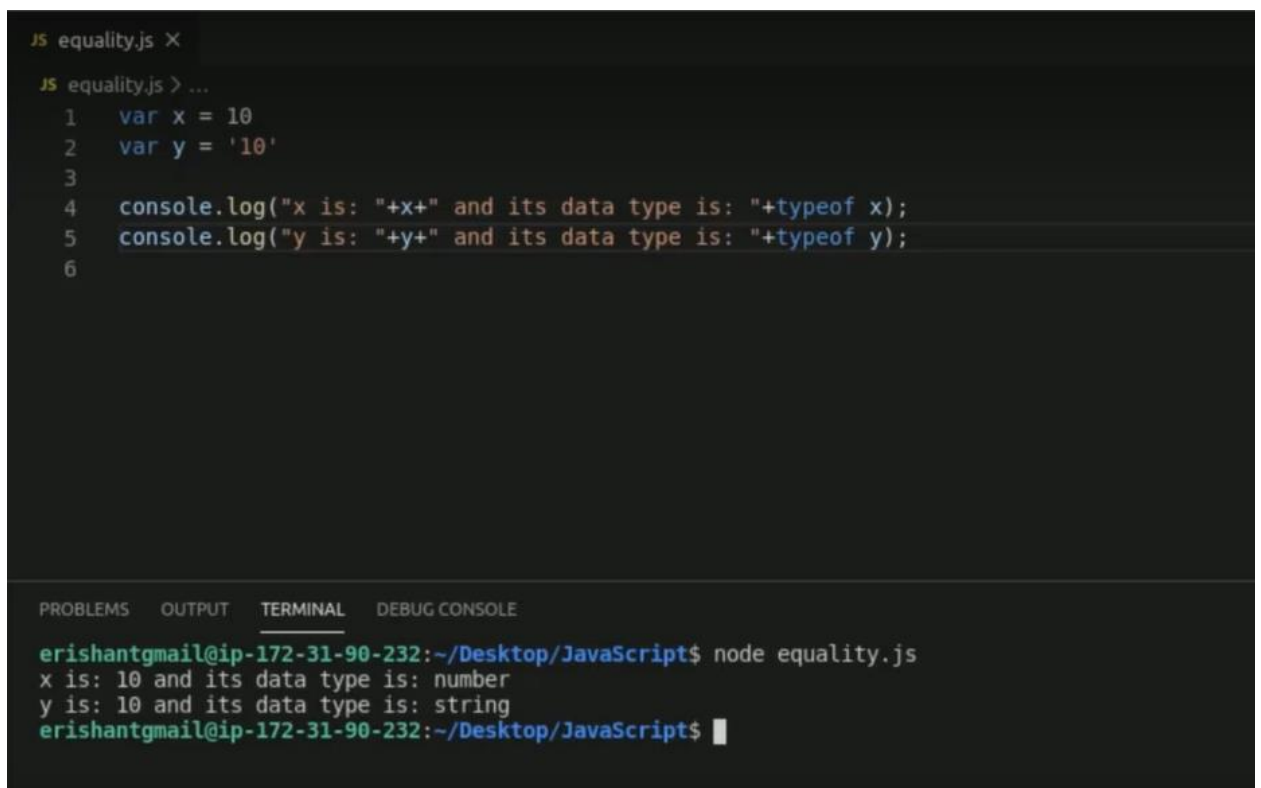
1.2 In the **equality.js** file, declare variables **x** and **y** with the value **10**:

```
var x = 10
```

```
var y = '10'
```



1.3 Log the values of **x** and **y** to verify their contents

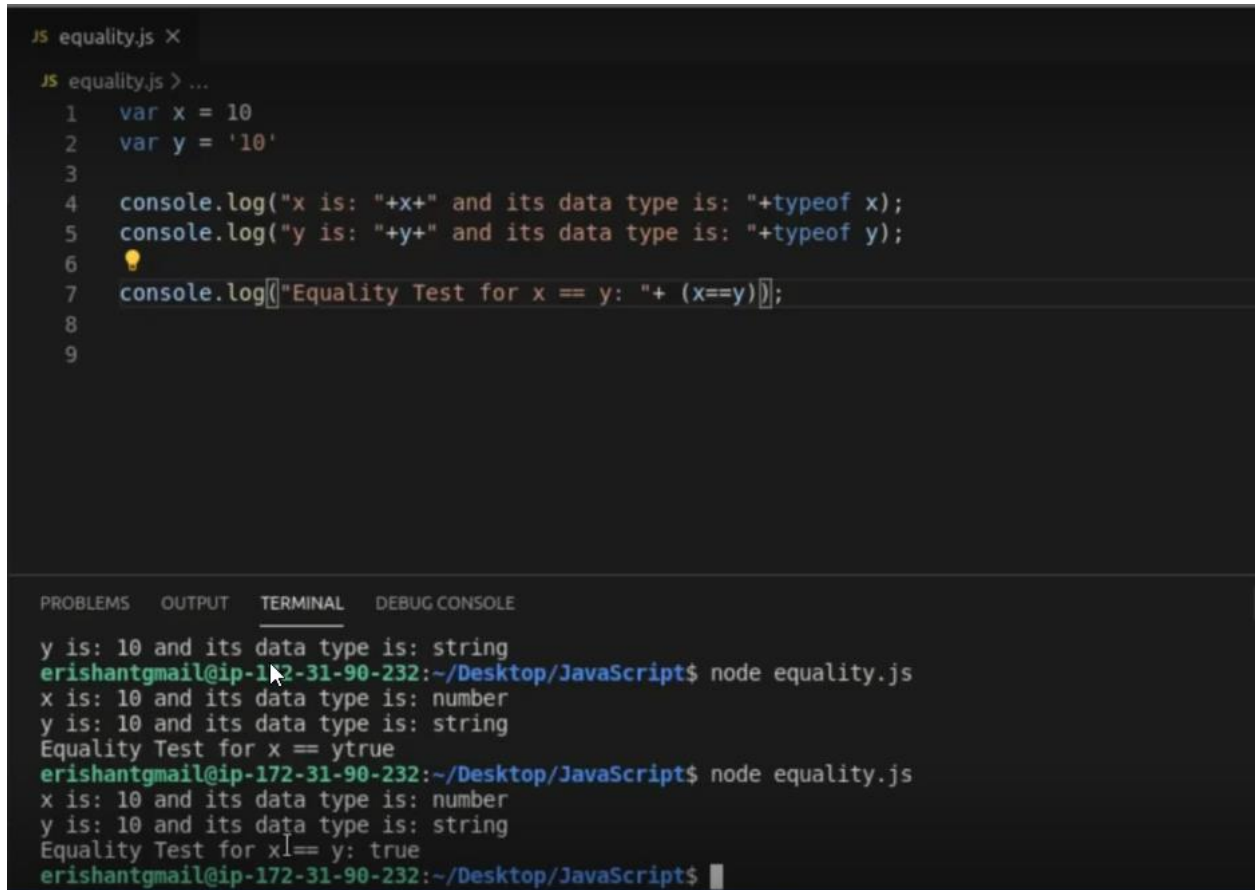


1.4 Use the **console.log()** function to compare **x** and **y** using the **==** operator

```
JS equality.js ×  
JS equality.js > ...  
1  var x = 10  
2  var y = '10'  
3  
4  console.log("x is: "+x+" and its data type is: "+typeof x);  
5  console.log("y is: "+y+" and its data type is: "+typeof y);  
6    
7  console.log("Equality Test for x == y: "+ (x==y));  
8  
9
```

1.5 Run the JavaScript program again using Node.js:

**node equality.js**



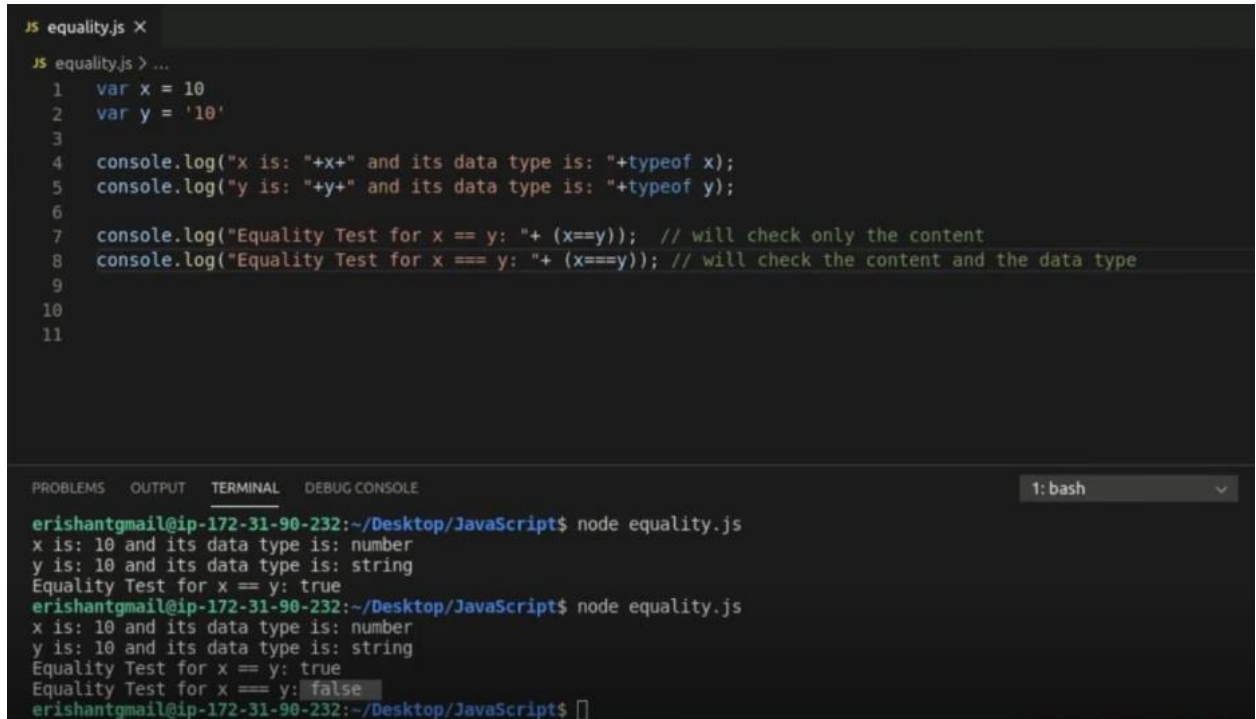
The screenshot shows a code editor with a file named `equality.js`. The code defines two variables, `x` and `y`, both with the value 10. `x` is a number and `y` is a string. The code then logs the value and data type of each variable. Finally, it performs an equality test `x == y` and logs the result. Below the code editor, the terminal output shows the results of running the program twice. In the first run, the equality test returns `true`. In the second run, the equality test returns `false` because the data types are different.

```
JS equality.js X
JS equality.js > ...
1  var x = 10
2  var y = '10'
3
4  console.log("x is: "+x+" and its data type is: "+typeof x);
5  console.log("y is: "+y+" and its data type is: "+typeof y);
6  ⚡
7  console.log("Equality Test for x == y: "+ (x==y));
8
9

PROBLEMS  OUTPUT  TERMINAL  DEBUG CONSOLE
y is: 10 and its data type is: string
erishantgmail@ip-172-31-90-232:~/Desktop/JavaScript$ node equality.js
x is: 10 and its data type is: number
y is: 10 and its data type is: string
Equality Test for x == y:true
erishantgmail@ip-172-31-90-232:~/Desktop/JavaScript$ node equality.js
x is: 10 and its data type is: number
y is: 10 and its data type is: string
Equality Test for x== y: false
erishantgmail@ip-172-31-90-232:~/Desktop/JavaScript$
```

Observe the output on the console. If `x` and `y` have the same value and data type, it will print **true**. Otherwise, it will print **false**.

- 1.6 Use the **console.log()** function to compare **x** and **y** using the triple equal to operator.  
Rerun the program



The image shows a code editor window titled 'equality.js' with the following code:

```
1 var x = 10
2 var y = '10'
3
4 console.log("x is: "+x+" and its data type is: "+typeof x);
5 console.log("y is: "+y+" and its data type is: "+typeof y);
6
7 console.log("Equality Test for x == y: "+ (x==y)); // will check only the content
8 console.log("Equality Test for x === y: "+ (x===y)); // will check the content and the data type
9
10
11
```

Below the code editor is a terminal window showing the output of running the script twice:

```
erishantgmail@ip-172-31-90-232:~/Desktop/JavaScript$ node equality.js
x is: 10 and its data type is: number
y is: 10 and its data type is: string
Equality Test for x == y: true
erishantgmail@ip-172-31-90-232:~/Desktop/JavaScript$ node equality.js
x is: 10 and its data type is: number
y is: 10 and its data type is: string
Equality Test for x == y: true
Equality Test for x === y: false
erishantgmail@ip-172-31-90-232:~/Desktop/JavaScript$
```

Analyze the outputs of the two comparison operations

**Note:** The **==** operator compares only the content, while the **===** operator checks both the content and the data type.

By following the above steps, you have successfully demonstrated the usage of equal to and strict equality operators in JavaScript.