Lesson 04 Demo 03

Differentiating between Value and Reference While Creating Storage Containers

Objective: To understand the difference between value and reference while creating storage containers in JavaScript

Tools Required: Visual Studio Code and Node.js

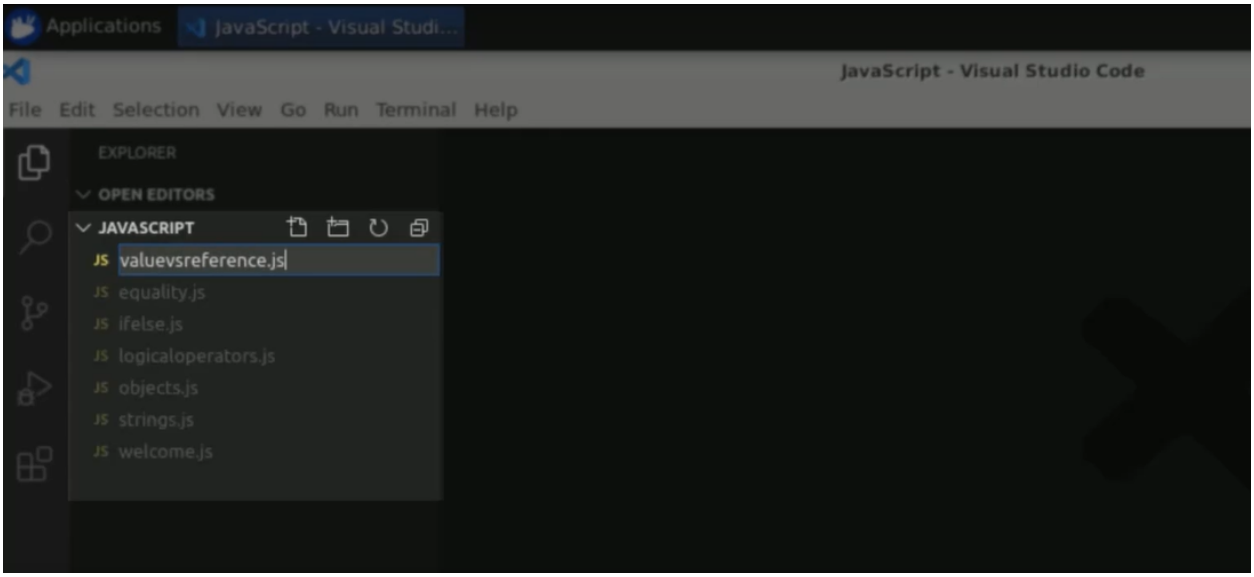
Prerequisites: None

Steps to be followed:

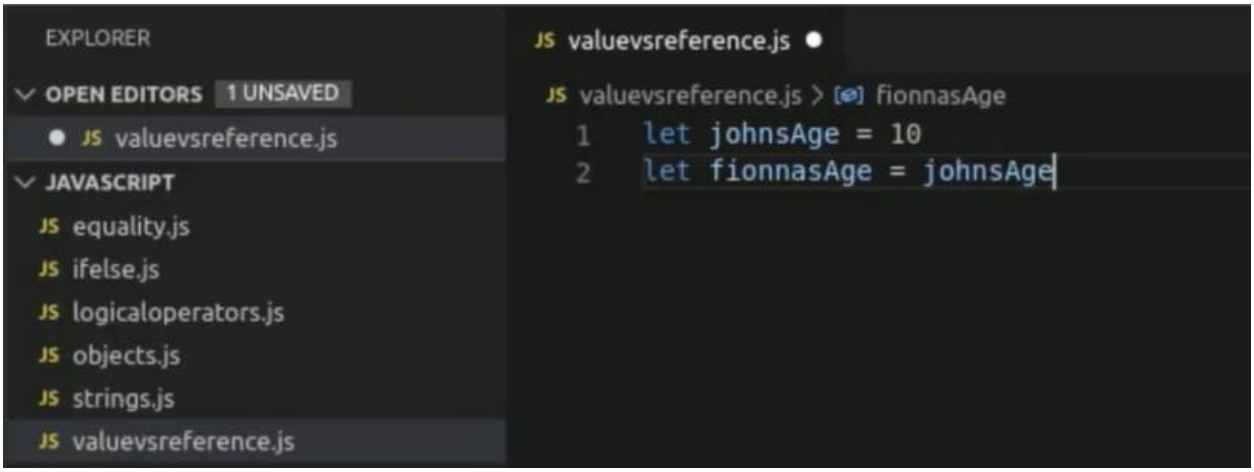
1. Analyze the concept of value versus reference

Step 1: Analyze the concept of value versus reference

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



1.2 Define two variables: **johnsAge** with a value of **10** and **fionnasAge** as a copy of John's age



1.3 Log the values of both variables to the console

1.4 Run the program and observe that both variables have the same value

```
JS valuevsreference.js ×
JS valuevsreference.js > ...
1   let johnsAge = 10
2   let fionnasAge = johnsAge
3
4   console.log("johnsAge: "+johnsAge);
5   console.log("fionnasAge: "+fionnasAge);
6
7
```

```
PROBLEMS   OUTPUT   TERMINAL   DEBUG CONSOLE
erishantgmail@ip-172-31-90-232:~/Desktop/JavaScript$ node valuevsreference.js
johnsAge: 10
fionnasAge: 10
erishantgmail@ip-172-31-90-232:~/Desktop/JavaScript$
```

1.5 Update the variable **johnsAge** to the new value. Then, log the values of John's and Fionna's ages again

```
JS valuevsreference.js ×
JS valuevsreference.js > ...
1   let johnsAge = 10
2   let fionnasAge = johnsAge
3
4   console.log("johnsAge: "+johnsAge);
5   console.log("fionnasAge: "+fionnasAge);
6
7   johnsAge = 12;
8
9   console.log("johnsAge now is: "+johnsAge);
10  console.log("fionnasAge now is: "+fionnasAge);
11
```

1.6 Run the program and note that only John's age is modified, while Fiona's age remains unchanged

```
JS valuevsreference.js ×
JS valuevsreference.js > ...
1   let johnsAge = 10
2   let fionnasAge = johnsAge
3
4   console.log("johnsAge: "+johnsAge);
5   console.log("fionnasAge: "+fionnasAge);
6
7   johnsAge = 12;
8
9   console.log("johnsAge now is: "+johnsAge);
10  console.log("fionnasAge now is: "+fionnasAge);
11

PROBLEMS   OUTPUT   TERMINAL   DEBUG CONSOLE

erishantgmail@ip-172-31-90-232:~/Desktop/JavaScript$ node valuevsreference.js
johnsAge: 10
fionnasAge: 10
erishantgmail@ip-172-31-90-232:~/Desktop/JavaScript$ node valuevsreference.js
johnsAge: 10
fionnasAge: 10
johnsAge now is: 12
fionnasAge now is: 10
erishantgmail@ip-172-31-90-232:~/Desktop/JavaScript$
```

Understand that copying by value means the data is duplicated into a new storage container, resulting in independent variables.

1.7 Create an array called **ages** with values of **10**, **20**, **30**, **40**, and **50**

```
JS valuevsreference.js ●
JS valuevsreference.js > ages
1   let johnsAge = 10
2   let fionnasAge = johnsAge // Copy Operation -> By Value
3
4   console.log("johnsAge: "+johnsAge);
5   console.log("fionnasAge: "+fionnasAge);
6
7   johnsAge = 12;
8
9   console.log("johnsAge now is: "+johnsAge);
10  console.log("fionnasAge now is: "+fionnasAge);
11
12  let ages = [10, 20, 30, 40, 50]
13
```

1.8 Copy the array into a new variable named **groupAges**

```
JS valuevsreference.js ●
JS valuevsreference.js > ...
1   let johnsAge = 10
2   let fionnasAge = johnsAge // Copy Operation -> By Value
3
4   console.log("johnsAge: "+johnsAge);
5   console.log("fionnasAge: "+fionnasAge);
6
7   johnsAge = 12;
8
9   console.log("johnsAge now is: "+johnsAge);
10  console.log("fionnasAge now is: "+fionnasAge);
11
12  let ages = [10, 20, 30, 40, 50]
13  let groupAges = ages;
14
```

1.9 Modify the value at a specific index in **groupAges** and log the **ages** array

```
JS valuevsreference.js ×
JS valuevsreference.js > ...
7   johnsAge = 12;
8
9   console.log("johnsAge now is: "+johnsAge);
10  console.log("fionnasAge now is: "+fionnasAge);
11
12  let ages = [10, 20, 30, 40, 50]
13  let groupAges = ages;   // Copy Operation -> By Reference
14
15  groupAges[1] = 22
16  groupAges[3] = 45
17
18  console.log("Ages: "+ages);
19
20
```

1.10 Run the program and observe that the changes made in **groupAges** also affect the **ages** array

```
JS valuevsreference.js ×
JS valuevsreference.js > ...
 7   johnsAge = 12;
 8
 9   console.log("johnsAge now is: "+johnsAge);
10   console.log("fionnasAge now is: "+fionnasAge);
11
12   let ages = [10, 20, 30, 40, 50]
13   let groupAges = ages;   // Copy Operation -> By Reference
14
15   groupAges[1] = 22
16   groupAges[3] = 45
17
18   console.log("Ages: "+ages);
19
20
```

```
PROBLEMS   OUTPUT   TERMINAL   DEBUG CONSOLE

fionnasAge: 10
johnsAge now is: 12
fionnasAge now is: 10
erishantgmail@ip-172-31-90-232:~/Desktop/JavaScript$ node valuevsreference.js
johnsAge: 10
fionnasAge: 10
johnsAge now is: 12
fionnasAge now is: 10
Ages: 10,22,30,45,50
erishantgmail@ip-172-31-90-232:~/Desktop/JavaScript$
```

Note: Copying by reference means creating a new reference to the same storage container, allowing changes to be reflected in both variables.

By following these steps, you have successfully demonstrated the difference between value and reference when creating storage containers in JavaScript, which is essential for managing data effectively and avoiding common pitfalls in your code.