2) **copy by value and copy by reference**

The terms “**copy by value**” and “**copy by reference**” are used to describe how variables are **passed** on. I simple words , **copy by value** means the actual **value** is **passed** on. **copy by reference** means a number (called an address) is **passed** on which defines where the **value** is stored.

Copy by value is applicable to **primitive** data types(number, string,Boolean)

Copy by reference is applicable to **composite** data type (Array & objects)

**Copy by value**

Eg - var a = 5;

var b = a;

a = 10;

console.log(a,b);

***result -- > 10 ,5***

**Copy by Reference**

**Eg –** var arr1 = [1,2,3];

var arr2 = arr1;

arr2[0] =10;

console.log(arr1,arr2);

***result 🡪 [10,2,3] ,[10,2,3]***

3) How to Copy by value a composite data type .

We can do by using **concat()** method**:** Create a new array variable and then concatenate the older one in the new array.

|  |
| --- |
| <script>    const players = ['Sam', 'Sarah', 'Ryan', 'Poppy'];    const playersCopy2 = [].concat(players);    playersCopy2[2]='hell';    document.write(players, playersCopy2);  </script>  Output –  ["Sam", "Sarah", "Ryan", "Poppy"]  ["Sam", "Sarah", "hell", "Poppy"] |

Also there are other ways to do it , below are the listed ones

* **By using slice() method:**
* **By using ES6 Spread operator**
* **By using Array.from():**