**Difference between copy by value and copy by reference**

* **Copy/Pass by Value** means that a copy of the original is created. In JavaScript both trivial and primitive data types such as (string, number, Boolean) which are the primitive data types and (null and undefined) which are the trivial data types are passed, or copied by value. This means that these datatypes are immutable or the existing value cannot be altered. These values are not shared between multiple variables, even if you set the variable equal to the other. Every variable representing a primitive or trivial value belongs to a unique memory location. None of the variables will ever point to the same memory location.
* **Copy/Pass by Reference** means creating an ALIAS to the original. On the other hand, in JavaScript, composite data types such as objects and arrays are passed, or copied by reference and point to a location in the memory for the value, not the value itself. When a value is copied, both variable will point to the same address. This means that if we reassign the value of a variable after it is copied to another variable the other variable will also be reassigned as both of them point to the same address.