Ganesh Kulkarni

GUVI Zen Class Dec – 20 Batch

**Question 1 :**

**Index.html file :**

**<!DOCTYPE html>**

**<html lang="en">**

**<head>**

**<meta charset="UTF-8">**

**<meta name="viewport" content="width=device-width, initial-scale=1.0">**

**<title>Document</title>**

**</head>**

**<body>**

**<script src = "script.js"> </script>**

**</body>**

**</html>**

**Script.js:**

**// step 1 create a variable for XMLHttpRequest**

**var request = new XMLHttpRequest();**

**request.open('GET', 'https://restcountries.eu/rest/v2/all', true);**

**request.send();**

**request.onload = function(){**

**var countryData = JSON.parse(this.response);**

**for(var i in countryData){**

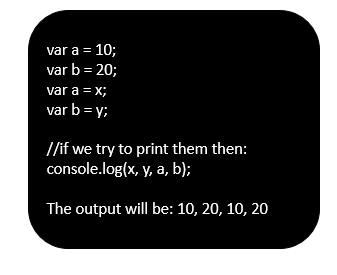
**console.log(countryData[i].name);**

**}**

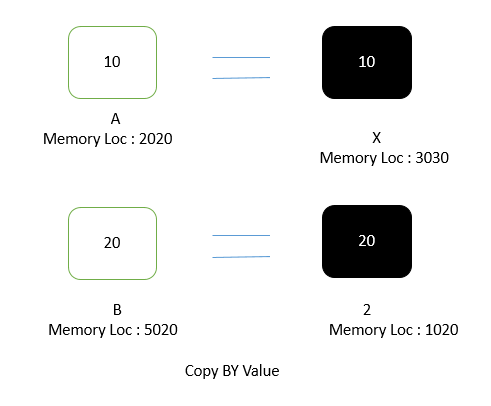
**}**

**Question 2: Difference between Copy by value and Copy by reference?**

* In copy by value, separate memory is allocated to each variable. On the other hand, in the copy by reference, the variable's reference is shared and that memory address is pointed by another variable.
* In JavaScript, copy by value is primarily used for primitive data types these are: Boolean, null, undefined, String, and Number. At the same time, copy by reference is used for the Array, Function, and Object. These 3 are technically called objects because they are stored as objects only in javascript behind the seen.
* If we assigned the values to the primitive data such as:



* We get the output as 10,20,10, 20 by printing separate x,y,a and b. This is happening because every time when we copy the primitive data type using ‘=’ sign then new memory location is created for every variable.



* In the memory it will create 4 different memory locations as given in above figure for a,b,x and y.
* Copy by reference is primarily used for non primitive data types where the variable actually consist of reference of the memory location which consist of the actual value. When we assign the value, for example:

var arr = [10, 20, 30,40]; // having memory location as 2020

var refArr = arr; //now the refArr will point to the memory address as 2020.

* Copy by reference has an advantage over copy by value is that, copy by reference saves memory as other variables directly points to the memory locations.

**Question 3: How to Copy by value a composite data type (array+object)?**

* Generally when we copy the data in composite data type then the new variable simply refers to the memory location where the first one variable is located. In simple words it only shares the memory location between two composite data types.
* But still we can achieve copy by value in composite data type such as array and objects using several ways.
* There are three ways by which we can achieve the copy by value in composite data type such as:

1. Using the spread (...) operator

* It is represented using … operator.
* Ex: var a = [10,20,30];
* var c = […a];
* // c will get 10,20,30 values without directly pointing to memory location.

1. Using the Object.assign() method

* For ex: var a = [10,20,30];
* var c = Object.assign([], a);
* // c will get 10,20,30 values without directly pointing to memory location.

1. Using the JSON.stringify() and JSON.parse() methods

* For ex: var a = [10,20,30];
* var c = JSON.parse(JSON.stringify(a));
* // c will get 10,20,30 values without directly pointing to memory location.
* In this three ways we can achieve Copy by value in composite data types.