**1.** Load the rest countries' data using your HTML and script.js file and run a for loop on the data and print all the country names in the console.

https://amazing-lamarr-eb0828.netlify.app/

https://github.com/JayMaru20/Full-stack/blob/main/index.html

**2. Give a write-up on the Difference between copy by value and copy by reference.**

**COPY-BY-VALUE :**

1. We have 5 data types that are passed by value ,they are :

(Boolean , null , undefined , string and number).

1. These are called as primitive types . when we assign these variables to other variables using = , we copy the value to new variable. They are copied by value.

**Example1:**

Var x = 10;

Var y = ‘abc’;

Var a =x;

Var b=y;

Console.log(x,y,a,b);

Output : 10, ‘abc’,10,’abc’.

Copying value to another variable will not have any relationships between them .

**Example 2:** var a =10;

Var b=10;

b=20;

output : 10 , 20 .

Here value of b has changed to updated value .

**COPY-BY-REFERENCE**

1. We have 3 data types that are passed by reference

(array , function and object).

1. Variables that are assigned a non-primitive value are given a reference to that value . That reference points to the object’s location in memory.
2. When a reference type value , an object is copied to another variable using = , the address of that value is copied over as primitive. Objects are copied by reference instead of by value.

Var x = {dog:”barks”}

Var y =x;

Console.log(y.dog) //output : barks

Now , I add object to y

y.legs = 4;

console.log(x) //output : dog : barks

legs : 4

Here if I add or remove any object from reference variable , it affects the first variable too .

since they are referring to same addresses.

**3.How to copy by value a composite datatype (array + objects) .**

There are ways to do copying for composite dataTypes through copy by value .

**Using Spread(…) operator :**

var arr1 = [10,20,30]

Var arr2 = [];

arr2 = […arr1];

console.log(arr2); //Output : [10,20,30]

**Using Object.assign() method :**

var arr1=[10,20,30]

Var arr2 = [];

Object.assign(arr2,arr1);

Console.log(arr2); //Output [10,20,30]

**Using array.map() :** This method will call provided callback function once for each element in an array , in order and contructs a new array from results .

**Example :** Var a=[2,3,5,7]

Var b = a.map(c => c)

Console.log(a,b) //Output : [2,3,5,7] [2,3,5,7]

b[1] = 10;

console.log(a,b) //output : [2,3,5,7] [2,10,5,7]

**Using JSON.stringify() and JSON.parse() method :**

Parse() is used for parsing data that was received as JSON(java script object notation)

Stringify() is used to create a JSON string out of an object array .

**Example :**

Var xyz = [1,2,3];

Var b = JSON.parse(JSON.Stringify(xyz));

Console.log(xyz , b) //output : [1,2,3] [1,2,3]