### **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.**

var request = new XMLHttpRequest();

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

request.send();

request.onload = function() {

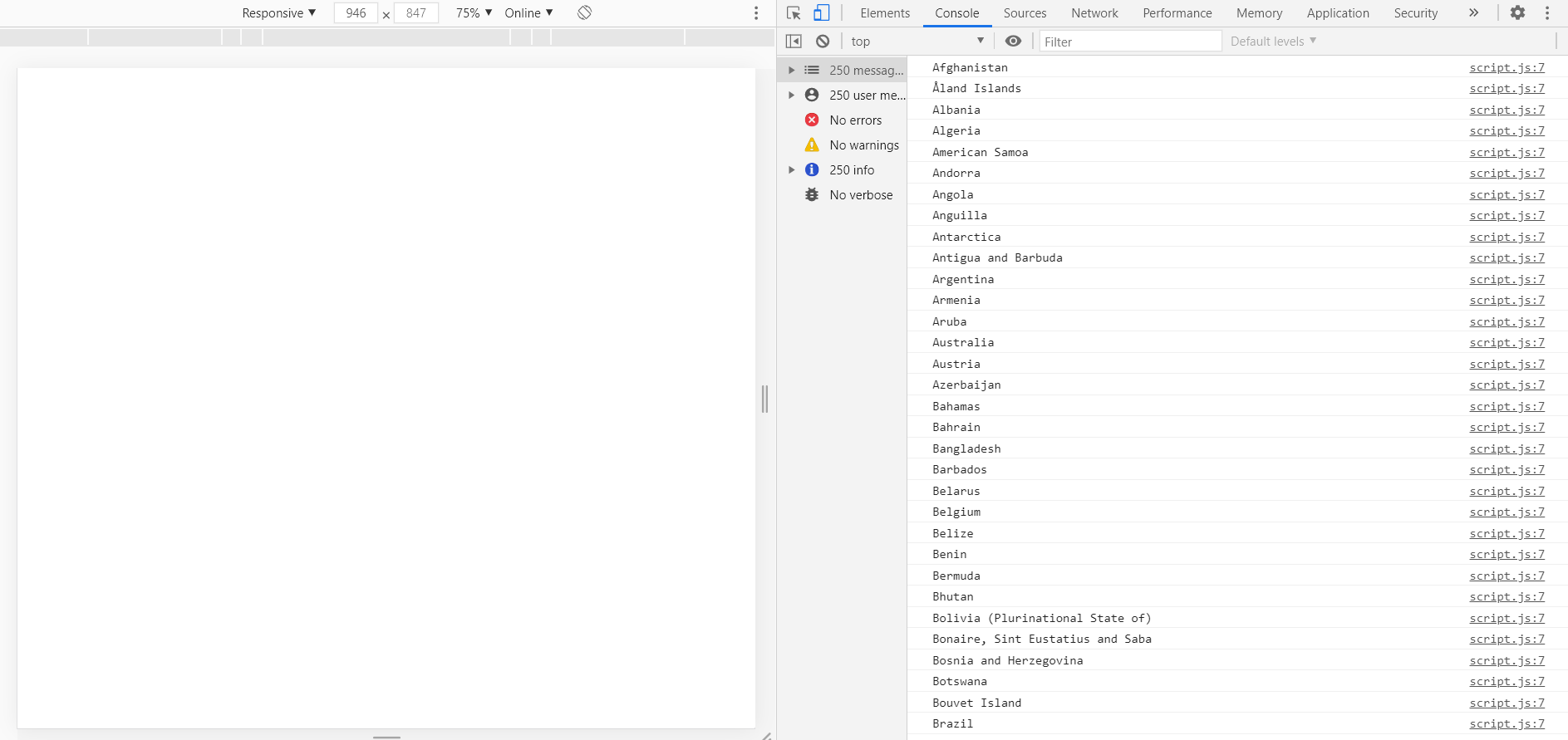
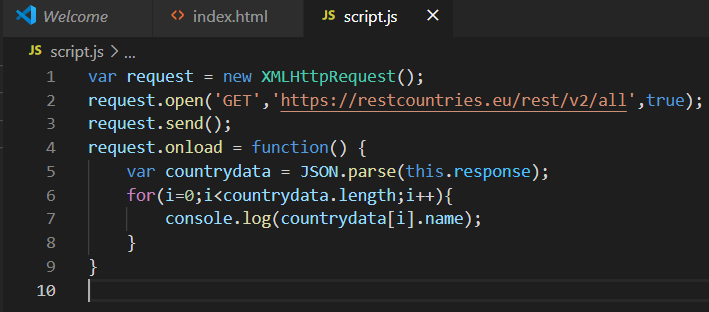
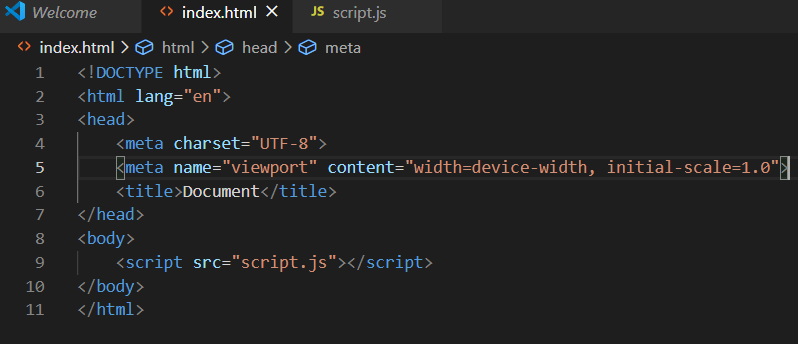
var countrydata = JSON.parse(this.response);

for(i=0;i<countrydata.length;i++){

console.log(countrydata[i].name);

}

}



### 

### 

### **Write a write up on Difference between copy by value and copy by reference.**

JavaScript has five primitive data types:Boolean ,null, undefined, String and Number.Primitive data types are copied by value and arrays and objects that are collectively known as objects or compound data types are copied by reference.

Copy by value vs copy by reference

If a variable of primitive type is assigned to another variable, a copy is made.Changes made on the original variable do not reflect on the copy.

For example,

var x = 10;

var y = x;

x++;

console.log(x,y);

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.The variables don't actually contain the value.They have the same address.

### **How to copy by value a composite data type.**

There are three ways to copy by value for composite data types:

1.Using the spread (...) operator.

2.Using the Object.assign() method.

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

1.Using spread (...) operator.

It allows an iterable to expand in places where more than 0 arguments are expected.This allows shallow copy of objects.

Example:

var a = [1,2,3];

var c = [...a];

C[2] = 5;

console.log(a,c);

2.Using Object.assign() method.

The Object.assign() method copies all enumerable properties from one or more source objects to a target object.

Example:

var b = Object.assign([],a);

b[2] = 8;

console.log(a,b);

3.Using JSON.parse() and JSON.stringify() methods.

These methods are used for nested objects and results in deep copy.