1. How to compare two JSON have the same properties without order?
   1. var obj1 = { name: "Person 1", age:5 };
   2. var obj2 = { age:5, name: "Person 1" };

var obj1 = { name: "Person 1", age:5 };

var obj2 = { age:5, name: "Person 1" };

var flag=true;

if(Object.keys(obj1).length==Object.keys(obj2).length){

    for(key in obj1) {

        if(obj1[key] == obj2[key]) {

            continue;

        }

        else {

            flag=false;

            break;

        }

    }

}

else {

    flag=false;

}

console.log("is object equal"+flag);

output:

is object equaltrue

2. Use the rest countries API url -><https://restcountries.eu/rest/v2/all> and display all the country flags in console

var request=new XMLHttpRequest();

request.open('GET','https://restcountries.com/v3.1/all',true);

request.send();

request.onload=function(){

    var data=JSON.parse(request.response);

    console.log(data);

    var args = [];

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

   console.log(data[i].flags);

}

}

Output:

(250) [{…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, {…}, …]

script.js:11 {png: 'https://upload.wikimedia.org/wikipedia/commons/thu…the\_Taliban.svg/320px-Flag\_of\_the\_Taliban.svg.png', svg: 'https://upload.wikimedia.org/wikipedia/commons/5/5c/Flag\_of\_the\_Taliban.svg'}

script.js:11 {png: 'https://flagcdn.com/w320/sd.png', svg: 'https://flagcdn.com/sd.svg'}

script.js:11 {png: 'https://flagcdn.com/w320/bi.png', svg: 'https://flagcdn.com/bi.svg'}

script.js:11 {png: 'https://flagcdn.com/w320/mx.png', svg: 'https://flagcdn.com/mx.svg'}

script.js:11 {png: 'https://flagcdn.com/w320/cu.png', svg: 'https://flagcdn.com/cu.svg'}

script.js:11 {png: 'https://flagcdn.com/w320/rs.png', svg: 'https://flagcdn.com/rs.svg'}

script.js:11 {png: 'https://flagcdn.com/w320/mc.png', svg: 'https://flagcdn.com/mc.svg'}

3.Use the same rest countries and print all countries name, region, sub region and population

var request=new XMLHttpRequest();

request.open('GET','https://restcountries.com/v3.1/all',true);

request.send();

request.onload=function(){

    var data=JSON.parse(request.response);

    console.log(data);

    var args = [];

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

   console.log(data[i].name,data[i].region,data[i].subregion,data[i].population);

}

}

Output:

{common: 'Afghanistan', official: 'Islamic Republic of Afghanistan', nativeName: {…}} 'Asia' 'Southern Asia' 2837743

script.js:11 {common: 'Sudan', official: 'Republic of the Sudan', nativeName: {…}} 'Africa' 'Northern Africa' 43849269

script.js:11 {common: 'Burundi', official: 'Republic of Burundi', nativeName: {…}} 'Africa' 'Eastern Africa' 11890781

script.js:11 {common: 'Mexico', official: 'United Mexican States', nativeName: {…}} 'Americas' 'North America' 128932753

script.js:11 {common: 'Cuba', official: 'Republic of Cuba', nativeName: {…}} 'Americas' 'Caribbean' 11326616

script.js:11 {common: 'Serbia', official: 'Republic of Serbia', nativeName: {…}} 'Europe' 'Southeast Europe' 6908224

script.js:11 {common: 'Monaco', official: 'Principality of Monaco', nativeName: {…}} 'Europe' 'Western Europe' 39244

script.js:11 {common: 'Bhutan', official: 'Kingdom of Bhutan', nativeName: {…}} 'Asia' 'Southern Asia' 771612

script.js:11 {common: 'Guyana', official: 'Co-operative Republic of Guyana', nativeName: {…}} 'Americas' 'South America' 786559

script.js:11 {common: 'South Georgia', official: 'South Georgia and the South Sandwich Islands', nativeName: {…}} 'Antarctic' undefined 30

script.js:11 {common: 'Bosnia and Herzegovina', official: 'Bosnia and Herzegovina', nativeName: {…}} 'Europe' 'Southeast Europe' 3280815