

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

Ans:

Code:

```
var obj1 = { name: "Person 1", age: 5 };
var obj2 = { age: 5, name: "Person 1" };
let arr1 = [], arr2 = [];
let c = 0, ans = '';

for (let prop in obj1) {
    arr1.push(prop)
};
for (let prop in obj2) {
    arr2.push(prop)
};

for (let i = 0; i < arr1.length; i++) {
    if (arr1[i] !== arr2[i]) {
        ++c
    }
}

c === 0 ? ans = "same" : ans = "not same";

console.log(`The given objects properties are ${ans}`)

console.log(arr1.sort(), arr2.sort())
```

Console:

```
The given objects properties are same      app.js:379
▶ (2) ['age', 'name'] ▶ (2) ['age', 'name']  app.js:381
Live reload enabled.                       (index):56
```

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

Ans:

Code:

```
script.js > onload
1  const req = new XMLHttpRequest();
2
3  req.open("GET", 'https://raw.githubusercontent.com/rvsp/
  restcountries-json-data/master/res-countries.json');
4
5  req.send();
6
7  req.onload = function () {
8      var data = JSON.parse(req.response);
9
10     data.forEach(val => {
11
12         console.log(`
13 Country:${val.name},
14 Flag:${val.flag}`)
15
16     });
17 }
18 }
```

Console:

The scroll down for the console image.

Live reload enabled.	(index):41
Country:Afghanistan, Flag: <a href="https://restcountries.eu/data/afg.svg">https://restcountries.eu/data/afg.svg</a>	<a href="#">script.js:12</a>
Country:Åland Islands, Flag: <a href="https://restcountries.eu/data/ala.svg">https://restcountries.eu/data/ala.svg</a>	<a href="#">script.js:12</a>
Country:Albania, Flag: <a href="https://restcountries.eu/data/alb.svg">https://restcountries.eu/data/alb.svg</a>	<a href="#">script.js:12</a>
Country:Algeria, Flag: <a href="https://restcountries.eu/data/dza.svg">https://restcountries.eu/data/dza.svg</a>	<a href="#">script.js:12</a>
Country:American Samoa, Flag: <a href="https://restcountries.eu/data/asm.svg">https://restcountries.eu/data/asm.svg</a>	<a href="#">script.js:12</a>
Country:Andorra, Flag: <a href="https://restcountries.eu/data/and.svg">https://restcountries.eu/data/and.svg</a>	<a href="#">script.js:12</a>
Country:Angola, Flag: <a href="https://restcountries.eu/data/ago.svg">https://restcountries.eu/data/ago.svg</a>	<a href="#">script.js:12</a>
Country:Anguilla, Flag: <a href="https://restcountries.eu/data/aia.svg">https://restcountries.eu/data/aia.svg</a>	<a href="#">script.js:12</a>
Country:Antarctica, Flag: <a href="https://restcountries.eu/data/ata.svg">https://restcountries.eu/data/ata.svg</a>	<a href="#">script.js:12</a>
Country:Antigua and Barbuda, Flag: <a href="https://restcountries.eu/data/atg.svg">https://restcountries.eu/data/atg.svg</a>	<a href="#">script.js:12</a>
Country:Argentina, Flag: <a href="https://restcountries.eu/data/arg.svg">https://restcountries.eu/data/arg.svg</a>	<a href="#">script.js:12</a>
Country:Armenia, Flag: <a href="https://restcountries.eu/data/arm.svg">https://restcountries.eu/data/arm.svg</a>	<a href="#">script.js:12</a>
Country:Aruba, Flag: <a href="https://restcountries.eu/data/abw.svg">https://restcountries.eu/data/abw.svg</a>	<a href="#">script.js:12</a>
Country:Australia, Flag: <a href="https://restcountries.eu/data/aus.svg">https://restcountries.eu/data/aus.svg</a>	<a href="#">script.js:12</a>
Country:Austria, Flag: <a href="https://restcountries.eu/data/aut.svg">https://restcountries.eu/data/aut.svg</a>	<a href="#">script.js:12</a>
Country:Azerbaijan, Flag: <a href="https://restcountries.eu/data/aze.svg">https://restcountries.eu/data/aze.svg</a>	<a href="#">script.js:12</a>
Country:Bahamas, Flag: <a href="https://restcountries.eu/data/bhs.svg">https://restcountries.eu/data/bhs.svg</a>	<a href="#">script.js:12</a>

Scroll down for the next question.

3. Use the same rest countries and print all country's name, region, sub-region, and populations.

Ans:

Code:

```
script.js > onload > data.forEach() callback
1  const req = new XMLHttpRequest();
2
3  req.open("GET", 'https://raw.githubusercontent.com/rvsp/
  restcountries-json-data/master/res-countries.json');
4
5  req.send();
6
7  req.onload = function () {
8      var data = JSON.parse(req.response);
9
10     data.forEach(val => {
11
12         console.log(`
13             Country    : ${val.name}
14             Continent  : ${val.region}
15             Sub-region : ${val.subregion}
16             Population : ${val.population}
17         `)
18     });
19 }
20
```

Console:

The scroll down for the console image.

Country : Afghanistan Continent : Asia Sub-region : Southern Asia Population : 27657145	<a href="#">script.js:12</a>
Country : Åland Islands Continent : Europe Sub-region : Northern Europe Population : 28875	<a href="#">script.js:12</a>
Country : Albania Continent : Europe Sub-region : Southern Europe Population : 2886026	<a href="#">script.js:12</a>
Country : Algeria Continent : Africa Sub-region : Northern Africa Population : 40400000	<a href="#">script.js:12</a>
Country : American Samoa Continent : Oceania Sub-region : Polynesia Population : 57100	<a href="#">script.js:12</a>
Country : Andorra Continent : Europe Sub-region : Southern Europe Population : 78014	<a href="#">script.js:12</a>
Country : Angola Continent : Africa Sub-region : Middle Africa Population : 25868000	<a href="#">script.js:12</a>
Country : Anguilla Continent : Americas Sub-region : Caribbean Population : 13452	<a href="#">script.js:12</a>
Country : Antarctica Continent : Polar Sub-region : Population : 1000	<a href="#">script.js:12</a>

Thank you!!!

Have a good day!!!