Get all the countries from Asia continent /region using Filter function

Code :

```javascript
1   var connect = new XMLHttpRequest();
2   connect.open('GET',"https://restcountries.eu/rest/v2/all",true);
3   connect.send();
4   connect.onload = function()
5   {
6     var data = JSON.parse(this.response);
7     var res = data.filter((i,p)=>
8     {
9       if(i.region=="Asia")
10      return true ;
11      else
12      return false;
13    })
14
15    var name = res.map((i)=>i.name);
16    console.log(name);
17  };
```

Output :

```
▼ (50) […]
    0: "Afghanistan"
    1: "Armenia"
    2: "Azerbaijan"
    3: "Bahrain"
    4: "Bangladesh"
    5: "Bhutan"
    6: "Brunei Darussalam"
    7: "Cambodia"
    8: "China"
    9: "Georgia"
   10: "Hong Kong"
   11: "India"
   12: "Indonesia"
   13: "Iran (Islamic Republic of)"
   14: "Iraq"
   15: "Israel"
   16: "Japan"
   17: "Jordan"
   18: "Kazakhstan"
   19: "Kuwait"
   20: "Kyrgyzstan"
   21: "Lao People's Democratic Republic"
   22: "Lebanon"
   23: "Macao"
   24: "Malaysia"
   25: "Maldives"
   26: "Mongolia"
   27: "Myanmar"
   28: "Nepal"
   29: "Korea (Democratic People's Republic of)"
   30: "Oman"
   31: "Pakistan"
   32: "Palestine, State of"
   33: "Philippines"
   34: "Qatar"
```

Get all the countries with population of less than 2 lacs using Filter function

Code :

```
1    var connect = new XMLHttpRequest();
2    connect.open('GET',"https://restcountries.eu/rest/v2/all",true);
3    connect.send();
4    connect.onload = function()
5    {
6        var data = JSON.parse(this.response);
7        // console.log(data);
8        var res = data.filter((i,p)=>
9        {
10           if(i.population<=200000)
11           return true ;
12           else
13           return false;
14       })
15
16       var name = res.map((i)=>i.name);
17       console.log(name);
18   };
```

Output :

▶ Array(63) ["Åland Islands", "American Samoa", "Andorra", "Anguilla", "Antarctica", "Antigua and Barbuda", "Aruba", "Bermuda", "Bonaire, Sint Eustatius and Saba", "Bouvet Island", …]

Print the following details name, capital, flag using forEach function

Code :

```js
JS script.js > ...
 1    var connect = new XMLHttpRequest();
 2    connect.open('GET',"https://restcountries.eu/rest/v2/all",true);
 3    connect.send();
 4  ∨ connect.onload = function()
 5    {
 6       var data = JSON.parse(this.response);
 7      // console.log(data);
 8  ∨    var res = data.forEach(i => {
 9          console.log(i.name,i.capital,i.flag);
10       })
11    };
```

Output:

Afghanistan Kabul *https://restcountries.eu/data/afg.svg*

Åland Islands Mariehamn *https://restcountries.eu/data/ala.svg*

Albania Tirana *https://restcountries.eu/data/alb.svg*

Algeria Algiers *https://restcountries.eu/data/dza.svg*

American Samoa Pago Pago *https://restcountries.eu/data/asm.svg*

Andorra Andorra la Vella *https://restcountries.eu/data/and.svg*

Angola Luanda *https://restcountries.eu/data/ago.svg*

Anguilla The Valley *https://restcountries.eu/data/aia.svg*

Antarctica <empty string> *https://restcountries.eu/data/ata.svg*

Antigua and Barbuda Saint John's *https://restcountries.eu/data/atg.svg*

Argentina Buenos Aires *https://restcountries.eu/data/arg.svg*

Armenia Yerevan *https://restcountries.eu/data/arm.svg*

Aruba Oranjestad *https://restcountries.eu/data/abw.svg*

Australia Canberra *https://restcountries.eu/data/aus.svg*

Austria Vienna *https://restcountries.eu/data/aut.svg*

Azerbaijan Baku *https://restcountries.eu/data/aze.svg*

Bahamas Nassau *https://restcountries.eu/data/bhs.svg*

Bahrain Manama *https://restcountries.eu/data/bhr.svg*

Bangladesh Dhaka *https://restcountries.eu/data/bgd.svg*

Barbados Bridgetown *https://restcountries.eu/data/brb.svg*

Belarus Minsk *https://restcountries.eu/data/blr.svg*

Belgium Brussels *https://restcountries.eu/data/bel.svg*

Belize Belmopan *https://restcountries.eu/data/blz.svg*

Benin Porto-Novo *https://restcountries.eu/data/ben.svg*

Print the total population of countries using reduce function

Code :

```js
JS script.js > ...
1    var connect = new XMLHttpRequest();
2    connect.open('GET',"https://restcountries.eu/rest/v2/all",true);
3    connect.send();
4    connect.onload = function()
5    {
6        var data = JSON.parse(this.response);
7      // console.log(data);
8        var res = data.reduce((sum,i)=>
9        {
10           return i.population+sum ;
11       },0);
12       console.log(res);
13   };
```

Output :

🗑 ▽ Filter Output
      7349137231
»

Print the country which use US Dollars as currency

Code:

```javascript
1    var connect = new XMLHttpRequest();
2    connect.open('GET', "https://restcountries.eu/rest/v2/all",true);
3    connect.send();        (local function)(): void
4    connect.onload = function()
5    {
6        var data = JSON.parse(this.response);
7        // console.log(data[239].currencies[0].code);
8        var res = data.filter((i)=>{
9            if(i.currencies[0].code==="USD")
10               return true ;
11           else
12               return false ;
13       })
14
15       var res = res.map((i)=>{
16           return i.name ;
17       })
18       console.log(res);
19   };
```

Output :

```
▼ (14) [_]
    0: "American Samoa"
    1: "Bonaire, Sint Eustatius and Saba"
    2: "British Indian Ocean Territory"
    3: "United States Minor Outlying Islands"
    4: "Virgin Islands (U.S.)"
    5: "Ecuador"
    6: "El Salvador"
    7: "Guam"
    8: "Marshall Islands"
    9: "Northern Mariana Islands"
    10: "Puerto Rico"
    11: "Timor-Leste"
    12: "Turks and Caicos Islands"
    13: "United States of America"
    length: 14
  ▶ <prototype>: Array []
```