1.Solving problems using array functions on rest countries data.

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

**var request=new XMLHttpRequest();**

**request.open("GET","https://raw.githubusercontent.com/rvsp/restcountries-json-data/master/res-countries.json")**

**request.send();**

**request.onload=function(){**

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

**console.log(data);**

**//Using the above code extract the country name which is in the asia region**

**let res= data.filter((ele)=>ele.region==="Asia");**

**console.log(res);**

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

**console.log(res[i].name);**

**}**

**}**

**////////////////////////**

**var request=new XMLHttpRequest();**

**request.open("GET","https://raw.githubusercontent.com/rvsp/restcountries-json-data/master/res-countries.json")**

**request.send();**

**request.onload=function(){**

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

**console.log(data);**

**//Using the above code extract the country name which is in the asia region**

**let res= data.filter((ele)=>ele.region==="Asia");**

**console.log(res);**

**}**

**B.Get all the countries with a population of less than 2 lakhs using Filter function**

**var request=new XMLHttpRequest();**

**request.open("GET","https://raw.githubusercontent.com/rvsp/restcountries-json-data/master/res-countries.json")**

**request.send();**

**request.onload=function(){**

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

**console.log(data);**

**//Get all the countries with a population of less than 2 lakhs using Filter function**

**var res=data.filter((ele)=>ele<200000===0);**

**for( var i=1;i<data.length;i++)**

**console.log(res);**

**}**

**C.Print the following details name, capital, flag using forEach function**

**var request=new XMLHttpRequest();**

**request.open("GET","https://raw.githubusercontent.com/rvsp/restcountries-json-data/master/res-countries.json")**

**request.send();**

**request.onload=function(){**

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

**console.log(data);**

**function initialize(countriesName);**

**countries=countriesName;**

**let options="";**

**countries.forEach(country=>countriesName**

**console.log(`countriesName:${data[i].countriesName}`);**

**});**

**D.Print the total population of countries using reduce function**

**var request=new XMLHttpRequest();**

**request.open("GET","https://raw.githubusercontent.com/rvsp/restcountries-json-data/master/res-countries.json")**

**request.send();**

**request.onload=function(){**

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

**console.log(data);**

**var res=data.filter((ele)=>ele.region==="Asia");**

**for( i=0;i<res.length;i++)**

**console.log(res[i].population);**

**function population(total,country){**

**let totalpopulation=0;**

**if(total.population){**

**totalpopulation=total.population;**

**}**

**else{**

**totalpopulation=total**

**}**

**return totalpopulation+country.population;**

**}**

**console.log(population);**

**}**

**E.Print the country which uses US Dollars as currency.**

**var request=new XMLHttpRequest();**

**request.open("GET","https://raw.githubusercontent.com/rvsp/restcountries-json-data/master/res-countries.json")**

**request.send();**

**request.onload=function(){**

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

**console.log(data);**

**//Print the country which uses US Dollars as currency**

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

**console.log(`USdollarsAsCurrency:${data[i].USdollarsAsCurrency}`);**

**}**