1.Html and script.js file and run a for loop on the data and print all the country names in the console.
Answer:- In the HTML and JS file.

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

Answer:- In Call by value method original value is not modified whereas, in Call by reference method, the original value is modified.In Call by value, a copy of the variable is passed whereas in Call by reference, a variable itself is passed.In Call by value, actual and formal arguments will be created in different memory locations whereas in Call by reference, actual and formal arguments will be created in the same memory location.Call by value is the default method in programming languages like C++, PHP, Visual Basic NET, and C# whereas Call by reference is supported only in the Java language.Call by Value, variables are passed using a straightforward method whereas Call by Reference, pointers are required to store the address of variables.

3.How to copy by value a composite data type (array+objects).

Answer:- There are three ways To Copy By Value Any Composite Data Type In JavaScript
- -Use the spread (...) syntax.
- -Use the Object.assign() method.
- -Use the JSON.stringify() and JSON.parse() methods.

4.JSON Tasks

Problem 0 : Part A :

```
var cat = {name: 'Fluffy', activities: ['play', 'eat cat food'],
 catFriends: [
 {
    name: 'bar', activities: ['be grumpy', 'eat bread omblet'],
    weight: 8, furcolor: 'white'

 },
 {
```

```
 name: 'foo',
 activities: ['sleep', 'pre-sleep naps'],
 weight: 3
 }
 ]
}
if(cat.name=='Fluffy'){
   cat.name='Fluffyy';
cat["height"]='1.2 ft';
cat["weight"]=6;
}
console.log(cat.height,cat.weight);

console.log(cat.name);

console.log(cat.catFriends[0].activities,cat.catFriends[1].activities);

console.log(cat.catFriends[0].name,cat.catFriends[1].name);

console.log(cat.catFriends[0].weight+cat.catFriends[1].weight);

let c=cat.activities;
let c1=cat.catFriends[0].activities;
let c2=cat.catFriends[1].activities;
console.log(c.length+c1.length+c2.length);

cat.catFriends[0].activities.push('eat well','sleep well');
cat.catFriends[1].activities.push('eat well','sleep well');
console.log(cat.catFriends[0].activities);
console.log(cat.catFriends[1].activities);

cat.catFriends[0].furcolor='Blue';
console.log(cat.catFriends[0].furcolor);
```

Output:

```
1.2 ft 6
Fluffyy
[ 'be grumpy', 'eat bread omblet' ] [ 'sleep', 'pre-sleep naps' ]
bar foo
11
6
[ 'be grumpy', 'eat bread omblet', 'eat well', 'sleep well' ]
[ 'sleep', 'pre-sleep naps', 'eat well', 'sleep well' ]
Blue
```

Execution Time:

```
0.068s
```

Memory Used:

```
8088kb
```

Problem 0 : Part B:

```
var myCar = {
 make: 'Bugatti', model: 'Bugatti La Voiture Noire', year: 2019,
 accidents: [
 {
 date: '3/15/2019',
 damage_points: '5000',
 atFaultForAccident: true
 },
 {
 date: '7/4/2022',
 damage_points: '2200',
 atFaultForAccident: true
 },
 {
 date: '6/22/2021',
 damage_points: '7900',
 atFaultForAccident: true
 }
 ]
}

for(let i in myCar.accidents){
   if(myCar.accidents[i].atFaultForAccident === true){
      myCar.accidents[i].atFaultForAccident = false;
    }
    console.log(myCar.accidents[i]);
}
for(let i in myCar.accidents){
    console.log(myCar.accidents[i].date);
}
```

Output:

```
{ date: '3/15/2019',
  damage_points: '5000',
  atFaultForAccident: false }
{ date: '7/4/2022',
  damage_points: '2200',
  atFaultForAccident: false }
{ date: '6/22/2021',
  damage_points: '7900',
  atFaultForAccident: false }
3/15/2019
7/4/2022
6/22/2021
```

Problem no.1 :

```
var obj = {name : 'RajiniKanth', age : 33, hasPets : false};
function printAllValues(obj) {
   let a=[];
   a.push(obj.name);
   a.push(obj.age);
   a.push(obj.hasPets);

   return a;
}
console.log(printAllValues(obj));
```

Output:

```
[ 'RajiniKanth', 33, false ]
```

Execution Time:

```
0.067s
```

Memory Used:

```
8124kb
```

Problem no.2 :

Parsing an JSON object's Keys:

```
var obj = {name : 'RajiniKanth', age : 33, hasPets : false};
```

```
function printAllValues(obj) {
    let a=[];
    a=Object.keys(obj);
    return a;
}
console.log(printAllValues(obj));
```

Output:

```
[ 'name', 'age', 'hasPets' ]
```

Execution Time:

```
0.067s
```

Memory Used:

```
8064kb
```

Problem no.3:

Parsing an JSON object and convert it to a list:

Write a function called “convertObjectToList” which converts an object literal into an array of arrays.

Input (Object):

var object = {name: “ISRO”, age: 35, role: “Scientist”};

Output:

[[“name”, “ISRO”], [“age”, 35], [“role”, “Scientist”]]

```
var obj = {name: 'ISRO', age: 35, role: 'Scientist'};

function convertListToObject(obj) {
```

```
// obj.split(',');

 let a=[];

 let b=[];

 for(let i in obj){

    //a.push();

    a.push([i,obj[i]]);

 }

console.log(a);}

convertListToObject(obj);
```

Output:

```
[ [ 'name', 'ISRO' ], [ 'age', 35 ], [ 'role', 'Scientist' ] ]
```

Execution Time:

```
0.068s
```

Memory Used:

```
8148kb
```

Problem no.4:

Parsing a list and transform the first and last elements of it:

Input (Array):

```
var array = [“GUVI”, “I”, “am”, “Geek”];
```

Output:

```
var object = {
```

```
GUVI : “Geek”
}

var arr = ['GUVI', 'I', 'am', 'a geek'];
function transformFirstAndLast(arr) {
 let len=arr.length;
 let obj={};
  obj[arr[0]]=arr[len-1];
  console.log(obj);

}
transformFirstAndLast(arr);
```

Output:

```
{ GUVI: 'a geek' }
```

Execution Time:

```
0.067s
```

Memory Used:

```
8148kb
```

PROBLEM no.7:

Parsing two JSON objects and Compare:

```
var expected = {foo: 5, bar: 6};
var actual = {foo: 6, bar: 6};
function assertObjectsEqual(actual, expected){
 if(JSON.stringify(actual) === JSON.stringify(expected)){
    console.log("Passed");
 }
 else{
     console.log("Failed");
 }
}assertObjectsEqual(actual, expected);
```

Output:

```
Passed
```

Execution Time:

```
0.068s
```

Memory Used:

```
8124kb
```

Problem no.8:

```
var securityQuestions = [
 {
 question: 'What was your first pet’s name?',
 expectedAnswer: 'FlufferNutter'
 },
 {
 question: 'What was the model year of your first car?',
 expectedAnswer: '1985'
 },
 {
 question: 'What city were you born in?',
 expectedAnswer: 'NYC'
 }
]
var realquestion='What was the model year of your first car?';
var answer='1985';
function chksecurityQuestions(securityQuestions,question,ans) {
var output=false;
  for(var i=0;i<securityQuestions.length;i++){
     if(securityQuestions[i].question===question){
         if(securityQuestions[i].expectedAnswer===ans)
           output = true;
      }
  }
  return output;
}
```

```
var stats=chksecurityQuestions(securityQuestions,realquestion,answer);

console.log(stats);
```

Output:

```
true
```

Execution Time:

```
0.067s
```

Memory Used:

```
8056kb
```

Problem no.9:

Parsing JSON objects and Compare:

Write a function to return the list of characters below 20 age

```
var students = [

 {

 name: 'Siddharth Abhimanyu', age: 21}, { name: 'Malar', age: 25},

 {name: 'Maari',age: 18},{name: 'Bhallala Deva',age: 17},

 {name: 'Baahubali',age: 16},{name: 'AAK chandran',age: 23},   {name:'Gabbar Singh',age: 33},{name:
'Mogambo',age: 53},

 {name: 'Munnabhai',age: 40},{name: 'Sher Khan',age: 20},

 {name: 'Chulbul Pandey',age: 19},{name: 'Anthony',age: 28},

 {name: 'Devdas',age: 56}
 ];

function returnMinors(arr)

{

    let b=[];

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

        if(students[i].age<20){

            b.push(students[i].name);

        }

    }

    console.log(b);

}

returnMinors(students);
```

Output:

```
[ 'Maari', 'Bhallala Deva', 'Baahubali', 'Chulbul Pandey' ]
```

Execution Time:

```
0.067s
```

Memory Used:

```
8104kb
```

5. Try the rest countries api. Extract and print the total population of all the countries in the console. use the html template. https://restcountries.eu/rest/v2/all

Answer:-

```
Afghanistan 27657145

Åland Islands 28875

Albania 2886026

Algeria 40400000

American Samoa 57100

Andorra 78014

Angola 25868000

Anguilla 13452

Antarctica 1000

Antigua and Barbuda 86295

Argentina 43590400

Armenia 2994400

Aruba 107394

Australia 24117360
```

Austria 8725931

Azerbaijan 9730500

Bahamas 378040

Bahrain 1404900

Bangladesh 161006790

Barbados 285000

Belarus 9498700

Belgium 11319511

Belize 370300

Benin 10653654

Bermuda 61954

Bhutan 775620

Bolivia (Plurinational State of) 10985059

Bonaire, Sint Eustatius and Saba 17408

Bosnia and Herzegovina 3531159

Botswana 2141206

Bouvet Island 0

Brazil 206135893

British Indian Ocean Territory 3000

United States Minor Outlying Islands 300

Virgin Islands (British) 28514

Virgin Islands (U.S.) 114743

Brunei Darussalam 411900

Bulgaria 7153784

Burkina Faso 19034397

Burundi 10114505

Cambodia 15626444

Cameroon 22709892

Canada 36155487

Cabo Verde 531239

Cayman Islands 58238

Central African Republic 4998000

Chad 14497000

Chile 18191900

China 1377422166

Christmas Island 2072

Cocos (Keeling) Islands 550

Colombia 48759958

Comoros 806153

Congo 4741000

Congo (Democratic Republic of the) 85026000

Cook Islands 18100

Costa Rica 4890379

Croatia 4190669

Cuba 11239004

Curaçao 154843

Cyprus 847000

Czech Republic 10558524

Denmark 5717014

Djibouti 900000

Dominica 71293

Dominican Republic 10075045

Ecuador 16545799

Egypt 91290000

El Salvador 6520675

Equatorial Guinea 1222442

Eritrea 5352000

Estonia 1315944

Ethiopia 92206005

Falkland Islands (Malvinas) 2563

Faroe Islands 49376

Fiji 867000

Finland 5491817

France 66710000

French Guiana 254541

French Polynesia 271800

French Southern Territories 140

Gabon 1802278

Gambia 1882450

Georgia 3720400

Germany 81770900

Ghana 27670174

Gibraltar 33140

Greece 10858018

Greenland 55847

Grenada 103328

Guadeloupe 400132

Guam 184200

Guatemala 16176133

Guernsey 62999

Guinea 12947000

Guinea-Bissau 1547777

Guyana 746900

Haiti 11078033

Heard Island and McDonald Islands 0

Holy See 451

Honduras 8576532

Hong Kong 7324300

Hungary 9823000

Iceland 334300

India 1295210000

Indonesia 258705000

Côte d'Ivoire 22671331

Iran (Islamic Republic of) 79369900

Iraq 37883543

Ireland 6378000

Isle of Man 84497

Israel 8527400

Italy 60665551

Jamaica 2723246

Japan 126960000

Jersey 100800

Jordan 9531712

Kazakhstan 17753200

Kenya 47251000

Kiribati 113400

Kuwait 4183658

Kyrgyzstan 6047800

Lao People's Democratic Republic 6492400

Latvia 1961600

Lebanon 5988000

Lesotho 1894194

Liberia 4615000

Libya 6385000

Liechtenstein 37623

Lithuania 2872294

Luxembourg 576200

Macao 649100

Macedonia (the former Yugoslav Republic of) 2058539

Madagascar 22434363

Malawi 16832910

Malaysia 31405416

Maldives 344023

Mali 18135000

Malta 425384

Marshall Islands 54880

Martinique 378243

Mauritania 3718678

Mauritius 1262879

Mayotte 226915

Mexico 122273473

Micronesia (Federated States of) 102800

Moldova (Republic of) 3553100

Monaco 38400

Mongolia 3093100

Montenegro 621810

Montserrat 4922

Morocco 33337529

Mozambique 26423700

Myanmar 51419420

Namibia 2324388

Nauru 10084

Nepal 28431500

Netherlands 17019800

New Caledonia 268767

New Zealand 4697854

Nicaragua 6262703

Niger 20715000

Nigeria 186988000

Niue 1470

Norfolk Island 2302

Korea (Democratic People's Republic of) 25281000

Northern Mariana Islands 56940

Norway 5223256

Oman 4420133

Pakistan 194125062

Palau 17950

Palestine, State of 4682467

Panama 3814672

Papua New Guinea 8083700

Paraguay 6854536

Peru 31488700

Philippines 103279800

Pitcairn 56

Poland 38437239

Portugal 10374822

Puerto Rico 3474182

Qatar 2587564

Republic of Kosovo 1733842

Réunion 840974

Romania	19861408
Russian Federation	146599183
Rwanda	11553188
Saint Barthélemy	9417
Saint Helena, Ascension and Tristan da Cunha	4255
Saint Kitts and Nevis	46204
Saint Lucia	186000
Saint Martin (French part)	36979
Saint Pierre and Miquelon	6069
Saint Vincent and the Grenadines	109991
Samoa	194899
San Marino	33005
Sao Tome and Principe	187356
Saudi Arabia	32248200
Senegal	14799859
Serbia	7076372
Seychelles	91400

Sierra Leone 7075641

Singapore 5535000

Sint Maarten (Dutch part) 38247

Slovakia 5426252

Slovenia 2064188

Solomon Islands 642000

Somalia 11079000

South Africa 55653654

South Georgia and the South Sandwich Islands 30

Korea (Republic of) 50801405

South Sudan 12131000

Spain 46438422

Sri Lanka 20966000

Sudan 39598700

Suriname 541638

Svalbard and Jan Mayen 2562

Swaziland 1132657

Sweden 9894888

Switzerland 8341600

Syrian Arab Republic 18564000

Taiwan 23503349

Tajikistan 8593600

Tanzania, United Republic of 55155000

Thailand 65327652

Timor-Leste 1167242

Togo 7143000

Tokelau 1411

Tonga 103252

Trinidad and Tobago 1349667

Tunisia 11154400

Turkey 78741053

Turkmenistan 4751120

Turks and Caicos Islands 31458

Tuvalu 10640

Uganda 33860700

Ukraine 42692393

United Arab Emirates 9856000

United Kingdom of Great Britain and Northern Ireland 65110000

United States of America 323947000

Uruguay 3480222

Uzbekistan 31576400

Vanuatu 277500

Venezuela (Bolivarian Republic of) 31028700

Viet Nam 92700000

Wallis and Futuna 11750

Western Sahara 510713

Yemen 27478000

Zambia 15933883

Zimbabwe 14240168