

Name: Aswin s

Email: aswinsrini1@gmail.com

GUVI TASK-2

11-02-2021

1.HTML and Script.js and run a for loop on the data and print all the country name in the console.

HTML:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>sample api data</title>
  <link rel="stylesheet" href="task.css">
</head>
<body>
  <script src="taskjs.js">
  </script>
</body>
</html>
```

SCRIPTJS

```
var req =new XMLHttpRequest();
req.open("GET", 'https://restcountries.eu/rest/v2/all',true);
req.send()
req.onload = function(){
  var data=JSON.parse(this.response);

  for(var i in data)
  {
    console.log(data[i].name);
    document.write("<br>");
    document.write(data[i].name);
    document.write(` &nbsp;-&nbsp; ${data[i].capital}`);
    document.write("<br>");
  }
}
```

2. Write a difference between call by value and call by reference

Call by value:

Ex

Var a =50;

Var b=a;

Here a is stored in address x00FF

And b is stored in different address say x0AAC

If we change the value of a=55 . again the value of b is not changed

Call by reference:

Ex

Var arr=[1,2,3]

Var acc=arr;

Now arr and acc points to the same memory location If any change made in any of those will change the value of both

1.Jason Task...

....1.cat

```
//add height weighy
cat.height=15;
cat.weight=5;

// fluffy name change
cat.name="Fluffyy";

//cat friends name
for(var i in cat.catFriends)
{
    console.log(cat.catFriends[i].name);
}

//activites of fluffy cat friends
for(i in cat.catFriends)
{
    console.log("activities of ",cat.catFriends[i].name," ",cat.catFriends[i].activities);
}

//total weight of cat friends
let tw=0;
for(i in cat.catFriends)
{
    tw+=cat.catFriends[i].weight;
}
console.log("total weight of cat friends:",tw);

//activities of all cat
var tacc=cat.activities.length;
for(i in cat.catFriends)
{
    tacc+=cat.catFriends[i].activities.length;
}
console.log("\n total activitiesof all cat",tacc);

//add two more activites of bar and foo;
for(i in cat.catFriends)
{
    cat.catFriends[i].activities.push("scratching posts");
    cat.catFriends[i].activities.push("finding rat");
}
console.log(cat.catFriends[0].activities);
console.log(cat.catFriends[1].activities);
```

Output:

```
cat friends name are
bar
foo
```

```
activities of bar [ 'be grumpy', 'eat bread omblet' ]
activities of foo [ 'sleep', 'pre-sleep naps' ]
```

```
total weight of cat friends: 11
```

```
total activitiesof all cat 5
```

```
the new activites are
```

```
activities of cat name bar [ 'be grumpy', 'eat bread omblet',
    'scratching posts',
    'finding rat' ]
activities of cat name foo [ 'sleep', 'pre-sleep naps', 'scratching posts', 'finding rat' ]
```

2.Mycar

```
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
    }
  ]
}

//loop over accident Array change atFaultForAccident to false
for(var i in myCar.accidents)
{
  myCar.accidents[i].atFaultForAccident=false;
  console.log(myCar.accidents[i]);
}
console.log("\n");

//print date of my accidents
var x=1
for(i in myCar.accidents)
{
  console.log(x, '- ', myCar.accidents[i].date);
  x++;
}
```

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 }
```

```
1 '- ' '3/15/2019'
2 '- ' '7/4/2022'
3 '- ' '6/22/2021'
```

3.Pass object as function and print array

```
function printAllvalues(obj)
{
    var arr=[]
    for(var i in obj)
    {
        arr.push(obj[i]);
    }
    return arr;
}

var obj = {name:"Rajinikanth", age: 33, hasPets : false};
var ans=printAllvalues(obj);
console.log(ans);
```

Output:

```
[ 'Rajinikanth', 33, false ]
```

Execution Time:

0.075s

Memory Used:

8312kb

4. Write a function called “printAllKeys” which returns an newArray of all the input object’s keys.

```
function printAllvalues(obj)
{
    var arr=[]
    for(var i in obj)
    {
        arr.push(i);
    }
    return arr;
}

var obj = {name:"Rajinikanth", age: 33, hasPets : false};
var ans=printAllvalues(obj);
console.log(ans);
```

Output:

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

Execution Time:

0.077s

Memory Used:

8344kb

5. Parsing an JSON object and convert it to a list:

```
function printAllvalues(obj)
{
    var arr=[]
    for(var i in obj)
    {
        var arr1=[];
        arr1.push(i);
        arr1.push(obj[i]);
        arr.push(arr1);
    }
    return arr;
}

var object = {name: "ISRO", age: 35, role: "Scientist"};
var ans=printAllvalues(object);
console.log(ans);
```

Output:

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

Execution Time:

0.076s

Memory Used:

8328kb

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

```
function transformFirstAndLast(arr) {  
  var obj={};  
  var key=arr[0];  
  obj[arr[0]]=arr[arr.length-1];  
  return obj;  
}  
var array = ["GUVI","IAM","A","GEEK"];  
var obj=transformFirstAndLast(array);  
console.log(obj);
```

Output:

```
{ GUVI: 'GEEK' }
```

Execution Time:

0.075s

Memory Used:

8404kb

Parsing a list of lists and convert into a JSON object:

EDITOR

JAVASCRIPT

Run

Debug

```
1- function fromListToObject(arr) {
2-   var newObject = {};
3-
4-   for(var i in arr)
5-   {
6-     newObject[arr[i][0]] = arr[i][1];
7-   }
8-   return newObject;
9- }
10
11
12 // Getting input via STDIN
13 const readline = require("readline");
14
15 const inp = readline.createInterface({
16   input: process.stdin
17 });
18
19 inp.on("line", (data) => {
20   var array = [["make", "Ford"], ["model", "Mustang"], ["year", 1964]];
21   var obj = fromListToObject(array);
22   console.log(obj);
23
24 });
```

OUTPUT

Output:

{ make: 'Ford', model: 'Mustang', year: 1964 }

Execution Time:

0.075s

Memory Used:

8372kb

Parsing a list of lists and convert into a JSON object:

```
1- function transformEmployeeData(arr) {
2-   var transformEmployeeList = [];
3-   var count=0;
4-
5-   for(var i in arr)
6-   {
7-     var obj={};
8-     for(var j in arr[i])
9-     {
10-       obj[arr[i][j][0]] = arr[i][j][1];
11-     }
12-     transformEmployeeList.push(obj);
13-   }
14-   //Your code
15-
16-   return transformEmployeeList;
17- }
18 // Getting input via STDIN
19 const readline = require("readline");
20
21 const inp = readline.createInterface({
22   input: process.stdin
23 });
24
25 inp.on("line", (data) => {
26   var array = [
27     [{"firstName", "Vasanth"}, {"lastName", "Raja"}, {"age", 24}, {"role", "JSWizard"}]
28     [{"firstName", "Sri"}, {"lastName", "Devi"}, {"age", 28}, {"role", "Coder"}]
29   ];
30
31   var ans = transformEmployeeData(array);
32   console.log(ans);
33
34
35 });
```

Output:

[{ firstName: 'Vasanth',
 lastName: 'Raja',
 age: 24,
 role: 'JSWizard' },
 { firstName: 'Sri', lastName: 'Devi', age: 28, role: 'Coder' }]

Execution Time:

0.075s

Memory Used:

8308kb

Parsing two JSON objects and Compare:

```
1- function assertObjectEqual(actual, expected, testName){
2-   var x = JSON.stringify(actual);
3-   var y = JSON.stringify(expected);
4-   console.log(x);
5-   console.log(y);
6-   if(x[2] === y[2])
7-   {
8-     if(x[15] === y[15])
9-     {
10-       console.log("Passed");
11-     }
12-     else
13-     {
14-       console.log("failed", "expected", y, "BUT got", x);
15-     }
16-   }
17-   else
18-   {
19-     console.log("failed", "expected", y, "BUT got", x);
20-   }
21-   // your code here
22- }
23 // Getting input via STDIN
24 const readline = require("readline");
25
26 const inp = readline.createInterface({
27   input: process.stdin
28 });
29
30 inp.on("line", (data) => {
31   var expected = {foo: 5, bar: 6};
32   var actual = {foo: 4, bar: 6};
33   assertObjectEqual(actual, expected, "hello");
34
35 });
```

Output:

{ "foo":4,"bar":6 }
{ "foo":5,"bar":6 }
failed expected { "foo":5,"bar":6 } BUT got { "foo":4,"bar":6 }

Execution Time:

0.073s

Memory Used:

8248kb

Parsing JSON objects and Compare:

```
1 function chksecurityQuestions(securityQuestions,question,ans) {
2
3 // your code here
4 for(var i in securityQuestions)
5 {
6     var n=JSON.stringify(securityQuestions[i].question).localeCompare(question);
7     if(n==0)
8     {
9         var c=JSON.stringify(securityQuestions[i].expectedAnswer).localeCompare(ans);
10        if(c==0)
11        {
12            return true;
13        }
14        else
15        {
16            return false;
17        }
18    }
19    else
20    {
21        return "wrong question";
22    }
23 }
24 return "test";
25 }
26
27 // Getting input via STDIN
28 const readline = require("readline");
29
30 const inp = readline.createInterface({
31     input: process.stdin
32 });
33
34 inp.on("line", (data) => {
35     var securityQuestions = [
36     {
37         question: "What was your first pet's name?",
38         expectedAnswer: "FlufferNutter"
39     },
40     {
41         question: "What was the model year of your first car?",
42         expectedAnswer: "1985"
43     },
44     {
45         question: "What city were you born in?",
46         expectedAnswer: "NYC"
47     }
48 ];
49
50 var ques = "What was your first pet's name?";
51 var ans = "FlufferNutter";
52
53 var status = chksecurityQuestions(securityQuestions, ques, ans);
54 console.log(status);
55
56 });
```

Output:
true

Execution Time:
0.075s

Memory Used:
8544kb

Parsing JSON objects and Compare:

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

```
1 function returnMinors(arr)
2 {
3     var x=[];
4     for(var i in arr)
5     {
6         if(arr[i].age<20)
7         {
8             x.push(arr[i]);
9         }
10    }
11    return x;
12 }
13
14 // Getting input via STDIN
15 const readline = require("readline");
16
17 const inp = readline.createInterface({
18     input: process.stdin
19 });
20
21 inp.on("line", (data) => {
22     var students = [
23     {name: "Siddharth Abhimanyu", age: 21},
24     { name: "Malar", age: 25},
25     {name: "Maari",age: 18},
26     {name: "Bhallala Deva",age: 17},
27     {name: "Baahubali",age: 16},
28     {name: "AAK chandran",age: 23},
29     {name: "Gabbbar Singh",age: 33},
30     {name: "Mogambo",age: 53},
31     {name: "Munnabhai",age: 40},
32     {name: "Shen Khan",age: 20},
33     {name: "Chulbul Pandey",age: 19},
34     {name: "Anthony",age: 28},
35     {name: "Devdas",age: 56}
36 ];
37
38 console.log(returnMinors(students));
39
40 });
```

Output:
[{ name: 'Maari', age: 18 },
 { name: 'Bhallala Deva', age: 17 },
 { name: 'Baahubali', age: 16 },
 { name: 'Chulbul Pandey', age: 19 }]

Execution Time:
0.075s

Memory Used:
8424kb