Name: Praveen kumar kumpatla
Batch: B21WD
Date: 10-02-2021 (Day 3)

Task 2 Guvi Zen class

1. Create HTML and JS files and run a for loop on data and print all country names.

A. <u>**HTML File:**</u> filename: index.html

```
<!DOCTYPE html>
<html>
<head>
        <title> Sample program </title>
        <script src="script.js/path></script>
</head>
<body>
</body>
</html>
```

<u>**JS file:**</u> Filename:script.js

```
Var request = new XMLHttpRequest();
request.open('GET',"https://restcountries.eu/rest/v2/all" ,true);
request.send();
request.onload = function(){
        var data =JSON.parse(this.response);
        for(var i in data)
        console.log(data[i].name);
        }
```

2. Difference between Copy by value and Copy by reference?

A.

<u>Copy by value</u>	<u>Copy by reference</u>
1. It passes the copies of data in a variable.	1. It passes the address of the variable
2. There is no effect on the main variable if we modified the copied variable.	2. There is an effect on the main variable.

3. How to copy by value a composite data type?

A. There are three ways to copy by value a composite data type:

1. Using (...variable name)
2. Using the Object.assign(variable name).
3. Using the JSON.parse(JSON.stringify(variable name)).

5. Try the rest countries' API. Extract and print the total population of all the countries in the console using the HTML template?

A.<u>**HTML File:**</u> filename: index.html

```
<!DOCTYPE html>
<html>
<head>
```

```
            <title> Sample program </title>
            <script src="script.js/path"></script>
      </head>
      <body>
            <p id="demo"></p>
      </body>
      </html>
```

JS file: Filename:script.js

```
      Var request = new XMLHttpRequest();
      request.open('GET',"https://restcountries.eu/rest/v2/all" ,true);
      request.send();
      request.onload = function(){
            var data =JSON.parse(this.response),total=0;
            for(var i in data)
            total+=data[i].population;
            document.getElementById("demo").innerHTML = "total population" + total;
            }
```

4. **JSON Task**
A. **Problem 0: Part A**:

 Tasks:

A. Add height and weight to Fluffy

B. Fluffy name is spelled wrongly. Update it to Fluffyy

C. List all the activities of Fluffyy's catFriends.

D. Print the catFriends names.

E. Print the total weight of catFriends

F. Print the total activities of all cats (op:6)

G. Add 2 more activities to bar & foo cats

H. Update the fur color of bar.

```
      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}]};
      //After calling these API
      var totalwei=0,totalact=[...cat.activities];
```

```javascript
cat.height=2,cat.weight=6;//task1
cat.name="Fluffyy";//task2
for(var i in cat.catFriends){
    console.log(cat.catFriends[i].activities);//task3
    console.log(cat.catFriends[i].name);//task4
    totalwei+=cat.catFriends[i].weight;
}
console.log(totalwei);//task5
for(var j in cat.catFriends){
    for(var k in cat.catFriends[j].activities)
    totalact.push(cat.catFriends[j].activities[k]);
}
console.log(totalact);//task6
cat.catFriends[0].activities.push("play","drinking milk");//task7
cat.catFriends[1].activities.push("play","drinking milk");
cat.catFriends[0].furcolor="black";//task8
```

Problem 0: Part B:

Tasks:

1. Loop over the accidents array. Change atFaultForAccident from true to false.
2. Print the dated of my accidents.

```javascript
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}]};
//After calling this API
for(var i in myCar.accidents){
        myCar.accidents[i].atFaultForAccident=false;//task1
        console.log(myCar.accidents[i].date);//task2
}
```

Problem 1:

Write a function called "printAllValues" which returns an newArray of all the input object's values?

A.
```javascript
function printAllValues(obj){
    console.log(Object.values(obj))
}
printAllValues(called JSON object name);
```

Problem 2:

Write a function called "printAllKeys" which returns an newArray of all the input object's values?

A. function printAllKeys(obj){
   console.log(Object.keys(obj))
}
printAllValues(called JSON object name);

Problem 3:

Write a function called "convertObjectToList" which converts an object literal into an array of arrays.?

A. function convertObjectToList(obj){
   key=Object.keys(obj);
   val=Object.values(obj);
   arr=[];
   for(i=0;i<key.length;i++){
      arr[i]=[key[i],val[i]];
   }
   console.log(arr);
}
convertObjectToList(JSON called object);

Problem 4:

Write a function 'transformFirstAndLast' that takes in an array, and returns an object with:
1) the first element of the array as the object's key, and
2) the last element of the array as that key's value.

A. var a=['guvi','i','am','geek'];
   function transformFirstAndLast(arr){
      newObject={};
      n=arr.length;
      for(i=0;i<n/2;i++)
      newObject[arr[i]]=arr[n-1-i];
      return newObject;
   }
   console.log(transformFirstAndLast(a));

Problem 5:

Write a function "fromListToObject" which takes in an array of arrays, and returns an object with each pair of elements in the array as a key-value pair.

A. var a=[['guvi','i'],['am','geek'],['since',1998]];
```
function fromListToObjects(arr){
   newObject={};
   n=arr.length;
   for(i=0;i<n;i++)
   newObject[arr[i][0]]=arr[i][1];
   return newObject;
}
console.log(fromListToObjects(a));
```

Problem 6:
Write a function called "transformGeekData" that transforms some set of data from one format to another.

A. var a=[[['firstName', 'Vasanth'], ['lastName', 'Raja'], ['age', 24], ['role', 'JSWizard']] , [['firstName', 'Sri'], ['lastName', 'Devi'], ['age', 28], ['role', 'Coder']]];
```
function transformGeekData(arr){
   obj={},transformEmployeeList=[];
   n=arr.length;
   for(i=0;i<n;i++){
      m=arr[i].length;
      for(j=0;j<m;j++)
      obj[arr[i][j][0]]=arr[i][j][1];
      transformEmployeeList.push({...obj});
   }
   return transformEmployeeList;
}
console.log(transformGeekData(a));
```

Problem 7:

Write an "assertObjectsEqual" function from scratch.
Assume that the objects in question contain only scalar values (i.e., simple values like strings or numbers).

It is OK to use JSON.stringify().
Note: The examples below represent different use cases for the same test. In practice, you should never have multiple tests with the same name.

A. var expected={foo:5,bar:6};
   var actual={foo:6,bar:6};
   assertObjectsEqual(actual,expected,'Detects that two objects are same');
   function assertObjectsEqual(obj,obj1,text){
      var values=Object.values(obj);
      var values1=Object.values(obj1);
      var check=1;
      for(i=0;i<values.length;i++){
         if(values[i]!==values1[i])
         check=0;
      }
      if(check===0)
      console.log("Failed [my text] Expected"+JSON.stringify(obj1)+"But
   got"+JSON.stringify(obj));
      else
      console.log("Passed");
   }

Problem 8:
Parsing JSON objects and Compare:
I have a mock data of security Questions and Answers. You function should take the object
and a pair of strings and should return if the quest is present and if its valid answer

A. function chksecurityQuestions(securityQuestions,ques,answer) {
      output=false;
      for(i=0;i<securityQuestions.length;i++){
         if(securityQuestions[i].question==ques){
            if(securityQuestions[i].expectedAnswer==answer)
            output=true;
         }
      }
      return output;
   }
   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 ques = 'What was your first pet's name?';
   var ans = 'FlufferNutter';
   var status = chksecurityQuestions(securityQuestions, ques, ans);
   console.log(status); // true
   var ques = 'What was your first pet's name?';
   var ans = 'BlufferNutter';
   status = chksecurityQuestions(securityQuestions, ques, ans);
   console.log(status); // false

Problem 9:

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

A. 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)
{
   result=[];
   for(var i in arr){
      if(arr[i].age<20)
      result.push(arr[i].name);
   }
   return result;
}
console.log(returnMinors(students));
```