Problem 0: Part A

Playing with JSON object’s Values

Fluffy sorry, Fluffyy is my fav cat and it has 2 catFriends
Write a code to get the below details of Fluffyy so that
I can take him to vet.

Object:

```
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
             }
        ]
    };
```

//1.Add height and weight to Fluffy

Solution::

```
cat.weight = 6;
cat.height = 2;
```

//2.Fluffy name is spelled wrongly. Update it to Fluffyy

Solution::

```
cat.name = "Fluffyy";
```

//3.List all the activities of Fluffyy’s catFriends.

Solution::

```
let arr = cat.catFriends;
for(var i=0;i<arr.length;i++)
 {
     let temp = arr[i].activities;
     for(var j=0;j<temp.length;j++)
     {
        Console.log(temp[j]);
     }
 }
```

//4.Print the catFriends names.

Solution::

```
for(var i=0;i<cat.catFriends.length;i++)
{
   console.log (cat.catFriends[i].name);
}
```

//5.Print the total weight of catFriends

Solution::

```
let weight = 0;
```

```
for(var i=0;i<cat.catFriends.length;i++)
{
   weight += cat.catFriends[i].weight;
}
console.log(weight);
```

//6.Print the total activities of all cats

Solution::

```
let arr = [];
arr.push(cat.activities,cat.catFriends[0].activities,cat.catFriends[1].activities);
for(var i=0;i<arr.length;i++)
{
   for(var j=0;j<arr[i].length;j++)
   {
      console.log(arr[i][j]);
   }
}
```

//7.Add 2 more activities to bar & foo cats

Solution::

```
cat.catFriends[0].activities.push("looking out of window","wandering around");
cat.catFriends[1].activities.push("play","be grumpy");
```

//8.Update the fur color of bar

Solution::

```
cat.catFriends[0].furcolor = 'brown';
```

Problem 0: Part B

Iterating with JSON object’s Values

Above is some information about my car. As you can see, I am not the best driver.
I have caused a few accidents.
Please update this driving record so that I can feel better about my driving skills

Object:

```
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
               }
```

```
        ]
      };
```

//1.Loop over the accidents array. Change atFaultForAccident from true to false.

Solution::

```
for(var i=0;i<myCar.accidents.length;i++)
{
   myCar.accidents[i].atFaultForAccident = false;
}
```

//2. Print the dated of my accidents

Solution::

```
for(var ind=0;ind<myCar.accidents.length;ind++)
{
   console.log(myCar.accidents[ind].date);
}
```

Problem 1:

Parsing an JSON object’s Values:

Q: Write a function called “printAllValues” which returns an newArray of all the input object’s values.

Input (Object):

var object = {name: “RajiniKanth”, age: 33, hasPets : false};

Output:

[“RajiniKanth”, 33, false]

Solution::

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


function printAllValues(obj) {
 return Object.values(obj);
}


console.log(printAllValues(obj));
```

Problem 2:

Parsing an JSON object’s Keys:

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

Example Input:

{name : ‘RajiniKanth’, age : 25, hasPets : true}

Example Output:

[‘name’, ‘age’, ‘hasPets’]

Solution::

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


function printAllKeys(obj) {
 return Object.keys(obj);
}
```

```
console.log(printAllKeys(obj));
```

Problem 3:

Parsing an JSON object and convert it to a list:

Q: 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”]]
```

Solution::

```
var object = {name: "ISRO", age: 35, role: "Scientist"};


function convertObjectToList(obj) {
 return Object.entries(obj);
}
console.log(convertObjectToList(object));
```

Problem 4:

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

Q: 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.

Input (Array):

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

Output:

```
var object = {
GUVI : “Geek”
}
```

Solution::

```
var arr = ["GUVI", "I", "am", "Geek"];

function transformFirstAndLast(arr) {
 var newObject = {};
 newObject[arr[0]] = arr[arr.length-1];
 return newObject;
}
console.log(transformFirstAndLast(arr));
```

Problem 5:

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

Q: 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.

Input (Array):

```
var array = [[“make”, “Ford”], [“model”, “Mustang”], [“year”, 1964]];
```

Output:

```
var object = {
```

```
make : “Ford”

model : “Mustang”,

year : 1964

}
```

Solution::

```
var arr = [["make", "Ford"], ["model", "Mustang"], ["year", 1964]];

function fromListToObject(arr){
    let newObject = {};
    for(i=0;i<arr.length;i++)
    {
        newObject[arr[i][0]] = arr[i][1];
    }
    return newObject;
}

var object = fromListToObject(arr);
console.log(object);
```

Problem 6:

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

Q: Write a function called “transformGeekData” that transforms some set of data from one format to another.

Input (Array):

var array = [[[“firstName”, “Vasanth”], [“lastName”, “Raja”], [“age”, 24], [“role”, “JSWizard”]], [[“firstName”, “Sri”], [“lastName”, “Devi”], [“age”, 28], [“role”, “Coder”]]];

Output:

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

Solution::

```
var array = [[["firstName", "Vasanth"], ["lastName", "Raja"], ["age", 24], ["role", "JSWizard"]], [["firstName", "Sri"], ["lastName", "Devi"], ["age", 28], ["role", "Coder"]]];


function transformGeekData(arr){
   let newObject = [];
   for(var i=0;i<arr.length;i++)
   {
      let object = {};
      for(var j=0;j<arr[0].length;j++)
      {
         object[arr[i][j][0]] = arr[i][j][1];
      }
      newObject.push(object);
   }
```

```
  return newObject;
}
var obj = transformGeekData(array);
console.log(obj);
```

Problem 7:

Parsing two JSON objects and Compare:

Read this : https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/JSON/stringify

Q: 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.

Success Case:

Input:

```
var expected = {foo: 5, bar: 6};
var actual = {foo: 5, bar: 6}
assertObjectsEqual(actual, expected, ‘detects that two objects are equal’);
```

Output:

```
Passed
```

Failure Case:

Input:var expected = {foo: 6, bar: 5};

```
var actual = {foo: 5, bar: 6}
assertObjectsEqual(actual, expected, ‘detects that two objects are equal’);
```

Output:

```
FAILED [my test] Expected {“foo”:6,”bar”:5}, but got {“foo”:5,”bar”:6}
```

Solution::

```
var expected = {foo: 5, bar: 6};

var actual = {foo: 5, bar: 6};


function assertsObjectEqual(actual,expected,testname)

{

   if(JSON.stringify(actual) === JSON.stringify(expected))

   return "Passed";


   return `FAILED [my test] Expected ${JSON.stringify(expected)}, but got ${JSON.stringify(actual)}`;

}


console.log(assertsObjectEqual(actual,expected,'detects that two objects are equal'));
```

Problem 8:

Parsing JSON objects and Compare:

Q: 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.

```
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”
 }
]
function chksecurityQuestions(securityQuestions,question) {

 // your code here
 return true or false;
}
//Test case1:
var ques = “What was your first pet’s name?”;
var ans  =  “FlufferNutter”;
var status = chksecurityQuestions(securityQuestions, ques, ans);
console.log(status); // true
//Test case2:
var ques = “What was your first pet’s name?”;
var ans  =  “DufferNutter”;
var status = chksecurityQuestions(securityQuestions, ques, ans);
console.log(status); // flase
```

Solution::

```
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"
 }
];

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

Problem 9:

Parsing JSON objects and Compare:

Q: 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)
{
}
console.log(returnMinors(students));
```

Solution::

```
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 res = [];
    for(var i=0;i<arr.length;i++)
    {
       if(students[i].age < 20)
          res.push(students[i].name);
    }
    return res;
 }

console.log(returnMinors(students));
```