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 the accidents array. Change atFaultForAccident from true to false.

Print the date of my accidents

**Answer:** for(var i=0;i<myCar.accidents.length;i++){

myCar.accidents[i].atFaultForAccident = false;

console.log(myCar.accidents[i].date)

}

2.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]

Answer:

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

var arr1=[]

function printAllKeys(obj) {

for(var k in obj){

arr1.push(obj[k])

}

console.log(arr1)

}

printAllKeys(obj);

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”]]

**Answer:**

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

var arr= []

function convertListToObject(obj) {

for ( var i in obj){

    arr.push([i,obj[i]])

}

console.log(arr);

}

convertListToObject(obj);

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.  
Input (Array):  
var array = [“GUVI”, “I”, “am”, “Geek”];  
Output:  
var object = {  
GUVI : “Geek”  
}

**Answer:**

var arr = ['GUVI', 'I', 'am', 'a geek'];

function transformFirstAndLast(arr) {

 var newObject = {}

 var len = arr.length - 1;

 newObject[arr[0]] = arr[len];

 return newObject;

}

var a = transformFirstAndLast(arr);

console.log(a);

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.  
Input (Array):  
var array = [[“make”, “Ford”], [“model”, “Mustang”], [“year”, 1964]];  
Output:  
var object = {  
make : “Ford”  
model : “Mustang”,  
year : 1964  
}

**Answer:**

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

function fromListToObject(arr) {

  var newObject = {};

 for( var i in arr){

  newObject[(arr[i][0])] = ((arr[i][1]))

   }

 return newObject;

}

var a = fromListToObject(arr);

console.log(a);

6. 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”}  
]

**Answer:**

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

function transformEmployeeData(arr) {

   var tranformEmployeeList = [];

for(let i in array){

    let tempobj = {}

 for(let j in array[i]){

    tempobj[array[i][j][0]] = array[i][j][1];

 }

 tranformEmployeeList.push(tempobj)

}

console.log(tranformEmployeeList)

}

transformEmployeeData(array);

7. Parsing two JSON objects and Compare:

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}

**Answer:**

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

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

function assertObjectsEqual(actual, expected, testName){

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

console.log("Passed "+testName ,"Expected "+JSON.stringify(actual), " and got "+JSON.stringify(expected) )

}else{

console.log("Failed "+testName ,"Expected "+JSON.stringify(actual), " but got "+JSON.stringify(expected) )

}

}

assertObjectsEqual(actual , expected,"Comparison");

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

**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,ans) {

for(var i in securityQuestions){

if (question == securityQuestions[i].question ){

if(ans == securityQuestions[i].expectedAnswer){

return true;

}

}else{

return 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); //false

Problem 9

Parsing JSON objects and Compare:

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

**Answer:**

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)

{

const ageabv20 = arr.filter(student =>{

return student.age < 20;

});

console.log(ageabv20);

}

returnMinors(students);