**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

}

]

}

1. Add height and weight to Fluffy

Ans: cat.weight=300;

Cat.height=200;

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

Ans: cat.name=”Fluffyy”;

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

Ans:

4. Print the catFriends names.

Ans: console.log(cat.catFriends[0].name);

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

5. Print the total weight of catFriends

Ans: var totalweight= cat.catFriends[0].weight + cat.catFriends[1].weight

console.log(totalweight);

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

Ans: console.log(cat.activities);

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

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

7. Add 2 more activities to bar & foo cats

Ans: cat.catFriends[0].activities[2] = “play”;

cat.catFriends[1].activities[2]=”play”;

8. Update the fur color of bar

Ans: cat.catFriends[0].furcolor = “brown”;

**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

}

]

}

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

Ans: for(var i=0;i<myCar.accidents.length;i++)

{

myCar.accidents[i].atFaultForAccident =false;

}

2. Print the dated of my accidents

Ans: for(var i=0;i<myCar.accidents.length;i++)

{

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

}

**Problem 1**

Parsing an JSON object’s Values:

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]

**Ans:**

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

function printAllValues(obj1) {

console.log(Object.values(obj));

}

printAllValues(obj);

**Problem 2:**

Parsing an JSON object’s Keys:

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’]

**Ans:**

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

function printAllKeys(obj){

console.log(Object.keys(obj));

}

printAllKeys(obj);

**Problem 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”]]

**Ans:**

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

function covertobjtoarrray(obj){

let arr = Object.entries(objj);

console.log(arr);

}

covertobjtoarrray(objj);

**Problem 4:**

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

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”

}

**Ans:**

function transformFirstAndLast(array) {

var myObject = {}

array.forEach(function(){

myObject[array[0]] = array[array.length-1]}

)

return myObject

}

console.log(transformFirstAndLast(["GUVI", "I", "am", "a geek"]))

**Problem 5 :**

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

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

}

**Ans:**

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

function fromListToObject(array) {

emptyObject = {}

for(var i = 0; i < array.length; i++){

var newArray = array[i];

emptyObject[newArray[0]] = newArray[1];

}

return emptyObject;

}

var object = fromListToObject(array);

console.log(object);

**Problem 6:**

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

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”}

]

**Ans:**

var arr= [

[["firstName", "Vasanth"], ["lastName", "Raja"], ["age", 24], ["role", "JSWizard"]],

[["firstName", "Sri"], ["lastName", "Devi"], ["age", 28], ["role", "Coder"]]];

function transformGeekData(employeeData) {

var arr = []

for (var i = 0; i < employeeData.length; i ++) {

obj = {}

for (var j = 0; j < employeeData[i].length; j ++) {

var key = employeeData[i][j][0];

var value = employeeData[i][j][1];

obj[key] = value;

}

arr.push(obj);

}

return arr;

}

console.log(transformGeekData(arr))

**Problem 7 :**

Parsing two JSON objects and Compare:

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

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

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

assertObjectsEqual(actual, expected, ‘detects that two objects are equal’); Failure Case:

Output:

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

Ans:

Success Case:

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

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

assertObjectsEqual(actual, expected, 'detects that two objects are equal');

// console output:

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

function assertObjectsEqual(actual, expected, testName) {

// your code here

//convert object to string.

var actualString = JSON.stringify(actual);

var expectedString = JSON.stringify(expected);

if(actualString != expectedString) {

console.log('FAILED [' + testName + '] Expected "' + expectedString + '", but got "' + actualString + '"');

} else {

console.log("passed");

}

}

Failure Case:

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

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

assertObjectsEqual(actual, expected, 'detects that two objects are equal');

// console output:

// passed

function assertObjectsEqual(actual, expected, testName) {

// your code here

//convert object to string.

var actualString = JSON.stringify(actual);

var expectedString = JSON.stringify(expected);

if(actualString != expectedString) {

console.log('FAILED [' + testName + '] Expected "' + expectedString + '", but got "' + actualString + '"');

} 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

**Ans :**

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,ques,ans) {

for(var i=0; i<securityQuestions.length;i++)

{

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

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

return true;

}

}

}

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); // flase

**Problem 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)

{}console.log(returnMinors(students));

**Ans:**

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)

{

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

if(arr[i].age <20)

console.log(arr[i]);

}

}

returnMinors(students);