GUVI Classroom Assignment-2

Name: Shahida Midhat

Q.1 Html and script.js file and run a for loop on the data and print all the country names in the console.

Ans. //Create a request variable

```
var request=new XMLHttpRequest();
//Create a connection
request.open("GET","https://restcountries.eu/rest/v2/all",true);
//Send the request
request.send();
//Process and load response
request.onload=function() {
 var data=JSON.parse(this.response);
 console.log(data);

 for(let i=0;i<data.length;i++){
    console.log(data[i].name);
}
```

Q.2.Write a write up on Difference between copy by value and copy by reference.

Ans. Copy By Value:

1. Apply on primitive data type.
2. Changing the source value does not affect the copied value.
3. Deep copy.

Copy by Reference:

1. Apply on composite data type.
2. Changing the original value or copied value affects both original and copied value.
3. Shallow copy

Q.3. How to copy by value a composite data type (array+objects).

Ans: There are 3 methods to copy by value a composite data type:

1. **Use of spread(...) operator:** Spread operator allows an iterable to expand in places where 0+ arguments are expected.Using spread will clone your object.
2. **Use of Object.assign() method**: The Object.assign() method copies all enumerable own properties from one or more *source objects* to a *target object*. It returns the target object.
3. **Use of JSON.parse() and JSON.stringify()**: This performs deep copy.

Q.4. Perform these JSON tasks.

1. **Problem 0 : Part A**

```
var cat = {

name: ‘Fluffy’,

activities: [‘play’, ‘eat cat food’],
catFriends: [
{
name: ‘bar’,
activities: [‘be grumpy’, ‘eat bread omelet’],
weight: 8,
furcolor: ‘white’
},
{
name: ‘foo’,
activities: [‘sleep’, ‘pre-sleep naps’],
weight: 3
}
]
}
console.log(cat);
```

1. **Add height and weight to Fluffy**: cat[height]=”0.5ft”;

   cat[weight]=”6kg”;

2. **Fluffy name is spelled wrongly. Update it to Fluffyy**:
   cat[name]=”Fluffyy”;

3. **List all the activities of Fluffyy’s catFriends:**
   for(let i=0;i<cat.catFriends.length;i++){
   console.log(cat.catFriends[i].activities);
   }

4. **Print the catFriends names:**
   for(let i=0;i<cat.catFriends.length;i++){
   console.log(cat.catFriends[i].name);
   }

5. **Print the total weight of catFriends:**
   let total_weight=0;
   for(let i=0;i<cat.catFriends.length;i++){
   total_weight+=cat.catFriends[i].weight;
   }
   console.log(total_weight);

6. **Add 2 more activities to bar & foo cats:**
   for(let i=0;i<cat.catFriends.length;i++){
   cat.catFriends[i].activities.push(“playing with ball”);

   cat.catFriends[i].activities.push(“jumping”);
   }

7. **Print the total activities of all cats:**
   let total_activities=0;
   for(let i=0;i<cat.catFriends.length;i++){
   total_activities+=(cat.catFriends[i].activities.length); }
   console.log(total_activities+ (cat.activities.length));

8. **Update the fur color of bar:**
   for(let i=0;i<cat.catFriends.length;i++){
   if(cat.catFriends[i].name==”bar”){
   cat.catFriends[i].furcolor=”black”;
   }

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:**
   for(let i=0;i<car.accidents.length;i++){
   car.accidents[i].atFaultForAccident=false; }

2. . **Print the date of my accidents:**
   for(let i=0;i<car.accidents.length;i++){
   console.log(car.accidents[i].date); }

PROBLEM 1:

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

Ans:

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

function printAllValues(obj) {

Let values=[];

 for(i in obj){

values.push(obj.i);
```

```
}

console.log(values)

}
```

PROBLEM 2:

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

```
function printAllKeys(obj) {

 console.log(Object.keys(obj))

}
```

PROBLEM 3:

Write a function called “convertObjectToList” which converts an object literal into an array of arrays.

```
var obj = {name: “ISRO”, age: 35, role: “Scientist”};

function convertListToObject(obj) {

let objList=[];

 for(i in obj){

objList.push([i,obj[i]);

}

}
```

PROBLEM 4:

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

```
var arr = [“GUVI”, “I”, “am”, “a geek”];

function transformFirstAndLast(arr) {

 let newObject={};

newObject[arr[0]]=arr[arr.length-1];

 return newObject;

}
```

PROBLEM 5:

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

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

function fromListToObject(arr) {

 var newObject = {};

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

newObject[arr[i][0]]=arr[i][1];

}

 return newObject;

}
```

PROBLEM 6:

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

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

function transformEmployeeData(arr) {

 var tranformEmployeeList = [];

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

 let obj={}

  for(let j=0;j<arr[i].length;j++){

   obj[arr[i][j][0]]=arr[i][j][1];

}

tranformEmployeeList.push(obj);

}


 return tranformEmployeeList;
```

```
}
```

Problem 7:

Parsing two JSON objects and Compare:

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

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

function assertObjectsEqual(actual, expected, testName){

if(JSON.stringify(actual)==JSON.stringify(expected)){
        console.log ("PASSED");
 }
 else{
        console.log ("FAILED ", testName, " Expected ", (expected) ," but got ",
(actual));
 }
}
assertObjectsEqual(actual, expected, testName);
```

Problem 8:

Parsing two JSON objects and Compare:

I have mock data of security Questions and Answers. Your 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) {
for(let i=0;i<securityQuestions.length;i++){
        if( securityQuestions[i].question==ques ){
        if (securityQuestions[i].expectedAnswer==ans){
        return true;
        }
 }
 }
 return false;
}
```

PROBLEM 9:

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)

{

let character=[];

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

        if(arr[i].age<20){

        character.push(arr[i].name);
```

```
        }
        }
        return character;
}
console.log(returnMinors(students));
```

Q.5 Try the rest countries api. Extract and print the total population of all the countries in the console. use the html template:

Code:

```
let total_population=0;
for(let i=0;i<data.length;i++){
        total_population+=+(data[0].population)
}
console.log(total_population);
```

Ans:

Total Population= 6914286250