**ASSESSMENT**

**1. How do you copy a composite data type (blog in medium) :** <https://visshnnutejaa.medium.com/how-to-copy-by-value-of-composite-datatype-b49738dfaece>

**2. Difference between HTTP1.1 vs HTTP2 (blog in medium) :** <https://visshnnutejaa.medium.com/http1-1-vs-http-2-afb81a6ad41>

**3. Why there is a difference in behaviour for copying contents in primitive and non-primitive type?**

<https://visshnnutejaa.medium.com/why-there-is-a-difference-in-behavior-for-copying-contents-in-primitive-and-non-primitive-type-774645579703>

**4.Write a blog about objects and its internal representation in java script:**

<https://visshnnutejaa.medium.com/objects-and-its-internal-representation-in-javascript-58a9a597248c>

**5.  execute and see at least 15 cli commands. like mkdir, ls etc.**

|  |  |
| --- | --- |
| ls | It shows list of contents in folder |
| cd | change directory |
| cd.. | go up one level in folder tree |
| up/down arrow key | to see previous typed commands |
| d: (disc name:) | to jump one local disk to another local disc |
| alt+ curser | to edit the command if it is long or a typo in it |
| mkdir | create directory |
| rmdir | remove directory |
| notepad demo.txt | creates a new file and opens in notepad |
| start demo.txt | opens a file |
| del demo.txt | deletes a file |
| start atom demo.txt | file opens in atom instead of notepad when atom is installed in your pc |
| rm demo.txt | remove a file |
| pwd | to know the present working directory |
| rm \* | removes everything in the directory or folder |

**6.What is the difference between window, screen and document in java script?**

<https://visshnnutejaa.medium.com/what-is-the-difference-between-window-screen-and-document-in-javascript-70e9da59059>

**7. JSON Tasks:**

These are suggested to do by arun Prakash sir before going to the main JSON Task.

**1. Write a function called “addFive”. Given a number, “addFive” returns 5 added to that number.**

var num = 10;

function addFive(num) {

return num+5

}

var result = addFive(num)

**2.Write a function called “getOpposite”. Given a number, return its opposite**

let num = 5.5;

function getOpposite(num){

if(num>0 && num === Math.floor(num)){

console.log(0-num)

}else if (num<0 && num === Math.floor(num)){

console.log(0-num)

}else if (num===0){

console.log(0)

}

else{

console.log("-1")

}

}

getOpposite(num)

**3. Fill in your code that takes an number minutes and converts it to seconds.**

var min = 5;

function toSeconds(min) {

return min\*60;

}

var secs = toSeconds(min)

**4. Create a function that takes a string and returns it as an integer.**

var mystr = "5";

function toInteger(mystr) {

let num = +mystr;

return num

}

var myint = toInteger(mystr)

**5. Create a function that takes a number as an argument, increments the number by +1 and returns the result.**

var myint = 0;

function nextNumber(myint) {

return myint+1

}

var myNextint = nextNumber(myint)

**6. Create a function that takes an array and returns the first element.**

var arr = [1, 2, 3];

function getFirstElement(arr) {

return arr[0]

}

var data = getFirstElement(arr)

**7. Convert Hours into Seconds. Write a function that converts hours into seconds.**

var arr = 10;

function hourToSeconds(arr) {

return arr\*60\*60

}

var data = hourToSeconds(arr)

**8. Find the Perimeter of a Rectangle. Create a function that takes height and width and finds the perimeter of a rectangle.**

function findPerimeter(num1,num2) {

let perimeter = 2\*(num1+num2);

return perimeter

}

var peri = findPerimeter(6,7)

**9. Given two numbers, return true if the sum of both numbers is less than 100. Otherwise return false.**

function lessThan100(num1,num2) {

let sum = num1+num2;

if(sum>100){

return false;

}else if(sum<100){

return true

}

}

var res = lessThan100(22,15)

**10. There is a single operator in JavaScript, capable of providing the remainder of a division operation. Two numbers are passed as parameters. The first parameter divided by the second parameter will have a remainder, possibly zero. Return that value.**

function remainder(num1,num2) {

return num1%num2

}

var res = remainder(1,3)

**11. Old macdonald had a farm:**

**MacDonald is asking you to tell him how many legs can be counted among all his animals. The farmer breeds three species:**

**turkey = 2 legs  
horse = 4 legs  
pigs = 4 legs**

**The farmer has counted his animals and he gives you a subtotal for each species. You have to implement a function that returns the total number of legs of all the animals.**

function CountAnimals(tur,horse,pigs) {

return tur\*2 + horse\*4 + pigs\*4

}

var legs = CountAnimals(2,3,5)

**12. Frames Per Second  
Create a function that returns the number of frames shown in a given number of minutes for a certain FPS.**

function frames(num1,num2) {

return num1 \* num2\*60

}

var fps = frames(10,25)

**13. Check if an Integer is Divisible By Five. Create a function that returns true if an integer is evenly divisible by 5, and false otherwise.**

function divisibleByFive(num1) {

let status = num1%5===0;

if(status===true){

return true;

}else{

return false

}

}

var divisible = divisibleByFive(5)

**14. Write a function called “isEven”. Given a number, “isEven” returns whether it is even.**

function isEven(num){

// your code here

let number = +num;

let status = isNaN(typeof(number));

if(status){

return -1;

}

if(status && num%2===0){

return true;

}else if(status && num%2!==0){

return false;

}

}

var even = isEven(5)

**15. Write a function called “areBothOdd”.  
Given 2 numbers, “areBothOdd” returns whether or not both of the given numbers are odd.**

function areBothOdd(num1, num2){

// your code here

if(num1%2!==0 && num2%2!==0){

return true;

}else{

return false;

}

}

let res = areBothOdd(1,3)

**16. Write a function called “getFullName”.  
Given a first and a last name, “getFullName” returns a single string with the given first and last names separated by a single space.**

function getFullName(firstName, lastName){

// your code here

return firstName+" "+lastName

}

let fullName = getFullName("Visshnnu","Tejaa")

**17. Write a function called “getLengthOfWord”.  
Given a word, “getLengthOfWord” returns the length of the given word.**

function getLengthOfWord(word1){

return word1.length;

}

let length = getLengthOfWord("Visshnnu")

**18. Write a function called “isSameLength”.  
Given two words, “isSameLength” returns whether the given words have the same length.**

function isSameLength(word1, word2){

if(word1.length===word2.length){

return true;

}else{

return false;

}

}

let length = isSameLength("guvi","geek")

console.log(length)

**19. Create a function to calculate the distance between two points defined by their x, y coordinates**

function getDistance(x1, y1, x2, y2)

{

let distance;

distance = Math.sqrt(Math.pow(x1-x2,2)+Math.pow(y1-y2,2))

return distance;

}

console.log(getDistance(100, 100, 400, 300));

**20. Write a function called “getNthElement”.  
Given an array and an integer, “getNthElement” returns the element at the given integer, within the given array. If the array has a length of 0, it should return ‘undefined’.**

function getNthElement(array,n){

return array[n]

}

console.log(getNthElement([1,2,3],1))

**21. Write a function called “getLastElement”.  
Given an array, “getLastElement” returns the last element of the given array. If the given array has a length of 0, it should return ‘-1’.**

function getLastElement(array){

if(array[array.length-1]===undefined){

return -1;

}

return array[array.length-1]

}

console.log(getLastElement[])

**22. Write a function called “getProperty”.  
Given an object and a key, “getProperty” returns the value of the property at the given key. If there is no property at the given key, it should return undefined.**

var obj = {

mykey: "value"

};

function getProperty(obj, key) {

return obj[key]

}

console.log(getProperty(obj,"mykey"));

**23. Write a function called “addProperty”.  
Given an object and a key, “addProperty” adds a new property on the given object with a value of true.**

var obj = {

mykey: "value"

};

function addProperty(obj, key){

// your code here

return obj[key]=true;

}

console.log(addProperty(obj,"mykey"));

**24. Write a function called “removeProperty”.  
Given an object and a key, “removeProperty” removes the given key from the given object.**

var obj = {

mykey: "value"

};

function removeProperty(obj, key){

delete obj[key]

}

console.log(removeProperty(obj,"mykey"))

**25. Return an array, where the first element is the count of positives numbers and the second element is sum of negative numbers.**

var arr = [-5, 10, -3, 12, -9, 5, 90, 0, 1];

let count = 0;

let sum = 0;

function countPositivesSumNegatives(arr) {

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

if(arr[i]>0){

count++;

}else{

sum = sum+arr[i]

}

}

return [count,sum];

}

console.log(countPositivesSumNegatives(arr));

**26. Create a function that receives an array of numbers and returns an array containing only the positive numbers**

function getPositives(ar)

{

// your code here

let pos = ar.filter(i=>i>0)

return pos;

}

var ar = [-5, 10, -3, 12, -9, 5, 90, 0, 1];

var ar2 = getPositives(ar);

console.log(ar2);

**27. Write a function `powersOfTwo` which will return list of all powers of 2 from 0 to n (where n is an exponent).**

function powersOfTwo(n){

let value = 2;

let res=[];

for(let i=0;i<n;i++){

let num = Math.pow(value,i);

res.push(num)

}

return res;

}

console.log(powersOfTwo(3))

**28. Find the maximum number in an array of numbers**

function findMax(ar)

{

// your code here

return Math.max.apply(null,ar)

}

var ar = [-5, 10, -3, 12, -9, 5, 90, 0, 1];

var max = findMax(ar);

console.log("Max: ", max);

**29. Print the first 100 prime numbers**

function printPrimes(nPrimes)

{

var n = 0;

var i = 2;

while(n < nPrimes)

{

if (isPrime(i))

{

console.log(n, " → ", i);

n++;

}

i++;

}

}

// Returns true if a number is prime

function isPrime(n)

{

// your code here

if(n){

return true

}else{

return false

}

}

console.log(isPrime(10))

**30. Create a function that will return in an array the first “nPrimes” prime numbers greater than a particular number “startAt”**

let primeNumbers = [];

function printPrimes(nPrimes)

{

for(let i=0;i<nPrimes;i++){

let flag = 0;

for(let j=2;j<i;j++){

if(i%j===0){

flag = 1

break;

}

}

if(flag===0&&i>1){

primeNumbers.push(i)

}

}

}

printPrimes(100);

// Returns true if a number is prime

function isPrime(n)

{

if(primeNumbers.includes(n)){

return true

}else{

return false

}

}

console.log(isPrime(10))

**31. Reverse a string**

var s = reverseString("JavaScript");

console.log(s);

function reverseString(s)

{

// your code here

return s.split("").reverse().join("")

}

**32. Create a function that will merge two arrays and return the result as a new array**

var ar1 = [1, 2, 3];

var ar2 = [4, 5, 6];

var ar = mergeArrays(ar1, ar2);

console.log(ar);

function mergeArrays(ar1, ar2)

{

var result = [];

//this will add the first array to the result array

for(let el of ar1)

{

result.push(el);

}

for(let el of ar2)

{

result.push(el);

}

//Some piece of code goes here

return result;

}

**33. Calculate the sum of numbers received in a comma delimited string**

console.log(sumCSV("1.5, 2.3, 3.1, 4, 5.5, 6, 7, 8, 9, 10.9"));

function sumCSV(s)

{

// your code here

let val = s.split(",")

let array = [];

val.forEach(i=>array.push(+i))

let sum = 0;

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

sum = sum+array[i]

}

return sum;

}

This is main JSON Task:

Problem 0 : Part A (15 mins):

## Playing with JSON object’s Values:

**Solution:**

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

}

]

};

// Add height and weight to Fluffy

cat.height = 5;

cat.weight = 4;

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

cat.name = "Fluffyy"

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

let activities = [];

let totalActivities = [];

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

for(let j=0;j<2;j++){

activities.push(cat.catFriends[i].activities[j])

totalActivities.push(cat.catFriends[i].activities[j])

}

}

console.log(activities)

// Print the catFriends names.

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

console.log(cat.catFriends[i].name)

}

// Print the total weight of catFriends

let weight = [];

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

weight.push(cat.catFriends[i].weight)

}

let sum = 0;

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

sum = sum+weight[i]

}

console.log(sum)

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

for(let i=0;i<activities.length-2;i++){

totalActivities.push(cat.activities[i])

}

// Add 2 more activities to bar & foo cats

cat.catFriends[0].activities.push("walking","drinking")

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

cat.catFriends[1].activities.push("peeing","sleeping")

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

// Update the fur color of bar

cat.catFriends[0].furcolor = "Black"

console.log(totalActivities)

console.log("\n")

console.log(cat);

# Problem 0 : Part B (15 mins):

## Iterating with JSON object’s Values

**Solution:**

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<myCar.accidents.length;i++){

myCar.accidents[i].atFaultForAccident = false

}

// 2. Print the dated of my accidents

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

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

}

console.log(myCar)

# ****Problem 1 (5 mins):****

## ****Parsing an JSON object’s Values:****

**Solution:**

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

// console.log(obj.length)

let array = []

function printAllValues(obj) {

// your code here

for(let i in obj){

let key = i;

let value = obj[i]

array.push(value)

}

return array;

}

console.log(printAllValues(obj))

# Problem 2(5 mins) :

## Parsing an JSON object’s Keys:

**Solution:**

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

// console.log(obj.length)

let array = []

function printAllValues(obj) {

// your code here

for(let i in obj){

let key = i;

let value = obj[i]

array.push(key)

}

return array;

}

console.log(printAllValues(obj))

# Problem 3( 7–9 mins):

## Parsing an JSON object and convert it to a list:

**Solution:**

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

// console.log(obj.length)

let array = []

function convertListToObject(obj) {

// your code here

for(let i in obj){

let singleArray = [];

let key = i;

let value = obj[i]

singleArray.push(key,value)

array.push(singleArray)

}

return array;

}

console.log(convertListToObject(obj))

# Problem 4( 5 mins):

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

**Solution:**

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

let key;

let value ;

function transformFirstAndLast(arr) {

let newObject = {

}

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

if(i===0){

key = arr[i]

}else if(i===arr.length-1){

value = arr[i]

}

}

console.log(key,value)

newObject.key = value;

return newObject;

}

console.log(transformFirstAndLast(arr))

# Problem 5 ( 7 -9 mins):

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

**Solution:**

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

function fromListToObject(arr) {

var newObject = {};

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

let key = arr[i][0];

let value = arr[i][1];

newObject[key] = value

}

return newObject;

}

console.log(fromListToObject(arr))

# Problem 6 (10 mins):

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

**Solution:**

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 j=0;j<arr.length;j++){

let newObject = {};

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

let key = arr[j][i][0];

let value = arr[j][i][1];

newObject[key] = value;

}

tranformEmployeeList.push(newObject)

}

return tranformEmployeeList;

}

console.log(transformEmployeeData(arr))

# Problem 7 (10 — 20 mins):

## Parsing two JSON objects and Compare:

**Solution:**

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

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

function assertObjectsEqual(actual, expected, testName){

// your code here

let falg = 0;

let str = "Failure"

for(let i in expected){

for(let j in actual){

if(i===j){

if(expected[i]===actual[j]){

return "Passed";

}else{

return `Failed expected:${expected} actual:${actual}`

}

}

}

}

}

console.log(assertObjectsEqual(actual,expected,true))

# Problem 8(10 mins):

## Parsing JSON objects and Compare:

**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) {

// your code here

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

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

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

return true;

}else{

return false

}

}

}

}

let ques = "What was your first pet’s name?";

let ans = "FlufferNutter";

console.log(chksecurityQuestions(securityQuestions, ques, ans));

# Problem 9(20 mins):

## Parsing JSON objects and Compare:

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 list = [];

for(let i in arr){

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

list.push(arr[i].name)

}

}

return list

}

console.log(returnMinors(students));