**Javascript Questions**

1. Write a JavaScript program to find duplicate values in a

JavaScript array.

var newarray = [1,2,2,3,3,4,4,5,6,7,10];

var sortarray = newarray.sort();

var duparray = [];

for(var i=0;i<sortarray.length-1;i++){

if(sortarray[i+1] == sortarray[i]){

duparray.push(sortarray[i]);

}

}

console.log(duparray);

# **2. JavaScript to find Odd or Even number!**

<input type="text" id="inputBtn">

var inputVal = document.getElementsById("inputBtn").value;

if(inputVal%2 == 0){

console.log("Even number");

}

else{

console.log("Odd number");

}

**3. Write a javascript program to get the integers in range(2,9);**

**Example: range(2,9);**

**Output: [3,4,5,6,7,8];**

var range = function(start,end){

if(end-start === 2){

return [start +1];

}

else{

var list = range(start,end - 1);

list.push(end-1);

return list;

}

}

console.log(range(2,9));

4. Write a JavaScript program to find the most frequent item of an array.

Sample array : var arr1=[3, 'a', 'a', 'a', 2, 3, 'a', 3, 'a', 2, 4, 9, 3];  
Sample Output : a (5 times);

var arr1=[3, 'a', 'a', 'a', 2, 3, 'a', 3, 'a', 2, 4, 9, 3];

var mf = 1;

var m = 0;

var item;

for (var i=0; i<arr1.length; i++)

{

for (var j=i; j<arr1.length; j++)

{

if (arr1[i] == arr1[j])

m++;

if (mf<m)

{

mf=m;

item = arr1[i];

}

}

m=0;

}

console.log(item+" ( " +mf +" times ) ") ;

5. Input = [1,2,3,4,5,6];

Output = 21;

var addArray = [1,2,3,4,5,6];

var output = 0;

for(i=0;i<addArray.length;i+=1){

output += addArray[i];

}

console.log(output);

6. Write a javascript program to calculate the factorial of a number.  
Example: 5! = 5x4x3x2x1 = 120

function factorial(x){

if(x == 0){

return 1;

}

else{

return x \* factorial(x-1);

}

}

console.log(factorial(5));

1. Write a javascript program to compute the exponent of a number.

Example: 8^2 = 8x8 = 64

function exponent(x,y){

if(y == 0){

return 1;

}

else{

return x \* exponent(x,y-1);

}

}

console.log(exponent(8,2));

8. Write a javascript program that accept two integers and display the larger.

var x = prompt("1st interger value");

var y = prompt("2st interger value");

if(x>y){

alert(x);

}

else{

alert(y);

}

9. Write a javascript program to construct the following pattern, using a number.

\*  
\*\*  
\*\*\*  
\*\*\*\*  
\*\*\*\*\*

var i,j,chareater;

for(i=1;i<=6;i++){

for(j=1;j<i;j++){

chareater = chareater+('\*');

}

console.log(chareater);

chareater = '';

}

10. Write a simple javascript program to join all elements of the following.

Output

"Red,Gree,White,Black"

"Red,Gree,White,Black"

"Red+Gree+White+Black"

var str = ["Red","Green","White","Black"];

console.log(str.toString());

console.log(str.join());

console.log(str.join('+'));

1. Write a program to validate a condition using ternary expression.

function ternary(student\_number){

return student\_number ? "2:00" : "4:00";

}

console.log(ternary(true));

console.log(ternary(false));

console.log(ternary(1));

**12. Write a javascript program to get even numbers on array.**

var array = [4,5,6,8,9,10,8,21,26,25];

var evenArray = [];

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

if(array[i]%2 == 0){

evenArray.push(array[i]);

}

}

console.log(evenArray);

1. **Write a javascript program to get textarea field only text not get the number.**

**var text = document.getElementById('textarea').value;**

**var text1 = /[0-9]/g;**

**var text2 = /[a-z]/g;**

**var number = text.match(text1);**

**var character = text.match(text2);**

**var a = number.join("");**

**var b = character.join("");**

**console.log(a,b);**

**14. Types of webstorage**

**1. Localstorage()**

**2. Sessionstorage()**

**3. Delete local storage();**

**1. Localstorage()**

The localStorage object stores the data with no expiration date. The data will not be deleted when the browser is closed, and will be available the next day, week, or year.

// Check if the localStorage object exists

if(localStorage){

$(document).ready(function(){

$(".save").click(function(){

// Get input name

var firstName = $("#firstName").val();

// Store data

localStorage.setItem("first\_name", firstName);

alert("Your first name is saved.");

});

$(".access").click(function(){

// Retrieve data

alert("Hi, " + localStorage.getItem("first\_name"));

});

});

} else{

alert("Sorry, your browser do not support local storage.");

}

1. **Sessionstorage()**

The sessionStorage object is equal to the localStorage object, except that it stores the data for only one session. The data is deleted when the user closes the specific browser tab.

// Check if the localStorage object exists

if(localStorage){

$(document).ready(function(){

$(".save").click(function(){

// Get input name

var lastName = $("#lastName").val();

// Store data

sessionStorage.setItem("last\_name", lastName);

alert("Your last name is saved.");

});

$(".access").click(function(){

// Retrieve data

alert("Hi, " + localStorage.getItem("first\_name") + " " + sessionStorage.getItem("last\_name"));

});

});

} else{

alert("Sorry, your browser do not support local storage.");

}

1. **Delete local storage and session storage.**

localStorage.removeItem(key);

sessionStorage.removeItem(key);

# 15. JavaScript Hoisting

<p id="demo"></p>

x = 5;

elem = document.getElementById("demo");

elem.innerHTML = x;

var x;

## 16. Self-Invoking Functions

(function (){

alert(“Hai”);

})();

1. Callback Functions

//Javascript callback function

function mySandwich(param1, param2, callback) {

alert('Started eating my sandwich.\n\nIt has: ' + param1 + ', ' + param2);

callback();

},

mySandwich('ham', 'cheese', function() {

alert('Finished eating my sandwich.');

});

//Jquery callback function

$(‘#id’).fadeIn(‘slow’, function(){

//callback function

});

1. Chaining Functions

$("button").click(function(){

$("#p1").css("color", "red").slideUp(2000).slideDown(2000);

});

19. What will the code below output to the console and why?

(function(){

var a = b = 3;

})();

console.log("a defined? " + (typeof a !== 'undefined'));

console.log("b defined? " + (typeof b !== 'undefined'));

20. Consider the two functions below. Will they both return the same thing? Why or why not?

function foo1()

{

return {

bar: "hello"

};

}

function foo2()

{

return

{

bar: "hello"

};

}

console.log("foo1 returns:");

console.log(foo1());

console.log("foo2 returns:");

console.log(foo2());

output

foo1 returns:

Object {bar: "hello"}

foo2 returns:

undefined

21. What will the code below output to the console and why?

var arr1 = "john".split('');

var arr2 = arr1.reverse();

var arr3 = "jones".split('');

arr2.push(arr3);

console.log("array 1: length=" + arr1.length + " last=" + arr1.slice(-1));

console.log("array 2: length=" + arr2.length + " last=" + arr2.slice(-1));

Output

"array 1: length=5 last=j,o,n,e,s"

"array 2: length=5 last=j,o,n,e,s"

1. SetTimeout function

setTimeout(function(){ alert("Hello"); }, 3000);

23. SetInterval function

setInterval(function(){ alert("Hello"); }, 3000);

1. What will be the output of the following code:

for (var i = 0; i < 5; i++) {

setTimeout(function() { console.log(i); }, i \* 1000 );

}

1. What is javascript slice method.

<p id="demo"></p>

function myFunction() {

var str = "Hello world!";

var res = str.slice(1, 5);

document.getElementById("demo").innerHTML = res;

}

**Output**

Ello

1. What is javascript splice method.

var fruits = ["Banana", "Orange", "Apple", "Mango"];

document.getElementById("demo").innerHTML = fruits;

function myFunction() {

fruits.splice(2, 0, "Lemon", "Kiwi");

document.getElementById("demo").innerHTML = fruits;

}

**Output**

Banana,Orange,**Lemon,Kiwi**,Apple,Mango

## JavaScript Nested Functions.

function add() {

var counter = 0;

function plus() {counter += 1;}

plus();

return counter;

}

## JavaScript Closures functions.

<button type="button" onclick="myFunction()">Count!</button>

var add = (function () {

var counter = 0;

return function () {

return counter += 1;

}

})();

function myFunction(){

document.getElementById("demo").innerHTML = add();

}

1. JavaScript hoisted functions and not hoisted function.
2. Hoisted function

var x; // Declare x

x = 5; // Assign 5 to x

elem = document.getElementById("demo"); // Find an element

elem.innerHTML = x; // Display x in the element

1. Not hoisted function

var x = 5; // Initialize x

var y = 7; // Initialize y

elem = document.getElementById("demo"); // Find an element

elem.innerHTML = x + " " + y; // Display x and y

1. Get the array last value.

var a = [1,2,3,4,5,6,4,5,10];

console.log(a[a.length-1]);

**32.** var array = [1,2,3,4,5];

array[10] = "10";

array[7] = ?

Output

undefined

1. Write a JavaScript program to below output.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| 2 | 4 | 6 | 8 | 10 | 12 | 14 | 16 | 18 | 20 |
| 3 | 6 | 9 | 12 | 15 | 18 | 21 | 24 | 27 | 30 |
| 4 | 8 | 12 | 16 | 20 | 24 | 28 | 32 | 36 | 40 |
| 5 | 10 | 15 | 20 | 25 | 30 | 35 | 40 | 45 | 50 |
| 6 | 12 | 18 | 24 | 30 | 36 | 42 | 48 | 54 | 60 |
| 7 | 14 | 21 | 28 | 35 | 42 | 49 | 56 | 63 | 70 |
| 8 | 16 | 24 | 32 | 40 | 48 | 56 | 64 | 72 | 80 |
| 9 | 18 | 27 | 36 | 45 | 54 | 63 | 72 | 81 | 90 |
| 10 | 20 | 30 | 40 | 50 | 60 | 70 | 80 | 90 | 100 |

var rowCount = 10;

var columnCount = 10;

var table = document.createElement('table');

for(i=1;i<=rowCount;i++){

var trCount = document.createElement('tr');

for(j=1;j<=columnCount;j++){

var tdCount = document.createElement('td');

tdCount.innerHTML = i\*j;

trCount.appendChild(tdCount);

}

table.appendChild(trCount);

}

var bodyCotainer = document.getElementsByTagName('body')[0];

bodyCotainer.appendChild(table);