JAVASCRIPT TRAINING MODULE		
Sr.No.	**QUESTIONS**	**ANSWERS**
01.	How can you create an object in javascript ?	We can create an object in javascript using either object literals or *new* keyword . ● Object literal is list of *name : value* pair (eg, age:10) inside curly braces {}. Eg- **var employee = {first_name = "Ruchi" ; last_name = "Gedam" ; age = 24 ; height = "5ft 5 inches" };** ● *new* keyword has similar use like in java, Eg- **var employee = new Object(); employee.first_name="Ruchi"; employee.last_name="Gedam"; employee.age=24; employee.height="5ft 5 inches";**
02.	How can you create an array in javascript ?	The easiest ways to create an array and store elements in array are, ● **Empty array :** Eg- **var arr = ["Ruchi" , "Swati", "Priya" , "Anjali"];** ● **Pre-filled array :** Eg- **var arr =[];** ● **Direct Storing Elements :** Eg- **arr[0] = "Ruchi";** **arr[1] = "Priya";** **arr[2] = "Swati";** **arr[3] = "Anjali";** ● *new keyword:* Eg- **var arr = new Array("Ruchi" ,"Swati" , "Anjali" , "Priya");**
03.	How can you create a cookie using javascript ?	● Cookies are data, stored in small text files, on your computer. JavaScript can create, read, and delete cookies with the document.cookie property. Eg - **document.cookie = "username=Ruchi Gedam";** ● You can also add an expiry date (in UTC time). By default, the cookie is deleted when the browser is closed: Eg - **document.cookie = "username=Ruchi Gedam"; expires=Fri, 23 Dec 2020 12:00:00 UTC";**
04.	What is the difference between Local storage & Session storage with example?	In Local Storage :- ● Data is shared between all tabs and windows from the same origin. ● The data will not expire. It will remain even after browser restart and survive OS reboot too. In Session Storage:- ● The sessionStorage exists only within the current browser tab. Another tab with the same page will have a different session storage. ● The data survives page refresh, but not

		closing/opening the tab. (Examples are in other files.)

05.	What will be the output of the code below?
	var Y = 1;
	if (function F(){})
	{
	y += Typeof F;
	}
	console.log(y);
		This code will give an error as the span tag is used to mark or highlight or color part of the text.

06.	What is the difference between Call & Apply with example?	● The call() method takes arguments separately. ● The apply() method takes arguments as an array.
07.	How to empty an Array in JavaScript?	If we just set array.length = 0; then automatically our array will get empty.

08.	What will be the output of the following code?
	var Output = (function(x)
	{
	Delete X;
	return X;
	}
	)(0);
	console.log(output);
		delete keyword used is wrong in this code. Thats why tha output will show error.

09.	What will be the output of the following code?
	var X = { Foo : 1};
	var Output = (function()
	{
	delete X.foo;
	return X.foo;
	}
	)();
	console.log(output);
		Semantical error in this code.
		console.log(output);

10.	What will be the output of the following code?
	var Employee =
	{
	company: 'xyz'
	}
	var Emp1 =
	Object.create(employee);
	delete Emp1.company
	Console.log(emp1.company);
		Again there will be errors in this code.
		So the right code is:
		
		var Emp1 = Object.create(Employee);
		delete Emp1.company;
		console.log(Emp1.company);

11.	Name the types of function with reason? function display()	Named Function - This function has name that can be used when we call that function. We use function keyword to define.

	{
	document.writeln("Named Function");
	}
	display();	

12.	Name the types of function with reason?
	var display=function()
	{
	document.writeln("Anonymous Function");
	}
	display();
		An anonymous function is a function without a name. An anonymous function is often not accessible after its initial creation.
		If we need to call the anonymous function later, we assign the function to the show method.

13.	If we want to return the character from a specific index which method is used, explain with example?	charAt(int **index**) **method** returns the **character** at the **specified index** in a string.