_____

WINTER – 2022 EXAMINATION

Subject Name: Client-Side Scripting   **Model Ans**   **Subject Code:**   22519

Important Instructions to examiners:

1) The answers should be examined by key words and not as word-to-word as given in the model answer scheme.
2) The model answer and the answer written by candidate may vary but the examiner may try to assess the understanding level of the candidate.
3) The language errors such as grammatical, spelling errors should not be given more Importance (Not applicable for subject English and Communication Skills.
4) While assessing figures, examiner may give credit for principal components indicated in the figure. The figures drawn by candidate and model answer may vary. The examiner may give credit for any equivalent figure drawn.
5) Credits may be given step wise for numerical problems. In some cases, the assumed constant values may vary and there may be some difference in the candidate's answers and model answer.
6) In case of some questions credit may be given by judgement on part of examiner of relevant answer based on candidate's understanding.
7) For programming language papers, credit may be given to any other program based on equivalent concept.
8) As per the policy decision of Maharashtra State Government, teaching in English/Marathi and Bilingual (English + Marathi) medium is introduced at first year of AICTE diploma Programme from academic year 2021-2022. Hence if the students in first year (first and second semesters) write answers in Marathi or bilingual language (English +Marathi), the Examiner shall consider the same and assess the answer based on matching of concepts with model answer.

Q. No.	Sub Q. N.	Answer	Marking Scheme
1		**Attempt any <u>FIVE</u> of the following:**	**10 M**
	a)	**State the use of method in javascript with the help of suitable example.**	**2 M**

	Ans	A method/function is a set of statements that take inputs, do some specific computation, and produce output. The idea is to put some commonly or repeatedly done tasks together and make a function so that instead of writing the same code again and again for different inputs, we can call that function.
		
		 Example:
		 function Addition (number1, number2)
		 {
		    return number1 + number2;
		 }
			Explanation - 1 M and Example-1 M.

	b)	**List & Explain datatypes in JavaScript.**	**2 M**
	Ans	JavaScript provides different data types to hold different types of values. There are two types of data types in JavaScript, Primitive data type and Non-primitive data type **i) There are five types of primitive data types in JavaScript. They are as follows:** **String -** represents sequence of characters e.g., "hello" **Number -** represents numeric values e.g., 100	Any four, ½ for each

_____

		Boolean - represents boolean value either false or true **Undefined -** represents undefined value **Null -** represents null i.e., no value at all **ii) The non-primitive data types are as follows:** **Object -** represents instance through which we can access members **Array -** represents group of similar values **RegExp -** represents regular expression	
	c)	**Write a simple calculator program using switch case in JavaScript.**	**2 M**

	Ans	```html
		<html>
		 <body>
		  <script>
		   const number1 = parseFloat(prompt("Enter first number: "));
		   const number2 = parseFloat(prompt("Enter second number: "));
		
		   const operator = prompt("Enter operator (either +, -, *, / or %): ");
		   let result;
		
		   switch (operator) {
		    case "+":
		     result = number1 + number2;
		     document.write(result);
		     break;
		    case "-":
		     result = number1 - number2;
		     document.write(result);
		     break;
		
		    case "*":
		     result = number1 * number2;
		     document.write(result);
		     break;
		
		    case "/":
		     result = number1 / number2;
		     document.write(result);
		     break;
		
		    case "%":
		     result = number1 % number2;
		     document.write(result);
		     break;
		``
			2 M for relevant program.

_____

| | | ```
        default:
         document.write("Invalid operator");
         break;
       }
     </script>
    </body>
   </html>
``` | |
| | **d)** | **Write a program using sort method of array object.** | **2 M** |
| | **Ans** | ```
<html>
<body>
  <script>
    var array =[5,1,9,7,5];
    // sorting the array
    sorted = array.sort();
    document.write(sorted);
  </script>
</body>
</html>
``` | 2 M for relevant program. |
| | **e)** | **Describe property Getters & Setters.** | **2 M** |
| | **Ans** | JavaScript object accessors are used to access and update the objects. Getter and setter are used as object accessors to get or set object properties.<br><br>Getter method helps in accessing the object methods as object properties.<br><br>Setter method is used to set object properties.<br><br>Using getter and setter the javascript provides better data security and data quality.<br><br>**Example:**<br><br>```
<!DOCTYPE html>
<html>
  <body>
    <script>
      var car = {
         brand: "Toyota",
         color: "Blue",

         get getBrand () {
           return this.brand;
         },
         get getColor () {
           return this.color;
``` | 1 m for each |

		``
		            },
		
		            set setBrand (newBrand) {
		              this.brand = newBrand;
		            },
		            set setColor (newColor) {
		              this.color = newColor;
		            }
		          };
		
		          document.write("Car Brand: " + car.brand + "
		Car Color: " + car.color);
		
		          car.setBrand = "Tesla";
		          car.setColor = "Red";
		          document.write("
		
		Car Brand: " + car.brand + "
		Car Color: " +
		car.color);
		        </script>
		      </body>
		</html>
		``	

	f)	**Enlist & explain the use of any two Intrinsic JavaScript Functions.**	**2 M**
	Ans	An intrinsic function (or built-in function) is a function (subroutine) available for use in a given programming language whose implementation is handled specially by the compiler. You can use intrinsic functions to make reference to a data item whose value is derived automatically during execution. abs() - The ABS function returns the absolute value of the argument. sin() - The SIN function returns a numeric value that approximates the sine of the angle or arc specified by the argument in radians. sqrt() - The SQRT function returns a numeric value that approximates the square root of the argument specified. Date(): return current date. Len(): returns number of characters in the text. parseInt() - parseInt() function takes string as a parameter and converts it to integer. parseFloat() - parseFloat() function takes a string as parameter and parses it to a floating point number.	1 M for each function
	g)	**Describe browser location object.**	**2 M**

	Ans	i) The location object contains information about the current URL.
		 ii) The location object is a property of the window object.
		 iii) The location object is accessed with: window.location or just location.
		 Example:
		 <!DOCTYPE html>
		 <html>
		 <body>
		    <h1>The Window Location Object</h1>
		    <p id="demo"></p>
		 <script>
		    let origin = window.location.origin;
		    document.getElementById("demo").innerHTML = origin;
		 </script>
		 </body>
		 </html>
			Explanation 1 M
			
			 Example- 1 M

2.		**Attempt any <u>THREE</u> of the following:**	**12 M**
	a)	**Write a JavaScript program that will display current date in DD/MM/YYYY format.**	**4 M**

	Ans	<!DOCTYPE *html*>
		
		 <html *lang*="en">
		
		 <head>
		
		    <meta *charset*="UTF-8">
		
		    <meta *http-equiv*="X-UA-Compatible" *content*="IE=edge">
		
		    <meta *name*="viewport" *content*="width=device-width, initial-scale=1.0">
		
		    <title>Document</title>
		
		 </head>
		
		 <body>
		
		    <script>
		
		    var d=new Date();
		
		    var currentDate=d.getDate()+'/'+(d.getMonth()+1)+'/'+d.getFullYear()
		
		    document.write(currentDate)
			Any relevant code 4 M.

|  |  | ```
</script>

</body>

</html>
``` |  |
| | **b)** | **Write a JavaScript program that will remove the duplicate element from an array.** | **4 M** |
|  | **Ans** | ```
<!DOCTYPE html>
<html lang="en">
 <body>
  <script>
   let arr = ["scale", "happy", "strength", "peace", "happy", "happy"];
   function removeDuplicates(arr) {
    let unique = [];
    for (i = 0; i < arr.length; i++) {
     if (unique.indexOf(arr[i]) === -1) {
      unique.push(arr[i]);
     }
    }
    return unique;
   }
   document.write(removeDuplicates(arr));
  </script>
 </body>
</html>
``` | Any relevant code 4 M. |
| | **c)** | **Write a JavaScript program that will display list of student in ascending order according to the marks & calculate the average performance of the class.** | **4 M** |

Student Name	Marks
Amit	70
Sumit	78
Abhishek	71

|  | **Ans** | ```
<html>

 <body>

  <script>

   var students = [["Amit", 70],["Sumit", 78],["Abhishek", 71],];

   var Avgmarks = 0;

   for (var i = 0; i < students.length; i++) {
``` | Any relevant code 4 M. |

		Avgmarks += students[i][1];
		
		  for (var j = i + 1; j < students.length; j++) {
		
		   if (students[i] > students[j]) {
		
		    a = students[i];
		
		    students[i] = students[j];
		
		    students[j] = a
		
		   }
		
		  }
		
		 }
		
		 var avg = Avgmarks / students.length;
		
		 document.write("Average grade: " + Avgmarks / students.length);
		
		 document.write("
		
		");
		
		 for (i = 0; i < students.length; ++i){
		
		  document.write(students[i]+"
		")
		
		 }
		
		 </script>
		
		 </body>
		
		</html>	

	d)	**Write and explain a string functions for converting string to number and number to string.**	**4 M**

	Ans	**To covert string to number** we can use parseInt() which converts a string number to a integer number. Similarly we can use parseFloat(), number() for converting string to number.
		
		Eg-
		var a=prompt('Enter a number');
		var b=parseInt(prompt('Enter a number'));
		document.write(typeof a+"
		");
		document.write(typeof b);
		To convert form number to string we can use toString()
		<html>
			Any relevant code with explanation 4 M.

_____

```
<body>
<p>toString() returns a number as a string:</p>
<script>
let num = 12;
let text = num.toString();
document.write(num)
</script>
</body>
</html>
```

3.		Attempt any __THREE__ of the following:	12 M
	a)	**Differentiate between concat( ) & join( ) methods of array object.**	4 M

Ans			Any 4 point=4M

concat()	join()
The concat() method concatenates (joins) two or more arrays. The concat() method returns a new array, containing the joined arrays.	The join() method returns an array as a string.
The concat() method separates each value with a comma only.	Any separator can be specified. The default is comma (,).
Syntax: *array1*.concat(*array2*, *array3*, ..., *arrayX*)	Syntax: *array*.join(*separator*)
Example: `<script>` `const arr1 = ["CO", "IF"];` `const arr2 = ["CM", "AI",4];` `const arr = arr1.concat(arr1, arr2);` `document.write(arr);` `</script>`	Example: `<script>` `var fruits = ["Banana", "Orange", "Apple", "Mango"];` `var text = fruits.join();` `document.write(text);` `var text1 = fruits.join("$$");` `document.write(" "+text1);` `</script>`

	b)	**Write a JavaScript function to check the first character of a string is uppercase or not.**	4 M

Ans		

```
<html>
<body>
<script>
function upper_case(str)
{
  regexp = /^[A-Z]/;
  if (regexp.test(str))
   {
    document.write("String's first character is uppercase");
```

Correct function logic=4M

(any other relevant

_____

| | ```
    }
   else
   {
    document.write("String's first character is not uppercase");
   }
 }
upper_case('Abcd');
</script>
</body>
</html>


                              **OR**
<script>
function firstIsUppercase(str)
{
  if (str.length === 0)
  {
    return false;
  }

  return str.charAt(0).toUpperCase() === str.charAt(0);
}

if (firstIsUppercase(prompt("Enter text")))
{
  document.write('First letter is uppercase');
} else {
  document.write('First letter is NOT uppercase');
}
</script>
``` | logic can consider) |
| **c)** | **Write a JavaScript function to merge two array & removes all duplicate values.** | **4 M** |
| **Ans** | ```
<html>
<body>
<script>
function merge_array(array1, array2)
  {
   var result_array = [];
   var arr = array1.concat(array2);
   var len = arr.length;
   var assoc = {};
``` | Correct function logic=4 M<br><br>(any other relevant logic can consider) |

_____

```
        while(len--)
         {
           var item = arr[len];

           if(!assoc[item])
           {
             result_array.unshift(item);
             assoc[item] = true;
           }
         }

         return result_array;
       }
var array1 = [1, 2, 3,4,7,9];
var array2 = [2, 30, 1,40,9];
document.write(merge_array(array1, array2));
</script>
</body>
</html>
```

Output:
3,4,7,2,30,1,40,9

OR

```
<html>
<body>
<script>
function mergearr(arr1, arr2)
 {
   // merge two arrays
   var arr = arr1.concat(arr2);
   var uniqueArr = [];
   // loop through array
   for(var i of arr) {
     if(uniqueArr.indexOf(i) === -1)
     {
       uniqueArr.push(i);
     }
   }
   document.write(uniqueArr);
}
```

		var array1 = [1, 2, 3,6,8];
		var array2 = [2, 3, 5,56,78,3]
		mergearr(array1, array2);
		</script>
		</body>
		</html>
		
		Output:
		1,2,3,6,8,5,56,78	

	d)	**Write a JavaScript function that will open new window when the user will clicks on the button.**	**4 M**

	Ans	<html>
		<body>
		<button onclick="openWin()">Open "New Window"</button>
		<script>
		 var myWindow;
		 function openWin()
		 {
		myWindow = window.open("", "myWindow", "width=400,height=400");
		 myWindow.document.write(" Hello Everyone.Welcome to new window. ");
		 }
		</script>
		</body>
		</html>
			Correct function logic=4 M

4.		**Attempt any THREE of the following:**	**12 M**
	a)	**Describe text Rollover with the help of example.**	**4 M**
	Ans	**Rollover means** a webpage changes when the user moves his or her mouse over an object on the page. It is often used in advertising. There are two ways to create rollover, using plain HTML or using a mixture of JavaScript and HTML. We will demonstrate the creation of rollovers using both methods. The keyword that is used to create rollover is the <onmousover> event. For example, we want to create a rollover text that appears in a text area. The text *"What is rollover?"* appears when the user place his or her mouse over the text area and the rollover text changes to *"Rollover means a webpage changes when the user moves his or her mouse over an object on the page"* when the user moves his or her mouse away from the text area.	Define Rollover-2 M Example-2 M (For example, any other relevant

		The HTML script is shown in the following example: Example: <html> <head></head> <Body> <textarea rows="2" cols="50" name="rollovertext" onmouseover="this.value='What is rollover?'" onmouseout="this.value='Rollover means a webpage changes when the user moves his or her mouse over an object on the page'"></textarea> </body> </html>	logic can be considered)
	b)	**Write a JavaScript program that will create pull-down menu with three options. Once the user will select the one of the options then user will redirected to that website.**	**4 M**

	Ans	```html
		<html>
		<head>
		<title>HTML Form</title>
		<script language="javascript" type="text/javascript">
		function getPage(choice)
		{
		page=choice.options[choice.selectedIndex].value;
		if(page != "")
		{
		window.location=page;
		}
		}
		</script>
		</head>
		<body>
		<form name="myform" action="" method="post">
		 Select Your Favourite Website:
		 <select name="MenuChoice" onchange="getPage(this)">
		 <option value="select any option">Select</option>
		<option value="https://www.codecademy.com/catalog/language/javascript/">
		CodeAcademy </option>
		<option value="https://www.msbte.org.in">MSBTE</option>
		<option value="https://www.javatpoint.com/javascript-tutorial">JavaTpoint</option>
		</form>
		</body>
		</html>```
			Creation of pull-down menus-1 M
			
			Correct function to redirect particular website-3 M
			
			OR
			
			any other relevant logic can be considered

Output:

Select Your Favourite Website: Select ▾

Select
CodeAcademy
MSBTE
JavaTpoint

c)		**Describe Quantifiers with the help of example.**	**4 M**
	Ans	The frequency or position of bracketed character sequences and single characters can be denoted by a special character. Each special character has a specific connotation. The +, *, ?, and $ flags all follow a character sequence.	Describe Quantifiers- 2 M

Sr.No.	Expression & Description
1	**p+** It matches any string containing one or more p's.
2	**p*** It matches any string containing zero or more p's.
3	**p?** It matches any string containing at most one p.(zero or one occurrences)
4	**p{N}** It matches any string containing a sequence of **N** p's
5	**p{2,3}** It matches any string containing a sequence of two or three p's.
6	**p{2, }** It matches any string containing a sequence of at least two p's.
7	**p$** It matches any string with p at the end of it.

For Example-2 M

Example:

```
<html>
<body>
<button onclick="myFunction()">Try it</button>
<p id="demo"></p>
<script>
function myFunction()
 {
  var str = "100, 1000 or 10000?";
  var patt1 = /\d{3,4}/g;
```

| | | ```
var result = str.match(patt1);
 document.getElementById("demo").innerHTML = result;
}
</script>
 </body>
 </html>
``` | |
| | **d)** | **Describe frameworks of JavaScript & its application.** | **4 M** |
| | **Ans** | **Frameworks of JavaScript:**
1. ReactJs
React is based on a reusable component. Simply put, these are code blocks that can be classified as either classes or functions. Each component represents a specific part of a page, such as a logo, a button, or an input box. The parameters they use are called props, which stands for properties.

Applications:
React is a JavaScript library developed by Facebook which, among other things, was used to build Instagram.com.

2. Angular
Google operates this framework and is designed to use it to develop a Single Page Application (SPA). This development framework is known primarily because it gives developers the best conditions to combine JavaScript with HTML and CSS. Google operates this framework and is designed to use it to develop a Single Page Application (SPA). This development framework is known primarily because it gives developers the best conditions to combine JavaScript with HTML and CSS.

Applications:
Microsoft Office ,Gmail, Forbes, PayPal, Grasshopper, Samsung, Delta

3. Vue.js
Vue is an open-source JavaScript framework for creating a creative UI. The integration with Vue in projects using other JavaScript libraries is simplified because it is designed to be adaptable.

Application:
VueJS is primarily used **to build web interfaces and one-page applications**. It can also be applied to both desktop and mobile app development.

4. jQuery
It is a cross-platform JavaScript library designed to simplify HTML client-side scripting. You can use the jQuery API to handle, animate, and manipulate an event in an HTML document, also known as DOM. Also, jQuery is used with Angular and React App building tools.
Applications:
JQuery can be used to develop Ajax based applications.
It can be used to make code simple, concise and reusable.
It simplifies the process of traversal of HTML DOM tree.
It can also handle events, perform animation and add ajax support in web applications. | Any 2

(1m for explanation and 1M for application) |

		5. Node.js Node.js is an open-source, server-side platform built on the Google Chrome JavaScript Engine. Node.js is an asynchronous, single-threaded, non-blocking I/O model that makes it lightweight and efficient. **Applications:** Paypal, LinkedIn, Yahoo, Mozilla, Netflix, Uber, Groupon, GoDaddy, eBay	
	e)	**Describe how to link banner advertisement to URL with example.**	**4 M**

	Ans	The banner advertisement is the hallmark of every commercial web page. It is typically positioned near the top of the web page, and its purpose is to get the visitor's attention by doing all sorts of clever things.
		To get additional information, the visitor is expected to click the banner so that a new web page opens. You can link a banner advertisement to a web page by inserting a hyperlink into your web page that calls a JavaScript function rather than the URL of a web page. The JavaScript then determines the URL that is associated with the current banner and loads the web page that is associated with the URL.
		
		Example:
		`<html>`
		`<head>`
		`<title>Link Banner Ads</title>`
		`<script language="Javascript" type="text/javascript">`
		`Banners = new Array('1.jpg','2.jpg','3.jpg')`
		`BannerLink = new Array(`
		`'google.com/','vpt.edu.in/', 'msbte.org.in/');`
		`CurrentBanner = 0;`
		`NumOfBanners = Banners.length;`
		`function LinkBanner()`
		`{`
		`document.location.href =`
		`"http://www." + BannerLink[CurrentBanner];`
		`}`
		`function DisplayBanners() {`
		`if (document.images) {`
		`CurrentBanner++`
		`if (CurrentBanner == NumOfBanners) {`
		`CurrentBanner = 0`
		`}`
		`document.RotateBanner.src= Banners[CurrentBanner]`
		`setTimeout("DisplayBanners()",1000)`
		`}`
		`}`
		`</script>`
		`</head>`
		`<body onload="DisplayBanners()" >`
		`<center>`
		`<a href="javascript: LinkBanner()"><img src="1.jpg"`
			Banner-1 M
			Example-3 M

		width="400" height="75" name="RotateBanner" /> </center> </body> </html>	
5.		**Attempt any TWO of the following:**	**12 M**
	a)	**Write HTML script that will display following structure**	**6 M**

Write the JavaScript code for below operations:
(1) Name, Email & Pin Code should not be blank.
(2) Pin Code must contain 6 digits & it should not be accept any characters.

| | **Ans** | ```html
<html>
<head>
<style>
table,tr,td
{
border: solid black 1px;
border-collapse: collapse;
}

td
{
padding: 10px;
}
</style>
</head>
<body>
<table>
<tbody>
<tr>
<td>Name : </td>
<td> <input type="text" id="name" required></td>
</tr>
<tr>
<td>Email : </td>
<td> <input type="email" id="email" required></td>
</tr>
<tr>
``` | Creation of correct form and calling event-2 M

Name, Email and Pin code should not be blank-2 M

Pin code must contain 6 digits and it should not be blank-2 M |

_____

| | | ```
<td>Pin code : </td>
<td> <input type="number" id="pin" required></td>
</tr>
<tr>
<td></td>
<td><button onclick="submit()">Submit</button></td>
</tr>
</tbody>
</table>
</body>
<script>
function submit()
    {
       var name = document.getElementById("name").value;
       var email = document.getElementById("email").value;
       var pin = Number(document.getElementById("pin").value);
       if(name.length==0 || email.length==0 || pin.length==0)
       {
          alert("Please enter value in all fields.")
       }
       else
       {
          var pinpattern =/^[4]{1}[0-9]{5}$/;
          if( pinpattern.test(pin))
          {
             alert("Perfect Pin code");
          }
          else
          {
             alert("Wrong Pin code.");
          }
       }
    }
</script>
</html>
``` | |
| | **b)** | **Write a webpage that displays a form that contains an input for user name and password. User is prompted to enter the input user name and password and password becomes the value of the cookie. Write the JavaScript function for storing the cookies. It gets executed when the password changes.** | **6 M** |
| | **Ans** | ```
<html>
<head>
<script>
 function storeCookie()
 {
  var pwd = document.getElementById('pwd').value
  document.cookie = "Password=" + pwd + ";"
``` | Creation of form=2 M<br><br>Storing and display cookie information- 4 M |

_____

		alert("Cookie Stored\n"+document.cookie);
		 }
		 </script>
		 </head>
		 <body>
		 <form name="myForm">
		 Enter Username <input type="text" id="uname"/>
		
		 Enter Password <input type="password" id="pwd"/>
		
		 <input type="button" value="Submit" onclick="storeCookie()"/>
		 <p id="panel"></p>
		 </form>
		 </body>
		 </html>	

	c)	**Write a JavaScript for creating following frame structure:** **Chapter 1 and Chapter 2 are linked to the webpage Ch1.html and ch2.html respectively. When user click on these links corresponding data appears in FRAME3.**	**6 M**
	Ans	**Step 1) create file frame1.html** <html> <body> FRAME1 </body> </html> **Step 2) create frame2.html** <html> <head> <title>FRAME 2</title> </head> <body><H1>Operating System</H1> <UL>Chapter 1</UL> <UL> Chapter 2</UL> </body> </html>	Correct frameset logic=6 M OR (any other relevant logic can be considered)

Step 3) create frame3.html
```
<html>
<body>
<h1>FRAME3</h1>
</body>
</html>
```

Step4) create frame_target.html
```
<html>
<head>
<title>Create a Frame</title>
</head>
<frameset rows="30%,*" border="1">
<frame src="frame1.html" name="a" />
<frameset cols="50%,*" border="1">
<frame src="frame2.html" name="b" />
<frame src="frame3.html" name="c" />
</frameset>
</frameset>
</html>
```

Output:

FRAME1

Operating System

Chapter 1

Chapter 2

```
History Of OS
Operating systems were first developed in the
late 1950s to manage tape storage
The General Motors Research Lab implemented
the first OS in the early 1950s for their IBM
701
In the mid-1960s, operating systems started
to use disks
In the late 1960s, the first version of the
Unix OS was developed
The first OS built by Microsoft was DOS. It
was built in 1981 by purchasing the 86-DOS
software from a Seattle company
The present-day popular OS Windows first came
to existence in 1985 when a GUI was created
and paired with MS-DOS.
```

6.		Attempt any **TWO** of the following:	12 M
	a)	Write HTML script that will display dropdown list containing options such as Red, Green, Blue and Yellow. Write a JavaScript program such that when the user selects any options. It will change the background colour of webpage.	6 M

_____

| **Ans** | ```html
<html>
<body>
<label for="color">Choose a Background Color:</label>
<select name="color" id="color" class="color" onchange="changeColor()">
<option value="red">Red</option>
<option value="green">Green</option>
<option value="blue">Blue</option>
<option value="yellow">Yellow</option>
</select>
<script type="text/javascript">
function changeColor() {
   var color = document.getElementById("color").value;
   switch(color){
      case "green":
         document.body.style.backgroundColor = "green";
         break;
      case "red":
         document.body.style.backgroundColor = "red";
          break;
      case "blue":
         document.body.style.backgroundColor = "blue";
         break;
      case "yellow":
         document.body.style.backgroundColor = "yellow";
         break;
      default:
         document.body.style.backgroundColor = "white";
         break;
   }
}
</script>
</body>
</html>
``` | | Creation of list-2 M

Correct logic to Change background color as per selection-4 M |
| **b)** | **Develop a JavaScript program to create Rotating Banner Ads.** | | **6 M** |
| **Ans** | ```html
<html >
<head>
<title>Banner Ads</title>
<script>
Banners = new Array('1.jpg','2.jpg','3.jpg');
CurrentBanner = 0;
function DisplayBanners()
{
if (document.images);
 {
CurrentBanner++;
if (CurrentBanner == Banners.length)
 {
CurrentBanner = 0;
``` | | Correct logic-6 M

OR
(any other relevant logic can be considered) |

| | | ```
}
document.RotateBanner.src= Banners[CurrentBanner];
setTimeout("DisplayBanners()",1000);
}
}
</script>
</head>
<body onload="DisplayBanners()" >
<center>
<img src="1.jpg" width="400"
height="75" name="RotateBanner" />
</center>
</body>
</html>
``` | |
| | **c)** | **Write a JavaScript for the folding tree menu.** | **6 M** |
| | **Ans** | ```
<html>
<head>
<style>
ul, #myUL {
  list-style-type: none;
}
.caret::before {
  content: "\25B6";
  color: black;
  display: inline-block;
  margin-right: 6px;
}

.caret-down::before {
 -ms-transform: rotate(90deg); /* IE 9 */
 -webkit-transform: rotate(90deg); /* Safari */'
  transform: rotate(90deg);
}

.nested {
  display: none;
}

.active {
  display: block;
}
</style>
</head>
<body>

<h2>Folding Tree Menu</h2>
<p>A tree menu represents a hierarchical view of information, where each item can have a
number of subitems.</p>
``` | Correct logic-6 M

OR
(any other relevant logic can be considered) |

```
<p>Click on the arrow(s) to open or close the tree branches.</p>

<ul id="myUL">
  <li><span class="caret">India</span>
    <ul class="nested">
      <li>Karnataka</li>
      <li>Tamilnaadu</li>
      <li><span class="caret">Maharashtra</span>
        <ul class="nested">
          <li>Mumbai</li>
          <li>Pune</li>
          <li><span class="caret">Navi Mumbai</span>
            <ul class="nested">
              <li>Nerul</li>
              <li>Vashi</li>
              <li>Panvel</li>

            </ul>
          </li>
        </ul>
      </li>
    </ul>
  </li>
</ul>
<script>
var toggler = document.getElementsByClassName("caret");
var i;
for (i = 0; i < toggler.length; i++) {
  toggler[i].addEventListener("click", function() {
    this.parentElement.querySelector(".nested").classList.toggle("active");
    this.classList.toggle("caret-down");
  });
}
</script>
</body>
</html>
```