Assignment-2

Q.1 Write a JavaScript program to find all the index positions of a given word within a given string.

```
function findAllIndexes(str, word) {
	let indexes = [];
	let index = -1;

	while ((index = str.indexOf(word, index + 1)) !== -1) {
	indexes.push(index);
	}

	return indexes;
}

let string = "hello world hello";
let word = "world";
let positions = findAllIndexes(string, word);
console.log("The word '" + word + "' appears at positions: " + positions.join(", "));
```

Q.2 Write a JavaScript program to find the first index of a given element in an array using the linear search algorithm.

```
function linearSearch(arr, element) {

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

if (arr[i] === element) {

return i;
}
}
return -1;
}
```

Q.3 Write a JavaScript program to sort a list of elements using Quick sort.

```
function quickSort(arr) {
	if (arr.length <= 1) {
	return arr;
	}

	const pivot = arr[Math.floor(arr.length / 2)];
	const left = [];
	const right = [];
```

```
        for (let i = 0; i < arr.length; i++) {
        if (i === Math.floor(arr.length / 2)) {
        continue;
        }
        if (arr[i] < pivot) {
        left.push(arr[i]);
        } else {
        right.push(arr[i]);
        }
        }

        return [...quickSort(left), pivot, ...quickSort(right)];
}

// Example usage:
const arr = [5, 3, 7, 2, 8, 4, 1];
const sortedArr = quickSort(arr);
console.log("Sorted array:", sortedArr);
```

Q.4 Write a JavaScript program to sort a list of elements using Merge sort.

```
function merge_Arrays(left_sub_array, right_sub_array) {
  let array = []
  while (left_sub_array.length && right_sub_array.length) {
        if (left_sub_array[0] < right_sub_array[0]) {
        array.push(left_sub_array.shift())
        } else {
        array.push(right_sub_array.shift())
        }
  }
  return [ ...array, ...left_sub_array, ...right_sub_array ]
}
function merge_sort(unsorted_Array) {
  const middle_index = unsorted_Array.length / 2
  if(unsorted_Array.length < 2) {
        return unsorted_Array
  }
  const left_sub_array = unsorted_Array.splice(0, middle_index)
  return merge_Arrays(merge_sort(left_sub_array),merge_sort(unsorted_Array))
}
unsorted_Array = [39, 28, 44, 4, 10, 83, 11];
console.log("The sorted array will be: ",merge_sort(unsorted_Array));
```

Q.5 Write a JavaScript program to sort a list of elements using Heap sort.

```
function customSort(arr) {
	var N = arr.length;

	for (var i = Math.floor(N / 2) - 1; i >= 0; i--)
	customHeapify(arr, N, i);

	for (var i = N - 1; i > 0; i--) {
	var temp = arr[0];
	arr[0] = arr[i];
	arr[i] = temp;
	customHeapify(arr, i, 0);
	}
}

function customHeapify(arr, N, i) {
	var largest = i;
	var l = 2 * i + 1;
	var r = 2 * i + 2;

	if (l < N && arr[l] > arr[largest])
	largest = l;

	if (r < N && arr[r] > arr[largest])
	largest = r;

	if (largest != i) {
	var swap = arr[i];
	arr[i] = arr[largest];
	arr[largest] = swap;
	customHeapify(arr, N, largest);
	}
}

function customPrintArray(arr) {
	var N = arr.length;
	for (var i = 0; i < N; ++i)
	console.log(arr[i]);
}

var originalArr = [12, 11, 13, 5, 6, 7];
customSort(originalArr);
console.log("Sorted array:" + originalArr);
```

Q.6 Write a JavaScript program to sort a list of elements using Insertion sort.

```
function insertionSort(arr) {
	const n = arr.length;

	for (let i = 1; i < n; i++) {
	let key = arr[i];
	let j = i - 1;

	while (j >= 0 && arr[j] > key) {
	arr[j + 1] = arr[j];
	j--;
	}

	arr[j + 1] = key;
	}

	return arr;
}

// Example usage:
const originalArr = [5, 3, 7, 2, 8, 4, 1];
const sortedArr = insertionSort(originalArr);
console.log("Sorted array:", sortedArr);
```

Q.7 Write a JavaScript program to sort a list of elements using Bubble sort

```
function bubbleSort(arr) {
	const n = arr.length;

	for (let i = 0; i < n - 1; i++) {
	for (let j = 0; j < n - i - 1; j++) {
	if (arr[j] > arr[j + 1]) {
		// Swap arr[j] and arr[j+1]
		let temp = arr[j];
		arr[j] = arr[j + 1];
		arr[j + 1] = temp;
	}
	}
	}

	return arr;
}

// Example usage:
```

```
const arr = [5, 3, 7, 2, 8, 4, 1];
const sortedArr = bubbleSort(arr);
console.log("Sorted array:", sortedArr);
```

Q.8 Write a JavaScript program to sort the characters in a string alphabetically.

```
function sortStringAlphabetically(str) {
        return str.split('').sort().join('');
}

const inputString = "hello world";
const sortedString = sortStringAlphabetically(inputString);
console.log("Original string:", inputString);
console.log("Sorted string:", sortedString);
```

Q.9 Write a JavaScript program to check if a numeric array is sorted or not.

```
function isArraySorted(arr) {
        for (let i = 0; i < arr.length - 1; i++) {
        if (arr[i] > arr[i + 1]) {
        return false;
        }
        }
        return true;
}

// Example usage:
const sortedArray = [1, 2, 3, 4, 5];
const unsortedArray = [5, 3, 7, 2, 8];
console.log("Is sortedArray sorted?", isArraySorted(sortedArray));
console.log("Is unsortedArray sorted?", isArraySorted(unsortedArray));
```

Q.10 Write a JavaScript function to validate whether a given value type is null or not.

```
function isNull(value) {
   return value === null;
}

// Example usage:
console.log(isNull(null)); // true
console.log(isNull(5));    // false
```

Q.11 Write a JavaScript function to validate whether a given value is a number or not.

```
function isNumber(value) {
   return typeof value === 'number' && !isNaN(value);
}

// Example usage:
console.log(isNumber(5));      // true
console.log(isNumber("hello")); // false
```

Q.12 Write a JavaScript function to validate whether a given value is RegExp or not.

```
function isRegExp(value) {
   return Object.prototype.toString.call(value) === '[object RegExp]';
}

// Example usage:
console.log(isRegExp(/test/)); // true
console.log(isRegExp("hello")); // false
```

Q.13 Write a JavaScript program to delete the rollno property from the following object. Also print the object before or after deleting the property.

Sample object:

```
var student = {
name : "David Rayy",
sclass : "VI",
rollno : 12 };
```

```
var student = {
        name: "David Rayy",
        sclass: "VI",
        rollno: 12
};

console.log("Object before deleting rollno property:", student);
```

```
delete student.rollno;

console.log("Object after deleting rollno property:", student);
```

Q.14 Write a JavaScript program to display the reading status (i.e. display book name, author name and reading status) of the following books.

```
var library = [
 {
 author: 'Bill Gates',
 title: 'The Road Ahead',
 readingStatus: true
 },
 {
 author: 'Steve Jobs',
 title: 'Walter Isaacson',
 readingStatus: true
 },
 {
 author: 'Suzanne Collins',
 title: 'Mockingjay: The Final Book of The Hunger Games',
 readingStatus: false
}];
```

```
var library = [
        {
        author: 'Bill Gates',
        title: 'The Road Ahead',
        readingStatus: true
        },
        {
        author: 'Steve Jobs',
        title: 'Walter Isaacson',
        readingStatus: true
        },
        {
        author: 'Suzanne Collins',
        title: 'Mockingjay: The Final Book of The Hunger Games',
        readingStatus: false
        }
];

for (var i = 0; i < library.length; i++) {
        var book = library[i];
```

```
        var bookInfo = "'" + book.title + "' by " + book.author;

        if (book.readingStatus) {
        console.log("You have already read " + bookInfo + ".");
        } else {
        console.log("You haven't read " + bookInfo + " yet.");
        }
}
```

Q.15 Write a JavaScript program to create a clock.
Note: The output will come every second.
Expected Console Output :
"14:37:42"
"14:37:43"
"14:37:44"
"14:37:45"
"14:37:46"
"14:37:47"

```
function displayTime() {
        var date = new Date();
        var hours = formatTime(date.getHours());
        var minutes = formatTime(date.getMinutes());
        var seconds = formatTime(date.getSeconds());
        console.log(hours + ":" + minutes + ":" + seconds);
}

function formatTime(time) {
        return time < 10 ? "0" + time : time;
}

setInterval(displayTime, 1000);
```

Q.16 Write a JavaScript function to parse an URL.

```
function parseURL(url) {
        var parser = document.createElement('a');
        parser.href = url;

        return {
        protocol: parser.protocol,
        hostname: parser.hostname,
        port: parser.port,
        pathname: parser.pathname,
        search: parser.search,
```

```
        hash: parser.hash,
        origin: parser.origin
        };
}

// Example usage:
var url =
"https://www.example.com:8080/path/to/page?key1=value1&key2=value2#section1";
var parsedURL = parseURL(url);
console.log(parsedURL);
```

Q.17 Write a JavaScript function to split a string and convert it into an array of words

```
function splitStringIntoWords(str) {
        // Use the split method to split the string into an array of words
        return str.split(/\s+/);
}

// Example usage:
var sentence = "This is a sample sentence.";
var wordsArray = splitStringIntoWords(sentence);
console.log(wordsArray);
```

Q.18 Write a JavaScript function that takes a string with both lowercase and upper case letters as a parameter. It converts upper case letters to lower case, and lower case letters to upper case.

```
function swapCase(str) {
        var swapped = "";
        for (var i = 0; i < str.length; i++) {
        var char = str[i];
        if (char === char.toUpperCase()) {
        swapped += char.toLowerCase();
        } else {
        swapped += char.toUpperCase();
        }
        }
        return swapped;
}

// Example usage:
var inputString = "Hello World";
var swappedString = swapCase(inputString);
console.log("Original string:", inputString);
console.log("Swapped string:", swappedString);
```

Q.19 Write a JavaScript function that returns the number of minutes in hours and minutes.

Input :
console.log(timeConvert(200));
Output :
"200 minutes = 3 hour(s) and 20 minute(s)."

```
function timeConvert(minutes) {
	var hours = Math.floor(minutes / 60);
	var remainingMinutes = minutes % 60;
	return minutes + " minutes = " + hours + " hour(s) and " + remainingMinutes + "
minute(s).";
}

// Example usage:
console.log(timeConvert(200));
```

Q.20 Write a JavaScript program to implement a stack that checks if a given element is present or not in the stack.

```
class Stack {
	constructor() {
	this.items = [];
	}

	push(element) {
	this.items.push(element);
	}

	search(element) {
	return this.items.includes(element);
	}
}

// Example usage:
var stack = new Stack();
stack.push(5);
stack.push(10);
stack.push(15);

console.log("Is 10 present in the stack?", stack.search(10)); // true
console.log("Is 20 present in the stack?", stack.search(20)); // false
```

Q.21 Write a JavaScript program to check whether a single linked list is empty or not. Return true otherwise false.

```
class Node {
        constructor(data) {
        this.data = data;
        this.next = null;
        }
}

class LinkedList {
        constructor() {
        this.head = null;
        }

        isEmpty() {
        return this.head === null;
        }
}

// Example usage:
var linkedList = new LinkedList();
console.log(linkedList.isEmpty()); // true

linkedList.head = new Node(10);
console.log(linkedList.isEmpty()); // false
```

Q.22 Write a JavaScript program to create a class called 'Rectangle' with properties for width and height. Include two methods to calculate rectangle area and perimeter. Create an instance of the 'Rectangle' class and calculate its area and perimeter.

```
class Rectangle {
        constructor(width, height) {
        this.width = width;
        this.height = height;
        }

        calculateArea() {
        return this.width * this.height;
        }

        calculatePerimeter() {
        return 2 * (this.width + this.height);
```

```
        }
}

var rectangle = new Rectangle(5, 10);

var area = rectangle.calculateArea();
var perimeter = rectangle.calculatePerimeter();

console.log("Area:", area); //50
console.log("Perimeter:", perimeter); //30
```

Q.23 Write a JavaScript program to create a slideshow that changes the displayed image when a next or previous button is clicked.

```
var images = ["image1.jpg", "image2.jpg", "image3.jpg"]; // Replace these with actual image URLs
var currentIndex = 0;

function showSlide(index) {
        var image = document.getElementById("image");
        if (index >= 0 && index < images.length) {
        image.src = images[index];
        currentIndex = index;
        }
}

function nextSlide() {
        currentIndex = (currentIndex + 1) % images.length;
        showSlide(currentIndex);
}

function prevSlide() {
        currentIndex = (currentIndex - 1 + images.length) % images.length;
        showSlide(currentIndex);
}

showSlide(currentIndex);
```

Q.24 Write a JavaScript program that uses a try-catch block to catch and handle a 'SyntaxError' when parsing an invalid JSON string.

```
var invalidJSONString = '{"name": "Jason", "age": 24,}';

try {
	var parsedObject = JSON.parse(invalidJSONString);
	console.log(parsedObject);
} catch (error) {
	if (error instanceof SyntaxError) {
	console.log("Error: Invalid JSON string.");
	console.log(error.message);
	} else {
	throw error;
	}
}
```

Q.25 Write a JavaScript program to redirect to a specified URL.

```
// Specified URL to Redirect To
var redirectURL = "https://www.example.com";

// Redirection to the Specified URL
window.location.href = redirectURL;
```

INSTITUTE OF ENGINEERING AND TECHNOLOGY, DAVV, INDORE

Name:- Ibrahim Saify

Subject:- Server Side Programming

Subject Code:- 6CERL4

Submitted To:- Mr. Aditya Makwe

Roll No:- 21C6032

Branch:- CS-A Third Year

HTML ASSIGNMENT – 1

Q1. Basic HTML Document

An element called HTML surrounds the whole document. This
element contains two sub-elements, HEAD and BODY. These elements are required to form any HTML document.

```
<html>
<Head>
<Title>The First Page</title>
</head>
<body>
Hello World
</body>
</html>
```

Just write down above code in the notepad editor and save this file with the extension of .html or .htm and then double click on that file you will get output on the default web browser.

Q2.Create a static webpage using table tags of HTML

Index.html:

```
<!DOCTYPE html>
<html lang="en">
  <head>
        <meta charset="UTF-8" />
        <meta name="viewport" content="width=device-width, initial-scale=1.0" />
        <title>Question 2</title>
        <link rel="stylesheet" href="style.css" />
  </head>
  <body>
        <div class="table-name"style="text-align: center; ">
        <h4 >Specification Table with Hours and Marks</h4>
        <table >
        <tr>
        <th rowspan="2">Unit No</th>
        <th rowspan="2" >Unit Title</th>
        <th rowspan="2">Teaching Hours</th>
```

<th colspan="4">Distribution of Theory Marks</th>
</tr>
<tr>
<td>R Level</td>
<td>U Level</td>
<td>A Level</td>
<td>Total Marks</td>
</tr>
<tr>
<td>I</td>
<td style="text-align: left;">Introduction to Internet Technology</td>
<td>2</td>
<td>4</td>
<td>4</td>
<td>0</td>
<td>8</td>
</tr>
<tr>
<td>II</td>
<td style="text-align: left;">Basics of HTML & CSS</td>
<td>6</td>
<td>0</td>
<td>2</td>
<td>6</td>
<td>8</td>
</tr>
<tr>
<td>III</td>
<td style="text-align: left;">Active Server Pages 3.0</td>
<td>6</td>
<td>4</td>
<td>8</td>
<td>0</td>
<td>12</td>
</tr>
<tr>
<td>IV</td>
<td style="text-align: left;">Server Side Coding with VBScript and XML</td>
<td>8</td>
<td>2</td>
<td>4</td>
<td>8</td>
<td>14</td>
</tr>

```
        <tr>
        <td>V</td>
        <td style="text-align: left;">ASP Objects & Components</td>
        <td>10</td>
        <td>4</td>
        <td>4</td>
        <td>6</td>
        <td>14</td>
        </tr>
        <tr>
        <td>IV</td>
        <td style="text-align: left;">Accessing database with ASP & ADO</td>
        <td>10</td>
        <td>4</td>
        <td>4</td>
        <td>6</td>
        <td>14</td>
        </tr>
        <tr>
        <th></th>
        <th>Total</th>
        <th>42</th>
        <th>18</th>
        <th>26</th>
        <th>26</th>
        <th>70</th>

        </tr>
        </table>
        </div>
  </body>
</html>
```

Q3. Create a static web page which defines all text formatting tags of HTML in tabular format

```
<h1 style="text-align:center;">Text Formatting Tags</h1>
<table style="border-collapse: collapse; width: 50%; margin: 0 auto;">
  <tr>
        <th style="background-color: #cccccc; border: 1px solid black; padding: 8px;">HTML
Tag</th>
        <th style="background-color: #cccccc; border: 1px solid black; padding:
8px;">Output</th>
```

```
  </tr>
  <tr>
        <td style="border: 1px solid black; padding: 8px;">normal text</td>
        <td style="border: 1px solid black; padding: 8px;">hello world</td>
  </tr>
  <tr>
        <td style="border: 1px solid black; padding: 8px;">Font &amp;amp; its attributes</td>
        <td style="border: 1px solid black; padding: 8px;"><font face="Arial">hello
world</font></td>
  </tr>
  <tr>
        <td style="border: 1px solid black; padding: 8px;">&lt;B&gt;</td>
        <td style="border: 1px solid black; padding: 8px;"><b>Bold</b></td>
  </tr>
  <tr>
        <td style="border: 1px solid black; padding: 8px;">&lt;I&gt;</td>
        <td style="border: 1px solid black; padding: 8px;"><i>Italic</i></td>
  </tr>
  <tr>
        <td style="border: 1px solid black; padding: 8px;">&lt;U&gt;</td>
        <td style="border: 1px solid black; padding: 8px;"><u>Underline</u></td>
  </tr>
  <tr>
        <td style="border: 1px solid black; padding: 8px;">&lt;EM&gt;</td>
        <td style="border: 1px solid black; padding: 8px;"><em>Emphasis</em></td>
  </tr>
  <tr>
        <td style="border: 1px solid black; padding: 8px;">&lt;STRONG&gt;</td>
        <td style="border: 1px solid black; padding: 8px;"><strong>STRONG</strong></td>
  </tr>
  <tr>
        <td style="border: 1px solid black; padding: 8px;">&lt;TELETYPE&gt;</td>
        <td style="border: 1px solid black; padding: 8px;"><tt>TELETYPE</tt></td>
  </tr>
  <tr>
        <td style="border: 1px solid black; padding: 8px;">&lt;CITE&gt;</td>
        <td style="border: 1px solid black; padding: 8px;"><cite>Citation</cite></td>
  </tr>
  <tr>
        <td style="border: 1px solid black; padding: 8px;">&lt;STRIKE&gt;</td>
        <td style="border: 1px solid black; padding: 8px;"><strike>strike through
text</strike></td>
  </tr>
  <tr>
```

```
        <td style="border: 1px solid black; padding: 8px;">&lt;BIG&gt;</td>
        <td style="border: 1px solid black; padding: 8px;"><big>text in a big font</big></td>
  </tr>
  <tr>
        <td style="border: 1px solid black; padding: 8px;">&lt;SMALL&gt;</td>
        <td style="border: 1px solid black; padding: 8px;"><small>text in a small
font</small></td>
  </tr>
  <tr>
        <td style="border: 1px solid black; padding: 8px;">&lt;SUB&gt;</td>
        <td style="border: 1px solid black; padding: 8px;">a<sub>b</sub></td>
  </tr>
  <tr>
        <td style="border: 1px solid black; padding: 8px;">&lt;SUP&gt;</td>
        <td style="border: 1px solid black; padding: 8px;">a<sup>b</sup></td>
  </tr>
</table>
```

Q4.Create webpage using list tags of HTML

```
<!DOCTYPE html>
<html lang="en">
<head>
        <meta charset="UTF-8">
        <meta name="viewport" content="width=device-width, initial-scale=1.0">
        <title>HTML Lists: Ordered, Unordered, and Definition</title>
</head>
<body>
        <div>
        <h1>HTML List: Ordered, Unordered & Definition List</h1>
        <hr>
        <p>Following is the list of proposed student activities:</p>
        <ol>
        <li>Develop programs related to unit-wise topics in the computer laboratory.</li>
        <li>Develop a module that can be useful in real-life applications.</li>
        <li>Multimedia presentation of the module developed by students.</li>
        </ol>
        <hr>
        <p>List of Software/Learning Websites</p>
        <ul>
        <li>ASP Tutorial - W3Schools <br> <a
href="https://www.w3schools.com/asp/">https://www.w3schools.com/asp/</a></li>
```

```
        <li>Classic ASP Tutorials & Articles - Web Wiz<br> <a
href="https://www.webwiz.co.uk/Knowledgebase/">https://www.webwiz.co.uk/Knowledgebase/</
a></li>
        <li>HTML Tutorial - W3Schools<br> <a
href="https://www.w3schools.com/html/">https://www.w3schools.com/html/</a></li>
        <li>CSS Tutorial <br> <a
href="https://www.csstutorial.net/">https://www.csstutorial.net/</a></li>
        <li>VBScript Tutorial - Tutorials Point <br> <a
href="https://www.tutorialspoint.com/vbscript/index.htm">https://www.tutorialspoint.com/vbscript/
index.htm</a></li>
        <li>ADO Tutorials - W3Schools <br> <a
href="https://www.w3schools.com/asp/asp_ado.asp">https://www.w3schools.com/asp/asp_ado.
asp</a></li>
        </ul>
        <dl>
        <dt>HTML</dt>
        <dd>HyperText Markup Language</dd>
        <dt>XML</dt>
        <dd>Extensible Markup Language</dd>
        </dl>
        </div>
</body>
</html>
```

Q5. Create webpage to include image using HTML tag

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Image Inclusion using HTML</title>
</head>
<body>

<h2>Peace and Serenity</h2>
<img src="scenery.jpg" alt="Scenery">

</body>
</html>
```

Q6. Modify your page so that the picture that is on your page will also serve as a link that leads to another page.

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Image leads to Another Page</title>
</head>
<body>

<h2>Click on the Image for Peace and Serenity!</h2>
<a href="https://www.youtube.com/watch?v=G1hKzCkywM8">
        <img src="scenery.jpg" alt="Scenery">
</a>

</body>
</html>
```

Q7. Create employee registration webpage using HTML form objects

```
<!DOCTYPE html>
<html>
<head>
        <title>Employee Registration Form</title>
</head>
<body>
        <h1>Employee Registration Form</h1>
        <form>
        <label>
        <input type="radio" name="title" value="Mr."> Mr.
        <input type="radio" name="title" value="Mrs."> Mrs.
        <input type="radio" name="title" value="Ms."> Ms.
        </label>
        <br><br>
        First Name:     <input type="text" placeholder="First Name"><br><br>
        Last Name:     <input type="text" placeholder="Last Name"><br><br>
        Mail Address1: <input type="text"><br><br>
        Mail Address2: <input type="text"><br><br>
        City:               <input type="text"><br><br>
        State: <select>
```

```
        <option selected>Gujarat</option>
        <!-- Add more options for states -->
        </select><br><br>
        Zip: <input type="text"><br><br>
        Upload Photo: <input type="file">No file selected.<br><br>
        E-Mail: <input type="email"><br><br>
        Mobile: <input type="text" value="+91"><br><br>
        Languages known:
        <br>
        <input type="checkbox" checked> Gujarati<br>
        <input type="checkbox" checked> Hindi<br>
        <input type="checkbox" checked> English<br>
        <input type="checkbox"> Marathi<br><br>
        Additional Information: <textarea rows="4" placeholder="optional"></textarea><br><br>
        <input type="submit" value="Submit">
        <input type="reset" value="Reset">
        </form>
</body>
</html>
```

Q8. College Website pushed to GitHub Repository. (Multiple Webpages)

Q9. Heritage Website pushed to GitHub Repository. (Multiple Webpages)

Q10. Create your personal web page (details in your resume)

```
<!DOCTYPE html>
<html lang="en">
<head>
        <meta charset="UTF-8">
        <meta name="viewport" content="width=device-width, initial-scale=1.0">
        <title>Ibrahim Saify - Cybersecurity Researcher</title>
        <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css">
        <link rel="stylesheet" href="styles.css">
</head>
<body>
        <nav class="navbar navbar-expand-lg navbar-dark bg-dark">
        <div class="container">
        <a class="navbar-brand" href="#">Ibrahim Saify</a>
```

```
        <button class="navbar-toggler" type="button" data-toggle="collapse"
data-target="#navbarNav" aria-controls="navbarNav" aria-expanded="false" aria-label="Toggle
navigation">
                <span class="navbar-toggler-icon"></span>
        </button>
        <div class="collapse navbar-collapse" id="navbarNav">
                <ul class="navbar-nav ml-auto">
                <li class="nav-item"><a class="nav-link" href="#about">About</a></li>
                <li class="nav-item"><a class="nav-link" href="#experience">Experience</a></li>
                <li class="nav-item"><a class="nav-link" href="#skills">Skills</a></li>
                <li class="nav-item"><a class="nav-link" href="#certificates">Certificates</a></li>
                <li class="nav-item"><a class="nav-link" href="#education">Education</a></li>
                </ul>
        </div>
        </div>
        </nav>

        <!-- Welcome message -->
        <div class="container mt-4 text-center">
        <h2 style="font-weight: bold;">Welcome to My Portfolio</h2>
        <p>Get to know more about me and my work</p>
        </div>

        <!-- Circular profile photo -->
        <div class="container">
        <div class="text-center">
        <img src="profile.jpeg" alt="Profile Photo" width="200px" class="mx-auto d-block">
        </div>
        </div>

        <section id="about" class="py-5">
        <div class="container">
        <h2 style="font-weight:bold">About Me</h2>
        <p>Cybersecurity researcher and enthusiast, eager to transition into a seasoned
professional. Currently a 3rd-year Computer Science Engineering student actively pursuing
industry experience in cybersecurity through internships.</p>
        <p>Email: ibrahim.saify110@gmail.com<br>Phone: 9904993852<br>Location: Indore,
India<br>LinkedIn: <a href="https://www.linkedin.com/in/ibrahim-saify">Ibrahim Saify</a></p>
        </div>
        </section>

        <section id="experience" class="bg-light py-5">
        <div class="container">
        <h2 style="font-weight:bold">Experience</h2>
```

```
        <div class="row">
                <div class="col-md-6">
                <h3>Security Researcher, VDP (Vulnerability Disclosure Programs)</h3>
                <p>Some of the platforms and vulnerabilities that I have reported and discovered are:</p>
                <ul>
                <li>Reflected XSS on one of NASA's subdomains (Hall of Fame and Resolved [P3])</li>
                <li>Reflected XSS and Open Redirection vulnerability on Duke University Website (Acknowledgement and Resolved)</li>
                <li>Host Header Injection on Merkle Inc. (Acknowledgement)</li>
                <li>CSRF on the quantity of product items on Ferrari's product page (Acknowledgement)</li>
                <li>Weak token code (4-digit) for sign up via a referral link on CaribouCoffee (Access to anyone's referral link by knowing their first name and bruteforcing 4-digit code using a python script).</li>
                </ul>
                </div>
                <div class="col-md-6">
                <h3>CTF Developer, YCF Team</h3>
                <p>Currently part of YCF Team's CTF Development team, crafting challenges for University-hosted CTFs and organization-hosted competitions.</p>
                <p>Some of my contributions include CyberMania 2.0, Cyber Knight CTF 2024, and Techonquer CTF 2024, as well as contributing to organization-hosted competitions like KnightCTF2024 and NexusCTF2024.</p>
                <p>Challenges cover a wide range of subjects, such as Web Application Security, Cryptography, Binary Exploitation, Steganography, and OSINT.</p>
                </div>
        </div>
        </div>
        </section>

        <section id="skills" class="py-5">
        <div class="container">
        <h2 style="font-weight:bold">Skills</h2>
        <ul>
                <li>Programming Languages (Python)</li>
                <li>Computer Networking</li>
                <li>Penetration Testing Tools (CTF tools)</li>
                <li>Cybersecurity/Pentesting Labs</li>
        </ul>
        </div>
        </section>
```

```
<section id="certificates" class="bg-light py-5">
        <div class="container">
        <h2 style="font-weight:bold">Certificates</h2>
        <div class="row">
        <div class="col-md-6">
                <div class="card mb-3">
                <div class="card-body">
                <h5 class="card-title">IIT Bombay Trust Lab's Nationwide CTF</h5>
                <p class="card-text">Finalist at IIT Bombay Trust Lab's Nationwide CTF</p>
                <a
href="https://drive.google.com/file/d/1KTu45-y8P381mFPNfZE7h2lkIwC0EcJ0/view?usp=sharin
g" class="btn btn-primary">View Certificate</a>
                </div>
                </div>
        </div>
        <div class="col-md-6">
                <div class="card mb-3">
                <div class="card-body">
                <h5 class="card-title">Pentathon 2024 CTF</h5>
                <p class="card-text">Finalist at Pentathon 2024 CTF organized by NCIIPC
India</p>
                <a href="#" class="btn btn-primary">View Certificate</a>
                </div>
                </div>
        </div>

        <div class="col-md-6">
                <div class="card mb-3">
                <div class="card-body">
                <h5 class="card-title">Advent of Cyber 2023 - TryHackMe</h5>
                <p class="card-text">Completed the Advent of Cyber 2023 on TryHackMe
Platform</p>
                <a
href="https://drive.google.com/file/d/1Kd4jFXz34GKcT0UsZilcSc0-U8DoacMe/view?usp=sharin
g" class="btn btn-primary">View Certificate</a>
                </div>
                </div>
        </div>
                <div class="col-md-6">
                <div class="card mb-3">
                <div class="card-body">
                <h5 class="card-title">King Of The Hill - TryHackMe</h5>
                <p class="card-text">Winner of a King of the hill tournament - TryHackMe</p>
```

```
                <a href="https://tryhackme.com/ibzsaify5/badges/koth-game" class="btn
btn-primary">View Certificate</a>
                </div>
                </div>
        </div>          </div>
        </div>
</section>


        <section id="education" class="py-5">
        <div class="container">
        <h2 style="font-weight:bold">Education</h2>
        <p>Bachelor of Engineering,</p><a href='https://ietdavv.edu.in/'><p> Institute of
Engineering and Technology, DAVV</p></a>
        <p>Majoring in Computer Science and Engineering</p>
        <p>CGPA - 8.4</p>
        <p>2021 – 2025</p>
        <p>Indore, India</p>
        </div>
        </section>

        <footer class="bg-dark text-white py-4 text-center">
        <div class="container">
        <p>&copy; 2024 Ibrahim Saify</p>
        </div>
        </footer>

        <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
        <script
src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.16.0/umd/popper.min.js"></script>
        <script
src="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js"></script>

        <script>
        // Smooth scrolling
        $(document).ready(function(){
        $("a").on('click', function(event) {
                if (this.hash !== "") {
                event.preventDefault();
                var hash = this.hash;
                $('html, body').animate({
                scrollTop: $(hash).offset().top
                }, 800, function(){
                window.location.hash = hash;
```

```
                });
                }
        });
        });
        </script>
</body>
</html>
```

Q11. Create the following web page (Model Question Paper for Summative Test-II Mathematics - Class VIII)

```
<!DOCTYPE html>
<html>
<head>
        <title>MODEL QUESTION PAPER FOR SUMMATIVE TEST-II</title>
        <style>
        ol {
        margin-left: 20px;
        padding-left: 0;
        }
        li {
        list-style-type: none;
        margin-bottom: 10px;
        }
        </style>
</head>
<body>
        <h3 style="text-align: center;"><u>MODEL QUESTION PAPER FOR SUMMATIVE
TEST-II</u></h3>
        <h3 style="text-align: center;"><u>MATHEMATICS</u></h3>
        <h3 style="text-align: center;"><u>CLASS-VIII</u></h3>
        <p>Maximum Marks: 20</p>
        <p style="font-weight: bold;">General instructions:</p>
        <ol>
        <li>1.All questions are compulsory</li>
        <li>2.The question paper consist of 7 questions divided in to 2 sections A and B</li>
        <li>3.Section A contains 4 questions of 2 marks each</li>
        <li>4.Section B contains 3 questions of 4 marks each</li>
        </ol>
        <h3 style="text-align: center;"><u>SECTION A</u></h3>
        <ol>
        <li>
```

```
	Q1- What will be the product of (a<sup>3</sup>) × (2a<sup>2</sup>) ×
(4a<sup>5</sup>)?<br>
	<br>
	<ol type="a">
		<li>(a)8a<sup>6</sup></li>
		<li>(b)8a<sup>8</sup></li>
		<li>(c)6a<sup>9</sup></li>
		<li>(d)8a<sup>10</sup></li>
	</ol>
	</li>
	<li>
	<br>
	Q2- Water is oxidized to oxygen by<br>
	<br>
	<ol type="a">
		<li>(a)H<sub>2</sub>O</li>
		<li>(b)KMnO<sub>4</sub></li>
		<li>(c)ClO<sub>2</sub></li>
		<li>(d)Fluorine</li>
	</ol>
	</li>
	</ol>
	<p style="text-align: center;"><strike>Cutting Is Not Allowed</strike> Select Only one
Option</p>
</body>
</html>
```

Assignment-2

Q.1 Write a JavaScript program to find all the index positions of a given word within a given string.

```
function findAllIndexes(str, word) {
        let indexes = [];
        let index = -1;

        while ((index = str.indexOf(word, index + 1)) !== -1) {
        indexes.push(index);
        }

        return indexes;
}

let string = "hello world hello";
let word = "world";
let positions = findAllIndexes(string, word);
console.log("The word '" + word + "' appears at positions: " + positions.join(", "));
```

Q.2 Write a JavaScript program to find the first index of a given element in an array using the linear search algorithm.

```
function linearSearch(arr, element) {

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

if (arr[i] === element) {

return i;
}
}
return -1;
}
```

Q.3 Write a JavaScript program to sort a list of elements using Quick sort.

```
function quickSort(arr) {
        if (arr.length <= 1) {
        return arr;
        }

        const pivot = arr[Math.floor(arr.length / 2)];
        const left = [];
        const right = [];
```

```
        for (let i = 0; i < arr.length; i++) {
        if (i === Math.floor(arr.length / 2)) {
        continue;
        }
        if (arr[i] < pivot) {
        left.push(arr[i]);
        } else {
        right.push(arr[i]);
        }
        }

        return [...quickSort(left), pivot, ...quickSort(right)];
}

// Example usage:
const arr = [5, 3, 7, 2, 8, 4, 1];
const sortedArr = quickSort(arr);
console.log("Sorted array:", sortedArr);
```

Q.4 Write a JavaScript program to sort a list of elements using Merge sort.

```
function merge_Arrays(left_sub_array, right_sub_array) {
  let array = []
  while (left_sub_array.length && right_sub_array.length) {
        if (left_sub_array[0] < right_sub_array[0]) {
        array.push(left_sub_array.shift())
        } else {
        array.push(right_sub_array.shift())
        }
  }
  return [ ...array, ...left_sub_array, ...right_sub_array ]
}
function merge_sort(unsorted_Array) {
  const middle_index = unsorted_Array.length / 2
  if(unsorted_Array.length < 2) {
        return unsorted_Array
  }
  const left_sub_array = unsorted_Array.splice(0, middle_index)
  return merge_Arrays(merge_sort(left_sub_array),merge_sort(unsorted_Array))
}
unsorted_Array = [39, 28, 44, 4, 10, 83, 11];
console.log("The sorted array will be: ",merge_sort(unsorted_Array));
```

Q.5 Write a JavaScript program to sort a list of elements using Heap sort.

```
function customSort(arr) {
	var N = arr.length;

	for (var i = Math.floor(N / 2) - 1; i >= 0; i--)
	customHeapify(arr, N, i);

	for (var i = N - 1; i > 0; i--) {
	var temp = arr[0];
	arr[0] = arr[i];
	arr[i] = temp;
	customHeapify(arr, i, 0);
	}
}

function customHeapify(arr, N, i) {
	var largest = i;
	var l = 2 * i + 1;
	var r = 2 * i + 2;

	if (l < N && arr[l] > arr[largest])
	largest = l;

	if (r < N && arr[r] > arr[largest])
	largest = r;

	if (largest != i) {
	var swap = arr[i];
	arr[i] = arr[largest];
	arr[largest] = swap;
	customHeapify(arr, N, largest);
	}
}

function customPrintArray(arr) {
	var N = arr.length;
	for (var i = 0; i < N; ++i)
	console.log(arr[i]);
}

var originalArr = [12, 11, 13, 5, 6, 7];
customSort(originalArr);
console.log("Sorted array:" + originalArr);
```

Q.6 Write a JavaScript program to sort a list of elements using Insertion sort.

```
function insertionSort(arr) {
	const n = arr.length;

	for (let i = 1; i < n; i++) {
	let key = arr[i];
	let j = i - 1;

	while (j >= 0 && arr[j] > key) {
	arr[j + 1] = arr[j];
	j--;
	}

	arr[j + 1] = key;
	}

	return arr;
}

// Example usage:
const originalArr = [5, 3, 7, 2, 8, 4, 1];
const sortedArr = insertionSort(originalArr);
console.log("Sorted array:", sortedArr);
```

Q.7 Write a JavaScript program to sort a list of elements using Bubble sort

```
function bubbleSort(arr) {
	const n = arr.length;

	for (let i = 0; i < n - 1; i++) {
	for (let j = 0; j < n - i - 1; j++) {
	if (arr[j] > arr[j + 1]) {
		// Swap arr[j] and arr[j+1]
		let temp = arr[j];
		arr[j] = arr[j + 1];
		arr[j + 1] = temp;
	}
	}
	}

	return arr;
}

// Example usage:
```

```
const arr = [5, 3, 7, 2, 8, 4, 1];
const sortedArr = bubbleSort(arr);
console.log("Sorted array:", sortedArr);
```

Q.8 Write a JavaScript program to sort the characters in a string alphabetically.

```
function sortStringAlphabetically(str) {
        return str.split('').sort().join('');
}

const inputString = "hello world";
const sortedString = sortStringAlphabetically(inputString);
console.log("Original string:", inputString);
console.log("Sorted string:", sortedString);
```

Q.9 Write a JavaScript program to check if a numeric array is sorted or not.

```
function isArraySorted(arr) {
        for (let i = 0; i < arr.length - 1; i++) {
        if (arr[i] > arr[i + 1]) {
        return false;
        }
        }
        return true;
}

// Example usage:
const sortedArray = [1, 2, 3, 4, 5];
const unsortedArray = [5, 3, 7, 2, 8];
console.log("Is sortedArray sorted?", isArraySorted(sortedArray));
console.log("Is unsortedArray sorted?", isArraySorted(unsortedArray));
```

Q.10 Write a JavaScript function to validate whether a given value type is null or not.

```
function isNull(value) {
   return value === null;
}

// Example usage:
console.log(isNull(null)); // true
console.log(isNull(5));    // false
```

Q.11 Write a JavaScript function to validate whether a given value is a number or not.

```
function isNumber(value) {
   return typeof value === 'number' && !isNaN(value);
}

// Example usage:
console.log(isNumber(5));      // true
console.log(isNumber("hello")); // false
```

Q.12 Write a JavaScript function to validate whether a given value is RegExp or not.

```
function isRegExp(value) {
   return Object.prototype.toString.call(value) === '[object RegExp]';
}

// Example usage:
console.log(isRegExp(/test/)); // true
console.log(isRegExp("hello")); // false
```

Q.13 Write a JavaScript program to delete the rollno property from the following object. Also print the object before or after deleting the property.

Sample object:

```
var student = {
name : "David Rayy",
sclass : "VI",
rollno : 12 };
```

```
var student = {
        name: "David Rayy",
        sclass: "VI",
        rollno: 12
};

console.log("Object before deleting rollno property:", student);
```

```
delete student.rollno;

console.log("Object after deleting rollno property:", student);
```

Q.14 Write a JavaScript program to display the reading status (i.e. display book name, author name and reading status) of the following books.

```
var library = [
 {
 author: 'Bill Gates',
 title: 'The Road Ahead',
 readingStatus: true
 },
 {
 author: 'Steve Jobs',
 title: 'Walter Isaacson',
 readingStatus: true
 },
 {
 author: 'Suzanne Collins',
 title: 'Mockingjay: The Final Book of The Hunger Games',
 readingStatus: false
}];
```

```
var library = [
        {
        author: 'Bill Gates',
        title: 'The Road Ahead',
        readingStatus: true
        },
        {
        author: 'Steve Jobs',
        title: 'Walter Isaacson',
        readingStatus: true
        },
        {
        author: 'Suzanne Collins',
        title: 'Mockingjay: The Final Book of The Hunger Games',
        readingStatus: false
        }
];

for (var i = 0; i < library.length; i++) {
        var book = library[i];
```

```
        var bookInfo = "'" + book.title + "' by " + book.author;

        if (book.readingStatus) {
        console.log("You have already read " + bookInfo + ".");
        } else {
        console.log("You haven't read " + bookInfo + " yet.");
        }
}
```

Q.15 Write a JavaScript program to create a clock.
Note: The output will come every second.
Expected Console Output :
"14:37:42"
"14:37:43"
"14:37:44"
"14:37:45"
"14:37:46"
"14:37:47"

```
function displayTime() {
        var date = new Date();
        var hours = formatTime(date.getHours());
        var minutes = formatTime(date.getMinutes());
        var seconds = formatTime(date.getSeconds());
        console.log(hours + ":" + minutes + ":" + seconds);
}

function formatTime(time) {
        return time < 10 ? "0" + time : time;
}

setInterval(displayTime, 1000);
```

Q.16 Write a JavaScript function to parse an URL.

```
function parseURL(url) {
        var parser = document.createElement('a');
        parser.href = url;

        return {
        protocol: parser.protocol,
        hostname: parser.hostname,
        port: parser.port,
        pathname: parser.pathname,
        search: parser.search,
```

```
        hash: parser.hash,
        origin: parser.origin
        };
}

// Example usage:
var url =
"https://www.example.com:8080/path/to/page?key1=value1&key2=value2#section1";
var parsedURL = parseURL(url);
console.log(parsedURL);
```

Q.17 Write a JavaScript function to split a string and convert it into an array of words

```
function splitStringIntoWords(str) {
        // Use the split method to split the string into an array of words
        return str.split(/\s+/);
}

// Example usage:
var sentence = "This is a sample sentence.";
var wordsArray = splitStringIntoWords(sentence);
console.log(wordsArray);
```

Q.18 Write a JavaScript function that takes a string with both lowercase and upper case letters as a parameter. It converts upper case letters to lower case, and lower case letters to upper case.

```
function swapCase(str) {
        var swapped = "";
        for (var i = 0; i < str.length; i++) {
        var char = str[i];
        if (char === char.toUpperCase()) {
        swapped += char.toLowerCase();
        } else {
        swapped += char.toUpperCase();
        }
        }
        return swapped;
}

// Example usage:
var inputString = "Hello World";
var swappedString = swapCase(inputString);
console.log("Original string:", inputString);
console.log("Swapped string:", swappedString);
```

Q.19 Write a JavaScript function that returns the number of minutes in hours and minutes.

Input :
console.log(timeConvert(200));
Output :
"200 minutes = 3 hour(s) and 20 minute(s)."

```
function timeConvert(minutes) {
	var hours = Math.floor(minutes / 60);
	var remainingMinutes = minutes % 60;
	return minutes + " minutes = " + hours + " hour(s) and " + remainingMinutes + "
minute(s).";
}

// Example usage:
console.log(timeConvert(200));
```

Q.20 Write a JavaScript program to implement a stack that checks if a given element is present or not in the stack.

```
class Stack {
	constructor() {
	this.items = [];
	}

	push(element) {
	this.items.push(element);
	}

	search(element) {
	return this.items.includes(element);
	}
}

// Example usage:
var stack = new Stack();
stack.push(5);
stack.push(10);
stack.push(15);

console.log("Is 10 present in the stack?", stack.search(10)); // true
console.log("Is 20 present in the stack?", stack.search(20)); // false
```

Q.21 Write a JavaScript program to check whether a single linked list is empty or not. Return true otherwise false.

```
class Node {
        constructor(data) {
        this.data = data;
        this.next = null;
        }
}

class LinkedList {
        constructor() {
        this.head = null;
        }

        isEmpty() {
        return this.head === null;
        }
}

// Example usage:
var linkedList = new LinkedList();
console.log(linkedList.isEmpty()); // true

linkedList.head = new Node(10);
console.log(linkedList.isEmpty()); // false
```

Q.22 Write a JavaScript program to create a class called 'Rectangle' with properties for width and height. Include two methods to calculate rectangle area and perimeter. Create an instance of the 'Rectangle' class and calculate its area and perimeter.

```
class Rectangle {
        constructor(width, height) {
        this.width = width;
        this.height = height;
        }

        calculateArea() {
        return this.width * this.height;
        }

        calculatePerimeter() {
        return 2 * (this.width + this.height);
```

```
        }
}

var rectangle = new Rectangle(5, 10);

var area = rectangle.calculateArea();
var perimeter = rectangle.calculatePerimeter();

console.log("Area:", area); //50
console.log("Perimeter:", perimeter); //30
```

Q.23 Write a JavaScript program to create a slideshow that changes the displayed image when a next or previous button is clicked.

```
var images = ["image1.jpg", "image2.jpg", "image3.jpg"]; // Replace these with actual image URLs
var currentIndex = 0;

function showSlide(index) {
        var image = document.getElementById("image");
        if (index >= 0 && index < images.length) {
        image.src = images[index];
        currentIndex = index;
        }
}

function nextSlide() {
        currentIndex = (currentIndex + 1) % images.length;
        showSlide(currentIndex);
}

function prevSlide() {
        currentIndex = (currentIndex - 1 + images.length) % images.length;
        showSlide(currentIndex);
}

showSlide(currentIndex);
```

Q.24 Write a JavaScript program that uses a try-catch block to catch and handle a 'SyntaxError' when parsing an invalid JSON string.

```
var invalidJSONString = '{"name": "Jason", "age": 24,}';

try {
	var parsedObject = JSON.parse(invalidJSONString);
	console.log(parsedObject);
} catch (error) {
	if (error instanceof SyntaxError) {
	console.log("Error: Invalid JSON string.");
	console.log(error.message);
	} else {
	throw error;
	}
}
```

Q.25 Write a JavaScript program to redirect to a specified URL.

```
// Specified URL to Redirect To
var redirectURL = "https://www.example.com";

// Redirection to the Specified URL
window.location.href = redirectURL;
```