Practical 3

<!DOCTYPE html>

<html lang=”en”>

<head>

<meta charset=”UTF-8”>

<meta name=”viewport” content=”width=device-width, initial-scale=1.0”>

<title>Frames Example</title>

<style>

Body {

Margin: 0;

Padding: 0;

Display: flex;

}

#frame1 {

Width: 20%;

Height: 100vh;

Border: 1px solid #ccc;

}

#frame2 {

Flex: 1;

Height: 100vh;

Border: 1px solid #ccc;

Overflow: auto;

}

Iframe {

Width: 100%;

Height: 300px; /\* Adjust height as needed \*/

Border: 1px solid #ccc;

}

</style>

</head>

<body>

<div id=”frame1”>

<ul>

<li><a href=”javascript:void(0);” onclick=”loadContent(‘content1.html’)”>Link 1</a></li>

<li><a href=”javascript:void(0);” onclick=”loadContent(‘content2.html’)”>Link 2</a></li>

<!—Add more links as needed 🡪

</ul>

</div>

<div id=”frame2”>

<p>Select a link to display content here.</p>

</div>

<iframe src=<https://www.example.com> frameborder=”0”></iframe>

<script>

Function loadContent(url) {

Var frame2 = document.getElementById(‘frame2’);

Frame2.innerHTML = ‘<object type=”text/html” data=”’ + url + ‘” style=”width: 100%; height: 100%;”></object>’;

}

</script>

</body>

</html>

Practical 5th

<!DOCTYPE html>

<html lang=”en”>

<head>

<meta charset=”UTF-8”>

<meta name=”viewport” content=”width=device-width, initial-scale=1.0”>

<title>Admission Form</title>

<style>

Body {

Font-family: Arial, sans-serif;

Background-color: #f4f4f4;

Margin: 0;

Padding: 0;

Display: flex;

Justify-content: center;

Align-items: center;

Height: 100vh;

}

Form {

Background-color: #fff;

Padding: 20px;

Border-radius: 8px;

Box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);

}

Label {

Display: block;

Margin-bottom: 8px;

Font-weight: bold;

}

Input, select {

Width: 100%;

Padding: 8px;

Margin-bottom: 16px;

Box-sizing: border-box;

}

Input[type=”checkbox”], input[type=”radio”] {

Margin-right: 8px;

}

Button {

Background-color: #4caf50;

Color: #fff;

Padding: 10px 15px;

Border: none;

Border-radius: 4px;

Cursor: pointer;

}

Button[type=”reset”] {

Background-color: #f44336;

}

</style>

</head>

<body>

<form>

<label for=”name”>Name:</label>

<input type=”text” id=”name” name=”name” required>

<label for=”password”>Password:</label>

<input type=”password” id=”password” name=”password” required>

<label for=”course”>Select Course:</label>

<select id=”course” name=”course”>

<option value=”engineering”>Engineering</option>

<option value=”science”>Science</option>

<option value=”arts”>Arts</option>

</select>

<label>Choose Specializations:</label>

<input type=”checkbox” id=”spec1” name=”specialization” value=”spec1”>

<label for=”spec1”>Specialization 1</label>

<input type=”checkbox” id=”spec2” name=”specialization” value=”spec2”>

<label for=”spec2”>Specialization 2</label>

<label>Gender:</label>

<input type=”radio” id=”male” name=”gender” value=”male”>

<label for=”male”>Male</label>

<input type=”radio” id=”female” name=”gender” value=”female”>

<label for=”female”>Female</label>

<button type=”submit”>Submit</button>

<button type=”reset”>Reset</button>

</form>

</body>

</html>

Practical 7

<!DOCTYPE html>

<html lang=”en”>

<head>

<meta charset=”UTF-8”>

<meta name=”viewport” content=”width=device-width, initial-scale=1.0”>

<title>Current Day and Time</title>

<script>

Function displayDateTime() {

Var currentDate = new Date();

Var daysOfWeek = [“Sunday”, “Monday”, “Tuesday”, “Wednesday”, “Thursday”, “Friday”, “Saturday”];

Var day = daysOfWeek[currentDate.getDay()];

Var hours = currentDate.getHours();

Var minutes = currentDate.getMinutes();

Var seconds = currentDate.getSeconds();

// Add leading zeros if needed

Hours = (hours < 10) ? “0” + hours : hours;

Minutes = (minutes < 10) ? “0” + minutes : minutes;

Seconds = (seconds < 10) ? “0” + seconds : seconds;

Var currentTime = day + “ “ + hours + “:” + minutes + “:” + seconds;

Document.getElementById(“currentDateTime”).innerHTML = currentTime;

}

// Call the function when the page loads

Window.onload = displayDateTime;

</script>

</head>

<body>

<p id=”currentDateTime”></p>

</body>

</html>

Practical 8

<!DOCTYPE html>

<html lang=”en”>

<head>

<meta charset=”UTF-8”>

<meta name=”viewport” content=”width=device-width, initial-scale=1.0”>

<title>String Manipulation</title>

<script>

// Function to remove a character at the specified position

Function removeCharAtPosition(inputString, position) {

If (position < 0 || position >= inputString.length) {

Return “Invalid position”;

}

Var resultString = inputString.substring(0, position) + inputString.substring(position + 1);

Return resultString;

}

// Function to change the case of a string

Function changeCase(inputString) {

Var resultString = “”;

For (var I = 0; I < inputString.length; i++) {

Var currentChar = inputString.charAt(i);

If (currentChar === currentChar.toUpperCase()) {

resultString += currentChar.toLowerCase();

} else {

resultString += currentChar.toUpperCase();

}

}

Return resultString;

}

// Example usage

Var originalString = “Hello, World!”;

Var positionToRemove = 7;

Var modifiedString1 = removeCharAtPosition(originalString, positionToRemove);

Var modifiedString2 = changeCase(originalString);

Console.log(“Original String:”, originalString);

Console.log(“After removing character at position”, positionToRemove + “:”, modifiedString1);

Console.log(“After changing case:”, modifiedString2);

</script>

</head>

<body>

<!—You can display the results in the HTML if needed 🡪

</body>

</html>

Practical 9

<!DOCTYPE html>

<html lang=”en”>

<head>

<meta charset=”UTF-8”>

<meta name=”viewport” content=”width=device-width, initial-scale=1.0”>

<title>Sum of Array Elements</title>

<script>

Function computeSum(array) {

Var sum = 0;

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

Sum += array[i];

}

Return sum;

}

// Example usage

Var numbers = [1, 2, 3, 4, 5];

Var result = computeSum(numbers);

Console.log(“Array:”, numbers);

Console.log(“Sum of Array Elements:”, result);

</script>

</head>

<body>

<!—You can display the results in the HTML if needed 🡪

</body>

</html>

Practical 10

<!DOCTYPE html>

<html lang=”en”>

<head>

<meta charset=”UTF-8”>

<meta name=”viewport” content=”width=device-width, initial-scale=1.0”>

<title>Name Parser</title>

<script>

Function parseName() {

Var fullName = document.getElementById(‘fullName’).value.trim();

Var names = fullName.split(‘ ‘);

Var firstName = names[0] || ‘NA’;

Var middleName = names.length > 2 ? names.slice(1, -1).join(‘ ‘) : ‘NA’;

Var lastName = names.length > 1 ? names[names.length – 1] : ‘NA’;

Document.getElementById(‘firstName’).innerText = ‘First Name: ‘ + firstName;

Document.getElementById(‘middleName’).innerText = ‘Middle Name: ‘ + middleName;

Document.getElementById(‘lastName’).innerText = ‘Last Name: ‘ + lastName;

}

</script>

</head>

<body>

<label for=”fullName”>Enter Full Name:</label>

<input type=”text” id=”fullName” placeholder=”John Doe”>

<button onclick=”parseName()”>Parse Name</button>

<div>

<label id=”firstName”>First Name: NA</label><br>

<label id=”middleName”>Middle Name: NA</label><br>

<label id=”lastName”>Last Name: NA</label>

</div>

</body>

</html>

Practical 11

<!DOCTYPE html>

<html lang=”en”>

<head>

<meta charset=”UTF-8”>

<meta name=”viewport” content=”width=device-width, initial-scale=1.0”>

<title>Student Record Form</title>

<style>

Body {

Font-family: Arial, sans-serif;

Background-color: #f4f4f4;

Margin: 0;

Padding: 0;

Display: flex;

Justify-content: center;

Align-items: center;

Height: 100vh;

}

Form {

Background-color: #fff;

Padding: 20px;

Border-radius: 8px;

Box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);

}

Label {

Display: block;

Margin-bottom: 8px;

Font-weight: bold;

}

Input, select {

Width: 100%;

Padding: 8px;

Margin-bottom: 16px;

Box-sizing: border-box;

}

Button {

Background-color: #4caf50;

Color: #fff;

Padding: 10px 15px;

Border: none;

Border-radius: 4px;

Cursor: pointer;

}

</style>

</head>

<body>

<form onsubmit=”return validateForm()”>

<label for=”studentName”>Student Name:</label>

<input type=”text” id=”studentName” name=”studentName” required>

<label for=”rollNumber”>Roll Number:</label>

<input type=”text” id=”rollNumber” name=”rollNumber” required pattern=”[0-9]+”>

<label for=”email”>Email Address:</label>

<input type=”email” id=”email” name=”email” required>

<label for=”course”>Select Course:</label>

<select id=”course” name=”course” required>

<option value=””>Select</option>

<option value=”engineering”>Engineering</option>

<option value=”science”>Science</option>

<option value=”arts”>Arts</option>

</select>

<label for=”semester”>Semester:</label>

<input type=”number” id=”semester” name=”semester” required>

<label for=”grade”>Grade:</label>

<input type=”text” id=”grade” name=”grade” pattern=”[A-Fa-f]”>

<button type=”submit”>Submit</button>

</form>

<script>

Function validateForm() {

Var studentName = document.getElementById(‘studentName’).value;

Var rollNumber = document.getElementById(‘rollNumber’).value;

Var email = document.getElementById(‘email’).value;

// Custom validation example: Check if roll number is numeric

If (isNaN(rollNumber)) {

Alert(‘Roll Number must be numeric’);

Return false;

}

// Custom validation example: Check if email contains @ symbol

If (!email.includes(‘@’)) {

Alert(‘Invalid Email Address’);

Return false;

}

// You can add more validation as needed

Return true; // Submit the form if all validations pass

}

</script>

</body>

</html>

Practical 12

<!DOCTYPE html>

<html lang=”en”>

<head>

<meta charset=”UTF-8”>

<meta name=”viewport” content=”width=device-width, initial-scale=1.0”>

<title>Text Animation</title>

<style>

#animatedText {

Color: red;

}

</style>

</head>

<body>

<p id=”animatedText”>Bigger Text</p>

<script>

Var animatedText = document.getElementById(‘animatedText’);

Var fontSize = 10;

Function animateText() {

If (fontSize <= 50) {

animatedText.style.fontSize = fontSize + ‘pt’;

fontSize += 1;

if (fontSize === 50) {

animatedText.innerText = ‘Smaller Text’;

animatedText.style.color = ‘green’;

}

} else if (fontSize > 5) {

animatedText.style.fontSize = fontSize + ‘pt’;

fontSize -= 1;

} else {

clearInterval(animationInterval);

}

}

// Set interval to run the animation every 10 milliseconds

Var animationInterval = setInterval(animateText, 10);

</script>

</body>

</html>

Practical 13

<!DOCTYPE html>

<html lang=”en”>

<head>

<meta charset=”UTF-8”>

<meta name=”viewport” content=”width=device-width, initial-scale=1.0”>

<title>Whitespace Remover</title>

<style>

Body {

Font-family: Arial, sans-serif;

Background-color: #f4f4f4;

Margin: 0;

Padding: 0;

Display: flex;

Justify-content: center;

Align-items: center;

Height: 100vh;

}

Form {

Background-color: #fff;

Padding: 20px;

Border-radius: 8px;

Box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);

}

Label {

Display: block;

Margin-bottom: 8px;

Font-weight: bold;

}

Input {

Width: 100%;

Padding: 8px;

Margin-bottom: 16px;

Box-sizing: border-box;

}

Button {

Background-color: #4caf50;

Color: #fff;

Padding: 10px 15px;

Border: none;

Border-radius: 4px;

Cursor: pointer;

}

</style>

</head>

<body>

<form method=”post” action=”index.php”>

<label for=”inputString”>Enter a String:</label>

<input type=”text” name=”inputString” required>

<button type=”submit”>Remove Whitespaces</button>

</form>

<?php

If ($\_SERVER[“REQUEST\_METHOD”] == “POST”) {

$inputString = $\_POST[“inputString”];

// Call Python script to remove whitespaces

$outputString = shell\_exec(“python3 remove\_whitespaces.py \”$inputString\””);

Echo “<p>String without Whitespaces: $outputString</p>”;

}

?>

</body>

</html>

Python program

Import sys

Def remove\_whitespaces(input\_string):

Return input\_string.replace(“ “, “”)

If \_\_name\_\_ == “\_\_main\_\_”:

Input\_string = sys.argv[1]

Output\_string = remove\_whitespaces(input\_string)

Print(output\_string)

Practical 14

<!DOCTYPE html>

<html lang=”en”>

<head>

<meta charset=”UTF-8”>

<meta name=”viewport” content=”width=device-width, initial-scale=1.0”>

<title>Login Page</title>

<style>

Body {

Font-family: Arial, sans-serif;

Background-color: #f4f4f4;

Margin: 0;

Padding: 0;

Display: flex;

Justify-content: center;

Align-items: center;

Height: 100vh;

}

Form {

Background-color: #fff;

Padding: 20px;

Border-radius: 8px;

Box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);

}

Label {

Display: block;

Margin-bottom: 8px;

Font-weight: bold;

}

Input {

Width: 100%;

Padding: 8px;

Margin-bottom: 16px;

Box-sizing: border-box;

}

Button {

Background-color: #4caf50;

Color: #fff;

Padding: 10px 15px;

Border: none;

Border-radius: 4px;

Cursor: pointer;

}

</style>

</head>

<body>

<form method=”post” action=”login.php”>

<label for=”username”>Username:</label>

<input type=”text” name=”username” required>

<label for=”password”>Password:</label>

<input type=”password” name=”password” required>

<button type=”submit”>Submit</button>

</form>

</body>

</html>

PHP programming

<?php

$servername = “your\_servername”;

$username = “your\_username”;

$password = “your\_password”;

$dbname = “your\_database”;

// Create connection

$conn = new mysqli($servername, $username, $password, $dbname);

// Check connection

If ($conn->connect\_error) {

Die(“Connection failed: “ . $conn->connect\_error);

}

If ($\_SERVER[“REQUEST\_METHOD”] == “POST”) {

$username = $\_POST[“username”];

$password = $\_POST[“password”];

// Use prepared statement to prevent SQL injection

$stmt = $conn->prepare(“SELECT username FROM users WHERE username = ? AND password = ?”);

$stmt->bind\_param(“ss”, $username, $password);

// Execute the statement

$stmt->execute();

// Bind the result variable

$stmt->bind\_result($result);

// Fetch the result

$stmt->fetch();

If ($result) {

Echo “Welcome, $username!”;

} else {

Echo “Error: Invalid credentials”;

}

// Close the statement

$stmt->close();

}

// Close the connection

$conn->close();

?>

Practical 15

<!DOCTYPE html>

<html lang=”en”>

<head>

<meta charset=”UTF-8”>

<meta name=”viewport” content=”width=device-width, initial-scale=1.0”>

<title>Birthday Countdown</title>

<style>

Body {

Font-family: Arial, sans-serif;

Background-color: #f4f4f4;

Margin: 0;

Padding: 0;

Display: flex;

Justify-content: center;

Align-items: center;

Height: 100vh;

}

.countdown-container {

Text-align: center;

}

#countdown {

Font-size: 24px;

Margin-top: 10px;

}

</style>

</head>

<body>

<div class=”countdown-container”>

<h1>Birthday Countdown</h1>

<label for=”birthdayDate”>Enter Your Birthday:</label>

<input type=”date” id=”birthdayDate” required>

<button onclick=”calculateDays()”>Calculate Days</button>

<p id=”countdown”></p>

</div>

<script>

Function calculateDays() {

Var birthdayInput = document.getElementById(‘birthdayDate’).value;

// Get current date

Var currentDate = new Date();

// Convert birthday input to a Date object

Var birthday = new Date(birthdayInput);

// Set the year of the birthday to the current year

Birthday.setFullYear(currentDate.getFullYear());

// Check if the birthday has already occurred this year

If (birthday < currentDate) {

// If yes, set the birthday to next year

Birthday.setFullYear(currentDate.getFullYear() + 1);

}

// Calculate the difference in days

Var timeDiff = birthday.getTime() – currentDate.getTime();

Var daysDiff = Math.ceil(timeDiff / (1000 \* 3600 \* 24));

// Display the countdown

Var countdownElement = document.getElementById(‘countdown’);

countdownElement.innerText = “Days until your next birthday: “ + daysDiff;

}

</script>

</body>

</html>