**PSGCOLLEGEOFTECHNOLOGY, COIMBATORE–641 004**

**Department of Applied Mathematics and Computational Sciences**

**M.Sc. Software Systems – Semester IV**

**23XW48 – Web Development Laboratory**

**PROBLEM SHEET –Java script Arrays, Strings, Date, Number and DOM Events**

1. Write a JavaScript function that finds the largest number in an array.
2. Given an array [1, 2, 3, 4, 5], write a program to reverse the array without using the .reverse() method.
3. Implement a function that removes duplicates from an array.
4. Write a JavaScript program that sorts an array of numbers in ascending order.
5. Create a function that finds the second smallest and second largest numbers in an array.
6. Given an array of numbers, write a function to return only the even numbers.
7. Write a JavaScript function that changes the background color of a button when the mouse hovers over it.
8. Create a simple program that displays the X and Y coordinates of the mouse pointer in real-time.
9. Implement a double-click event that changes the text content of a paragraph.
10. Write a program where a button moves to a random position when clicked.
11. Create a draggable div element using mouse events in JavaScript
12. Write a JavaScript program that displays the key pressed by the user in a text box.
13. Create a simple program where pressing the Enter key submits a form automatically.
14. Implement a function that prevents typing the letter ‘a’ in a text field.
15. Develop a program that listens for arrow key presses and moves a div accordingly.
16. Write a script that counts and displays the number of characters typed in a textarea.
17. Write a JavaScript function that displays an alert message when the page is fully loaded.
18. Create an event that detects when the user resizes the browser window and displays the new width and height.
19. Implement a program that saves the scroll position of a page and restores it when reloaded.
20. Write a script that warns users when they try to leave a webpage.
21. Develop a program that counts and displays the number of times the page has been loaded.
22. Design and implement a form validation system using JavaScript that ensures users input valid data before submission. The validation should provide real-time feedback using onkeyup, onblur, and onsubmit events.

**Specifications:**

1. **Form Fields:**
   * **Email Field**: Must be validated using a regular expression for correct format.
   * **Password Field**: Must check password strength while the user is typing.
   * **Confirm Password Field**: Must ensure that it matches the password field.
2. **Validation Requirements:**
   * **Email Validation:**
     + Must display an error message if the format is incorrect.
     + The onblur event should be used to trigger validation.
   * **Password Strength Indicator:**
     + Should classify the password as Weak, Medium, or Strong.
     + The onkeyup event should dynamically update the strength status.
   * **Confirm Password Validation:**
     + Should verify if both passwords match.
     + Display an error message if they do not match.
   * **Empty Field Check:**
     + The submit button should remain **disabled** until all fields are correctly filled.
   * **Form Submission Handling:**
     + Use the onsubmit event to prevent form submission if any validation fails.
     + Display an alert message when submission is blocked due to errors.
3. **User Experience Enhancements:**
   * Error messages should be displayed in **red**.
   * The password strength indicator should use **color coding** (Red for Weak, Orange for Medium, Green for Strong).
   * The submit button should **only enable when all fields are correctly filled**.

**Expected Output:**

* **Valid Email**: No error message.
* **Invalid Email**: Displays “Invalid email format!” error.
* **Password Strength**: Displays Weak, Medium, or Strong dynamically.
* **Password Mismatch**: Displays “Passwords do not match!” error.
* **Submit Button Behavior**: Disabled until all validations are cleared.

1. Write a JavaScript function to check if a string is a palindrome (e.g., "madam").
2. Given a string, write a function that counts the number of vowels in the string.
3. Implement a function to reverse a string without using .reverse().
4. Write a function that replaces all spaces in a string with a hyphen (-).
5. Given a string "hello world", write a function to capitalize the first letter of each word.
6. Write a JavaScript function to count the occurrences of each character in a given string.
7. Create a function that extracts all numbers from a given alphanumeric string (e.g., "a1b2c3" → [1,2,3]).
8. Implement a function to perform **string compression** (e.g., "aaabbc" → "a3b2c1").
9. Write a function that returns the current day of the week (e.g., "Monday").
10. Implement a function that calculates a person's age based on their birth date.
11. Given a date, write a function to add n days to it and return the new date.
12. Write a function that finds the number of days between two given dates.
13. Implement a function that checks if a given year is a leap year.
14. Create a function that converts a date into a readable format like "12th March 2025".
15. Write a function that generates a random number between a given range.
16. Implement a function that checks if a given number is prime.
17. Write a function that finds the factorial of a given number.
18. Create a function that converts a decimal number to binary.
19. Given a floating-point number, write a function that rounds it to the nearest integer.
20. Implement a program to find the sum of digits of a given number.
21. Write a JavaScript program that converts a number into its **word representation** (e.g., 123 → "One Hundred Twenty-Three").