

# Assignment 4.2

**Subject:** PPWJ (CSE 3838)

**Session:** Sep 2025 to Jan 2026

**Branch:** CSIT

**Section:** All

**Course Outcomes:** CO2

**Learning Levels:** Remembering (L1), Understanding (L2), Application (L3), and Analysis (L4)

<b>Q No.</b>	<b>Questions</b>	<b>Learning Levels</b>
Q1	<p>Suppose an ATM system requires users to enter a unique security code (an integer). The system calculates the sum of the digits of this code. To verify additional security, the system checks whether this sum is a prime number. A prime digit sum means the code passes an extra security filter.</p> <p>Write a JavaScript program that performs the following:</p> <ol style="list-style-type: none"><li>1. Takes an integer input (the security code) from the user.</li><li>2. Calculates the sum of its digits using a <code>while</code> loop.</li><li>3. Determines whether the digit sum is prime or not.</li><li>4. Displays:<ul style="list-style-type: none"><li>• The original number entered,</li><li>• The sum of its digits,</li><li>• Whether the sum is a prime number.</li></ul></li></ol>	L2,L3

<p><b>Q2</b></p> <p>In some mobile banking and authentication systems, a user's security PIN may need to be reversed for encryption or verification purposes. For example, if the PIN entered is 1234, the system generates a reversed PIN 4321 before applying encryption.</p> <p>Write a JavaScript program that performs the following:</p> <ol style="list-style-type: none"> <li>1. Takes an integer input (the user's PIN code) from the user.</li> <li>2. Uses a <b>for</b> loop to reverse the digits of the entered number.</li> <li>3. Displays: <ul style="list-style-type: none"> <li>• The original number (PIN),</li> <li>• The reversed number (Reversed PIN).</li> </ul> </li> </ol>	<p>L3</p>
<p><b>Q3</b></p> <p>In mathematics, Pascal's Triangle is a triangular array of numbers where each number is the sum of the two numbers directly above it. It is widely used in probability, algebra, and combinatorics. As part of a mathematical learning application, you are required to generate Pascal's Triangle using nested loops in JavaScript. This will help visualize how binomial coefficients are arranged in a triangular format.</p> <p>Write a JavaScript program that:</p> <ol style="list-style-type: none"> <li>1. Uses nested <b>for</b> loops to generate Pascal's Triangle.</li> <li>2. Displays Pascal's Triangle up to 5 rows.</li> </ol> <p><b>Example Output (5 rows):</b></p> <pre style="margin-left: 40px;">    1    1   1   1   2   1  1   3   3   1 1   4   6   4   1</pre>	<p>L2,L3</p>

Q4	<p>A retail shop wants to keep track of its sales performance over a week (7 days). The sales amounts are stored in an array. The shop manager wants a small program to:</p> <ul style="list-style-type: none"> <li>• Calculate the total weekly sales,</li> <li>• Find the average sales per day,</li> <li>• Identify the day with the highest sales,</li> <li>• Identify the day with the lowest sales.</li> </ul>	L3
Q5	<p>In a school, marks of students in different subjects are often stored in a 2D array (rows represent students and columns represent subjects). The teacher wants a program that can calculate the total marks obtained by each student across all subjects. This helps in quickly evaluating students' performance.</p> <p>Write a JavaScript program that:</p> <ol style="list-style-type: none"> <li>1. Stores the marks of 3 students in 3 subjects using a 2D array.</li> <li>2. Uses nested <code>for</code> loops to calculate the total marks of each student.</li> <li>3. Prints the total marks of each student.</li> </ol>	L2, L3
Q6	<p>Write a program that demonstrates how to control loop execution using the <code>break</code> and <code>continue</code> statements. The program should display numbers from 1 to 20 with the following conditions:</p> <ol style="list-style-type: none"> <li>1. Uses a loop to print numbers from 1 to 20.</li> <li>2. Skips multiples of 5 using the <code>continue</code> statement.</li> <li>3. Stops the loop entirely when the number reaches 15 using the <code>break</code> statement.</li> </ol>	L3

Q7	<p>Write a program that performs the following operations on a given array of numbers using loops (no built-in array methods like sort() or reverse()): Write a JavaScript program that include a menu driven program with choice 1,2 ..:</p> <ol style="list-style-type: none"> <li>1. Display all elements of the array.</li> <li>2. Calculate and print the sum, maximum, and minimum value of the array.</li> <li>3. Count and display how many numbers are even and how many are odd.</li> <li>4. Reverse the array and print the reversed version.</li> <li>5. Search for a specific element in the array and display its index (if present).</li> <li>6. Remove duplicates from the array and show the resulting array.</li> <li>7. Sort the array in ascending order using loops only.</li> </ol>	L3
Q8	<p>Math multiplication table In this project, you will create a math multiplication table using loops. You can do this using your own creativity or by following some of the following suggested steps:</p> <ol style="list-style-type: none"> <li>1. Set up a blank array to contain the final multiplication table.</li> <li>2. Set a value variable to specify how many values you want to multiply with one another and show the results for.</li> <li>3. Create an outer for loop to iterate through each row and a temp array to store the row values. Each row will be an array of cells that will be nested into the final table.</li> <li>4. Add an inner for loop for the column values, which will push the multiplied row and column values to the temp array.</li> <li>5. Add the temporary row data that contains the calculated solutions to the main array of the final table. The final result will add a row of values for the calculations.</li> </ol>	L3
<b>-END-</b>		