**Traffic Data Analysis:**

**Collect real-time traffic data (e.g., car counts, speed, and entry/exit times) on multiple road lanes using a dynamic array. Analyse traffic flow patterns for peak hours using array operations like sorting, searching, and aggregation.**

<html>

<head>

<title>Traffic Data Collection</title>

<style>

body {

font-family: Arial, sans-serif;

}

table {

margin: 20px 0;

border-collapse: collapse;

}

th, td {

border: 1px solid #000;

padding: 10px;

text-align: center;

}

</style>

</head>

<body>

<h1>Real-Time Traffic Data Collection</h1>

<h2>Add Traffic Data</h2>

<form id="trafficForm">

<input type="number" id="laneNumber" placeholder="Lane Number" required>

<input type="number" id="carCount" placeholder="Car Count" required>

<input type="number" id="averageSpeed" placeholder="Average Speed (km/h)" required>

<input type="datetime-local" id="entryTime" required>

<input type="datetime-local" id="exitTime" required>

<button type="submit">Add Data</button>

</form>

<h2>Traffic Data</h2>

<table id="trafficTable">

<thead>

<tr>

<th>Lane</th>

<th>Car Count</th>

<th>Average Speed (km/h)</th>

<th>Entry Time</th>

<th>Exit Time</th>

</tr>

</thead>

<tbody id="trafficBody"></tbody>

</table>

<h2>Peak Hour Analysis</h2>

<button id="analyzeButton">Analyze Peak Hours</button>

<div id="analysisResult"></div>

<script>

let trafficData = []; // Dynamic array to store traffic data

// Function to add traffic data

function addTrafficData(lane, carCount, averageSpeed, entryTime, exitTime) {

const data = {

lane,

carCount,

averageSpeed,

entryTime: new Date(entryTime),

exitTime: new Date(exitTime)

};

trafficData.push(data);

renderTrafficData();

}

// Function to render the traffic data table

function renderTrafficData() {

const trafficBody = document.getElementById('trafficBody');

trafficBody.innerHTML = ''; // Clear existing entries

trafficData.forEach(data => {

const row = document.createElement('tr');

row.innerHTML = `

<td>${data.lane}</td>

<td>${data.carCount}</td>

<td>${data.averageSpeed}</td>

<td>${data.entryTime.toLocaleString()}</td>

<td>${data.exitTime.toLocaleString()}</td>

`;

trafficBody.appendChild(row);

});

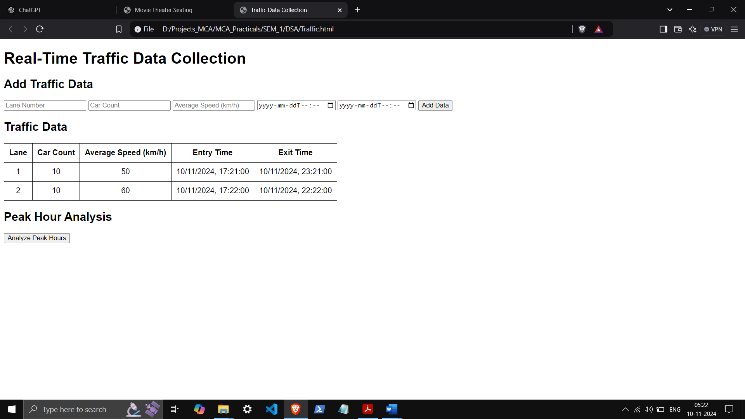
}

// Function to analyze peak hours

function analyzePeakHours() {

const hourCounts = {};

trafficData.forEach(data => {

 const entryHour = data.entryTime.getHours();

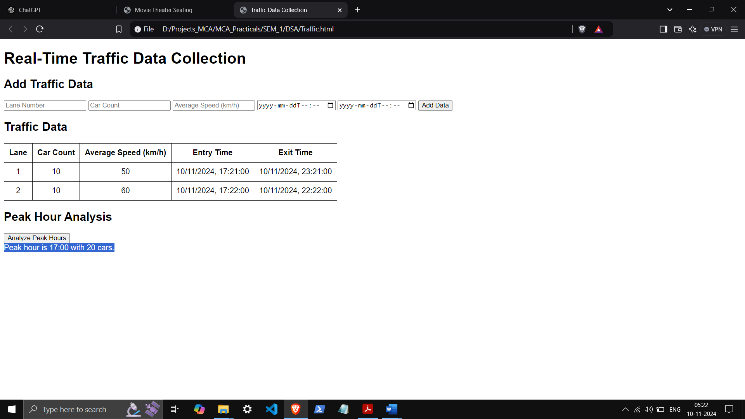
hourCounts[entryHour] = (hourCounts[entryHour] || 0) + data.carCount;

});

const sortedHours = Object.entries(hourCounts).sort((a, b) => b[1] - a[1]);

const peakHour = sortedHours[0] ? sortedHours[0] : [0, 0];

const analysisResult = document.getElementById('analysisResult');

 analysisResult.innerHTML = `Peak hour is ${peakHour[0]}:00 with ${peakHour[1]} cars.`;

}

// Event listener for adding traffic data

document.getElementById('trafficForm').addEventListener('submit', function(event) {

event.preventDefault();

const laneNumber = parseInt(document.getElementById('laneNumber').value);

const carCount = parseInt(document.getElementById('carCount').value);

const averageSpeed = parseInt(document.getElementById('averageSpeed').value);

const entryTime = document.getElementById('entryTime').value;

const exitTime = document.getElementById('exitTime').value;

addTrafficData(laneNumber, carCount, averageSpeed, entryTime, exitTime);

this.reset();

});

// Event listener for analyzing peak hours

document.getElementById('analyzeButton').addEventListener('click', analyzePeakHours);

</script>

</body>

</html>