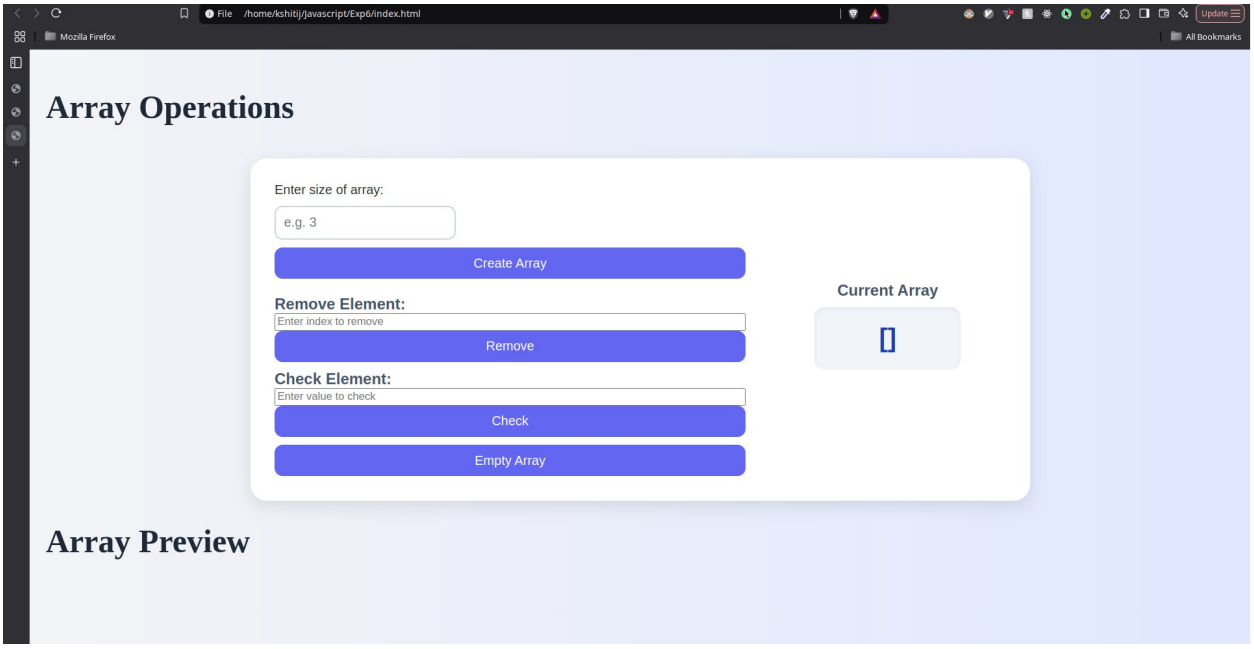
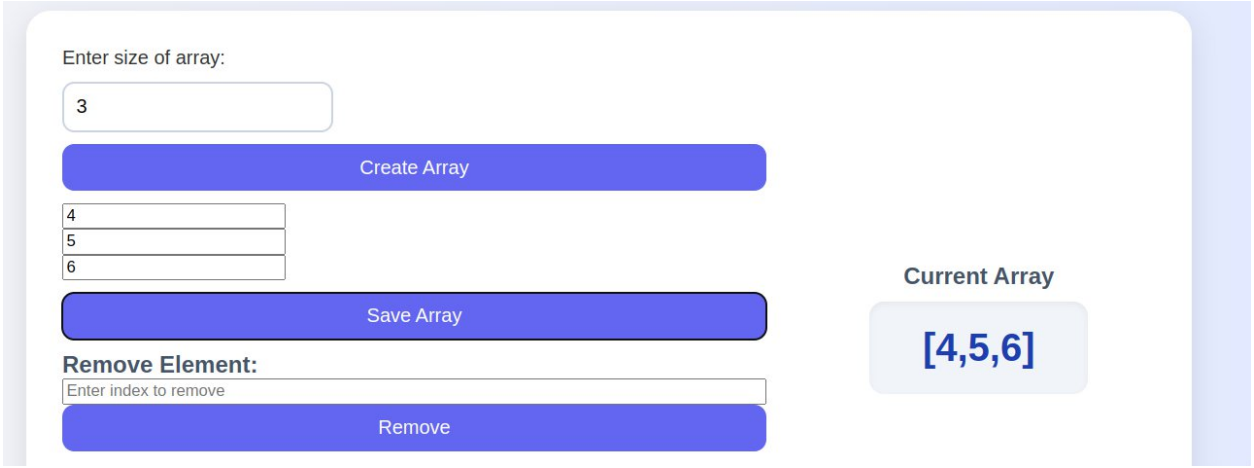
Experiment 6

Roll no: 42405 Batch: Q6

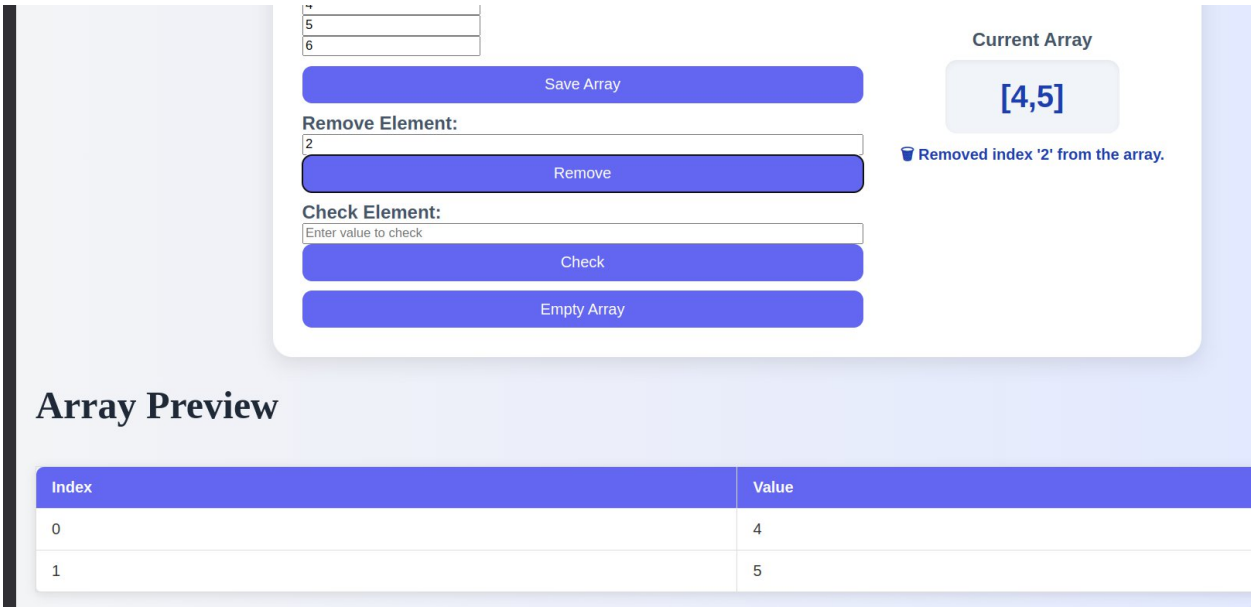
Output:

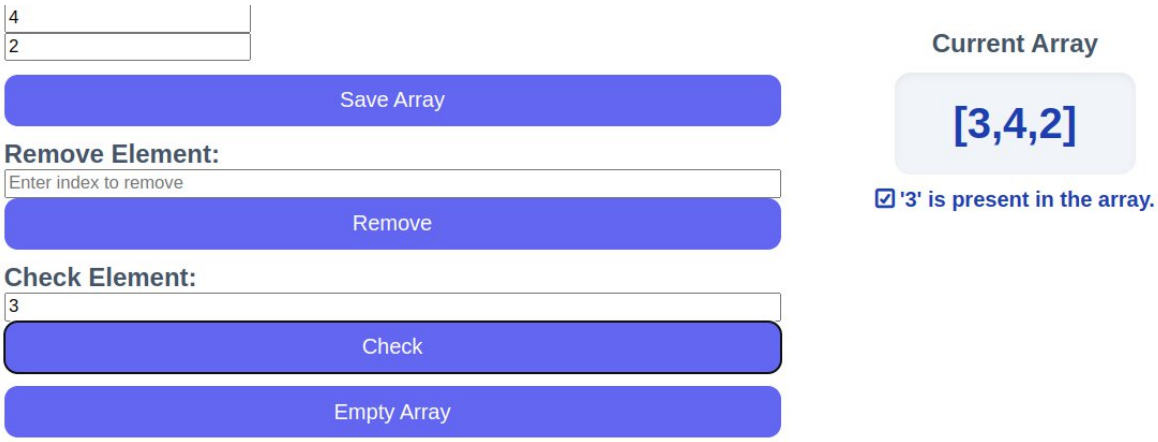




Experiment 6

Roll no: 42405 Batch: Q6





Experiment 6

Roll no: 42405 Batch: Q6

Code:

1. **HTML:**

```
<!DOCTYPE html>
<html lang="en">
<head>
<script type="text/javascript" src="array_objects.js"></script>
<link rel="stylesheet" href="style.css">

    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>
  <body>
    <h1>Array Operations</h1>

    <div class="shape-box">
      <!-- Input Section -->
      <div class="form-section">
        <label for="arraySize">Enter size of array:</label>
        <input type="number" id="arraySize" min="1" placeholder="e.g. 3" />
        <button onclick="createArray()">Create Array</button>
        <div id="arrayInputs"></div>
        <button id="saveArrayBtn" style="display: none" onclick="saveArray()">
          Save Array
        </button>
        <div class="operation">
          <label class="area-label">Remove Element:</label>
          <input
            type="text"
            id="removeInput"
            placeholder="Enter index to remove"
          />
          <button onclick="removeElement()">Remove</button>
        </div>

        <div class="operation">
          <label class="area-label">Check Element:</label>
```

```
            <input
              type="text"
              id="checkInput"
              placeholder="Enter value to check"
            />
            <button onclick="checkElement()">Check</button>
          </div>

          <button onclick="emptyArray()">Empty Array</button>
        </div>

        <!-- Result Section -->
        <div class="result-section">
          <div class="area-label">Current Array</div>
          <div class="area-box" id="arrayDisplay">[]</div>
          <div id="message" style="color: #1e40af; font-weight: 600"></div>
        </div>
      </div>

      <h1>Array Preview</h1>
      <!-- Table Section -->
      <table id="arrayTable" style="display: none">
        <thead>
          <tr>
            <th>Index</th>
            <th>Value</th>
          </tr>
        </thead>
        <tbody></tbody>
      </table>
    </body>
```

2. **Javascript:**

```
let arr = [];

// Update array display and table
function updateDisplay(message = "") {
  document.getElementById("arrayDisplay").textContent = JSON.stringify(arr);
```

```
  document.getElementById("message").textContent = message;

  const table = document.getElementById("arrayTable");
  const tbody = table.querySelector("tbody");
  tbody.innerHTML = "";

  if (arr.length > 0) {
    table.style.display = "table";
    arr.forEach((val, index) => {
      const row = document.createElement("tr");
      row.innerHTML = `<td>${index}</td><td>${val}</td>`;
      tbody.appendChild(row);
    });
  } else {
    table.style.display = "none";
  }
}

function saveArray() {
  const size = parseInt(document.getElementById("arraySize").value);
  userArray = [];

  for (let i = 0; i < size; i++) {
    let val = document.getElementById(`element-${i}`).value.trim();

    // Try parsing numbers or arrays
    try {
      val = JSON.parse(val);
    } catch (e) {
      // keep as string if not JSON parsable
    }

    userArray.push(val);
  }

  document.getElementById("arrayDisplay").innerText = JSON.stringify(userArray);
  arr = userArray;
}
```

```
function createArray() {
  const size = parseInt(document.getElementById("arraySize").value);
  const container = document.getElementById("arrayInputs");
  container.innerHTML = "";

  if (isNaN(size) || size <= 0) {
    alert("Please enter a valid array size.");
    return;
  }

  for (let i = 0; i < size; i++) {
    const input = document.createElement("input");
    input.type = "text";
    input.placeholder = `Element ${i + 1}`;
    input.id = `element-${i}`;
    container.appendChild(input);
    container.appendChild(document.createElement("br"));
  }

  document.getElementById("saveArrayBtn").style.display = "inline-block";
}

// Remove element
function removeElement() {
  if (arr.length === 0) {
    updateDisplay("⚠ Array is empty. Nothing to remove.");
    return;
  }
  const value = document.getElementById("removeInput").value.trim();
  if (!value) {
    updateDisplay("⚠ Enter an index to remove.");
    return;
  }
  const index = value;
  if (index !== -1) {
    arr.splice(index, 1);
    updateDisplay(`🗑 Removed index '${value}' from the array.`);
```

```
  } else {
    updateDisplay(`✗ index '${value}' not found in the array.`);
  }
}

// Check element
function checkElement() {
  if (arr.length === 0) {
    updateDisplay("⚠ Array is empty. Nothing to check.");
    return;
  }
  const rawValue = document.getElementById("checkInput").value.trim();
  const numValue = Number(rawValue);
  if (!rawValue) {
    updateDisplay("⚠ Enter a value to check.");
    return;
  }
  if (arr.includes(rawValue) || arr.includes(numValue)) {
    updateDisplay(`☑ '${rawValue}' is present in the array.`);
  } else {
    updateDisplay(`✗ '${rawValue}' is NOT in the array.`);
  }
}

// Empty array
function emptyArray() {
  if (arr.length === 0) {
    updateDisplay("⚠ Array is already empty.");
    return;
  }
  arr = [];
  updateDisplay("🧹 Array emptied.");
}
```