HTML Structure

1. **Input for Column Count**:

   ```html
   Copy code
   <label for="columns">Number of Columns:</label>
   <input type="number" id="columns" min="1" value="3">
   <button onclick="createTable()">Create Table</button>
   ```

   - A labeled input field allows the user to specify the number of columns.
   - The `min="1"` ensures the user cannot set a column count below 1.
   - The default value (`value="3"`) is set to 3 columns initially.
   - The button triggers the `createTable()` function to build the table dynamically.

2. **Table Container**:

   ```html
   Copy code
   <table id="dynamicTable">
     <!-- Table will be dynamically generated here -->
   </table>
   ```

   - An empty table element (`<table>`) with an ID `dynamicTable` acts as a placeholder.
   - The content of the table is dynamically generated by JavaScript.

3. **Button to Add Rows**:

   ```html
   Copy code
   <button onclick="addRow()">Add Row</button>
   ```

   - A button triggers the `addRow()` function to append new rows to the table.

CSS Styling

```css
Copy code
table {
  border-collapse: collapse;
  width: 100%;
  margin-top: 20px;
}
table, th, td {
  border: 1px solid black;
}
th, td {
  padding: 8px;
  text-align: center;
}
```

- **border-collapse: collapse** ensures no extra spacing between table cells.
- Borders and padding make the table visually clear.
- The table spans the full width of the page (`width: 100%`).

JavaScript Functions

1. `createTable()`

```javascript
Copy code
function createTable() {
  const columnCount = parseInt(document.getElementById("columns").value, 10);
  const table = document.getElementById("dynamicTable");

  // Clear any existing table content
  table.innerHTML = "";

  // Create the initial table with 3 rows
  for (let i = 0; i < 3; i++) {
    const row = table.insertRow();
    for (let j = 0; j < columnCount; j++) {
      const cell = row.insertCell();
      const input = document.createElement("input");
      input.type = "text";
      cell.appendChild(input);
    }
  }
}
```

- **Step-by-Step**:
  1. **Fetch Column Count**:

     ```javascript
     Copy code
     const columnCount =
     parseInt(document.getElementById("columns").value, 10);
     ```

     Retrieves the column count from the input field and converts it to an integer.

  2. **Clear Existing Table**:

     ```javascript
     Copy code
     table.innerHTML = "";
     ```

     Clears all content in the table before generating a new one. This avoids stacking multiple tables.

  3. **Generate Rows and Columns**:

- The outer `for` loop creates 3 rows (`for (let i = 0; i < 3; i++)`).
- The inner `for` loop creates the required number of cells (columns) in each row.

```javascript
Copy code
const cell = row.insertCell();
const input = document.createElement("input");
input.type = "text";
cell.appendChild(input);
```

Creates an input field (`<input>`) in each cell.

---

2. `addRow()`

```javascript
Copy code
function addRow() {
  const table = document.getElementById("dynamicTable");
  const columnCount = parseInt(document.getElementById("columns").value, 10);

  // Add a new row
  const row = table.insertRow();
  for (let i = 0; i < columnCount; i++) {
    const cell = row.insertCell();
    const input = document.createElement("input");
    input.type = "text";
    cell.appendChild(input);
  }
}
```

- **Step-by-Step**:
    1. **Fetch Column Count**:

        ```javascript
        Copy code
        const columnCount =
        parseInt(document.getElementById("columns").value, 10);
        ```

        Determines how many columns the new row should have.

    2. **Add a New Row**:

        ```javascript
        Copy code
        const row = table.insertRow();
        ```

        Adds a new row to the existing table.

      3. **Add Cells with Input Fields**:
- The inner `for` loop creates cells, just like in the `createTable()` function.
- Each cell contains an input field for user input.

---

3. Table Creation on Page Load

```javascript
Copy code
window.onload = createTable;
```

- Automatically generates the table with 3 columns and 3 rows when the page is loaded.

Working Flow

1. The user enters the number of columns and clicks "Create Table."
   - This generates a table with the specified number of columns and 3 rows, each containing input fields.
2. Clicking "Add Row" appends a new row to the table, with input fields matching the specified number of columns.

Key Concepts Used

1. **DOM Manipulation**:
   - `document.getElementById()` retrieves the input field and table elements.
   - `table.insertRow()` and `row.insertCell()` dynamically add rows and cells.
   - `document.createElement("input")` dynamically creates input fields.
2. **Dynamic Input Handling**:
   - The table adapts to the column count specified by the user.
3. **Responsive Table Update**:
   - Re-creating the table clears old content to reflect updated configurations.