

# Student Management System - Dynamic Data Management Web Application

## Project Description

This is a dynamic web application that demonstrates advanced DOM manipulation, event handling, and responsive interface creation. The system allows users to add or delete a student record. Also there is a feature where the user is also allowed to look at student details by clicking on an expandable arrow button.

## Web site Features

- **Responsive Design:** Works seamlessly on all device types (desktop, tablet, mobile)
- **Interactive Table:** Dynamic student record management with expandable details
- **Real-time Feedback:** Popup messages and visual state changes
- **Dynamic Content:** Add, edit, and delete functionality with automatic numbering

## HTML/HTML5 Elements Used

### Required Elements

#### 1. Semantic Table Structure

- **Complete table implementation** with proper `<thead>`, `<tbody>`, `<th>`, `<tr>`, and `<td>` structure
- **Dynamic row management** for student records and expandable content

#### 2. Interactive Forms and Inputs

- **Checkbox inputs** for row selection with feature to edit or delete the record
- **Text input fields** in edit popup with validation
- **Button elements** with various types (submit, delete, edit, cancel)

#### 3. Dynamic Content Management

- **Expandable rows** with detailed student information
- **Dynamic button creation** based on user selections
- **Real-time DOM manipulation** for adding/removing content

#### 4. Popup Modal System

- **Edit popup** with form elements and action buttons
- **Message popup** for success/error notifications
- **Keyboard navigation** support for accessibility

#### 5. Responsive Layout

- **Mobile-first design** approach
- **Flexible grid system** for expandable content
- **Adaptive styling** for different screen sizes

## HTML Tags Used in Student Management System

### Document Structure Tags

- `<!DOCTYPE html>` - Declares the document type as HTML5
- `<html lang="en">` - Root element with language specification for accessibility
- `<head>` - Container for document metadata (title, styles, meta tags)
- `<body>` - Main content container that users interact with

### Metadata Tags

- `<meta charset="UTF-8">` - Specifies UTF-8 character encoding for proper text display
- `<meta name="viewport" content="width=device-width, initial-scale=1.0">` - Ensures responsive behavior on mobile devices
- `<title>` - Sets the browser tab title for the application
- `<style>` - Contains embedded CSS for styling the entire application in the css file

### Semantic Structure Tags (HTML5)

- `<div class="header">` - Top section containing application title and developer information
- `<div class="controls">` - Container for action buttons (Add Student, Submit)
- `<div class="table-container">` - Wrapper for the main data table with styling
- `<div class="success-message">` - Banner container for success notifications

### Content Organization Tags

- `<div class="...">` - Generic containers for layout and styling purposes
- `<h1>`, `<h3>`, `<h4>` - Hierarchical headings for content structure

- **<p>** - Paragraph elements for popup message content

## Table Structure Tags

- **<table id="studentTable">** - Main container for the student data grid
- **<thead>** - Table header section containing column titles
- **<tbody id="studentTableBody">** - Table body containing dynamic student rows
- **<tr>** - Table rows for both data records and expandable content
- **<th>** - Header cells defining column names and structure
- **<td>** - Data cells containing student information and interactive elements
- **<td colspan="12">** - Spanning cells for expandable content across all columns

## Interactive Form Elements

- **<button class="btn btn-primary" id="addStudentBtn">** - Primary action button for adding new students
- **<button class="btn btn-submit" id="submitBtn" disabled>** - Submit button with dynamic state management
- **<button class="btn btn-delete">** - Dynamically created delete buttons for selected rows
- **<button class="btn btn-edit">** - Dynamically created edit buttons for selected rows
- **<input type="checkbox" class="row-checkbox">** - Selection checkboxes for table rows
- **<input type="text" id="editInput">** - Text input field in edit popup modal

## Media and Interactive Elements

- **** - SVG arrow icons for expand/collapse functionality
- **<form> equivalent functionality** - Handled through JavaScript event management

## Modal and Popup Elements

- **<div class="popup" id="editPopup">** - Modal container for edit functionality
- **<div class="popup-content">** - Content wrapper for popup dialogs
- **<div class="popup-buttons">** - Button container for popup actions

## Data Display and Organization

- **<div class="detail-grid">** - Grid layout for expanded student details

- **<div class="detail-item">** - Individual detail containers in expanded rows
- **<div class="detail-label">** - Labels for student information fields
- **<div class="detail-value">** - Values for student information display

## Accessibility and Semantic Tags

- **class attributes** - CSS styling hooks and JavaScript selectors for functionality
- **id attributes** - Unique identifiers for DOM manipulation and event handling
- **data-student-id attributes** - Custom data storage for student record identification
- **data-row attributes** - Row identification for expand/collapse functionality
- **data-parent attributes** - Relationship mapping for expandable content
- **disabled attributes** - State management for form controls

## Styling and Layout Tags

- **<style> block** - Comprehensive CSS styling including:
  - Responsive grid layouts
  - Gradient backgrounds and modern UI elements
  - Animation keyframes for smooth transitions
  - Media queries for mobile responsiveness
  - Interactive hover effects and state changes

## JavaScript Functions Used

### Main Application Functions

- **setupEventListeners()** - Initializes all event listeners for buttons, checkboxes, keyboard events, and popup interactions
- **addNewStudent()** - Creates new student records with randomized dummy data and maintains sequential ID numbering
- **handleRowSelection(checkbox)** - Manages row selection states, background colors, and dynamic button creation/removal
- **updateSubmitButton()** - Dynamically enables/disables submit button based on checkbox selection count
- **deleteStudent(button)** - Removes student records, updates numbering system, and maintains table order
- **updateStudentNumbers()** - Recalculates and maintains sequential student ID numbering after deletions

## UI Interaction Functions

- **toggleExpandableRow(icon)** - Handles expand/collapse functionality with smooth animations for detailed student information
- **editStudent(button)** - Opens edit popup modal with student-specific title and pre-configured input field
- **handleEditOk()** - Processes edit confirmation, shows success message regardless of input content
- **closeEditPopup()** - Closes the edit popup modal and resets form state

## Popup Management Functions

- **showMessagePopup(message)** - Displays popup messages for add/delete/edit operations with consistent styling
- **closeMessagePopup()** - Closes message popup modal and handles cleanup
- **showSuccessMessage(message, isError)** - Shows banner-style messages for submit functionality

## Event Handler Functions

- **document.addEventListener('DOMContentLoaded')** - Initializes application state and setup on page load
- **document.addEventListener('change')** - Handles all checkbox state changes with event delegation
- **document.addEventListener('click')** - Manages clicks on expand icons, delete buttons, edit buttons with event delegation
- **document.addEventListener('keydown')** - Provides comprehensive keyboard navigation for all popups