**Program No: 01**

**Develop the HTML page named as “Myfirstwebpage.html”. Add the following tags with relevant content.**

1. **Set the title of the page as “My First Web Page”**
2. **Within the body use the following tags:**
3. **Moving text = “Basic HTML Tags”**
4. **Different heading tags (h1 to h6)**
5. **Paragraph**
6. **Horizontal line**
7. **Line Break**
8. **Block Quote**
9. **Pre tag**
10. **Different Logical Style.**

**Code:**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>My First Web Page</title>

<style>

/\* Basic styling for the moving text \*/

.moving-text {

animation: move 30s linear infinite;

}

@keyframes move {

from { transform: translateX(100%); }

to { transform: translateX(-100%); }

}

</style>

</head>

<body>

<!-- Moving text -->

<div class="moving-text">

Basic HTML Tags

</div>

<!-- Different heading tags -->

<h1>Heading Level 1</h1>

<h2>Heading Level 2</h2>

<h3>Heading Level 3</h3>

<h4>Heading Level 4</h4>

<h5>Heading Level 5</h5>

<h6>Heading Level 6</h6>

<!-- Paragraph -->

<p>This is a paragraph of text. It provides a brief explanation or description of the content on the page.</p>

<!-- Horizontal line -->

<hr>

<!-- Line Break -->

<p>Line break below:<br>Notice how the text continues on the next line.</p>

<!-- Block Quote -->

<blockquote>

This is a block quote. It is used to indicate that the text is a quotation from another source. Typically, it is indented from the rest of the text.

</blockquote>

<!-- Pre tag -->

<pre>

This is a preformatted text block.

It maintains whitespace and line breaks

exactly as they are written in the HTML file.

</pre>

<!-- Different Logical Styles -->

<p>

<b>This text is bold.</b><br>

<i>This text is italicized.</i><br>

<u>This text is underlined.</u><br>

<sub>This text is subscript.</sub><br>

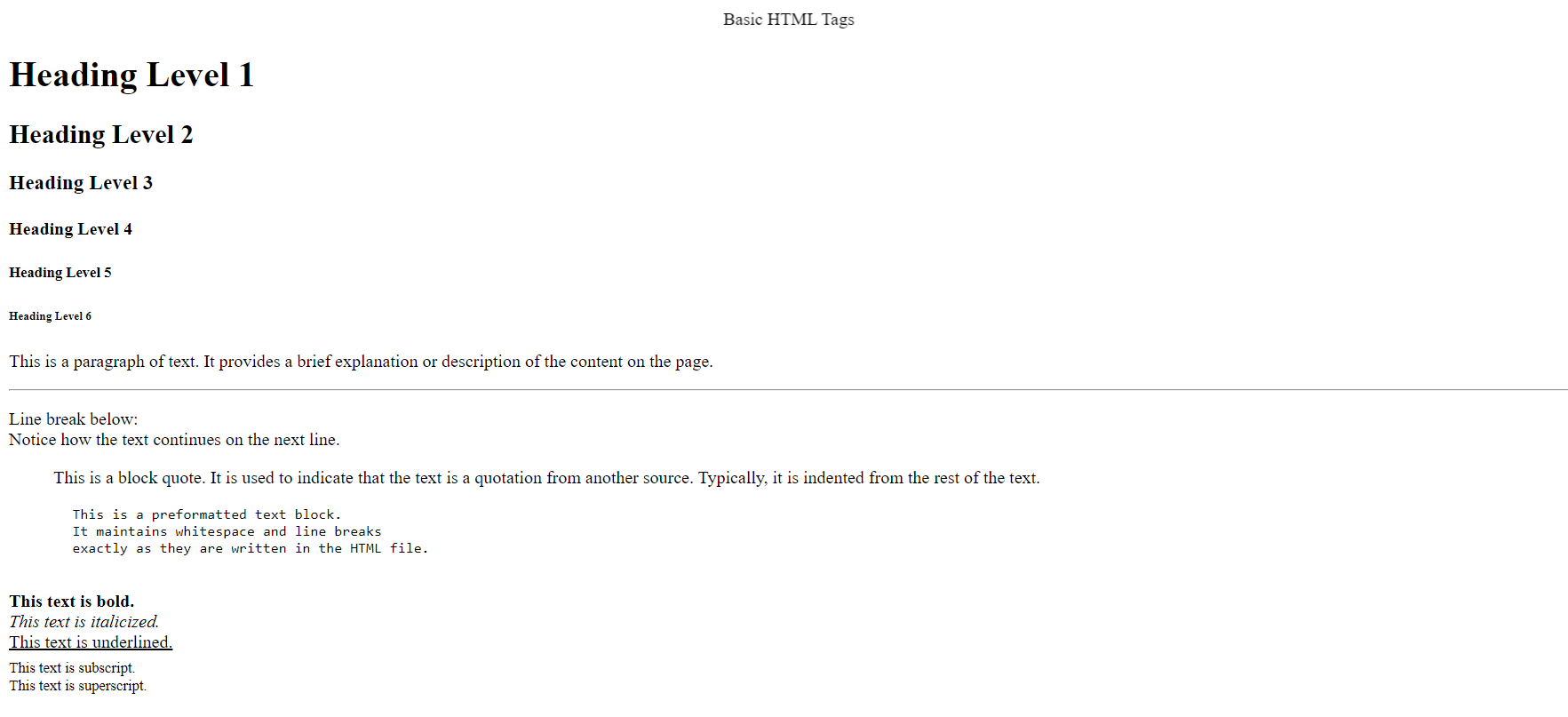
<sup>This text is superscript.</sup>

</p>

</body>

</html>

**Output:**

****

**Program No: 02**

**Develop the HTML page named as “Table.html” to display your class time table.**

1. **Provide the title as Time Table with table header and table footer, row-span and col-span etc.**
2. **Provide various colour options to the cells (Highlight the lab hours and elective hours with different colours.)**
3. **Provide colour options for rows.**

**Code:**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Time Table</title>

<style>

/\* Basic styling for the table \*/

table {

width: 100%;

border-collapse: collapse;

}

th, td {

border: 1px solid #ddd;

padding: 8px;

text-align: center;

}

th {

background-color: #f2f2f2;

}

.header {

background-color: #f9f9f9;

font-weight: bold;

}

.footer {

background-color: #f9f9f9;

font-weight: bold;

}

.lab-hour {

background-color: #e0f7fa; /\* Light cyan \*/

}

.elective-hour {

background-color: #ffecb3; /\* Light yellow \*/

}

.evening {

background-color: #ffe0b2; /\* Light orange \*/

}

.morning {

background-color: #c8e6c9; /\* Light green \*/

}

.highlight-row {

background-color: #d1c4e9; /\* Light purple \*/

}

</style>

</head>

<body>

<h1>Class Time Table</h1>

<table>

<!-- Table Header -->

<thead>

<tr class="header">

<th rowspan="2">Time</th>

<th colspan="5">Days</th>

</tr>

<tr class="header">

<th>Monday</th>

<th>Tuesday</th>

<th>Wednesday</th>

<th>Thursday</th>

<th>Friday</th>

</tr>

</thead>

<!-- Table Body -->

<tbody>

<tr class="highlight-row">

<td>8:00 - 9:00</td>

<td class="morning">Math</td>

<td class="morning">Physics</td>

<td class="morning">Chemistry</td>

<td class="highlight-row lab-hour">Lab</td>

<td class="highlight-row elective-hour">Elective</td>

</tr>

<tr>

<td>9:00 - 10:00</td>

<td class="morning">English</td>

<td class="morning">Math</td>

<td class="morning">Physics</td>

<td class="highlight-row">Free</td>

<td class="highlight-row lab-hour">Lab</td>

</tr>

<tr class="highlight-row">

<td>10:00 - 11:00</td>

<td class="highlight-row elective-hour">Elective</td>

<td class="highlight-row">Free</td>

<td class="highlight-row elective-hour">Elective</td>

<td class="highlight-row">Free</td>

<td class="highlight-row lab-hour">Lab</td>

</tr>

<tr>

<td>11:00 - 12:00</td>

<td class="morning">History</td>

<td class="morning">Geography</td>

<td class="morning">Biology</td>

<td class="evening">Lunch</td>

<td class="evening">Lunch</td>

</tr>

<tr class="highlight-row">

<td>12:00 - 1:00</td>

<td class="highlight-row">Free</td>

<td class="highlight-row lab-hour">Lab</td>

<td class="highlight-row elective-hour">Elective</td>

<td class="highlight-row lab-hour">Lab</td>

<td class="highlight-row elective-hour">Elective</td>

</tr>

</tbody>

<!-- Table Footer -->

<tfoot>

<tr class="footer">

<td colspan="6">End of Timetable</td>

</tr>

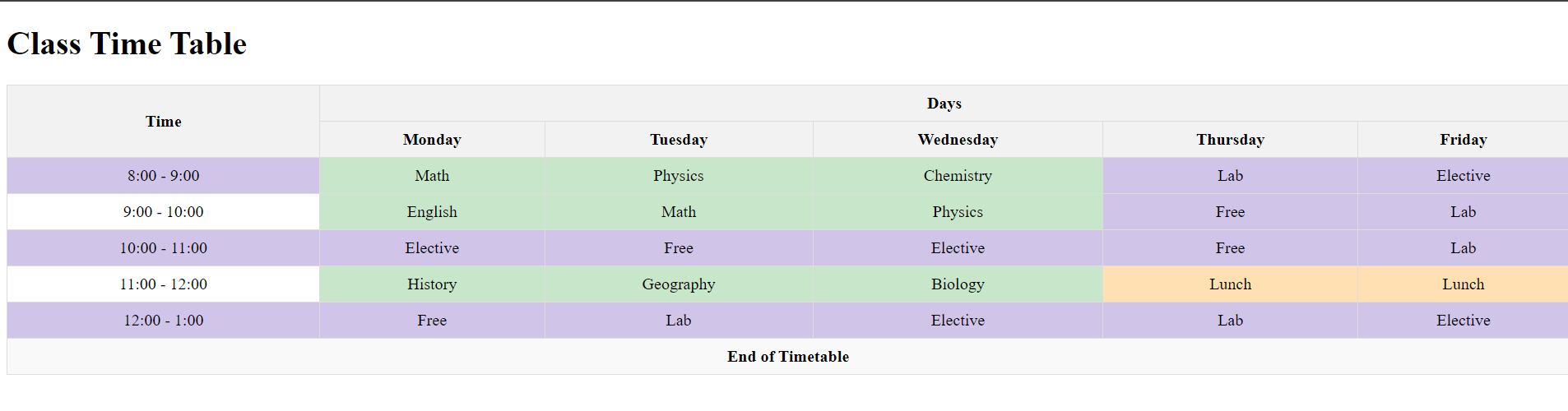
</tfoot>

</table>

</body>

</html>

**Output:**

****

**Program No: 03**

**Develop an external style sheet named as “style.css” and provide different styles for h2, h3, hr, p, div, span, time, img & tags. Apply different CSS selectors for tags and demonstrate the significance of each.**

**Code:**

Create a file pgm03.html

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Styled Page</title>

<link rel="stylesheet" href="styles.css">

</head>

<body>

<h2>Section Header</h2>

<h3>Subsection Header</h3>

<hr>

<p>

This is a paragraph of text to demonstrate the <span>span</span> styling in action.

Notice how different elements are styled based on the <time datetime="2024-09-01">date</time> provided.

</p>

<div>

<p>Content inside a <span>div</span> element with a styled border and background.</p>

<img src="https://via.placeholder.com/400" alt="Placeholder Image">

</div>

<a href="https://www.example.com">Visit Example.com</a>

</body>

</html>

Create a file styles.css

\* {

margin: 0;

padding: 0;

box-sizing: border-box;

}

h2 {

color: #2c3e50;

font-size: 2em;

margin-bottom: 10px;

}

h3 {

color: #34495e;

font-size: 1.5em;

margin-bottom: 8px;

}

hr {

border: 0;

height: 2px;

background-color: #e74c3c;

margin: 20px 0;

}

p {

font-family: 'Arial', sans-serif;

line-height: 1.6;

margin: 10px 0;

}

div {

padding: 15px;

border: 1px solid #bdc3c7;

background-color: #ecf0f1;

}

span {

color: #e67e22;

font-weight: bold;

}

time::before {

content: '⏰ ';

color: #16a085;

}

img {

margin-left: 15px;

height: 300px;

width: 400px;

border-radius: 8px;

box-shadow: 0 4px 8px rgba(0, 0, 0, 0.2);

max-width: 100%;

}

a {

text-decoration: none;

color: #ea0e4c;

}

a:hover {

color: #6200ee;

text-decoration: underline;

}

.highlight {

background-color: yellow;

padding: 3px;

}

.center {

text-align: center;

}

#special-paragraph {

font-size: 1.2em;

color: #8e44ad;

background-color: #f5f5f5;

padding: 10px;

border-left: 5px solid #8e44ad;

}

h2,

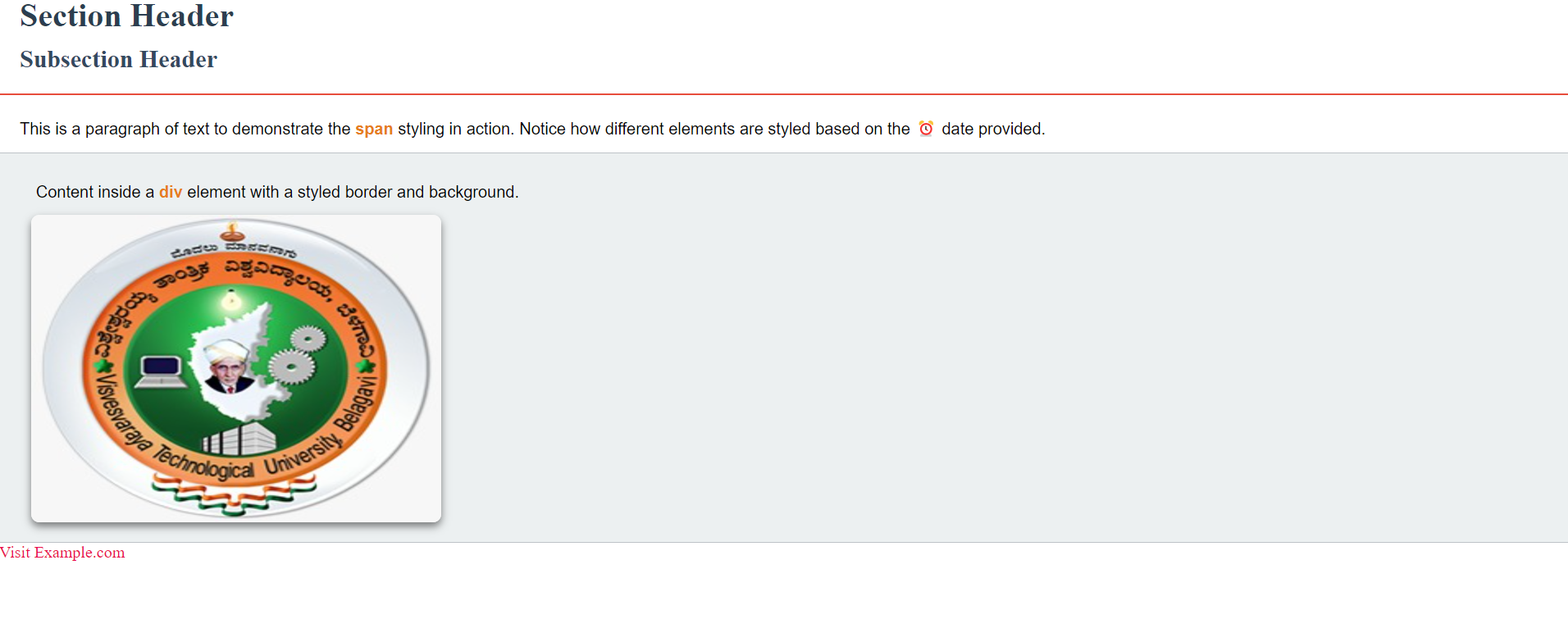
h3,

p {

margin-left: 20px;

}

**Output:**

****

**Program No: 04**

**Develop HTML page named as “registration.html” having variety of HTML input elements with background colors, table for alignment & provide font colors & size using CSS styles.**

**Code:**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Registration Form</title>

<link rel="stylesheet" href="styles.css">

<style>

/\* Inline styles for demonstration \*/

body {

font-family: Arial, sans-serif;

background-color: #f4f4f4; /\* Light grey background for the page \*/

color: #333; /\* Dark grey text color \*/

margin: 20px;

}

table {

width: 100%;

max-width: 600px;

margin: 0 auto;

border-collapse: collapse;

}

td {

padding: 10px;

vertical-align: top;

}

label {

display: block;

margin-bottom: 5px;

font-weight: bold;

color: #555; /\* Darker grey for labels \*/

}

input, select, textarea {

width: 100%;

padding: 8px;

border: 1px solid #ccc;

border-radius: 4px;

}

input[type="text"], input[type="email"], input[type="password"], select {

background-color: #ffffff; /\* White background for text inputs \*/

}

input[type="submit"] {

background-color: #4CAF50; /\* Green background for submit button \*/

color: white; /\* White text color \*/

border: none;

cursor: pointer;

font-size: 16px;

}

input[type="submit"]:hover {

background-color: #45a049; /\* Darker green on hover \*/

}

textarea {

background-color: #ffffff; /\* White background for textarea \*/

resize: vertical; /\* Allow vertical resizing \*/

}

.form-container {

background-color: #ffffff; /\* White background for the form container \*/

border: 1px solid #ddd;

border-radius: 8px;

padding: 20px;

box-shadow: 0px 0px 10px rgba(0, 0, 0, 0.1); /\* Subtle shadow effect \*/

}

</style>

</head>

<body>

<h1 style="text-align: center;">Registration Form</h1>

<div class="form-container">

<form action="#" method="post">

<table>

<tr>

<td><label for="firstName">First Name:</label></td>

<td><input type="text" id="firstName" name="firstName" required></td>

</tr>

<tr>

<td><label for="lastName">Last Name:</label></td>

<td><input type="text" id="lastName" name="lastName" required></td>

</tr>

<tr>

<td><label for="email">Email:</label></td>

<td><input type="email" id="email" name="email" required></td>

</tr>

<tr>

<td><label for="password">Password:</label></td>

<td><input type="password" id="password" name="password" required></td>

</tr>

<tr>

<td><label for="gender">Gender:</label></td>

<td>

<select id="gender" name="gender" required>

<option value="">Select</option>

<option value="male">Male</option>

<option value="female">Female</option>

<option value="other">Other</option>

</select>

</td>

</tr>

<tr>

<td><label for="comments">Comments:</label></td>

<td><textarea id="comments" name="comments" rows="4"></textarea></td>

</tr>

<tr>

<td colspan="2" style="text-align: center;">

<input type="submit" value="Register">

</td>

</tr>

</table>

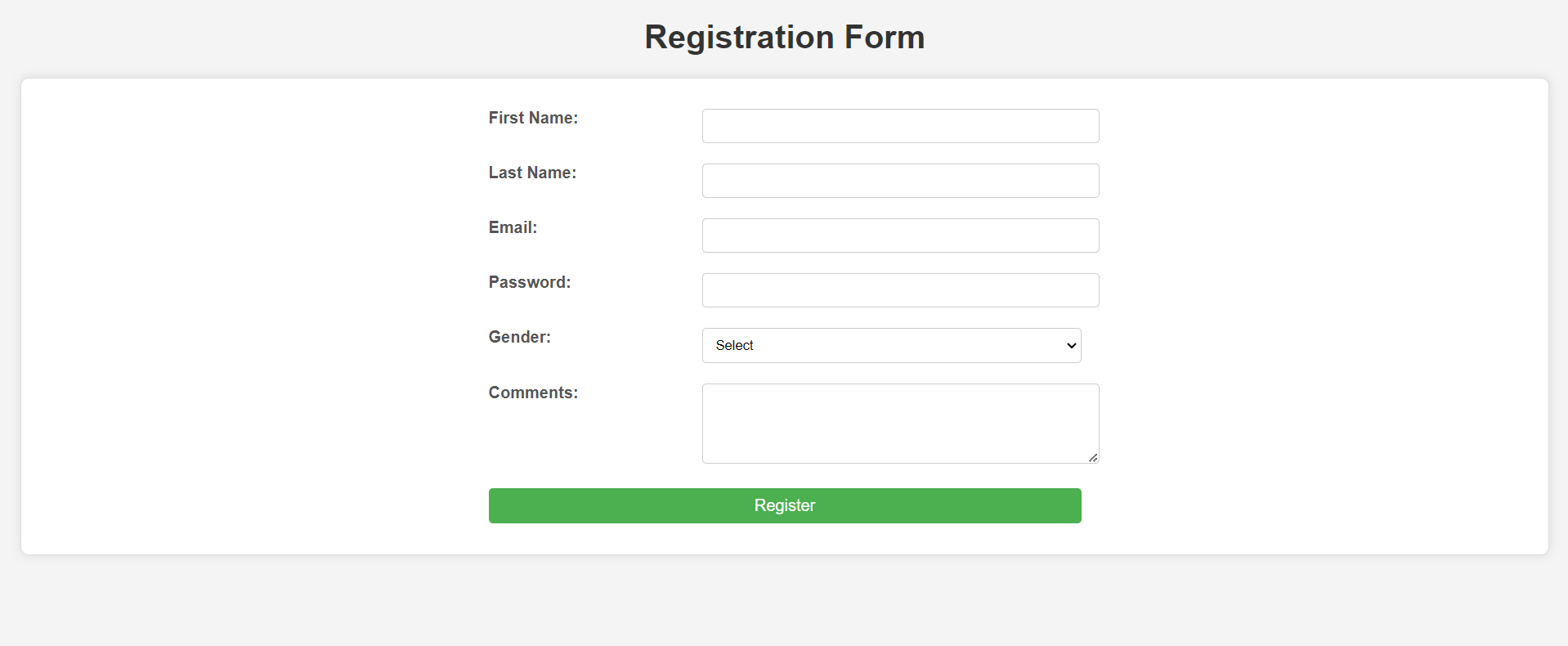
</form>

</div>

</body>

</html>

**Output:**

****

**Program No: 05**

**Develop HTML page named as “newpaper.html” having variety of HTML semantic elements with background colors, text-colors & size for figure, table, aside, section, article, header, footer… etc.**

**Code:**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Newspaper Layout</title>

<link rel="stylesheet" href="styles.css">

<style>

/\* Inline styles for demonstration \*/

body {

font-family: 'Arial', sans-serif;

background-color: #f0f0f0; /\* Light grey background for the page \*/

color: #333; /\* Dark grey text color \*/

margin: 0;

padding: 0;

}

header {

background-color: #333; /\* Dark background for header \*/

color: white; /\* White text color \*/

padding: 20px;

text-align: center;

}

nav {

margin: 10px 0;

text-align: center;

}

nav a {

color: #f0f0f0; /\* Light color for links \*/

text-decoration: none;

margin: 0 15px;

}

nav a:hover {

text-decoration: underline; /\* Underline on hover \*/

}

.container {

display: flex;

flex-wrap: wrap;

margin: 20px;

}

main {

flex: 3;

padding: 20px;

background-color: #ffffff; /\* White background for the main content \*/

margin-right: 20px;

border-radius: 8px;

box-shadow: 0px 0px 10px rgba(0, 0, 0, 0.1); /\* Subtle shadow effect \*/

}

aside {

flex: 1;

padding: 20px;

background-color: #e8e8e8; /\* Light grey background for sidebar \*/

border-radius: 8px;

box-shadow: 0px 0px 10px rgba(0, 0, 0, 0.1); /\* Subtle shadow effect \*/

}

section {

margin-bottom: 20px;

}

article {

background-color: #fafafa; /\* Very light grey background for articles \*/

padding: 15px;

margin-bottom: 20px;

border-radius: 8px;

box-shadow: 0px 0px 5px rgba(0, 0, 0, 0.1); /\* Subtle shadow effect \*/

}

footer {

background-color: #333; /\* Dark background for footer \*/

color: white; /\* White text color \*/

padding: 10px;

text-align: center;

position: relative;

bottom: 0;

width: 100%;

}

table {

width: 100%;

border-collapse: collapse;

margin-top: 20px;

}

th, td {

border: 1px solid #ddd;

padding: 8px;

text-align: left;

}

th {

background-color: #4CAF50; /\* Green background for table headers \*/

color: white; /\* White text color for table headers \*/

}

td {

background-color: #ffffff; /\* White background for table cells \*/

}

figure {

margin: 20px 0;

padding: 10px;

background-color: #e0e0e0; /\* Light grey background for figures \*/

border-radius: 8px;

box-shadow: 0px 0px 5px rgba(0, 0, 0, 0.1); /\* Subtle shadow effect \*/

}

figcaption {

text-align: center;

font-style: italic;

color: #666; /\* Grey text color for captions \*/

}

h1, h2, h3 {

color: #333; /\* Dark grey text color for headings \*/

}

h1 {

font-size: 2em;

margin-top: 0;

}

h2 {

font-size: 1.5em;

margin-bottom: 0;

}

h3 {

font-size: 1.2em;

margin-bottom: 0;

}

</style>

</head>

<body>

<header>

<h1>Newspaper Daily</h1>

<nav>

<a href="#">Home</a>

<a href="#">World</a>

<a href="#">Politics</a>

<a href="#">Business</a>

<a href="#">Culture</a>

</nav>

</header>

<div class="container">

<main>

<section>

<h2>Top Stories</h2>

<article>

<h3>Breaking News: Major Event</h3>

<p>Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus imperdiet, nulla in iaculis interdum, nulla orci tempus orci, at ullamcorper dui purus in mauris.</p>

</article>

<article>

<h3>Another Headline</h3>

<p>Vestibulum bibendum, ligula eu convallis fermentum, augue eros posuere erat, ac sollicitudin purus nunc eget metus.</p>

</article>

</section>

<section>

<h2>World News</h2>

<figure>

<img src="https://via.placeholder.com/600x400" alt="World News Image" style="width: 100%; height: auto;">

<figcaption>World News Image Caption</figcaption>

</figure>

<table>

<thead>

<tr>

<th>Country</th>

<th>News</th>

</tr>

</thead>

<tbody>

<tr>

<td>Country A</td>

<td>Headline for Country A</td>

</tr>

<tr>

<td>Country B</td>

<td>Headline for Country B</td>

</tr>

</tbody>

</table>

</section>

</main>

<aside>

<h2>Trending</h2>

<p>Check out the latest trends and popular articles of the day.</p>

</aside>

</div>

<footer>

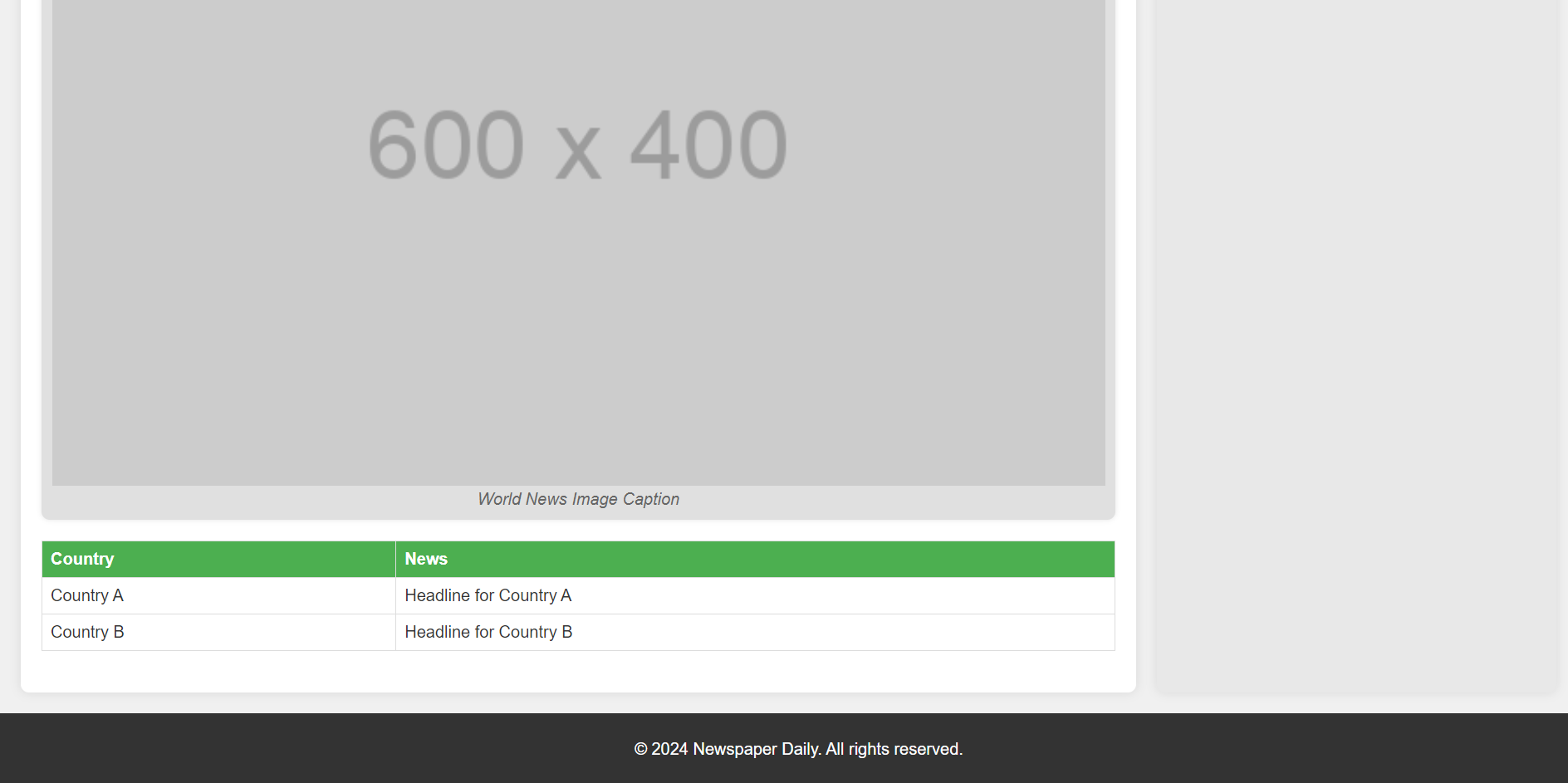
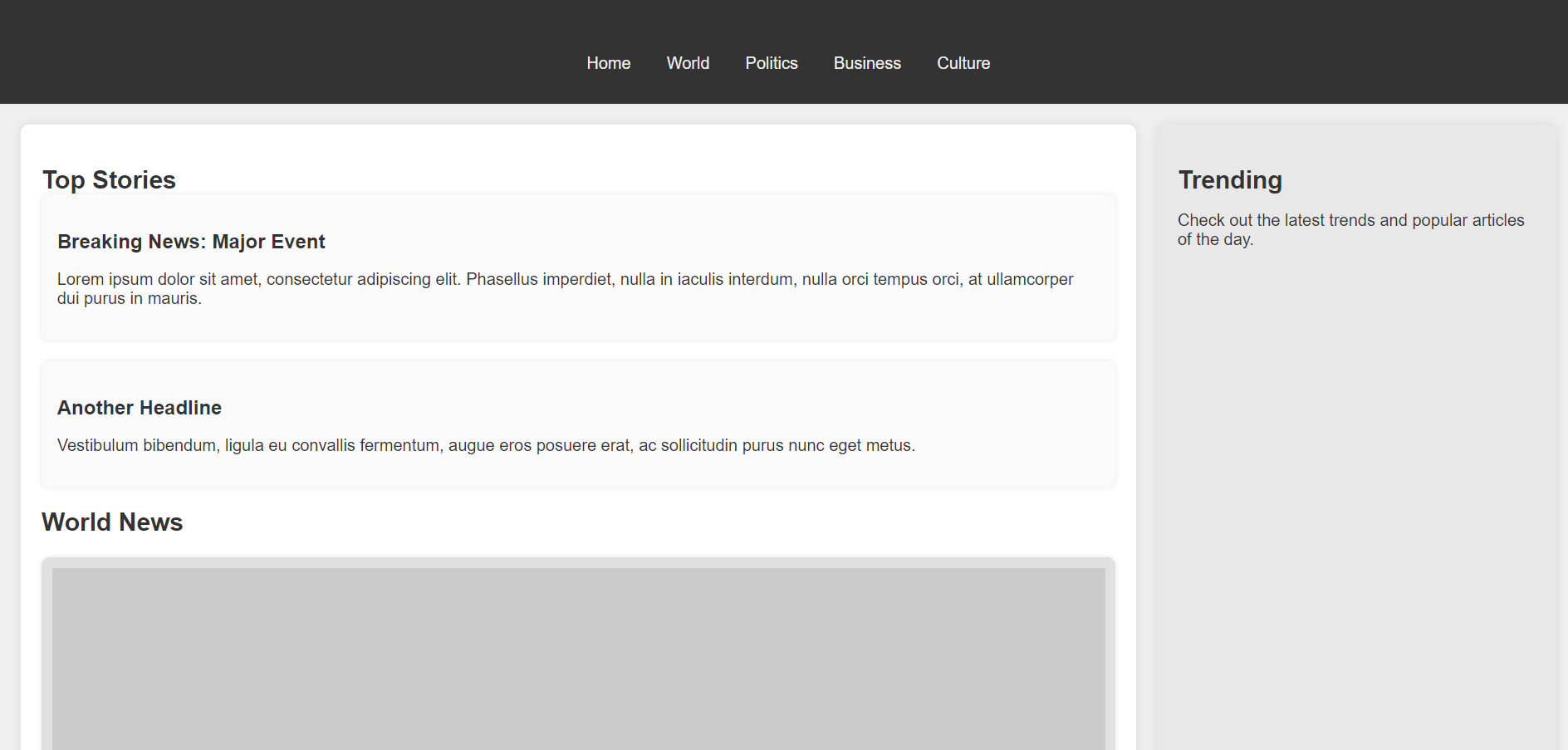
<p>&copy; 2024 Newspaper Daily. All rights reserved.</p>

</footer>

</body>

</html>

**Output:**

****

**Program No: 06**

**Apply HTML, CSS and JavaScript to design a simple calculator to perform the following operations: sum, product, difference, remainder, quotient, power, square-root and square.**

**Code:**

pgm06.html

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Simple Calculator</title>

<link rel="stylesheet" href="styles.css">

</head>

<body>

<div class="calculator">

<h1>Simple Calculator</h1>

<form id="calculator-form">

<input type="number" id="num1" placeholder="Enter first number" required>

<input type="number" id="num2" placeholder="Enter second number" required>

<div class="operations">

<button type="button" onclick="calculate('sum')">Sum</button>

<button type="button" onclick="calculate('product')">Product</button>

<button type="button" onclick="calculate('difference')">Difference</button>

<button type="button" onclick="calculate('remainder')">Remainder</button>

<button type="button" onclick="calculate('quotient')">Quotient</button>

<button type="button" onclick="calculate('power')">Power</button>

<button type="button" onclick="calculate('squareRoot')">Square Root</button>

<button type="button" onclick="calculate('square')">Square</button>

</div>

</form>

<div id="result">

<h2>Result:</h2>

<p id="result-output"></p>

</div>

</div>

<script src="scripts.js"></script>

</body>

</html>

styles.css

body {

font-family: Arial, sans-serif;

background-color: #f4f4f4;

display: flex;

justify-content: center;

align-items: center;

height: 100vh;

margin: 0;

}

.calculator {

background: #fff;

padding: 20px;

border-radius: 8px;

box-shadow: 0px 0px 15px rgba(0, 0, 0, 0.2);

width: 300px;

}

h1 {

text-align: center;

color: #333;

}

input[type="number"] {

width: calc(100% - 20px);

padding: 10px;

margin: 10px 0;

border: 1px solid #ddd;

border-radius: 4px;

}

.operations {

display: flex;

flex-wrap: wrap;

justify-content: space-between;

}

button {

background-color: #4CAF50;

color: white;

border: none;

padding: 10px;

border-radius: 4px;

cursor: pointer;

width: 48%;

margin: 4px 0;

font-size: 16px;

}

button:hover {

background-color: #45a049;

}

#result {

margin-top: 20px;

}

#result-output {

font-size: 1.2em;

color: #333;

}

scripts.js

function calculate(operation) {

// Get input values

const num1 = parseFloat(document.getElementById('num1').value);

const num2 = parseFloat(document.getElementById('num2').value);

// Initialize result variable

let result;

// Perform calculations based on the operation

switch (operation) {

case 'sum':

result = num1 + num2;

break;

case 'product':

result = num1 \* num2;

break;

case 'difference':

result = num1 - num2;

break;

case 'remainder':

result = num1 % num2;

break;

case 'quotient':

result = num1 / num2;

break;

case 'power':

result = Math.pow(num1, num2);

break;

case 'squareRoot':

result = Math.sqrt(num1);

break;

case 'square':

result = Math.pow(num1, 2);

break;

default:

result = 'Invalid operation';

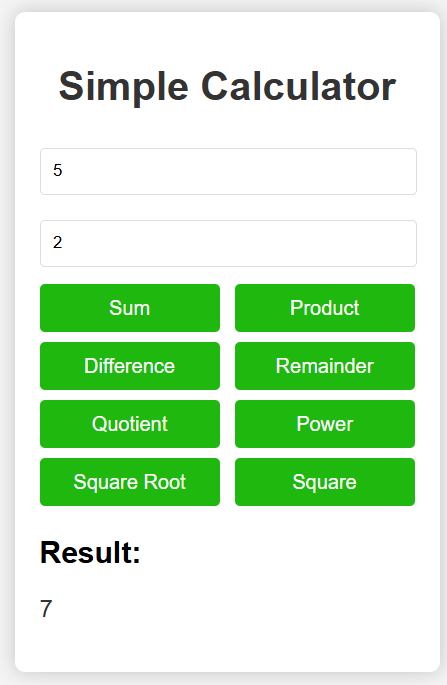
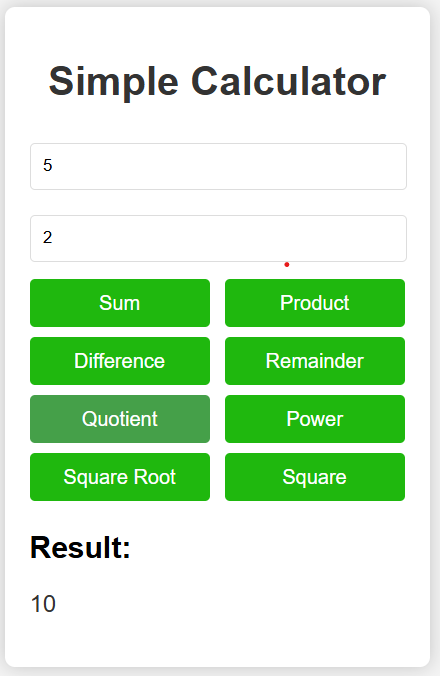
}

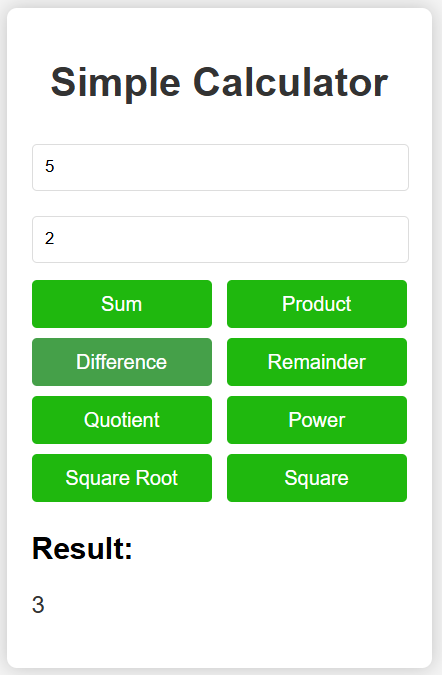
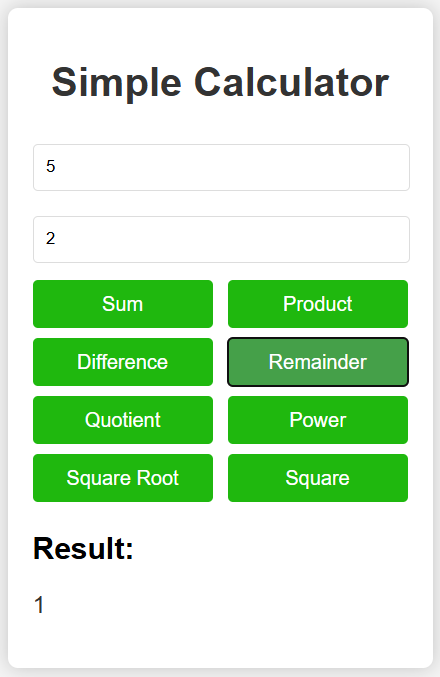
// Display the result

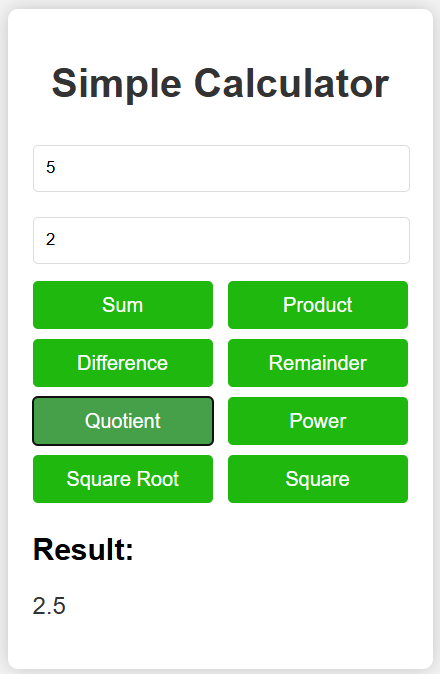
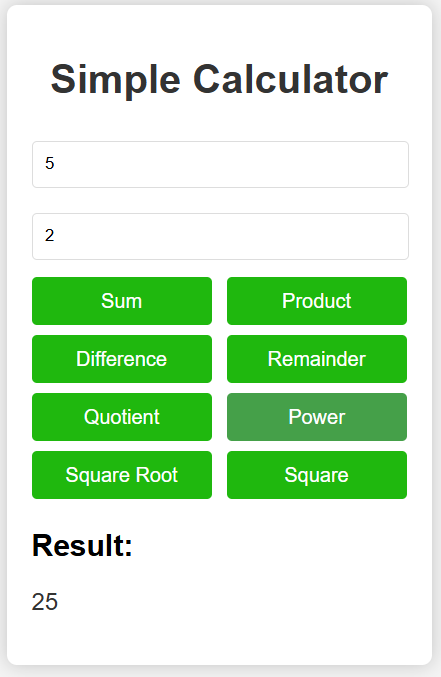
document.getElementById('result-output').innerText = result;

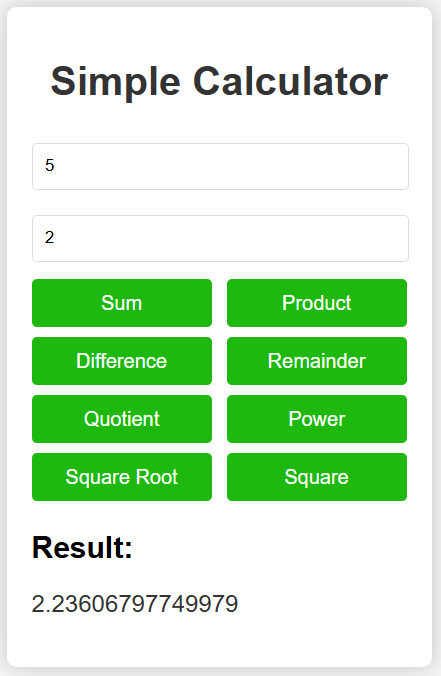
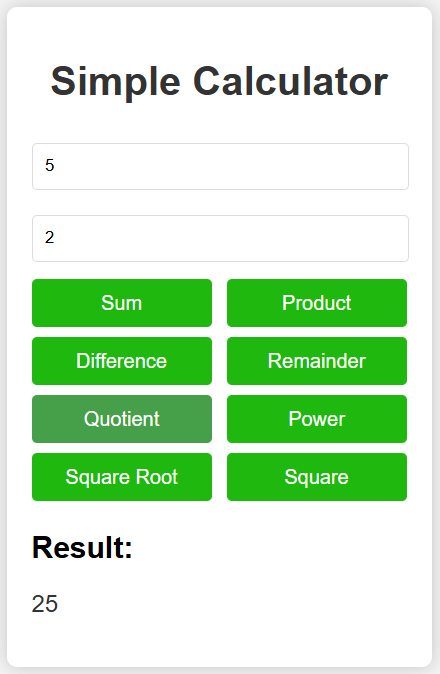
}

**Output:**

** **

** **

** **

** **

**Program No: 07**

**Develop JavaScript program (with HTML/CSS) for:**

* 1. **Converting JSON text to JavaScript Object**
  2. **Convert JSON results into a date**
  3. **Converting From JSON To CSV and CSV to JSON**
  4. **Create hash from string using crypto.createHash() method.**

**Code:**

pgm07.html

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>JSON and Hash Operations</title>

<link rel="stylesheet" href="styles.css">

</head>

<body>

<div class="container">

<h1>JSON and Hash Operations</h1>

<section>

<h1>1. Convert JSON Text to JavaScript Object</h1>

<textarea id="jsonInput" placeholder="Enter JSON here..."></textarea>

<button onclick="convertJSON()">Convert to Object</button>

<div id="output"></div>

</section>

<section>

<h2>2. Convert JSON Results into a Date</h2>

<textarea id="jsonDateInput" rows="5" placeholder='Enter JSON date string here'></textarea>

<button onclick="convertJsonToDate()">Convert to Date</button>

<pre id="dateResult"></pre>

</section>

<section>

<h2>3. Convert JSON to CSV and CSV to JSON</h2>

<textarea id="jsonCsvInput" rows="5" placeholder='Enter JSON array here'></textarea>

<button onclick="jsonToCsv()">JSON to CSV</button>

<pre id="csvResult"></pre>

<textarea id="csvInput" rows="5" placeholder='Enter CSV here'></textarea>

<button onclick="csvToJson()">CSV to JSON</button>

<pre id="jsonResultFromCsv"></pre>

</section>

<section>

<h2>4. Create Hash from String</h2>

<input type="text" id="hashInput" placeholder="Enter text to hash">

<button onclick="createHash()">Create Hash</button>

<pre id="hashResult"></pre>

</section>

</div>

<script src="scripts.js"></script>

</body>

</html>

styles.css

body {

font-family: Arial, sans-serif;

background-color: #f9f9f9;

margin: 0;

padding: 0;

}

.container {

width: 80%;

max-width: 800px;

margin: 20px auto;

padding: 20px;

background: #fff;

border-radius: 8px;

box-shadow: 0px 0px 10px rgba(0, 0, 0, 0.1);

}

h1, h2 {

color: #333;

}

textarea, input[type="text"] {

width: 100%;

padding: 10px;

margin: 10px 0;

border: 1px solid #ddd;

border-radius: 4px;

box-sizing: border-box;

}

button {

padding: 10px;

border: none;

border-radius: 4px;

color: #fff;

background-color: #4CAF50;

cursor: pointer;

}

button:hover {

background-color: #45a049;

}

pre {

background: #f1f1f1;

padding: 10px;

border: 1px solid #ddd;

border-radius: 4px;

overflow: auto;

white-space: pre-wrap;

}

scripts.js

// Function to convert JSON text to JavaScript Object

function convertJSON() {

const input = document.getElementById('jsonInput').value;

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

try {

// Parse the JSON input

const obj = JSON.parse(input);

// Convert the JavaScript object to a natural format string

const formattedObject = formatObject(obj);

// Display the result

output.innerHTML = `

<h3>JavaScript Object:</h3>

<pre>${formattedObject}</pre>

`;

} catch (error) {

// Handle invalid JSON input

output.innerHTML = `

<h3 style="color: red;">Error:</h3>

<p>${error.message}</p>

`;

}

}

function formatObject(obj, indent = 0) {

const spaces = ' '.repeat(indent);

if (typeof obj === 'object' && !Array.isArray(obj) && obj !== null) {

let result = '{\n';

for (let key in obj) {

result += `${spaces} ${key}: ${formatObject(obj[key], indent + 2)},\n`;

}

result += `${spaces}}`;

return result;

} else if (Array.isArray(obj)) {

return `[${obj.map(item => formatObject(item, indent + 2)).join(', ')}]`;

} else if (typeof obj === 'string') {

return `"${obj}"`;

} else {

return obj;

}

}

// Function to convert JSON string to Date

function convertJsonToDate() {

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

try {

const jsonObject = JSON.parse(jsonDateInput);

const date = new Date(jsonObject.date);

document.getElementById('dateResult').textContent = date.toString();

} catch (error) {

document.getElementById('dateResult').textContent = 'Invalid JSON or Date Format';

}

}

// Function to convert JSON to CSV

function jsonToCsv() {

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

try {

const jsonArray = JSON.parse(jsonCsvInput);

const csv = jsonArray.map(row => Object.values(row).join(',')).join('\n');

document.getElementById('csvResult').textContent = csv;

} catch (error) {

document.getElementById('csvResult').textContent = 'Invalid JSON';

}

}

// Function to convert CSV to JSON

function csvToJson() {

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

try {

const rows = csvInput.trim().split('\n');

const headers = rows[0].split(',');

const json = rows.slice(1).map(row => {

const values = row.split(',');

return headers.reduce((obj, header, index) => {

obj[header] = values[index];

return obj;

}, {});

});

document.getElementById('jsonResultFromCsv').textContent = JSON.stringify(json, null, 2);

} catch (error) {

document.getElementById('jsonResultFromCsv').textContent = 'Invalid CSV';

}

}

// Function to create a hash from a string using Web Crypto API

async function createHash() {

const text = document.getElementById('hashInput').value;

const encoder = new TextEncoder();

const data = encoder.encode(text);

const hashBuffer = await crypto.subtle.digest('SHA-256', data);

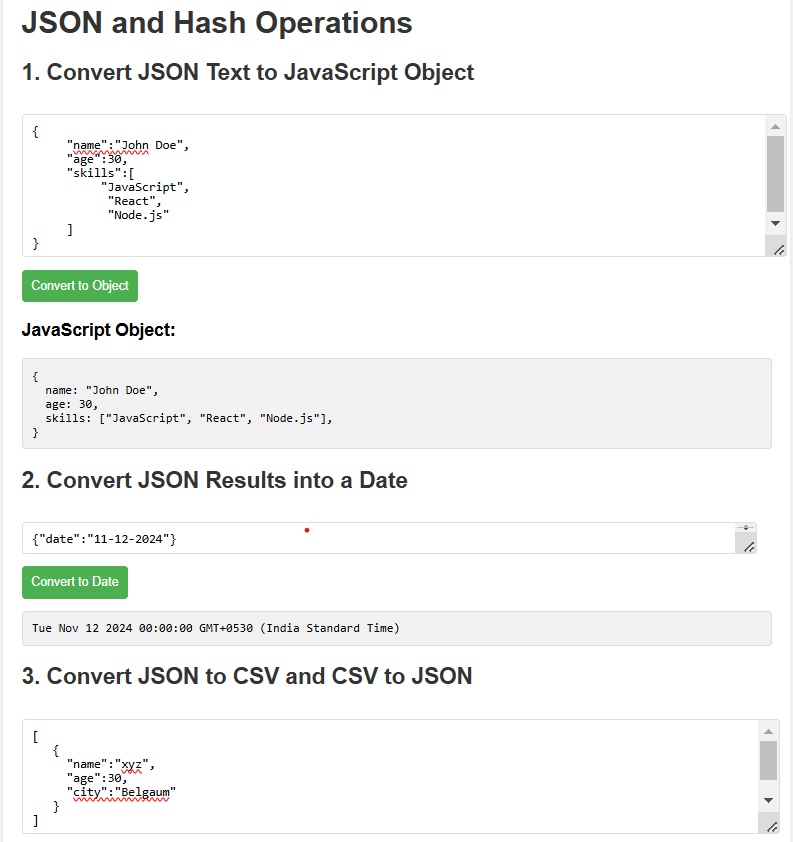
const hashArray = Array.from(new Uint8Array(hashBuffer));

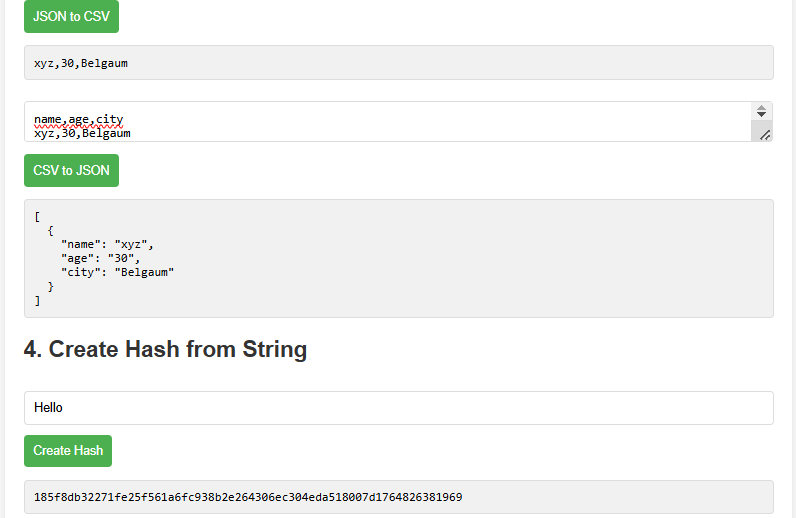
const hashHex = hashArray.map(b => b.toString(16).padStart(2, '0')).join('');

document.getElementById('hashResult').textContent = hashHex;

}

**Output:**

****

****

**Program No: 08**

**A) Develop a PHP program (with HTML/CSS) to keep track of the number of visitors visiting the web page and to display this count of visitors, with relevant headings.**

**Code:**

pgm08a.php

<?php

// Define the file where the visitor count will be stored

$countFile = 'visitor\_count.txt';

// Check if the file exists; if not, create it and set the initial count to 0

if (!file\_exists($countFile)) {

file\_put\_contents($countFile, '0');

}

// Read the current count from the file

$count = (int)file\_get\_contents($countFile);

// Increment the count

$count++;

// Write the updated count back to the file

file\_put\_contents($countFile, $count);

// Display the visitor count

?>

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Visitor Counter</title>

<link rel="stylesheet" href="styles.css">

</head>

<body>

<div class="container">

<h1>Visitor Counter</h1>

<p>Welcome to our website!</p>

<h2>Visitor Count:</h2>

<p class="count"><?php echo $count; ?></p>

</div>

</body>

</html>

style.css

body {

font-family: Arial, sans-serif;

background-color: #f4f4f4;

margin: 0;

padding: 0;

display: flex;

justify-content: center;

align-items: center;

height: 100vh;

}

.container {

background: #fff;

padding: 20px;

border-radius: 8px;

box-shadow: 0px 0px 10px rgba(0, 0, 0, 0.1);

width: 300px;

text-align: center;

}

h1 {

color: #333;

}

h2 {

color: #4CAF50;

}

.count {

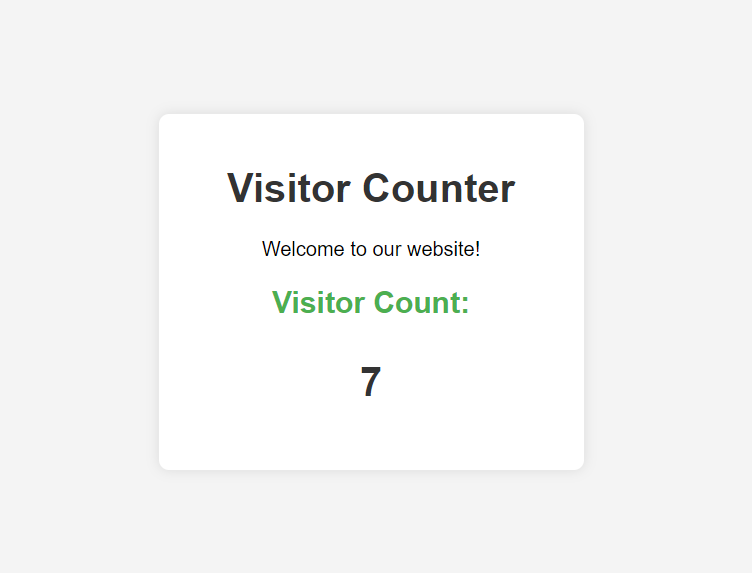
font-size: 2em;

color: #333;

font-weight: bold;

}

**Output:**

****

**Program No: 08**

**B)** **Develop a PHP program (with HTML/CSS) to sort the student records which are stored in the database using selection sort.**

**Code:**

<?php

// Database connection details

$servername = "localhost";

$username = "root"; // replace with your DB username

$password = ""; // replace with your DB password

$dbname = "school"; // replace with your DB name

// Create connection

$conn = new mysqli($servername, $username, $password, $dbname);

// Check connection

if ($conn->connect\_error) {

die("Connection failed: " . $conn->connect\_error);

}

// Fetch student records

$sql = "SELECT id, name, grade FROM students";

$result = $conn->query($sql);

$students = [];

if ($result->num\_rows > 0) {

// Fetch data into an array

while ($row = $result->fetch\_assoc()) {

$students[] = $row;

}

}

// Selection sort function

function selectionSort(&$array, $key) {

$n = count($array);

for ($i = 0; $i < $n - 1; $i++) {

$minIndex = $i;

for ($j = $i + 1; $j < $n; $j++) {

if ($array[$j][$key] < $array[$minIndex][$key]) {

$minIndex = $j;

}

}

if ($minIndex != $i) {

$temp = $array[$i];

$array[$i] = $array[$minIndex];

$array[$minIndex] = $temp;

}

}

}

// Sort students by grade

selectionSort($students, 'grade');

// Close connection

$conn->close();

?>

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Student Records</title>

<style>

body {

font-family: Arial, sans-serif;

margin: 0;

padding: 0;

background-color: #f4f4f4;

}

.container {

width: 80%;

margin: auto;

overflow: hidden;

}

table {

width: 100%;

border-collapse: collapse;

margin: 20px 0;

}

table, th, td {

border: 1px solid #ddd;

}

th, td {

padding: 12px;

text-align: left;

}

th {

background-color: #4CAF50;

color: white;

}

tr:nth-child(even) {

background-color: #f2f2f2;

}

h1 {

text-align: center;

color: #333;

}

</style>

</head>

<body>

<div class="container">

<h1>Sorted Student Records</h1>

<table>

<thead>

<tr>

<th>ID</th>

<th>Name</th>

<th>Grade</th>

</tr>

</thead>

<tbody>

<?php foreach ($students as $student): ?>

<tr>

<td><?php echo htmlspecialchars($student['id']); ?></td>

<td><?php echo htmlspecialchars($student['name']); ?></td>

<td><?php echo htmlspecialchars($student['grade']); ?></td>

</tr>

<?php endforeach; ?>

</tbody>

</table>

</div>

</body>

</html>

students.sql

CREATE TABLE students (

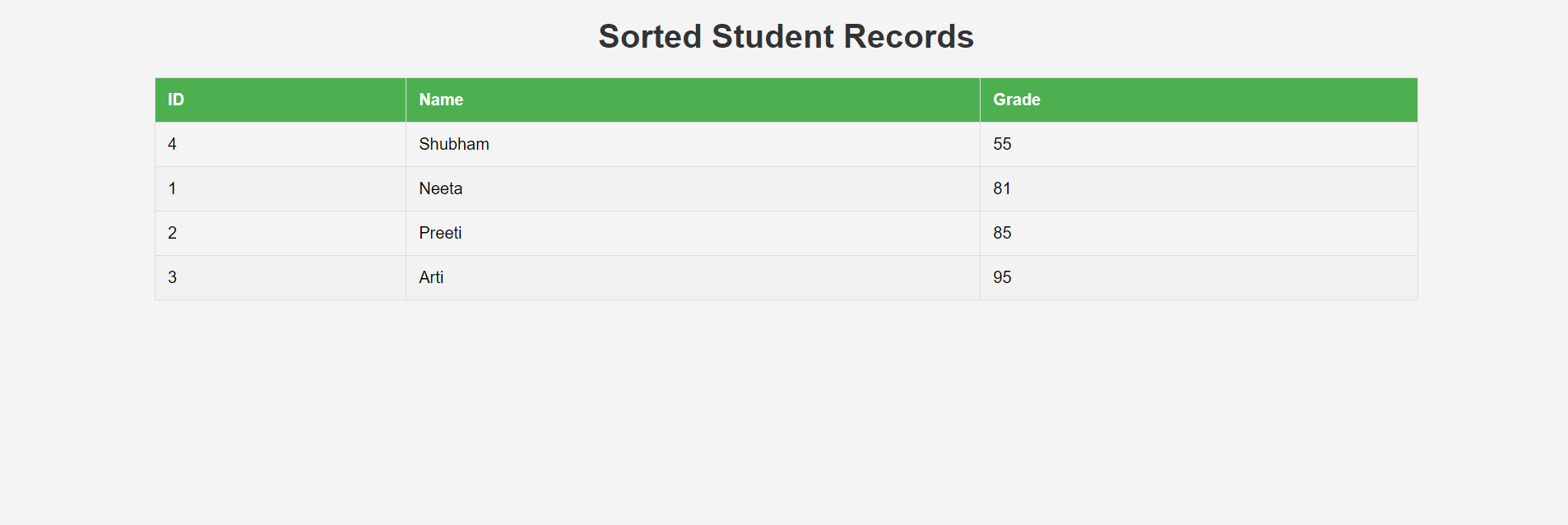
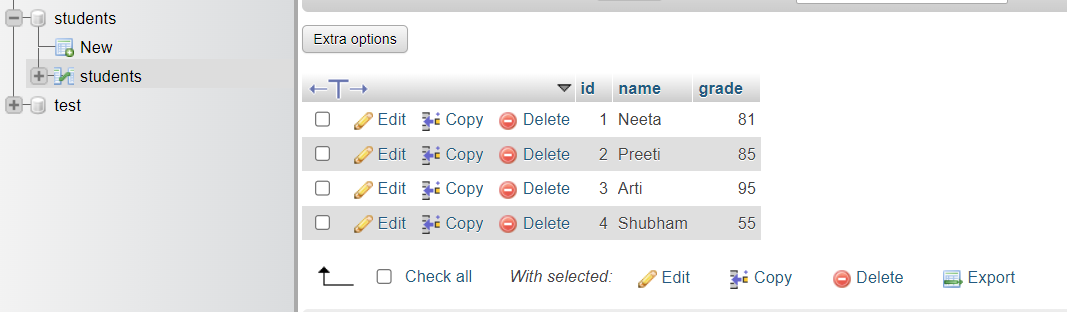
id INT AUTO\_INCREMENT PRIMARY KEY,

name VARCHAR(100),

grade INT

);

**Output:**



**Program No: 09**

**Develop jQuery script (with HTML/CSS) for:**

**a. Appends the content at the end of the existing paragraph and list.**

**b. Change the state of the element with CSS style using animate() method**

**c. Change the color of any div that is animated.**

**Code:**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>jQuery Example</title>

<style>

body {

font-family: Arial, sans-serif;

margin: 20px;

}

p, ul {

font-size: 18px;

}

.animate-div {

width: 200px;

height: 100px;

background-color: lightblue;

margin: 20px 0;

border: 2px solid #007BFF;

transition: background-color 1s; /\* Smooth color transition \*/

}

.highlight {

background-color: yellow;

}

.hidden {

display: none;

}

</style>

<script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>

<script>

$(document).ready(function() {

// Append content to paragraph and list

$('#appendParagraph').click(function() {

$('#myParagraph').append(' <b>Appended text to the paragraph.</b>');

});

$('#appendList').click(function() {

$('#myList').append('<li>Appended list item</li>');

});

// Animate div and change its color

$('#animateDiv').click(function() {

$('.animate-div')

.animate({ width: '300px', height: '150px' }, 1000)

.addClass('highlight')

.fadeOut(500)

.fadeIn(500)

.queue(function(next) {

$(this).removeClass('highlight');

next();

});

});

});

</script>

</head>

<body>

<h1>jQuery Animation and Content Manipulation</h1>

<p id="myParagraph">This is a paragraph.</p>

<button id="appendParagraph">Append Text to Paragraph</button>

<ul id="myList">

<li>First item</li>

<li>Second item</li>

</ul>

<button id="appendList">Append Item to List</button>

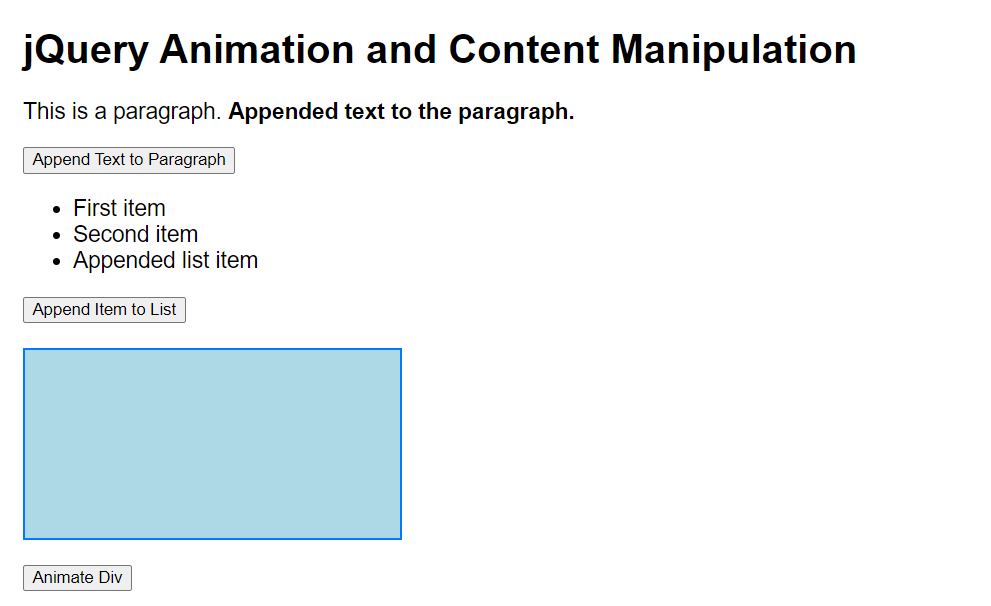
<div class="animate-div"></div>

<button id="animateDiv">Animate Div</button>

</body>

</html>

**Output:**

****

**Program No: 10**

**Develop a JavaScript program with Ajax (with HTML/CSS) for:**

**a. Use ajax() method (without Jquery) to add the text content from the text file by sending ajax request.**

**b. Use ajax() method (with Jquery) to add the text content from the text file by sending ajax request.**

**c. Illustrate the use of getJSON() method in jQuery**

**d. Illustrate the use of parseJSON() method to display JSON values.**

**Code:**

pgm10.html

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>AJAX Examples | vtucode</title>

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.0/jquery.min.js"></script>

<style>

body {

font-family: Arial, sans-serif;

margin: 0;

padding: 0;

background-color: #f4f4f9;

}

h1 {

text-align: center;

color: #333;

padding: 20px 0;

}

#content {

flex-direction: column;

display: flex;

max-width: 600px;

margin: 20px auto;

padding: 20px;

border: 1px solid #ddd;

border-radius: 8px;

background-color: #fff;

box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);

}

button {

display: inline-block;

padding: 10px 15px;

margin: 12px;

border: none;

border-radius: 5px;

background-color: #007bff;

color: #fff;

font-size: 16px;

cursor: pointer;

transition: box-shadow 0.3s;

}

button:hover {

box-shadow: 0 0 0 2px #fff, 0 0 0 4px #007bff;

}

button:focus {

box-shadow: 0 0 0 2px #fff, 0 0 0 4px #007bff;

}

#output {

display: none;

margin-top: 20px;

padding: 10px;

border-radius: 5px;

white-space: pre-wrap;

max-height: 300px;

overflow-y: auto;

}

#output.plain-ajax {

background-color: #f0f8ff;

border: 1px solid #b0c4de;

}

#output.jquery-ajax {

background-color: #f5fffa;

border: 1px solid #98fb98;

}

#output.jquery-json {

background-color: #fffaf0;

border: 1px solid #ffd700;

}

#output.parse-json {

background-color: #fff0f5;

border: 1px solid #ff69b4;

}

</style>

</head>

<body>

<h1>AJAX Examples</h1>

<div id="content">

<button id="plain-ajax-btn">Load Text (Plain AJAX)</button>

<button id="jquery-ajax-btn">Load Text (jQuery AJAX)</button>

<button id="jquery-json-btn">Load JSON (jQuery getJSON)</button>

<button id="parse-json-btn">Load and Parse JSON (jQuery get)</button>

<div id="output" aria-live="polite"></div>

</div>

<script>

function showOutput(className) {

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

output.className = className;

output.style.display = 'block';

}

function handleError(xhr) {

let errorMessage = 'Error loading file.';

if (xhr.status === 0) {

errorMessage = 'Network error. Please check your connection.';

} else if (xhr.status === 404) {

errorMessage = 'File not found.';

} else if (xhr.status === 500) {

errorMessage = 'Server error.';

}

return errorMessage;

}

function updateOutput(data, className) {

$('#output').text(data).removeClass().addClass(className).show();

}

document.getElementById('plain-ajax-btn').addEventListener('click', function () {

var xhr = new XMLHttpRequest();

xhr.open('GET', 'textfile.txt', true);

xhr.onload = function () {

if (xhr.status === 200) {

updateOutput(xhr.responseText, 'plain-ajax');

} else {

updateOutput(handleError(xhr), 'plain-ajax');

}

};

xhr.send();

});

$('#jquery-ajax-btn').on('click', function () {

$.ajax({

url: 'textfile.txt',

method: 'GET'

})

.done(function (data) {

updateOutput(data, 'jquery-ajax');

})

.fail(function (xhr) {

updateOutput(handleError(xhr), 'jquery-ajax');

});

});

$('#jquery-json-btn').on('click', function () {

$.getJSON('data.json')

.done(function (data) {

updateOutput(JSON.stringify(data, null, 2), 'jquery-json');

})

.fail(function (xhr) {

updateOutput(handleError(xhr), 'jquery-json');

});

});

$('#parse-json-btn').on('click', function () {

$.get('data.json')

.done(function (data) {

try {

let jsonData = typeof data === 'string' ? JSON.parse(data) : data;

updateOutput(JSON.stringify(jsonData, null, 2), 'parse-json');

} catch (e) {

updateOutput('Error parsing JSON: ' + e.message, 'parse-json');

}

})

.fail(function (xhr) {

updateOutput(handleError(xhr), 'parse-json');

});

});

</script>

</body>

</html>

textfile.txt

hi this is example text...

data.json

{

"name":"John Doe",

"age":30,

"city":"New York",

"skills":["JavaScript","React","Node.js"],

"address":{"street":"123 Elm Street","zipcode":"10001"},

"projects":[

{"name":"Website Redesign",

"year":2023,

"technologies":["HTML","CSS","JavaScript"]

},

{"name":"Mobile App",

"year":2024,

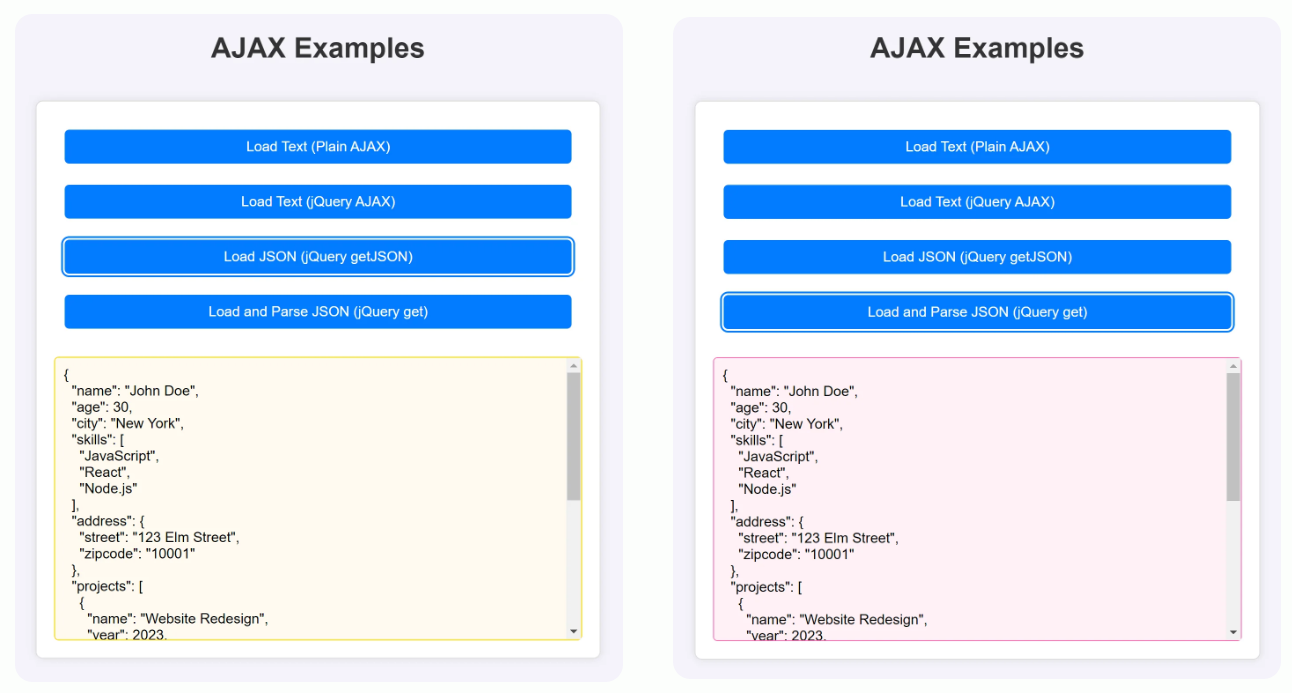
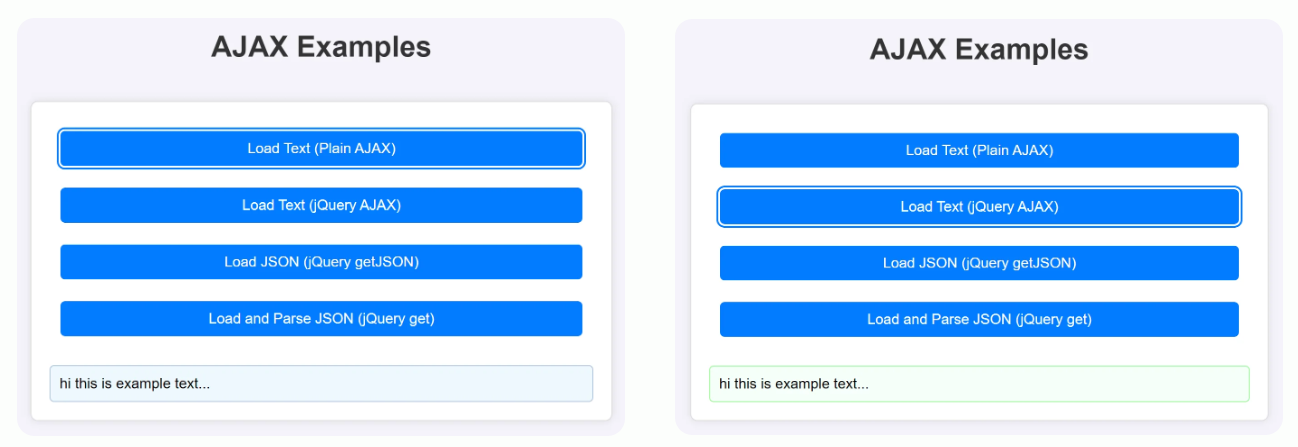
"technologies":["React Native","Expo"]

}

]

}

**Output:**

****