<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Document</title>

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

<style>

div{

color: bisque;

font-size: 16px;

background-color: black;

}

</style>

</head>

<body>

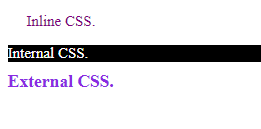
<p style="color: purple; margin-left: 20px;">Inline CSS.</p>

<div>Internal CSS.</div>

<h3>External CSS.</h3>

</body>

</html>



<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Document</title>

<script>

function hov() {

var e=document.getElementById('hover');

e.style.display='none';

}

function out() {

var e=document.getElementById('out');

e.style.display='none';

}

function focused() {

var e=document.getElementById('inp');

if(confirm('Got it?')) {

e.blur();

}

}

function myFunction() {

alert("You pressed a key inside the input field");

}

</script>

</head>

<body>

<div id="hover"

onmouseover="hov()"

style="background-color:burlywood;

height:200px;

width:200px;">

</div>

<div id="out" onmouseout="out()" style="background-color:green;height:200px;width:200px;">

</div>

<input onchange="alert(this.value)"

type="number"

style="margin-left: 45%;">

<img onload="alert('Image completely loaded')"

alt="Image Not Loaded"

src="./Skills\_Image.jpg" style="height: 220px; width: 400px;">

<p style="margin-left: 45%;">

Take the focus into the input box below:

</p>

<input id="inp"

onfocus="focused()""

style=" margin-left: 45%;">

<h2>The onkeypress Event</h2>

<input type="text" onkeypress="myFunction()">

</body>

</html>

<!DOCTYPE html>

<html>

<head>

<style>

table {

border-collapse: collapse;

width: 50%;

margin: auto;

margin-top: 20px;

}

th, td {

border: 1px solid black;

padding: 8px;

text-align: center;

}

th {

background-color: #f2f2f2;

}

</style>

</head>

<body>

<h2>Table with Row and Column Labels</h2>

<table id="dataTable">

<tr>

<th>Row</th>

<th>Column 1</th>

<th>Column 2</th>

<th>Column 3</th>

</tr>

</table>

<script>

// Access the table element by its id

const table = document.getElementById("dataTable");

// Generating rows and cells

const numRows = 50;

const numCols = 3;

for (let row = 1; row <= numRows; row++) {

const newRow = table.insertRow();

// Insert the "Row" cell

const rowCell = newRow.insertCell(0);

rowCell.textContent = row;

for (let col = 1; col <= numCols; col++) {

const newCell = newRow.insertCell(col);

newCell.textContent = `Row ${row}, Column ${col}`;

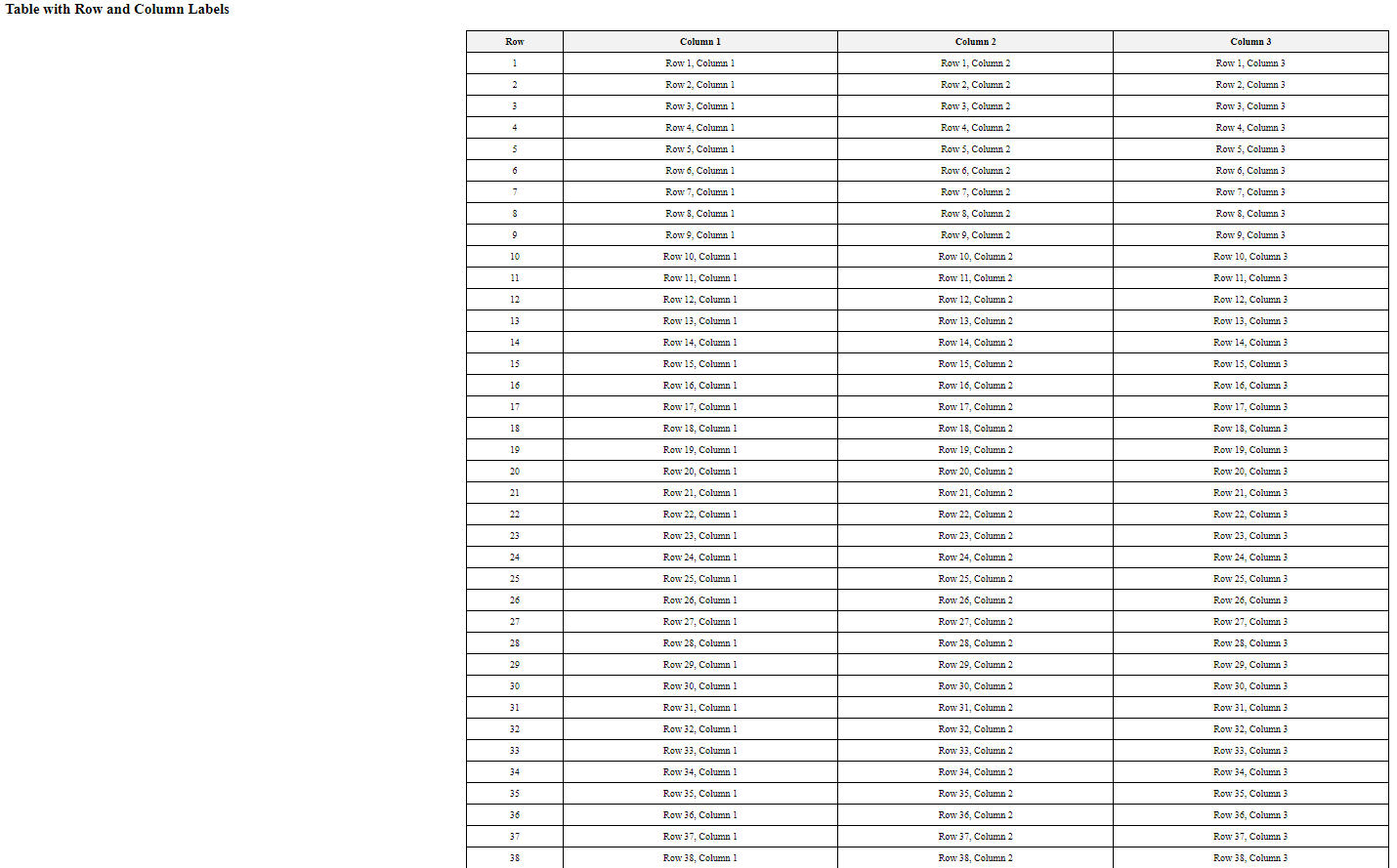
}

}

</script>

</body>

</html>



<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Table Generator</title>

<style>

table {

border-collapse: collapse;

width: 60%;

height: 100px;

}

th, td {

border: 1px solid black;

padding: 8px;

text-align: center;

}

th {

background-color: #f2f2f2;

}

</style>

</head>

<body>

<input type="text" id="inputValue" placeholder="Enter a value">

<button onclick="pushValue()">Push</button>

<button onclick="popValue()">Pop</button>

<button onclick="unshiftValue()">Unshift</button>

<button onclick="shiftValue()">Shift</button>

<button onclick="sortArray()">Sort</button>

<button onclick="mergeArray()">Merge</button>

<button onclick="reverseArray()">Reverse</button>

<br><br>

<table id="myTable">

<tr>

<th>Index</th>

<th>Value</th>

</tr>

</table>

<script>

var myArray = [];

function updateTable() {

var table = document.getElementById("myTable");

table.innerHTML = ""; // Clear table contents

// Add table headers

var headerRow = table.insertRow(0);

var indexHeader = document.createElement("th");

indexHeader.textContent = "Index";

var valueHeader = document.createElement("th");

valueHeader.textContent = "Value";

headerRow.appendChild(indexHeader);

headerRow.appendChild(valueHeader);

// Add table rows

for (var i = 0; i < myArray.length; i++) {

var row = table.insertRow(i + 1);

var indexCell = row.insertCell(0);

indexCell.textContent = i;

var valueCell = row.insertCell(1);

valueCell.textContent = myArray[i];

}

}

function pushValue() {

var inputValue = document.getElementById("inputValue").value;

myArray.push(inputValue);

updateTable();

}

function popValue() {

myArray.pop();

updateTable();

}

function unshiftValue() {

var inputValue = document.getElementById("inputValue").value;

myArray.unshift(inputValue);

updateTable();

}

function shiftValue() {

myArray.shift();

updateTable();

}

function sortArray() {

myArray.sort();

updateTable();

}

function mergeArray() {

var inputValue = document.getElementById("inputValue").value;

var mergeArray = inputValue.split(",").map(item => item.trim());

myArray = myArray.concat(mergeArray);

updateTable();

}

function reverseArray() {

myArray.reverse();

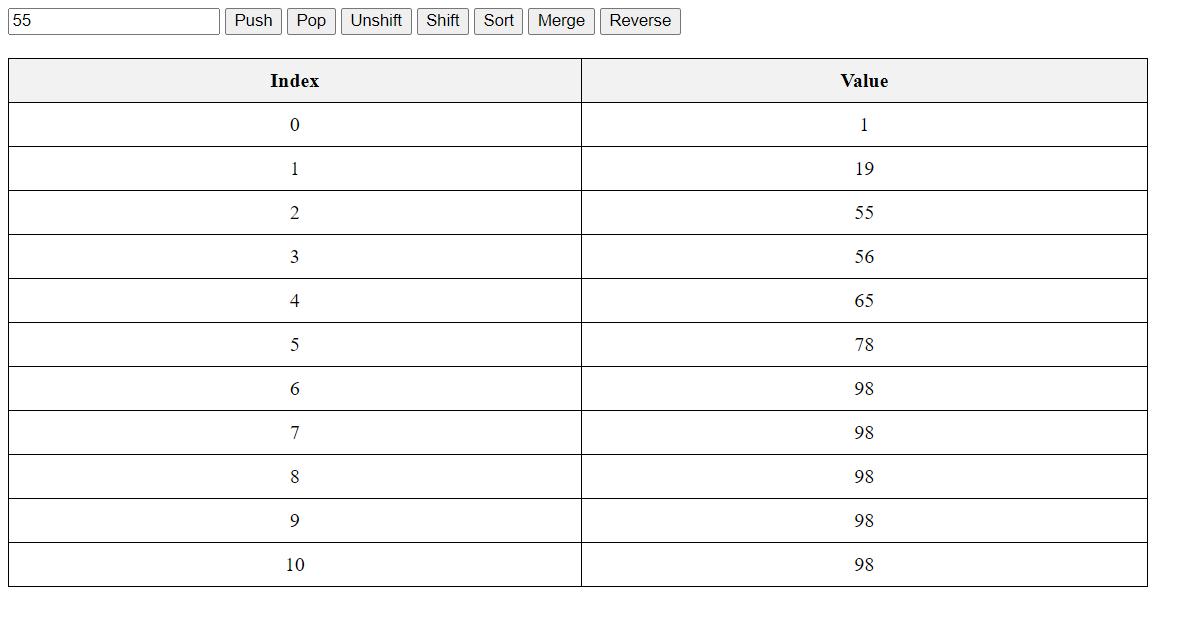
updateTable();

}

</script>

</body>

</html>



-------------------------------------------------------------------------------------------------------------------------------------

<!DOCTYPE html>

<html>

<head>

<style>

body {

transition: background-color 0.5s ease;

}

select {

margin-top: 20px;

}

</style>

</head>

<body>

<h2>Select Background Color</h2>

<select id="colorSelect">

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

<option value="red">Red</option>

<option value="blue">Blue</option>

<option value="green">Green</option>

<option value="yellow">Yellow</option>

</select>

<script>

const colorSelect = document.getElementById("colorSelect");

// Array of color options and their corresponding values

const colorOptions = [

{ label: "Red", value: "red" },

{ label: "Blue", value: "blue" },

{ label: "Green", value: "green" },

{ label: "Yellow", value: "yellow" }

];

// Add options to the select element

colorOptions.forEach(option => {

const optionElement = document.createElement("option");

optionElement.value = option.value;

optionElement.textContent = option.label;

colorSelect.appendChild(optionElement);

});

// Event listener for color selection

colorSelect.addEventListener("change", function() {

const selectedColor = colorSelect.value;

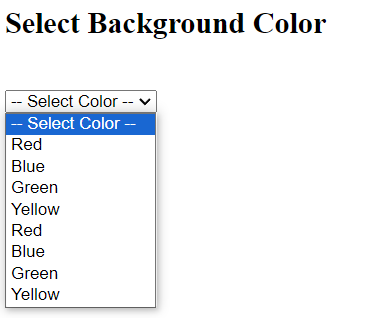
document.body.style.backgroundColor = selectedColor;

});

</script>

</body>

</html>



--------------------------------------------------------------------------------------------------------------------------------

<!DOCTYPE html>

<html>

<head>

<style>

body {

transition: background-color 0.5s ease;

}

</style>

</head>

<body>

<h2>Changing Background Color</h2>

<script>

// Array of colors to choose from

const colors = ["red", "blue", "green", "yellow", "purple", "orange"];

function changeBackgroundColor() {

const randomIndex = Math.floor(Math.random() \* colors.length);

const randomColor = colors[randomIndex];

document.body.style.backgroundColor = randomColor;

}

// Change background color every 5 seconds (5000 milliseconds)

setInterval(changeBackgroundColor, 500);

</script>

</body>

</html>



<!DOCTYPE html>

<html>

<head>

<title>Font Size Animation</title>

</head>

<body>

<div id="textElement">SILVER OAK UNIVERSITY</div>

<script>

// Get the element where you want to display the text

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

// Set initial font size and color

let fontSize = 10; // Starting font size

textElement.style.fontSize = `${fontSize}px`;

textElement.style.color = 'blue';

// Function to increase font size at an interval

const increaseFontSize = () => {

if (fontSize <= 50) {

textElement.style.fontSize = `${fontSize}px`;

fontSize++;

} else {

clearInterval(intervalId);

}

};

// Start the interval

const intervalId = setInterval(increaseFontSize, 1000);

</script>

</body>

</html>



--------------------------------------------------------------------------------------------------------------------------------------

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Document</title>

<script>

function Alert() {

alert("An Online Computer Science"

+ " Portal ");

}

function Prompt() {

let x = prompt("Enter your mail here : ");

document.write("Your ID : " + x);

}

function Confirm() {

var x;

if (confirm("Press a button!") == true) {

x = "OK pressed!";

} else {

x = "Cancel!";

}

document.getElementById("confirm").innerHTML = x;

}

</script>

</head>

<body>

<h3>Alert Box</h3>

<button onclick="Alert()">

Click here for alert box

</button>

<h3>Prompt Box</h3>

<input type="button" onclick="Prompt();"

value="Click here for Prompt box"/>

<h3>Confirm Box</h3>

<button onclick="Confirm()">

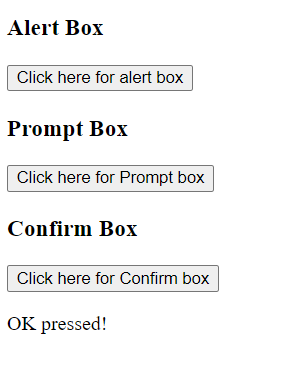
Click here for Confirm box

</button>

<p id="confirm"></p>

</body>

</html>



<html lang="en">

<head>

<meta charset="UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

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

<title>Form Validation</title>

<link rel="preconnect" href="https://fonts.googleapis.com">

<link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>

<link href="https://fonts.googleapis.com/css2?family=Poppins:wght@400;700&display=swap" rel="stylesheet">

<link rel="stylesheet" href="./style.css">

<script defer src="./index.js"></script>

<style>

body {

background: linear-gradient(to right, #0f2027, #203a43, #2c5364);

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

}

#form {

width: 300px;

margin: 20vh auto 0 auto;

padding: 20px;

background-color: whitesmoke;

border-radius: 4px;

font-size: 12px;

}

#form h1 {

color: #0f2027;

text-align: center;

}

#form button {

padding: 10px;

margin-top: 10px;

width: 100%;

color: white;

background-color: rgb(41, 57, 194);

border: none;

border-radius: 4px;

}

.input-control {

display: flex;

flex-direction: column;

}

.input-control input {

border: 2px solid #f0f0f0;

border-radius: 4px;

display: block;

font-size: 12px;

padding: 10px;

width: 100%;

}

.input-control input:focus {

outline: 0;

}

.input-control.success input {

border-color: #09c372;

}

.input-control.error input {

border-color: #ff3860;

}

.input-control .error {

color: #ff3860;

font-size: 9px;

height: 13px;

}

</style>

</head>

<body>

<div class="container">

<form id="form" action="/">

<h1>Registration</h1>

<div class="input-control">

<label for="username">Username</label>

<input id="username" name="username" type="text">

<div class="error"></div>

</div>

<div class="input-control">

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

<input id="email" name="email" type="text">

<div class="error"></div>

</div>

<div class="input-control">

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

<input id="password"name="password" type="password">

<div class="error"></div>

</div>

<div class="input-control">

<label for="password2">Password again</label>

<input id="password2"name="password2" type="password">

<div class="error"></div>

</div>

<button type="submit">Sign Up</button>

</form>

</div>

<script>

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

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

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

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

const password2 = document.getElementById('password2');

form.addEventListener('submit', e => {

e.preventDefault();

validateInputs();

});

const setError = (element, message) => {

const inputControl = element.parentElement;

const errorDisplay = inputControl.querySelector('.error');

errorDisplay.innerText = message;

inputControl.classList.add('error');

inputControl.classList.remove('success')

}

const setSuccess = element => {

const inputControl = element.parentElement;

const errorDisplay = inputControl.querySelector('.error');

errorDisplay.innerText = '';

inputControl.classList.add('success');

inputControl.classList.remove('error');

};

const isValidEmail = email => {

const re = /^(([^<>()[\]\\.,;:\s@"]+(\.[^<>()[\]\\.,;:\s@"]+)\*)|(".+"))@((\[[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}\])|(([a-zA-Z\-0-9]+\.)+[a-zA-Z]{2,}))$/;

return re.test(String(email).toLowerCase());

}

const validateInputs = () => {

const usernameValue = username.value.trim();

const emailValue = email.value.trim();

const passwordValue = password.value.trim();

const password2Value = password2.value.trim();

if(usernameValue === '') {

setError(username, 'Username is required');

} else {

setSuccess(username);

}

if(emailValue === '') {

setError(email, 'Email is required');

} else if (!isValidEmail(emailValue)) {

setError(email, 'Provide a valid email address');

} else {

setSuccess(email);

}

if(passwordValue === '') {

setError(password, 'Password is required');

} else if (passwordValue.length < 8 ) {

setError(password, 'Password must be at least 8 character.')

} else {

setSuccess(password);

}

if(password2Value === '') {

setError(password2, 'Please confirm your password');

} else if (password2Value !== passwordValue) {

setError(password2, "Passwords doesn't match");

} else {

setSuccess(password2);

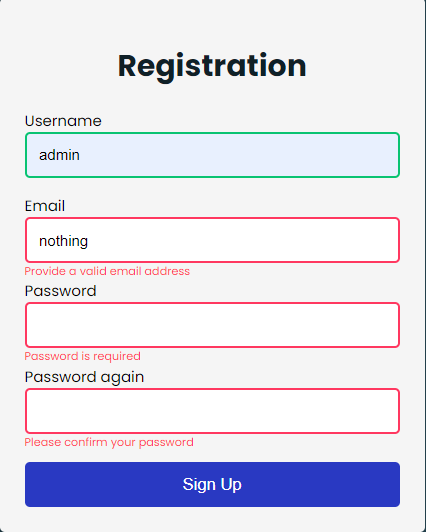
}

};

</script>

</body>

</html>



<!DOCTYPE html>

<html>

<head>

<title>

Check a number is Prime or

not using JavaScript

</title>

<script type="text/javascript">

// Function to check prime number

function p() {

var n, i, flag = true;

// Getting the value form text

// field using DOM

n = document.myform.n.value;

n = parseInt(n)

for(i = 2; i <= n - 1; i++)

if (n % i == 0) {

flag = false;

break;

}

// Check and display alert message

if (flag == true)

alert(n + " is prime");

else

alert(n + " is not prime");

}

</script>

</head>

<body>

<center>

<h4>check number is prime or not</h4>

<hr color="Green">

<form name="myform">

Enter the number:

<input type="text" name=n value="">

<br><br>

<input type="button" value="Check" onClick="p()">

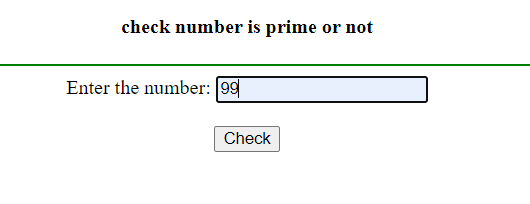
<br>

</form>

</center>

</body>

</html>



<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Fibonacci Numbers</title>

<script>

function generateFibonacci(n) {

var fibonacciNumbers = [0, 1];

for (var i = 2; i < n; i++) {

fibonacciNumbers.push(fibonacciNumbers[i - 1] + fibonacciNumbers[i - 2]);

}

return fibonacciNumbers;

}

function displayFibonacci() {

var n = parseInt(document.getElementById("n").value);

var fibonacciList = document.getElementById("fibonacciList");

while (fibonacciList.firstChild) {

fibonacciList.removeChild(fibonacciList.firstChild);

}

var fibonacciNumbers = generateFibonacci(n);

for (var i = 0; i < n; i++) {

var listItem = document.createElement("li");

listItem.textContent = fibonacciNumbers[i];

fibonacciList.appendChild(listItem);

}

}

</script>

</head>

<body>

<h1>Fibonacci Number Generator</h1>

<label for="n">Enter the value of 'n': </label>

<input type="number" id="n" min="1">

<button onclick="displayFibonacci()">Generate Fibonacci</button>

<ul id="fibonacciList"></ul>

</body>

</html>

