

LATIHAN PRAKTIKUM PENGEMBANGAN APLIKASI WEB

BAB : JavaScript (1)
NAMA : DANI ADRIAN
NIM : 225150201111009
ASISTEN : - IBAR HUTTAQI SULTHON
 - MUHAMMAD AMMAR
TANGGAL PRAKTIKUM : 23/05/2024

1. Fundamental JavaScript

LATIHAN 1

A. Soal

- a. Jawablah beberapa pertanyaan berikut!
 - a) Jelaskan sejarah singkat JS, dan siapakah yang menggagasnya pada tahun berapa?
 - b) Kenapa JavaScript (JS) disebut sebagai High Level Programming Language?
 - c) Kenapa JavaScript (JS) disebut sebagai interpreter?
 - d) Apakah sama JS dengan Java Programming Language, jelaskan perbedaannya?
 - e) Jelaskan perbedaan JS Lib. Vs JS Framework dan sebutkan contoh masing-masing!
 - f) Apakah JS mendukung terminology OOP?
 - g) Sejak kapan JS support pada server side, siapakah penggagasnya?

B. Screenshoot

C. Syntax

1	font courier new 11
---	---------------------

D. Penjelasan

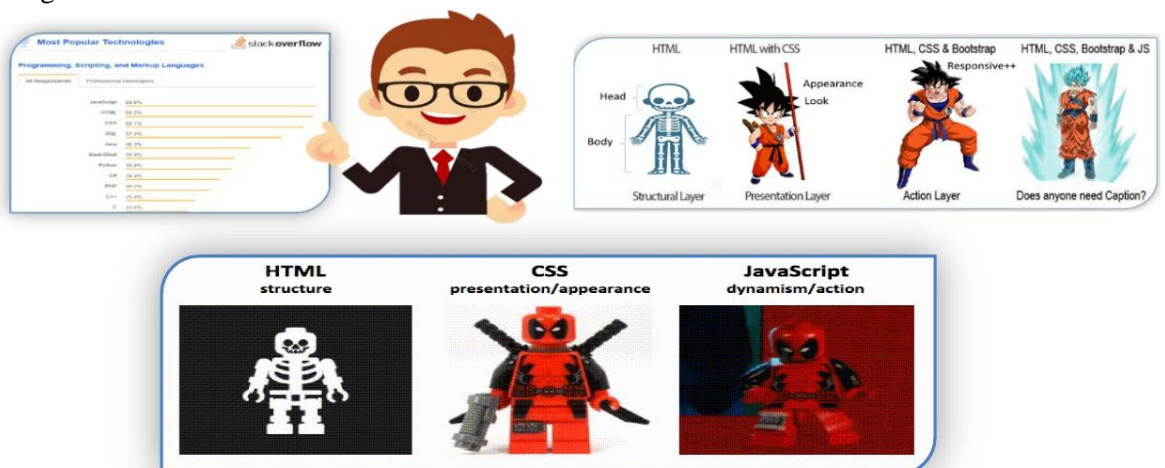
- a) JavaScript ditemukan oleh Brendan Eich pada tahun 1995 dan dikembangkan awalnya untuk Netscape 2. Pada tahun 1997, JavaScript menjadi standar ECMA-262. Setelah Netscape menyerahkan JavaScript ke ECMA, Mozilla, perusahaan di belakang browser Firefox, terus mengembangkan bahasa tersebut.
- b) JavaScript dianggap sebagai high-level language karena diabstraksi dari operasi-operasi tingkat rendah dari komputer tempat ia dijalankan. Mengonversi tugas-tugas pemrograman tingkat tinggi menjadi eksekusi tingkat rendah membutuhkan waktu. Javascript dikategorikan sebagai high-level language karena mereka menawarkan tingkat abstraksi yang membuat mereka lebih mirip satu sama lain daripada dengan bahasa mesin.

- c) Karena JavaScript tidak memerlukan compiler untuk menjalankannya.
- d) Perbedaanya diantaranya adalah :
- JavaScript dikembangkan oleh Netscape, Inc. Sedangkan, Java dikembangkan oleh Java.
 - JavaScript tidak bisa membuat aplikasi sendiri dan memberikan interaksi ke halaman web yang tidak dapat dijangkau oleh HTML. Berbeda dengan JavaScript, Java bisa membuat aplikasi berdiri sendiri.
 - JavaScript termasuk OOP Programming Language dan Java termasuk OOP Scripting Language.
 - JavaScript hanya bisa dijalankan di browser. Sedangkan, Java dapat membuat aplikasi yang berjalan di virtual machine atau browser.
 - Java perlu dikompilasi. Sedangkan, JavaScript semuanya ada dalam teks.
- e) JS Lib adalah library pada JavaScript. Contohnya adalah React.js dan jQuery, yang membuat dirinya berbeda dari JS Framework adalah berfokus pada penggunaan ulang kode (reuse), tools yang special tidak semuanya ada, dan menjadi bagian dari basis kode yang sudah ada, JS Framework adalah framework pada JavaScript. Contohnya adalah Vue, Laravel, dan Vite, yang membuat dirinya berbeda dari JS Lib adalah berfokus pada refactoring, dapat mengabstraksikan pola arsitektur, perilaku, dan logika, serta terdapat kebutuhan tools.
- f) Ya, JS mendukung terminologi OOP seperti prototype, object, dan encapsulation.
- g) Server side didukung pada JS, semenjak kehadiran Ryan Dahl pada tahun 2009.

LATIHAN 2

A. Soal

Dari gambar berikut :



Jelaskan keunggulan JS, jika dibandingkan dengan Programming lainnya, dan keunggulan dari Aplikasi Web atau situs Web jika menggunakan JS, bukan hanya HTML dan CSS!

B. Screenshoot

C. Syntax

1	
---	--

D. Penjelasan

Keunggulan JS antara lain adalah fleksibilitasnya yang tinggi. Sedangkan, untuk keunggulan pada web adalah memungkinkan halaman web lebih interaktif dan juga mudah untuk dipelajari dan multiplatform.

2. Konsep dan Implementasi JS Library & Framework

LATIHAN 1

A. Soal

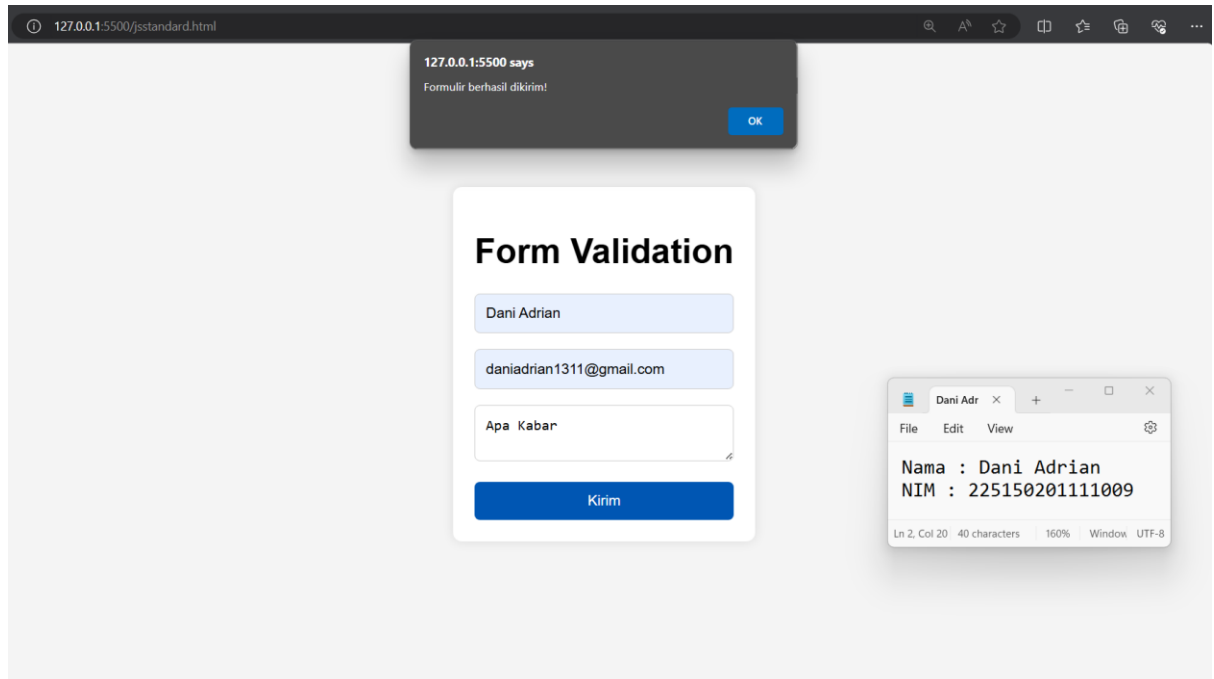
Jelaskan Javascript (JS) dari beberapa istilah berikut jika dihubungkan untuk pengembangan Web App atau Hybrid Web /Mobile App dan khusus untuk 3point awal, berikan contoh case sederhana beserta implementasinya:

- a) JS standard
- b) CDN TensorFlow.js
- c) jQuery
- d) JS Framework yang support pada client side dan server side (Node.js) vs Native Script vs lainnya

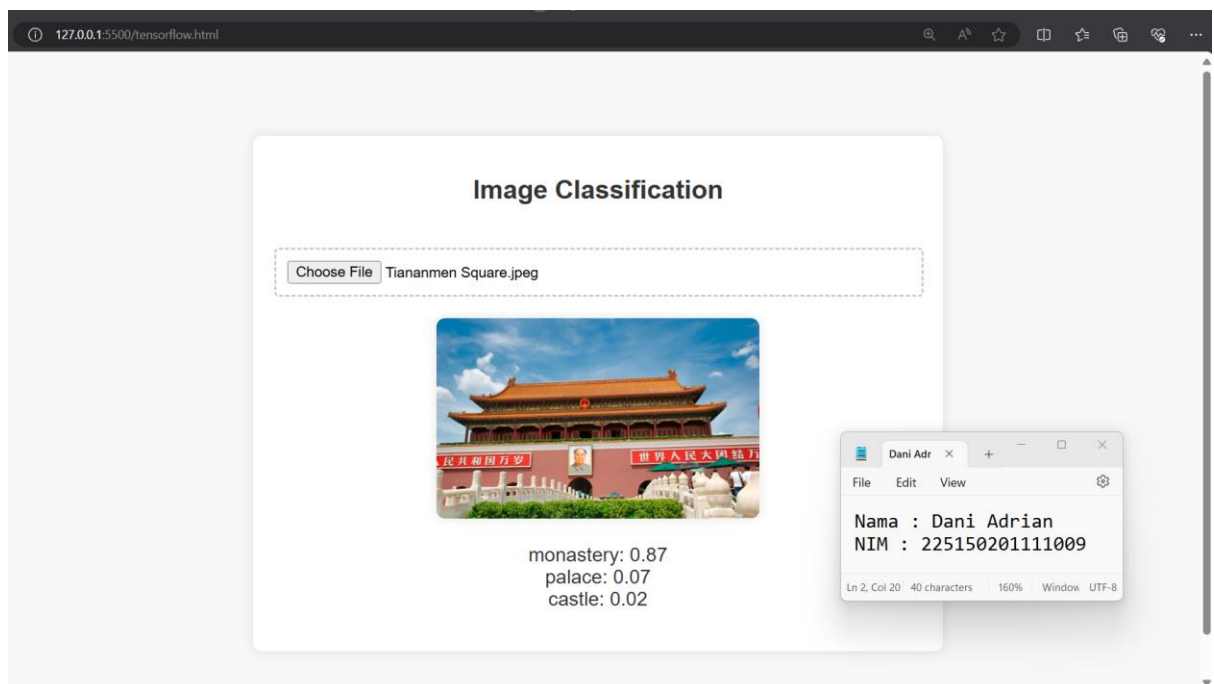


B. Screenshoot

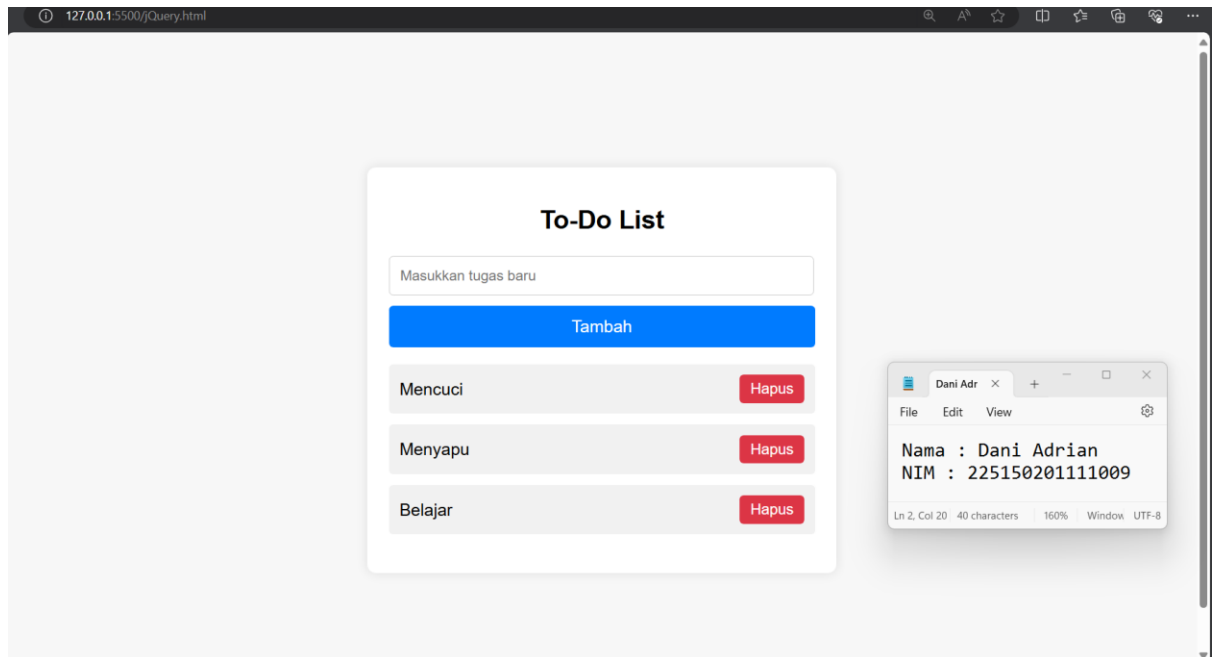
JS standard



CDN TensorFlow.js



jQuery



C. Syntax

jStandard.html	
1.	<!DOCTYPE html>
2.	<html lang="en">
3.	
4.	<head>
5.	<meta charset="UTF-8">
6.	<meta name="viewport" content="width=device-
7.	width, initial-scale=1.0">
8.	<title>Form Validation</title>
9.	<style>
10.	body {
11.	font-family: Arial, sans-serif;
12.	background-color: #f4f4f4;
13.	margin: 0;
14.	padding: 0;
15.	display: flex;
16.	justify-content: center;
17.	align-items: center;
18.	height: 100vh;
19.	}
20.	.container {
21.	max-width: 400px;
22.	padding: 20px;
23.	background-color: #fff;
24.	border-radius: 8px;
25.	box-shadow: 0 0 10px rgba(0, 0, 0,
26.	0.1);
27.	}

28.	h1 {
29.	text-align: center;
30.	margin-bottom: 20px;
31.	}
32.	
33.	input[type="text"],
34.	input[type="email"],
35.	textarea {
36.	width: 100%;
37.	padding: 10px;
38.	margin-bottom: 15px;
39.	border: 1px solid #ccc;
40.	border-radius: 5px;
41.	box-sizing: border-box;
42.	}
43.	
44.	textarea {
45.	resize: vertical;
46.	}
47.	
48.	button {
49.	width: 100%;
50.	padding: 10px;
51.	background-color: #007bff;
52.	color: #fff;
53.	border: none;
54.	border-radius: 5px;
55.	cursor: pointer;
56.	}
57.	
58.	button:hover {
59.	background-color: #0056b3;
60.	}
61.	
62.	.error-message {
63.	color: #dc3545;
64.	font-size: 14px;
65.	margin-top: -10px;
66.	margin-bottom: 10px;
67.	}
68.	</style>
69.	</head>
70.	
71.	<body>
72.	<div class="container">
73.	<h1>Form Validation</h1>
74.	<form id="myForm">
75.	<input type="text" id="name"
76.	placeholder="Nama">
77.	<div class="error-message"
78.	id="nameError"></div>
	<input type="email" id="email"
	placeholder="Email">
	<div class="error-message"
	id="emailError"></div>

79.	<textarea id="message"
80.	placeholder="Pesan"></textarea>
81.	<div class="error-message"
82.	id="messageError"></div>
83.	<button type="button"
84.	onclick="validateForm()">Kirim</button>
85.	</form>
86.	</div>
87.	<script>
88.	function validateForm() {
89.	var name =
90.	document.getElementById("name").value;
91.	var email =
92.	document.getElementById("email").value;
93.	var message =
94.	document.getElementById("message").value;
95.	var nameError =
96.	document.getElementById("nameError");
97.	var emailError =
98.	document.getElementById("emailError");
99.	var messageError =
100.	document.getElementById("messageError");
101.	var isValid = true;
102.	
103.	// Validasi nama
104.	if (name === "") {
105.	nameError.innerHTML = "Nama harus
106.	diisi";
107.	isValid = false;
108.	} else {
109.	nameError.innerHTML = "";
110.	}
111.	
112.	// Validasi email
113.	if (email === "") {
114.	emailError.innerHTML = "Email harus
115.	diisi";
116.	isValid = false;
117.	} else {
118.	emailError.innerHTML = "";
119.	}
120.	
121.	// Validasi pesan
	if (message === "") {
	messageError.innerHTML = "Pesan
	harus diisi";
	isValid = false;
	} else {
	messageError.innerHTML = "";
	}
	// Jika semua valid, kirim formulir
	if (isValid) {
	alert("Formulir berhasil
	dikirim!");

122.	// Proses pengiriman formulir
123.	}
124.	}
125.	</script>
126.	</body>
127.	
128.	</html>

tensorflow.html	
1.	<!DOCTYPE html>
2.	<html lang="en">
3.	
4.	<head>
5.	<meta charset="UTF-8">
6.	<meta name="viewport" content="width=device-width, initial-scale=1.0">
7.	<title>Image Classification</title>
8.	<script src="https://cdn.jsdelivr.net/npm/@tensorflow/tfjs" ></script>
9.	<script src="https://cdn.jsdelivr.net/npm/@tensorflow-models/mobilenet"></script>
10.	<style>
11.	body {
12.	font-family: Arial, sans-serif;
13.	background-color: #f7f7f7;
14.	margin: 0;
15.	padding: 20px;
16.	display: flex;
17.	justify-content: center;
18.	align-items: center;
19.	height: 100vh;
20.	}
21.	
22.	.container {
23.	background: #fff;
24.	padding: 20px;
25.	border-radius: 8px;
26.	box-shadow: 0 0 10px rgba(0, 0, 0,
27.	0.1);
28.	width: 100%;
29.	max-width: 600px;
30.	text-align: center;
31.	}
32.	h1 {
33.	font-size: 24px;
34.	margin-bottom: 20px;
35.	color: #333;
36.	}
37.	
38.	input[type="file"] {
39.	padding: 10px;

40.	margin-top: 20px;
41.	border: 2px dashed #ccc;
42.	border-radius: 4px;
43.	width: calc(100% - 22px);
44.	cursor: pointer;
45.	transition: border-color 0.3s;
46.	}
47.	
48.	input[type="file"]:hover {
49.	border-color: #007bff;
50.	}
51.	
52.	#uploadedImage {
53.	margin-top: 20px;
54.	max-width: 100%;
55.	height: auto;
56.	border-radius: 8px;
57.	box-shadow: 0 0 10px rgba(0, 0, 0,
	0.1);
58.	}
59.	
60.	#result {
61.	margin-top: 20px;
62.	font-size: 18px;
63.	color: #333;
64.	}
65.	</style>
66.	</head>
67.	
68.	<body>
69.	<div class="container">
70.	<h1>Image Classification</h1>
71.	<input type="file" accept="image/*"
	id="imageUpload">
72.	
73.	<p id="result"></p>
74.	</div>
75.	
76.	<script>
77.	const imageUpload =
	document.getElementById('imageUpload');
78.	const uploadedImage =
	document.getElementById('uploadedImage');
79.	const result =
	document.getElementById('result');
80.	
81.	imageUpload.addEventListener('change',
	async (event) => {
82.	const file = event.target.files[0];
83.	const reader = new FileReader();
84.	
85.	reader.onload = function (event) {
86.	uploadedImage.src =
	event.target.result;
87.	}

88.	reader.readAsDataURL(file);
89.	
90.	reader.onloadend = async () => {
91.	const model = await
	mobilenet.load();
92.	const predictions = await
	model.classify(uploadedImage);
93.	result.innerText =
	predictions.map(p => `\${p.className}:`
	`\${p.probability.toFixed(2)}').join('\n');
94.	}
95.	});
96.	</script>
97.	</body>
98.	
99.	</html>

jQuery.html

1.	<!DOCTYPE html>
2.	<html lang="en">
3.	
4.	<head>
5.	<meta charset="UTF-8">
6.	<meta name="viewport" content="width=device-
	width, initial-scale=1.0">
7.	<title>To-Do List dengan jQuery</title>
8.	<script src="https://code.jquery.com/jquery-
	3.6.0.min.js"></script>
9.	<style>
10.	body {
11.	font-family: Arial, sans-serif;
12.	background-color: #f7f7f7;
13.	margin: 0;
14.	padding: 20px;
15.	display: flex;
16.	justify-content: center;
17.	align-items: center;
18.	height: 100vh;
19.	}
20.	
21.	.container {
22.	background: #fff;
23.	padding: 20px;
24.	border-radius: 8px;
25.	box-shadow: 0 0 10px rgba(0, 0, 0,
	0.1);
26.	width: 100%;
27.	max-width: 400px;
28.	}
29.	
30.	h1 {
31.	font-size: 24px;
32.	margin-bottom: 20px;
33.	text-align: center;

34.	}
35.	
36.	input[type="text"] {
37.	width: calc(100% - 22px);
38.	padding: 10px;
39.	margin-bottom: 10px;
40.	border: 1px solid #ccc;
41.	border-radius: 4px;
42.	}
43.	
44.	button {
45.	width: 100%;
46.	padding: 10px;
47.	border: none;
48.	background: #007bff;
49.	color: white;
50.	font-size: 16px;
51.	border-radius: 4px;
52.	cursor: pointer;
53.	transition: background 0.3s;
54.	}
55.	
56.	button:hover {
57.	background: #0056b3;
58.	}
59.	
60.	ul {
61.	list-style-type: none;
62.	padding: 0;
63.	}
64.	
65.	li {
66.	display: flex;
67.	justify-content: space-between;
68.	align-items: center;
69.	background: #f1f1f1;
70.	padding: 10px;
71.	margin-bottom: 10px;
72.	border-radius: 4px;
73.	}
74.	
75.	.removeTodo {
76.	background: #dc3545;
77.	color: white;
78.	border: none;
79.	padding: 5px 10px;
80.	font-size: 14px;
81.	border-radius: 4px;
82.	cursor: pointer;
83.	transition: background 0.3s;
84.	width: auto;
85.	min-width: 60px;
86.	/* Tambahkan ini */
87.	}
88.	

89.	.removeTodo: hover {
90.	background: #c82333;
91.	}
92.	</style>
93.	</head>
94.	
95.	<body>
96.	<div class="container">
97.	<h1>To-Do List</h1>
98.	<input type="text" id="todoInput"
	placeholder="Masukkan tugas baru">
99.	<button id="addTodo">Tambah</button>
100.	<ul id="todoList">
101.	</div>
102.	
103.	<script>
104.	\$(document).ready(function () {
105.	\$('#addTodo').click(function () {
106.	const todoText =
	\$('#todoInput').val();
107.	if (todoText) {
108.	\$('#todoList').append('' +
	todoText + ' <button
	class="removeTodo">Hapus</button>');
109.	\$('#todoInput').val('');
110.	}
111.	});
112.	
113.	\$(document).on('click', '.removeTodo',
	function () {
114.	\$(this).parent().remove();
115.	});
116.	});
117.	</script>
118.	</body>
119.	
120.	</html>

D. Penjelasan

a) JS standard

JavaScript Standard (JS Standard) mengacu pada praktik dan pedoman pengkodean yang disarankan untuk mengembangkan aplikasi JavaScript. Hal ini mencakup aturan sintaksis, penanganan kesalahan, penggunaan variabel, gaya penulisan kode, dan sebagainya. JS Standard bertujuan meningkatkan kualitas, keberlanjutan, dan kemudahan pemeliharaan kode JavaScript. Penggunaan JS Standard membantu dalam membuat kode yang lebih konsisten, mudah dibaca, dan dipahami oleh pengembang lain.

Kasus: Validasi Formulir

Implementasi: Ketika pengguna mengisi formulir dengan nama, email, dan pesan, JavaScript akan memvalidasi apakah semua bidang telah diisi sebelum formulir dapat dikirim.

Meskipun kita menggunakan sedikit CSS untuk styling, itu tidak mempengaruhi karakteristik dasar dari JavaScript yang digunakan dalam validasi formulir. Oleh karena itu, implementasi ini termasuk dalam kategori JavaScript standar karena tidak mengandalkan pustaka atau framework eksternal.

b) CDN TensorFlow.js

TensorFlow.js adalah library JavaScript open-source yang digunakan untuk pengembangan machine learning dan deep learning di browser atau di sisi klien. TensorFlow.js memungkinkan pengembang untuk membangun model machine learning dan menjalankannya di browser dengan menggunakan JavaScript tanpa memerlukan backend server yang terpisah. Penggunaan Content Delivery Network (CDN) dalam konteks TensorFlow.js merujuk pada penyediaan dan pengiriman library TensorFlow.js melalui jaringan server global yang tersebar secara geografis untuk memastikan pengiriman yang cepat dan andal.

c) jQuery

jQuery adalah library JavaScript yang populer dan kuat yang menyederhanakan interaksi dengan DOM, penanganan event, animasi, permintaan AJAX, dan banyak lagi. Menyediakan antarmuka yang lebih mudah digunakan dan konsisten untuk memanipulasi elemen HTML dan berinteraksi dengan server.

Contoh Implementasi:

Kita membuat halaman web sederhana di mana pengguna bisa menambahkan item ke dalam daftar tugas (to-do list) menggunakan jQuery

d) JS Framework yang support pada client side dan server side (Node.js) vs Native Script vs lainnya

- JSFramework
 - Node.js: Node.js adalah runtime environment JavaScript yang memungkinkan pengembangan sisi server. Dengan Node.js, pengembang dapat menulis server-side logic menggunakan JavaScript. Ekosistemnya yang luas, termasuk npm (Node Package Manager), membuatnya sangat populer untuk pengembangan server-side.
 - React Native: Digunakan untuk membangun aplikasi mobile yang native menggunakan JavaScript dan React. React Native memungkinkan pengembang untuk menggunakan kode yang sama di berbagai platform (iOS dan Android), yang mempercepat pengembangan dan mengurangi biaya.
 - NativeScript: Framework untuk membangun aplikasi mobile menggunakan JavaScript, TypeScript, atau Angular. NativeScript memungkinkan akses langsung ke API native dari iOS dan Android, yang memberikan performa mendekati aplikasi native.

- Keunggulan
 - Node.js: Memungkinkan pengembangan full-stack JavaScript, efisien untuk aplikasi real-time, dan memiliki model I/O non-blocking yang sangat cocok untuk aplikasi dengan banyak koneksi simultan.
 - React Native: Menyediakan pengalaman pengguna yang lebih baik dengan performa yang hampir setara dengan aplikasi native, sambil tetap memungkinkan penggunaan kembali sebagian besar kode untuk berbagai platform.
 - NativeScript: Memberikan akses penuh ke API native, memungkinkan pengembangan aplikasi mobile dengan performa tinggi dan kemampuan untuk menggunakan kembali kode JavaScript/TypeScript dari aplikasi web.

LATIHAN 2

A. Soal

Berdasarkan contoh koding JS standard dengan atau tanpa CDN, misal dgn CDN TensorFlow.js, buatlah koding untuk melakukan komputasi berikut:

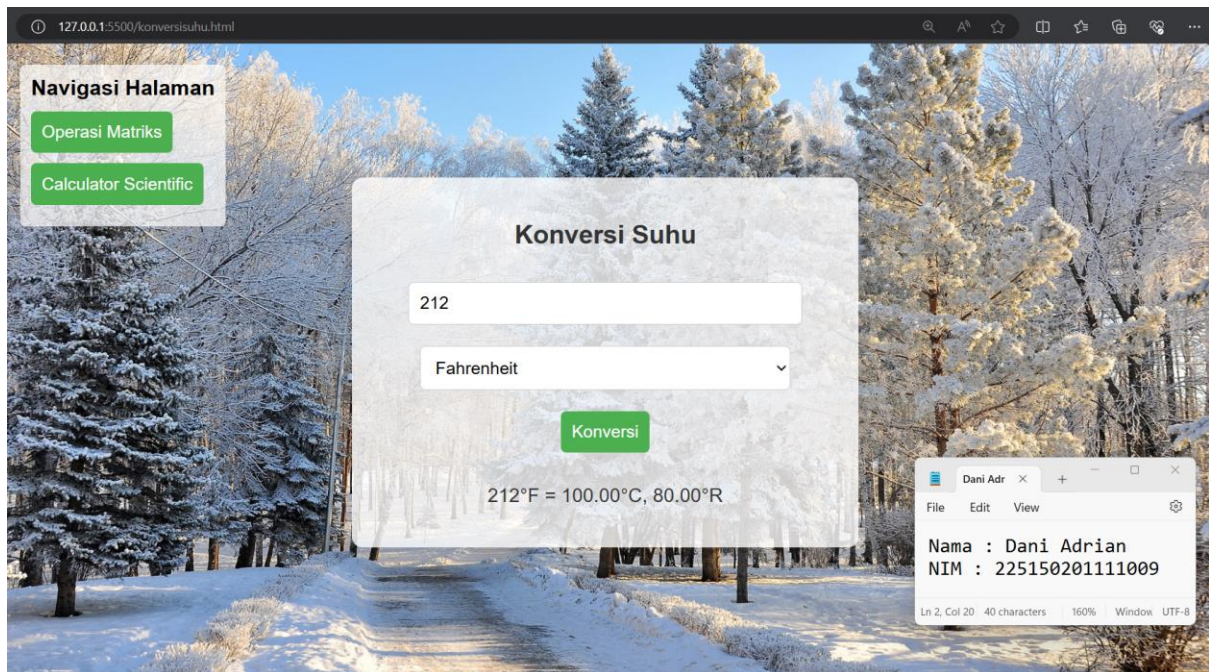
- Konversi antar satuan suhu dari Celcius (C), Fahrenheit (F) dan Reamur (R)
- Operasi Matrik Dasar:
 - Transpose, Penjumlahan,
 - Perkalian dan Inverse
- Membuat Kalkulator Scientific

B. Screenshoot

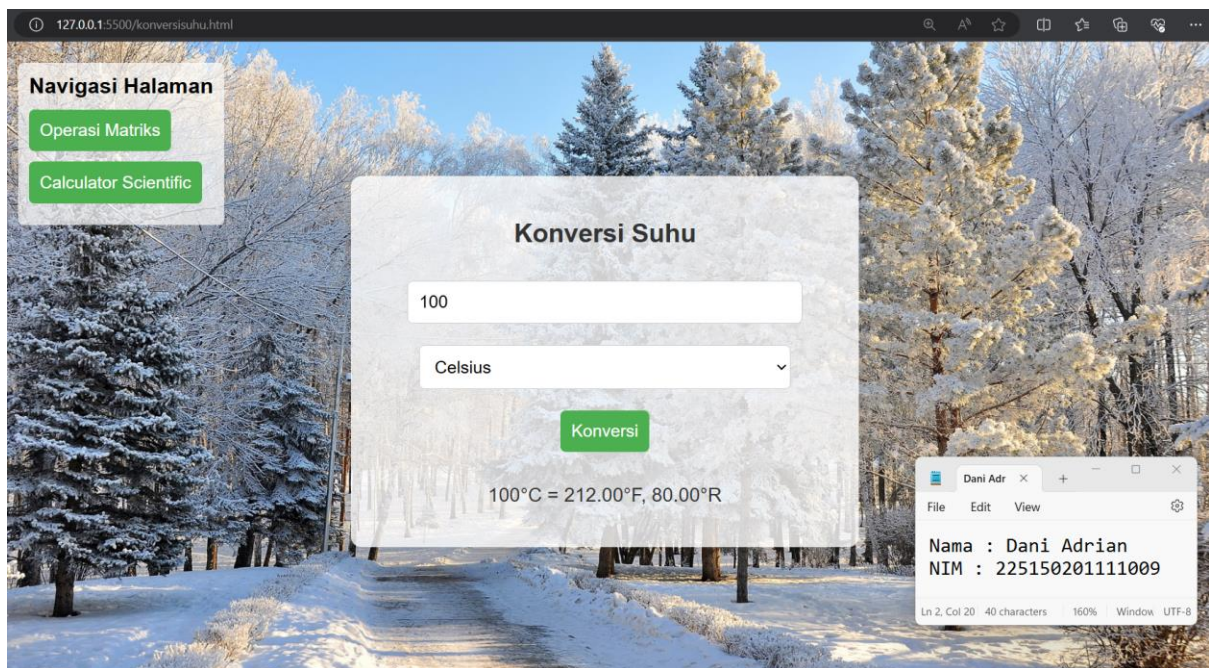
Konversi antar satuan Suhu

konversisuhu.html

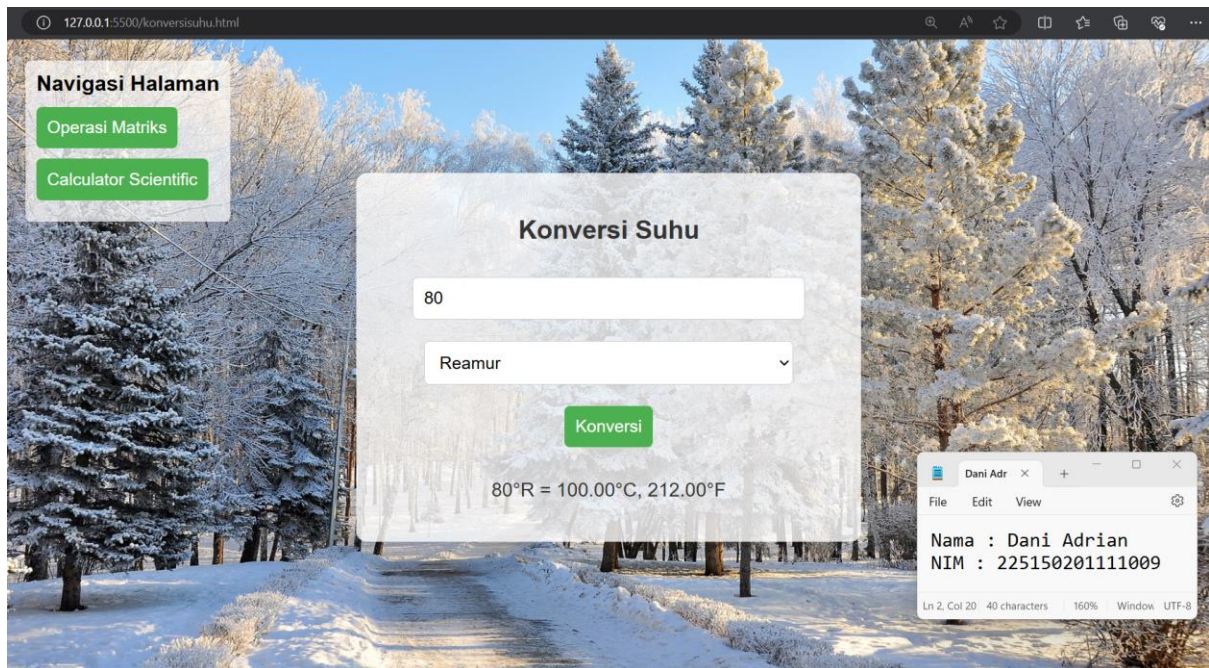
- Fahrenheit



- Celsius

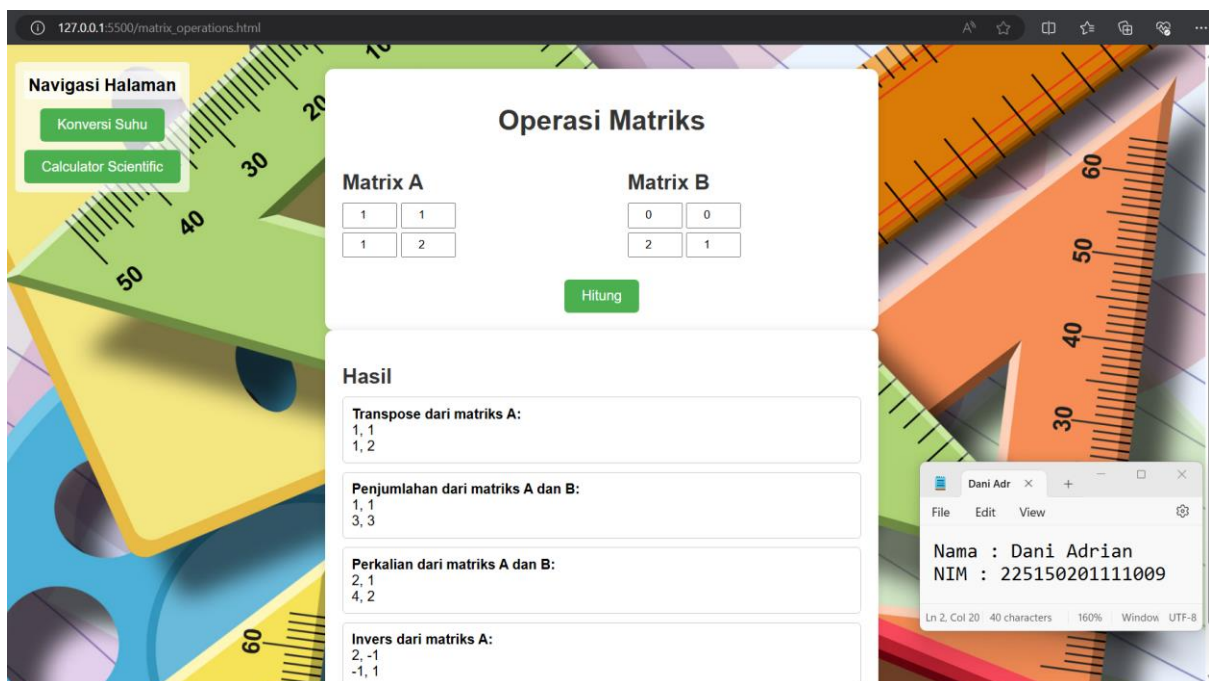


- Reamur



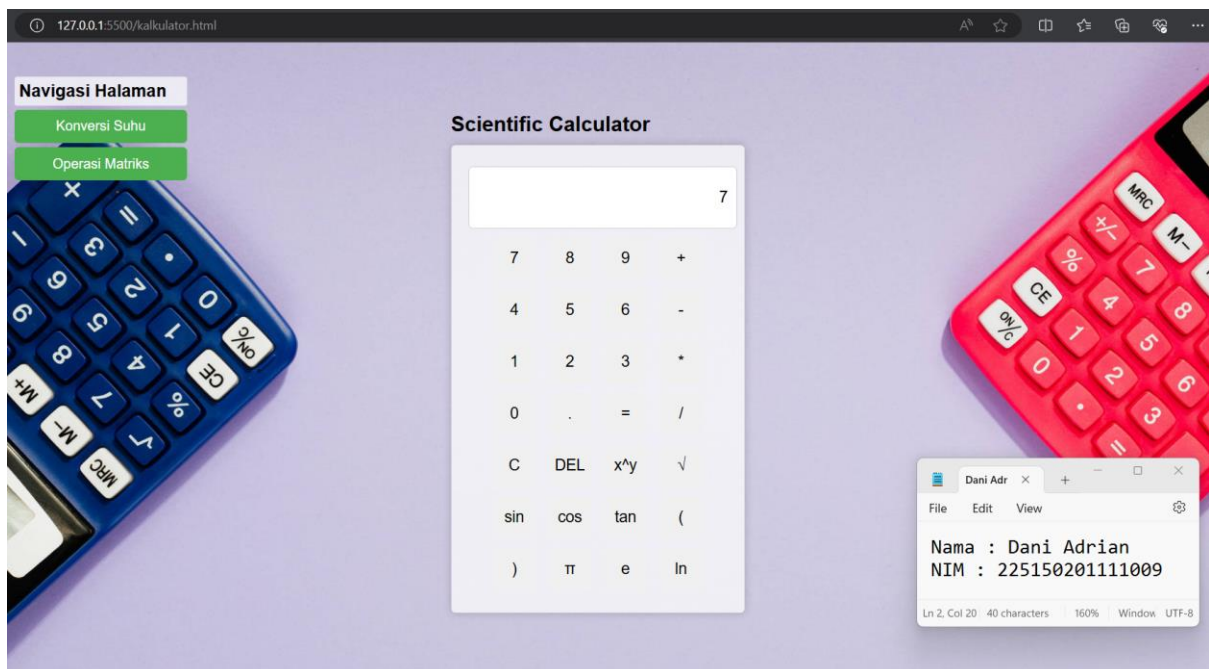
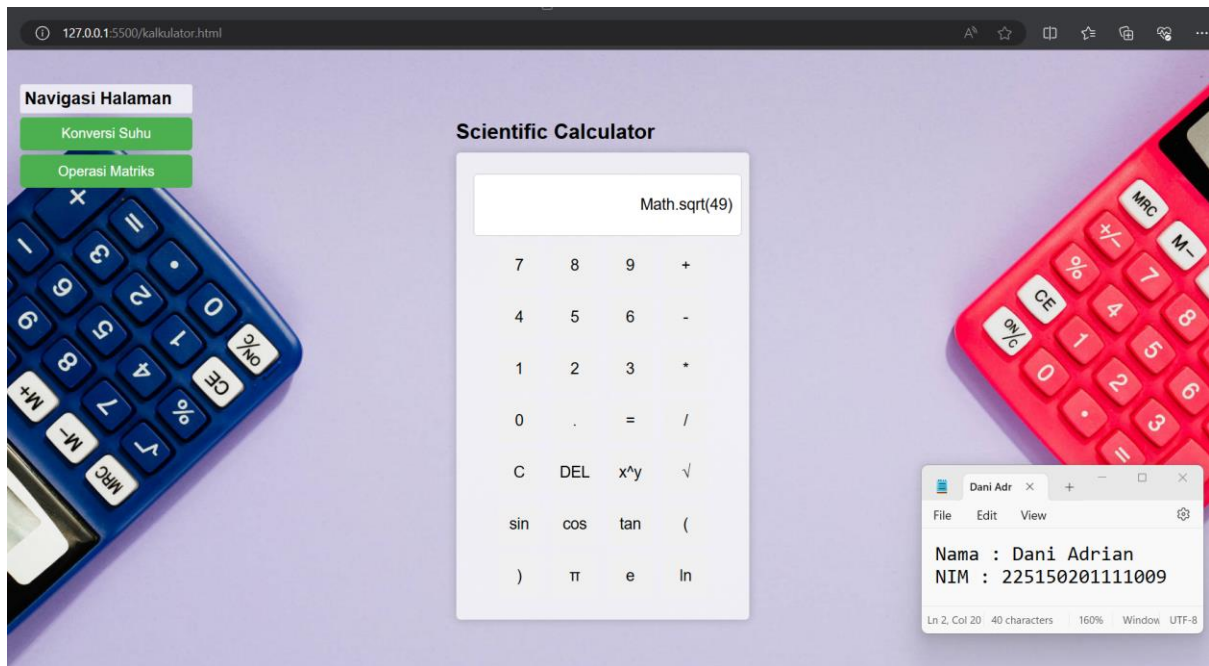
Operasi Matrik Dasar

matrix_operations.html



Kalkulator Scientific

kalkulator.html



C. Syntax

konversisuhu.html	
1.	<!DOCTYPE html>
2.	<html lang="en">
3.	
4.	<head>
5.	<title>Konversi Suhu</title>
6.	<style>
7.	body {
8.	font-family: Arial, sans-serif;
9.	display: flex;
10.	justify-content: center;

11.		align-items: center;
12.		height: 100vh;
13.		margin: 0;
14.		background-image: url('winter.jpg');
15.		background-size: cover;
16.		background-position: center;
17.		background-repeat: no-repeat;
18.		}
19.		
20.		.container {
21.		background-color: rgba(255, 255, 255,
22.	0.8);	
23.		padding: 20px;
24.		border-radius: 10px;
25.	0.1);	box-shadow: 0 0 10px rgba(0, 0, 0,
26.		text-align: center;
27.		}
28.		h2 {
29.		color: #333;
30.		}
31.		
32.		input,
33.		select,
34.		button {
35.		margin: 10px 0;
36.		padding: 10px;
37.		width: 80%;
38.		font-size: 16px;
39.		border: 1px solid #ccc;
40.		border-radius: 5px;
41.		}
42.		
43.		button {
44.		background-color: #4CAF50;
45.		color: white;
46.		border: none;
47.		cursor: pointer;
48.		width: auto;
49.		}
50.		
51.		button:hover {
52.		background-color: #45a049;
53.		}
54.		
55.		p#result {
56.		font-size: 18px;
57.		color: #333;
58.		margin-top: 20px;
59.		}
60.		
61.		/* Tombol navigasi */
62.		.navigation {
63.		position: absolute;

64.	top: 20px;
65.	left: 20px;
66.	text-align: left;
67.	font-weight: bold;
68.	font-size: 20px;
69.	background-color: rgba(255, 255, 255,
70.	0.7);
71.	/* Latar belakang putih agak transparan
72.	*/
73.	padding: 10px;
74.	border-radius: 5px;
75.	}
76.	.navigation p {
77.	margin: 0;
78.	}
79.	.navigation button {
80.	display: block;
81.	margin-top: 10px;
82.	}
83.	</style>
84.	</head>
85.	
86.	<body>
87.	<!-- Tombol navigasi -->
88.	<div class="navigation">
89.	<p>Navigasi Halaman</p>
90.	<button
91.	onclick="window.location.href='http://127.0.0.1:5500/matrix_operations.html'">Operasi Matriks</button>
92.	<button
93.	onclick="window.location.href='http://127.0.0.1:5500/kalkulator.html'">Calculator Scientific</button>
94.	</div>
95.	<div class="container">
96.	<h2>Konversi Suhu</h2>
97.	<input id="inputTemp" type="number"
98.	placeholder="Masukkan suhu">
99.	<select id="inputUnit">
100.	<option value="C">Celsius</option>
101.	<option value="F">Fahrenheit</option>
102.	<option value="R">Reamur</option>
103.	</select>
104.	<button
105.	onclick="convertTemperature()">Konversi</button>
106.	<p id="result"></p>
107.	</div>
108.	<script>
	function convertTemperature() {
	let temp =
	parseFloat(document.getElementById('inputTemp').value);

109.	let unit =
	document.getElementById('inputUnit').value;
110.	let result = '';
111.	
112.	if (unit === 'C') {
113.	result = `\${temp}°C = \${((temp * 9 /
	5 + 32).toFixed(2)}°F, \${((temp * 4 /
	5).toFixed(2)}°R`;
114.	} else if (unit === 'F') {
115.	result = `\${temp}°F = \${(((temp -
	32) * 5 / 9).toFixed(2)}°C, \${(((temp - 32) * 4 /
	9).toFixed(2)}°R`;
116.	} else if (unit === 'R') {
117.	result = `\${temp}°R = \${((temp * 5 /
	4).toFixed(2)}°C, \${((temp * 9 / 4 +
	32).toFixed(2)}°F`;
118.	}
119.	
120.	document.getElementById('result').innerText =
	result;
121.	}
122.	</script>
123.	</body>
124.	
125.	</html>

matrix_operations.html	
1.	<!DOCTYPE html>
2.	<html lang="en">
3.	
4.	<head>
5.	<meta charset="UTF-8">
6.	<title>Operasi Matriks</title>
7.	<script
	src="https://cdn.jsdelivr.net/npm/@tensorflow/tfjs"
	></script>
8.	<style>
9.	body {
10.	font-family: Arial, sans-serif;
11.	padding: 20px;
12.	background-image: url('matriksbg.jpg');
13.	background-size: cover;
14.	background-position: center;
15.	height: 100vh;
16.	}
17.	
18.	h1 {
19.	color: #333;
20.	text-align: center;
21.	margin-bottom: 20px;
22.	}
23.	
24.	h2 {
25.	color: #333;

26.	margin-bottom: 10px;
27.	}
28.	
29.	.container {
30.	background-color: #fff;
31.	padding: 20px;
32.	border-radius: 10px;
33.	box-shadow: 0 0 10px rgba(0, 0, 0,
	0.1);
34.	max-width: 600px;
35.	margin: 0 auto;
36.	}
37.	
38.	.input-container {
39.	display: flex;
40.	justify-content: space-between;
41.	margin-bottom: 20px;
42.	}
43.	
44.	.input-matrix {
45.	width: 45%;
46.	}
47.	
48.	input {
49.	margin-bottom: 5px;
50.	padding: 5px;
51.	width: 50px;
52.	text-align: center;
53.	}
54.	
55.	button {
56.	background-color: #4CAF50;
57.	color: white;
58.	border: none;
59.	cursor: pointer;
60.	padding: 10px 20px;
61.	border-radius: 5px;
62.	font-size: 16px;
63.	display: block;
64.	margin: 0 auto;
65.	}
66.	
67.	button:hover {
68.	background-color: #45a049;
69.	}
70.	
71.	#output-container {
72.	background-color: #fff;
73.	padding: 20px;
74.	border-radius: 10px;
75.	box-shadow: 0 0 10px rgba(0, 0, 0,
	0.1);
76.	max-width: 600px;
77.	margin: 0 auto;
78.	}

79.	
80.	#output {
81.	margin-top: 10px;
82.	}
83.	
84.	#output div {
85.	margin-bottom: 10px;
86.	border: 1px solid #ccc;
87.	padding: 10px;
88.	border-radius: 5px;
89.	}
90.	
91.	.warning-box {
92.	background-color: #ffcccc;
93.	color: #cc0000;
94.	border: 1px solid #cc0000;
95.	padding: 10px;
96.	border-radius: 5px;
97.	margin-bottom: 10px;
98.	}
99.	
100.	/* Tombol navigasi */
101.	.navigation {
102.	position: absolute;
103.	top: 20px;
104.	left: 20px;
105.	text-align: left;
106.	font-weight: bold;
107.	font-size: 20px;
108.	background-color: rgba(255, 255, 255,
109.	0.7);
110.	/* Latar belakang putih agak transparan
111.	*/
112.	padding: 10px;
113.	border-radius: 5px;
114.	}
115.	.navigation p {
116.	margin: 0;
117.	}
118.	.navigation button {
119.	display: block;
120.	margin-top: 10px;
121.	}
122.	</style>
123.	<script>
124.	async function matrixToString(matrix) {
125.	return matrix.array().then(array =>
126.	array.map(row => row.join(', ')).join(' '));
127.	}
128.	function displayResult(elementId,
	resultString) {

129.	const outputDiv =
130.	document.getElementById(elementId);
131.	const resultDiv =
132.	document.createElement('div');
133.	resultDiv.innerHTML = resultString;
134.	outputDiv.appendChild(resultDiv);
135.	}
136.	function getMatrixValues(idPrefix) {
137.	const values = [
138.	[parseFloat(document.getElementById(`\${idPrefix}00`)
139.	.value 0),
140.	parseFloat(document.getElementById(`\${idPrefix}01`)
141.	.value 0)],
142.	[parseFloat(document.getElementById(`\${idPrefix}10`)
143.	.value 0),
144.	parseFloat(document.getElementById(`\${idPrefix}11`)
145.	.value 0)]
146.];
147.	return tf.tensor2d(values);
148.	}
149.	function calculateInverse(matrix) {
150.	const array = matrix.arraySync();
151.	const determinant = array[0][0] *
152.	array[1][1] - array[0][1] * array[1][0];
153.	if (determinant === 0) {
154.	throw new Error("Matrix is singular
155.	and cannot be inverted");
156.	}
157.	const inverseArray = [
158.	[array[1][1] / determinant, -
159.	array[0][1] / determinant],
160.	[-array[1][0] / determinant,
161.	array[0][0] / determinant]
162.];
163.	return tf.tensor2d(inverseArray);
164.	}
165.	async function matrixOperations() {
166.	// Clear previous results and warnings
167.	document.getElementById('output').innerHTML = '';
168.	document.getElementById
	('warning').innerHTML = '';
	// Get input values
	const a = getMatrixValues('a');
	const b = getMatrixValues('b');
	// Check if any input is empty
	let anyEmpty = false;

169.	document.querySelectorAll('input').forEach(input =>
170.	{
171.	if (input.value === '') {
172.	anyEmpty = true;
173.	return;
174.	}
175.	});
176.	// Show warning if any input is empty
177.	if (anyEmpty) {
178.	document.getElementById('warning').innerHTML =
179.	'<div class="warning-box">Mohon isi semua kolom
180.	input</div>';
181.	return;
182.	}
183.	// Transpose
184.	const aTranspose = a.transpose();
185.	const aTransposeStr = await
186.	matrixToString(aTranspose);
187.	displayResult('output', `Transpose
188.	dari matriks
189.	A: \${aTransposeStr}`);
190.	// Penjumlahan
191.	const sum = a.add(b);
192.	const sumStr = await
193.	matrixToString(sum);
194.	displayResult('output', `Penjumlahan
195.	dari matriks A dan B: \${sumStr}`);
196.	// Perkalian
197.	const product = a.matMul(b);
198.	const productStr = await
199.	matrixToString(product);
200.	displayResult('output', `Perkalian
201.	dari matriks A dan B: \${productStr}`);
202.	// Invers
203.	try {
204.	const inverseA =
205.	calculateInverse(a);
206.	const inverseAStr = await
207.	matrixToString(inverseA);
208.	displayResult('output', `Invers
209.	dari matriks A: \${inverseAStr}`);
	} catch (error) {
	displayResult('output', `Invers
	dari matriks A: Error: \${error.message}`);
	}
	}
	</script>
	</head>

210.	<body>
211.	<!-- Tombol navigasi -->
212.	<div class="navigation">
213.	<p style="background-color: rgba(255, 255, 255, 0.7); padding: 5px; border-radius: 3px;">Navigasi Halaman</p>
214.	<button onclick="window.location.href='http://127.0.0.1:5500/konversisuhu.html'">Konversi Suhu</button>
215.	<button onclick="window.location.href='http://127.0.0.1:5500/kalkulator.html'">Calculator Scientific</button>
216.	</div>
217.	<div class="container">
218.	<h1>Operasi Matriks</h1>
219.	<div class="input-container">
220.	<div class="input-matrix">
221.	<h2>Matrix A</h2>
222.	<input type="number" id="a00"
223.	placeholder="">
224.	<input type="number" id="a01"
225.	placeholder="">
226.	
227.	<input type="number" id="a10"
228.	placeholder="">
229.	<input type="number" id="a11"
230.	placeholder="">
231.	</div>
232.	<div class="input-matrix">
233.	<h2>Matrix B</h2>
234.	<input type="number" id="b00"
235.	placeholder="">
236.	<input type="number" id="b01"
237.	placeholder="">
238.	
239.	<input type="number" id="b10"
240.	placeholder="">
241.	<input type="number" id="b11"
242.	placeholder="">
243.	</div>
244.	</div>
245.	<button onclick="matrixOperations()">Hitung</button>
246.	<div id="warning"></div>
247.	</div>
248.	
249.	<div class="container" id="output-container">
250.	<h2>Hasil</h2>
251.	<div id="output"></div>
252.	</div>
253.	</body>
254.	</html>

1.	<!DOCTYPE html>
2.	<html lang="en">
3.	
4.	<head>
5.	<meta charset="UTF-8">
6.	<meta name="viewport" content="width=device-
7.	width, initial-scale=1.0">
8.	<title>Scientific Calculator</title>
9.	<style>
10.	body {
11.	font-family: Arial, sans-serif;
12.	margin: 0;
13.	padding: 0;
14.	background-image:
15.	url('calculator.jpg');
16.	background-size: cover;
17.	background-position: center;
18.	height: 100vh;
19.	display: flex;
20.	justify-content: center;
21.	align-items: center;
22.	}
23.	.calculator-container {
24.	width: 360px;
25.	}
26.	.calculator {
27.	width: 300px;
28.	padding: 20px;
29.	background-color: rgba(255, 255, 255,
30.	0.7);
31.	border-radius: 5px;
32.	box-shadow: 0 0 10px rgba(0, 0, 0,
33.	0.2);
34.	text-align: center;
35.	}
36.	input[type="button"] {
37.	width: 50px;
38.	height: 50px;
39.	margin: 5px;
40.	font-size: 18px;
41.	cursor: pointer;
42.	border: none;
43.	border-radius: 5px;
44.	}
45.	input[type="text"] {
46.	width: calc(100% - 10px);
47.	height: 50px;
48.	margin: 5px auto;
49.	padding: 10px;
50.	font-size: 18px;
51.	border: 1px solid #ccc;

52.	border-radius: 5px;
53.	text-align: right;
54.	}
55.	
56.	.title {
57.	text-align: left;
58.	margin-bottom: 10px;
59.	font-size: 24px;
60.	font-weight: bold;
61.	}
62.	
63.	/* Tombol navigasi */
64.	.navigation {
65.	position: absolute;
66.	top: 20px;
67.	left: 20px;
68.	text-align: left;
69.	font-weight: bold;
70.	font-size: 20px;
71.	}
72.	
73.	.navigation button {
74.	display: block;
75.	margin-bottom: 5px;
76.	width: 200px;
77.	background-color: #4CAF50;
78.	color: white;
79.	border: none;
80.	cursor: pointer;
81.	padding: 10px;
82.	border-radius: 5px;
83.	font-size: 16px;
84.	}
85.	
86.	.navigation button:hover {
87.	background-color: #45a049;
88.	}
89.	
90.	.navigation p {
91.	background-color: rgba(255, 255, 255,
92.	0.7);
93.	padding: 5px;
94.	border-radius: 3px;
95.	margin-bottom: 5px;
96.	}
97.	</style>
98.	</head>
99.	<body>
100.	<!-- Tombol navigasi -->
101.	<div class="navigation">
102.	<p>Navigasi Halaman</p>
103.	<button
	onclick="window.location.href='http://127.0.0.1:5500/konversisuhu.html'">Konversi Suhu</button>

104.	<button onclick="window.location.href='http://127.0.0.1:5500/matrix_operations.html'">Operasi Matriks</button>
105.	</div>
106.	
107.	<div class="calculator-container">
108.	<div class="title">Scientific Calculator</div>
109.	<div class="calculator">
110.	<input type="text" id="display" readonly>
111.	
112.	<input type="button" value="7" onclick="appendToDisplay('7') ">
113.	<input type="button" value="8" onclick="appendToDisplay('8') ">
114.	<input type="button" value="9" onclick="appendToDisplay('9') ">
115.	<input type="button" value="+" onclick="appendToDisplay('+') ">
116.	
117.	<input type="button" value="4" onclick="appendToDisplay('4') ">
118.	<input type="button" value="5" onclick="appendToDisplay('5') ">
119.	<input type="button" value="6" onclick="appendToDisplay('6') ">
120.	<input type="button" value="-" onclick="appendToDisplay('-') ">
121.	
122.	<input type="button" value="1" onclick="appendToDisplay('1') ">
123.	<input type="button" value="2" onclick="appendToDisplay('2') ">
124.	<input type="button" value="3" onclick="appendToDisplay('3') ">
125.	<input type="button" value="*" onclick="appendToDisplay('*') ">
126.	
127.	<input type="button" value="0" onclick="appendToDisplay('0') ">
128.	<input type="button" value="." onclick="appendToDisplay('.') ">
129.	<input type="button" value="=" onclick="calculate() ">
130.	<input type="button" value="/"> onclick="appendToDisplay('/') ">
131.	
132.	<input type="button" value="C" onclick="clearDisplay() ">
133.	<input type="button" value="DEL" onclick="deleteLast() ">
134.	<input type="button" value="x^y" onclick="appendToDisplay('**') ">
135.	<input type="button" value="√" onclick="sqrt() ">

136.	
137.	<input type="button" value="sin" onclick="trigFunction('sin')">
138.	<input type="button" value="cos" onclick="trigFunction('cos')">
139.	<input type="button" value="tan" onclick="trigFunction('tan')">
140.	<input type="button" value="(" onclick="appendToDisplay('(')">
141.	
142.	<input type="button" value=")" onclick="appendToDisplay(')')">
143.	<input type="button" value="π" onclick="appendToDisplay(Math.PI)">
144.	<input type="button" value="e" onclick="appendToDisplay(Math.E)">
145.	<input type="button" value="ln" onclick="log()">
146.	</div>
147.	</div>
148.	
149.	<script>
150.	function appendToDisplay(value) {
151.	document.getElementById('display').value += value;
152.	}
153.	
154.	function clearDisplay() {
155.	document.getElementById('display').value = '';
156.	}
157.	
158.	function deleteLast() {
159.	var
160.	value =
161.	document.getElementById('display').value;
162.	document.getElementById('display').value =
163.	value.substring(0, value.length - 1);
164.	}
165.	
166.	function calculate() {
167.	try {
168.	var result =
169.	eval(document.getElementById('display').value.replace('^', '**'));
170.	document.getElementById('display').value = result;
171.	} catch (error) {
172.	document.getElementById('display').value = 'Error';
173.	}
	}
	function trigFunction(func) {

174.	var value =
175.	document.getElementById('display').value;
176.	if (value === '') {
177.	document.getElementById('display').value =
178.	'Math.' + func + '(';
179.	} else {
180.	document.getElementById('display').value = 'Math.'
181.	+ func + '(' + value + ')';
182.	}
183.	}
184.	function log() {
185.	var value =
186.	document.getElementById('display').value;
187.	if (value === '') {
188.	document.getElementById('display').value =
189.	'Math.log(';
190.	} else {
191.	document.getElementById('display').value =
192.	'Math.log(' + value + ')';
193.	}
194.	}
195.	function sqrt() {
196.	var value =
197.	document.getElementById('display').value;
198.	if (value === '') {
199.	document.getElementById('display').value =
200.	'Math.sqrt(';
201.	} else {
202.	document.getElementById('display').value =
203.	'Math.sqrt(' +
204.	value + ')';
205.	}
	}
	</script>
	</body>
	</html>

D. Penjelasan

konversisuhu.html

Halaman ini digunakan untuk konversi suhu antara tiga skala: Celsius (C), Fahrenheit (F), dan Reamur (R).

matrix_operations.html

Halaman ini menyediakan berbagai operasi matriks seperti transpose, penjumlahan, perkalian, dan invers.

kalkulator.html

Halaman ini digunakan sebagai kalkulator ilmiah yang menyediakan berbagai fungsi matematika.

3. Membuat Implementasi Interaktif dan mendukung CRUD dengan Full JS (Dome, dll.)

LATIHAN 1

A. Soal

- a. Perhatikan JS DOM berikut:

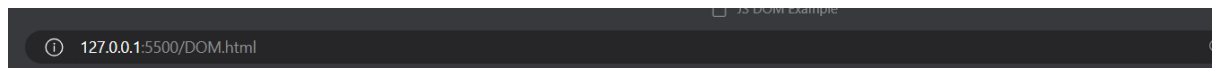
```
..  
  
<body>  
  <p class="id">Belajar Form</p>  
  <p class="id">Belajar DB MySQL</p>  
  <p class="id">Belajar JS</p>  
  <p class="id">Belajar Ajax</p>  
  <script>  
    var val = document.getElementsByClassName("id");  
    console.log(val);  
  </script>  
..
```

Dan

```
setInterval(function() {  
  val[0].style.color = "red";  
  val[1].style.color = "green";  
  val[2].style.color = "blue";  
  val[3].style.color = "orange";  
  
  setTimeout(function() {  
    val[0].style.color = "black";  
    val[1].style.color = "black";  
    val[2].style.color = "black";  
    val[3].style.color = "black";  
  }, 1000);  
, 2000);
```

Bagaimana hasil yang ditampilkan jika pada “console..” diubah dengan “setInterval..” di atas!

B. Screenshoot

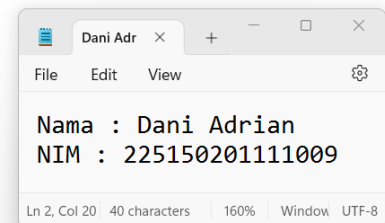


Belajar Form

Belajar DB MySQL

Belajar JS

Belajar Ajax

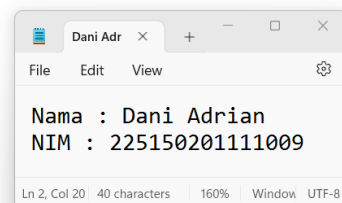


Belajar Form

Belajar DB MySQL

Belajar JS

Belajar Ajax



C. Syntax

DOM.html	
1.	<!DOCTYPE html>
2.	<html lang="en">
3.	
4.	<head>

5.	<meta charset="UTF-8">
6.	<meta name="viewport" content="width=device-
7.	width, initial-scale=1.0">
8.	<title>JS DOM Example</title>
9.	</head>
10.	<body>
11.	<p class="id">Belajar Form</p>
12.	<p class="id">Belajar DB MySQL</p>
13.	<p class="id">Belajar JS</p>
14.	<p class="id">Belajar Ajax</p>
15.	<script>
16.	var val =
17.	document.getElementsByClassName("id");
18.	console.log(val);
19.	setInterval(function () {
20.	val[0].style.color = "red";
21.	val[1].style.color = "green";
22.	val[2].style.color = "blue";
23.	val[3].style.color = "orange";
24.	
25.	setTimeout(function () {
26.	val[0].style.color = "black";
27.	val[1].style.color = "black";
28.	val[2].style.color = "black";
29.	val[3].style.color = "black";
30.	}, 1000);
31.	}, 2000);
32.	</script>
33.	</body>
34.	
35.	</html>

D. Penjelasan

Setiap 2 detik:

- Warna teks paragraf pertama (<p class="id">Belajar Form</p>) berubah menjadi **merah**.
- Warna teks paragraf kedua (<p class="id">Belajar DB MySQL</p>) berubah menjadi **hijau**.
- Warna teks paragraf ketiga (<p class="id">Belajar JS</p>) berubah menjadi **biru**.
- Warna teks paragraf keempat (<p class="id">Belajar Ajax</p>) berubah menjadi **oranye**.

Setelah 1 detik dari perubahan warna di atas:

Semua paragraf kembali berubah warna menjadi hitam.

Visualisasi Perubahan

- 0 detik:
 - Semua teks berwarna hitam (default).
- 2 detik:
 - Teks berubah warna:
 - "Belajar Form" -> Merah
 - "Belajar DB MySQL" -> Hijau
 - "Belajar JS" -> Biru
 - "Belajar Ajax" -> Oranye
- 3 detik:
 - Semua teks kembali berwarna hitam.
- 4 detik:
 - Ulangi perubahan warna seperti di 2 detik.
- 5 detik:
 - Semua teks kembali berwarna hitam.

Dan seterusnya setiap 2 detik, perubahan warna akan terjadi lagi, dan setiap 1 detik setelah perubahan warna, teks akan kembali ke warna hitam.

Kesimpulan

Paragraf-paragraf dengan kelas "id" akan mengalami perubahan warna secara berulang-ulang setiap 2 detik. Warna tersebut akan berubah kembali ke hitam setelah 1 detik dari setiap perubahan warna.

LATIHAN 2

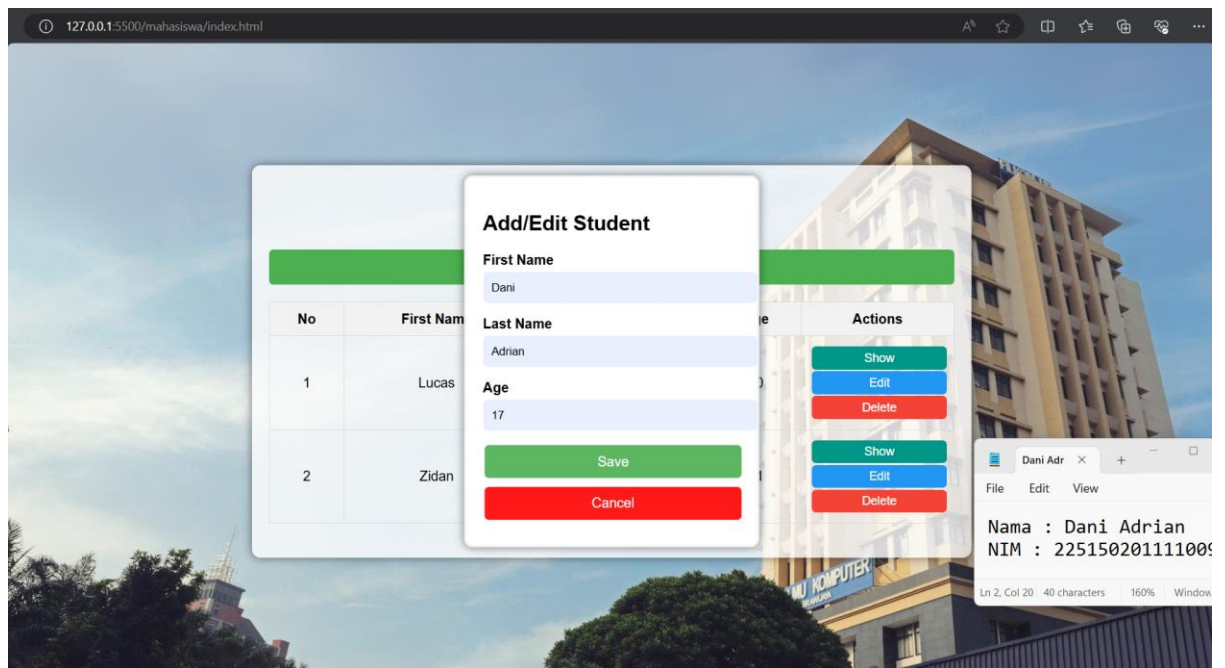
A. Soal

Buatlah Web “Students App” CRUD sederhana yang datanya pada temp. dgn kode full JS (Lib & Framework JS) + HTML + CSS, seperti tampilan berikut untuk manage data Mahasiswa secara dinamis dan interaktif!

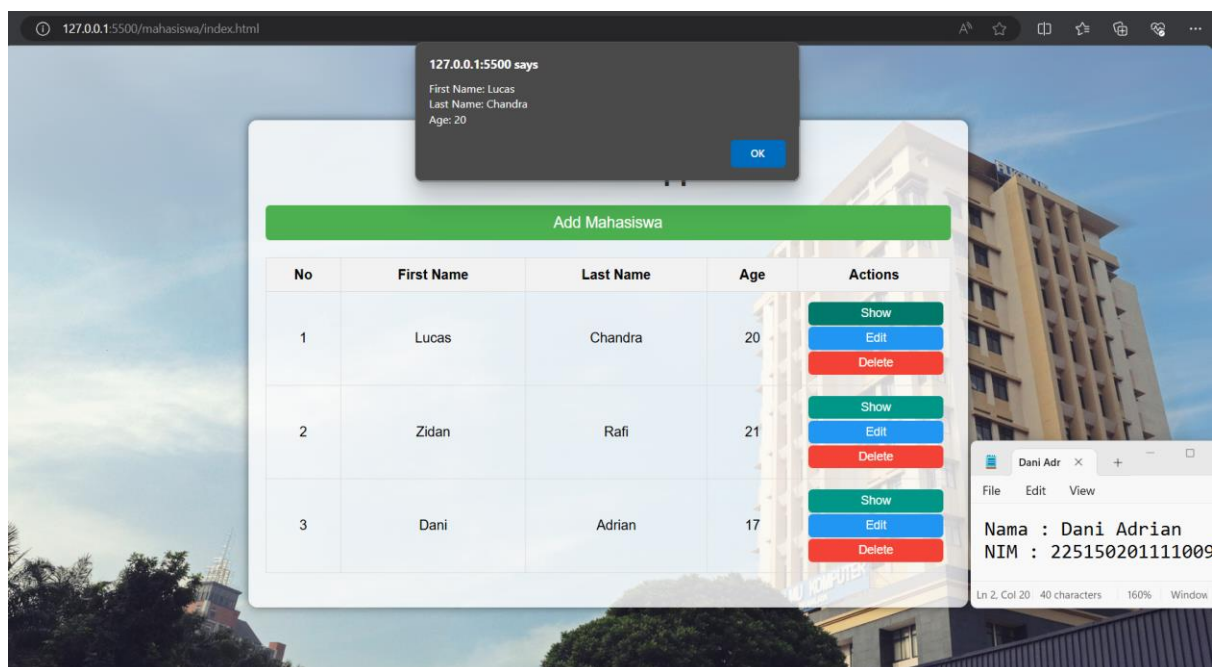


B. Screenshoot

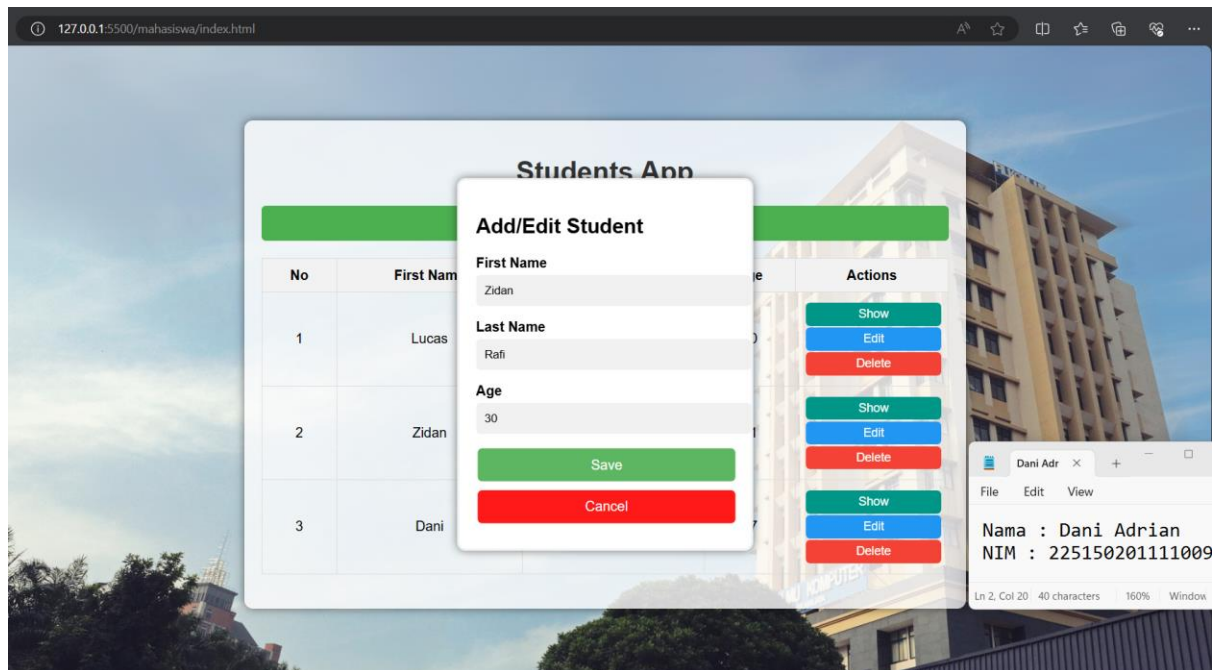
Add



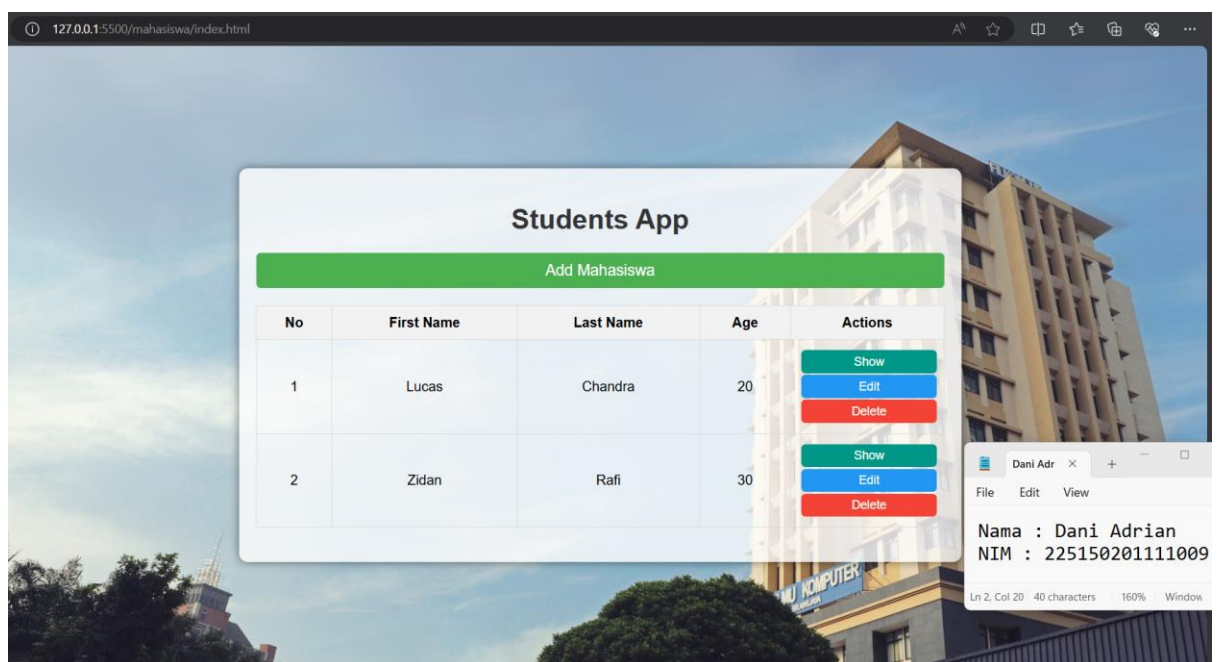
Show



Edit



Delete



C. Syntax

index.html	
1.	<!DOCTYPE html>
2.	<html lang="en">
3.	
4.	<head>
5.	<meta charset="UTF-8">

6.	<meta name="viewport" content="width=device-
7.	width, initial-scale=1.0">
8.	<title>Students App</title>
9.	<link rel="stylesheet" href="styles.css">
10.	</head>
11.	<body>
12.	<div class="container">
13.	<h1>Students App</h1>
14.	<button id="addStudentBtn"
15.	onclick="openForm()">Add Mahasiswa</button>
16.	<table id="studentsTable">
17.	<thead>
18.	<tr>
19.	<th>No</th>
20.	<th>First Name</th>
21.	<th>Last Name</th>
22.	<th>Age</th>
23.	<th>Actions</th>
24.	</tr>
25.	</thead>
26.	<tbody>
27.	<!-- Students will be dynamically
28.	added here -->
29.	</tbody>
30.	</table>
31.	</div>
32.	<div class="form-popup" id="studentForm">
33.	<form id="form" class="form-container"
34.	onsubmit="saveStudent(event)">
35.	<h2>Add/Edit Student</h2>
36.	<label for="firstName">First
37.	Name</label>
38.	<input type="text" id="firstName"
39.	placeholder="Enter First Name" required>
40.	<label for="lastName">Last
41.	Name</label>
42.	<input type="text" id="lastName"
43.	placeholder="Enter Last Name" required>
44.	<label for="age">Age</label>
45.	<input type="number" id="age"
46.	placeholder="Enter Age" required>
47.	<input type="hidden" id="studentIndex">
48.	<button type="submit"
49.	class="btn">Save</button>
50.	<button type="button" class="btn
	cancel" onclick="closeForm()">Cancel</button>
	</form>
	</div>

51.	<pre> <script src="script.js"></script> </body> </html> </pre>
52.	
53.	
54.	
55.	

styles.css	
1.	body {
2.	font-family: Arial, sans-serif;
3.	margin: 0;
4.	padding: 0;
5.	background-image: url('filkom.jpg');
6.	background-size: cover;
7.	background-position: center;
8.	height: 100vh;
9.	display: flex;
10.	justify-content: center;
11.	align-items: center;
12.	}
13.	
14.	.container {
15.	background-color: rgba(255, 255, 255, 0.8);
16.	padding: 20px;
17.	border-radius: 10px;
18.	box-shadow: 0 0 10px rgba(0, 0, 0, 0.5);
19.	width: 90%;
20.	max-width: 800px;
21.	margin: 20px;
22.	}
23.	
24.	h1 {
25.	text-align: center;
26.	color: #333;
27.	}
28.	
29.	button {
30.	background-color: #4CAF50;
31.	color: white;
32.	padding: 10px 20px;
33.	margin: 20px 0;
34.	border: none;
35.	cursor: pointer;
36.	border-radius: 5px;
37.	display: block;
38.	width: 100%;
39.	text-align: center;
40.	font-size: 18px;
41.	}
42.	
43.	button:hover {
44.	background-color: #45a049;
45.	}
46.	
47.	table {

48.	width: 100%;
49.	border-collapse: collapse;
50.	margin-bottom: 20px;
51.	}
52.	
53.	table, th, td {
54.	border: 1px solid #ddd;
55.	}
56.	
57.	th, td {
58.	padding: 10px;
59.	text-align: center;
60.	}
61.	
62.	th {
63.	background-color: #f2f2f2;
64.	}
65.	
66.	.btn {
67.	padding: 5px 10px;
68.	margin: 2px;
69.	border: none;
70.	border-radius: 5px;
71.	cursor: pointer;
72.	font-size: 14px;
73.	}
74.	
75.	.btn.show {
76.	background-color: #009688;
77.	color: white;
78.	transition: background-color 0.3s ease;
79.	}
80.	
81.	.btn.show:hover {
82.	background-color: #00796b;
83.	}
84.	
85.	.btn.edit {
86.	background-color: #2196F3;
87.	color: white;
88.	transition: background-color 0.3s ease;
89.	}
90.	
91.	.btn.edit:hover {
92.	background-color: #1976d2;
93.	}
94.	
95.	.btn.delete {
96.	background-color: #f44336;
97.	color: white;
98.	transition: background-color 0.3s ease;
99.	}
100.	
101.	.btn.delete:hover {
102.	background-color: #d32f2f;

103.	}
104.	
105.	.form-popup {
106.	display: none;
107.	position: fixed;
108.	top: 50%;
109.	left: 50%;
110.	transform: translate(-50%, -50%);
111.	border: 3px solid #f1f1f1;
112.	z-index: 9;
113.	background-color: white;
114.	padding: 20px;
115.	box-shadow: 0 0 10px rgba(0, 0, 0, 0.5);
116.	border-radius: 10px;
117.	}
118.	
119.	.form-container {
120.	max-width: 300px;
121.	}
122.	
123.	.form-container input[type="text"],
124.	.form-container input[type="number"] {
125.	width: 100%;
126.	padding: 10px;
127.	margin: 5px 0 15px 0;
128.	border: none;
129.	background: #f1f1f1;
130.	border-radius: 5px;
131.	}
132.	
133.	.form-container input[type="text"]:focus,
134.	.form-container input[type="number"]:focus {
135.	background-color: #ddd;
136.	outline: none;
137.	}
138.	
139.	.form-container .btn {
140.	background-color: #4CAF50;
141.	color: white;
142.	border: none;
143.	cursor: pointer;
144.	width: 100%;
145.	margin-bottom: 10px;
146.	opacity: 0.9;
147.	padding: 10px;
148.	font-size: 16px;
149.	}
150.	
151.	.form-container .cancel {
152.	background-color: red;
153.	}
154.	
155.	.form-container .btn:hover, .open-button:hover {
156.	opacity: 1;
157.	}

script.js	
1.	let students = [];
2.	let editIndex = -1;
3.	
4.	function openForm() {
5.	document.getElementById("studentForm").style.display
	= "block";
6.	document.getElementById("form").reset();
7.	editIndex = -1;
8.	}
9.	
10.	function closeForm() {
11.	document.getElementById("studentForm").style.display
	= "none";
12.	}
13.	
14.	function saveStudent(event) {
15.	event.preventDefault();
16.	
17.	let firstName =
	document.getElementById("firstName").value;
18.	let lastName =
	document.getElementById("lastName").value;
19.	let age = document.getElementById("age").value;
20.	
21.	if (editIndex === -1) {
22.	students.push({ firstName, lastName, age });
23.	} else {
24.	students[editIndex] = { firstName, lastName, age
	};
25.	}
26.	
27.	renderStudents();
28.	closeForm();
29.	}
30.	
31.	function renderStudents() {
32.	let tbody = document.querySelector("#studentsTable
	tbody");
33.	tbody.innerHTML = "";
34.	
35.	students.forEach((student, index) => {
36.	let row = `<tr>
37.	<td>\${index + 1}</td>
38.	<td>\${student.firstName}</td>
39.	<td>\${student.lastName}</td>
40.	<td>\${student.age}</td>
41.	<td>
42.	<button class="btn show"
	onclick="showStudent(\${index})">Show</button>
43.	<button class="btn edit"
	onclick="editStudent(\${index})">Edit</button>

44.	<code><button class="btn delete"</code>
45.	<code>onclick="deleteStudent(\${index})">Delete</button></code>
46.	<code></td></code>
47.	<code></tr>`;</code>
48.	<code>tbody.innerHTML += row;</code>
49.	<code>});</code>
50.	<code>}</code>
51.	<code>function showStudent(index) {</code>
52.	<code>let student = students[index];</code>
53.	<code>alert(</code>
54.	<code>`First Name: \${student.firstName}\nLast Name:</code>
55.	<code>\${student.lastName}\nAge: \${student.age}`</code>
56.	<code>);</code>
57.	<code>}</code>
58.	<code>function editStudent(index) {</code>
59.	<code>let student = students[index];</code>
60.	<code>document.getElementById("firstName").value =</code>
61.	<code>student.firstName;</code>
62.	<code>document.getElementById("lastName").value =</code>
63.	<code>student.lastName;</code>
64.	<code>document.getElementById("age").value =</code>
65.	<code>student.age;</code>
66.	<code>document.getElementById("studentForm").style.display</code>
67.	<code>= "block";</code>
68.	<code>editIndex = index;</code>
69.	<code>}</code>
70.	<code>function deleteStudent(index) {</code>
	<code>students.splice(index, 1);</code>
	<code>renderStudents();</code>
	<code>}</code>

D. Penjelasan

Syntax di atas membangun sebuah aplikasi web sederhana yang memungkinkan pengguna untuk melakukan operasi CRUD (Create, Read, Update, Delete) pada data mahasiswa. Tujuan utama dari aplikasi ini untuk memberikan antarmuka yang interaktif dan dinamis bagi pengguna untuk mengelola data mahasiswa. Aplikasi ini dibuat menggunakan HTML, CSS, dan JavaScript tanpa menggunakan library atau framework tambahan.