**Langkah 1: Buat Struktur Direktori**

Buat direktori untuk proyek Anda, misalnya spa-example, dan buat tiga file di dalamnya: index.html, styles.css, dan script.js.

**Langkah 2: Buat File HTML**

Buat file index.html dengan isi berikut:

html

Salin kode

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>SPA Example</title>

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

</head>

<body>

<header>

<nav>

<ul>

<li><a href="#home" onclick="navigate('home')">Home</a></li>

<li><a href="#about" onclick="navigate('about')">About</a></li>

<li><a href="#contact" onclick="navigate('contact')">Contact</a></li>

</ul>

</nav>

</header>

<main id="content">

<!-- Content will be injected here -->

</main>

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

</body>

</html>

**Penjelasan:**

* File ini berisi struktur dasar halaman HTML.
* Ada elemen header dengan navigasi yang mengarahkan ke halaman berbeda tanpa memuat ulang halaman.
* Elemen <main id="content"> akan menjadi tempat untuk konten halaman yang berubah-ubah.
* script.js di-include di akhir untuk menangani logika navigasi.

**Langkah 3: Buat File CSS**

Buat file styles.css dengan isi berikut:

css

Salin kode

body {

font-family: Arial, sans-serif;

margin: 0;

padding: 0;

}

header {

background-color: #333;

color: white;

padding: 10px 0;

}

nav ul {

list-style-type: none;

padding: 0;

display: flex;

justify-content: center;

}

nav ul li {

margin: 0 15px;

}

nav ul li a {

color: white;

text-decoration: none;

}

nav ul li a:hover {

text-decoration: underline;

}

main {

padding: 20px;

text-align: center;

}

**Penjelasan:**

* File ini memberikan gaya dasar untuk halaman.
* Mengatur tata letak header dan navigasi, serta memberikan padding dan margin yang sesuai.

**Langkah 4: Buat File JavaScript**

Buat file script.js dengan isi berikut:

javascript

Salin kode

const routes = {

'home': `

<h1>Home</h1>

<p>Welcome to the Home page!</p>

`,

'about': `

<h1>About</h1>

<p>This is the About page.</p>

`,

'contact': `

<h1>Contact</h1>

<p>This is the Contact page.</p>

`

};

function navigate(page) {

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

content.innerHTML = routes[page] || '<h1>404 - Page Not Found</h1>';

}

// Load default page

window.onload = () => {

const defaultPage = 'home';

navigate(defaultPage);

};

**Penjelasan:**

* routes adalah objek yang menyimpan konten HTML untuk setiap halaman.
* Fungsi navigate mengganti konten dalam elemen <main id="content"> sesuai halaman yang diklik.
* window.onload memastikan halaman default (Home) dimuat saat pertama kali membuka aplikasi.

**Langkah 5: Jalankan Aplikasi**

1. Buka file index.html di browser.
2. Anda akan melihat halaman Home.
3. Klik link navigasi (Home, About, Contact) untuk melihat konten halaman berubah tanpa memuat ulang halaman.

**Penjelasan Tambahan**

* **Navigasi Tanpa Reload**: Fungsi onclick pada link navigasi memanggil fungsi navigate untuk mengganti konten halaman secara dinamis.
* **Routing Sederhana**: Objek routes menyimpan konten halaman. Fungsi navigate mengganti konten elemen dengan ID content berdasarkan halaman yang dipilih.
* **Halaman Default**: Saat halaman pertama kali dimuat, window.onload memastikan halaman default (Home) ditampilkan.

Dengan langkah-langkah ini, Anda dapat membuat SPA sederhana yang dapat dikembangkan lebih lanjut sesuai kebutuhan Anda.