**MODUL V : search engine**

**KODE MODUL** : TIK.RPL02.002.001.01

**DESKRIPSI MODUL** : Pembuatan Search Engine Sederhana dengan JavaScript

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Elemen Kompetensi | Indikator Kinerja | Jml Jam | halaman |
| 1 | Menggunakan jQuery untuk validasi form | Form dapat divalidasi dengan jQuery untuk memastikan tidak ada kolom yang kosong dan panjang kata sandi minimal 6 karakter sebelum dikirim. | 2 |  |
| 2 | Memberikan feedback visual dengan jQuery | Elemen input yang tidak valid akan ditandai dengan warna merah dan pesan kesalahan akan ditampilkan menggunakan alert. | 1 |  |
|  | Mengelola event form dengan jQuery | Event submit form ditangani dengan jQuery untuk mencegah pengiriman form standar dan mengeksekusi validasi. | 1 |  |

**PRELAB**

Persiapan Modul Praktikum Validasi Form dengan JavaScript:

1. jQuery (<https://www.w3schools.com/jquery/>) untuk memahami dasar-dasar jQuery dan manipulasi DOM.
2. Validasi Form dengan jQuery (<https://www.w3schools.com/jquery/jquery_validation.asp>) untuk mempelajari cara melakukan validasi form menggunakan jQuery.

**DAFTAR PERTANYAAN**

1. Bagaimana kita bisa mengubah daftar buku yang saat ini disimpan di localStorage menjadi format JSON, dan apa keuntungan menggunakan JSON untuk menyimpan data buku tersebut?
2. Jika kita ingin mengintegrasikan sebuah search engine eksternal ke dalam website kita untuk mencari informasi tambahan tentang buku (seperti ulasan atau sinopsis), bagaimana JSON bisa memainkan peran dalam pertukaran data antara website kita dan search engine tersebut?
3. Apakah ada metode dalam JavaScript atau jQuery yang dapat digunakan untuk mengirim request ke search engine eksternal dan menerima response dalam format JSON, dan bagaimana metode tersebut dapat diintegrasikan ke dalam kode yang sudah ada?
4. Bagaimana implementasi sistem indeks pada mesin pencari dapat menginspirasi cara kita menyimpan dan mencari data buku di localStorage? Bisakah kita memanfaatkan prinsip indeksasi untuk meningkatkan efisiensi pencarian buku pada website tersebut?

**TEORI SINGKAT**

Search engine, atau mesin pencari, adalah sistem perangkat lunak yang dirancang untuk mencari informasi di World Wide Web. Informasi tersebut dapat berupa halaman web, gambar, video, atau jenis file lainnya. Mesin pencari bekerja dengan melakukan crawling, yaitu proses mengunjungi dan membaca halaman web untuk membuat indeks informasi. Kemudian, melalui proses yang disebut indexing, informasi tersebut diorganisir ke dalam database.

Cara Kerja:

Ketika pengguna memasukkan query pencarian, search engine akan mencari melalui indeksnya untuk menemukan informasi yang paling relevan dengan query tersebut. Algoritma khusus digunakan untuk menentukan urutan atau ranking hasil pencarian berdasarkan berbagai faktor, termasuk relevansi, popularitas, otoritas, dan penggunaan kata kunci.

Komponen Utama:

Crawler: Sebuah bot yang mengunjungi website secara sistematis untuk mengumpulkan data.

Indexer: Komponen yang mengatur dan menyimpan data yang dikumpulkan oleh crawler.

Algorithm: Aturan dan formula yang menentukan bagaimana data disajikan sebagai hasil pencarian.

JSON (JavaScript Object Notation) adalah format ringan untuk pertukaran data. JSON sangat mudah dibaca oleh manusia dan mesin. JSON banyak digunakan untuk pertukaran data antara client dan server dalam pengembangan web.

Struktur:

Object: Kumpulan pasangan nama/nilai yang tidak berurutan. Ditulis dalam kurung kurawal {}.

Array: Kumpulan nilai yang berurutan. Ditulis dalam kurung siku [].

Nilai: Dapat berupa string, angka, objek, array, true, false, atau null.

Menggunakan JSON Eksternal:

JSON eksternal sering digunakan dalam aplikasi web untuk memuat data dari server tanpa harus memuat ulang halaman. Ini memungkinkan aplikasi web menjadi lebih responsif dan interaktif.

Contoh Penerapan:

AJAX (Asynchronous JavaScript and XML): AJAX memungkinkan web pages untuk memuat data dari server secara asinkron melalui API. JSON digunakan sebagai format untuk menerima data dari server dan mengolahnya di client-side.

Web APIs: Banyak API web menyediakan data dalam format JSON. Ini mempermudah developer untuk mengintegrasikan fungsi seperti maps, media sosial, dan informasi cuaca ke dalam aplikasi mereka.

Dengan memahami konsep-konsep dasar dari search engine dan JSON eksternal, pengembang web dapat meningkatkan fungsionalitas dan user experience dari aplikasi web yang mereka kembangkan.

**ELEMEN KOMPETENSI**

**Deskripsi :** Formulir pendaftaran yang disimpan ke dalam json, kemudian ada fungsi search engine yang bisa mencari berdasarkan kolom masing masing table

**Kompetensi Dasar** : mampu menggunakan localStorage pada browser sebagai tempat penyimpanan data buku yang telah diinput dan mampu mengimplementasikan algoritma pencarian dinamis yang dapat menyesuaikan tampilan tabel berdasarkan kriteria pencarian yang diinput oleh pengguna.

**Latihan 1**

* 1. Copy code kadalam sublime atau visual code

<!DOCTYPE html>

<html>

<head>

<title>Pencarian dan Penyimpanan Data Buku</title>

<style>

table, th, td {

border: 1px solid black;

border-collapse: collapse;

}

th, td {

padding: 10px;

}

th {

text-align: left;

}

img {

width: 100px; /\* Atur lebar gambar \*/

}

</style>

</head>

<body>

<h2>Form Input Data Buku</h2>

<!-- Form untuk menambahkan buku baru -->

<form id="bookForm">

Nama Buku: <input type="text" id="bookName" name="bookName"><br><br>

Penulis: <input type="text" id="author" name="author"><br><br>

Tahun Terbit: <input type="number" id="yearPublished" name="yearPublished" min="1900" max="2099" placeholder="Tahun"><br><br>

Gambar Buku: <input type="file" id="bookImage" accept="image/\*"><br><br>

<input type="button" value="Tambah" onclick="addBook()">

</form>

<h2>Pencarian Buku</h2>

<!-- Input untuk mencari buku dalam tabel -->

Cari Buku: <input type="text" id="searchInput" onkeyup="searchBook()" placeholder="Cari berdasarkan tahun, nama, atau penulis">

<h2>Daftar Buku</h2>

<!-- Tabel untuk menampilkan buku -->

<table id="booksTable">

<tr>

<th>Nama Buku</th>

<th>Penulis</th>

<th>Tahun Terbit</th>

<th>Gambar</th>

</tr>

<!-- Data buku akan ditampilkan di sini -->

</table>

<script>

// Fungsi untuk mengkonversi gambar ke Base64

function convertImageToBase64(file, callback) {

const reader = new FileReader();

reader.onload = function(e) {

callback(e.target.result);

};

reader.readAsDataURL(file);

}

// Fungsi untuk menambahkan buku ke dalam 'localStorage' dan memperbarui tabel

function addBook() {

var name = document.getElementById("bookName").value;

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

var year = document.getElementById("yearPublished").value;

var imageFile = document.getElementById("bookImage").files[0];

if (name && author && year) {

if (imageFile) {

convertImageToBase64(imageFile, function(base64Image) {

saveBook(name, author, year, base64Image);

document.getElementById("bookForm").reset();

});

} else {

saveBook(name, author, year, '');

document.getElementById("bookForm").reset();

}

} else {

alert("Silakan lengkapi semua bidang formulir.");

}

}

// Fungsi untuk menyimpan buku ke dalam 'localStorage'

function saveBook(name, author, year, base64Image) {

let books = localStorage.getItem('books') ? JSON.parse(localStorage.getItem('books')) : [];

books.push({ name, author, year, image: base64Image });

localStorage.setItem('books', JSON.stringify(books));

loadBooks(); // Memperbarui tabel

}

// Fungsi untuk memuat buku dari 'localStorage' dan menampilkan ke dalam tabel

function loadBooks() {

let books = localStorage.getItem('books') ? JSON.parse(localStorage.getItem('books')) : [];

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

while (table.rows.length > 1) {

table.deleteRow(1);

}

books.forEach(book => {

var row = table.insertRow();

row.insertCell().innerHTML = book.name;

row.insertCell().innerHTML = book.author;

row.insertCell().innerHTML = book.year;

row.insertCell().innerHTML = book.image ? `<img src="${book.image}" alt="Gambar Buku">` : 'Tidak Ada Gambar';

});

}

function searchBook() {

var input, filter, table, tr, td, i, txtValue;

input = document.getElementById("searchInput");

filter = input.value.toUpperCase();

table = document.getElementById("booksTable");

tr = table.getElementsByTagName("tr");

// Loop through all table rows, and hide those who don't match the search query

for (i = 1; i < tr.length; i++) {

td = tr[i].getElementsByTagName("td");

var displayStyle = 'none'; // Assume the row will not match the search

// Loop through all cells in the row

for (var j = 0; j < td.length; j++) {

if (td[j]) {

txtValue = td[j].textContent || td[j].innerText;

if (txtValue.toUpperCase().indexOf(filter) > -1) {

displayStyle = ''; // The row matches the search, so it will be displayed

break;

}

}

}

tr[i].style.display = displayStyle;

}

}

// Inisialisasi data buku saat halaman web dimuat

window.onload = function() {

loadBooks();

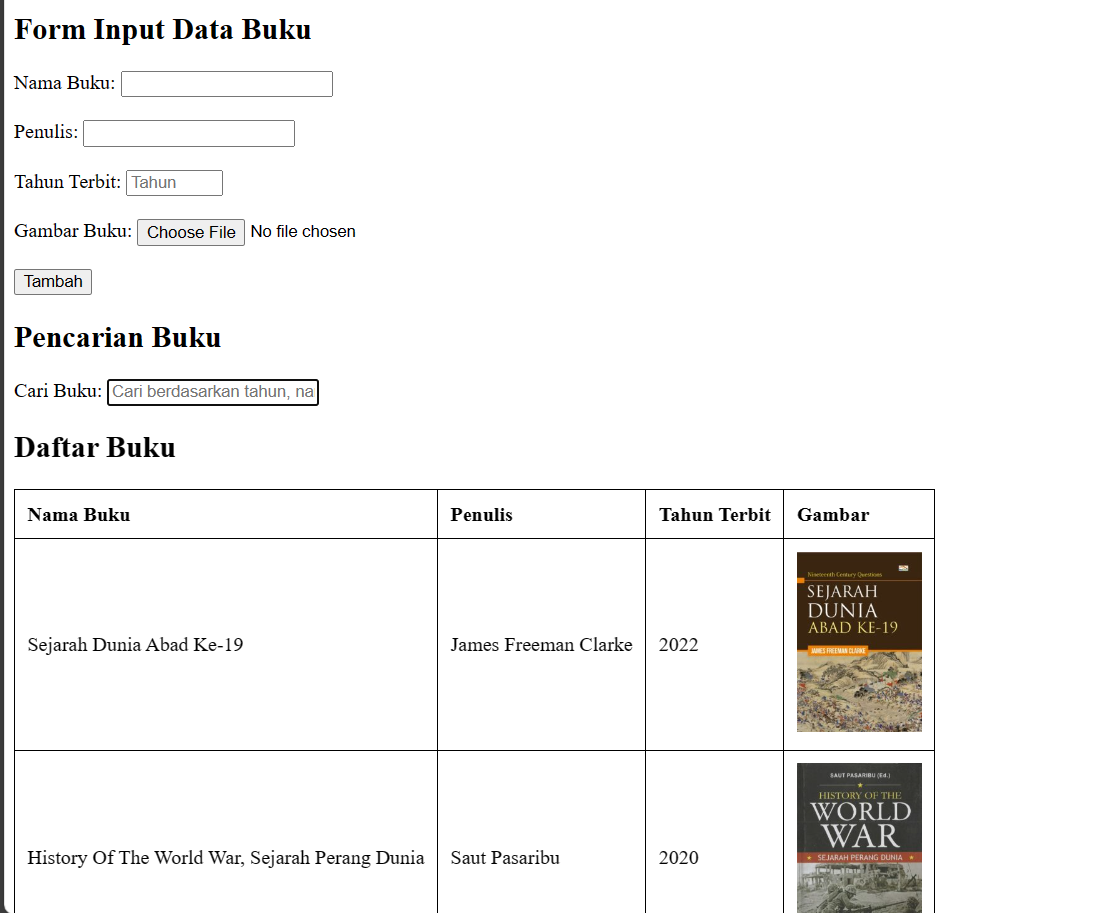
};

</script>

</body>

</html>

* 1. Hasilnya seperti di bawah ini jadi formnya tersimpan ke dalam json kemudian bisa search engine



* 1. Simpanlah ke dalams \*.html (contoh : web\_NAMA\_ANDA.html) pada sebuah folder.

**Latihan 2**

* 1. pembagian aplikasi perkelompok dengan data form masing – masing aplikasi yang sudah di tentukan dengan menambahkan fungsi jQuery dan CSS. Seperti latihan 1 memunculkan seperti output pada latihan 1 jadi bisa tersimpan dalam json dan ada fungsi search engine

**Kelompok 1 Aplikasi PM:**

Minimal merekam informasi:

* Nama
* Tujuan
* perkiraan tanggal mulai dan selesai
* PIC
* status penyelesaian Proyek berupa pilihan: selesai, berlangsung, gagal; serta deskripsi catatan akhir kegiatan. Sistem juga merekap waktu perekaman, dan waktu update informasi terakhir.

**Kelompok 2 Aplikasi IE:**

Minimal merekam informasi:

* Manfaat Proyek bagi Organisasi
* Leading Indicator
* User Outcomes dan Model Properties
* tanggal mulai dan selesai dan
* status penyelesaian perancangan berupa pilihan: selesai, berlangsung, gagal; serta deskripsi catatan akhir kegiatan. Sistem juga merekap waktu perekaman, dan waktu update informasi terakhir.

**Kelompok 3 Aplikasi IC:**

Minimal merekam informasi:

* deskripsi masalah
* deskripsi fitur
* deskripsi label
* kategori masalah berupa: Klasifikasi, prediksi numerik, prediksi probabilistik, Deteksi, Rekognisi;
* tanggal mulai dan selesai pemodelan
* dan status penyelesaian perancangan berupa pilihan: selesai, berlangsung, gagal; serta deskripsi catatan akhir kegiatan. Sistem juga merekap waktu perekaman, dan waktu update informasi terakhir.

**Kelompok 4 Aplikasi Dataset:**

Minimal merekam informasi:

* Nama dataset
* pembuat dataset
* verifikator dataset
* Jumlah Records
* Jumlah attribute data
* deskripsi sumber data
* tanggal mulai dan selesai dibuat
* status penyelesaian berupa pilihan: selesai, berlangsung, gagal; serta deskripsi catatan akhir kegiatan. Sistem juga merekap waktu perekaman, dan waktu update informasi terakhir.

**Kelompok 5 Aplikasi Implementasi:**

Minimal merekam informasi:

* Deskripsi perangkat keras yang diperlukan
* deskripsi para pihak yang terlibat
* deskripsi aktivitas yang akan dilaksanakan
* tanggal mulai dan selesai implementasi
* status penyelesaian implementasi berupa pilihan: selesai, berlangsung, gagal; serta deskripsi catatan akhir kegiatan. Sistem juga merekap waktu perekaman, dan waktu update informasi terakhir.

**KESIMPULAN**

**Berikan kesimpulan berdasarkan apa yang sudah di kerjakan pada modul ini minimal 200 kata**

**CEK LIST (√)**

1. Menerapkan validasi pada form input data dengan jQuery ( )
2. Menangani event submit form dengan jQuery untuk memvalidasi data sebelum pengiriman ( )

**FORM UMPAN BALIK**

|  |  |  |  |
| --- | --- | --- | --- |
| **Elemen Kompetensi** | **Tingkat Kesulitan**  **(1: sangat mudah,2: mudah,3: biasa,4: sulit,5: sangat sulit)** | **Tingkat Ketertarikan**  **(1: tidak tertarik, 2: cukup tertarik, 3: tertarik; 4: sangat tertarik)** | **Waktu Penyelesaian**  **Dalam menit** |
| **Menangani event submit form untuk memvalidasi data sebelum pengiriman** |  |  |  |
| **Menerapkan validasi pada form input data dengan jQuery** |  |  |  |