**LAPORAN TUGAS PRAKTIKUM DESIGN WEB**



Oleh:

Dendy Anugrahi Rabbi

4523210034

**Dosen:**

**Adi Wahyu Pribadi, S.Si., M.Kom**

**S1-Teknik Informatika**

**Fakultas Teknik Universitas Pancasila**

**2024/2025**

**INDEX.HTML**

<!DOCTYPE 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>Aplikasi Buku Favorit</title>

        <script src="https://cdn.tailwindcss.com"></script>

    </head>

<body class="italic m-5 bg-yellow-400">

    <h1 class="text-5xl font-bold text-center text-violet-700">

        Aplikasi Buku Favorit</h1>

        <!--input nama-->

    <div id="session-container" class="mb-7">

        <input type="text" id="namaPengguna" class="p-2 my-1.5 mx-0 w-full box-border " placeholder="Masukkan Nama Anda"/>

        <button class="italic px-4 bg-violet-700 cursor-pointer rounded hover:bg-red-500 text-white" id="btnSimpanNama">Simpan Nama</button>

        <h3 id="salamPengguna" class="text-black"></h3>

    </div>

    <!--form tambah buku-->

    <form id="form-tambah-buku" class="mb-5">

        <input type="text" id="judul" class="p-2 my-1.5 mx-0 w-full box-border" placeholder="Judul Buku" required />

        <input type="text" id="penulis" class="p-2 my-1.5 mx-0 w-full box-border" placeholder="Penulis" required />

        <input type="number" id="tahun" class="p-2 my-1.5 mx-0 w-full box-border" placeholder="Tahun Terbit" required />

        <button type="submit" class="italic px-4 bg-violet-700 cursor-pointer rounded hover:bg-red-500 text-white">Tambah Buku</button>

    </form>

    <!-- Daftar Buku -->

    <h2 class="text-2xl font-bold text-violet-700 ">Daftar Buku</h2>

    <div id="daftar-buku" class="p-5 border-2 outline-violet-500"></div>

    <!-- Buku Favorit -->

    <h2 class="text-2xl font-bold text-violet-700">Daftar Buku Favorit</h2>

    <div id="buku-favorit" class="p-5 border-2"></div>

    <script src="script.js"></script>

</body>

</html

**SCRIPT.JS**

// Definisi Class Buku

class Buku {

    constructor(judul, penulis, tahun) {

        this.judul = judul;

        this.penulis = penulis;

        this.tahun = tahun;

    }

    tampilkanInfo() {

        return `${this.judul} oleh ${this.penulis} (${this.tahun})`;

    }

}

// Array untuk menyimpan daftar buku dan buku favorit

let daftarBuku = [];

let bukuFavorit = [];

// Mengambil elemen DOM

const formTambahBuku = document.getElementById('form-tambah-buku');

const divDaftarBuku = document.getElementById('daftar-buku');

const divBukuFavorit = document.getElementById('buku-favorit');

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

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

// Event Listener untuk Form Tambah Buku

formTambahBuku.addEventListener('submit', function (e) {

    e.preventDefault();

    tambahBuku();

});

// Fungsi untuk menambahkan buku ke daftar

function tambahBuku() {

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

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

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

    // Validasi input

    if (judul === '' || penulis === '' || tahun === '') {

        alert('Semua kolom harus diisi!');

        return;

    }

    const bukuBaru = new Buku(judul, penulis, tahun);

    daftarBuku.push(bukuBaru);

    simpanDaftarBuku(); // Simpan daftar buku setelah menambahkan buku baru

    tampilkanDaftarBuku();

    formTambahBuku.reset();

}

// Fungsi untuk menyimpan daftar buku ke Local Storage

function simpanDaftarBuku() {

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

}

// Fungsi untuk menampilkan daftar buku

function tampilkanDaftarBuku() {

    divDaftarBuku.innerHTML = '';

    daftarBuku.forEach((buku, index) => {

        const divBuku = document.createElement('div');

        divBuku.classList.add('buku');

        divBuku.innerHTML = `

            <p>${buku.tampilkanInfo()}</p>

            <button class="py-2 px-4 bg-violet-700 cursor-pointer rounded hover:bg-red-500 text-white" onclick="tambahKeFavorit(${index})">Tambah ke Favorit</button>

            <button class="py-2 px-4 bg-red-200 cursor-pointer rounded hover:bg-red-500" onclick="hapusDariDaftarBuku(${index})">Hapus</button>

        `;

        divDaftarBuku.appendChild(divBuku);

    });

}

function hapusDariDaftarBuku(index) {

    daftarBuku.splice(index, 1);

    simpanDaftarBuku();

    tampilkanDaftarBuku();

    if (hapusDariDaftarBuku) {

        alert ('Buku Ini Akan di Hapus dari List DaftarBuku ?');

        return;

    }

}

// Fungsi untuk menambahkan buku ke favorit

function tambahKeFavorit(index) {

    const buku = daftarBuku[index];

    // Cek apakah buku sudah ada di favorit

    const sudahAda = bukuFavorit.some(favBuku => {

        return favBuku.judul === buku.judul &&

               favBuku.penulis === buku.penulis &&

               favBuku.tahun === buku.tahun;

    });

    if (sudahAda) {

        alert('Buku ini sudah ada di daftar favorit!');

        return;

    }

    bukuFavorit.push(buku);

    simpanBukuFavorit();

    tampilkanBukuFavorit();

}

// Fungsi untuk menyimpan buku favorit ke Local Storage

function simpanBukuFavorit() {

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

}

// Fungsi untuk menampilkan buku favorit

function tampilkanBukuFavorit() {

    divBukuFavorit.innerHTML = '';

    bukuFavorit.forEach((buku, index) => {

        const divBuku = document.createElement('div');

        divBuku.classList.add('buku');

        divBuku.innerHTML = `

            <p>${buku.tampilkanInfo()}</p>

            <button class="py-2 px-4 bg-red-200 cursor-pointer rounded hover:bg-red-500 " onclick="hapusDariFavorit(${index})">Hapus</button>

        `;

        divBukuFavorit.appendChild(divBuku);

    });

}

// Fungsi untuk menghapus buku dari favorit

function hapusDariFavorit(index) {

    bukuFavorit.splice(index, 1);

    simpanBukuFavorit();

    tampilkanBukuFavorit();

    if (hapusDariFavorit) {

        alert ('Buku Ini Akan di Hapus dari List Favorit ?');

        return;

    }

}

// Event Listener untuk tombol simpan nama pengguna

btnSimpanNama.addEventListener('click', function () {

    const nama = document.getElementById('namaPengguna').value;

    if (nama === '') {

        alert('Masukkan nama Anda!');

        return;

    }

    sessionStorage.setItem('namaPengguna', nama);

    tampilkanNamaPengguna();

    document.getElementById('namaPengguna').value = '';

});

// Fungsi untuk menampilkan nama pengguna

function tampilkanNamaPengguna() {

    const nama = sessionStorage.getItem('namaPengguna');

    if (nama) {

        salamPengguna.textContent = `Selamat datang, ${nama}!`;

    } else {

        salamPengguna.textContent = '';

    }

}

// Memuat data saat halaman dimuat

window.onload = function () {

    // Memuat daftar buku dari Local Storage

    if (localStorage.getItem('daftarBuku')) {

        const storedBooks = JSON.parse(localStorage.getItem('daftarBuku'));

        // Rekonstruksi objek buku menjadi instance dari kelas Buku

        daftarBuku = storedBooks.map(book => {

            return new Buku(book.judul, book.penulis, book.tahun);

        });

        tampilkanDaftarBuku();

    }

    // Memuat buku favorit dari Local Storage

    if (localStorage.getItem('bukuFavorit')) {

        const storedFavorites = JSON.parse(localStorage.getItem('bukuFavorit'));

        // Rekonstruksi objek buku menjadi instance dari kelas Buku

        bukuFavorit = storedFavorites.map(book => {

            return new Buku(book.judul, book.penulis, book.tahun);

        });

        tampilkanBukuFavorit();

    }

    // Menampilkan nama pengguna dari Session Storage

    tampilkanNamaPengguna();

};

**HASIL**

****

