

**UNIVERSITAS GUNADARMA PRAKTIKUM
PEMROGRAMAN WEB**



MANUAL BOOK

**“Perancangan CRUD Dengan Menggunakan Bahasa HTML CSS Bootstrap v5 dan
JavaScript Local Storage Berbasis Website”**

Nama Anggota :

1. Rizki Eka Fachrudin (51421340)

Kelas : 3IA07

Fakultas : Teknologi Industri

Jurusan : Informatika

Ketua Asisten : Ryan Arya

Nama Asisten :

1. AprilYandi Dwi
2. Qonita Abdah

**Ditulis Guna Melengkapi Sebagian Syarat
Praktikum Pemrograman WEB Jenjang S1 Universitas
Gunadarma
2023**

BAB I

PENDAHULUAN

1.1 Latar Belakang

Dalam dunia pengembangan web, aplikasi CRUD (Create, Read, Update, Delete) adalah salah satu dari fondasi dasar dalam membangun interaksi antara pengguna dan data. Dalam banyak kasus, penggunaan teknologi web seperti HTML, CSS, JavaScript, dan kerangka kerja seperti Bootstrap v5 sangatlah umum.

Tujuan utama dari aplikasi CRUD adalah untuk mengelola data, baik itu data pengguna, entri blog, produk dalam sebuah toko daring, dan sebagainya. Melalui proses CRUD, pengguna dapat membuat data baru, membaca data yang ada, memperbarui data yang ada, dan menghapus data yang tidak diperlukan.

Penggunaan Local Storage sebagai tempat penyimpanan sementara di sisi klien (client-side) memungkinkan aplikasi web untuk menyimpan data dalam perangkat pengguna, memungkinkan akses cepat dan perubahan data tanpa koneksi internet.

1.2 Tujuan

Melalui manual ini, kami bertujuan untuk memberikan panduan langkah demi langkah bagi pemula atau mereka yang ingin memperluas pemahaman mereka tentang bagaimana membuat aplikasi CRUD dengan menggunakan teknologi yang umum digunakan dalam pengembangan web saat ini. Diharapkan dengan panduan ini, pembaca dapat memahami konsep dasar dalam perancangan dan implementasi aplikasi CRUD berbasis web dengan HTML, CSS, Bootstrap v5, dan JavaScript dengan menggunakan Local Storage sebagai penyimpanan sementara data.

Dengan demikian, manual ini dapat menjadi landasan bagi pengembang web pemula untuk memulai perjalanan mereka dalam membangun aplikasi web interaktif dengan kemampuan dasar CRUD serta memperluas pemahaman mereka tentang konsep pengembangan web secara keseluruhan.

BAB II

PEMBAHASAN

2.1 HTML

HTML (Hypertext Markup Language) adalah bahasa markup yang digunakan untuk membuat struktur dasar sebuah halaman web. Ini merupakan fondasi dari hampir semua halaman web di internet. Berikut adalah beberapa poin penting tentang HTML:

2.1.1 Struktur Dasar

- HTML menggunakan serangkaian tag atau elemen untuk menandai bagian-bagian dari konten dalam sebuah halaman web.
- Tag-tag ini membentuk struktur dasar halaman, seperti `<html>`, `<head>`, `<body>`, `<header>`, `<footer>`, `<section>`, `<div>`, dan lain-lain.

2.1.2 Fungsi Utama

- Menyediakan kerangka kerja untuk menampilkan konten di browser.
- Mengatur tata letak dan struktur dokumen dengan elemen seperti judul, paragraf, gambar, link, tabel, dan formulir.

2.1.3. Kekurangan HTML

- HTML hanya dapat digunakan untuk membuat halaman web statis, sehingga untuk fitur dinamis memerlukan bahasa pemrograman tambahan seperti PHP atau JavaScript
- Tidak memungkinkan user atau pengguna untuk menjalankan logic
- Fitur baru tidak dapat dijalankan secara cepat di sebagian browser

2.1.4. Kelebihan HTML

- Bersifat open-source dan gratis
- Memiliki Bahasa Markup yang rapi dan konsisten
- Memiliki banyak sumber dan komunitas yang besar
- Mudah diintegrasikan dengan bahasa backend seperti PHP, Node.js, dan JavaScript

- Dapat dijalankan oleh semua browser
- Dapat dipelajari dengan mudah oleh pengembang web pemula
- Menampilkan gambar, tabel, video, dan objek lainnya
- Menandai elemen dan membuat form dengan mudah

2.2 JavaScript

JavaScript adalah bahasa pemrograman yang digunakan untuk membuat halaman web yang lebih dinamis dan interaktif. JavaScript adalah bahasa scripting yang termasuk ke dalam bahasa pemrograman, tetapi digunakan untuk memanipulasi, menyesuaikan, dan mengotomatiskan apa yang sudah ada di sistem. JavaScript umumnya digunakan pada dokumen HTML dan dapat dijalankan oleh semua browser. JavaScript juga fleksibel dan dapat digunakan untuk sisi server dengan adanya Node.js. Fungsi JavaScript yang paling umum adalah untuk pengembangan aplikasi web dan mobile, membangun web server dan aplikasi server, membuat website yang interaktif, serta game development. JavaScript memiliki banyak framework dan library yang bisa membantu seperti AngularJS, jQuery, dan ReactJS. Meskipun JavaScript mudah dipelajari oleh pemula, namun JavaScript hanya dapat digunakan untuk membuat halaman web statis, sehingga untuk fitur dinamis memerlukan bahasa pemrograman tambahan seperti PHP atau JavaScript

2.2.1 Kelebihan JavaScript:

- Dapat bekerja pada banyak platform, seperti web browser, aplikasi server, sistem operasi, dll.
- Dapat memanipulasi konten halaman web secara langsung, sehingga membuat halaman web menjadi lebih interaktif dan responsif.
- Fleksibel dan dapat digunakan untuk sisi server dengan adanya Node.js.
- Mudah dipelajari oleh pemula.
- Memiliki banyak sumber dan komunitas yang besar.
- Menawarkan berbagai keunggulan, seperti mudah dipelajari, ringkas, dan dapat digunakan untuk berbagai jenis aplikasi.

2.2.2 Kekurangan JavaScript:

- Tidak dapat berjalan sempurna tanpa dukungan browser.

- Memiliki kelemahan dalam hal keamanan, seperti cross-site scripting (XSS) dan cross-site request forgery (CSRF).
- Memiliki keterbatasan dalam hal pengolahan data yang besar.
- Memerlukan perhatian khusus dalam hal performa dan optimasi.

2.3 Local Storage

Local Storage adalah media penyimpanan yang digunakan dalam pengembangan web untuk menyimpan data secara lokal di browser pengguna. Berbeda dengan cloud storage yang memanfaatkan koneksi internet, local storage tidak memerlukan koneksi internet untuk menyimpan dan mengakses data. Beberapa poin penting tentang local storage:

2.3.1 Bentuk Fisik

Local storage adalah media penyimpanan yang memiliki bentuk fisik, seperti hard disk, flash disk, memory card, compact disk, dan memori internal pada perangkat lainnya.

2.3.2 Kapasitas Terbatas

Local storage biasanya memiliki kapasitas penyimpanan yang lebih terbatas jika dibandingkan dengan cloud storage.

2.3.3 Keamanan

Data yang disimpan di local storage dapat hilang atau dicuri jika perangkat fisiknya hilang atau dicuri. Namun, local storage tidak memerlukan autentikasi dua faktor seperti cloud storage.

2.3.4 Penggunaan dalam Pengembangan Web

Dalam pengembangan web, local storage dapat digunakan untuk menyimpan data secara lokal di browser pengguna, seperti preferensi pengguna, riwayat, atau data aplikasi.

Dalam pengembangan web, local storage sering digunakan untuk menyimpan data yang diperlukan oleh aplikasi web secara lokal di browser pengguna tanpa perlu terhubung ke server. Meskipun memiliki keterbatasan dalam kapasitas penyimpanan dan keamanan, local storage tetap menjadi pilihan yang berguna dalam pengembangan web untuk keperluan-keperluan tertentu.

2.4 CSS

CSS (Cascading Style Sheets) adalah bahasa yang digunakan untuk mengendalikan tata letak (layout) dan tampilan (styling) dari elemen-elemen pada halaman web. Berikut adalah penjelasan lengkap tentang CSS:

CSS (Cascading Style Sheets)

CSS adalah bahasa pemrograman yang digunakan untuk mengendalikan tampilan dari halaman web yang ditulis dalam bahasa HTML. CSS memungkinkan pengembang web untuk mengatur tata letak, warna, ukuran, dan gaya teks, serta berbagai aspek tampilan lainnya dari elemen-elemen pada halaman web.

2.4.1 Kelebihan CSS

- Memisahkan tata letak dan tampilan dari struktur konten, sehingga memudahkan dalam pemeliharaan dan pengembangan halaman web.
- Memungkinkan untuk membuat tampilan yang konsisten di seluruh halaman web dengan menggunakan aturan gaya yang sama.
- Dapat digunakan untuk meresponsifkan tampilan halaman web terhadap berbagai ukuran layar, seperti pada perangkat mobile dan desktop.
- Memiliki berbagai fitur seperti animasi, transisi, dan transformasi yang memungkinkan untuk menciptakan tampilan yang menarik.

2.4.2 Kekurangan CSS

- Kompatibilitas: Beberapa fitur CSS mungkin tidak didukung oleh semua browser, sehingga memerlukan penanganan khusus untuk memastikan tampilan yang konsisten di berbagai browser.
- Keterbatasan dalam tata letak: Meskipun CSS memiliki fitur tata letak yang cukup lengkap, namun masih terdapat keterbatasan dalam hal tata letak yang kompleks.

Dengan menggunakan CSS, pengembang web dapat menciptakan tampilan yang menarik dan responsif untuk halaman web mereka. CSS juga memungkinkan untuk memisahkan tata letak dan tampilan dari struktur konten, sehingga memudahkan dalam pemeliharaan dan pengembangan halaman web.

2.5 Bootstrap v5

Bootstrap v5 adalah versi terbaru dari salah satu framework front-end terkemuka yang digunakan untuk mempercepat dan menyederhanakan proses pengembangan website. Framework ini berbasis HTML, CSS, dan JavaScript, dan dirancang untuk memungkinkan pengembang web untuk membuat tampilan yang responsif dan menarik dengan mudah. Berikut adalah penjelasan lengkap tentang Bootstrap v5 berdasarkan sumber yang diberikan

2.5.1 Pengertian Bootstrap v5

Bootstrap v5 adalah versi terbaru dari framework Bootstrap yang resmi dirilis pada tanggal 5 Mei 2021. Framework ini menyediakan berbagai komponen siap pakai, seperti sistem grid, typography, form, button, navbar, card, modal, dan lain-lain, yang memungkinkan pengembang web untuk membangun tampilan website dengan cepat dan mudah

2.5.2 Fitur Bootstrap v5

- **Tidak menggunakan jQuery:**
Bootstrap v5 beralih ke Vanilla JS dan tidak lagi menggunakan jQuery, yang membuat ukuran file CSS dan JavaScript menjadi lebih kecil.
- **Sistem grid yang lebih ringkas:**
Bootstrap v5 menggunakan sistem grid yang lebih ringkas, sehingga menghasilkan ukuran file yang lebih kecil.
- **Fleksibilitas dalam warna dan typography:**
Bootstrap v5 lebih fleksibel dalam hal warna dan typography, sehingga pengembang web dapat lebih leluasa dalam mendesain website mereka.
- **Optimasi performa:**
Bootstrap v5 dirancang untuk bekerja lebih cepat dan lebih efisien, sehingga website yang dibangun dengan framework ini dapat terasa lebih responsif dan memiliki waktu loading yang lebih cepat.

2.5.3 Keunggulan Bootstrap v5

- Mempercepat proses pengembangan website dengan menyediakan komponen-komponen siap pakai.
- Ukuran file CSS dan JavaScript yang lebih kecil.

- Fleksibilitas dalam warna dan typography.
- Optimasi performa untuk website yang lebih responsif dan cepat.

2.5.4 Cara Menggunakan Bootstrap v5

Untuk menggunakan Bootstrap v5, pengembang web perlu melakukan import CSS dan JS Bootstrap ke file HTML di proyek mereka. Mereka juga dapat menggunakan Bootstrap secara online dengan memasukkan lokasi file bootstrap.min.css dan bootstrap.min.js ke dalam file HTML proyek mereka.

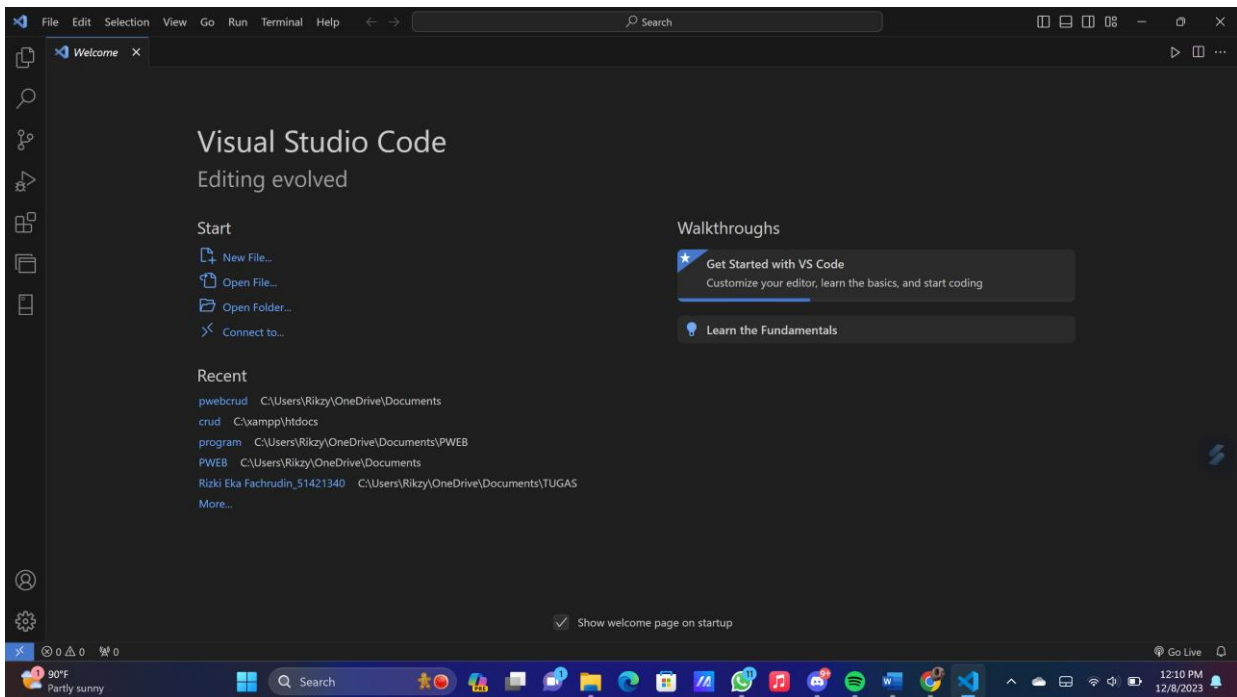
Dengan fitur-fitur terbaru dan keunggulan-keunggulannya, Bootstrap v5 menjadi pilihan yang menarik untuk mempercepat dan menyederhanakan proses pengembangan website. Dengan ukuran file yang lebih kecil, fleksibilitas dalam desain, dan optimasi performa, framework ini dapat membantu pengembang web dalam menciptakan tampilan website yang responsif, menarik, dan efisien.



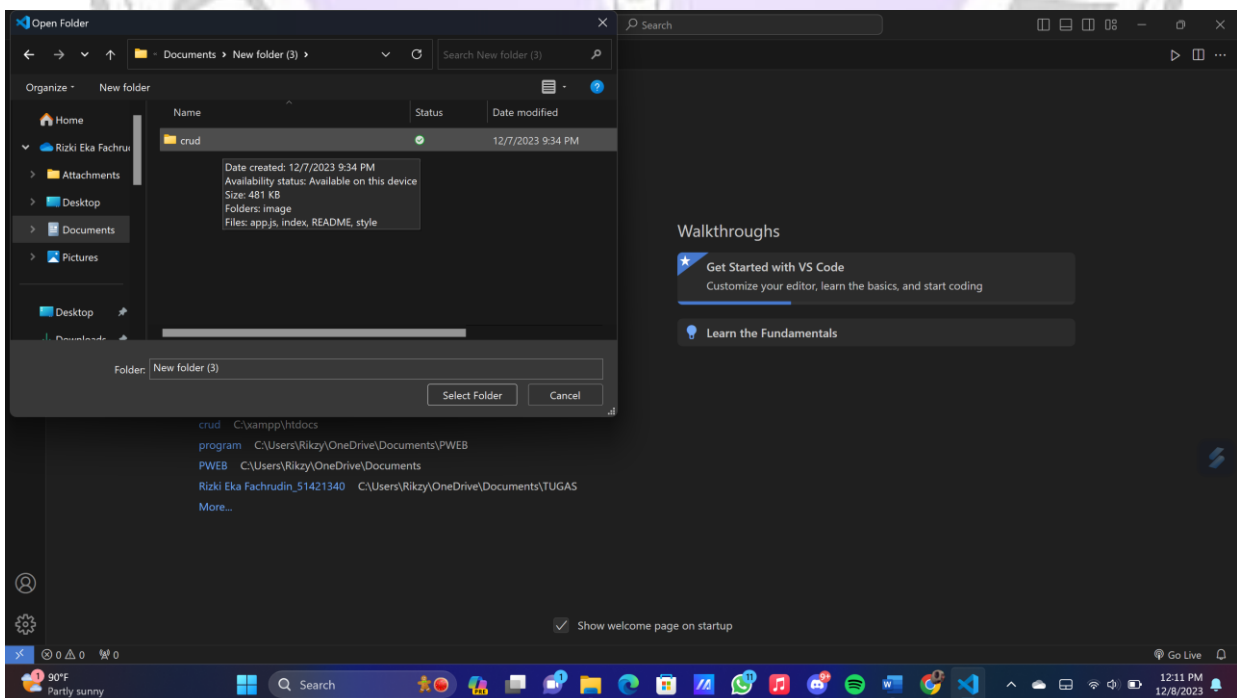
BAB III

ANALISA DAN PERANCANGAN

Projek yang dibuat yaitu, perancangan crud dengan menggunakan HTML, CSS, Bootstrap V5 dan JavaScript Local Storage berbasis website dengan memanfaatkan visual studio code sebagai sarana pengaplikasiannya. Di bawah ini merupakan tampilan awal visual studio code :



Kemudian membuka folder dari project yang telah kami buat



Lalu akan terbuka file index.html, main.js, dan style.css yang telah dibuat

Source Code Program:

```
1 <!doctype html>
2 <html lang="en">
3   <head>
4     <!-- Required meta tags -->
5     <meta charset="utf-8">
6     <meta name="viewport" content="width=device-width, initial-scale=1">
7
8     <!-- Bootstrap CSS -->
9     <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-EVSTQN3/...
```

```
38 <th>Age</th>
39 <th>City</th>
40 <th>Email</th>
41 <th>Phone</th>
42 <th>Post</th>
43 <th>Start Date</th>
44 <th>Action</th>
45 </tr>
46 </thead>
47
48 <tbody id="data"></tbody>
49 </table>
50 </div>
51 </section>
52
53 <!-- Modal Form -->
54 <div class="modal fade" id="userForm">
55   <div class="modal-dialog modal-dialog-centered modal-lg">
56     <div class="modal-content">
57       <div class="modal-header">
58         <h4 class="modal-title">Fill the Form</h4>
59         <button type="button" class="btn-close" data-bs-dismiss="modal" aria-label="Close"></button>
60       </div>
61       <div class="modal-body">
62         <form action="#" id="myForm">
63           <div class="card imgholder">
```

```
File Edit Selection View Go Run Terminal Help < -> New folder (3)
EXPLORER
NEW FO...
crud
index.html
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
</label>

</div>
<div class="inputField">
  <div>
    <label for="name">Name:</label>
    <input type="text" name="" id="name" required>
  </div>
  <div>
    <label for="age">Age:</label>
    <input type="number" name="" id="age" required>
  </div>
  <div>
    <label for="city">City:</label>
    <input type="text" name="" id="city" required>
  </div>
  <div>
    <label for="email">E-mail:</label>
    <input type="email" name="" id="email" required>
  </div>
  <div>
    <label for="phone">Number:</label>
    <input type="text" name="" id="phone" minlength="11" maxlength="11" required>
  </div>
  <div>
    <label for="post">Post:</label>
    <input type="text" name="" id="post" required>
  </div>
  <div>
    <label for="sDate">Start Date:</label>
    <input type="date" name="" id="sDate" required>
  </div>
</div>
```

```
File Edit Selection View Go Run Terminal Help < -> New folder (3)
EXPLORER
NEW FO...
crud
index.html
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
</div>
</form>
</div>
<div class="modal-footer">
  <button type="button" class="btn btn-secondary" data-bs-dismiss="modal">Close</button>
  <button type="submit" form="myForm" class="btn btn-primary submit">Submit</button>
</div>
</div>
</div>
<!--Read Data Modal-->
<div class="modal fade" id="readData">
  <div class="modal-dialog modal-dialog-centered modal-lg">
    <div class="modal-content">
      <div class="modal-header">
        <h4 class="modal-title">Profile</h4>
        <button type="button" class="btn-close" data-bs-dismiss="modal" aria-label="Close"></button>
      </div>
      <div class="modal-body">
        <form action="#" id="myForm">
          <div class="card imgholder">
            
          </div>
          <div class="inputField">
            <div>
              <label for="name">Name:</label>
              <input type="text" name="" id="name" required>
            </div>
            <div>
              <label for="age">Age:</label>
              <input type="number" name="" id="age" required>
            </div>
            <div>
              <label for="city">City:</label>
              <input type="text" name="" id="city" required>
            </div>
            <div>
              <label for="email">E-mail:</label>
              <input type="email" name="" id="email" required>
            </div>
            <div>
              <label for="phone">Number:</label>
              <input type="text" name="" id="phone" minlength="11" maxlength="11" required>
            </div>
            <div>
              <label for="post">Post:</label>
              <input type="text" name="" id="post" required>
            </div>
            <div>
              <label for="sDate">Start Date:</label>
              <input type="date" name="" id="sDate" required>
            </div>
          </div>
        </form>
      </div>
    </div>
  </div>
</div>
```

```
File Edit Selection View Go Run Terminal Help < -> New folder (3)

EXPLORER
NEW FO...
crud
index.html > ...

142 <input type="text" name="" id="showName" disabled>
143 </div>
144 <div>
145 <label for="age">Age:</label>
146 <input type="number" name="" id="showAge" disabled>
147 </div>
148 <div>
149 <label for="city">City:</label>
150 <input type="text" name="" id="showCity" disabled>
151 </div>
152 <div>
153 <label for="email">E-mail:</label>
154 <input type="email" name="" id="showEmail" disabled>
155 </div>
156 <div>
157 <label for="phone">Number:</label>
158 <input type="text" name="" id="showPhone" minlength="11" maxlength="11" disabled>
159 </div>
160 <div>
161 <label for="post">Post:</label>
162 <input type="text" name="" id="showPost" disabled>
163 </div>
164 <div>
165 <label for="sDate">Start Date:</label>
166 <input type="date" name="" id="showsDate" disabled>
167 </div>
168 </div>
169 </form>
170 </div>
171 </div>
172 </div>
173 </div>
174 </div>
175 </div>

OUTLINE
TIMELINE

Ln 1, Col 1 Spaces: 4 UTF-8 LF HTML Go Live Prettier
```

```
File Edit Selection View Go Run Terminal Help < -> New folder (3)

EXPLORER
NEW FO...
crud
index.html > ...

150 <input type="text" name="" id="showCity" disabled>
151 </div>
152 <div>
153 <label for="email">E-mail:</label>
154 <input type="email" name="" id="showEmail" disabled>
155 </div>
156 <div>
157 <label for="phone">Number:</label>
158 <input type="text" name="" id="showPhone" minlength="11" maxlength="11" disabled>
159 </div>
160 <div>
161 <label for="post">Post:</label>
162 <input type="text" name="" id="showPost" disabled>
163 </div>
164 <div>
165 <label for="sDate">Start Date:</label>
166 <input type="date" name="" id="showsDate" disabled>
167 </div>
168 </div>
169 </form>
170 </div>
171 </div>
172 </div>
173 </div>
174 </div>
175 </div>
176 </div>
177 <!-- Option 1: Bootstrap Bundle with Popper -->
178 <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.bundle.min.js" integrity="sha384-MrcN6ZMFY1zcL8N1+N"
179 </script>
180 <script src="app.js"></script>
181 </body>
182 </html>

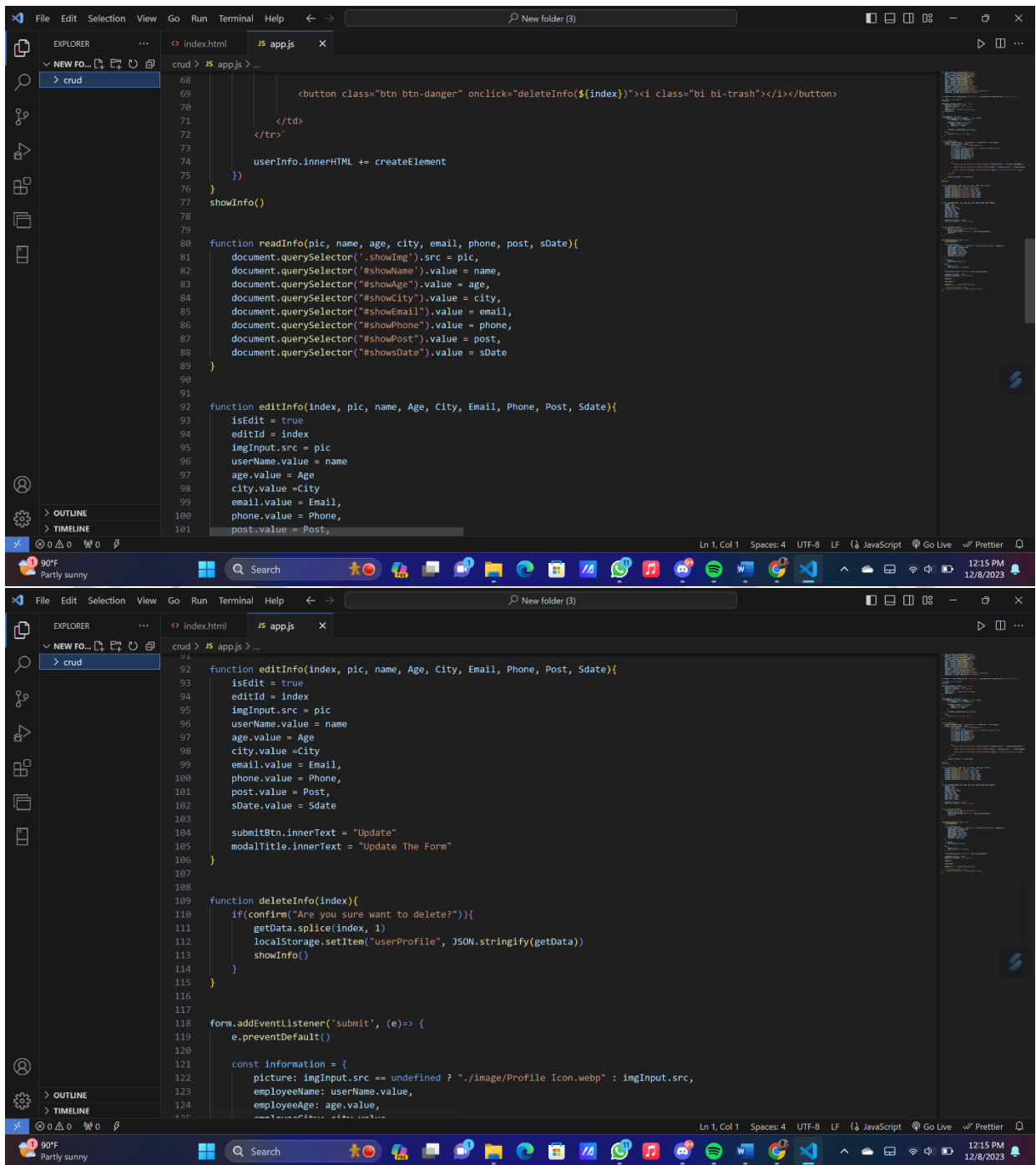
OUTLINE
TIMELINE

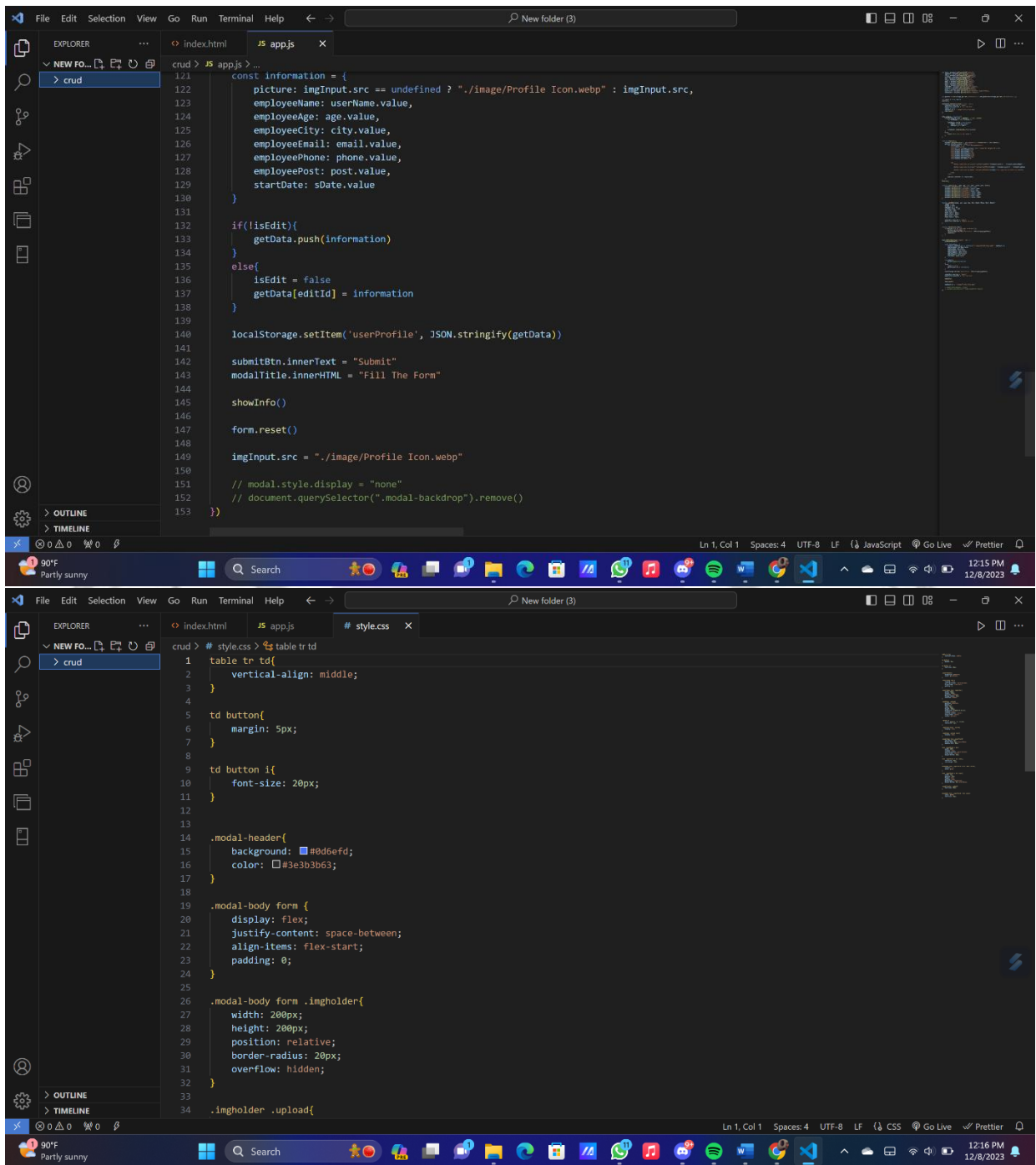
Ln 1, Col 1 Spaces: 4 UTF-8 LF HTML Go Live Prettier
```

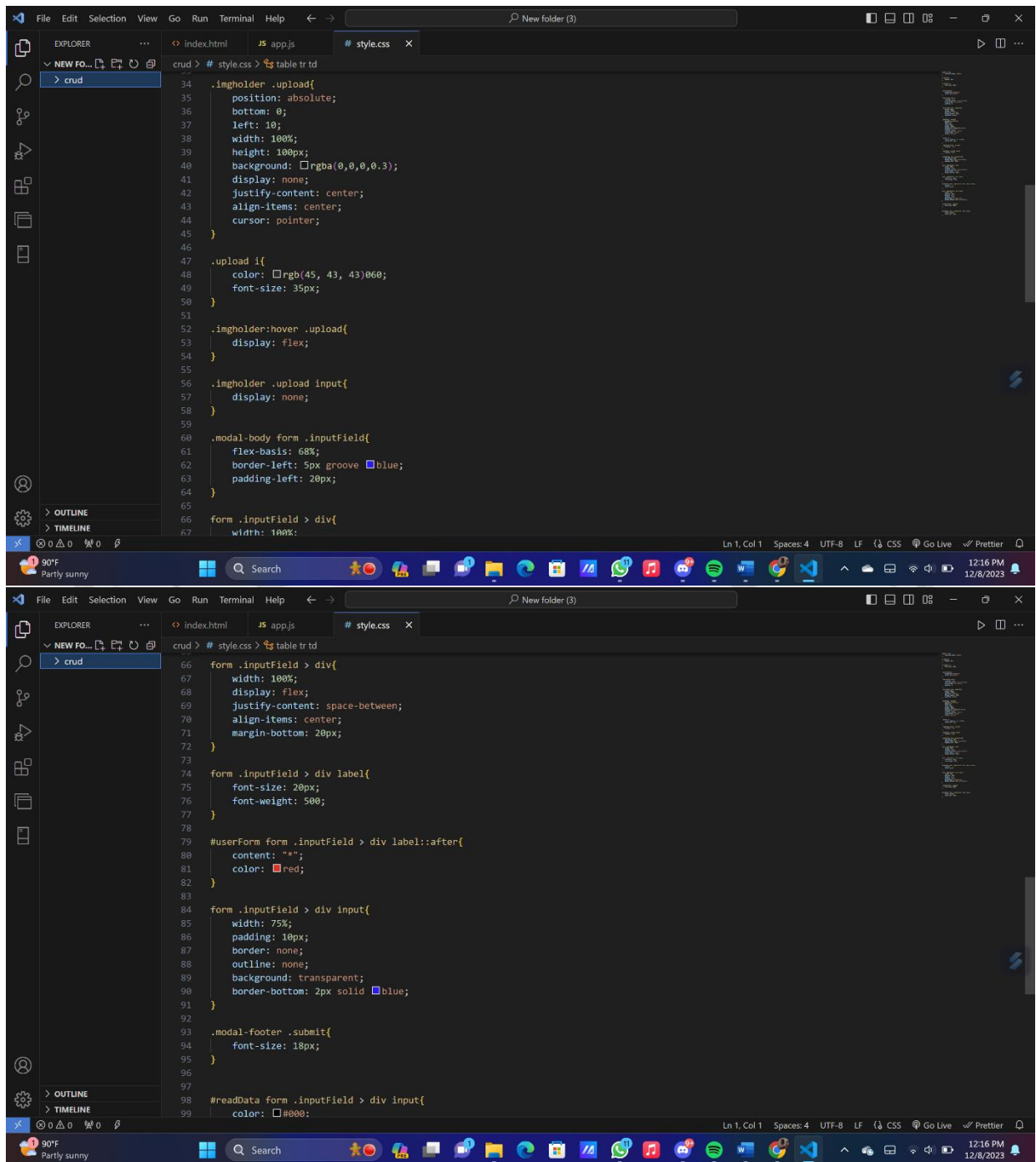


```
File Edit Selection View Go Run Terminal Help < -> New folder (3)
EXPLORER
NEW FO...
crud
index.html JS appjs X
crud > JS appjs > ...
1 var form = document.getElementById("myForm"),
2   imgInput = document.querySelector(".img"),
3   file = document.getElementById("imgInput"),
4   userName = document.getElementById("name"),
5   age = document.getElementById("age"),
6   city = document.getElementById("city"),
7   email = document.getElementById("email"),
8   phone = document.getElementById("phone"),
9   post = document.getElementById("post"),
10  sDate = document.getElementById("sDate"),
11  submitBtn = document.querySelector(".submit"),
12  userInfo = document.getElementById("data"),
13  modal = document.getElementById("userForm"),
14  modalTitle = document.querySelector("#userForm .modal-title"),
15  newUserBtn = document.querySelector(".newUser")
16
17
18 let getData = localStorage.getItem('userProfile') ? JSON.parse(localStorage.getItem('userProfile')) : []
19
20 let isEdit = false, editId
21 showInfo()
22
23 newUserBtn.addEventListener('click', () => {
24   submitBtn.innerText = 'Submit',
25   modalTitle.innerText = "Fill the Form"
26   isEdit = false
27   imgInput.src = "../image/Profile Icon.webp"
28   form.reset()
29 })
30
31
32 file.onChange = function(){
33   if(file.files[0].size < 1000000){ // 1MB = 1000000
34     var fileReader = new FileReader();
```

```
Ln 1, Col 1 Spaces: 4 UTF-8 LF JavaScript Go Live Prettier
30°F Partly sunny
Search
File Edit Selection View Go Run Terminal Help < -> New folder (3)
EXPLORER
NEW FO...
crud
index.html JS appjs X
crud > JS appjs > ...
35
36   fileReader.onload = function(e){
37     imgUrl = e.target.result
38     imgInput.src = imgUrl
39   }
40
41   fileReader.readAsDataURL(file.files[0])
42 }
43 else{
44   alert("This file is too large!")
45 }
46
47
48
49 function showInfo(){
50   document.querySelectorAll('.employeeDetails').forEach(info => info.remove())
51   getData.forEach((element, index) => {
52     let createElement = `<tr class="employeeDetails">
53       <td>${index+1}</td>
54       <td></td>
55       <td>${element.employeeName}</td>
56       <td>${element.employeeAge}</td>
57       <td>${element.employeeCity}</td>
58       <td>${element.employeeEmail}</td>
59       <td>${element.employeePhone}</td>
60       <td>${element.employeePost}</td>
61       <td>${element.startDate}</td>
62     </tr>
63
64     <td>
65       <button class="btn btn-success" onclick="readInfo('${element.picture}', '${element.employeeName}', '${element.employeeAge}', '${element.employeeCity}', '${element.employeeEmail}', '${element.employeePhone}', '${element.employeePost}', '${element.startDate}')">Read Info
66     </td>
67     <td>
68       <button class="btn btn-primary" onclick="editInfo(${index}, '${element.picture}', '${element.employeeName}', '${element.employeeAge}', '${element.employeeCity}', '${element.employeeEmail}', '${element.employeePhone}', '${element.employeePost}', '${element.startDate}')">Edit Info
69     </td>
70   `;
71   document.querySelector('.employeeDetails').insertAdjacentHTML('beforeend', createElement);
72 }
```







Penjelasan Source Code Program Index.html:

- `<!doctype html>`: Mendefinisikan tipe dokumen sebagai HTML5.
- `<html lang="en">`: Menandai awal dari elemen HTML dan menentukan bahasa untuk halaman (dalam hal ini, bahasa Inggris).
- `<head>`: Bagian kepala dokumen yang berisi metadata, referensi ke file eksternal (CSS dan JavaScript), dan informasi lainnya yang diperlukan untuk halaman web.

- `<meta charset="utf-8">`: Menentukan set karakter yang digunakan dalam dokumen (UTF-8).
- `<meta name="viewport" content="width=device-width, initial-scale=1">`: Memberikan instruksi kepada browser tentang tata letak dan skala halaman pada perangkat mobile.
- `<link>`: Menghubungkan dokumen ke berkas-berkas eksternal, seperti Bootstrap CSS dan ikon Bootstrap, yang diunduh dari CDN (Content Delivery Network) untuk memastikan penggunaan versi terbaru dan dapat diakses secara cepat.
- `<title>CRUD Operations</title>`: Menetapkan judul halaman.
- `<link rel="stylesheet" href="style.css">`: Menghubungkan dokumen ke berkas CSS kustom (jika ada) untuk gaya tambahan atau penyesuaian tertentu pada halaman.
- `<body>`: Bagian utama dari dokumen HTML yang berisi konten yang akan ditampilkan kepada pengguna.
- `<script>`: Menghubungkan dokumen ke berkas JavaScript, termasuk Bootstrap Bundle, untuk menambahkan fungsionalitas interaktif atau pengaturan lainnya pada halaman.

Penjelasan Source Code Program app.js:

- `form, imgInput, file, userName, age, city, email, phone, post, sDate, submitBtn, userInfo, modal, modalTitle, newUserBtn`: Variabel ini digunakan untuk menyimpan referensi ke elemen-elemen HTML yang dibutuhkan dalam manipulasi atau pengambilan nilai.
- `getData`: Variabel ini menggunakan `localStorage` untuk mengambil data yang disimpan sebelumnya dalam bentuk array. Jika data tidak ada, variabel ini akan diinisialisasi sebagai array kosong.
- `isEdit` dan `editId`: Variabel ini digunakan untuk mengelola status apakah pengguna sedang melakukan mode edit pada data tertentu.

➤ Fungsi:

- `newUserBtn.addEventListener('click', ()=> {...})`: Menetapkan perilaku saat tombol "New User" diklik, mengatur beberapa elemen ke kondisi awal.

- `file.onChange = function(){...}`: Mendeteksi perubahan pada file input (`file.onChange`) untuk menampilkan gambar yang dipilih ke dalam `` jika ukuran file kurang dari 1MB.
- `showInfo()`: Menampilkan informasi dari `getData` ke dalam tabel.
- `readInfo()`: Menampilkan informasi dari tabel ke dalam modal pembacaan data.
- `editInfo()`: Mengisi data dari tabel ke dalam form edit.
- `deleteInfo()`: Menghapus informasi dari `getData` dan `LocalStorage` sesuai dengan indeks yang dipilih.
- `form.addEventListener('submit', (e)=> {...})`: Menangani aksi saat form disubmit, baik untuk menambahkan data baru maupun memperbarui data yang sudah ada.

Penjelasan Source Code Program `style.css`:

1. `table tr td { vertical-align: middle; }`

Ini adalah aturan CSS untuk mengatur bagaimana sel-sel dalam tabel akan diatur secara vertikal. Pada kasus ini, `vertical-align: middle;` digunakan untuk menengahkan isi vertikal dari setiap sel dalam baris tabel.

1. `td button { margin: 5px; }`

Aturan ini memberikan jarak (`margin`) sebesar 5 piksel pada tombol yang berada di dalam sel tabel (`<td>`).

2. `td button i { font-size: 20px; }`

Aturan ini mengatur ukuran font menjadi 20 piksel pada ikon (elemen `<i>`) yang berada di dalam tombol yang berada di dalam sel tabel.

3. `.modal-header { ... }`

Ini adalah aturan CSS yang mengkustomisasi tampilan bagian header dari sebuah modal. Dalam hal ini, mengubah warna latar belakang dan warna teks.

4. `.modal-body form .imgholder { ... }`

Mendefinisikan tampilan dari elemen yang mengelilingi gambar pada sebuah form di dalam modal. Menentukan ukuran, bentuk, dan tata letak dari elemen tersebut.

5. `.imgholder .upload { ... }`

Membuat tampilan saat elemen `.imgholder` (yang mengelilingi gambar) dihover. Pada saat dihover, menampilkan tampilan unggah (`upload`) yang semula tersembunyi.

6. `.modal-body form .inputField { ... }`

Menentukan tampilan dari bagian form yang berisi bidang input. Dalam hal ini, memberikan ukuran, warna border, dan tata letak.

7. `form .inputField > div label { ... }`

Mendefinisikan tampilan label pada bagian form. Menetapkan ukuran font, ketebalan teks, dan properti lainnya.

8. `#userForm form .inputField > div label::after { ... }`

Mengkustomisasi label pada form untuk elemen yang memiliki id userForm. Menambahkan simbol asterisk (*) setelah label dengan warna teks merah.

9. `.modal-footer .submit { font-size: 18px; }`

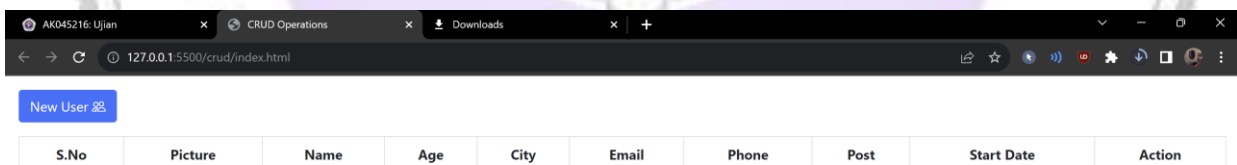
Menentukan ukuran font dari tombol submit pada bagian footer modal.

10. `#readData form .inputField > div input { ... }`

Mengkustomisasi tampilan dari input pada form yang berada di dalam elemen dengan id readData. Misalnya, mengubah warna teks dan ukuran font.

Output program:

- Tampilan halaman awal



- **Output Create** untuk pembuatan data baru dimana user harus menginput data pada form dengan menginput nama peminjam, judul buku, penerbit, tanggal pinjam dan tanggal pengembalian.

The screenshot shows a web browser window with the URL `127.0.0.1:5500/crud/index.html`. A modal titled "Fill the Form" is open, allowing a user to create a new user. The form fields are as follows:

Field	Value
Name	Rizki Eka Fachrudin
Age	21
City	jakarta
E-mail	rizkiekachrudin@gmail.com
Number	08132486739
Post	29018
Start Date	05/03/2002

Buttons: Close, Submit

- **Output Read** untuk proses pengembalian data yang telah diisikan pada form dengan menekan button tambah pada form untuk mensubmit, maka data yang telah diinput akan tampil pada halaman utama.

The screenshot shows the same web browser window. The "Fill the Form" modal is closed, and the data is now displayed in a table on the main page. The table structure is as follows:





S.No	Picture	Name	Age	City	Email	Phone	Post	Start Date	Action
1		Rizki Eka Fachrudin	21	jakarta	rizkiekachrudin@gmail.com	08132486739	29018	2002-05-03	

- **Output Update** untuk mengubah data pada output yang telah tampil pada halaman utama dengan klik button edit yang direpresentasikan dengan logo pensil, yang nantinya akan dikembalikan kembali ke dalam form untuk mengganti data yang ingin diganti. Kemudian klik button tambah untuk menampilkan hasil perubahannya pada halaman utama.

The screenshot shows a web browser window with a tab titled 'CRUD Operations'. The address bar shows '127.0.0.1:5500/crud/index.html'. A modal window titled 'Update The Form' is open, displaying a form with the following fields:

- Name: Rizki Eka Fachrudin
- Age: 21
- City: jakarta
- E-mail: rizkiekafachrudin@gmail.com
- Number: 08132486739
- Post: 29018
- Start Date: 05/03/2002

At the bottom of the modal are 'Close' and 'Update' buttons. In the background, a table is visible with the following data:

S.No	Picture	Name	Age	City	Email	Phone	Post	Start Date	Action
1		Rizki Eka Fachrudin	21	jakarta	rizkiekafachrudin@gmail.com	08132486739	29018	2002-05-03	  



- **Output Delete** untuk melakukan proses penghapusan output pada halaman utama dengan menekan button delete yang direpresentasikan dengan logo tempat sampah.

AK045216: Ujian

CRUD Operations

127.0.0.1:5500/crud/index.html

New User

127.0.0.1:5500 says
Are you sure want to delete?

OK

Cancel

S.No	Picture	Name	Age	City	Email	Phone	Post	Start Date	Action
1		Rizki Eka Fachrudin	21	jakarta	rizkiekafachrudin@gmail.com	08132486739	29018	2002-05-03	
2		Gilang Akmal Hanif	20	jakarta	gialngakmalhanif412@gmail.com	01826816388	12827	2003-03-15	
3		Muhammad Hafid Mabruuri	20	jakarta	muhammadhafidmabruri@gmail.com	08173625671	12343	2003-12-03	
4		Jamal	23	jakarta	jamal@gmail.com	09873645623	13435	2002-12-06	

AK045216: Ujian

CRUD Operations

127.0.0.1:5500/crud/index.html

New User

S.No	Picture	Name	Age	City	Email	Phone	Post	Start Date	Action
1		Rizki Eka Fachrudin	21	jakarta	rizkiekafachrudin@gmail.com	08132486739	29018	2002-05-03	
2		Gilang Akmal Hanif	20	jakarta	gialngakmalhanif412@gmail.com	01826816388	12827	2003-03-15	
3		Muhammad Hafid Mabruuri	20	jakarta	muhammadhafidmabruri@gmail.com	08173625671	12343	2003-12-03	

AK045216: Ujian

CRUD Operations

127.0.0.1:5500/crud/index.html

New User

BAB IV

PENUTUP

4.1 Kesimpulan

Dengan mengikuti langkah-langkah di atas, Anda dapat membuat aplikasi web yang mampu melakukan CRUD operations dengan menggunakan HTML, CSS, Bootstrap v5, dan JavaScript Local Storage. Manfaatkanlah pengetahuan dan kreativitas Anda untuk mengembangkan aplikasi yang lebih kompleks dan bermanfaat.

4.2 Saran

Lebih teliti lagi pada saat memberikan sebuah quiz baik itu pada pre test, act, dan post test karena masih adanya kesalahan pada pemberian jawaban yang diberikan oleh kakak asisten lab. Selain itu, terkadang masih adanya kesalahan teknis lainnya seperti link gmeet yang terlambat, dsb. Oleh karena itu, mohon dikurangi kesalahan yang telah terjadi sebelumnya agar praktikan lebih nyaman mengerjakan praktikum.

