

**LAPORAN AKHIR PRAKTIKUM
PEMROGRAMAN WEB II**



Oleh:

Wanda Karliyanti NIM. 2210817220034

**PROGRAM STUDI TEKNOLOGI INFORMASI
FAKULTAS TEKNIK
UNIVERSITAS LAMBUNG MANGKURAT
2024**

LEMBAR PENGESAHAN

LAPORAN AKHIR PRAKTIKUM PEMROGRAMAN WEB II

Laporan Praktikum Pemrograman Web II

Modul 1: Variabel, Operator dan Array

Modul 2: HTTP Request dan Kondisional

Modul 3: Perulangan dan Kondisional

Modul 4: Array

Modul 5: Function dan Database

Modul 6: CRUD dan Login pada Codeigniter

ini disusun sebagai syarat lulus mata kuliah Praktikum Pemrograman Web II. Laporan Akhir Praktikum ini dikerjakan oleh:

Nama Praktikan : Wanda Karliyanti

NIM : 2210817220034

Menyetujui,
Asisten Praktikum

Mengetahui,
Dosen Penanggung Jawab Praktikum

Muhammad Ammarin Ihsan
NIM. 2010817210002

Ir. Eka Setya Wijaya, S.T., M.Kom.
NIP. 198205082008011010

DAFTAR ISI

LEMBAR PENGESAHAN	2
DAFTAR ISI	3
DAFTAR GAMBAR	6
DAFTAR TABEL	7
MODUL 1:.....	9
SOAL 1	9
A. Source Code.....	9
B. Output Program	9
C. Pembahasan.....	9
SOAL 2	11
A. Source Code.....	11
B. Output Program	11
C. Pembahasan	12
SOAL 3	13
A. Source Code.....	13
B. Output Program	13
C. Pembahasan	14
SOAL 4	15
A. Source Code	15
B. Output Program	16
C. Pembahasan	16
SOAL 5	18
A. Source Code.....	18
B. Output Program	20
C. Pembahasan	20
MODUL 2:.....	22
SOAL 1	22
A. Source Code.....	22

B. Output Program	24
C. Pembahasan	24
SOAL 2	26
A. Source Code.....	26
B. Output Program	29
C. Pembahasan	29
SOAL 3	31
A. Source Code.....	31
B. Output Program	34
C. Pembahasan	34
SOAL 4	37
A. Source Code.....	37
B. Output Program	38
C. Pembahasan	38
MODEL 3:	40
SOAL 1	40
A. Source Code.....	40
B. Output Program	42
C. Pembahasan	42
SOAL 2	44
A. Source Code.....	44
B. Output Program	46
C. Pembahasan	46
SOAL 3	49
A. Source Code.....	49
B. Output Program	51
C. Pembahasan	51
SOAL 4	53
A. Source Code.....	53
B. Output Program	55
C. Pembahasan	56
SOAL 5	58

A.	Source Code.....	58
B.	Output Program	60
C.	Pembahasan	60
MODUL 4:	62
SOAL 1		62
A.	Source Code.....	62
B.	Output Program	65
C.	Pembahasan	65
SOAL 2		68
A.	Source Code.....	68
B.	Output Program	71
C.	Pembahasan	71
SOAL 3		73
A.	Source Code.....	74
B.	Output Program	78
C.	Pembahasan	78
MODUL 5:	81
SOAL 1		81
A.	Source Code.....	82
B.	Output Program	103
C.	Pembahasan	104
MODUL 6:	122
SOAL 1		122
A.	Source Code.....	122
B.	Output Program	146
A.	Pembahasan	147
LINK GITHUB		160

DAFTAR GAMBAR

MODUL 1: VARIABEL, OPERATOR DAN ARRAY

Gambar 1. Screenshot Hasil Jawaban Soal 1	9
Gambar 2. Screenshot Hasil Jawaban Soal 2	11
Gambar 3. Screenshot Hasil Jawaban Soal 3	13
Gambar 4. Screenshot Hasil Jawaban Soal 4	16
Gambar 5. Screenshot Hasil Jawaban Soal 5	20

MODUL 2: HTTP REQUEST DAN KONDISIONAL

Gambar 6. Screenshot Hasil Jawaban Soal 1	24
Gambar 7. Screenshot Hasil Jawaban Soal 2	29
Gambar 8. Screenshot Hasil Jawaban Soal 3	34
Gambar 9. Screenshot Hasil Jawaban Soal 4	38

MODUL 3: PERULANGAN DAN KONDISIONAL

Gambar 10. Screenshot Hasil Jawaban Soal 1	42
Gambar 11. Screenshot Hasil Jawaban Soal 2	46
Gambar 12. Screenshot Hasil Jawaban Soal 3	51
Gambar 13. Screenshot Hasil Jawaban Soal 4	55
Gambar 14. Screenshot Hasil Jawaban Soal 5	60

MODUL 4: ARRAY

Gambar 15. Screenshot Hasil Jawaban Soal 1	65
Gambar 16. Screenshot Hasil Jawaban Soal 2	71
Gambar 17. Screenshot Hasil Jawaban Soal 3	78

MODUL 5: FUNCTION DAN DATABASE

Gambar 18. Screenshot Hasil Jawaban Soal 1	104
--	-----

MODUL 6: CRUD DAN LOGIN PADA CODEIGNITER

Gambar 19. Screenshot Hasil Jawaban Soal 1	147
--	-----

DAFTAR TABEL

MODUL 1: VARIABEL, OPERATOR DAN ARRAY

Tabel 1. Source Code Soal 1	9
Tabel 2. Source Code Soal 2	11
Tabel 3. Source Code Soal 3	13
Tabel 4. Source Code Soal 4	16
Tabel 5. Source Code Soal 5	19

MODUL 2: HTTP REQUEST DAN KONDISIONAL

Tabel 6. Source Code Soal 1	23
Tabel 7. Source Code Soal 2	28
Tabel 8. Source Code Soal 3	34
Tabel 9. Source Code Soal 4	38

MODUL 3: PERULANGAN DAN KONDISIONAL

Tabel 10. Source Code Soal 1	41
Tabel 11. Source Code Soal 2	46
Tabel 12. Source Code Soal 3	50
Tabel 13. Source Code Soal 4	55
Tabel 14. Source Code Soal 5	59

MODUL 4: ARRAY

Tabel 15. Source Code Soal 1	64
Tabel 16. Source Code Soal 2	71
Tabel 17. Source Code Soal 3	78

MODUL 5: FUNCTION DAN DATABASE

Tabel 18. Source Code Soal 1 Koneksi.php	82
Tabel 19. Source Code Soal 1 Model.php	84
Tabel 20. Source Code Soal 1 FormBuku.php	87
Tabel 21. Source Code Soal 1 Buku.php	90
Tabel 22. Source Code Soal 1 FormMember.php	94
Tabel 23. Source Code Soal 1 Member.php	97
Tabel 24. Source Code Soal 1 FormPeminjaman.php	100

Tabel 25. Source Code Soal 1 Peminjaman.php	102
---	-----

MODUL 6: CRUD DAN LOGIN PADA CODEIGNITER

Tabel 26. Source Code Soal 1 Routes.php	123
---	-----

Tabel 27. Source Code Soal 1 Auth.php.....	125
--	-----

Tabel 28. Source Code Soal 1 Buku.php	129
---	-----

Tabel 29. Source Code Soal 1 2024-06-10-064318_CreateDbModul6.php	130
---	-----

Tabel 30. Source Code Soal 1 UserSeeder.php.....	131
--	-----

Tabel 31. Source Code Soal 1 AuthFilter.php.....	131
--	-----

Tabel 32. Source Code Soal 1 BukuModel.php	132
--	-----

Tabel 33. Source Code Soal 1 UserModel.php	133
--	-----

Tabel 34. Source Code Soal 1 CreateView.php	137
---	-----

Tabel 35. Source Code Soal 1 EditView.php	140
---	-----

Tabel 36. Source Code Soal 1 IndexView.php	143
--	-----

Tabel 37. Source Code Soal 1 LoginView.php	145
--	-----

MODUL 1: VARIABEL, OPERATOR DAN ARRAY

SOAL 1

Buatlah program yang dapat mencetak kalimat “Hello World in PHP” menggunakan Bahasa pemrograman PHP.

Output
Hello World in PHP Nama Praktikan : {diganti Nama Anda} NIM : {diganti NIM anda}

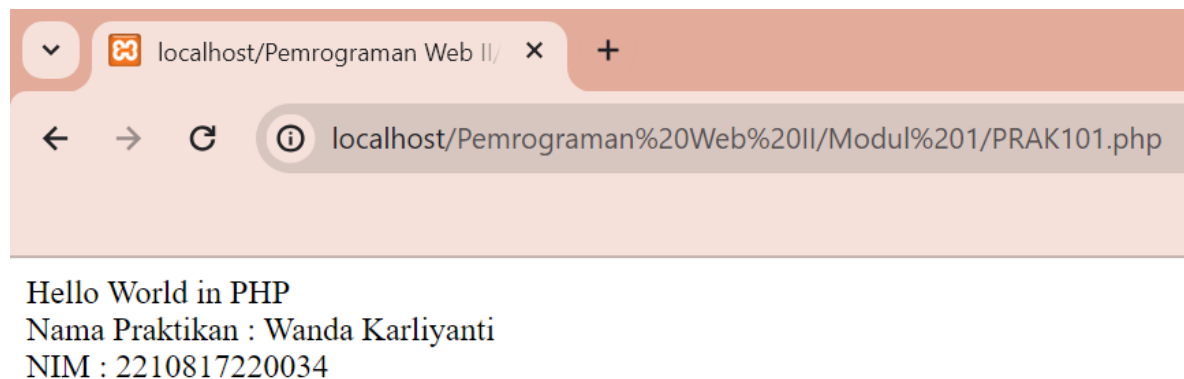
Simpan dengan nama file: PRAK101.php

A. Source Code

1	<?php
2	echo "Hello World in PHP ";
3	echo "Nama Praktikan : Wanda Karliyanti ";
4	echo "NIM : 2210817220034 ";
5	?>

Tabel 1. Source Code Soal 1

B. Output Program



Gambar 1. Screenshot Hasil Jawaban Soal 1

C. Pembahasan

Line [1], [5], <?php digunakan sebagai tag pembuka dan ?> digunakan sebagai tag penutup program PHP.

Line [2], [3], [4], `echo` digunakan untuk menampilkan teks. `
` digunakan untuk menyisipkan baris baru.

SOAL 2

Buatlah program dengan Bahasa pemrograman PHP yang dapat menghitung volume bangun ruang dan menampilkan hasilnya dengan 3 desimal dibelakang koma. Parameter awal yang diperlukan dideklarasikan secara **statis**. Bangun ruang yang digunakan disesuaikan dengan ketentuan berikut:

- NIM dengan akhiran 0/1 : Bangun ruang tabung
- NIM dengan akhiran 2/3 : Bangun ruang kerucut
- NIM dengan akhiran 4/5 : Bangun ruang bola
- NIM dengan akhiran 6/7 : Bangun ruang prisma alas segitiga
- NIM dengan akhiran 8/9 : Bangun ruang limas alas persegi Panjang

Gunakan nilai berikut yang diperlukan sesuai bangun ruang	Output
Jari-jari = 4.2 Tinggi = 5.4 Panjang = 8.9 Lebar = 14.7 Sisi = 7.9	? m3

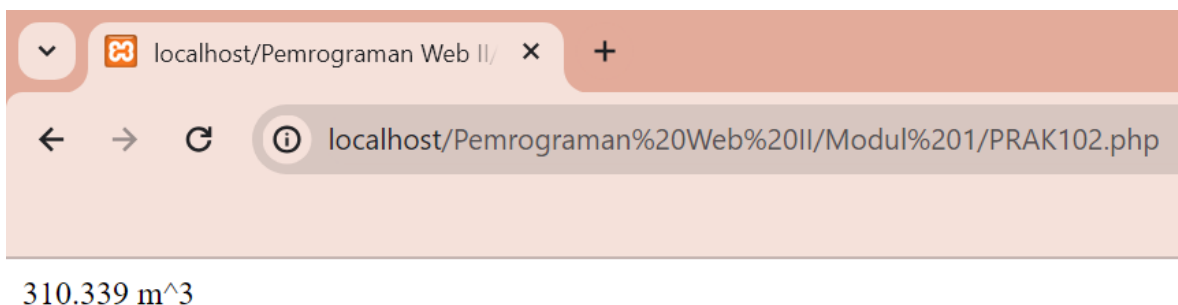
Simpan dengan nama file: PRAK102.php

A. Source Code

1	<?php
2	\$jariJari = 4.2;
3	\$volumeBola = (4/3) * pi() * pow(\$jariJari, 3);
4	
5	echo number_format(\$volumeBola, 3, '.', '') . " m^3";
6	?>

Tabel 2. Source Code Soal 2

B. Output Program



Gambar 2. Screenshot Hasil Jawaban Soal 2

C. Pembahasan

Line [1], [6], `<?php` digunakan sebagai tag pembuka dan `?>` digunakan sebagai tag penutup program PHP.

Line [2], `$jariJari = 4.2;` adalah deklarasi statis untuk nilai jari jari.

Line [3], `$volumeBola = (4/3) * pi() * pow($jariJari, 3);` adalah deklarasi statis untuk rumus volume bola. `pow()` digunakan untuk mengangkat suatu bilangan.

Line [5], `echo` digunakan untuk menampilkan teks. `number_format()` digunakan untuk memformat tampilan angka.

SOAL 3

Buatlah program dalam Bahasa pemrograman php yang dapat mengkonversi nilai suhu celcius menjadi Reamur, Fahrenheit dan Kelvin. Tampilkan hasilnya dengan format 4 desimal dibelakang koma. Nilai input dideklarasikan secara **statis**.

Input	Output
Celcius = 37.841	Fahrenheit (F) = 100,1138 Reamur (R) = 30,2728 Kelvin (K) = 310,991

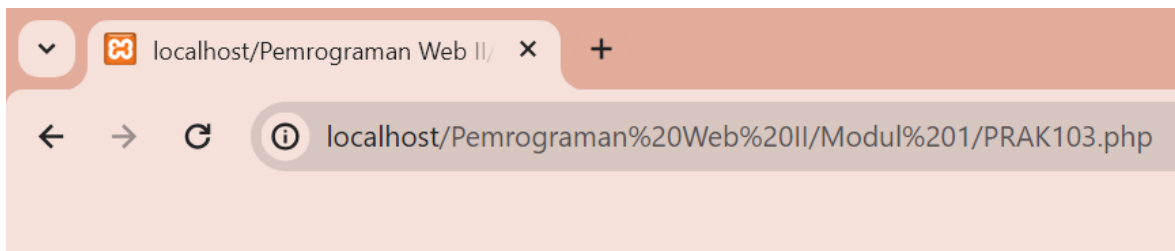
Simpan dengan nama file: PRAK103.php

A. Source Code

1	<?php
2	\$celcius = 37.841;
3	\$reamur = (4/5) * \$celcius;
4	\$fahrenheit = ((9/5) * \$celcius)+32;
5	\$kelvin = \$celcius + 273.15;
6	
7	echo "Fahrenheit (F) = " . number_format(\$fahrenheit, 4,
	',' , '') . " ";
8	echo "Reamur (R) = " . number_format(\$reamur, 4, ',' , '')
	. " ";
9	echo "Kelvin (K) = " . rtrim(number_format(\$kelvin, 4,
	',' , ''), "0") . " ";
10	?>

Tabel 3. Source Code Soal 3

B. Output Program



Fahrenheit (F) = 100,1138
Reamur (R) = 30,2728
Kelvin (K) = 310,991

Gambar 3. Screenshot Hasil Jawaban Soal 3

C. Pembahasan

Line [1], [10], `<?php` digunakan sebagai tag pembuka dan `?>` digunakan sebagai tag penutup program PHP.

Line [2], `$celcius = 37.841;` adalah deklarasi statis untuk nilai celcius.

Line [3], `$reamur = (4/5) * $celcius;` adalah deklarasi statis untuk rumus konversi dari celcius ke reamur.

Line [4], `$fahrenheit = ((9/5) * $celcius)+32;` adalah deklarasi statis untuk rumus konversi dari celcius ke fahrenheit.

Line [5], `$ kelvin = $celcius + 273.15;` adalah deklarasi statis untuk rumus konversi dari celcius ke kelvin.

Line [7 – 9], `echo` digunakan untuk menampilkan teks. `number_format()` digunakan untuk memformat tampilan angka. `
` digunakan untuk menyisipkan baris baru.

Line [9], `rtrim()` digunakan untuk menghapus karakter yang ditentukan.

SOAL 4

Buatlah program dalam Bahasa pemrograman PHP yang menampilkan daftar data seperti ditunjukkan pada Gambar 4. Gunakan **indexed array** dalam implementasinya serta Gunakan kombinasi HTML, CSS dan PHP.

Daftar Smartphone Samsung
Samsung Galaxy S22
Samsung Galaxy S22+
Samsung Galaxy A03
Samsung Galaxy Xcover 5

Simpan dengan nama file: PRAK104.php

A. Source Code

```
1 <html>
2     <head>
3         <title>Data Smartphone Samsung</title>
4         <style>
5             table {
6                 width: 195px;
7                 border-collapse: collapse;
8             }
9             table, th, td {
10                border: 1px solid black;
11            }
12            th, td {
13                padding: 2px;
14                text-align: left;
15            }
16        </style>
17    </head>
18    <body>
19        <table>
20            <tr><td><table>
21                <th>Data Smartphone Samsung</th>
22            </table></td></tr>
23
24            <?php
25                $dataSmartphone = ["Samsung Galaxy S22",
26                "Samsung Galaxy S22+", "Samsung Galaxy A03", "Samsung
27                Galaxy Xcover 5"];
28                foreach ($dataSmartphone as $smartphone):
```

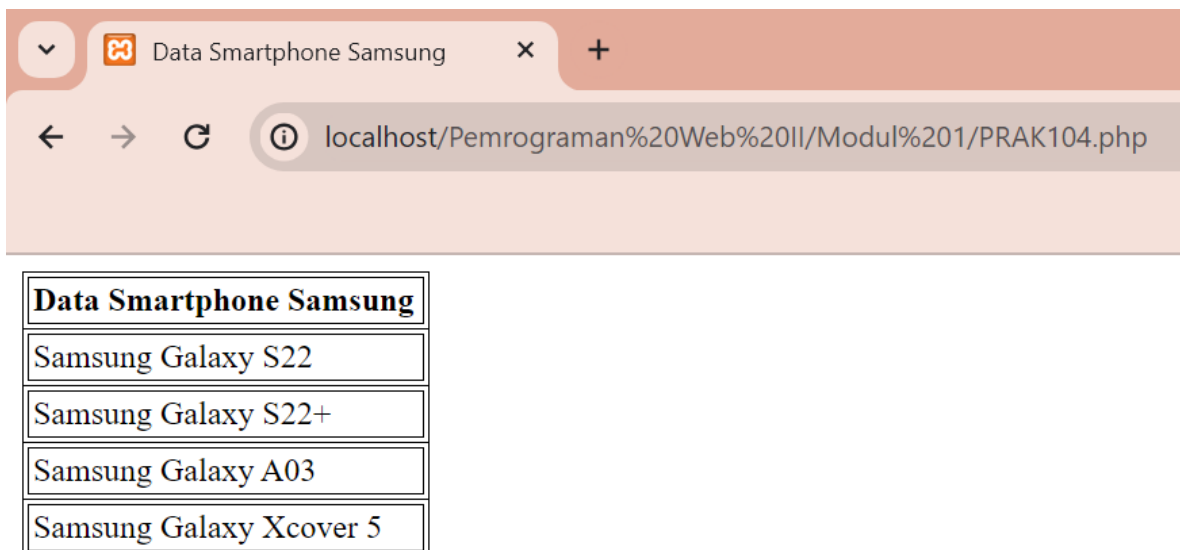
```

27         ?>
28
29         <tr><td><table>
30             <td>
31                 <?php
32                     echo htmlspecialchars($smartphone);
33                 ?>
34             </td>
35         </table></td></tr>
36
37         <?php
38             endforeach;
39         ?>
40     </table>
41 </body>
42 </html>

```

Tabel 4. Source Code Soal 4

B. Output Program



Gambar 4. Screenshot Hasil Jawaban Soal 4

C. Pembahasan

Line [1], [42], <html> digunakan sebagai tag pembuka dan </html> sebagai tag penutup dokumen HTML.

Line [2], [17], <head> digunakan sebagai tag pembuka dan </head> sebagai tag penutup bagian kepala dari dokumen HTML.

Line [3], `<title>` digunakan sebagai tag pembuka dan `</title>` sebagai tag penutup untuk membuat judul dari sebuah halaman.

Line [4], [16], `<style>` digunakan sebagai tag pembuka dan `</style>` sebagai tag penutup untuk menyisipkan kode style atau CSS ke dalam sebuah dokumen HTML.

Line [5 – 8], `table {}` digunakan untuk menargetkan elemen tabel di halaman web. `width: 195px;` digunakan untuk mengatur lebar tabel. `border-collapse: collapse;` digunakan untuk mengatur penggabungan batas sel tabel.

Line [9–11], `table, th, td {}` digunakan untuk menargetkan elemen tabel, sel header, dan sel data di halaman web. `border: 1px solid black;` digunakan untuk mengatur tebal dan warna border.

Line [12 – 15], `th, td {}` digunakan untuk menargetkan elemen sel header dan sel data di halaman web. `padding: 2px;` digunakan untuk mengatur jarak dalam sel. `text-align: left` digunakan untuk mengatur penjumlahan teks.

Line [18], [41], `<body>` digunakan sebagai tag pembuka dan `</body>` sebagai tag penutup untuk bagian tubuh atau isi dari sebuah halaman.

Line [19], [20], [22], [29], [35], [40], `<table>` digunakan sebagai tag pembuka dan `</table>` sebagai tag penutup untuk membuat tabel.

Line [20], [22], [29], [35], `<tr>` digunakan sebagai tag pembuka dan `</tr>` sebagai tag penutup untuk membuat baris dalam sebuah tabel.

Line [20], [22], [29], [30], [34], [35], `<td>` digunakan sebagai tag pembuka dan `</td>` sebagai tag penutup untuk membuat sel atau kotak dalam tabel.

Line [21], `<th>` digunakan sebagai tag pembuka dan `</th>` sebagai tag penutup untuk membuat sebuah sel header tabel.

Line [24], [27], [31], [33], [37], [39], `<?php` digunakan sebagai tag pembuka dan `?>` digunakan sebagai tag penutup program PHP.

Line [25], `$dataSmartphone = ["Samsung Galaxy S22", "Samsung Galaxy S22+", "Samsung Galaxy A03", "Samsung Galaxy Xcover 5"];` adalah deklarasi array yang berisi list data smartphone.

Line [26], `foreach ()` digunakan untuk perulangan khusus untuk memproses array.

Line [32], `echo` digunakan untuk menampilkan teks. `htmlspecialchars()` digunakan untuk mengonversi beberapa karakter menjadi entitas HTML.

Line [38], `endforeach` digunakan untuk penutup perulangan `foreach`.

SOAL 5

Buatlah program dalam Bahasa pemrograman PHP yang menampilkan daftar data seperti ditunjukkan pada Gambar 5. Gunakan **associative array** dalam implementasinya serta Gunakan kombinasi HTML, CSS dan PHP.

Daftar Smartphone Samsung
Samsung Galaxy S22
Samsung Galaxy S22+
Samsung Galaxy A03
Samsung Galaxy Xcover 5

Simpan dengan nama file: PRAK105.php

A. Source Code

```
1 <html>
2   <head>
3       <title>Data Smartphone Samsung</title>
4       <style>
5           table {
6               width: 288px;
7               border-collapse: collapse;
8           }
9           table, th, td {
10              border: 1px solid black;
11          }
12          td {
13              padding: 2px;
14              text-align: left;
15          }
16          th {
17              padding-left: 2px;
18              padding-right: 2px;
19              padding-top: 5px;
20              padding-bottom: 5px;
21              background-color: #FF0000;
22              text-align: left;
23          }
24      </style>
25  </head>
26  <body>
```

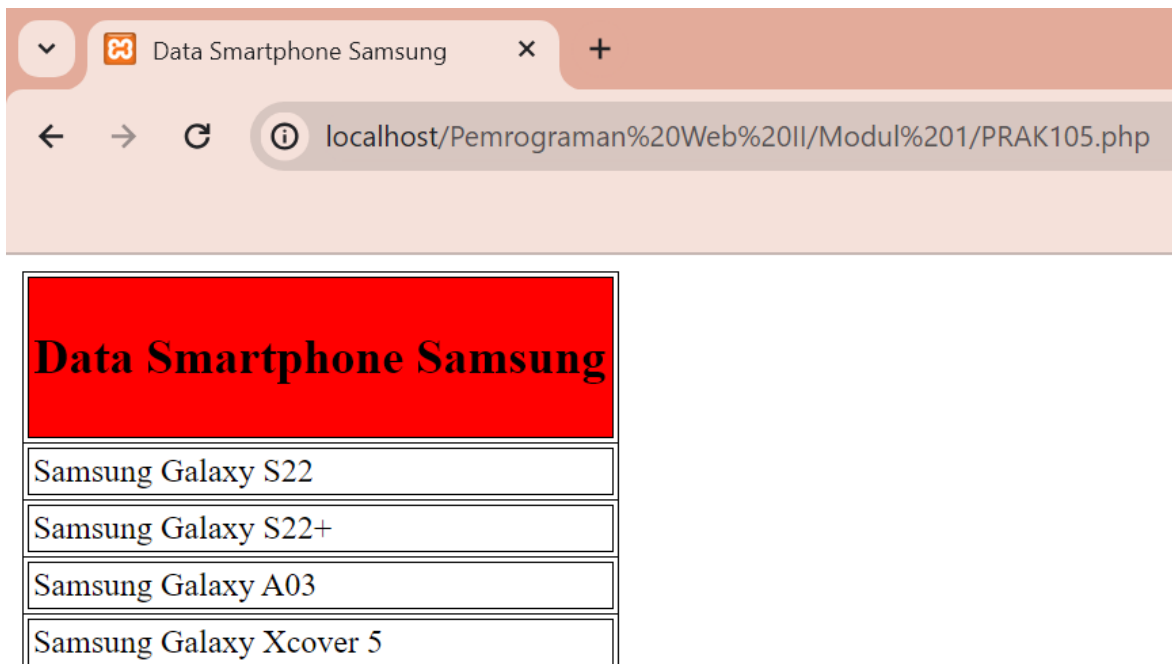
```

27         <table>
28             <tr><td><table>
29                 <th><h2>Data Smartphone Samsung</h2></th>
30             </tr></td></table>
31
32             <?php
33                 $dataSmartphone = ["S22" => "Samsung Galaxy
34 S22", "S22+" => "Samsung Galaxy S22+", "A03" => "Samsung
35 Galaxy A03", "Xcover5" => "Samsung Galaxy Xcover 5"];
36                 foreach ($dataSmartphone as $key => $value):
37                     <tr><td><table>
38                         <td>
39                             <?php
40                                 echo htmlspecialchars($value);
41                             ?>
42                         </td>
43                     </tr></td></table>
44
45                     <?php
46                         endforeach;
47                     ?>
48                 </table>
49             </body>
50 </html>

```

Tabel 5. Source Code Soal 5

B. Output Program



Gambar 5. Screenshot Hasil Jawaban Soal 5

C. Pembahasan

Line [1], [50], `<html>` digunakan sebagai tag pembuka dan `</html>` sebagai tag penutup dokumen HTML.

Line [2], [25], `<head>` digunakan sebagai tag pembuka dan `</head>` sebagai tag penutup bagian kepala dari dokumen HTML.

Line [3], `<title>` digunakan sebagai tag pembuka dan `</title>` sebagai tag penutup untuk membuat judul dari sebuah halaman.

Line [4], [24], `<style>` digunakan sebagai tag pembuka dan `</style>` sebagai tag penutup untuk menyisipkan kode style atau CSS ke dalam sebuah dokumen HTML.

Line [5 – 8], `table {}` digunakan untuk menargetkan elemen tabel di halaman web. `width: 288px;` digunakan untuk mengatur lebar tabel. `border-collapse: collapse;` digunakan untuk mengatur penggabungan batas sel tabel.

Line [9–11], `table, th, td {}` digunakan untuk menargetkan elemen tabel, sel header, dan sel data di halaman web. `border: 1px solid black;` digunakan untuk mengatur tebal dan warna border.

Line [12 – 15], `td {}` digunakan untuk menargetkan elemen sel data di halaman web. `padding: 2px;` digunakan untuk mengatur jarak dalam sel. `text-align: left` digunakan untuk mengatur penjumlahan teks.

Line [16 – 23], `th {}` digunakan untuk menargetkan elemen sel header di halaman web. `padding-left: 2px;` dan `padding-right: 2px;` digunakan untuk mengatur jarak sebelah kiri dan kanan dalam sel. `padding-top: 5px;` dan `padding-bottom: 5px;` digunakan untuk mengatur jarak sebelah atas dan bawah dalam sel. `background-color: #FF0000;` digunakan untuk mengatur warna background. `text-align: left` digunakan untuk mengatur penjumlahan teks.

Line [26], [49], `<body>` digunakan sebagai tag pembuka dan `</body>` sebagai tag penutup untuk bagian tubuh atau isi dari sebuah halaman.

Line [27], [28], [30], [37], [43], [48], `<table>` digunakan sebagai tag pembuka dan `</table>` sebagai tag penutup untuk membuat tabel.

Line [28], [30], [37], [43], `<tr>` digunakan sebagai tag pembuka dan `</tr>` sebagai tag penutup untuk membuat baris dalam sebuah tabel.

Line [28], [30], [37], [38], [42], [43], `<td>` digunakan sebagai tag pembuka dan `</td>` sebagai tag penutup untuk membuat sel atau kotak dalam tabel.

Line [29], `<th>` digunakan sebagai tag pembuka dan `</th>` sebagai tag penutup untuk membuat sebuah sel header tabel. `<h2>` digunakan sebagai tag pembuka dan `</h2>` sebagai tag penutup untuk membuat judul dalam halaman dengan ukuran heading 2.

Line [32], [35], [39], [41], [45], [47], `<?php` digunakan sebagai tag pembuka dan `?>` digunakan sebagai tag penutup program PHP.

Line [33], `$dataSmartphone = ["S22" => "Samsung Galaxy S22", "S22+" => "Samsung Galaxy S22+", "A03" => "Samsung Galaxy A03", "Xcover5" => "Samsung Galaxy Xcover 5"];` adalah deklarasi array yang berisi list data smartphone.

Line [34], `foreach ()` digunakan untuk perulangan khusus untuk memproses array.

Line [40], `echo` digunakan untuk menampilkan teks. `htmlspecialchars ()` digunakan untuk mengonversi beberapa karakter menjadi entitas HTML.

Line [46], `endforeach` digunakan untuk penutup perulangan `foreach`.

MODUL 2: HTTP REQUEST DAN KONDISIOANAL

SOAL 1

Buatlah sebuah program untuk mengurutkan 3 nama inputan dari urutan huruf abjad terkecil ke terbesar menggunakan metode kondisional. Input merupakan nama secara acak dengan output adalah hasil dari pengurutan.

Hasil yang diinginkan:

Nama: 1	<input type="text" value="andry"/>
Nama: 2	<input type="text" value="andre"/>
Nama: 3	<input type="text" value="andra"/>
<input type="button" value="Urutkan"/>	

Output
andra
andre
andry

Simpan dengan nama file: PRAK201.php

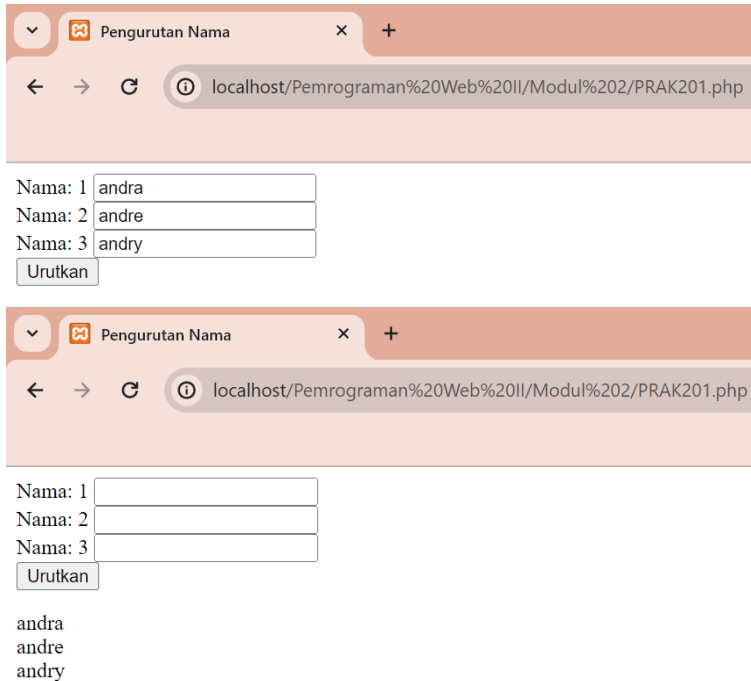
A. Source Code

```
1 <html>
2     <head>
3         <title>Pengurutan Nama</title>
4     </head>
5     <body>
6         <form method = "post" action = ">
7             <label for = "nama1">Nama: 1</label>
8             <input type = "text" name = "nama1"><br>
9
10            <label for = "nama2">Nama: 2</label>
11            <input type = "text" name = "nama2"><br>
12
13            <label for = "nama3">Nama: 3</label>
14            <input type = "text" name = "nama3"><br>
15
16            <input type = "submit" name = "submit" value
= "Urutkan">
17        </form>
18
19        <?php
20            if(isset($_POST['submit'])) {
```

21	\$nama1 = \$_POST['nama1'];
22	\$nama2 = \$_POST['nama2'];
23	\$nama3 = \$_POST['nama3'];
24	
25	if(\$nama1 < \$nama2 && \$nama1 < \$nama3){
26	\$urutan1 = \$nama1;
27	if(\$nama2 < \$nama3){
28	\$urutan2 = \$nama2;
29	\$urutan3 = \$nama3;
30	} else{
31	\$urutan2 = \$nama3;
32	\$urutan3 = \$nama2;
33	}
34	} elseif(\$nama2 < \$nama1 && \$nama2 < \$nama3){
35	\$urutan1 = \$nama2;
36	if(\$nama1 < \$nama3){
37	\$urutan2 = \$nama1;
38	\$urutan3 = \$nama3;
39	} else{
40	\$urutan2 = \$nama3;
41	\$urutan3 = \$nama1;
42	}
43	} else{
44	\$urutan1 = \$nama3;
45	if(\$nama1 < \$nama2){
46	\$urutan2 = \$nama1;
47	\$urutan3 = \$nama2;
48	} else{
49	\$urutan2 = \$nama2;
50	\$urutan3 = \$nama1;
51	}
52	}
53	
54	echo "\$urutan1 ";
55	echo "\$urutan2 ";
56	echo "\$urutan3";
57	}
58	?>
59	</body>
60	</html>

Tabel 6. Source Code Soal 1

B. Output Program



The image displays two screenshots of a web browser window titled 'Pengurutan Nama'. The address bar shows the URL 'localhost/Pemrograman%20Web%20II/Modul%202/PRAK201.php'. The first screenshot shows a form with three input fields labeled 'Nama: 1', 'Nama: 2', and 'Nama: 3' containing the values 'andra', 'andre', and 'andry' respectively. Below the fields is a button labeled 'Urutkan'. The second screenshot shows the same form, but the input fields are empty. Below the form, the names 'andra', 'andre', and 'andry' are listed vertically.

Gambar 6. Screenshot Hasil Jawaban Soal 1

C. Pembahasan

Line [1], [60], `<html>` digunakan sebagai tag pembuka dan `</html>` sebagai tag penutup dokumen HTML.

Line [2], [4], `<head>` digunakan sebagai tag pembuka dan `</head>` sebagai tag penutup bagian kepala dari dokumen HTML.

Line [3], `<title>` digunakan sebagai tag pembuka dan `</title>` sebagai tag penutup untuk membuat judul dari sebuah halaman.

Line [5], [59], `<body>` digunakan sebagai tag pembuka dan `</body>` sebagai tag penutup untuk bagian tubuh atau isi dari sebuah halaman.

Line [6], [17], `<form>` digunakan sebagai tag pembuka dan `</form>` sebagai tag penutup untuk membuat sebuah form. `method` adalah atribut untuk mengatur metode pengiriman data dan `action` adalah atribut untuk menentukan aksi yang akan dilakukan saat data dikirim.

Line [7], [10], [13], `<label>` digunakan sebagai tag pembuka dan `</label>` sebagai tag penutup untuk pelengkap keterangan elemen form. `for` adalah atribut untuk menghubungkan dengan elemen form.

Line [8], [11], [14], [16], `<input>` digunakan untuk membuat kontrol input. `type` adalah atribut untuk menentukan jenis kontrol input, `name` adalah atribut untuk memberi nama kontrol input tersebut dan `value` adalah atribut untuk mendefinisikan teks pada tombol.

Line [8], [11], [14], [54], [55], `
` digunakan untuk menyisipkan baris baru.

Line [19], [58], `<?php` digunakan sebagai tag pembuka dan `?>` digunakan sebagai tag penutup program PHP.

Line [20], [25], [27], [36], [45], `if()` digunakan untuk penentuan keputusan atau conditional statement pada kondisi utama.

Line [20], `isset()` digunakan untuk memeriksa apakah suatu variabel telah di atur.

Line [21 – 23], `$nama1 = $_POST['nama1'];` adalah inisialisasi variabel untuk mengambil nilai.

Line [26], [28], [29], [31], [32], [35], [37], [38], [40], [41], [44], [46], [47], [49], [50], `$urutan1 = $nama1;` adalah inisialisasi variabel untuk menyatakan nilai.

Line [30], [39], [43], [48], `else()` digunakan untuk penentuan keputusan atau conditional statement pada kondisi terakhir di mana semua kondisi sebelumnya tidak ada yang terpenuhi.

Line [34], `elseif()` digunakan untuk penentuan keputusan atau conditional statement digunakan pada kondisi kedua hingga seterusnya.

Line [54 – 56], `echo` digunakan untuk menampilkan teks.

SOAL 2

Buatlah program yang menerima input nama, nim, dan jenis kelamin seperti berikut:

Nama: *

Nim: *

Jenis Kelamin : *

☐ Laki-Laki

☐ Perempuan

Jika nama **atau** nim **atau** jenis kelamin kosong ketika tombol submit diklik, maka muncul pesan error seperti berikut

Nama: *

Nim: * nim tidak boleh kosong

Jenis Kelamin : * jenis kelamin tidak boleh kosong

☐ Laki-Laki

☐ Perempuan

Jika semua isian sudah terisi maka tampilkan cetak hasil input seperti berikut

Output:

Muhammad Fulan
123456
Laki-laki

Simpan dengan nama file: PRAK202.php

A. Source Code

```
1 <html>
2   <head>
3       <title>Input Nama, NIM dan Jenis Kelamin</title>
4       <style>
5           .error-message{
6               color: red;
7               font-size: 12px;
8           }
9       </style>
10  </head>
```

```

11     <body>
12         <form method = "post" action = "">
13             <label for = "nama">Nama:</label>
14             <input type = "text" name = "nama"><span
style="color: red">*</span>
15
16             <?php
17                 if ($_SERVER['REQUEST_METHOD'] == 'POST' &&
empty($_POST['nama'])) :
18                 ?>
19
20                 <span class = "error-message">nama tidak
boleh kosong</span>
21
22                 <?php
23                 endif;
24                 ?>
25                 <br>
26
27                 <label for = "nim">Nim:</label>
28                 <input type = "text" name = "nim"><span
style="color: red">*</span>
29
30                 <?php
31                 if ($_SERVER['REQUEST_METHOD'] == 'POST' &&
empty($_POST['nim'])) :
32                 ?>
33
34                 <span class = "error-message">nim tidak
boleh kosong</span>
35
36                 <?php
37                 endif;
38                 ?>
39                 <br>
40
41                 <label for = "jenisKelamin">Jenis
Kelamin:<span style="color: red">*</span></label>
42                 <?php
43                 if ($_SERVER['REQUEST_METHOD'] == 'POST' &&
empty($_POST['jenisKelamin'])) :
44                 ?>
45
46                 <span class = "error-message">jenis
kelamin tidak boleh kosong</span>
47
48                 <?php

```

```

49         endif;
50     ?>
51     <br>
52
53     <input type = "radio" id = "laki" name =
"jenisKelamin" value = "Laki-Laki">
54     <label for = "laki">Laki-Laki</label><br>
55     <input type = "radio" id = "perempuan" name
= "jenisKelamin" value = "Perempuan">
56     <label                for                =
"perempuan">Perempuan</label><br>
57
58     <input type = "submit" name = "submit" value
= "Submit">
59     </form>
60
61     <?php
62         if ($_SERVER['REQUEST_METHOD'] == 'POST' &&
!empty($_POST['nama']) && !empty($_POST['nim']) &&
!empty($_POST['jenisKelamin'])) {
63             $nama = $_POST['nama'];
64             $nim = $_POST['nim'];
65             $jenisKelamin
=
$_POST['jenisKelamin'];
66
67             echo "$nama<br>";
68             echo "$nim<br>";
69             echo "$jenisKelamin";
70         }
71     ?>
72 </body>
73 </html>

```

Tabel 7. Source Code Soal 2

B. Output Program

The first screenshot shows a web browser window with the address bar displaying 'localhost/Pemrograman%20Web%20II/Modul%202/PRAK202.php'. The form contains the following fields and labels:

- Nama: [text input] * nama tidak boleh kosong
- Nim: [text input] * nim tidak boleh kosong
- Jenis Kelamin: * jenis kelamin tidak boleh kosong
 - ☐ Laki-Laki
 - ☐ Perempuan

A 'Submit' button is located below the form. The second screenshot shows the same form after successful submission, displaying the following output:

Muhammad Fulan
123456
Laki-Laki

Gambar 7. Screenshot Hasil Jawaban Soal 2

C. Pembahasan

Line [1], [73], `<html>` digunakan sebagai tag pembuka dan `</html>` sebagai tag penutup dokumen HTML.

Line [2], [10], `<head>` digunakan sebagai tag pembuka dan `</head>` sebagai tag penutup bagian kepala dari dokumen HTML.

Line [3], `<title>` digunakan sebagai tag pembuka dan `</title>` sebagai tag penutup untuk membuat judul dari sebuah halaman.

Line [4], [9], `<style>` digunakan sebagai tag pembuka dan `</style>` sebagai tag penutup untuk menyisipkan kode style atau CSS ke dalam sebuah dokumen HTML.

Line [5 – 7], `.error-message{}` digunakan untuk menarget elemen error message. `color: red;` digunakan untuk mengatur warna teks dan `font-size: 12px;` digunakan untuk mengatur ukuran font.

Line [11], [72], `<body>` digunakan sebagai tag pembuka dan `</body>` sebagai tag penutup untuk bagian tubuh atau isi dari sebuah halaman.

Line [12], [59], `<form>` digunakan sebagai tag pembuka dan `</form>` sebagai tag penutup untuk membuat sebuah form. `method` adalah atribut untuk mengatur metode pengiriman data dan `action` adalah atribut untuk menentukan aksi yang akan dilakukan saat data dikirim.

Line [13], [27], [41], [54], [56], `<label>` digunakan sebagai tag pembuka dan `</label>` sebagai tag penutup untuk pelengkap keterangan elemen form. `for` adalah atribut untuk menghubungkan dengan elemen form.

Line [14], [28], [53], [55], [58], `<input>` digunakan untuk membuat kontrol input. `type` adalah atribut untuk menentukan jenis kontrol input, `id` adalah atribut untuk mendefinisikan id, `name` adalah atribut untuk memberi nama kontrol input tersebut dan `value` adalah atribut untuk menentukan nilai yang akan dikirim dan mendefinisikan teks pada tombol.

Line [14], [20], [28], [34], [41], [46], `` digunakan sebagai tag pembuka dan `` sebagai tag penutup untuk memformat atau mengatur style sebagian teks. `style` adalah atribut untuk menetapkan warna teks. `class` adalah atribut untuk menentukan kelas yang akan diterapkan.

Line [16], [18], [22], [24], [30], [32], [36], [38], [42], [44], [48], [50], [61], [71], `<?php` digunakan sebagai tag pembuka dan `?>` digunakan sebagai tag penutup program PHP.

Line [17], [31], [43], [62], `if()` digunakan untuk penentuan keputusan atau conditional statement pada kondisi utama.

Line [23], [36], [49], `endif` digunakan untuk menandai akhir dari sebuah conditional if.

Line [25], [39], [51], [54], [56], [67], [68], `
` digunakan untuk menyisipkan baris baru.

Line [64 – 66], `$nama = $_POST['nama']; $nim = $_POST['nim']; $jenisKelamin = $_POST['jenisKelamin'];` adalah inisialisasi variabel untuk mengambil nilai.

Line [67 – 69], `echo` digunakan untuk menampilkan teks.

SOAL 3

Buatlah sebuah program untuk melakukan konversi suhu antara Celcius (C), Fahrenheit (F), Reamur (Re) dan Kelvin (K).

Output yang diinginkan:

Nilai :

Dari :

☒ Celcius
☐ Fahrenheit
☐ Rheamur
☐ Kelvin

Ke :

☐ Celcius
☒ Fahrenheit
☐ Rheamur
☐ Kelvin

Hasil Konversi: 136.4 °F

Simpan dengan nama file: PRAK203.php

A. Source Code

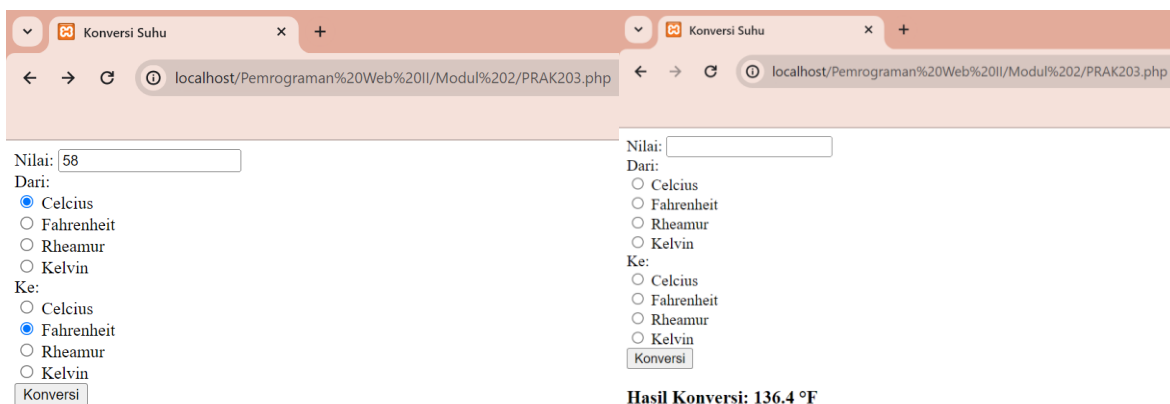
```
1 <html>
2   <head>
3     <title>Konversi Suhu</title>
4   </head>
5   <body>
6     <form method = "post" action = "">
7       <label for = "nilai">Nilai:</label>
8       <input type = "number" name = "nilai"><br>
9
10      <label for = "dari">Dari:</label><br>
11      <input type = "radio" id = "dariCelcius"
name = "dari" value = "C">
12      <label for
=
"dariCelcius">Celcius</label><br>
13      <input type = "radio" id = "dariFahrenheit"
name = "dari" value = "F">
14      <label for
=
"dariFahrenheit">Fahrenheit</label><br>
15      <input type = "radio" id = "dariRheamur"
name = "dari" value = "R">
16
```


51	\$hasil = 4/5 * \$nilai;
52	break;
53	case 'K':
54	\$hasil = \$nilai + 273.15;
55	break;
56	}
57	break;
58	case 'F':
59	switch(\$menjadi){
60	case 'C':
61	\$hasil = 5/9 * (\$nilai -
62	32);
63	break;
64	case 'F':
65	\$hasil = \$nilai;
66	break;
67	case 'R':
68	\$hasil = 4/9 * (\$nilai -
69	32);
70	break;
71	case 'K':
72	\$hasil = 5/9 * (\$nilai - 32)
73	+ 273.15;
74	break;
75	}
76	break;
77	case 'R':
78	switch(\$menjadi){
79	case 'C':
80	\$hasil = 5/4 * \$nilai;
81	break;
82	case 'F':
83	\$hasil = (9/4 * \$nilai) +
84	32;
85	break;
86	case 'R':
87	\$hasil = \$nilai;
88	break;
89	case 'K':
90	\$hasil = (5/4 * \$nilai) +
91	273.15;
92	break;
	}
	break;
	case 'K':
	switch(\$menjadi){
	case 'C':

93	\$hasil = \$nilai - 273.15;
94	break;
95	case 'F':
96	\$hasil = 9/5 * (\$nilai -
97	273.15) + 32;
98	break;
99	case 'R':
100	\$hasil = 4/5 * (\$nilai -
101	273.15);
102	break;
103	case 'K':
104	\$hasil = \$nilai;
105	break;
106	}
107	break;
108	echo "<h3>Hasil Konversi: \$hasil
109	°\$menjadi<h3>";
110	}
111	?>
	</body>
	</html>

Tabel 8. Source Code Soal 3

B. Output Program



Gambar 8. Screenshot Hasil Jawaban Soal 3

C. Pembahasan

Line [1], [111], <html> digunakan sebagai tag pembuka dan </html> sebagai tag penutup dokumen HTML.

Line [2], [4], `<head>` digunakan sebagai tag pembuka dan `</head>` sebagai tag penutup bagian kepala dari dokumen HTML.

Line [3], `<title>` digunakan sebagai tag pembuka dan `</title>` sebagai tag penutup untuk membuat judul dari sebuah halaman.

Line [5], [110], `<body>` digunakan sebagai tag pembuka dan `</body>` sebagai tag penutup untuk bagian tubuh atau isi dari sebuah halaman.

Line [6], [31], `<form>` digunakan sebagai tag pembuka dan `</form>` sebagai tag penutup untuk membuat sebuah form. `method` adalah atribut untuk mengatur metode pengiriman data dan `action` adalah atribut untuk menentukan aksi yang akan dilakukan saat data dikirim.

Line [7], [10], [12], [14], [16], [18], [20], [22], [24], [26], [28], `<label>` digunakan sebagai tag pembuka dan `</label>` sebagai tag penutup untuk pelengkap keterangan elemen form. `for` adalah atribut untuk menghubungkan dengan elemen form.

Line [8], [11], [13], [15], [17], [21], [23], [25], [27], [30], `<input>` digunakan untuk membuat kontrol input. `type` adalah atribut untuk menentukan jenis kontrol input, `id` adalah atribut untuk mendefinisikan id, `name` adalah atribut untuk memberi nama kontrol input tersebut, `value` adalah atribut untuk menentukan nilai yang akan dikirim dan mendefinisikan teks pada tombol.

Line [8], [10], [12], [14], [16], [18], [20], [22], [24], [26], [28], `
` digunakan untuk menyisipkan baris baru.

Line [33], [109], `<?php` digunakan sebagai tag pembuka dan `?>` digunakan sebagai tag penutup program PHP.

Line [34], `if()` digunakan untuk penentuan keputusan atau conditional statement pada kondisi utama. `isset()` digunakan untuk memeriksa apakah suatu variabel telah di atur.

Line [35 – 37], `$nilai = $_POST[nilai]; $dari = $_POST['dari']; $menjadi = $_POST['menjadi'];` adalah inisialisasi variabel untuk mengambil nilai.

Line [38], `$hasil = 0` adalah inisialisasi variabel untuk menyatakan nilai.

Line [40], [42], [58], [74], [90], `switch()` digunakan untuk melakukan tindakan berbeda berdasarkan kondisi berbeda.

Line [41], [43], [46], [49], [52], [57], [59], [62], [65], [68], [73], [75], [78], [81], [84], [89], [91], [94], [97], [100], `case` adalah kunci dari suatu kondisi.

Line [44], [47], [50], [53], [60], [63], [66], [69], [76], [79], [82], [85], [92], [95], [98], [101], `$hasil =` adalah deklarasi statis untuk rumus hasil konversi tiap suhu ke suhu.

Line [45], [48], [51], [54], [56], [61], [64], [67], [70], [72], [77], [80], [83], [86], [88], [93], [96], [99], [102], [104], `break` digunakan untuk mengakhiri sebuah eksekusi dalam statement.

Line [107], `echo` digunakan untuk menampilkan teks. `<h3>` digunakan sebagai tag pembuka dan `</h3>` sebagai tag penutup untuk membuat judul dalam halaman dengan ukuran heading 3.

SOAL 4

Buatlah program untuk membaca ejaan dari bilangan cacah

Format Masukan:

Sebuah baris berisi satu buah bilangan cacah dan batas maksimal 999. ($a \geq 0 < 1000$)

Format Keluaran:

Sebuah baris berisi sebuah bilangan cacah yang merupakan Ejaan dari bilangan tersebut atau batas limit bilangan.

Input	Output
3	Satuan
0	Nol
1000	Anda Menginput Melebihi Bilangan
632	Ratusan
13	Belasan

Output yang diinginkan:

Nilai :

Hasil: belasan

Simpan dengan nama file: PRAK204.php

A. Source Code

```
1 <html>
2   <head>
3       <title>Membaca Ejaan Bilangan</title>
4   </head>
5   <body>
6       <form method = "post" action = "">
7           <label for = "nilai">Nilai:</label>
8           <input type = "number" name = "nilai"><br>
9           <input type = "submit" name = "submit" value
= "Konversi">
10       </form>
11
12       <?php
13           if(isset($_POST['submit'])){
14               $nilai = $_POST['nilai'];
15
16               if($nilai == 0){
```

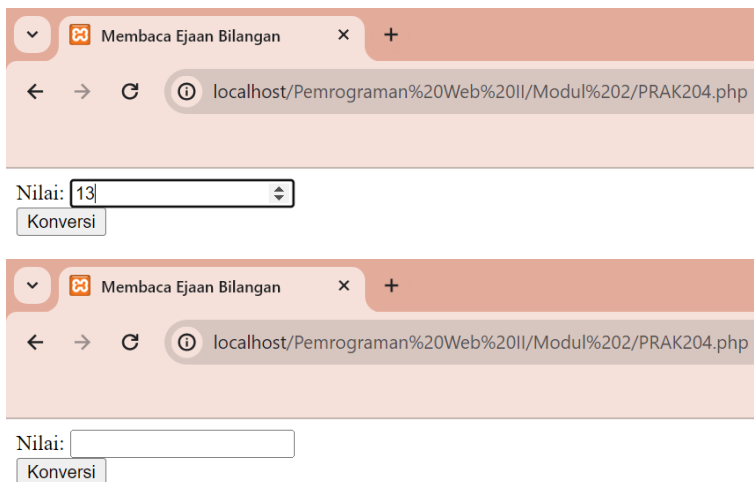
```

17         $hasil = "Nol";
18     } elseif($nilai >= 1 && $nilai < 10){
19         $hasil = "Satuan";
20     } elseif($nilai > 10 && $nilai < 20){
21         $hasil = "Belasan";
22     } elseif($nilai == 10){
23         $hasil = "Puluhan";
24     } elseif($nilai >= 20 && $nilai < 100){
25         $hasil = "Puluhan";
26     } elseif($nilai >= 100 && $nilai < 1000){
27         $hasil = "Ratusan";
28     } else{
29         $hasil = "Anda Menginput Melebihi
30 Bilangan";
31     }
32     echo "<h3>Hasil: $hasil</h3>";
33 }
34 ?>
35 </body>
36 </html>

```

Tabel 9. Source Code Soal 4

B. Output Program



Gambar 9. Screenshot Hasil Jawaban Soal 4

C. Pembahasan

Line [1], [36], <html> digunakan sebagai tag pembuka dan </html> sebagai tag penutup dokumen HTML.

Line [2], [4], `<head>` digunakan sebagai tag pembuka dan `</head>` sebagai tag penutup bagian kepala dari dokumen HTML.

Line [3], `<title>` digunakan sebagai tag pembuka dan `</title>` sebagai tag penutup untuk membuat judul dari sebuah halaman.

Line [5], [35], `<body>` digunakan sebagai tag pembuka dan `</body>` sebagai tag penutup untuk bagian tubuh atau isi dari sebuah halaman.

Line [6], [10], `<form>` digunakan sebagai tag pembuka dan `</form>` sebagai tag penutup untuk membuat sebuah form. `method` adalah atribut untuk mengatur metode pengiriman data dan `action` adalah atribut untuk menentukan aksi yang akan dilakukan saat data dikirim.

Line [7], `<label>` digunakan sebagai tag pembuka dan `</label>` sebagai tag penutup untuk pelengkap keterangan elemen form. `for` adalah atribut untuk menghubungkan dengan elemen form.

Line [8], [9], `<input>` digunakan untuk membuat kontrol input. `type` adalah atribut untuk menentukan jenis kontrol input, `name` adalah atribut untuk memberi nama kontrol input tersebut dan `value` adalah atribut untuk mendefinisikan teks pada tombol.

Line [8], `
` digunakan untuk menyisipkan baris baru.

Line [12], [34], `<?php` digunakan sebagai tag pembuka dan `?>` digunakan sebagai tag penutup program PHP.

Line [13], [16], `if()` digunakan untuk penentuan keputusan atau conditional statement pada kondisi utama.

Line [13], `isset()` digunakan untuk memeriksa apakah suatu variabel telah di atur.

Line [14], `$nilai = $_POST[nilai];` adalah inisialisasi variabel untuk mengambil nilai.

Line [17], [19], [21] [23], [25], [27], [29], `$hasil =` adalah inisialisasi variabel untuk menyatakan nilai.

Line [18], [20], [22], [24], [26], `elseif()` digunakan untuk penentuan keputusan atau conditional statement digunakan pada kondisi kedua hingga seterusnya.

Line [28], `else()` digunakan untuk penentuan keputusan atau conditional statement pada kondisi terakhir di mana semua kondisi sebelumnya tidak ada yang terpenuhi.

Line [32], `echo` digunakan untuk menampilkan teks. `<h3>` digunakan sebagai tag pembuka dan `</h3>` sebagai tag penutup untuk membuat judul dalam halaman dengan ukuran heading 3.

MODEL 3: PERULANGAN DAN KONDISIONAL

SOAL 1

Buatlah sebuah program yang dapat mencetak output tergantung banyaknya jumlah input seperti yang diberikan contoh. Wajib menggunakan perulangan **while** dalam implementasi.

Contoh Output 1:

Jumlah Peserta :

Peserta ke-1

Peserta ke-2

Peserta ke-3

Peserta ke-4

Contoh Output 2:

Jumlah Peserta :

Peserta ke-1

Peserta ke-2

Peserta ke-3

Simpan dengan nama file: PRAK301.php

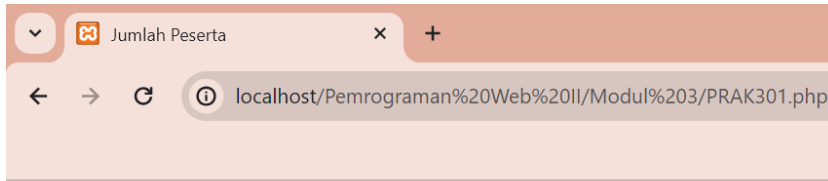
A. Source Code

```
1 <html>
2   <head>
3       <title>Jumlah Peserta</title>
4       <style>
5           .merah{
6               color: red;
7           }
```


8	.hijau{
9	color: green;
10	}
11	</style>
12	</head>
13	<body>
14	<form method = "post" action = "" autocomplete =
	"off">
15	<label for = "jumlahPeserta">Jumlah Peserta
	:</label>
16	<input type = "number" name =
	"jumlahPeserta" value = "<?php echo
	isset(\$_POST['jumlahPeserta']) ? \$_POST['jumlahPeserta']
	: ''; ?>">
17	
18	<input type = "submit" name = "submit" value
	= "Cetak">
19	</form>
20	
21	<?php
22	if(isset(\$_POST['submit'])){
23	\$jumlahPeserta = \$_POST['jumlahPeserta'];
24	\$bilangan = 1;
25	
26	while(\$bilangan <= \$jumlahPeserta){
27	if(\$bilangan %2 == 0){
28	echo " <span class =
	'hijau'><h3>Peserta ke-\$bilangan</h3>";
29	}else{
30	echo " <span class =
	'merah'><h3>Peserta ke-\$bilangan</h3>";
31	}
32	\$bilangan++;
33	}
34	}
35	?>
36	</body>
37	</html>

Tabel 10. Source Code Soal 1

B. Output Program

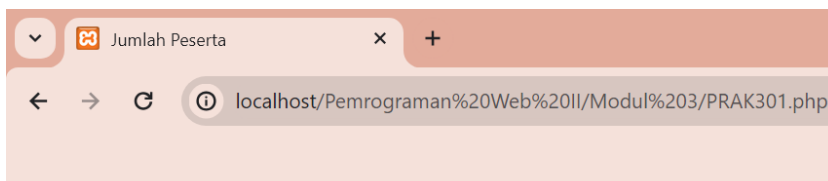


Peserta ke-1

Peserta ke-2

Peserta ke-3

Peserta ke-4



Peserta ke-1

Peserta ke-2

Peserta ke-3

Gambar 10. Screenshot Hasil Jawaban Soal 1

C. Pembahasan

Line [1], [37], `<html>` digunakan sebagai tag pembuka dan `</html>` sebagai tag penutup dokumen HTML.

Line [2], [12], `<head>` digunakan sebagai tag pembuka dan `</head>` sebagai tag penutup bagian kepala dari dokumen HTML.

Line [3], `<title>` digunakan sebagai tag pembuka dan `</title>` sebagai tag penutup untuk membuat judul dari sebuah halaman.

Line [4], [11], `<style>` digunakan sebagai tag pembuka dan `</style>` sebagai tag penutup untuk menyisipkan kode style atau CSS ke dalam sebuah dokumen HTML.

Line [5 – 7], `.merah{}` adalah deklarasi gaya dalam CSS pada kelas merah. `color: red;` digunakan untuk mengatur warna teks.

Line [8 – 10], `.hijau{}` adalah deklarasi gaya dalam CSS pada kelas hijau. `color: green;` digunakan untuk mengatur warna teks

Line [13], [36], `<body>` digunakan sebagai tag pembuka dan `</body>` sebagai tag penutup untuk bagian tubuh atau isi dari sebuah halaman.

Line [14], [19], `<form>` digunakan sebagai tag pembuka dan `</form>` sebagai tag penutup untuk membuat sebuah form. `method` adalah atribut untuk mengatur metode pengiriman data, `action` adalah atribut untuk menentukan aksi yang akan dilakukan saat data dikirim dan `autocomplete` adalah atribut untuk menampilkan opsi atau saran inputan sebelumnya yang telah dimasukkan.

Line [15], `<label>` digunakan sebagai tag pembuka dan `</label>` sebagai tag penutup untuk pelengkap keterangan elemen form. `for` adalah atribut untuk menghubungkan dengan elemen form.

Line [16], [18], `<input>` digunakan untuk membuat kontrol input. `type` adalah atribut untuk menentukan jenis kontrol input, `name` adalah atribut untuk memberi nama kontrol input tersebut dan `value` adalah atribut untuk menetapkan nilai awal input dan mendefinisikan teks pada tombol.

Line [16], [21], [35], `<?php` digunakan sebagai tag pembuka dan `?>` digunakan sebagai tag penutup program PHP.

Line [16], [22], `isset()` digunakan untuk memeriksa apakah suatu variabel telah di atur.

Line [16], [28], [30], `echo` digunakan untuk menampilkan teks. `` digunakan sebagai tag pembuka dan `` sebagai tag penutup untuk memformat atau mengatur style sebagian teks. `class` adalah atribut untuk menentukan kelas yang akan diterapkan. `<h3>` digunakan sebagai tag pembuka dan `</h3>` sebagai tag penutup untuk membuat judul dalam halaman dengan ukuran heading 3.

Line [16], `
` digunakan untuk menyisipkan baris baru.

Line [22], [27], `if()` digunakan untuk penentuan keputusan atau conditional statement pada kondisi utama.

Line [23], `$jumlahPeserta = $_POST['jumlahPeserta'];` adalah inisialisasi variabel untuk mengambil nilai.

Line [24], `$bilangan = 1;` adalah inisialisasi variabel untuk menyatakan nilai.

Line [26], `while()` berfungsi untuk mengulang suatu proses yang belum diketahui jumlahnya dengan pengecekan kondisi akan dilakukan terlebih dahulu.

Line [29], `else()` digunakan untuk penentuan keputusan atau conditional statement pada kondisi terakhir di mana semua kondisi sebelumnya tidak ada yang terpenuhi.

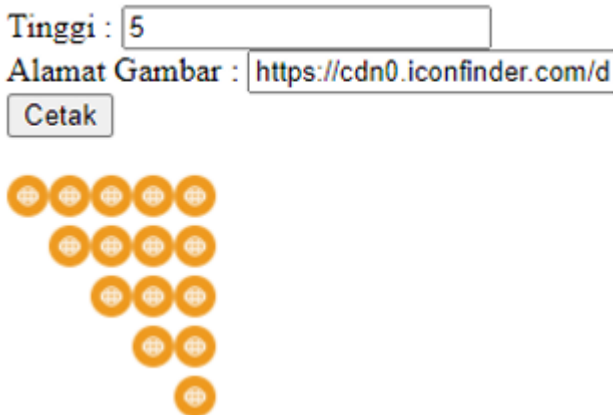
SOAL 2

Buatlah program yang dapat menampilkan bentuk segitiga dari gambar yang dimasukkan. Input baris pertama adalah tinggi segitiga. Input berikutnya adalah alamat gambar yang akan digunakan untuk membangun segitiga. Gunakan link berikut sebagai inputan gambar: <https://cdn0.iconfinder.com/data/icons/web-and-mobile-icons-volume-2/128/52-512.png>. Wajib menggunakan perulangan **while** dalam implementasinya.

Contoh Output 1:



Contoh Output 2:



Simpan dengan nama file: PRAK302.php

A. Source Code

```
1 <html>
2   <head>
3       <title>Menampilkan Gambar</title>
4   </head>
5   <body>
6       <form method = "post" action = "" autocomplete =
    "off">
```

```

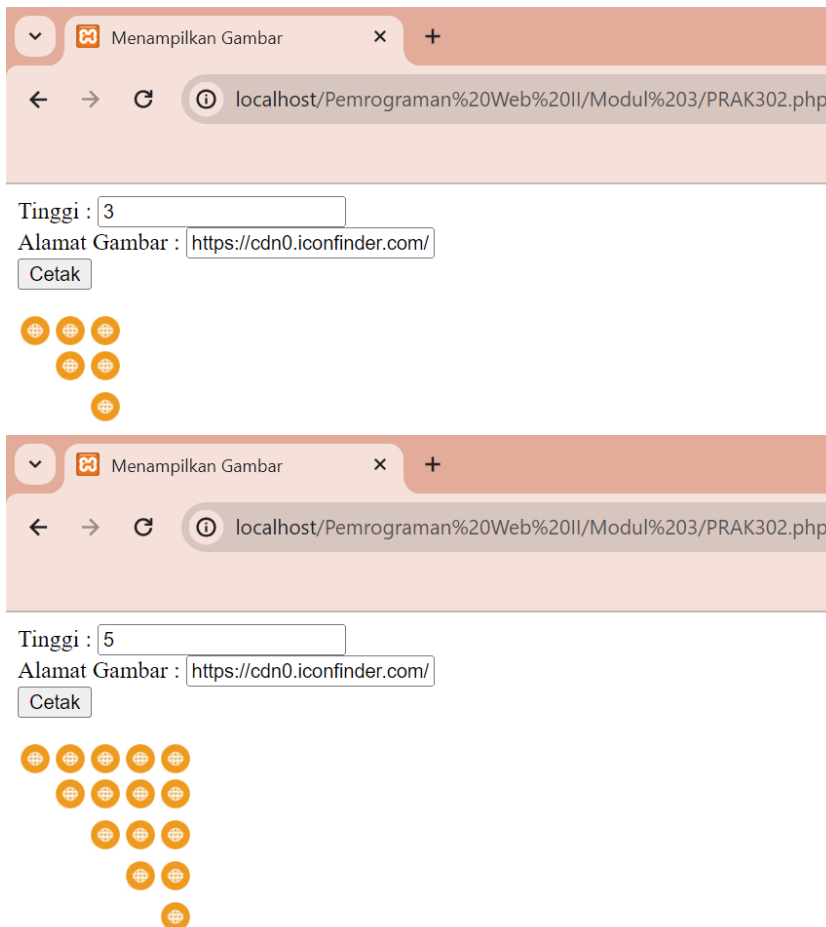
7         <label for = "tinggi">Tinggi :</label>
8         <input type = "number" name="tinggi" value =
9 "<?php echo isset($_POST['tinggi']) ? $_POST['tinggi'] :
10 '' ; ?>"><br>
11
12         <label for = "alamatGambar">Alamat Gambar
13 :</label>
14         <input type = "text" name = "alamatGambar"
15 value = "<?php echo isset($_POST['alamatGambar']) ?
16 $_POST['alamatGambar'] : '' ; ?>"><br>
17
18         <input type = "submit" name = "submit" value
19 = "Cetak">
20     </form>
21
22     <?php
23     function tampilkanGambar($alamatGambar){
24         echo "<img src='\$alamatGambar' style='width:
25 20px; height: 20px; margin: 2px;'>";
26     }
27
28     function cetakBentuk($tinggi, $gambar){
29         $i = $tinggi;
30         while($i >= 1){
31             $j = 1;
32             while($j <= $tinggi - $i){
33                 echo
34                 "&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&~";
35                 $j++;
36             }
37             $k = 1;
38             while($k <= $i){
39                 tampilkanGambar($gambar);
40                 $k++;
41             }
42             echo "<br>";
43             $i--;
44         }
45     }
46
47     if(isset($_POST['submit'])){
48         $tinggi = $_POST['tinggi'];
49         $gambar = $_POST['alamatGambar'];
50
51         if($tinggi >= 0){
52             cetakBentuk($tinggi, $gambar);
53         }else{

```

46	<pre> echo "Tinggi segitiga harus bilangan asli!"; } } ?> </body> </html> </pre>
47	
48	
49	
50	
51	

Tabel 11. Source Code Soal 2

B. Output Program



Gambar 11. Screenshot Hasil Jawaban Soal 2

C. Pembahasan

Line [1], [51], `<html>` digunakan sebagai tag pembuka dan `</html>` sebagai tag penutup dokumen HTML.

Line [2], [4], `<head>` digunakan sebagai tag pembuka dan `</head>` sebagai tag penutup bagian kepala dari dokumen HTML.

Line [3], `<title>` digunakan sebagai tag pembuka dan `</title>` sebagai tag penutup untuk membuat judul dari sebuah halaman.

Line [5], [50], `<body>` digunakan sebagai tag pembuka dan `</body>` sebagai tag penutup untuk bagian tubuh atau isi dari sebuah halaman.

Line [6], [14], `<form>` digunakan sebagai tag pembuka dan `</form>` sebagai tag penutup untuk membuat sebuah form. `method` adalah atribut untuk mengatur metode pengiriman data, `action` adalah atribut untuk menentukan aksi yang akan dilakukan saat data dikirim dan `autocomplete` adalah atribut untuk menampilkan opsi atau saran inputan sebelumnya yang telah dimasukkan.

Line [7], [10], `<label>` digunakan sebagai tag pembuka dan `</label>` sebagai tag penutup untuk pelengkap keterangan elemen form. `for` adalah atribut untuk menghubungkan dengan elemen form.

Line [8], [11], [13], `<input>` digunakan untuk membuat kontrol input. `type` adalah atribut untuk menentukan jenis kontrol input, `name` adalah atribut untuk memberi nama kontrol input tersebut dan `value` adalah atribut untuk menetapkan nilai awal input dan mendefinisikan teks pada tombol.

Line [8], [11], [34] `
` digunakan untuk menyisipkan baris baru.

Line [8], [11], [6], [49], `<?php` digunakan sebagai tag pembuka dan `?>` digunakan sebagai tag penutup program PHP.

Line [8], [11], [18], [26], [34], [46], `echo` digunakan untuk menampilkan teks.

Line [8], [11], [39], `isset()` digunakan untuk memeriksa apakah suatu variabel telah di atur.

Line [17], [21], `function` digunakan untuk menyelesaikan tugas tertentu.

Line [18], `` digunakan untuk menampilkan gambar. `src` adalah atribut untuk menentukan Alamat gambar. `style` adalah atribut untuk mengatur gaya atau penampilan elemen.

Line [22], [24], [29], `$i = $tinggi; $j = 1; $k = 1;` adalah inisialisasi variabel untuk menyatakan nilai.

Line [23], [25], [30], `while()` berfungsi untuk mengulang suatu proses yang belum diketahui jumlahnya dengan pengecekan kondisi akan dilakukan terlebih dahulu.

Line [26], ` ` digunakan untuk menentukan spasi yang tidak boleh dipecah.

Line [39], [43], `if()` digunakan untuk penentuan keputusan atau conditional statement pada kondisi utama.

Line [40], [41], `$tinggi = $_POST['tinggi']; $gambar = $_POST['alamatGambar'];` adalah inisialisasi variabel untuk mengambil nilai.

Line [45], `else()` digunakan untuk penentuan keputusan atau conditional statement pada kondisi terakhir di mana semua kondisi sebelumnya tidak ada yang terpenuhi.

Line [46], `` digunakan sebagai tag pembuka dan `` sebagai tag penutup untuk memformat atau mengatur style sebagian teks. `style` adalah atribut untuk menetapkan warna teks.

SOAL 3

Buatlah sebuah program untuk mengecek suatu bilangan pada deret, jika bilangan tersebut ditambah 7 apakah merupakan bilangan kelipatan 5. Input pertama adalah batas bawah deret. Input baris kedua adalah batas atas deret bilangan. Output adalah menampilkan deret bilangan dan bilangan yang diminta diganti dengan gambar bintang. Wajib menggunakan perulangan **do while** dalam implementasi. Source gambar bisa didownload disini: <https://www.freepnglogos.com/images/star-images-9441.html>

Contoh Output 1:

Batas Bawah :
Batas Atas :

1 2 ★ 4 5 6 7 ★ 9 10

Contoh Output 2:

Batas Bawah :
Batas Atas :

37 ★ 39 40 41 42 ★ 44 45 46 47 ★ 49 50 51 52 ★ 54 55

Simpan dengan nama file: PRAK303.php

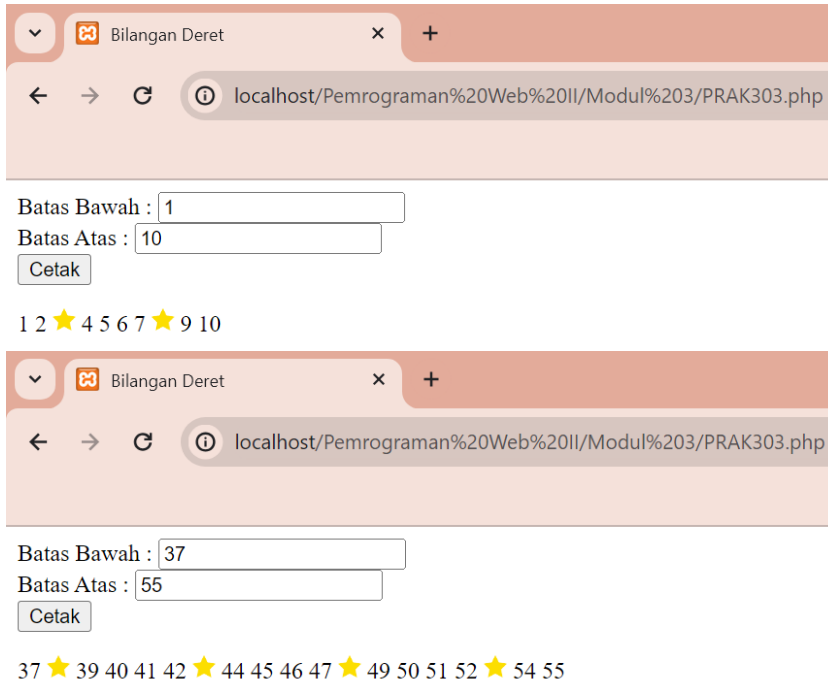
A. Source Code

```
1 <html>
2   <head>
3       <title>Bilangan Deret</title>
4   </head>
5   <body>
6       <form method = "post" action = "" autocomplete =
"off">
7           <label for = "batasBawah">Batas Bawah
: </label>
8           <input type = "number" name = "batasBawah"
value = "<?php echo isset($_POST['batasBawah']) ?
$_POST['batasBawah'] : ' '; ?>"><br>
9
10          <label for = "batasAtas">Batas Atas : </label>
```

11	<pre> <input type = "number" name = "batasAtas" value = "<?php echo isset(\$_POST['batasAtas']) ? \$_POST['batasAtas'] : ''; ?>">
 <input type = "submit" name = "submit" value = "Cetak"> </form> <?php if(isset(\$_POST['submit'])) { \$batasBawah = \$_POST['batasBawah']; \$batasAtas = \$_POST['batasAtas']; \$bilangan = \$batasBawah; do{ if((\$bilangan + 7) % 5 == 0){ echo " "; }else{ echo "\$bilangan "; } \$bilangan++; }while(\$bilangan <= \$batasAtas); } ?> </body> </html> </pre>
12	
13	
14	
15	
16	
17	
18	
19	
20	
21	
22	
23	
24	
25	
26	
27	
28	
29	
30	
31	
32	
33	

Tabel 12. Source Code Soal 3

B. Output Program



Gambar 12. Screenshot Hasil Jawaban Soal 3

C. Pembahasan

Line [1], [33], `<html>` digunakan sebagai tag pembuka dan `</html>` sebagai tag penutup dokumen HTML.

Line [2], [4], `<head>` digunakan sebagai tag pembuka dan `</head>` sebagai tag penutup bagian kepala dari dokumen HTML.

Line [3], `<title>` digunakan sebagai tag pembuka dan `</title>` sebagai tag penutup untuk membuat judul dari sebuah halaman.

Line [5], [32], `<body>` digunakan sebagai tag pembuka dan `</body>` sebagai tag penutup untuk bagian tubuh atau isi dari sebuah halaman.

Line [6], [14], `<form>` digunakan sebagai tag pembuka dan `</form>` sebagai tag penutup untuk membuat sebuah form. `method` adalah atribut untuk mengatur metode pengiriman data, `action` adalah atribut untuk menentukan aksi yang akan dilakukan saat data dikirim dan `autocomplete` adalah atribut untuk menampilkan opsi atau saran inputan sebelumnya yang telah dimasukkan..

Line [7], [10], `<label>` digunakan sebagai tag pembuka dan `</label>` sebagai tag penutup untuk pelengkap keterangan elemen form. `for` adalah atribut untuk menghubungkan dengan elemen form.

Line [8], [11], [13], `<input>` digunakan untuk membuat kontrol input. `type` adalah atribut untuk menentukan jenis kontrol input, `name` adalah atribut untuk memberi nama kontrol input tersebut, `value` adalah atribut untuk menentukan nilai awal input dan mendefinisikan teks pada tombol.

Line [8], [11], [16], [31], `<?php` digunakan sebagai tag pembuka dan `?>` digunakan sebagai tag penutup program PHP.

Line [8], [11], [24], [26], `echo` digunakan untuk menampilkan teks.

Line [8], [11], [17], `isset()` digunakan untuk memeriksa apakah suatu variabel telah di atur.

Line [8], [11], `
` digunakan untuk menyisipkan baris baru.

Line [17], [23], `if()` digunakan untuk penentuan keputusan atau conditional statement pada kondisi utama.

Line [18 – 20], `$batasBawah = $_POST['batasBawah']; $batasAtas = $_POST['batasAtas']; $bilangan = $batasBawah;` adalah inisialisasi variabel untuk mengambil nilai.

Line [22], [29], `do{} while()` berfungsi untuk menjalankan kode terlebih dahulu sebelum dilakukan pengecekan kondisi.

Line [24], `` digunakan untuk menampilkan gambar. `src` adalah atribut untuk menentukan Alamat gambar. `style` adalah atribut untuk mengatur gaya atau penampilan elemen.

Line [25], `else()` digunakan untuk penentuan keputusan atau conditional statement pada kondisi terakhir di mana semua kondisi sebelumnya tidak ada yang terpenuhi.

SOAL 4

Buatlah sebuah program untuk menampilkan banyak bintang sesuai dengan jumlah bintang pada saat diinput. Output program berupa banyak gambar bintang sesuai input serta tombol “**Tambah**” dan “**Kurang**”. Saat tombol “**Tambah**” ditekan maka jumlah bintang akan bertambah 1, apabila tombol “**Kurang**” ditekan maka jumlah bintang akan berkurang 1. Source gambar bisa didownload disini: <https://www.freepnglogos.com/images/star-images-9441.html>

Contoh Output 1 (Sebelum disubmit):

Jumlah bintang

Contoh Output 2 (Sesudah disubmit):

Jumlah bintang 5



Contoh Output 2 (Ketika tombol “**Tambah**” ditekan):

Jumlah bintang 6



Simpan dengan nama file: PRAK404.php

A. Source Code

```
1 <html>
2   <head>
3     <title>Banyak Bintang</title>
4     <style>
5       .bintang{
6         width: 50px;
```

```

7             height: 50px;
8         }
9     </style>
10 </head>
11 <body>
12     <?php
13         $jumlahBintang = isset($_POST['jumlahBintang'])
14         ? $_POST['jumlahBintang'] : '';
15
16         if(isset($_POST['submit'])) {
17             $jumlahBintang = $_POST['jumlahBintang'];
18         }
19
20         if(isset($_POST['tambah'])) {
21             $jumlahBintang += 1;
22         } elseif(isset($_POST['kurang'])) {
23             if($jumlahBintang > 0) {
24                 $jumlahBintang -= 1;
25             }
26         }
27     <?php
28         if(isset($_POST['submit']) || $jumlahBintang >
29         0) :
30         ?>
31
32         <p>Jumlah        bintang        <?=        $jumlahBintang
33         ?><br><br></p>
34
35         <?php
36         for($i = 0; $i < $jumlahBintang; $i++):
37         ?>
38
39         <img src = 'star-images-9441.png' class =
40         'bintang'>
41         <?php endfor; ?>
42
43         <form method = "post" action = "">
44             <input type = "submit" name = "tambah" value
45             = "Tambah">
46             <input type = "submit" name = "kurang" value
47             = "Kurang">
48             <input type = "hidden" name = "jumlahBintang"
49             value = "<?= $jumlahBintang ?>">
50         </form>
51         <?php else: ?>

```

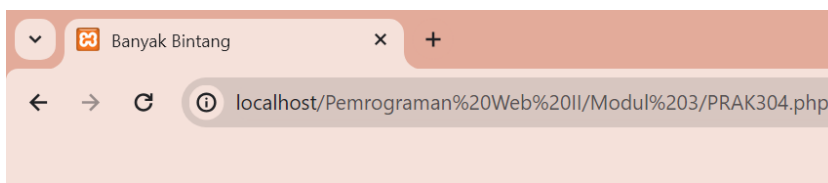
```

47
48         <form method = "post" action = "" autocomplete =
"off">
49             <label    for    =    "jumlahBintang">Jumlah
bintang</label>
50             <input type = "number" name = "jumlahBintang"
value = "<?= $jumlahBintang ?>"><br>
51             <input type = "submit" name = "submit" value
= "Submit">
52         </form>
53         <?php endif; ?>
54     </body>
55 </html>

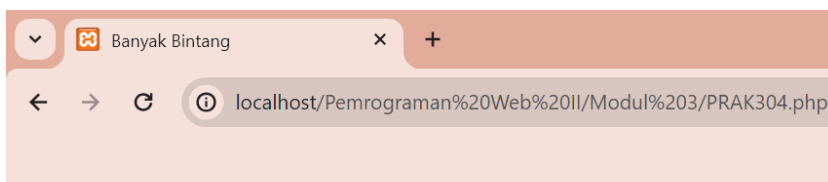
```

Tabel 13. Source Code Soal 4

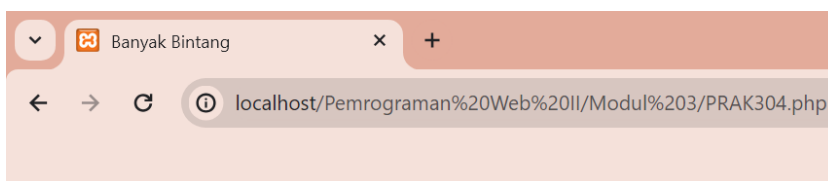
B. Output Program



A screenshot of a web browser window with the title 'Banyak Bintang'. The address bar shows 'localhost/Pemrograman%20Web%20II/Modul%203/PRAK304.php'. The page content displays a form with the label 'Jumlah bintang' followed by an empty text input field and a 'Submit' button.



A screenshot of the same web browser window. The page content now shows 'Jumlah bintang 5' followed by five yellow stars. Below the stars are two buttons: 'Tambah' and 'Kurang'.



A screenshot of the same web browser window. The page content now shows 'Jumlah bintang 6' followed by six yellow stars. Below the stars are two buttons: 'Tambah' and 'Kurang'.

Gambar 13. Screenshot Hasil Jawaban Soal 4

C. Pembahasan

Line [1], [55], `<html>` digunakan sebagai tag pembuka dan `</html>` sebagai tag penutup dokumen HTML.

Line [2], [10], `<head>` digunakan sebagai tag pembuka dan `</head>` sebagai tag penutup bagian kepala dari dokumen HTML.

Line [3], `<title>` digunakan sebagai tag pembuka dan `</title>` sebagai tag penutup untuk membuat judul dari sebuah halaman.

Line [4], [9], `<style>` digunakan sebagai tag pembuka dan `</style>` sebagai tag penutup untuk menyisipkan kode style atau CSS ke dalam sebuah dokumen HTML.

Line [5 – 8], `.bintang{ }` adalah deklarasi gaya dalam CSS pada kelas bintang. `width: 50px;` digunakan untuk mengatur lebar elemen. `height: 50px` digunakan untuk mengatur tinggi elemen.

Line [11], [54], `<body>` digunakan sebagai tag pembuka dan `</body>` sebagai tag penutup untuk bagian tubuh atau isi dari sebuah halaman.

Line [12], [26], [28], [30], [34], [36], [39], [46], [53], `<?php` digunakan sebagai tag pembuka dan `?>` digunakan sebagai tag penutup program PHP.

Line [13], [20], [23], `$jumlahBintang = isset($_POST['jumlahBintang']) ? $_POST['jumlahBintang'] : ''; $jumlahBintang += 1; $jumlahBintang-= 1;` adalah inisialisasi variabel untuk menyatakan nilai.

Line [13], [15], [19], [21], [29], `isset()` digunakan untuk memeriksa apakah suatu variabel telah di atur.

Line [15], [19], [22], [29], `if()` digunakan untuk penentuan keputusan atau conditional statement pada kondisi utama.

Line [16], `$jumlahBintang = $_POST['jumlahBintang'];` adalah inisialisasi variabel untuk mengambil nilai.

Line [21], `elseif()` digunakan untuk penentuan keputusan atau conditional statement digunakan pada kondisi kedua hingga seterusnya.

Line [32], `<p>` adalah tag pembuka dan `</p>` adalah tag penutup untuk membuat paragraf.

Line [32], [50], `
` digunakan untuk menyisipkan baris baru.

Line [35], `for()` digunakan untuk membuat sebuah perulangan.

Line [38], `` digunakan untuk menampilkan gambar. `src` adalah atribut untuk menentukan Alamat gambar. `class` adalah atribut untuk menentukan kelas yang akan diterapkan.

Line [39], `endfor` digunakan untuk menandai akhir dari perulangan.

Line [41], [45], [48], [52], `<form>` digunakan sebagai tag pembuka dan `</form>` sebagai tag penutup untuk membuat sebuah form. `method` adalah atribut untuk mengatur metode pengiriman data, `action` adalah atribut untuk menentukan aksi yang akan dilakukan saat data dikirim dan `autocomplete` adalah atribut untuk menampilkan opsi atau saran inputan sebelumnya yang telah dimasukkan.

Line [42 – 44], [50], [51], `<input>` digunakan untuk membuat kontrol input. `type` adalah atribut untuk menentukan jenis kontrol input, `name` adalah atribut untuk memberi nama kontrol input tersebut dan `value` adalah atribut untuk mendefinisikan teks pada tombol dan menetapkan nilai awal input.

Line [47], `else()` digunakan untuk penentuan keputusan atau conditional statement pada kondisi terakhir di mana semua kondisi sebelumnya tidak ada yang terpenuhi.

Line [49], `<label>` digunakan sebagai tag pembuka dan `</label>` sebagai tag penutup untuk pelengkap keterangan elemen form. `for` adalah atribut untuk menghubungkan dengan elemen form.

Line [53], `endif` digunakan untuk menandai akhir dari sebuah condisional if.

SOAL 5

Buatlah sebuah program yang menerima masukan string kemudian mencetak string tersebut dengan masing-masing karakter pada string tercetak sebanyak panjang dari string masukan. Pada masing-masing karakter, karakter pertama harus tercetak kapital sedangkan sisanya harus tidak kapital.

Contoh program:

Contoh 1

Input:

BanDunG

Output:

BbbbbbbAaaaaaaNnnnnnnDddddddUuuuuuuNnnnnnnGgggggg

Contoh 2

Input:

BANJAR

Output:

BbbbbbbAaaaaaNnnnnJjjjjAaaaaRrrrr

Simpan dengan nama file: PRAK305.php

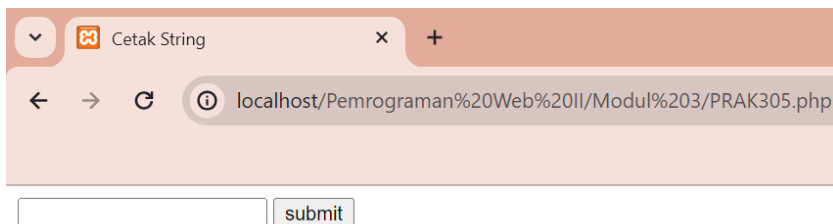
A. Source Code

1	<html>
2	<head>
3	<title>Cetak String</title>
4	</head>
5	<body>

6	<form method = "post" action = "" autocomplete =
	"off">
7	<input type = "text" name = "string">
8	
9	<input type = "submit" name = "submit" value
	= "submit">
10	</form>
11	
12	<?php
13	if(isset(\$_POST['submit'])) {
14	\$string = \$_POST['string'];
15	\$ubahString = strtolower(\$string);
16	\$karakter = str_split(\$ubahString);
17	\$panjang = strlen(\$ubahString);
18	\$hasil = "";
19	
20	foreach(\$karakter as \$index => \$char) {
21	\$karakterBerulang = "";
22	\$karakterBerulang .= strtoupper(\$char);
23	\$karakterBerulang
	str_repeat(strtolower(\$char), \$panjang - 1);
24	\$hasil .= \$karakterBerulang;
25	}
26	echo "<h3>Input:</h3>";
27	echo "\$string";
28	echo "<h3>Output:</h3>";
29	echo "\$hasil";
30	}
31	?>
32	</body>
33	</html>

Tabel 14. Source Code Soal 5

B. Output Program

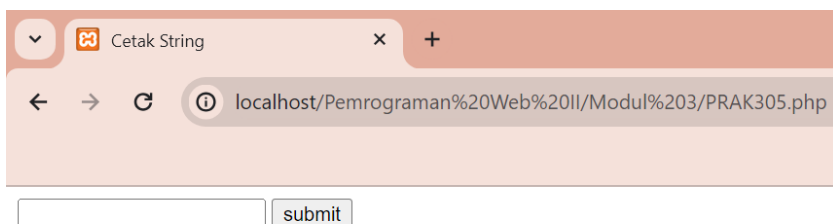


Input:

BanDunG

Output:

BbbbbbbAaaaaaaNnnnnnnDddddddUuuuuuuNnnnnnnGggggggg



Input:

BANJAr

Output:

BbbbbbbAaaaaaNnnnnnJjjjjAaaaaRrrrr

Gambar 14. Screenshot Hasil Jawaban Soal 5

C. Pembahasan

Line [1], [33], `<html>` digunakan sebagai tag pembuka dan `</html>` sebagai tag penutup dokumen HTML.

Line [2], [4], `<head>` digunakan sebagai tag pembuka dan `</head>` sebagai tag penutup bagian kepala dari dokumen HTML.

Line [3], `<title>` digunakan sebagai tag pembuka dan `</title>` sebagai tag penutup untuk membuat judul dari sebuah halaman.

Line [5], [32], `<body>` digunakan sebagai tag pembuka dan `</body>` sebagai tag penutup untuk bagian tubuh atau isi dari sebuah halaman.

Line [6], [10], `<form>` digunakan sebagai tag pembuka dan `</form>` sebagai tag penutup untuk membuat sebuah form. `method` adalah atribut untuk mengatur metode pengiriman data, `action` adalah atribut untuk menentukan aksi yang akan dilakukan saat data dikirim

dan `autocomplete` adalah atribut untuk menampilkan opsi atau saran inputan sebelumnya yang telah dimasukkan.

Line [7], [9], `<input>` digunakan untuk membuat kontrol input. `type` adalah atribut untuk menentukan jenis kontrol input, `name` adalah atribut untuk memberi nama kontrol input tersebut dan `value` adalah atribut untuk mendefinisikan teks pada tombol.

Line [12], [31], `?php` digunakan sebagai tag pembuka dan `?>` digunakan sebagai tag penutup program PHP.

Line [13], `if()` digunakan untuk penentuan keputusan atau conditional statement pada kondisi utama. `isset()` digunakan untuk memeriksa apakah suatu variabel telah di atur.

Line [14], `$string = $_POST['string'];` adalah inisialisasi variabel untuk mengambil nilai.

Line [15 – 18], [21], `$ubahString = strtolower($string); $karakter = str_split($ubahString); $panjang = strlen($ubahString); $hasil = ""; $karakterBerulang = "";` adalah inisialisasi variabel untuk menyatakan nilai.

Line [15], [23], `strtolower()` digunakan untuk mengonversi karakter dalam string menjadi huruf kecil.

Line [16], `str_split()` digunakan untuk memecah string menjadi array karakter.

Line [17], `strlen()` digunakan untuk mengembalikan panjang karakter dari string.

Line [20], `foreach()` digunakan untuk perulangan khusus untuk memproses array.

Line [22 – 24], `.=` digunakan untuk menggabungkan nilai dari variabel dengan nilai yang baru serta menyimpan hasilnya kembali ke variabel yang sama.

Line [22], `strtoupper()` digunakan untuk mengonversi karakter dalam string menjadi huruf besar.

Line [23], `str_repeat()` digunakan untuk mengulang string sebanyak yang ditentukan.

Line [26 – 29], `echo` digunakan untuk menampilkan teks.

Line [26], [28], `<h3>` digunakam sebagai tag pembuka dan `</h3>` sebagai tag penutup untuk membuat judul dalam halaman dengan ukuran heading 3.

MODUL 4: ARRAY

SOAL 1

Buatlah sebuah program yang dapat mencetak output berupa matriks tergantung banyaknya jumlah input seperti yang diberikan contoh. Input pertama adalah Panjang dari matriks, input kedua adalah lebar dari matriks, input ketiga adalah nilai dari matriks dipisahkan spasi.

Contoh Output 1:

Panjang :

Lebar :

Nilai :

34	56	41
45	36	37
51	32	46

Contoh Output 2:

Panjang :

Lebar :

Nilai :

Panjang nilai tidak sesuai dengan ukuran matriks

Simpan dengan nama file: PRAK401.php

A. Source Code

```
1 <html>
2     <head>
3         <title>Matriks</title>
4         <style>
5             table{
6                 border-collapse: collapse;
7             }
8             td{
9                 padding-right: 6px;
10                padding-left: 6px;
```

```

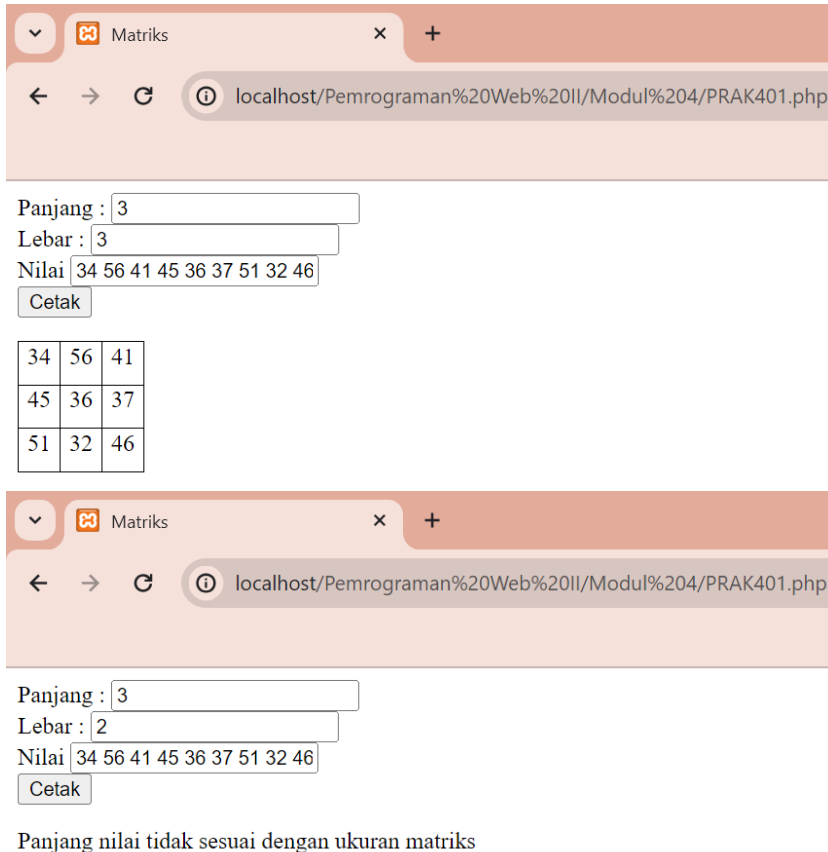
11         padding-bottom: 10px;
12     }
13     table, td{
14         border: 1px solid black;
15     }
16     </style>
17 </head>
18 <body>
19     <form                method="post"                action=""
autocomplete="off">
20         <label for="panjang">Panjang :</label>
21         <input            type="number"            name="panjang"
value="<?php            echo            isset($_POST['panjang'])            ?
$_POST['panjang'] : ''; ?>"><br>
22
23         <label for="lebar">Lebar :</label>
24         <input            type="number"            name="lebar"
value="<?php            echo            isset($_POST['lebar'])            ?
$_POST['lebar'] : ''; ?>"><br>
25
26         <label for="nilai">Nilai</label>
27         <input type="text" name="nilai" value="<?php
echo isset($_POST['nilai']) ? $_POST['nilai'] : '';
?>"><br>
28
29         <input            type="submit"            name="submit"
value="Cetak">
30     </form>
31
32     <?php
33     function cetakMatriks($panjang, $lebar, $nilai){
34         $matriks = array_chunk(explode(' ', $nilai),
$lebar);
35         echo "<table>";
36
37         for($i = 0; $i < $panjang; $i++){
38             echo "<tr>";
39             for($j = 0; $j < $lebar; $j++){
40                 echo "<td>" . $matriks[$i][$j] .
"</td>";
41             }
42         }
43         echo "</tr>";
44         echo "</table>";
45     }
46
47     if(isset($_POST['submit'])){

```

48	\$panjang = isset(\$_POST['panjang']) ?
	\$_POST['panjang'] : 0;
49	\$lebar = isset(\$_POST['lebar']) ?
	\$_POST['lebar'] : 0;
	\$nilai = isset(\$_POST['nilai']) ?
50	\$_POST['nilai'] : '';
51	\$jumlahNilai = count(explode(' ', \$nilai));
52	\$ukuranMatriks = \$panjang * \$lebar;
53	
54	if(\$jumlahNilai != \$ukuranMatriks){
55	echo "Panjang nilai tidak sesuai dengan
56	ukuran matriks";
	}else{
57	cetakMatriks(\$panjang, \$lebar, \$nilai);
58	}
59	}
60	?>
61	</body>
62	</html>
63	

Tabel 15. Source Code Soal 1

B. Output Program



Gambar 15. Screenshot Hasil Jawaban Soal 1

C. Pembahasan

Line [1], [63], `<html>` digunakan sebagai tag pembuka dan `</html>` sebagai tag penutup dokumen HTML.

Line [2], [17], `<head>` digunakan sebagai tag pembuka dan `</head>` sebagai tag penutup bagian kepala dari dokumen HTML.

Line [3], `<title>` digunakan sebagai tag pembuka dan `</title>` sebagai tag penutup untuk membuat judul dari sebuah halaman.

Line [4], [16], `<style>` digunakan sebagai tag pembuka dan `</style>` sebagai tag penutup untuk menyisipkan kode style atau CSS ke dalam sebuah dokumen HTML.

Line [5 – 7], `table{}` digunakan untuk menargetkan elemen tabel di halaman web. `border-collapse: collapse;` digunakan untuk mengatur penggabungan batas sel tabel.

Line [8 – 12], `td{}` digunakan untuk menargetkan elemen sel data di halaman web. `padding-right: 6px; padding-left: 6px; padding-bottom: 10px;` digunakan untuk mengatur jarak dalam sel sebelah kanan, kiri dan bawah.

Line [13 – 15], `table`, `td{}` digunakan untuk menargetkan elemen tabel dan sel data di halaman web. `border: 1px solid black;` digunakan untuk mengatur tebal dan warna border.

Line [18], [62], `<body>` digunakan sebagai tag pembuka dan `</body>` sebagai tag penutup untuk bagian tubuh atau isi dari sebuah halaman.

Line [19], [30], `<form>` digunakan sebagai tag pembuka dan `</form>` sebagai tag penutup untuk membuat sebuah form. `method` adalah atribut untuk mengatur metode pengiriman data, `action` adalah atribut untuk menentukan aksi yang akan dilakukan saat data dikirim dan `autocomplete` adalah atribut untuk menampilkan opsi atau saran inputan sebelumnya yang telah dimasukkan.

Line [20], [23], [26], `<label>` digunakan sebagai tag pembuka dan `</label>` sebagai tag penutup untuk pelengkap keterangan elemen form. `for` adalah atribut untuk menghubungkan dengan elemen form.

Line [21], [24], [27], [29], `<input>` digunakan untuk membuat kontrol input. `type` adalah atribut untuk menentukan jenis kontrol input, `name` adalah atribut untuk memberi nama kontrol input tersebut dan `value` adalah atribut untuk menetapkan nilai awal input dan mendefinisikan teks pada tombol.

Line [21], [24], [27], [32], [61], `<?php` digunakan sebagai tag pembuka dan `?>` digunakan sebagai tag penutup program PHP.

Line [21], [24], [27], [35], [38], [40], [43], [44], [56], `echo` digunakan untuk menampilkan teks.

Line [21], [24], [27], [47 – 50], `isset()` digunakan untuk memeriksa apakah suatu variabel telah di atur.

Line [21], [24], [27], `
` digunakan untuk menyisipkan baris baru.

Line [33], `function` digunakan untuk menyelesaikan tugas tertentu.

Line [34]. `array_chunk()` digunakan untuk membagi sebuah array menjadi potongan-potongan dengan jumlah elemen yang sama dan menyimpannya dalam array baru.

Line [34], [52], `explode()` digunakan untuk memisahkan sebuah string menjadi array, berdasarkan suatu delimiter yang ditentukan.

Line [35], [44], `<table>` digunakan sebagai tag pembuka dan `</table>` sebagai tag penutup untuk membuat tabel.

Line [37], [39], `for()` digunakan untuk membuat sebuah perulangan.

Line [38], [43], `<tr>` digunakan sebagai tag pembuka dan `</tr>` sebagai tag penutup untuk membuat baris dalam sebuah tabel.

Line [40], `<td>` digunakan sebagai tag pembuka dan `</td>` sebagai tag penutup untuk membuat sel atau kotak dalam tabel.

Line [47], [55], `if()` digunakan untuk penentuan keputusan atau conditional statement pada kondisi utama.

Line [52], `count()` digunakan untuk menghitung jumlah elemen dalam sebuah array atau objek.

Line [53], `$ukuranMatriks = $panjang * $lebar;` adalah deklarasi statis untuk rumus ukuran matriks.

Line [57], `else()` digunakan untuk penentuan keputusan atau conditional statement pada kondisi terakhir di mana semua kondisi sebelumnya tidak ada yang terpenuhi.

SOAL 2

Diberikan sebuah kumpulan data seperti berikut:

Nama	NIM	Nilai UTS	Nilai UAS
Andi	2101001	87	65
Budi	2101002	76	79
Tono	2101003	50	41
Jessica	2101004	60	75

Implementasikan data diatas menjadi multi-dimensional associative array pada Bahasa pemrograman PHP kemudian tambahkan dua kolom baru. Kolom pertama adalah hasil perhitungan nilai akhir dari nilai UTS dan UAS dengan rumus seperti berikut: 40% UTS + 60% UAS. Kolom kedua berisi nilai huruf dari nilai akhir berdasarkan ketentuan berikut:

HURUF	NILAI
A	>= 80
B	70 - 79
C	60 - 69
D	50 - 59
E	< 50

***dua kolom baru tersebut didapatkan berdasarkan baris kode logika hasil implementasi bukan dihitung manual**

Sehingga didapatkan output seperti berikut:

Nama	NIM	Nilai UTS	Nilai UAS	Nilai Akhir	Huruf
Andi	2101001	87	65	73.8	B
Budi	2101002	76	79	77.8	B
Tono	2101003	50	41	44.6	E
Jessica	2101004	60	75	69	C

Simpan dengan nama file: PRAK402.php

A. Source Code

1	<html>
2	<head>
3	<title>Data Nilai</title>
4	<style>

```

5         table{
6             border-collapse: collapse;
7         }
8         th{
9             text-align: left;
10            background: #D0CECE;
11        }
12        th, td{
13            width: 100px;
14            padding-left: 5px;
15            padding-bottom: 6px;
16        }
17        table, th, td{
18            border: 1px solid black;
19        }
20    </style>
21 </head>
22 <body>
23     <table>
24         <tr>
25             <th>Nama</th>
26             <th>NIM</th>
27             <th>Nilai UTS</th>
28             <th>Nilai UAS</th>
29             <th>Nilai Akhir</th>
30             <th>Huruf</th>
31         </tr>
32
33         <?php
34         $data=array(
35             array(
36                 "Nama" => "Andi",
37                 "NIM" => "2101001",
38                 "Nilai UTS" => 87,
39                 "Nilai UAS" => 65
40             ),
41             array(
42                 "Nama" => "Budi",
43                 "NIM" => "2101002",
44                 "Nilai UTS" => 76,
45                 "Nilai UAS" => 79
46             ),
47             array(
48                 "Nama" => "Tono",
49                 "NIM" => "2101003",
50                 "Nilai UTS" => 50,
51                 "Nilai UAS" => 41

```

```

52         ),
53         array(
54             "Nama" => "Jessica",
55             "NIM" => "2101004",
56             "Nilai UTS" => 60,
57             "Nilai UAS" => 75
58         )
59     );
60
61     function hitungNilaiHuruf($nilaiUts,
62     $nilaiUas){
63         $nilaiAkhir = (40/100 * $nilaiUts) +
64         (60/100 * $nilaiUas);
65         if($nilaiAkhir >= 80){
66             return "A";
67         }elseif($nilaiAkhir >= 70 && $nilaiAkhir
68         <= 79){
69             return "B";
70         }elseif($nilaiAkhir >= 60 && $nilaiAkhir
71         <= 69){
72             return "C";
73         }elseif($nilaiAkhir >= 50 && $nilaiAkhir
74         <= 59){
75             return "D";
76         }else{
77             return "E";
78         }
79     }
80
81     foreach($data as $key => $row){
82         $nilaiAkhir = (40/100 * $row["Nilai
83         UTS"]) + (60/100 * $row["Nilai UAS"]);
84         $Huruf = hitungNilaiHuruf($row["Nilai
85         UTS"], $row["Nilai UAS"]);
86         $data[$key]["Nilai Akhir"] =
87         $nilaiAkhir;
88         $data[$key]["Huruf"] = $Huruf;
89     }
90
91     foreach($data as $row){
92         echo "<tr>";
93         echo "<td>{$row['Nama']}</td>";
94         echo "<td>{$row['NIM']}</td>";
95         echo "<td>{$row['Nilai UTS']}</td>";
96         echo "<td>{$row['Nilai UAS']}</td>";
97         echo "<td>{$row['Nilai Akhir']}</td>";
98         echo "<td>{$row['Huruf']}</td>";

```

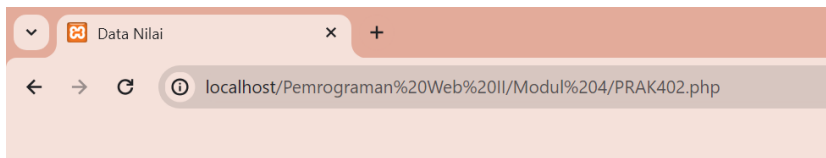
```

91         echo "</tr>";
92     }
93     ?>
94     </table>
95 </body>
96 </html>

```

Tabel 16. Source Code Soal 2

B. Output Program



Nama	NIM	Nilai UTS	Nilai UAS	Nilai Akhir	Huruf
Andi	2101001	87	65	73.8	B
Budi	2101002	76	79	77.8	B
Tono	2101003	50	41	44.6	E
Jessica	2101004	60	75	69	C

Gambar 16. Screenshot Hasil Jawaban Soal 2

C. Pembahasan

Line [1], [96], <html> digunakan sebagai tag pembuka dan </html> sebagai tag penutup dokumen HTML.

Line [2], [21], <head> digunakan sebagai tag pembuka dan </head> sebagai tag penutup bagian kepala dari dokumen HTML.

Line [3], <title> digunakan sebagai tag pembuka dan </title> sebagai tag penutup untuk membuat judul dari sebuah halaman.

Line [4], [20], <style> digunakan sebagai tag pembuka dan </style> sebagai tag penutup untuk menyisipkan kode style atau CSS ke dalam sebuah dokumen HTML.

Line [5 – 7], table{} digunakan untuk menargetkan elemen tabel di halaman web. border-collapse: collapse; digunakan untuk mengatur penggabungan batas sel tabel.

Line [8 – 11], th{} digunakan untuk menargetkan elemen sel header di halaman web. text-align: left; digunakan untuk mengatur penjumlahan teks. background: #D0CECE; digunakan untuk mengatur latar belakang.

Line [12 – 16], th, td{} digunakan untuk menargetkan elemen sel header dan sel data di halaman web. width: 100px; digunakan untuk mengatur lebar sel. padding-left:

5px; dan padding-bottom: 6px; digunakan untuk mengatur jarak sebelah kiri dan bawah dalam sel.

Line [17 – 19], `table`, `th`, `td {}` digunakan untuk menargetkan elemen tabel, sel header, dan sel data di halaman web. `border: 1px solid black;` digunakan untuk mengatur tebal dan warna border.

Line [22], [95], `<body>` digunakan sebagai tag pembuka dan `</body>` sebagai tag penutup untuk bagian tubuh atau isi dari sebuah halaman.

Line [23], [94], `<table>` digunakan sebagai tag pembuka dan `</table>` sebagai tag penutup untuk membuat tabel.

Line [24], [31], [84], [91], `<tr>` digunakan sebagai tag pembuka dan `</tr>` sebagai tag penutup untuk membuat baris dalam sebuah tabel.

Line [25 – 30], `<th>` digunakan sebagai tag pembuka dan `</th>` sebagai tag penutup untuk membuat sebuah sel header tabel.

Line [33], [93], `<?php` digunakan sebagai tag pembuka dan `?>` digunakan sebagai tag penutup program PHP.

Line [34 – 59], `$data=array()` adalah deklarasi array yang berisi data tabel.

Line [61], `function` digunakan untuk menyelesaikan tugas tertentu.

Line [62], `$nilaiAkhir = (40/100 * $nilaiUts) + (60/100 * $nilaiUas);` adalah deklarasi statis untuk rumus nilai akhir.

Line [63], `if()` digunakan untuk penentuan keputusan atau conditional statement pada kondisi utama.

Line [64], [66], [68], [70], [72], `return` digunakan untuk mengembalikan sebuah nilai.

Line [65], [67], [69], `elseif()` digunakan untuk penentuan keputusan atau conditional statement digunakan pada kondisi kedua hingga seterusnya.

Line [71], `else()` digunakan untuk penentuan keputusan atau conditional statement pada kondisi terakhir di mana semua kondisi sebelumnya tidak ada yang terpenuhi.

Line [76], [83], `foreach()` digunakan untuk perulangan khusus untuk memproses array.

Line [84 – 91], `echo` digunakan untuk menampilkan teks.

Line [85 – 90], `<td>` digunakan sebagai tag pembuka dan `</td>` sebagai tag penutup untuk membuat sel atau kotak dalam tabel.

SOAL 3

Diberikan sebuah kumpulan data seperti berikut:

No	Nama	Mata Kuliah diambil	SKS
1	Ridho	Pemrograman I	2
		Praktikum Pemrograman I	1
		Pengantar Lingkungan Lahan Basah	2
		Arsitektur Komputer	3
2	Ratna	Basis Data I	2
		Praktikum Basis data I	1
		Kalkulus	3
3	Tono	Rekayasa Perangkat Lunak	3
		Analisis dan Perancangan Sistem	3
		Komputasi Awan	3
		Kecerdasan Bisnis	3

Implementasikan data diatas menjadi multi-dimensional associative array pada Bahasa pemrograman PHP kemudian tambahkan dua kolom baru. Kolom pertama adalah total sks dari mata kuliah yang diambil mahasiswa. Kolom kedua adalah keterangan jika total sks yang diambil kurang dari 7 maka diberi keterangan “Revisi KRS”, jika tidak maka diberi keterangan “Tidak Revisi”.

***dua kolom baru tersebut didapatkan berdasarkan baris kode logika hasil implementasi bukan dihitung manual**

Cetak hasil output seperti berikut:

No	Nama	Mata Kuliah diambil	SKS	Total SKS	Keterangan
1	Ridho	Pemrograman I	2	8	Tidak Revisi
		Praktikum Pemrograman I	1		
		Pengantar Lingkungan Lahan Basah	2		
		Arsitektur Komputer	3		
2	Ratna	Basis Data I	2	6	Revisi KRS

		Praktikum Basis data I	1		
		Kalkulus	3		
3	Tono	Rekayasa Perangkat Lunak	3	12	Tidak Revisi
		Analisis dan Perancangan Sistem	3		
		Komputasi Awan	3		
		Kecerdasan Bisnis	3		

Simpan dengan nama file: PRAK403.php

A. Source Code

1	<html>
2	<head>
3	<title>Data Mata Kuliah</title>
4	<style>
5	table{
6	border-collapse: collapse;
7	}
8	th{
9	text-align: left;
10	background: #D0CECE;
11	}
12	th, td{
13	padding-left: 5px;
14	padding-bottom: 6px;
15	}
16	table, th, td{
17	border: 1px solid black;
18	}
19	.Keterangan_Tidak_Revisi{
20	background-color: #00AF50;
21	}
22	.Keterangan_Revisi_KRS{
23	background-color: #FF0000;
24	}
25	</style>
26	</head>
27	<body>
28	<table>
29	<tr>
30	<th width='30'>No</th>
31	<th width='90'>Nama</th>

		<th width='180'>Mata Kuliah
32	diambil</th>	
		<th width='45'>SKS</th>
33		<th width='85'>Total SKS</th>
34		<th width='90'>Keterangan</th>
35		</tr>
36		
37		<?php
38		\$data=array(
39		array(
40		"no" => "1",
41		"nama" => "Ridho",
42		"matkul" => array(
43		array(
44		"pilihan" => "Pemrograman
45	I",	"sks" => 2
46),
47		array(
48		"pilihan" => "Praktikum
49	Pemrograman I",	"sks" => 1
50),
51		array(
52		"pilihan" => "Pengantar
53	Lingkungan Lahan Basah",	"sks" => 2
54),
55		array(
56		"pilihan" => "Arsitektur
57	Komputer",	"sks" => 3
58)
59),
60),
61		array(
62		"no" => "2",
63		"nama" => "Ratna",
64		"matkul" => array(
65		array(
66		"pilihan" => "Basis Data
67	I",	"sks" => 2
68),
69		array(
70		"pilihan" => "Praktikum
71	Basis Data I",	

```

72         "sks" => 1
73     ),
74     array(
75         "pilihan" => "Kalkulus",
76         "sks" => 3
77     )
78 ),
79 array(
80     "no" => "3",
81     "nama" => "Tono",
82     "matkul" => array(
83         array(
84             "pilihan" => "Rekayasa
85 Perangkat Lunak",
86             "sks" => 3
87         ),
88         array(
89             "pilihan" => "Analisis dan
90 Perancangan Sistem",
91             "sks" => 3
92         ),
93         array(
94             "pilihan" => "Komputasi
95 Awan",
96             "sks" => 3
97         ),
98         array(
99             "pilihan" => "Kecerdasan
100 Bisnis",
101             "sks" => 3
102         )
103     ),
104 );
105
106 function hitungTotalSKS($matkul){
107     $totalSks = 0;
108     foreach($matkul as $matakuliah){
109         $totalSks += $matakuliah["sks"];
110     }
111     return $totalSks;
112 }
113
114 function tentukanKeterangan($totalSks){
115     return $totalSks < 7 ? "Revisi KRS" :
116 "Tidak Revisi";

```

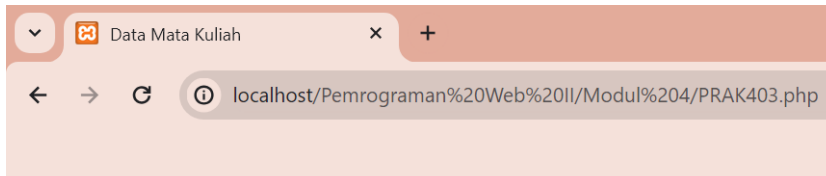
```

114         }
115         foreach($data as &$mahasiswa){
116             $mahasiswa["totalSks"] =
117             hitungTotalSKS($mahasiswa["matkul"]);
118             $mahasiswa["keterangan"] =
119             tentukanKeterangan($mahasiswa["totalSks"]);
120         }
121         unset($mahasiswa);
122         foreach($data as $mahasiswa){
123             foreach($mahasiswa["matkul"] as $index
=> $matakuliah){
124                 echo "<tr>";
125                 if($index == 0){
126                     echo
127                     "<td>".$mahasiswa["no"]."</td>";
128                     echo
129                     "<td>".$mahasiswa["nama"]."</td>";
130                     echo
131                     "<td>".$matakuliah["pilihan"]."</td>";
132                     echo
133                     "<td>".$matakuliah["sks"]."</td>";
134                     echo
135                     "<td>".$mahasiswa["totalSks"]."</td>";
136                     echo ' <td
class="'.($mahasiswa["keterangan"] == "Revisi KRS" ?
137                     "Keterangan_Revisi_KRS"
138                     :
139                     "Keterangan_Tidak_Revisi").' ">'. $mahasiswa["keterangan
140                     "]."</td>";
141                     }else{
142                         echo "<td></td>";
143                         echo "<td></td>";
144                         echo
145                         "<td>".$matakuliah["pilihan"]."</td>";
146                         echo
147                         "<td>".$matakuliah["sks"]."</td>";
148                         echo "<td></td>";
149                         echo "<td></td>";
150                     }
151                     echo "</tr>";
152             }
153         }
154     }>
155 </table>
156 </body>
157 </html>

```

Tabel 17. Source Code Soal 3

B. Output Program



No	Nama	Mata Kuliah diambil	SKS	Total SKS	Keterangan
1	Ridho	Pemrograman I	2	8	Tidak Revisi
		Praktikum Pemrograman I	1		
		Pengantar Lingkungan Lahan Basah	2		
		Arsitektur Komputer	3		
2	Ratna	Basis Data I	2	6	Revisi KRS
		Praktikum Basis Data I	1		
		Kalkulus	3		
3	Tono	Rekayasa Perangkat Lunak	3	12	Tidak Revisi
		Analisis dan Perancangan Sistem	3		
		Komputasi Awan	3		
		Kecerdasan Bisnis	3		

Gambar 17. Screenshot Hasil Jawaban Soal 3

C. Pembahasan

Line [1], [146], `<html>` digunakan sebagai tag pembuka dan `</html>` sebagai tag penutup dokumen HTML.

Line [2], [26], `<head>` digunakan sebagai tag pembuka dan `</head>` sebagai tag penutup bagian kepala dari dokumen HTML.

Line [3], `<title>` digunakan sebagai tag pembuka dan `</title>` sebagai tag penutup untuk membuat judul dari sebuah halaman.

Line [4], [25], `<style>` digunakan sebagai tag pembuka dan `</style>` sebagai tag penutup untuk menyisipkan kode style atau CSS ke dalam sebuah dokumen HTML.

Line [5 – 7], `table{}` digunakan untuk menargetkan elemen tabel di halaman web. `border-collapse: collapse;` digunakan untuk mengatur penggabungan batas sel tabel.

Line [8 – 11], `th{}` digunakan untuk menargetkan elemen sel header di halaman web. `text-align: left;` digunakan untuk mengatur penjumlahan teks. `background: #D0CECE;` digunakan untuk mengatur latar belakang.

Line [12 – 15], `th, td{}` digunakan untuk menargetkan elemen sel header dan sel data di halaman web. `padding-left: 5px;` dan `padding-bottom: 6px;` digunakan untuk mengatur jarak sebelah kiri dan bawah dalam sel.

Line [16 – 18], `table, th, td {}` digunakan untuk menargetkan elemen tabel, sel header, dan sel data di halaman web. `border: 1px solid black;` digunakan untuk mengatur tebal dan warna border.

Line [19 – 21], `.Keterangan_Tidak_Revisi{}` adalah deklarasi gaya dalam CSS pada kelas `Keterangan_Tidak_Revisi`. `background-color: #00AF50;` digunakan untuk mengatur warna latar belakang.

Line [22 – 24], `.Keterangan_Revisi_KRS{}` adalah deklarasi gaya dalam CSS pada kelas `Keterangan_Tidak_Revisi`. `background-color: #FF0000;` digunakan untuk mengatur warna latar belakang.

Line [27], [145], `<body>` digunakan sebagai tag pembuka dan `</body>` sebagai tag penutup untuk bagian tubuh atau isi dari sebuah halaman.

Line [28], [144], `<table>` digunakan sebagai tag pembuka dan `</table>` sebagai tag penutup untuk membuat tabel.

Line [29], [36], [124], [140], `<tr>` digunakan sebagai tag pembuka dan `</tr>` sebagai tag penutup untuk membuat baris dalam sebuah tabel.

Line [30 – 35], `<th>` digunakan sebagai tag pembuka dan `</th>` sebagai tag penutup untuk membuat sebuah sel header tabel. `width` digunakan untuk mengatur lebar sel.

Line [38], [143], `<?php` digunakan sebagai tag pembuka dan `?>` digunakan sebagai tag penutup program PHP.

Line [39 – 102], `$data=array()` adalah deklarasi array yang berisi data tabel.

Line [104], [112], `function` digunakan untuk menyelesaikan tugas tertentu.

Line [105], `$totalSks = 0` adalah inisialisasi variabel untuk menyatakan nilai.

Line [106], [116], [122], [123], `foreach()` digunakan untuk perulangan khusus untuk memproses array.

Line [109], [113], [68], [70], [72], `return` digunakan untuk mengembalikan sebuah nilai.

Line [120], `unset()` menghapus variabel tertentu atau elemen tertentu dari array.

Line [124], [126 – 131], [133 – 139], [141], `echo` digunakan untuk menampilkan teks.

Line [125], `if ()` digunakan untuk penentuan keputusan atau conditional statement pada kondisi utama.

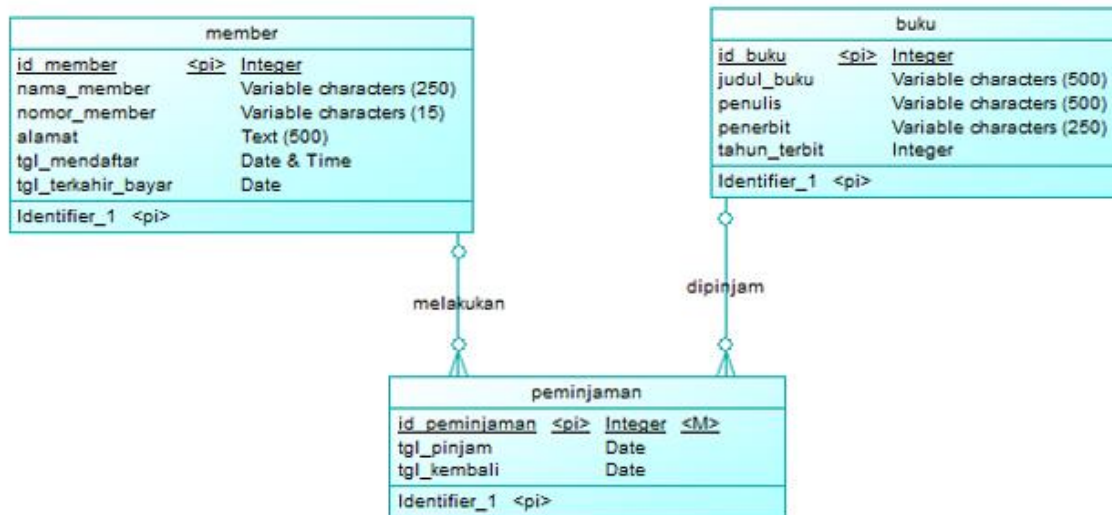
Line [126 – 131], [133 – 139], `<td>` digunakan sebagai tag pembuka dan `</td>` sebagai tag penutup untuk membuat sel atau kotak dalam tabel

Line [132], `else ()` digunakan untuk penentuan keputusan atau conditional statement pada kondisi terakhir di mana semua kondisi sebelumnya tidak ada yang terpenuhi.

MODUL 5: FUNCTION DAN DATABASE

SOAL 1

Diberikan desain konseptual sebuah basis data seperti berikut



Buatlah sebuah aplikasi berbasis web sederhana yang dapat melakukan operasi CRUD (Create, Read, Update, Delete) dari hasil implementasi desain basis data yang diberikan. Adapun ketentuan pembuatannya sebagai berikut:

1. Koneksi database dibuat menjadi satu file sendiri yaitu Koneksi.php, kemudian gunakan fungsi require ketika ingin melakukan operasi ke basis data.
2. Operasi data seperti Insert, Update, Delete, Get Data dibuat menjadi fungsi sendiri masing-masing dan disimpan di dalam satu file khusus yaitu Model.php
3. Minimal terdapat file-file berikut di dalam proyek

Nama File	Keterangan
Koneksi.php	Berisi fungsi-fungsi untuk melakukan koneksi ke basis data.
Model.php	Berisi fungsi-fungsi untuk melakukan operasi data seperti insert, update, delete, get data untuk semua tabel.
Member.php	Menampilkan semua data member dalam bentuk tabel dan terdapat tombol untuk melakukan tambah, edit, dan hapus data member.

FormMember.php	Menampilkan formulir isian untuk menambah atau mengubah data member.
Buku.php	Menampilkan semua data buku dalam bentuk tabel dan terdapat tombol untuk melakukan tambah, edit dan hapus data buku.
FormBuka.php	Menampilkan formulir isian untuk menambah atau mengubah data buku.
Peminjaman.php	Menampilkan semua data peminjaman dalam bentuk tabel dan terdapat tombol untuk melakukan tambah, edit, dan hapus data peminjaman.
FormPeminjaman.php	Menampilkan formulir isian untuk menambah atau mengubah data peminjaman.

Simpan dengan nama projek di Github: PRAK501

A. Source Code

1	<?php
2	\$host="localhost";
3	\$database="modul5";
4	\$user="root";
5	\$pass="";
6	
7	try{
8	\$koneksi = new
9	PDO("mysql:host=".\$host.";dbname=".\$database, \$user, \$pass);
10	\$koneksi->setAttribute(PDO::ATTR_ERRMODE,
11	PDO::ERRMODE_EXCEPTION);
12	} catch(PDOException \$e){
13	echo "Koneksi Gagal ".\$e->getMessage();
14	}
15	?>

Tabel 18. Source Code Soal 1 Koneksi.php

1	<?php
2	require_once 'Koneksi.php';
3	
4	function generateId(\$koneksi, \$table, \$column){

```

5      try{
6          $sql = "SELECT MAX($column) AS max_id FROM $table";
7          $stmt = $koneksi->query($sql);
8          $result = $stmt->fetch(PDO::FETCH_ASSOC);
9          $max_id = $result['max_id'];
10         if($max_id === null){
11             return 1;
12         }else{
13             return $max_id + 1;
14         }
15     }catch(PDOException $e){
16         return "Error: " . $e->getMessage();
17     }
18 }
19
20 function getId($koneksi, $table, $column, $id){
21     try{
22         $sql = "SELECT * FROM $table WHERE $column = ?";
23         $stmt = $koneksi->prepare($sql);
24         $stmt->execute([$id]);
25         return $stmt->fetch(PDO::FETCH_ASSOC);
26     }catch(PDOException $e){
27         return "Error: " . $e->getMessage();
28     }
29 }
30
31 function insertData($koneksi, $table, $data){
32     try{
33         $columns = implode(", ", array_keys($data));
34         $placeholders = implode(", ", array_fill(0,
count($data), "?"));
35         $sql = "INSERT INTO $table ($columns) VALUES
($placeholders)";
36         $stmt = $koneksi->prepare($sql);
37         $stmt->execute(array_values($data));
38         return true;
39     }catch(PDOException $e){
40         return "Error: " . $e->getMessage();
41     }
42 }
43
44 function updateData($koneksi, $table, $data, $id_column,
$id_value){
45     try{
46         $columns = implode(" = ?, ", array_keys($data)) . "
= ?";
47         $sql = "UPDATE $table SET $columns WHERE $id_column
= ?";
48         $stmt = $koneksi->prepare($sql);
49         $data_values = array_values($data);
50         $data_values[] = $id_value;
51         $stmt->execute($data_values);

```

52	return true;
53	}catch(PDOException \$e){
54	return "Error: " . \$e->getMessage();
55	}
56	}
57	
58	function deleteData(\$koneksi, \$table, \$column, \$id){
59	try{
60	\$sql = "DELETE FROM \$table WHERE \$column = ?";
61	\$stmt = \$koneksi->prepare(\$sql);
62	\$stmt->execute([\$id]);
63	return true;
64	}catch(PDOException \$e){
65	return "Error: " . \$e->getMessage();
66	}
67	}
68	
69	function getData(\$koneksi, \$table){
70	try{
71	\$sql = "SELECT * FROM \$table";
72	\$stmt = \$koneksi->query(\$sql);
73	return \$stmt->fetchAll(PDO::FETCH_ASSOC);
74	}catch(PDOException \$e){
75	return "Error: " . \$e->getMessage();
76	}
77	}
78	?>

Tabel 19. Source Code Soal 1 Model.php

1	<?php
2	require_once 'Model.php';
3	
4	\$message = '';
5	\$book = [];
6	
7	if(\$_SERVER['REQUEST_METHOD'] == 'POST'){
8	if(isset(\$_POST['menambahkan'])){
9	if(isset(\$_POST['judul_buku']) &&
	isset(\$_POST['penulis']) && isset(\$_POST['penerbit']) &&
	isset(\$_POST['tahun_terbit'])){
10	\$id_buku = generateId(\$koneksi, 'buku',
	'id_buku');
11	\$dataBuku = [
12	'id_buku' => \$id_buku,
13	'judul_buku' => \$_POST['judul_buku'],
14	'penulis' => \$_POST['penulis'],
15	'penerbit' => \$_POST['penerbit'],
16	'tahun_terbit' => \$_POST['tahun_terbit']
17];
18	\$result = insertData(\$koneksi, 'buku',
	\$dataBuku);
19	header("Location: Buku.php");

```

20         exit;
21     }
22 }
23
24     if(isset($_POST['mengubah'])){
25         if(isset($_POST['id_buku'])      &&
isset($_POST['judul_buku']) && isset($_POST['penulis']) &&
isset($_POST['penerbit']) && isset($_POST['tahun_terbit'])){
26             $dataBuku = [
27                 'judul_buku' => $_POST['judul_buku'],
28                 'penulis' => $_POST['penulis'],
29                 'penerbit' => $_POST['penerbit'],
30                 'tahun_terbit' => $_POST['tahun_terbit']
31             ];
32             $result = updateData($koneksi, 'buku', $dataBuku,
'id_buku', $_POST['id_buku']);
33             header("Location: Buku.php");
34             exit;
35         }
36     }
37 }
38
39 if(isset($_GET['id_buku'])){
40     $book_id = $_GET['id_buku'];
41     $book = getById($koneksi, 'buku', 'id_buku', $book_id);
42 }
43 ?>
44
45 <html>
46     <head>
47         <title>Form Buku</title>
48         <style>
49             body{
50                 margin: 0;
51                 padding: 0;
52                 display: flex;
53                 justify-content: center;
54                 align-items: center;
55                 background: linear-gradient(to bottom left,
#DAF0F6, #DEF0FF, #C7DEF8, #85B7F1, #73AFF6);
56                 background-size: cover;
57                 background-position: center;
58                 font-family: Arial, sans-serif;
59             }
60             .container{
61                 width: 60%;
62                 background-color: white;
63                 border-radius: 12px;
64                 box-shadow: 0 2px 4px rgba(0, 0, 0, 0.1);
65                 overflow: hidden;
66             }
67             .form-header{

```

```

68         padding: 10px 20px;
69         align-items: center;
70         background-color: #F7F7F7;
71         border-bottom: 1px solid #E0E0E0;
72     }
73     .form-header h3{
74         margin-top: 20px;
75         margin-bottom: 20px;
76         text-align: left;
77         font-size: 25px;
78         color: black;
79     }
80     .form-container{
81         padding: 20px;
82     }
83     form label{
84         margin-bottom: 10px;
85         display: block;
86         font-size: 14px;
87         font-weight: bold;
88     }
89     form input[type="text"]{
90         width: 100%;
91         margin-bottom: 20px;
92         padding: 10px;
93         border: 1px solid #CCCCCC;
94         border-radius: 5px;
95         box-sizing: border-box;
96     }
97     .button-container{
98         display: flex;
99         justify-content: space-between;
100    }
101    .button-container input[type="submit"]{
102        width: 49%;
103        margin-bottom: 0;
104        padding: 10px;
105        background-color: #73AFF6;
106        border: 1px solid #CCCCCC;
107        border-radius: 5px;
108        box-sizing: border-box;
109    }
110    </style>
111    </head>
112    <body>
113        <div class="container">
114            <div class="form-header">
115                <h3>KELOLA DATA BUKU</h3>
116            </div>
117
118            <div class="form-container">
119

```

120	<code><form method="post" action=""</code>
121	<code>autocomplete="off"></code>
122	<code><?php if(isset(\$book['id_buku'])):?></code>
123	<code><input type="hidden" name="id_buku"</code>
124	<code>value="<?php echo \$book['id_buku']; ?>"</code>
125	<code><?php endif; ?></code>
126	<code><label for="judul_buku">Judul</code>
127	<code>Buku</label></code>
128	<code><input type="text" name="judul_buku"</code>
129	<code>id="judul_buku" value="<?php echo isset(\$_GET['id_buku']) ?</code>
130	<code>\$book['judul_buku'] : ''; ?>" required>
</code>
131	<code><label for="penulis">Penulis</label></code>
132	<code><input type="text" name="penulis"</code>
133	<code>id="penulis" value="<?php echo isset(\$_GET['id_buku']) ?</code>
134	<code>\$book['penulis'] : ''; ?>" required>
</code>
135	<code><label for="penerbit">Penerbit</label></code>
136	<code><input type="text" name="penerbit"</code>
137	<code>id="penerbit" value="<?php echo isset(\$_GET['id_buku']) ?</code>
138	<code>\$book['penerbit'] : ''; ?>" required>
</code>
139	<code><label for="tahun_terbit">Tahun</code>
140	<code>Terbit</label></code>
141	<code><input type="text" name="tahun_terbit"</code>
142	<code>id="tahun_terbit" value="<?php echo isset(\$_GET['id_buku']) ?</code>
143	<code>\$book['tahun_terbit'] : ''; ?>" required>
</code>
144	<code><div class="button-container"></code>
145	<code><input type="submit"</code>
	<code>name="menambahkan" value="Tambah"></code>
	<code><input type="submit" name="mengubah"</code>
	<code>value="Update"></code>
	<code></div></code>
	<code></form></code>
	<code></div></code>
	<code></div></code>
	<code></body></code>
	<code></html></code>

Tabel 20. Source Code Soal 1 FormBuku.php

1	<code><?php</code>
2	<code>require_once 'Model.php';</code>
3	
4	<code>\$books = getData(\$koneksi, 'buku');</code>
5	
6	<code>if(isset(\$_POST['delete'])) {</code>
7	<code> \$result = deleteData(\$koneksi, 'buku', 'id_buku',</code>
8	<code> \$_POST['id_buku']);</code>
	<code> header("Location: Buku.php");</code>

```

9      exit;
10   }
11   ?>
12
13   <html>
14       <head>
15           <title>Buku</title>
16           <style>
17               body{
18                   margin: 0;
19                   padding: 0;
20                   display: flex;
21                   justify-content: center;
22                   align-items: center;
23                   background: linear-gradient(to bottom left,
#DAF0F6, #DEF0FF, #C7DEF8, #85B7F1, #73AFF6);
24                   background-size: cover;
25                   background-position: center;
26                   font-family: Arial, sans-serif;
27               }
28               .container{
29                   width: 90%;
30                   background-color: #FFFFFF;
31                   border-radius: 10px;
32                   box-shadow: 0 2px 10px rgba(0, 0, 0, 0.1);
33                   overflow: hidden;
34               }
35               .table-header{
36                   padding: 10px 20px;
37                   display: flex;
38                   justify-content: space-between;
39                   align-items: center;
40                   background-color: #F7F7F7;
41                   border-bottom: 1px solid #E0E0E0;
42               }
43               .table-header h3{
44                   margin-top: 20px;
45                   margin-bottom: 20px;
46                   text-align: left;
47                   font-size: 25px;
48                   color: black;
49               }
50               .table-header .add-button{
51                   padding: 5px 10px;
52                   background-color: #73AFF6;
53                   color: black;
54                   border: none;
55                   border-radius: 5px;
56                   font-size: 14px;
57               }
58               .table-container{
59                   padding: 20px;

```



```

60         }
61         table{
62             width: 100%;
63             border-collapse: collapse;
64         }
65         table thead{
66             background-color: #F7F7F7;
67         }
68         table th, table td{
69             padding: 10px;
70             text-align: left;
71             border-right: 1px solid white;
72             border-left: 1px solid white;
73             border-bottom: 1px solid white;
74             font-size: 14px;
75         }
76         table th{
77             background-color: #E9ECEF;
78             font-weight: bold;
79         }
80         table tbody tr:nth-child(even){
81             background-color: #F2F2F2;
82         }
83         .action-buttons{
84             display: flex;
85             gap: 10px;
86         }
87         .action-buttons a{
88             text-decoration: none;
89             color: #007BFF;
90             font-weight: bold;
91         }
92         .action-buttons form{
93             display: inline;
94         }
95         .action-buttons form button{
96             background: none;
97             border: none;
98             color: #007BFF;
99             font-weight: bold;
100        }
101        </style>
102    </head>
103    <body>
104        <div class="container">
105            <div class="table-header">
106                <h3>DATA BUKU</h3>
107
108                <button                                class="add-button"
onclick="window.location.href='FormBuku.php'">Tambah
Buku</button>
109            </div>

```

110	<div class="table-container">
111	<table>
112	<thead>
113	<tr>
114	<th width='90'>ID Buku</th>
115	<th width='170'>Judul Buku</th>
116	<th width='160'>Penulis</th>
117	<th width='170'>Penerbit</th>
118	<th width='90'>Tahun Terbit</th>
119	<th width='90'>Aksi</th>
120	</tr>
121	</thead>
122	<tbody>
123	<?php
124	foreach(\$books as \$book){
125	echo "<tr>
126	<td>{\$book['id_buku']}
127	<td>{\$book['judul_buku']}
128	<td>{\$book['penulis']}
129	<td>{\$book['penerbit']}
130	<td>{\$book['tahun_terbit']}
131	<td class='action-buttons'>
132	<a
133	href='FormBuku.php?id_buku={\$book['id_buku']}'>Edit
134	<form method='post'
135	action=''>
136	<input type='hidden'
137	name='id_buku' value='{\$book['id_buku']}'>
138	<button
139	type='submit' name='delete'>Delete</button>
140	</form>
141	</td>
142	</tr>"
143	}</tbody>
144	</table>
145	</div>
146	</body>
147	</html>

Tabel 21. Source Code Soal 1 Buku.php

1	<?php
2	require_once 'Model.php';
3	
4	\$message = '';
5	\$member = [];
6	

```

7  if($_SERVER['REQUEST_METHOD'] == 'POST'){
8      if(isset($_POST['menambahkan'])){
9          if(isset($_POST['nama_member']))                &&
isset($_POST['nomor_member']) && isset($_POST['alamat']) &&
isset($_POST['tgl_mendaftar']) &&
isset($_POST['tgl_terakhir_bayar'])){
10             $id_member = generateId($koneksi, 'member',
'id_member');
11             $dataMember = [
12                 'id_member' => $id_member,
13                 'nama_member' => $_POST['nama_member'],
14                 'nomor_member' => $_POST['nomor_member'],
15                 'alamat' => $_POST['alamat'],
16                 'tgl_mendaftar' => $_POST['tgl_mendaftar'],
17                 'tgl_terakhir_bayar' =>
$_POST['tgl_terakhir_bayar']
18             ];
19             $result = insertData($koneksi, 'member',
$dataMember);
20             header("Location: Member.php");
21             exit;
22         }
23     }
24
25     if(isset($_POST['mengubah'])){
26         if(isset($_POST['id_member']))                &&
isset($_POST['nama_member']) &&
isset($_POST['nomor_member']) && isset($_POST['alamat']) &&
isset($_POST['tgl_mendaftar']) &&
isset($_POST['tgl_terakhir_bayar'])){
27             $dataMember = [
28                 'nama_member' => $_POST['nama_member'],
29                 'nomor_member' => $_POST['nomor_member'],
30                 'alamat' => $_POST['alamat'],
31                 'tgl_mendaftar' => $_POST['tgl_mendaftar'],
32                 'tgl_terakhir_bayar' =>
$_POST['tgl_terakhir_bayar']
33             ];
34             $result = updateData($koneksi, 'member',
$dataMember, 'id_member', $_POST['id_member']);
35             header("Location: Member.php");
36             exit;
37         }
38     }
39 }
40
41 if(isset($_GET['id_member'])){
42     $member_id = $_GET['id_member'];
43     $member = getById($koneksi, 'member', 'id_member',
$member_id);
44 }
45 ?>

```

```
46
47 <html>
48   <head>
49     <title>Form Member</title>
50     <style>
51       body{
52         margin: 0;
53         padding: 0;
54         display: flex;
55         justify-content: center;
56         align-items: center;
57         background: linear-gradient(to bottom left,
#DAF0F6, #DEF0FF, #C7DEF8, #85B7F1, #73AFF6);
58         background-size: cover;
59         background-position: center;
60         font-family: Arial, sans-serif;
61       }
62       .container{
63         width: 60%;
64         background-color: white;
65         border-radius: 12px;
66         box-shadow: 0 2px 4px rgba(0, 0, 0, 0.1);
67         overflow: hidden;
68       }
69       .form-header{
70         padding: 10px 20px;
71         align-items: center;
72         background-color: #F7F7F7;
73         border-bottom: 1px solid #E0E0E0;
74       }
75       .form-header h3{
76         margin-top: 20px;
77         margin-bottom: 20px;
78         text-align: left;
79         font-size: 25px;
80         color: black;
81       }
82       .form-container{
83         padding: 20px;
84       }
85       form label{
86         margin-bottom: 10px;
87         display: block;
88         font-size: 14px;
89         font-weight: bold;
90       }
91       form input[type="text"]{
92         width: 100%;
93         margin-bottom: 20px;
94         padding: 10px;
95         border: 1px solid #CCCCCC;
96         border-radius: 5px;
```

```

97         box-sizing: border-box;
98     }
99     .button-container{
100         display: flex;
101         justify-content: space-between;
102     }
103     .button-container input[type="submit"]{
104         width: 49%;
105         margin-bottom: 0;
106         padding: 10px;
107         background-color: #73AFF6;
108         border: 1px solid #CCCCCC;
109         border-radius: 5px;
110         box-sizing: border-box;
111     }
112     </style>
113 </head>
114 <body>
115     <div class="container">
116         <div class="form-header">
117             <h3>KELOLA DATA MEMBER</h3>
118         </div>
119
120         <div class="form-container">
121             <form          method="post"          action=""
122 autocomplete="off">
123                 <?php if (isset($member['id_member'])):
124                 ?>
125                     <input          type="hidden"
126 name="id_member" value="<?php echo $member['id_member'];
127 ?>">
128                 <?php endif; ?>
129
130                 <label          for="nama_member">Nama
131 Member</label>
132                 <input type="text" name="nama_member"
133 id="nama_member" value="<?php echo
134 isset($member['nama_member']) ? $member['nama_member'] : '';
135 ?>" required><br>
136
137                 <label          for="nomor_member">Nomor
138 Member</label>
139                 <input type="text" name="nomor_member"
140 id="nomor_member" value="<?php echo
141 isset($member['nomor_member']) ? $member['nomor_member'] :
142 ''; ?>" required><br>
143
144                 <label for="alamat">Alamat</label>
145                 <input type="text" name="alamat"
146 id="alamat" value="<?php echo isset($member['alamat']) ?
147 $member['alamat'] : ''; ?>" required><br>
148

```

135	<code><label for="tgl_mendaftar">Tanggal</code>
136	<code>Mendaftar</label></code>
137	<code><input type="text" name="tgl_mendaftar"</code>
138	<code>id="tgl_mendaftar" placeholder="YYYY-MM-DD HH:MM:SS"</code>
139	<code>value="<?php echo isset(\$member['tgl_mendaftar']) ?</code>
140	<code>\$member['tgl_mendaftar'] : ''; ?>" required>
</code>
141	<code><label for="tgl_terakhir_bayar">Tanggal</code>
142	<code>Terakhir Bayar</label></code>
143	<code><input type="text"</code>
144	<code>name="tgl_terakhir_bayar" id="tgl_terakhir_bayar"</code>
145	<code>placeholder="YYYY-MM-DD" value="<?php echo</code>
146	<code>isset(\$member['tgl_terakhir_bayar']) ?</code>
147	<code>\$member['tgl_terakhir_bayar'] : ''; ?>" required>
</code>
148	<code><div class="button-container"></code>
149	<code><input type="submit"</code>
150	<code>name="menambahkan" value="Tambah"></code>
	<code><input type="submit" name="mengubah"</code>
	<code>value="Update"></code>
	<code></div></code>
	<code></form></code>
	<code></div></code>
	<code></div></code>
	<code></body></code>
	<code></html></code>

Tabel 22. Source Code Soal 1 FormMember.php

1	<code><?php</code>
2	<code>require_once 'Model.php';</code>
3	
4	<code>\$members = getData(\$koneksi, 'member');</code>
5	
6	<code>if(isset(\$_POST['delete'])) {</code>
7	<code> \$result = deleteData(\$koneksi, 'member', 'id_member',</code>
8	<code> \$_POST['id_member']);</code>
9	<code> header("Location: Member.php");</code>
10	<code> exit;</code>
11	<code>}</code>
12	<code>?></code>
13	<code><html></code>
14	<code> <head></code>
15	<code> <title>Member</title></code>
16	<code> <style></code>
17	<code> body{</code>
18	<code> margin: 0;</code>
19	<code> padding: 0;</code>
20	<code> display: flex;</code>
21	<code> justify-content: center;</code>
22	<code> align-items: center;</code>
23	<code> background: linear-gradient(to bottom left,</code>
	<code> #DAF0F6, #DEF0FF, #C7DEF8, #85B7F1, #73AFF6);</code>

```
24         background-size: cover;
25         background-position: center;
26         font-family: Arial, sans-serif;
27     }
28     .container{
29         width: 90%;
30         background-color: #FFFFFF;
31         border-radius: 10px;
32         box-shadow: 0 2px 10px rgba(0, 0, 0, 0.1);
33         overflow: hidden;
34     }
35     .table-header{
36         padding: 10px 20px;
37         display: flex;
38         justify-content: space-between;
39         align-items: center;
40         background-color: #F7F7F7;
41         border-bottom: 1px solid #E0E0E0;
42     }
43     .table-header h3{
44         margin-top: 20px;
45         margin-bottom: 20px;
46         text-align: left;
47         font-size: 25px;
48         color: black;
49     }
50     .table-header .add-button{
51         padding: 5px 10px;
52         background-color: #73AFF6;
53         color: black;
54         border: none;
55         border-radius: 5px;
56         font-size: 14px;
57     }
58     .table-container{
59         padding: 20px;
60     }
61     table{
62         width: 100%;
63         border-collapse: collapse;
64     }
65     table thead{
66         background-color: #F7F7F7;
67     }
68     table th, table td{
69         padding: 10px;
70         text-align: left;
71         border-right: 1px solid white;
72         border-left: 1px solid white;
73         border-bottom: 1px solid white;
74         font-size: 14px;
75     }
76     table th{
77         background-color: #E9ECEF;
78         font-weight: bold;
79     }
80     table tbody tr:nth-child(even){
```

```

81         background-color: #F2F2F2;
82     }
83     .action-buttons{
84         display: flex;
85         gap: 10px;
86     }
87     .action-buttons a{
88         text-decoration: none;
89         color: #007BFF;
90         font-weight: bold;
91     }
92     .action-buttons form{
93         display: inline;
94     }
95     .action-buttons form button{
96         background: none;
97         border: none;
98         color: #007BFF;
99         font-weight: bold;
100    }
101    </style>
102    </head>
103    <body>
104        <div class="container">
105            <div class="table-header">
106                <h3>DATA MEMBER</h3>
107
108                <button                                class="add-button"
onclick="window.location.href='FormMember.php'">Tambah
Member</button>
109            </div>
110            <div class="table-container">
111                <table>
112                    <thead>
113                        <tr>
114                            <th width='90'>ID Member</th>
115                            <th width='170'>Nama Member</th>
116                            <th width='105'>Nomor Member</th>
117                            <th width='170'>Alamat</th>
118                            <th width='125'>Tanggal Mendaftar</th>
119                            <th width='155'>Tanggal Terakhir
Bayar</th>
120                            <th width='90'>Aksi</th>
121                        </tr>
122                    </thead>
123                    <tbody>
124                        <?php
125                            foreach($members as $member){
126                                echo "<tr>
127                                    <td>{$member['id_member']}</td>
128                                    <td>{$member['nama_member']}</td>
129                                    <td>{$member['nomor_member']}</td>
130                                    <td>{$member['alamat']}</td>
131                                    <td>{$member['tgl_mendaftar']}</td>
132                                    <td>{$member['tgl_terakhir_bayar']}</td>

```


133	<td class='action-buttons'>
134	<a
135	href='FormMember.php?id_member={\$member['id_member']}'>Edit
136	<form method='post' action=''>
137	<input type='hidden'
138	name='id_member' value='{\$member['id_member']}'>
139	<button type='submit'
140	name='delete'>Delete</button>
141	</form>
142	</td>
143	</tr>";
144	}
145	?>
146	</tbody>
147	</table>
148	</div>
149	</div>
	</body>
	</html>

Tabel 23. Source Code Soal 1 Member.php

1	<?php
2	require_once 'Model.php';
3	
4	\$message = '';
5	\$loan = [];
6	
7	if(\$_SERVER['REQUEST_METHOD'] == 'POST'){
8	if(isset(\$_POST['menambahkan'])){
9	if(isset(\$_POST['tgl_pinjam']))
10	isset(\$_POST['tgl_kembali'])){
11	\$id_peminjaman = generateId(\$koneksi,
12	'peminjaman', 'id_peminjaman');
13	\$dataPeminjaman = [
14	'id_peminjaman' => \$id_peminjaman,
15	'tgl_pinjam' => \$_POST['tgl_pinjam'],
16	'tgl_kembali' => \$_POST['tgl_kembali']
17];
18	\$result = insertData(\$koneksi, 'peminjaman',
19	\$dataPeminjaman);
20	header("Location: Peminjaman.php");
21	exit;
22	}
23	}
24	
25	if(isset(\$_POST['mengubah'])){
26	if(isset(\$_POST['id_peminjaman']))
27	isset(\$_POST['tgl_pinjam'])
	isset(\$_POST['tgl_kembali'])){
	\$dataPeminjaman = [
	'tgl_pinjam' => \$_POST['tgl_pinjam'],
	'tgl_kembali' => \$_POST['tgl_kembali']
];

```

28         $result = updateData($koneksi, 'peminjaman',
29         $dataPeminjaman, 'id_peminjaman', $_POST['id_peminjaman']);
30         header("Location: Peminjaman.php");
31         exit;
32     }
33 }
34
35 if(isset($_GET['id_peminjaman'])) {
36     $peminjaman_id = $_GET['id_peminjaman'];
37     $loan = getById($koneksi, 'peminjaman', 'id_peminjaman',
38     $peminjaman_id);
39 }
40 ?>
41 <html>
42     <head>
43         <title>Form Peminjaman</title>
44         <style>
45             body{
46                 margin: 0;
47                 padding: 0;
48                 display: flex;
49                 justify-content: center;
50                 align-items: center;
51                 background: linear-gradient(to bottom left,
52                 #DAF0F6, #DEF0FF, #C7DEF8, #85B7F1, #73AFF6);
53                 background-size: cover;
54                 background-position: center;
55                 font-family: Arial, sans-serif;
56             }
57             .container{
58                 width: 60%;
59                 background-color: white;
60                 border-radius: 12px;
61                 box-shadow: 0 2px 4px rgba(0, 0, 0, 0.1);
62                 overflow: hidden;
63             }
64             .form-header{
65                 padding: 10px 20px;
66                 align-items: center;
67                 background-color: #F7F7F7;
68                 border-bottom: 1px solid #E0E0E0;
69             }
70             .form-header h3{
71                 margin-top: 20px;
72                 margin-bottom: 20px;
73                 text-align: left;
74                 font-size: 25px;
75                 color: black;
76             }
77             .form-container{

```

```

77         padding: 20px;
78     }
79     form label{
80         margin-bottom: 10px;
81         display: block;
82         font-size: 14px;
83         font-weight: bold;
84     }
85     form input[type="text"]{
86         width: 100%;
87         margin-bottom: 20px;
88         padding: 10px;
89         border: 1px solid #CCCCCC;
90         border-radius: 5px;
91         box-sizing: border-box;
92     }
93     .button-container{
94         display: flex;
95         justify-content: space-between;
96     }
97     .button-container input[type="submit"]{
98         width: 49%;
99         margin-bottom: 0;
100        padding: 10px;
101        background-color: #73AFF6;
102        border: 1px solid #CCCCCC;
103        border-radius: 5px;
104        box-sizing: border-box;
105    }
106    </style>
107    </head>
108    <body>
109        <div class="container">
110            <div class="form-header">
111                <h3>KELOLA DATA PEMINJAMAN</h3>
112            </div>
113            <div class="form-container">
114                <form                method="post"                action=""
115                autocomplete="off">
116                    <?php                                if
117                    (isset($loan['id_peminjaman'])): ?>
118                        <input                                type="hidden"
119                        name="id_peminjaman"                value="<?php echo
120                        $loan['id_peminjaman']; ?>"
121                        <?php endif; ?>
122                        <label                                for="tgl_pinjam">Tanggal
123                        Pinjam</label>
124                        <input type="text" name="tgl_pinjam"
125                        id="tgl_pinjam" placeholder="YYYY-MM-DD" value="<?php echo
126                        isset($loan['tgl_pinjam']) ? $loan['tgl_pinjam'] : ''; ?>"
127                        required><br>

```

121	
122	<code><label for="tgl_kembali">Tanggal</code>
123	<code>Kembali</label></code>
123	<code><input type="text" name="tgl_kembali"</code>
	<code>id="tgl_kembali" placeholder="YYYY-MM-DD" value="<?php echo</code>
	<code>isset(\$loan['tgl_kembali']) ? \$loan['tgl_kembali'] : ''; ?>"</code>
	<code>required>
</code>
124	<code><div class="button-container"></code>
125	<code><input type="submit"</code>
126	<code>name="menambahkan" value="Tambah"></code>
127	<code><input type="submit" name="mengubah"</code>
128	<code>value="Update"></code>
	<code></div></code>
129	<code></form></code>
130	<code></div></code>
131	<code></div></code>
132	<code></body></code>
133	<code></html></code>
134	

Tabel 24. Source Code Soal 1 FormPeminjaman.php

1	<code><?php</code>
2	<code>require_once 'Model.php';</code>
3	
4	<code>\$loans = getData(\$koneksi, 'peminjaman');</code>
5	
6	<code>if(isset(\$_POST['delete'])){</code>
7	<code>\$result = deleteData(\$koneksi, 'peminjaman', 'id_peminjaman',</code>
	<code>\$_POST['id_peminjaman']);</code>
8	<code>header("Location: Peminjaman.php");</code>
9	<code>exit;</code>
10	<code>}</code>
11	<code>?></code>
12	
13	<code><html></code>
14	<code><head></code>
15	<code><title>Data Peminjaman</title></code>
16	<code><style></code>
17	<code>body{</code>
18	<code>margin: 0;</code>
19	<code>padding: 0;</code>
20	<code>display: flex;</code>
21	<code>justify-content: center;</code>
22	<code>align-items: center;</code>
23	<code>background: linear-gradient(to bottom left, #DAF0F6,</code>
	<code>#DEF0FF, #C7DEF8, #85B7F1, #73AFF6);</code>
24	<code>background-size: cover;</code>
25	<code>background-position: center;</code>
26	<code>font-family: Arial, sans-serif;</code>
27	<code>}</code>
28	<code>.container{</code>
29	<code>width: 90%;</code>
30	<code>background-color: #FFFFFF;</code>
31	<code>border-radius: 10px;</code>
32	<code>box-shadow: 0 2px 10px rgba(0, 0, 0, 0.1);</code>
33	<code>overflow: hidden;</code>

34	}
35	.table-header{
36	padding: 10px 20px;
37	display: flex;
38	justify-content: space-between;
39	align-items: center;
40	background-color: #F7F7F7;
41	border-bottom: 1px solid #E0E0E0;
42	}
43	.table-header h3{
44	margin-top: 20px;
45	margin-bottom: 20px;
46	text-align: left;
47	font-size: 25px;
48	color: black;
49	}
50	.table-header .add-button{
51	padding: 5px 10px;
52	background-color: #73AFF6;
53	color: black;
54	border: none;
55	border-radius: 5px;
56	font-size: 14px;
57	}
58	.table-container{
59	padding: 20px;
60	}
61	table{
62	width: 100%;
63	border-collapse: collapse;
64	}
65	table thead{
66	background-color: #F7F7F7;
67	}
68	table th, table td{
69	padding: 10px;
70	text-align: left;
71	border-right: 1px solid white;
72	border-left: 1px solid white;
73	border-bottom: 1px solid white;
74	font-size: 14px;
75	}
76	table th{
77	background-color: #E9ECEF;
78	font-weight: bold;
79	}
80	table tbody tr:nth-child(even){
81	background-color: #F2F2F2;
82	}
83	.action-buttons{
84	display: flex;
85	gap: 10px;
86	}
87	.action-buttons a{
88	text-decoration: none;
89	color: #007BFF;
90	font-weight: bold;
91	}
92	.action-buttons form{
93	display: inline;
94	}
95	.action-buttons form button{
96	background: none;

97	border: none;
98	color: #007BFF;
99	font-weight: bold;
100	cursor: pointer;
101	}
102	</style>
103	</head>
104	<body>
105	<div class="container">
106	<div class="table-header">
107	<h3>DATA PEMINJAMAN</h3>
108	
109	<button class="add-button"
	onclick="window.location.href='FormPeminjaman.php'">Tambah
	Pinjaman</button>
110	</div>
111	
112	<div class="table-container">
113	<table>
114	<thead>
115	<tr>
116	<th>ID Peminjaman</th>
117	<th>Tanggal Pinjam</th>
118	<th>Tanggal Kembali</th>
119	<th width='90'>Aksi</th>
120	</tr>
121	</thead>
122	<tbody>
123	<?php
124	foreach(\$loans as \$loan){
125	echo "<tr>
126	<td>{\$loan['id_peminjaman']}</td>
127	<td>{\$loan['tgl_pinjam']}</td>
128	<td>{\$loan['tgl_kembali']}</td>
129	<td class='action-buttons'>
130	Edit
131	<form method='post' action=''>
132	<input type='hidden'
	name='id_peminjaman' value='{\$loan['id_peminjaman']}'>
133	
134	<button type='submit'
	name='delete'>Delete</button>
135	</form>
136	</td>
137	</tr>";
138	}
139	?>
140	</tbody>
141	</table>
142	</div>
143	</div>
144	</body>
145	</html>

Tabel 25. Source Code Soal 1 Peminjaman.php

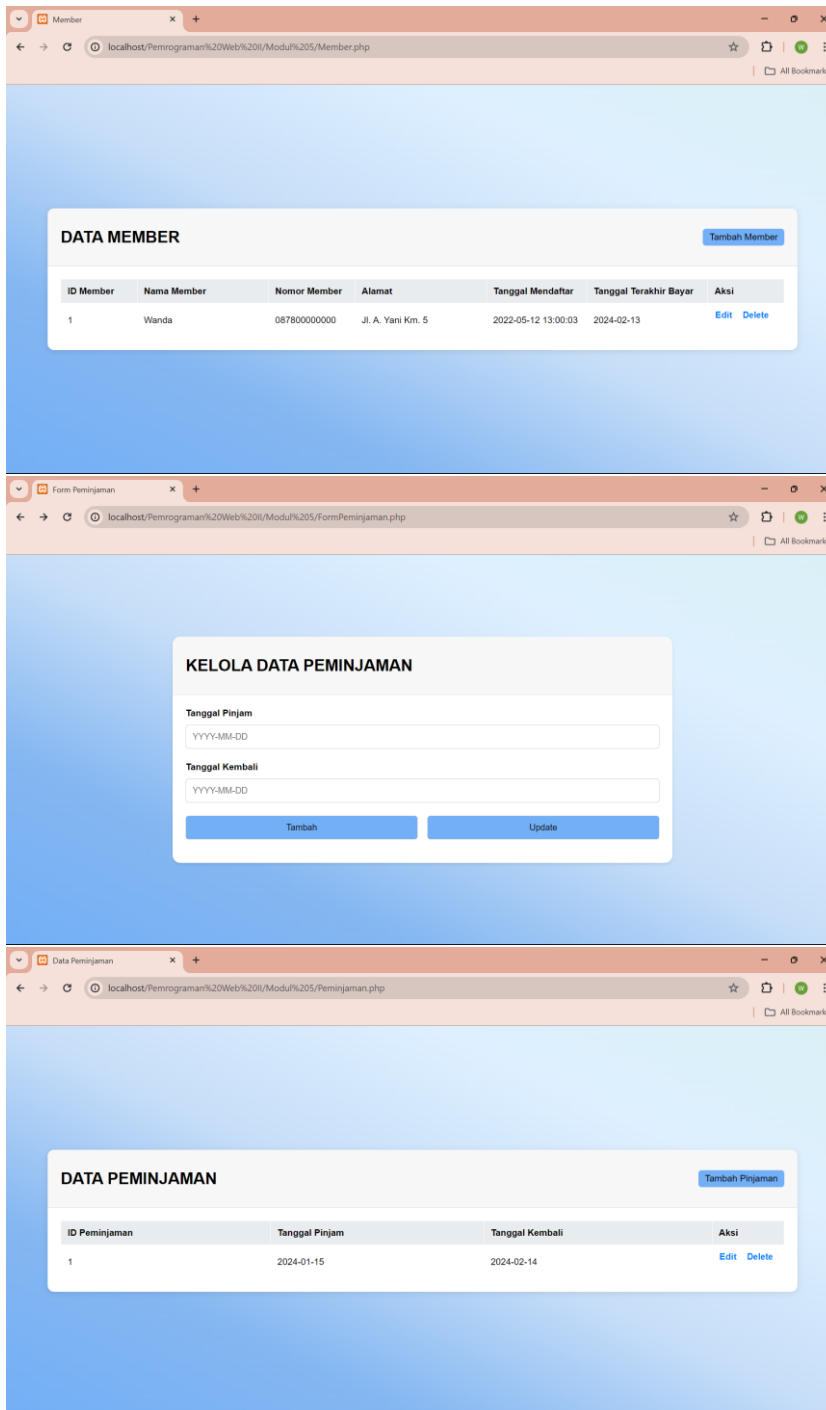
B. Output Program

The first screenshot shows the 'KELOLA DATA BUKU' (Manage Book Data) form. It includes input fields for 'Judul Buku' (Book Title), 'Penulis' (Author), 'Penerbit' (Publisher), and 'Tahun Terbit' (Year Published). At the bottom are 'Tambah' (Add) and 'Update' buttons.

The second screenshot shows the 'DATA BUKU' (Book Data) table. It has a 'Tambah Buku' (Add Book) button in the top right corner. The table contains one row of data:

ID Buku	Judul Buku	Penulis	Penerbit	Tahun Terbit	Aksi
1	Greatest Delight	Salma Hadid	PT. Bukune Kreatif Cipta	2021	Edit Delete

The third screenshot shows the 'KELOLA DATA MEMBER' (Manage Member Data) form. It includes input fields for 'Nama Member' (Member Name), 'Nomor Member' (Member Number), 'Alamat' (Address), 'Tanggal Mendaftar' (Registration Date) with a YYYY-MM-DD HH:MM:SS format, and 'Tanggal Terakhir Bayar' (Last Payment Date) with a YYYY-MM-DD format. At the bottom are 'Tambah' (Add) and 'Update' buttons.



Gambar 18. Screenshot Hasil Jawaban Soal 1

C. Pembahasan

- Koneksi.php

Line [1], [13], <?php digunakan sebagai tag pembuka dan ?> digunakan sebagai tag penutup program PHP.

Line [2 – 5], `$host="localhost"; $database="modul5"; $user="root"; $pass=""`; adalah inisialisasi variabel untuk memberikan nilai awal.

Line [7], `try{}` digunakan untuk menentukan blok kode yang akan diuji kesalahannya saat sedang dieksekusi.

Line [8], `new PDO()` digunakan untuk membuat objek baru dari kelas PHP data Objects.

Line [9], `setAttribute()` digunakan untuk mengatur atribut tertentu. `PDO::ATTR_ERRMODE` digunakan untuk mengatur mode kesalahan. `PDO::ERRMODE_EXCEPTION` digunakan untuk mengatur mode kesalahan menjadi pengecualian.

Line [10], `catch{}` digunakan untuk menentukan blok kode yang akan dieksekusi jika terjadi kesalahan pada blok kode yang diuji.

Line [11], `echo` digunakan untuk menampilkan teks. `
` digunakan untuk menyisipkan baris baru. `getMessage()` digunakan untuk mendapatkan pesan kesalahan dari pengecualian.

- **Model.php**

Line [1], [78], `<?php` digunakan sebagai tag pembuka dan `?>` digunakan sebagai tag penutup program PHP.

Line [2], `require_once` digunakan untuk menyertakan dan menjalankan file PHP hanya satu kali selama skrip.

Line [3], [20], [31], [44], [58], [69], `function` digunakan untuk menyelesaikan tugas tertentu.

Line [4], [21], [32], [45], [59], [70], `try{}` digunakan untuk menentukan blok kode yang akan diuji kesalahannya saat sedang dieksekusi.

Line [6], [22], [35], [47], [60], [71], `$sql = "SELECT MAX($column) AS max_id FROM $table"`; adalah inisialisasi variabel untuk menyimpan sebuah string SQL yang akan dieksekusi.

Line [7], [72], `query()` digunakan untuk mengeksekusi pernyataan SQL.

Line [8], [25], `fetch()` digunakan untuk mengambil baris hasil dari query yang dieksekusi,

Line [8], [25], [73], `PDO::FETCH_ASSOC` digunakan untuk mengembalikan baris hasil sebagai array asosiatif.

Line [9], `$max_id = $result['max_id'];` adalah inisialisasi variabel untuk mengambil nilai.

Line [10], `if ()` digunakan untuk penentuan keputusan atau conditional statement pada kondisi utama.

Line [11], [13], [16], [25], [27], [38], [40], [52], [54], [63], [65], [73], [75], `return` digunakan untuk mengembalikan sebuah nilai.

Line [12], `else ()` digunakan untuk penentuan keputusan atau conditional statement pada kondisi terakhir di mana semua kondisi sebelumnya tidak ada yang terpenuhi.

Line [15], [26], [39], [53], [64], [74], `catch {}` digunakan untuk menentukan blok kode yang akan dieksekusi jika terjadi kesalahan pada blok kode yang diuji.

Line [16], [27], [40], [54], [65], [75], `getMessage ()` digunakan untuk mendapatkan pesan kesalahan dari pengecualian.

Line [23], [36], [48], [61], `prepare ()` digunakan untuk menyiapkan pernyataan SQL sebelum dieksekusi.

Line [24], [37], [51], [62], `execute ()` digunakan untuk mengeksekusi pernyataan SQL yang telah disiapkan.

Line [33], [34], [46], `implode ()` digunakan untuk menggabungkan elemen-elemen dari sebuah array.

Line [33], [46], `array_keys ()` digunakan untuk mengambil semua kunci dari sebuah array asosiatif dan mengembalikan dalam bentuk array baru.

Line [34], `array_fill ()` digunakan untuk membuat sebuah array dengan menetapkan nilai yang sam untuk semua elemennya. `count ()` menghitung jumlah elemen dalam sebuah array atau objek.

Line [37], [49], `array_values ()` digunakan untuk mengambil semua nilai dari sebuah array asosiatif dan mengembalikan dalam bentuk array baru.

Line [50], `$data_values = $id_value;` adalah inisialisasi variabel untuk memberikan nilai.

Line [73], `fetchAll ()` digunakan untuk mengambil semua baris dari query yang dieksekusi.

- **FormBuku.php**

Line [1], [43], [120 – 122], [125], [127], [131], [134], `<?php` digunakan sebagai tag pembuka dan `?>` digunakan sebagai tag penutup program PHP.

Line [2], `require_once` digunakan untuk menyertakan dan menjalankan file PHP hanya satu kali selama skrip.

Line [4], [5], `$message = ''`; `$book = []`; adalah inisialisasi untuk memberikan nilai awal.

Line [7 – 9], [24], [25], [39], [120], `if()` digunakan untuk penentuan keputusan atau conditional statement pada kondisi utama.

Line [8], [9], [24], [25], [39], [121], [125], [128], [131], [134], `isset()` digunakan untuk memeriksa apakah suatu variabel telah di atur.

Line [10], [18], [32], [41], `$id_buku = generateId($koneksi, 'buku', 'id_buku')`; `$result = insertData($koneksi, 'buku', $dataBuku)`; `$result = updateData($koneksi, 'buku', $dataBuku, 'id_buku', $_POST['id_buku'])`; `$book = getById($koneksi, 'buku', 'id_buku', $book_id)`; adalah inisialisasi untuk mengambil nilai.

Line [11 – 17], [26 – 31] `$dataBuku = []` adalah inisialisasi untuk menyimpan nilai.

Line [19], [33], `header()` digunakan untuk mengarahkan pengguna ke halaman tertentu.

Line [20], [34], `exit` digunakan untuk menghentikan eksekusi skrip PHP.

Line [40], `$book_id = $_GET['id_buku']` adalah inisialisasi untuk mendapatkan nilai.

Line [45], [145], `<html>` digunakan sebagai tag pembuka dan `</html>` sebagai tag penutup dokumen HTML.

Line [46], [111], `<head>` digunakan sebagai tag pembuka dan `</head>` sebagai tag penutup bagian kepala dari dokumen HTML.

Line [47], `<title>` digunakan sebagai tag pembuka dan `</title>` sebagai tag penutup untuk membuat judul dari sebuah halaman.

Line [48], [110], `<style>` digunakan sebagai tag pembuka dan `</style>` sebagai tag penutup untuk menyisipkan kode style atau CSS ke dalam sebuah dokumen HTML.

Line [50 – 59], `body{ }` digunakan untuk menargetkan elemen body di halaman web.

Line [60 – 66], `.container{ }` digunakan untuk menargetkan elemen container di halaman web.

Line [67 – 72], `.form-header{ }` digunakan untuk menargetkan elemen form header di halaman web.

Line [73 – 79], `.form-header h3{ }` digunakan untuk menargetkan elemen form header h3 di halaman web.

Line [80 – 82], `.form-container{ }` digunakan untuk menargetkan elemen form container di halaman web.

Line [83 – 88], `form label{}` digunakan untuk menargetkan elemen form label di halaman web.

Line [89 – 96], `form input{}` digunakan untuk menargetkan elemen form input di halaman web.

Line [97 – 100], `button-container{}` digunakan untuk menargetkan elemen button container di halaman web.

Line [101 – 109], `button-container input{}` digunakan untuk menargetkan elemen button container input di halaman web.

Line [103], [144], `<body>` digunakan sebagai tag pembuka dan `</body>` sebagai tag penutup untuk bagian tubuh atau isi dari sebuah halaman.

Line [113], [114], [116], [118], [136], [140], [142], [143], `<div>` digunakan sebagai tag pembuka dan `</div>` sebagai tag penutup untuk membuat sebuah blok atau kotak. `class` adalah atribut untuk menerapkan gaya CSS dari suatu kelas.

Line [115], `<h3>` digunakan sebagai tag pembuka dan `</h3>` sebagai tag penutup untuk membuat judul dalam halaman dengan ukuran heading 3.

Line [119], [141], `<form>` digunakan sebagai tag pembuka dan `</form>` sebagai tag penutup untuk membuat sebuah form. `method` adalah atribut untuk mengatur metode pengiriman data, `action` adalah atribut untuk menentukan aksi yang akan dilakukan saat data dikirim dan `autocomplete` adalah atribut untuk menampilkan opsi atau saran inputan sebelumnya yang telah dimasukkan.

Line [121], [125], [128], [131], [134], [137], [139], `<input>` digunakan untuk membuat kontrol input. `type` adalah atribut untuk menentukan jenis kontrol input, `name` adalah atribut untuk memberi nama kontrol input tersebut, `id` adalah atribut untuk mendefinisikan id dan `value` adalah atribut untuk menetapkan nilai awal input dan mendefinisikan teks pada tombol.

Line [121], [125], [128], [131], [134], `echo` digunakan untuk menampilkan teks.

Line [124], [127], [130], [133], `<label>` digunakan sebagai tag pembuka dan `</label>` sebagai tag penutup untuk pelengkap keterangan elemen form. `for` adalah atribut untuk menghubungkan dengan elemen form.

Line [125], [128], [131], [134], `required` digunakan untuk menandai bahwa input harus diisi. `
` digunakan untuk menyisipkan baris baru.

Line [122], `endif` digunakan untuk menandai akhir dari sebuah kondisional if.

- **Buku.php**

Line [1], [11], [123], [141], `<?php` digunakan sebagai tag pembuka dan `?>` digunakan sebagai tag penutup program PHP.

Line [2], `require_once` digunakan untuk menyertakan dan menjalankan file PHP hanya satu kali selama skrip.

Line [4], `$books = getData($koneksi, 'buku');` `$result = deleteData($koneksi, 'buku', 'id_buku', $_POST['id_buku']);` adalah inisialisasi untuk mengambil nilai.

Line [6], `if()` digunakan untuk penentuan keputusan atau conditional statement pada kondisi utama. `isset()` digunakan untuk memeriksa apakah suatu variabel telah di atur.

Line [7], `header()` digunakan untuk mengarahkan pengguna ke halaman tertentu.

Line [8], `exit` digunakan untuk menghentikan eksekusi skrip PHP.

Line [13], [147], `<html>` digunakan sebagai tag pembuka dan `</html>` sebagai tag penutup dokumen HTML.

Line [14], [102], `<head>` digunakan sebagai tag pembuka dan `</head>` sebagai tag penutup bagian kepala dari dokumen HTML.

Line [15], `<title>` digunakan sebagai tag pembuka dan `</title>` sebagai tag penutup untuk membuat judul dari sebuah halaman.

Line [16], [101], `<style>` digunakan sebagai tag pembuka dan `</style>` sebagai tag penutup untuk menyisipkan kode style atau CSS ke dalam sebuah dokumen HTML.

Line [17 – 27], `body{ }` digunakan untuk menargetkan elemen body di halaman web.

Line [28 – 34], `.container{ }` digunakan untuk menargetkan elemen container di halaman web.

Line [35 – 42], `.table-header{ }` digunakan untuk menargetkan elemen tabel header di halaman web.

Line [43 – 49], `.table-header h3{ }` digunakan untuk menargetkan elemen tabel header h3 di halaman web.

Line [50 – 57], `.table-header .add-button{ }` digunakan untuk menargetkan elemen tabel header add button di halaman web.

Line [58 – 60], `.table-container{ }` digunakan untuk menargetkan elemen tabel container di halaman web.

Line [61 – 64], `table{ }` digunakan untuk menargetkan elemen tabel di halaman web.

Line [65 – 67], `table thead{}` digunakan untuk menargetkan elemen tabel thead di halaman web.

Line [68 – 75], `table th`, `table td{}` digunakan untuk menargetkan elemen tabel sel header dan tabel sel data di halaman web.

Line [76 – 79], `table th{}` digunakan untuk menargetkan elemen tabel sel header di halaman web.

Line [80 – 82], `table tbody tr:nth-child(even){}` digunakan untuk menargetkan elemen tabel tbody dan tabel baris genap di halaman web.

Line [83 – 86], `.action-buttons{}` digunakan untuk menargetkan elemen action buttons di halaman web.

Line [87 – 91], `.action-buttons a{}` digunakan untuk menargetkan elemen action buttons a di halaman web.

Line [92 – 94], `.action-buttons form{}` digunakan untuk menargetkan elemen action buttons form di halaman web.

Line [95 – 100], `.action-buttons form button{}` digunakan untuk menargetkan elemen action buttons form button di halaman web.

Line [103], [146], `<body>` digunakan sebagai tag pembuka dan `</body>` sebagai tag penutup untuk bagian tubuh atau isi dari sebuah halaman.

Line [104], [105], [109], [110], [144], [145], `<div>` digunakan sebagai tag pembuka dan `</div>` sebagai tag penutup untuk membuat sebuah blok atau kotak. `class` adalah atribut untuk menerapkan gaya CSS dari suatu kelas.

Line [106], `<h3>` digunakan sebagai tag pembuka dan `</h3>` sebagai tag penutup untuk membuat judul dalam halaman dengan ukuran heading 3.

Line [108], [136], `<button>` digunakan sebagai tag pembuka dan `</button>` sebagai tag penutup untuk membuat sebuah tombol. `class` adalah atribut untuk menerapkan gaya CSS dari suatu kelas. `onclick` adalah atribut untuk menentukan file yang akan dijalankan ketika tombol diklik. `type` adalah atribut untuk menentukan jenis kontrol input, `name` adalah atribut untuk memberi nama kontrol input tersebut

Line [111], [143], `<table>` digunakan sebagai tag pembuka dan `</table>` sebagai tag penutup untuk membuat tabel.

Line [112], [121], `<thead>` digunakan sebagai tag pembuka dan `</thead>` sebagai tag penutup untuk mengelompokkan baris-baris header dalam tabel.

Line [113], [120], [125], [140], `<tr>` digunakan sebagai tag pembuka dan `</tr>` sebagai tag penutup untuk membuat baris dalam sebuah tabel.

Line [114 – 119], `<th>` digunakan sebagai tag pembuka dan `</th>` sebagai tag penutup untuk membuat sebuah sel header tabel. `width` digunakan untuk mengatur lebar sel.

Line [122], [142], `<tbody>` digunakan sebagai tag pembuka dan `</tbody>` sebagai tag penutup untuk mengelompokkan baris-baris data utama dalam tabel.

Line [124], `foreach()` digunakan untuk perulangan khusus untuk memproses array.

Line [125], `echo` digunakan untuk menampilkan teks.

Line [126 – 131], [139], `<td>` digunakan sebagai tag pembuka dan `</td>` sebagai tag penutup untuk membuat sel atau kotak dalam tabel. `class` adalah atribut untuk menerapkan gaya CSS dari suatu kelas.

Line [132], `<a>` adalah tag pembuka dan `` adalah tag penutup untuk membuat hyperlink. `href` adalah atribut untuk menentukan alamat tujuan yang dituju ketika mengklik hyperlink.

Line [133], [138], `<form>` digunakan sebagai tag pembuka dan `</form>` sebagai tag penutup untuk membuat sebuah form. `method` adalah atribut untuk mengatur metode pengiriman data dan `action` adalah atribut untuk menentukan aksi yang akan dilakukan saat data dikirim.

Line [134], `<input>` digunakan untuk membuat kontrol input. `type` adalah atribut untuk menentukan jenis kontrol input, `name` adalah atribut untuk memberi nama kontrol input tersebut dan `value` adalah atribut untuk menetapkan nilai awal input dan mendefinisikan teks pada tombol.

- **FormMember.php**

Line [1], [45], [122 – 124], [127], [130], [133], [136], `<?php` digunakan sebagai tag pembuka dan `?>` digunakan sebagai tag penutup program PHP.

Line [2], `require_once` digunakan untuk menyertakan dan menjalankan file PHP hanya satu kali selama skrip.

Line [4], [5], `$message = ''`; `$member = []`; adalah inisialisasi untuk memberikan nilai awal.

Line [7 – 9], [25], [26], [41], [122], `if()` digunakan untuk penentuan keputusan atau conditional statement pada kondisi utama.

Line [8], [9], [25], [26], [41], [122], [127], [130], [133], [136], [138], `isset()` digunakan untuk memeriksa apakah suatu variabel telah di atur.

Line [10], [18], [32], [41], `$id_member = generateId($koneksi, 'member', 'id_member')`; `$result = insertData($koneksi, 'member',`

`$dataMember); $result = updateData($koneksi, 'member', $dataMember, 'id_member', $_POST['id_member']); $member = getById($koneksi, 'member', 'id_member', $member_id);` adalah inisialisasi untuk mengambil nilai.

Line [11 – 18], [27 – 34] `$dataBuku = []` adalah inisialisasi untuk menyimpan nilai.

Line [20], [35], `header()` digunakan untuk mengarahkan pengguna ke halaman tertentu.

Line [21], [35], `exit` digunakan untuk menghentikan eksekusi skrip PHP.

Line [42], `$member_id = $_GET['id_member']` adalah inisialisasi untuk mendapatkan nilai.

Line [47], [150], `<html>` digunakan sebagai tag pembuka dan `</html>` sebagai tag penutup dokumen HTML.

Line [48], [113], `<head>` digunakan sebagai tag pembuka dan `</head>` sebagai tag penutup bagian kepala dari dokumen HTML.

Line [49], `<title>` digunakan sebagai tag pembuka dan `</title>` sebagai tag penutup untuk membuat judul dari sebuah halaman.

Line [50], [112], `<style>` digunakan sebagai tag pembuka dan `</style>` sebagai tag penutup untuk menyisipkan kode style atau CSS ke dalam sebuah dokumen HTML.

Line [51 – 61], `body{}` digunakan untuk menargetkan elemen body di halaman web.

Line [62 – 68], `.container{}` digunakan untuk menargetkan elemen container di halaman web.

Line [69 – 74], `.form-header{}` digunakan untuk menargetkan elemen form header di halaman web.

Line [75 – 81], `.form-header h3{}` digunakan untuk menargetkan elemen form header h3 di halaman web.

Line [82 – 84], `.form-container{}` digunakan untuk menargetkan elemen form container di halaman web.

Line [85 – 90], `form label{}` digunakan untuk menargetkan elemen form label di halaman web.

Line [91 – 98], `form input{}` digunakan untuk menargetkan elemen form input di halaman web.

Line [99 – 102], `button-container{}` digunakan untuk menargetkan elemen button container di halaman web.

Line [103 – 111], `button-container input{}` digunakan untuk menargetkan elemen button container input di halaman web.

Line [114], [149], `<body>` digunakan sebagai tag pembuka dan `</body>` sebagai tag penutup untuk bagian tubuh atau isi dari sebuah halaman.

Line [115], [116], [118], [120], [141], [145], [147], [148], `<div>` digunakan sebagai tag pembuka dan `</div>` sebagai tag penutup untuk membuat sebuah blok atau kotak. `class` adalah atribut untuk menerapkan gaya CSS dari suatu kelas.

Line [117], `<h3>` digunakan sebagai tag pembuka dan `</h3>` sebagai tag penutup untuk membuat judul dalam halaman dengan ukuran heading 3.

Line [121], [146], `<form>` digunakan sebagai tag pembuka dan `</form>` sebagai tag penutup untuk membuat sebuah form. `method` adalah atribut untuk mengatur metode pengiriman data, `action` adalah atribut untuk menentukan aksi yang akan dilakukan saat data dikirim dan `autocomplete` adalah atribut untuk menampilkan opsi atau saran inputan sebelumnya yang telah dimasukkan.

Line [123], [127], [130], [133], [136], [139], [142], [144], `<input>` digunakan untuk membuat kontrol input. `type` adalah atribut untuk menentukan jenis kontrol input, `name` adalah atribut untuk memberi nama kontrol input tersebut, `id` adalah atribut untuk mendefinisikan id, `placeholder` adalah atribut untuk menyediakan petunjuk dan `value` adalah atribut untuk menetapkan nilai awal input dan mendefinisikan teks pada tombol.

Line [123], [127], [130], [133], [136], [139], `echo` digunakan untuk menampilkan teks.

Line [124], [127], [131], [133], `<label>` digunakan sebagai tag pembuka dan `</label>` sebagai tag penutup untuk pelengkap keterangan elemen form. `for` adalah atribut untuk menghubungkan dengan elemen form.

Line [127], [130], [133], [136], [139], `required` digunakan untuk menandai bahwa input harus diisi. `
` digunakan untuk menyisipkan baris baru.

Line [124], `endif` digunakan untuk menandai akhir dari sebuah kondisional if.

- **Member.php**

Line [1], [11], [125], [144], `<?php` digunakan sebagai tag pembuka dan `?>` digunakan sebagai tag penutup program PHP.

Line [2], `require_once` digunakan untuk menyertakan dan menjalankan file PHP hanya satu kali selama skrip.

Line [4], [7] `$members = getData($koneksi, 'member'); $result = deleteData($koneksi, 'member', 'id_member', $_POST['id_member']);` adalah inisialisasi untuk mengambil nilai.

Line [6], `if()` digunakan untuk penentuan keputusan atau conditional statement pada kondisi utama. `isset()` digunakan untuk memeriksa apakah suatu variabel telah di atur.

Line [8], `header()` digunakan untuk mengarahkan pengguna ke halaman tertentu.

Line [9], `exit` digunakan untuk menghentikan eksekusi skrip PHP.

Line [13], [149], `<html>` digunakan sebagai tag pembuka dan `</html>` sebagai tag penutup dokumen HTML.

Line [14], [102], `<head>` digunakan sebagai tag pembuka dan `</head>` sebagai tag penutup bagian kepala dari dokumen HTML.

Line [15], `<title>` digunakan sebagai tag pembuka dan `</title>` sebagai tag penutup untuk membuat judul dari sebuah halaman.

Line [16], [101], `<style>` digunakan sebagai tag pembuka dan `</style>` sebagai tag penutup untuk menyisipkan kode style atau CSS ke dalam sebuah dokumen HTML.

Line [17 – 27], `body{}` digunakan untuk menargetkan elemen body di halaman web.

Line [28 – 34], `.container{}` digunakan untuk menargetkan elemen container di halaman web.

Line [35 – 42], `.table-header{}` digunakan untuk menargetkan elemen tabel header di halaman web.

Line [43 – 49], `.table-header h3{}` digunakan untuk menargetkan elemen tabel header h3 di halaman web.

Line [50 – 57], `.table-header .add-button{}` digunakan untuk menargetkan elemen tabel header add button di halaman web.

Line [58 – 60], `.table-container{}` digunakan untuk menargetkan elemen tabel container di halaman web.

Line [61 – 64], `table{}` digunakan untuk menargetkan elemen tabel di halaman web.

Line [65 – 67], `table thead{}` digunakan untuk menargetkan elemen tabel thead di halaman web.

Line [68 – 75], `table th, table td{}` digunakan untuk menargetkan elemen tabel sel header dan tabel sel data di halaman web.

Line [76 – 79], `table th{}` digunakan untuk menargetkan elemen tabel sel header di halaman web.

Line [80 – 82], `table tbody tr:nth-child(even){}` digunakan untuk menargetkan elemen tabel tbody dan tabel baris genap di halaman web.

Line [83 – 86], `.action-buttons{}` digunakan untuk menargetkan elemen action buttons di halaman web.

Line [87 – 91], `.action-buttons a{}` digunakan untuk menargetkan elemen action buttons a di halaman web.

Line [92 – 94], `.action-buttons form{}` digunakan untuk menargetkan elemen action buttons form di halaman web.

Line [95 – 100], `.action-buttons form button{}` digunakan untuk menargetkan elemen action buttons form button di halaman web.

Line [103], [148], `<body>` digunakan sebagai tag pembuka dan `</body>` sebagai tag penutup untuk bagian tubuh atau isi dari sebuah halaman.

Line [104], [105], [109], [110], [146], [147], `<div>` digunakan sebagai tag pembuka dan `</div>` sebagai tag penutup untuk membuat sebuah blok atau kotak. `class` adalah atribut untuk menerapkan gaya CSS dari suatu kelas.

Line [106], `<h3>` digunakan sebagai tag pembuka dan `</h3>` sebagai tag penutup untuk membuat judul dalam halaman dengan ukuran heading 3.

Line [108], [138], `<button>` digunakan sebagai tag pembuka dan `</button>` sebagai tag penutup untuk membuat sebuah tombol. `class` adalah atribut untuk menerapkan gaya CSS dari suatu kelas. `onclick` adalah atribut untuk menentukan file yang akan dijalankan ketika tombol diklik. `type` adalah atribut untuk menentukan jenis kontrol input, `name` adalah atribut untuk memberi nama kontrol input tersebut

Line [111], [145], `<table>` digunakan sebagai tag pembuka dan `</table>` sebagai tag penutup untuk membuat tabel.

Line [112], [122], `<thead>` digunakan sebagai tag pembuka dan `</thead>` sebagai tag penutup untuk mengelompokkan baris-baris header dalam tabel.

Line [113], [121], [126], [141], `<tr>` digunakan sebagai tag pembuka dan `</tr>` sebagai tag penutup untuk membuat baris dalam sebuah tabel.

Line [114 – 120], `<th>` digunakan sebagai tag pembuka dan `</th>` sebagai tag penutup untuk membuat sebuah sel header tabel. `width` digunakan untuk mengatur lebar sel.

Line [122], [142], `<tbody>` digunakan sebagai tag pembuka dan `</tbody>` sebagai tag penutup untuk mengelompokkan baris-baris data utama dalam tabel.

Line [125], `foreach()` digunakan untuk perulangan khusus untuk memproses array.

Line [126], `echo` digunakan untuk menampilkan teks.

Line [127 – 133], [140], <td> digunakan sebagai tag pembuka dan </td> sebagai tag penutup untuk membuat sel atau kotak dalam tabel. class adalah atribut untuk menerapkan gaya CSS dari suatu kelas.

Line [134], <a> adalah tag pembuka dan adalah tag penutup untuk membuat hyperlink. href adalah atribut untuk menentukan alamat tujuan yang dituju ketika mengklik hyperlink.

Line [135], [139], <form> digunakan sebagai tag pembuka dan </form> sebagai tag penutup untuk membuat sebuah form. method adalah atribut untuk mengatur metode pengiriman data dan action adalah atribut untuk menentukan aksi yang akan dilakukan saat data dikirim.

Line [136], <input> digunakan untuk membuat kontrol input. type adalah atribut untuk menentukan jenis kontrol input, name adalah atribut untuk memberi nama kontrol input tersebut dan value adalah atribut untuk menetapkan nilai awal input dan mendefinisikan teks pada tombol.

- **FormPeminjaman.php**

Line [1], [39], [115 – 117], [120], [123], <?php digunakan sebagai tag pembuka dan ?> digunakan sebagai tag penutup program PHP.

Line [2], require_once digunakan untuk menyertakan dan menjalankan file PHP hanya satu kali selama skrip.

Line [4], [5], \$message = ''; \$loan = []; adalah inisialisasi untuk memberikan nilai awal.

Line [7 – 9], [22], [23], [35], [115], if() digunakan untuk penentuan keputusan atau conditional statement pada kondisi utama.

Line [8], [9], [22], [23], [35], [115], [120], [123], isset() digunakan untuk memeriksa apakah suatu variabel telah di atur.

Line [10], [18], [32], [41], \$id_peminjaman = generateId(\$koneksi, 'peminjaman', 'id_peminjaman'); \$result = insertData(\$koneksi, 'peminjaman', \$dataPeminjaman); \$result = updateData(\$koneksi, 'peminjaman', \$dataPeminjaman, 'id_peminjaman', \$_POST['id_peminjaman']); \$loan = getById(\$koneksi, 'peminjaman', 'id_peminjaman', \$peminjaman_id); adalah inisialisasi untuk mengambil nilai.

Line [11 – 15], [24 – 27] \$dataBuku = [] adalah inisialisasi untuk menyimpan nilai.

Line [17], [29], header() digunakan untuk mengarahkan pengguna ke halaman tertentu.

Line [18], [30], exit digunakan untuk menghentikan eksekusi skrip PHP.

Line [42], `$peminjaman_id = $_GET['id_peminjaman'];` adalah inisialisasi untuk mendapatkan nilai.

Line [41], [134], `<html>` digunakan sebagai tag pembuka dan `</html>` sebagai tag penutup dokumen HTML.

Line [42], [107], `<head>` digunakan sebagai tag pembuka dan `</head>` sebagai tag penutup bagian kepala dari dokumen HTML.

Line [43], `<title>` digunakan sebagai tag pembuka dan `</title>` sebagai tag penutup untuk membuat judul dari sebuah halaman.

Line [44], [106], `<style>` digunakan sebagai tag pembuka dan `</style>` sebagai tag penutup untuk menyisipkan kode style atau CSS ke dalam sebuah dokumen HTML.

Line [45 – 55], `body{ }` digunakan untuk menargetkan elemen body di halaman web.

Line [56 – 62], `.container{ }` digunakan untuk menargetkan elemen container di halaman web.

Line [63 – 68], `.form-header{ }` digunakan untuk menargetkan elemen form header di halaman web.

Line [69 – 75], `.form-header h3{ }` digunakan untuk menargetkan elemen form header h3 di halaman web.

Line [76 – 78], `.form-container{ }` digunakan untuk menargetkan elemen form container di halaman web.

Line [79 – 84], `form label{ }` digunakan untuk menargetkan elemen form label di halaman web.

Line [85 – 92], `form input{ }` digunakan untuk menargetkan elemen form input di halaman web.

Line [93 – 96], `button-container{ }` digunakan untuk menargetkan elemen button container di halaman web.

Line [97 – 105], `button-container input{ }` digunakan untuk menargetkan elemen button container input di halaman web.

Line [108], [133], `<body>` digunakan sebagai tag pembuka dan `</body>` sebagai tag penutup untuk bagian tubuh atau isi dari sebuah halaman.

Line [109], [110], [112], [113], [125], [129], [131], [132], `<div>` digunakan sebagai tag pembuka dan `</div>` sebagai tag penutup untuk membuat sebuah blok atau kotak. `class` adalah atribut untuk menerapkan gaya CSS dari suatu kelas.

Line [111], `<h3>` digunakan sebagai tag pembuka dan `</h3>` sebagai tag penutup untuk membuat judul dalam halaman dengan ukuran heading 3.

Line [114], [130], `<form>` digunakan sebagai tag pembuka dan `</form>` sebagai tag penutup untuk membuat sebuah form. `method` adalah atribut untuk mengatur metode pengiriman data, `action` adalah atribut untuk menentukan aksi yang akan dilakukan saat data dikirim dan `autocomplete` adalah atribut untuk menampilkan opsi atau saran inputan sebelumnya yang telah dimasukkan.

Line [116], [120], [123], [126], [129], `<input>` digunakan untuk membuat kontrol input. `type` adalah atribut untuk menentukan jenis kontrol input, `name` adalah atribut untuk memberi nama kontrol input tersebut, `id` adalah atribut untuk mendefinisikan id, `placeholder` adalah atribut untuk menyediakan petunjuk dan `value` adalah atribut untuk menetapkan nilai awal input dan mendefinisikan teks pada tombol.

Line [116], [120], [123], `echo` digunakan untuk menampilkan teks.

Line [119], [122], `<label>` digunakan sebagai tag pembuka dan `</label>` sebagai tag penutup untuk pelengkap keterangan elemen form. `for` adalah atribut untuk menghubungkan dengan elemen form.

Line [120], [123], `required` digunakan untuk menandai bahwa input harus diisi. `
` digunakan untuk menyisipkan baris baru.

Line [117], `endif` digunakan untuk menandai akhir dari sebuah kondisional `if`.

- **Peminjaman.php**

Line [1], [11], [122], [138], `<?php` digunakan sebagai tag pembuka dan `?>` digunakan sebagai tag penutup program PHP.

Line [2], `require_once` digunakan untuk menyertakan dan menjalankan file PHP hanya satu kali selama skrip.

Line [4], `$loans = getData($koneksi, 'peminjaman'); $result = deleteData($koneksi, 'peminjaman', 'id_peminjaman', $_POST['id_peminjaman']);` adalah inisialisasi untuk mengambil nilai.

Line [6], `if()` digunakan untuk penentuan keputusan atau conditional statement pada kondisi utama. `isset()` digunakan untuk memeriksa apakah suatu variabel telah di atur.

Line [8], `header()` digunakan untuk mengarahkan pengguna ke halaman tertentu.

Line [9], `exit` digunakan untuk menghentikan eksekusi skrip PHP.

Line [13], [144], `<html>` digunakan sebagai tag pembuka dan `</html>` sebagai tag penutup dokumen HTML.

Line [14], [102], `<head>` digunakan sebagai tag pembuka dan `</head>` sebagai tag penutup bagian kepala dari dokumen HTML.

Line [15], `<title>` digunakan sebagai tag pembuka dan `</title>` sebagai tag penutup untuk membuat judul dari sebuah halaman.

Line [16], [101], `<style>` digunakan sebagai tag pembuka dan `</style>` sebagai tag penutup untuk menyisipkan kode style atau CSS ke dalam sebuah dokumen HTML.

Line [17 – 27], `body{ }` digunakan untuk menargetkan elemen body di halaman web.

Line [28 – 34], `.container{ }` digunakan untuk menargetkan elemen container di halaman web.

Line [35 – 42], `.table-header{ }` digunakan untuk menargetkan elemen tabel header di halaman web.

Line [43 – 49], `.table-header h3{ }` digunakan untuk menargetkan elemen tabel header h3 di halaman web.

Line [50 – 57], `.table-header .add-button{ }` digunakan untuk menargetkan elemen tabel header add button di halaman web.

Line [58 – 60], `.table-container{ }` digunakan untuk menargetkan elemen tabel container di halaman web.

Line [61 – 64], `table{ }` digunakan untuk menargetkan elemen tabel di halaman web.

Line [65 – 67], `table thead{ }` digunakan untuk menargetkan elemen tabel thead di halaman web.

Line [68 – 75], `table th, table td{ }` digunakan untuk menargetkan elemen tabel sel header dan tabel sel data di halaman web.

Line [76 – 79], `table th{ }` digunakan untuk menargetkan elemen tabel sel header di halaman web.

Line [80 – 82], `table tbody tr:nth-child(even){ }` digunakan untuk menargetkan elemen tabel tbody dan tabel baris genap di halaman web.

Line [83 – 86], `.action-buttons{ }` digunakan untuk menargetkan elemen action buttons di halaman web.

Line [87 – 91], `.action-buttons a{ }` digunakan untuk menargetkan elemen action buttons a di halaman web.

Line [92 – 94], `.action-buttons form{ }` digunakan untuk menargetkan elemen action buttons form di halaman web.

Line [95 – 100], `.action-buttons form button{}` digunakan untuk menargetkan elemen action buttons form button di halaman web.

Line [103], [143], `<body>` digunakan sebagai tag pembuka dan `</body>` sebagai tag penutup untuk bagian tubuh atau isi dari sebuah halaman.

Line [104], [105], [109], [111], [141], [142], `<div>` digunakan sebagai tag pembuka dan `</div>` sebagai tag penutup untuk membuat sebuah blok atau kotak. `class` adalah atribut untuk menerapkan gaya CSS dari suatu kelas.

Line [106], `<h3>` digunakan sebagai tag pembuka dan `</h3>` sebagai tag penutup untuk membuat judul dalam halaman dengan ukuran heading 3.

Line [108], [133], `<button>` digunakan sebagai tag pembuka dan `</button>` sebagai tag penutup untuk membuat sebuah tombol. `class` adalah atribut untuk menerapkan gaya CSS dari suatu kelas. `onclick` adalah atribut untuk menentukan file yang akan dijalankan ketika tombol diklik. `type` adalah atribut untuk menentukan jenis kontrol input, `name` adalah atribut untuk memberi nama kontrol input tersebut

Line [112], [140], `<table>` digunakan sebagai tag pembuka dan `</table>` sebagai tag penutup untuk membuat tabel.

Line [113], [120], `<thead>` digunakan sebagai tag pembuka dan `</thead>` sebagai tag penutup untuk mengelompokkan baris-baris header dalam tabel.

Line [114], [119], [124], [136], `<tr>` digunakan sebagai tag pembuka dan `</tr>` sebagai tag penutup untuk membuat baris dalam sebuah tabel.

Line [115 – 118], `<th>` digunakan sebagai tag pembuka dan `</th>` sebagai tag penutup untuk membuat sebuah sel header tabel. `width` digunakan untuk mengatur lebar sel.

Line [121], [139], `<tbody>` digunakan sebagai tag pembuka dan `</tbody>` sebagai tag penutup untuk mengelompokkan baris-baris data utama dalam tabel.

Line [123], `foreach()` digunakan untuk perulangan khusus untuk memproses array.

Line [124], `echo` digunakan untuk menampilkan teks.

Line [125 – 128], [135], `<td>` digunakan sebagai tag pembuka dan `</td>` sebagai tag penutup untuk membuat sel atau kotak dalam tabel. `class` adalah atribut untuk menerapkan gaya CSS dari suatu kelas.

Line [129], `<a>` adalah tag pembuka dan `` adalah tag penutup untuk membuat hyperlink. `href` adalah atribut untuk menentukan alamat tujuan yang dituju ketika mengklik hyperlink.

Line [130], [134], `<form>` digunakan sebagai tag pembuka dan `</form>` sebagai tag penutup untuk membuat sebuah form. `method` adalah atribut untuk mengatur metode

pengiriman data dan `action` adalah atribut untuk menentukan aksi yang akan dilakukan saat data dikirim.

Line [131], `<input>` digunakan untuk membuat kontrol input. `type` adalah atribut untuk menentukan jenis kontrol input, `name` adalah atribut untuk memberi nama kontrol input tersebut dan `value` adalah atribut untuk menetapkan nilai awal input dan mendefinisikan teks pada tombol.

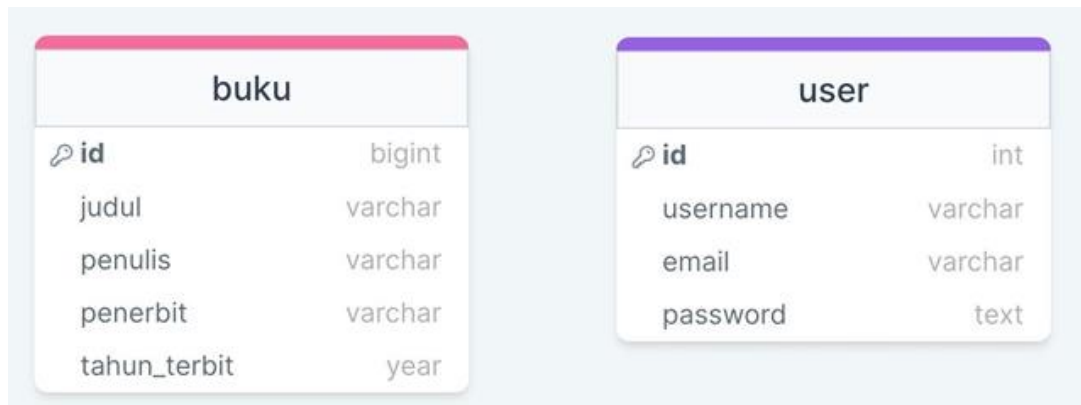
MODUL 6: CRUD DAN LOGIN PADA CODEIGNITER

SOAL 1

Buatlah sebuah website yang memiliki fitur login dan logout. Setelah login pengguna website dapat melakukan CRUD yang terhubung langsung dengan database.

Beberapa syarat yang harus ada pada website:

- Jika seorang pengguna belum login dan mengakses halaman yang mengharuskan login, maka tampilkan sebuah pesan peringatan di halaman login, peringatan berisi “Login terlebih dahulu!”.
- Berikan validasi pada bagian ‘Create’ atau ‘tambah data’ di form, sebagai berikut:
 - Judul : Harus diisi dan berupa string
 - penulis : Harus diisi dan berupa string
 - penerbit : Harus diisi dan berupa string
 - tahun_terbit : Harus diisi dan berupa angka, angka harus lebih besar dari 1800 dan lebih kecil dari 2024
- (opsional) buatlah custom pesan validasi menggunakan bahasa indonesia
- Buat file migration pada codeigniter dengan Desain konseptual database sebagai berikut:



Simpan dalam folder github PRAK601

A. Source Code

```
1 <?php
2
3 namespace Config;
4
5 $routes = Services::routes();
6
7 $routes->setDefaultNamespace('App\Controllers');
```

```

8 $routes->setDefaultController('Home');
9 $routes->setDefaultMethod('index');
10 $routes->setTranslateURIDashes(false);
11 $routes->set404Override();
12 $routes->setAutoRoute(false);
13
14 $routes->get('/', 'Home::index', ['filter' => 'auth']);
15 $routes->get('/buku', 'Buku::index', ['filter' =>
    'auth']);
16 $routes->get('/buku/create', 'Buku::create', ['filter'
    => 'auth']);
17 $routes->post('/buku/store', 'Buku::store', ['filter' =>
    'auth']);
18 $routes->get('/buku/(:num)/edit', 'Buku::edit/$1',
    ['filter' => 'auth']);
19 $routes->post('/buku/(:num)/update', 'Buku::update/$1',
    ['filter' => 'auth']);
20 $routes->post('/buku/(:num)/delete', 'Buku::delete/$1',
    ['filter' => 'auth']);
21 $routes->get('/dashboard', 'Home::dashboard', ['filter'
    => 'auth']);
22 $routes->get('/login', 'Auth::login');
23 $routes->post('/login', 'Auth::loginAuth');
24 $routes->get('/logout', 'Auth::logout');
25
26 // Include environment specific routes if available
27 if (is_file(APPPATH . 'Config/' . ENVIRONMENT .
    '/Routes.php')) {
28     require APPPATH . 'Config/' . ENVIRONMENT .
    '/Routes.php';
29 }

```

Table 26. Source Code Soal 1 Routes.php

```

1 <?php
2
3 namespace App\Controllers;
4
5 use App\Models\UserModel;
6 use CodeIgniter\Controller;
7
8 class Auth extends Controller{
9     public function store(){
10         $userModel = new UserModel();
11         $data = [
12             'username' => $this->request-
>getVar('username'),
13             'email' => $this->request-
>getVar('email'),

```

```

14         'password' => password_hash($this->request-
15 >getVar('password'), PASSWORD_DEFAULT),
16     ];
17     $userModel->save($data);
18     return redirect()->to('/login');
19 }
20 public function login(){
21     helper(['form']);
22     if ($this->request->getMethod() == 'post') {
23         return $this->loginAuth();
24     }
25     return view('LoginView');
26 }
27
28 public function loginAuth(){
29     $session = session();
30     $userModel = new UserModel();
31     $identifier = $this->request-
32 >getVar('username');
33     $password = $this->request->getVar('password');
34     $data = $userModel-
35 >getUserByEmailOrUsername($identifier);
36
37     if($data){
38         $pass = $data['password'];
39         $authenticatePassword =
40 password_verify($password, $pass);
41
42         if($authenticatePassword){
43             $ses_data = [
44                 'id' => $data['id'],
45                 'username' => $data['username'],
46                 'logged_in' => TRUE
47             ];
48             $session->set($ses_data);
49             return redirect()->to('/buku');
50         }else{
51             $session->setFlashdata('msg', 'Password
52 salah');
53             return redirect()->to('/login');
54         }
55     }else{
56         $session->setFlashdata('msg', 'Username atau
57 Email salah');
58         return redirect()->to('/login');
59     }
60 }

```

55	}
56	
57	public function logout(){
58	\$session = session();
59	\$session->destroy();
60	return redirect()->to('/login');
61	}
62	}

Tabel 27. Source Code Soal 1 Auth.php

1	<?php
2	
3	namespace App\Controllers;
4	
5	use App\Models\BukuModel;
6	use CodeIgniter\Controller;
7	
8	class Buku extends Controller{
9	public function __construct(){
10	helper(['form', 'url']);
11	}
12	
13	public function index(){
14	\$bukuModel = new BukuModel();
15	\$search = \$this->request->getGet('search');
16	\$sort = \$this->request->getGet('sort');
17	\$sort_order = \$this->request->
18	getVar('sort_order');
19	\$query = \$bukuModel;
20	if (\$search) {
21	\$query = \$query->like('judul', \$search)
22	->orLike('penulis', \$search)
23	->orLike('penerbit', \$search)
24	->orLike('tahun_terbit', \$search);
25	}
26	
27	if (\$sort && in_array(\$sort, ['judul',
28	'penulis', 'penerbit', 'tahun_terbit'])) {
29	\$bukuModel->orderBy(\$sort, \$sort_order ?
30	\$sort_order : 'ASC');
31	}
32	\$data['bukus'] = \$query->findAll();
33	\$data['search'] = \$search;
34	\$data['sort'] = \$sort;
35	\$data['sort_order'] = \$sort_order;

```

36         return view('IndexView', $data);
37     }
38
39     public function create(){
40         return view('CreateView');
41     }
42
43     public function store(){
44         $validation = \Config\Services::validation();
45
46         $validation->setRules([
47             'judul' => [
48                 'label' => 'Judul',
49                 'rules' => 'required|string',
50                 'errors' => [
51                     'required' => 'Judul harus diisi.',
52                     'string' => 'Judul harus berupa
string.'
53                 ]
54             ],
55             'penulis' => [
56                 'label' => 'Penulis',
57                 'rules' => 'required|string',
58                 'errors' => [
59                     'required' => 'Penulis harus
diisi.',
60                     'string' => 'Penulis harus berupa
string.'
61                 ]
62             ],
63             'penerbit' => [
64                 'label' => 'Penerbit',
65                 'rules' => 'required|string',
66                 'errors' => [
67                     'required' => 'Penerbit harus
diisi.',
68                     'string' => 'Penerbit harus berupa
string.'
69                 ]
70             ],
71             'tahun_terbit' => [
72                 'label' => 'Tahun Terbit',
73                 'rules' => 'required|numeric|greater_than[1800]|less_than[2024]',
74                 'errors' => [
75                     'required' => 'Tahun Terbit harus
diisi.',

```

```

76         'numeric' => 'Tahun Terbit harus
berupa angka.',
77         'greater_than' => 'Tahun Terbit
harus lebih besar dari 1800.',
        'less_than' => 'Tahun Terbit harus
78 kurang dari 2024.'
    ]
79 ]
80 });
81
82     if($validation->withRequest($this->request)-
83 >run() == FALSE){
        return view('CreateView', ['validation' =>
84 $validation]);
    }else{
85         $bukuModel = new BukuModel();
86         $data = [
87             'judul' => $this->request-
88 >getPost('judul'),
            'penulis' => $this->request-
89 >getPost('penulis'),
            'penerbit' => $this->request-
90 >getPost('penerbit'),
            'tahun_terbit' => $this->request-
91 >getPost('tahun_terbit'),
        ];
92         $bukuModel->save($data);
93         return redirect()->to('/buku');
94     }
95 }
96
97 public function edit($id){
98     $bukuModel = new BukuModel();
99     $data['book'] = $bukuModel->find($id);
100     return view('EditView', $data);
101 }
102
103 public function update($id){
104     $validation = \Config\Services::validation();
105
106     $validation->setRules([
107         'judul' => [
108             'label' => 'Judul',
109             'rules' => 'required|string',
110             'errors' => [
111                 'required' => 'Judul harus diisi.',
112

```

```

113         'string' => 'Judul harus berupa
string.'
114     ],
115     ],
116     'penulis' => [
117         'label' => 'Penulis',
118         'rules' => 'required|string',
119         'errors' => [
120             'required' => 'Penulis harus
diisi.',
121             'string' => 'Penulis harus berupa
string.'
122         ]
123     ],
124     'penerbit' => [
125         'label' => 'Penerbit',
126         'rules' => 'required|string',
127         'errors' => [
128             'required' => 'Penerbit harus
diisi.',
129             'string' => 'Penerbit harus berupa
string.'
130         ]
131     ],
132     'tahun_terbit' => [
133         'label' => 'Tahun Terbit',
134         'rules' => 'required|numeric|greater_than[1800]|less_than[2024]',
135         'errors' => [
136             'required' => 'Tahun Terbit harus
diisi.',
137             'numeric' => 'Tahun Terbit harus
berupa angka.',
138             'greater_than' => 'Tahun Terbit
harus lebih besar dari 1800.',
139             'less_than' => 'Tahun Terbit harus
kurang dari 2024.'
140         ]
141     ]
142 ];
143
144 if($validation->withRequest($this->request)-
>run() == FALSE){
145     $bukuModel = new BukuModel();
146     return view('EditView', [
147         'validation' => $validation,
148         'book' => $bukuModel->find($id)

```



```

149         });
150     }else{
151         $bukuModel = new BukuModel();
152         $data = [
153             'judul' => $this->request-
>getPost('judul'),
154             'penulis' => $this->request-
>getPost('penulis'),
155             'penerbit' => $this->request-
>getPost('penerbit'),
156             'tahun_terbit' => $this->request-
>getPost('tahun_terbit'),
157         ];
158         $bukuModel->update($id, $data);
159         return redirect()->to('/buku');
160     }
161 }
162
163 public function delete($id){
164     $bukuModel = new BukuModel();
165     $bukuModel->delete($id);
166     return redirect()->to('/buku');
167 }
168 }

```

Tabel 28. Source Code Soal 1 Buku.php

```

1 <?php
2
3 namespace App\Database\Migrations;
4
5 use CodeIgniter\Database\Migration;
6
7 class CreateDbModul6 extends Migration{
8     public function up(){
9         $this->forge->addField([
10             'id' => [
11                 'type' => 'INT',
12                 'constraint' => 7,
13                 'unsigned' => TRUE,
14                 'auto_increment' => TRUE
15             ],
16             'username' => [
17                 'type' => 'VARCHAR',
18                 'constraint' => '200',
19             ],
20             'email' => [
21                 'type' => 'VARCHAR',
22                 'constraint' => '200',

```

23],
24	'password' => [
25	'type' => 'TEXT',
26],
27]);
28	\$this->forge->addKey('id', TRUE);
29	\$this->forge->createTable('user');
30	
31	\$this->forge->addField([
32	'id' => [
33	'type' => 'BIGINT',
34	'constraint' => 10,
35	'unsigned' => TRUE,
36	'auto_increment' => TRUE
37],
38	'judul' => [
39	'type' => 'VARCHAR',
40	'constraint' => '250',
41],
42	'penulis' => [
43	'type' => 'VARCHAR',
44	'constraint' => '250',
45],
46	'penerbit' => [
47	'type' => 'VARCHAR',
48	'constraint' => '250',
49],
50	'tahun_terbit' => [
51	'type' => 'YEAR',
52],
53]);
54	\$this->forge->addKey('id', TRUE);
55	\$this->forge->createTable('buku');
56	}
57	
58	public function down(){
59	\$this->forge->dropTable('user');
60	\$this->forge->dropTable('buku');
61	}
62	}

Tabel 29. Source Code Soal 1 2024-06-10-064318_CreateDbModul6.php

1	<?php
2	
3	namespace App\Database\Seeds;
4	
5	use CodeIgniter\Database\Seeder;
6	

7	class UserSeeder extends Seeder{
8	public function run(){
9	\$data = [
10	'username' => 'wnda',
11	'email' => 'pengguna123@example.com',
12	'password' => password_hash('password',
	PASSWORD_DEFAULT),
13];
14	
15	\$this->db->table('user')->insert(\$data);
16	}
17	}

Tabel 30. Source Code Soal 1 UserSeeder.php

1	<?php
2	
3	namespace App\Filters;
4	
5	use CodeIgniter\Filters\FilterInterface;
6	use CodeIgniter\HTTP\RequestInterface;
7	use CodeIgniter\HTTP\ResponseInterface;
8	
9	class AuthFilter implements FilterInterface{
10	public function before(RequestInterface \$request,
	\$arguments = null){
11	if (!session()->get('logged_in')) {
12	return redirect()->to('/login')-
	>with('msg', 'Login terlebih dahulu!');
13	}
14	}
15	
16	public function after(RequestInterface \$request,
	ResponseInterface \$response, \$arguments = null){
17	// Do nothing
18	}
19	}

Tabel 31. Source Code Soal 1 AuthFilter.php

1	<?php
2	
3	namespace App\Models;
4	
5	use CodeIgniter\Model;
6	
7	class BukuModel extends Model{
8	protected \$table = 'buku';
9	protected \$primaryKey = 'id';
10	protected \$useAutoIncrement = true;

11	protected \$returnType	= 'array';
12	protected \$useSoftDeletes	= false;
13	protected \$protectFields	= true;
14	protected \$allowedFields	= ['judul', 'penulis', 'penerbit', 'tahun_terbit'];
15	protected bool \$allowEmptyInserts	= false;
16	protected bool \$updateOnlyChanged	= true;
17		
18	protected array \$casts	= [];
19	protected array \$castHandlers	= [];
20		
21	// Dates	
22	protected \$useTimestamps	= false;
23	protected \$dateFormat	= 'datetime';
24	protected \$createdField	= 'created_at';
25	protected \$updatedField	= 'updated_at';
26	protected \$deletedField	= 'deleted_at';
27		
28	// Validation	
29	protected \$validationRules	= [];
30	protected \$validationMessages	= [];
31	protected \$skipValidation	= false;
32	protected \$cleanValidationRules	= true;
33		
34	// Callbacks	
35	protected \$allowCallbacks	= true;
36	protected \$beforeInsert	= [];
37	protected \$afterInsert	= [];
38	protected \$beforeUpdate	= [];
39	protected \$afterUpdate	= [];
40	protected \$beforeFind	= [];
41	protected \$afterFind	= [];
42	protected \$beforeDelete	= [];
43	protected \$afterDelete	= [];
44	}	
45		

Tabel 32. Source Code Soal 1 BukuModel.php

1	<?php	
2		
3	namespace App\Models;	
4		
5	use CodeIgniter\Model;	
6		
7	class UserModel extends Model{	
8	protected \$table	= 'user';
9	protected \$primaryKey	= 'id';

10	protected \$useAutoIncrement = true;
11	protected \$returnType = 'array';
12	protected \$useSoftDeletes = false;
13	protected \$protectFields = true;
14	protected \$allowedFields = ['username', 'email', 'password', 'created_at', 'updated_at'];
15	
16	public function getUserByEmailOrUsername(\$identifier)
17	{
18	return \$this->where('email', \$identifier)
19	->orWhere('username', \$identifier)
20	->first();
21	}
22	
23	protected bool \$allowEmptyInserts = false;
24	protected bool \$updateOnlyChanged = true;
25	
26	protected array \$casts = [];
27	protected array \$castHandlers = [];
28	
29	// Dates
30	protected \$useTimestamps = true;
31	protected \$dateFormat = 'datetime';
32	protected \$createdField = 'created_at';
33	protected \$updatedField = 'updated_at';
34	protected \$deletedField = 'deleted_at';
35	
36	// Validation
37	protected \$validationRules = [];
38	protected \$validationMessages = [];
39	protected \$skipValidation = false;
40	protected \$cleanValidationRules = true;
41	
42	// Callbacks
43	protected \$allowCallbacks = true;
44	protected \$beforeInsert = [];
45	protected \$afterInsert = [];
46	protected \$beforeUpdate = [];
47	protected \$afterUpdate = [];
48	protected \$beforeFind = [];
49	protected \$afterFind = [];
50	protected \$beforeDelete = [];
51	protected \$afterDelete = [];
52	}

Tabel 33. Source Code Soal 1 UserModel.php

1	<!DOCTYPE html>
---	-----------------

2	<html>
3	<head>
4	<title>Form Buku</title>
5	<style>
6	body {
7	margin: 0;
8	padding: 0;
9	display: flex;
10	justify-content: center;
11	align-items: center;
12	height: 100vh;
13	background: linear-gradient(to bottom
14	left, #DAF0F6, #DEF0FF, #C7DEF8, #85B7F1, #73AFF6);
15	font-family: Arial, sans-serif;
16	}
17	.container {
18	width: 40%;
19	background-color: white;
20	border-radius: 12px;
21	box-shadow: 0 2px 4px rgba(0, 0, 0,
22	0.1);
23	overflow: hidden;
24	padding: 20px;
25	}
26	.form-header {
27	padding: 20px;
28	background-color: #F7F7F7;
29	border-bottom: 1px solid #E0E0E0;
30	text-align: center;
31	}
32	.form-header h3 {
33	margin: 0;
34	font-size: 25px;
35	color: black;
36	}
37	.form-container {
38	padding: 20px;
39	}
40	form label {
41	margin-bottom: 10px;
42	display: block;
43	font-size: 14px;
44	font-weight: bold;
45	}
	form input[type="text"], form
	input[type="number"] {
	width: 100%;

```

46         margin-bottom: 20px;
47         padding: 10px;
48         border: 1px solid #CCCCCC;
49         border-radius: 5px;
50         box-sizing: border-box;
51     }
52     .button-container {
53         text-align: center;
54     }
55     .button-container input[type="submit"] {
56         width: 100%;
57         padding: 10px;
58         background-color: #73AFF6;
59         border: none;
60         border-radius: 5px;
61         color: white;
62         font-size: 16px;
63     }
64     .button-container
input[type="submit"]:hover {
65         background-color: #5D9BE1;
66     }
67     .error {
68         color: red;
69         font-size: 12px;
70         margin-top: -15px;
71         margin-bottom: 10px;
72     }
73     </style>
74     </head>
75     <body>
76         <div class="container">
77             <div class="form-header">
78                 <h3>Tambah Data Buku</h3>
79             </div>
80
81             <div class="form-container">
82                 <?php if (isset($validation)): ?>
83                     <div class="error">
84                         <?= $validation->listErrors()
?>
85
86                     </div>
87                 <?php endif; ?>
88
89                 <form        method="post"        action="<?=
base_url('buku/store') ?>" autocomplete="off">
                    <?= csrf_field() ?>

```

```

90
91         <label for="judul">Judul</label>
92         <input type="text" name="judul"
93 id="judul" value="<?= set_value('judul') ?>" required>
94         <?php if (isset($validation) &&
95 $validation->hasError('judul')): ?>
96         <div class="error"><?=
97 $validation->getError('judul') ?></div>
98         <?php endif; ?>
99
100        <label
101 for="penulis">Penulis</label>
102        <input type="text" name="penulis"
103 id="penulis" value="<?= set_value('penulis') ?>"
104 required>
105        <?php if (isset($validation) &&
106 $validation->hasError('penulis')): ?>
107        <div class="error"><?=
108 $validation->getError('penulis') ?></div>
109        <?php endif; ?>
110
111        <label
112 for="penerbit">Penerbit</label>
113        <input type="text" name="penerbit"
114 id="penerbit" value="<?= set_value('penerbit') ?>"
115 required>
116        <?php if (isset($validation) &&
117 $validation->hasError('penerbit')): ?>
118        <div class="error"><?=
119 $validation->getError('penerbit') ?></div>
120        <?php endif; ?>
121
122        <label for="tahun_terbit">Tahun
123 Terbit</label>
124        <input type="number"
125 name="tahun_terbit" id="tahun_terbit" value="<?=
126 set_value('tahun_terbit') ?>" required>
127        <?php if (isset($validation) &&
128 $validation->hasError('tahun_terbit')): ?>
129        <div class="error"><?=
130 $validation->getError('tahun_terbit') ?></div>
131        <?php endif; ?>
132
133        <div class="button-container">
134        <input type="submit"
135 name="submit" value="Tambah">
136        </div>

```


118	</form>
119	</div>
120	</div>
121	</body>
122	</html>

Tabel 34. Source Code Soal 1 CreateView.php

1	<!DOCTYPE html>
2	<html>
3	<head>
4	<title>Edit Book</title>
5	<style>
6	body {
7	margin: 0;
8	padding: 0;
9	display: flex;
10	justify-content: center;
11	align-items: center;
12	height: 100vh;
13	background: linear-gradient(to bottom
14	left, #DAF0F6, #DEF0FF, #C7DEF8, #85B7F1, #73AFF6);
15	font-family: Arial, sans-serif;
16	}
17	.container {
18	width: 40%;
19	background-color: white;
20	border-radius: 12px;
21	box-shadow: 0 2px 4px rgba(0, 0, 0,
22	0.1);
23	overflow: hidden;
24	padding: 20px;
25	}
26	.form-header {
27	padding: 20px;
28	background-color: #F7F7F7;
29	border-bottom: 1px solid #E0E0E0;
30	text-align: center;
31	}
32	.form-header h3 {
33	margin: 0;
34	font-size: 25px;
35	color: black;
36	}
37	.form-container {
38	padding: 20px;
39	}
40	form label {
41	margin-bottom: 10px;

```

40             display: block;
41             font-size: 14px;
42             font-weight: bold;
43         }
44         form          input[type="text"],          form
input[type="number"] {
45             width: 100%;
46             margin-bottom: 20px;
47             padding: 10px;
48             border: 1px solid #CCCCCC;
49             border-radius: 5px;
50             box-sizing: border-box;
51         }
52         .button-container {
53             text-align: center;
54         }
55         .button-container input[type="submit"] {
56             width: 100%;
57             padding: 10px;
58             background-color: #73AFF6;
59             border: none;
60             border-radius: 5px;
61             color: white;
62             font-size: 16px;
63         }
64         .button-container
input[type="submit"]:hover {
65             background-color: #5D9BE1;
66         }
67         .error {
68             color: red;
69             font-size: 12px;
70             margin-top: -15px;
71             margin-bottom: 10px;
72         }
73     </style>
74 </head>
75 <body>
76     <div class="container">
77         <div class="form-header">
78             <h3>Edit Data Buku</h3>
79         </div>
80
81         <div class="form-container">
82             <form    action="/buku/<?=    $book['id']
?>/update" method="post">
83                 <?= csrf_field() ?>

```

```

84
85         <label>Judul</label>
86         <input type="text" name="judul"
value="<?= set_value('judul', $book['judul']) ?>"
87         <?php if (isset($validation) &&
$validation->hasError('judul')): ?>
88         <div class="error"><?=
$validation->getError('judul') ?></div>
89         <?php endif; ?>
90
91         <label>Penulis</label>
92         <input type="text" name="penulis"
value="<?= set_value('penulis', $book['penulis']) ?>"
93         <?php if (isset($validation) &&
$validation->hasError('penulis')): ?>
94         <div class="error"><?=
$validation->getError('penulis') ?></div>
95         <?php endif; ?>
96
97         <label>Penerbit</label>
98         <input type="text" name="penerbit"
value="<?= set_value('penerbit', $book['penerbit'])
?>"
99         <?php if (isset($validation) &&
$validation->hasError('penerbit')): ?>
100        <div class="error"><?=
$validation->getError('penerbit') ?></div>
101        <?php endif; ?>
102
103        <label>Tahun Terbit</label>
104        <input type="number"
name="tahun_terbit" value="<?=
set_value('tahun_terbit', $book['tahun_terbit']) ?>"
105        <?php if (isset($validation) &&
$validation->hasError('tahun_terbit')): ?>
106        <div class="error"><?=
$validation->getError('tahun_terbit') ?></div>
107        <?php endif; ?>
108
109        <div class="button-container">
110        <input type="submit"
value="Update">
111        </div>
112    </form>
113</div>
114</div>
115</body>

```

116	</html>
-----	---------

Tabel 35. Source Code Soal 1 EditView.php

1	<!DOCTYPE html>
2	<html>
3	<head>
4	<title>Data Buku</title>
5	<style>
6	body {
7	margin: 0;
8	padding: 0;
9	display: flex;
10	flex-direction: column;
11	align-items: center;
12	background: linear-gradient(to bottom
13	left, #DAF0F6, #DEF0FF, #C7DEF8, #85B7F1, #73AFF6);
14	background-size: cover;
15	background-position: center;
16	font-family: Arial, sans-serif;
17	min-height: 100vh;
18	}
19	.navbar {
20	width: 100%;
21	display: flex;
22	justify-content: flex-end;
23	padding: 10px 20px 10px 0;
24	background-color: white;
25	border-radius: 5px;
26	box-shadow: 0 2px 4px rgba(0, 0, 0,
27	0.1);
28	}
29	.navbar a {
30	color: #73AFF6;
31	font-weight: bold;
32	text-decoration: underline;
33	}
34	.container {
35	width: 90%;
36	max-width: 1200px;
37	background-color: #FFFFFF;
38	border-radius: 10px;
39	box-shadow: 0 2px 10px rgba(0, 0, 0,
40	0.1);
41	margin: 20px 0;
42	padding: 20px;
	}
	.table-header {
	padding: 10px 20px;

```
43         display: flex;
44         justify-content: space-between;
45         align-items: center;
46         background-color: #F7F7F7;
47         border-bottom: 1px solid #E0E0E0;
48     }
49     .table-header h2 {
50         margin: 0;
51         font-size: 25px;
52         color: black;
53     }
54     .table-header .add-button {
55         padding: 5px 10px;
56         background-color: #73AFF6;
57         color: black;
58         border: none;
59         border-radius: 5px;
60         font-size: 14px;
61         text-decoration: none;
62     }
63     table {
64         width: 100%;
65         border-collapse: collapse;
66         margin-top: 20px;
67     }
68     table thead {
69         background-color: #F7F7F7;
70     }
71     table th, table td {
72         padding: 10px;
73         text-align: left;
74         border-right: 1px solid white;
75         border-left: 1px solid white;
76         border-bottom: 1px solid white;
77         font-size: 14px;
78     }
79     table th {
80         background-color: #E9ECEF;
81         font-weight: bold;
82     }
83     table tbody tr:nth-child(even) {
84         background-color: #F2F2F2;
85     }
86     .action-buttons {
87         display: flex;
88         gap: 10px;
89     }
```

```

90         .action-buttons a {
91             text-decoration: none;
92             color: #007BFF;
93             font-weight: bold;
94         }
95         .action-buttons form {
96             display: inline;
97         }
98         .action-buttons form button {
99             background: none;
100            border: none;
101            color: #007BFF;
102            font-weight: bold;
103            cursor: pointer;
104        }
105    </style>
106 </head>
107 <body>
108     <div class="navbar">
109         <a href="/logout">Logout</a>
110     </div>
111
112     <div class="container">
113         <div class="table-header">
114             <h2>Data Buku</h2>
115             <a href="<?= base_url('buku/create')
116 ?>" class="add-button">Add New</a>
117         </div>
118         <table>
119             <thead>
120                 <tr>
121                     <th>ID</th>
122                     <th>Judul</th>
123                     <th>Penulis</th>
124                     <th>Penerbit</th>
125                     <th>Tahun Terbit</th>
126                     <th>Aksi</th>
127                 </tr>
128             </thead>
129             <tbody>
130                 <?php if (!empty($bukus)) : ?>
131                 <?php foreach ($bukus as $book)
132 : ?>
133                     <tr>
134                         <td><?= $book['id']
135 ?></td>

```

134	<td><?= \$book['judul'] ?></td>
135	<td><?= \$book['penulis'] ?></td>
136	<td><?= \$book['penerbit'] ?></td>
137	<td><?= \$book['tahun_terbit'] ?></td>
138	<td class="action-buttons">
139	<a href="<?= base_url("buku/{\$book['id']}/edit") ?>">Edit
140	<form method="post" action="<?= base_url("buku/{\$book['id']}/delete") ?>">
141	<?= csrf_field() ?>
142	
143	<button type="submit" name="delete">Delete</button>
144	</form>
145	</td>
146	</tr>
147	<?php endforeach; ?>
148	<?php else : ?>
149	<tr>
150	<td colspan="6">No records found</td>
151	</tr>
152	<?php endif; ?>
153	</tbody>
154	</table>
155	</div>
156	</body>
157	</html>

Tabel 36. Source Code Soal 1 IndexView.php

1	<!DOCTYPE html>
2	<html>
3	<head>
4	<title>Login</title>
5	<style>
6	body {
7	margin: 0;
8	padding: 0;
9	display: flex;
10	justify-content: center;
11	align-items: center;
12	height: 100vh;

13	background: linear-gradient(to bottom
14	left, #DAF0F6, #DEF0FF, #C7DEF8, #85B7F1, #73AFF6);
15	font-family: Arial, sans-serif;
16	}
17	.container {
18	width: 30%;
19	background-color: white;
20	border-radius: 12px;
21	box-shadow: 0 2px 4px rgba(0, 0, 0,
22	0.1);
23	overflow: hidden;
24	}
25	.form-header {
26	padding: 20px;
27	background-color: #F7F7F7;
28	border-bottom: 1px solid #E0E0E0;
29	text-align: center;
30	}
31	.form-header h3 {
32	margin: 0;
33	font-size: 25px;
34	color: black;
35	}
36	.form-container {
37	padding: 20px;
38	}
39	form label {
40	margin-bottom: 10px;
41	display: block;
42	font-size: 14px;
43	font-weight: bold;
44	}
45	form input[type="text"], form
46	input[type="password"] {
47	width: 100%;
48	margin-bottom: 20px;
49	padding: 10px;
50	border: 1px solid #CCCCCC;
51	border-radius: 5px;
52	box-sizing: border-box;
53	}
54	.button-container {
55	text-align: center;
56	}
	.button-container input[type="submit"] {
	width: 100%;
	padding: 10px;

57	background-color: #73AFF6;
58	border: none;
59	border-radius: 5px;
60	color: white;
61	font-size: 16px;
62	}
63	.button-container
	input[type="submit"]:hover {
64	background-color: #5D9BE1;
65	}
66	</style>
67	</head>
68	<body>
69	<div class="container">
70	<div class="form-header">
71	<h3>Login</h3>
72	</div>
73	
74	<div class="form-container">
75	<form method="post" action="/login"
	autocomplete="off">
76	<?= csrf_field() ?>
77	
78	<label
	for="username">Username</label>
79	<input type="text" name="username"
	id="username" required>
80	
81	<label
	for="password">Password</label>
82	<input type="password"
	name="password" id="password" required>
83	
84	<div class="button-container">
85	<input type="submit"
	name="login" value="Login">
86	</div>
87	</form>
88	</div>
89	</div>
90	</body>
91	</html>

Tabel 37. Source Code Soal 1 LoginView.php

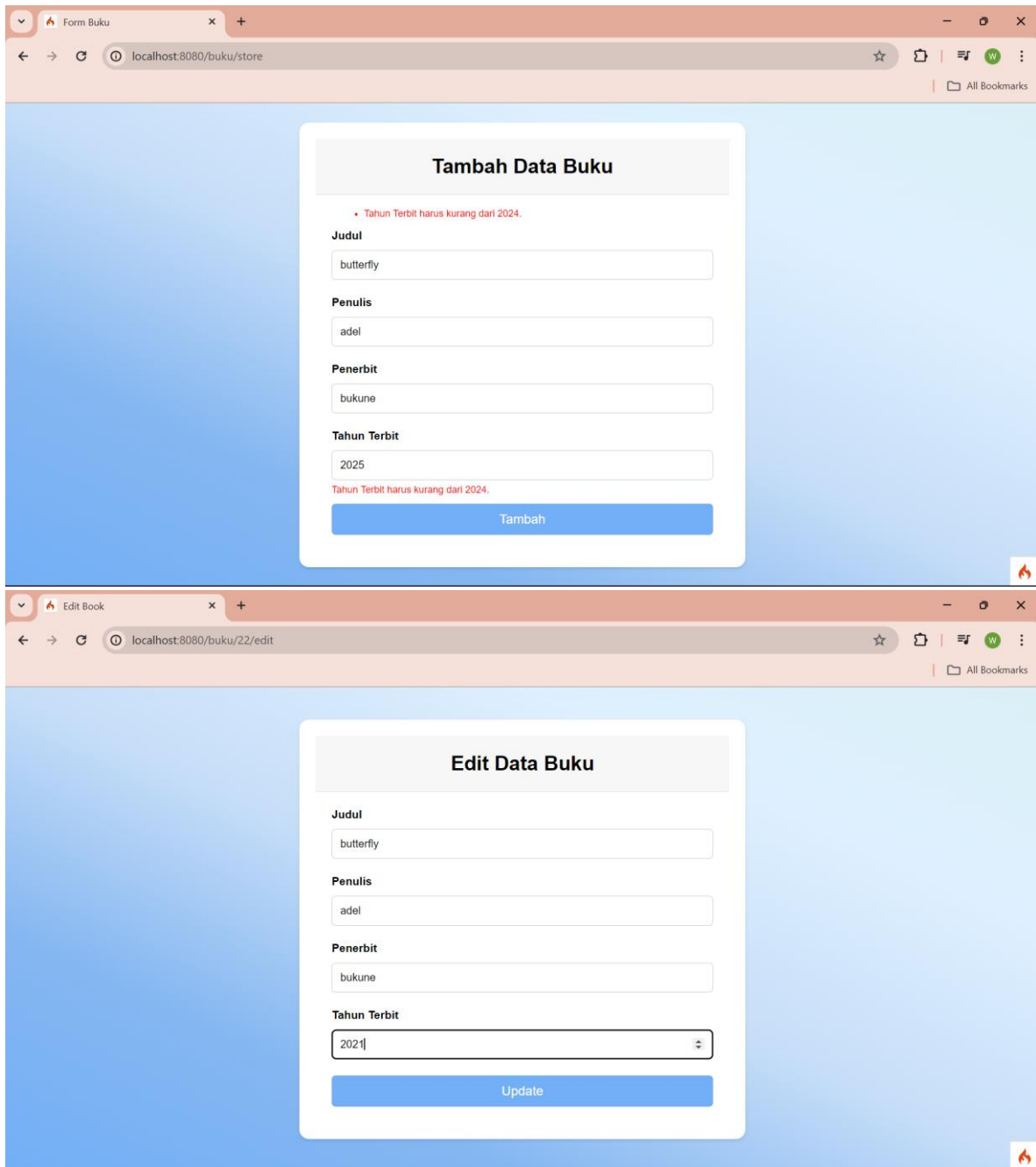
B. Output Program

The image displays two screenshots of a web application running on a local host (localhost:8080).

The top screenshot shows the **Login** page. The page has a light blue background. In the center, there is a white login form with a grey header containing the word **Login**. The form contains two input fields: **Username** (with the text 'wnda') and **Password** (with masked characters '.....'). Below the password field is a blue button labeled **Login**.

The bottom screenshot shows the **Data Buku** page. The page has a light blue background. At the top, there is a white header bar with the text **Data Buku** on the left and an **Add New** button on the right. Below the header is a table with the following data:

ID	Judul	Penulis	Penerbit	Tahun Terbit	Aksi
1	terpikat	sasa	bukune	2022	Edit Delete



Gambar 19. Screenshot Hasil Jawaban Soal 1

A. Pembahasan

- Routes.php

`namespace Config` digunakan untuk mendeklarasikan bahwa kode ini berada dalam `namespace Config`. `$routes = Services::routes()`; adalah inisialisasi layanan rute dari CodeIgniter, yang memungkinkan kita untuk mendefinisikan berbagai rute.

- `setDefaultNamespace('App\Controllers')`: Menetapkan namespace default untuk controller, yaitu `App\Controllers`.
- `setDefaultController('Home')`: Menentukan controller default yang akan dipanggil jika tidak ada controller yang spesifik dalam URL, yaitu `Home`.
- `setDefaultMethod('index')`: Menentukan metode default yang akan dipanggil dalam controller, yaitu `index`.
- `setTranslateURIDashes(false)`: Menentukan apakah tanda hubung dalam URI harus diterjemahkan menjadi garis bawah dalam nama metode.
- `set404Override()`: Menentukan apakah akan ada penanganan khusus untuk halaman 404.
- `setAutoRoute(false)`: Menonaktifkan rute otomatis untuk alasan keamanan dan kontrol lebih ketat atas rute yang tersedia.

Berbagai rute didefinisikan menggunakan method `$routes->get()`, `$routes->post()`, dan lainnya. Setiap rute memiliki URI, target controller, dan metode, serta filter `auth` yang memastikan hanya pengguna yang terotentikasi yang bisa mengakses rute tersebut.

- `get('/', 'Home::index', ['filter' => 'auth'])`: Menetapkan rute untuk halaman utama yang akan memanggil metode `index` dari controller `Home` dengan filter `auth`.
- `get('/buku', 'Buku::index', ['filter' => 'auth'])`: Menetapkan rute untuk halaman daftar buku yang memanggil metode `index` dari controller `Buku` dengan filter `auth`.
- `get('/buku/create', 'Buku::create', ['filter' => 'auth'])`: Rute untuk halaman pembuatan buku baru.
- `post('/buku/store', 'Buku::store', ['filter' => 'auth'])`: Rute untuk menyimpan data buku baru.
- `get('/buku/(:num)/edit', 'Buku::edit/$1', ['filter' => 'auth'])`: Rute untuk mengedit buku dengan ID tertentu.
- `post('/buku/(:num)/update', 'Buku::update/$1', ['filter' => 'auth'])`: Rute untuk memperbarui data buku dengan ID tertentu.
- `post('/buku/(:num)/delete', 'Buku::delete/$1', ['filter' => 'auth'])`: Rute untuk menghapus buku dengan ID tertentu.
- `get('/dashboard', 'Home::dashboard', ['filter' => 'auth'])`: Rute untuk halaman dashboard.
- `get('/login', 'Auth::login')`: Rute untuk halaman login tanpa filter `auth` karena perlu diakses oleh pengguna yang belum login.
- `post('/login', 'Auth::loginAuth')`: Rute untuk proses autentikasi login.
- `get('/logout', 'Auth::logout')`: Rute untuk logout pengguna.

- Auth.php

namespace `App\Controllers` digunakan untuk mendeklarasikan bahwa kelas `Auth` berada dalam folder `Controllers` di aplikasi. `use App\Models\UserModel;` dan `use CodeIgniter\Controller;` digunakan untuk mengimpor model `UserModel` dan kelas `Controller` dari `CodeIgniter`.

Fungsi `store` bertanggung jawab untuk menyimpan data pengguna baru ke database.

- `UserModel` diinisialisasi untuk berinteraksi dengan tabel pengguna.
- Data pengguna diambil dari permintaan (`request`) menggunakan `$this->request->getVar('fieldname')`.
- Kata sandi di-hash menggunakan `password_hash` dengan algoritma default.
- Data disimpan ke database menggunakan metode `save` dari `UserModel`.
- Setelah penyimpanan berhasil, pengguna diarahkan ke halaman login dengan `redirect()->to('/login')`.

Fungsi `login` bertanggung jawab untuk menampilkan halaman login dan memproses data login.

- `helper(['form'])` memuat helper form untuk memudahkan pengelolaan form.
- Jika metode permintaan adalah `POST`, maka fungsi `loginAuth` dipanggil untuk memproses login.
- Jika tidak, halaman login (`LoginView`) ditampilkan.

Fungsi `loginAuth` bertanggung jawab untuk memproses autentikasi login pengguna.

- Sesi diinisialisasi dengan `session()`.
- `UserModel` diinisialisasi untuk berinteraksi dengan tabel pengguna.
- Identifikasi pengguna (bisa `username` atau `email`) dan kata sandi diambil dari permintaan.
- `getUserByEmailOrUsername` adalah metode model yang digunakan untuk mendapatkan data pengguna berdasarkan `username` atau `email`.
- Jika data pengguna ditemukan, kata sandi yang di-hash diverifikasi menggunakan `password_verify`.
- Jika verifikasi kata sandi berhasil, data sesi pengguna diatur dan pengguna diarahkan ke halaman buku (`/buku`).
- Jika kata sandi salah, pesan kesalahan disimpan dalam sesi (`setFlashdata`) dan pengguna diarahkan kembali ke halaman login.
- Jika `username` atau `email` tidak ditemukan, pesan kesalahan juga disimpan dalam sesi dan pengguna diarahkan kembali ke halaman login.

Fungsi `logout` bertanggung jawab untuk mengakhiri sesi pengguna.

- Sesi diinisialisasi dengan `session()`.
- Sesi dihancurkan menggunakan `$session->destroy()`.
- Pengguna diarahkan kembali ke halaman login dengan `redirect()->to('/login')`.

- Buku.php

`namespace App\Controllers` digunakan untuk mendeklarasikan bahwa kelas `Buku` berada dalam folder `Controllers` di aplikasi. `use App\Models\BukuModel;` dan `use CodeIgniter\Controller;` digunakan untuk mengimpor model `BukuModel` dan kelas `Controller` dari `CodeIgniter`.

Fungsi `__construct` memuat helper `form` dan `url` untuk mempermudah pengelolaan form dan URL dalam metode-metode lain di controller ini.

Fungsi `index` bertanggung jawab untuk menampilkan daftar buku dengan opsi pencarian dan pengurutan.

- `BukuModel` diinisialisasi untuk berinteraksi dengan tabel buku.
- `search`, `sort`, dan `sort_order` diambil dari parameter GET.
- Query awal disiapkan dari `BukuModel`.
- Jika ada parameter `search`, query diperbarui untuk mencakup kondisi `like` untuk kolom `judul`, `penulis`, `penerbit`, dan `tahun_terbit`.
- Jika ada parameter `sort` dan nilai `sort` valid, query diperbarui untuk mengurutkan berdasarkan kolom yang dipilih dengan urutan yang sesuai.
- Data buku diambil dengan `findAll()` dan disimpan dalam array `$data`.
- Halaman `IndexView` ditampilkan dengan data yang telah dikumpulkan.

Fungsi `create` bertanggung jawab untuk menampilkan form pembuatan buku baru.

- Menampilkan halaman `CreateView`.

Fungsi `store` bertanggung jawab untuk memvalidasi dan menyimpan data buku baru.

- Validasi diatur untuk memastikan semua kolom wajib (`judul`, `penulis`, `penerbit`, `tahun_terbit`) diisi dan sesuai dengan aturan.
- Jika validasi gagal, form pembuatan buku ditampilkan kembali dengan pesan kesalahan.
- Jika validasi berhasil, data buku disimpan menggunakan `save` dari `BukuModel`.
- Pengguna diarahkan ke halaman daftar buku dengan `redirect()->to('/buku')`.

Fungsi `edit` bertanggung jawab untuk menampilkan form pengeditan buku dengan data buku yang ada.

- Data buku diambil berdasarkan ID dengan `find`.
- Halaman `EditView` ditampilkan dengan data buku.

Fungsi `update` bertanggung jawab untuk memvalidasi dan memperbarui data buku yang ada.

- Validasi diatur sama seperti pada fungsi `store`.
- Jika validasi gagal, form pengeditan buku ditampilkan kembali dengan pesan kesalahan dan data buku yang ada.
- Jika validasi berhasil, data buku diperbarui menggunakan `update` dari `BukuModel`.
- Pengguna diarahkan ke halaman daftar buku dengan `redirect()->to('/buku')`.

Fungsi `delete` bertanggung jawab untuk menghapus data buku berdasarkan ID.

- Data buku dihapus menggunakan `delete` dari `BukuModel`.
- Pengguna diarahkan ke halaman daftar buku dengan `redirect()->to('/buku')`.

- **2024-06-10-064318_CreateDbModul6.php**

`namespace App\Database\Migrations` digunakan untuk mendeklarasikan bahwa kelas `CreateDbModul6` berada dalam folder `Migrations` di aplikasi. `use CodeIgniter\Database\Migration;` digunakan untuk mengimpor kelas `Migration` dari `CodeIgniter`.

Fungsi `up` bertanggung jawab untuk membuat tabel dalam database.

- `addField` digunakan untuk menambahkan kolom ke tabel.
- `addKey` digunakan untuk menambahkan kunci utama (primary key) ke tabel.
- `createTable` digunakan untuk membuat tabel dengan nama yang diberikan.

Fungsi `down` bertanggung jawab untuk menghapus tabel dalam database jika diperlukan.

- `dropTable('user')` menghapus tabel `user`.
- `dropTable('buku')` menghapus tabel `buku`.

- **UserSeeder.php**

`namespace App\Database\Seeds` digunakan untuk mendeklarasikan bahwa kelas `UserSeeder` berada dalam folder `Seeds` di aplikasi. `use CodeIgniter\Database\Seeder;` digunakan untuk mengimpor kelas `Seeder` dari `CodeIgniter`.

Fungsi `run` adalah fungsi utama yang akan dijalankan ketika seeder dieksekusi.

- Array `$data` digunakan untuk menyimpan data pengguna yang akan dimasukkan ke dalam tabel `user`.
 - `username` diatur dengan nilai `'wnda'`.
 - `email` diatur dengan nilai `'pengguna123@example.com'`.
 - `password` diatur dengan nilai hash dari string `'password'` menggunakan fungsi `password_hash` dengan algoritma default (`PASSWORD_DEFAULT`).
- `$this->db->table('user')->insert($data)` digunakan untuk memasukkan data ke dalam tabel `user`.
 - `$this->db->table('user')` memilih tabel `user`.
 - `insert($data)` memasukkan data yang ada di array `$data` ke dalam tabel yang dipilih.

- **AuthFilter.php**

`namespace App\Filters` digunakan untuk mendeklarasikan bahwa kelas `AuthFilter` berada dalam folder `Filters` di aplikasi `use CodeIgniter\Filters\FilterInterface;`, `use CodeIgniter\HTTP\RequestInterface;` dan `use CodeIgniter\HTTP\ResponseInterface;` digunakan untuk mengimpor antarmuka dan kelas yang diperlukan dari `CodeIgniter`.

Fungsi `before` dijalankan sebelum permintaan (request) diproses lebih lanjut.

- `if (!session()->get('logged_in')) { ... }` memeriksa apakah sesi `logged_in` tidak diatur atau bernilai `false`.
 - Jika pengguna belum login, maka pengguna akan diarahkan ke halaman login dengan `return redirect()->to('/login')->with('msg', 'Login terlebih dahulu!');`
 - `redirect()->to('/login')` mengarahkan pengguna ke halaman login.
 - `->with('msg', 'Login terlebih dahulu!')` menambahkan pesan flashdata yang akan ditampilkan di halaman login, memberi tahu pengguna untuk login terlebih dahulu.

Fungsi `after` dijalankan setelah permintaan diproses dan sebelum respons dikirim ke klien.

- Pada implementasi ini, fungsi `after` tidak melakukan apa-apa, sebagaimana dinyatakan dengan komentar `// Do nothing`.

- **BukuModel.php**

`namespace App\Models` digunakan untuk mendeklarasikan bahwa kelas `BukuModel` berada dalam folder `Models` di aplikasi. `use CodeIgniter\Model;` digunakan untuk mengimpor kelas `Model` dari `CodeIgniter`.

- `$table`: Menentukan nama tabel yang digunakan, yaitu `buku`.
- `$primaryKey`: Menentukan kolom kunci utama dari tabel, yaitu `id`.
- `$useAutoIncrement`: Menentukan bahwa kolom `id` akan menggunakan `auto increment`.
- `$returnType`: Menentukan tipe data yang dikembalikan sebagai `array`.
- `$useSoftDeletes`: Menentukan apakah `soft deletes` digunakan atau tidak. Pada kasus ini, nilainya `false` berarti tidak menggunakan `soft deletes`.
- `$protectFields`: Menentukan apakah hanya kolom yang diizinkan (`allowedFields`) yang dapat dimanipulasi. Pada kasus ini, nilainya `true`.
- `$allowedFields`: Menentukan kolom-kolom yang diizinkan untuk diisi, yaitu `judul, penulis, penerbit, dan tahun_terbit`.
- `$allowEmptyInserts`: Menentukan apakah `insert` tanpa data diperbolehkan. Nilainya `false` berarti tidak diperbolehkan.
- `$updateOnlyChanged`: Menentukan apakah hanya kolom yang berubah saja yang akan diperbarui. Nilainya `true`.
- `$casts` **dan** `$castHandlers`: Digunakan untuk `casting` data, namun pada kasus ini tidak ada `casting` yang didefinisikan.
- `$useTimestamps`: Menentukan apakah `timestamps` digunakan atau tidak. Pada kasus ini, nilainya `false`.
- `$dateFormat`: Menentukan format tanggal yang digunakan. Nilainya `datetime`.

- `$createdField`, `updatedField`, `deletedField`: Menentukan nama kolom untuk timestamps, namun tidak digunakan dalam kasus ini karena `$useTimestamps` bernilai `false`.
- `$validationRules`: Menyimpan aturan validasi, namun tidak ada aturan yang didefinisikan pada kasus ini.
- `$validationMessages`: Menyimpan pesan validasi khusus, namun tidak ada pesan yang didefinisikan.
- `$skipValidation`: Menentukan apakah validasi akan dilewati. Nilainya `false` berarti validasi tidak dilewati.
- `$cleanValidationRules`: Menentukan apakah aturan validasi akan dibersihkan. Nilainya `true`.
- `$allowCallbacks`: Menentukan apakah callback diizinkan. Nilainya `true`.
- `$beforeInsert`, `$afterInsert`, `$beforeUpdate`, `$afterUpdate`, `$beforeFind`, `$afterFind`, `$beforeDelete`, `$afterDelete`: Menyimpan nama-nama metode callback yang akan dipanggil pada waktu yang sesuai, namun pada kasus ini tidak ada callback yang didefinisikan.

- **UserModel.php**

Kelas `UserModel` berada dalam namespace `App\Models` dan meng-extend kelas `Model` dari `CodeIgniter`, yang menyediakan fitur-fitur untuk berinteraksi dengan database.

- `$table`: Menentukan nama tabel yang digunakan, yaitu `user`.
- `$primaryKey`: Menentukan kolom kunci utama dari tabel, yaitu `id`.
- `$useAutoIncrement`: Menentukan bahwa kolom `id` akan menggunakan auto increment.
- `$returnType`: Menentukan tipe data yang dikembalikan sebagai array.
- `$useSoftDeletes`: Menentukan apakah soft deletes digunakan atau tidak. Pada kasus ini, nilainya `false` berarti tidak menggunakan soft deletes.
- `$protectFields`: Menentukan apakah hanya kolom yang diizinkan (`allowedFields`) yang dapat dimanipulasi. Pada kasus ini, nilainya `true`.
- `$allowedFields`: Menentukan kolom-kolom yang diizinkan untuk diisi, yaitu `username`, `email`, `password`, `created_at`, dan `updated_at`.

Metode `getUserByEmailOrUsername` digunakan untuk mengambil data pengguna berdasarkan email atau username yang diberikan sebagai parameter `$identifier`. Metode ini menggunakan query `where` untuk mencocokkan entri di kolom `email` atau `username`, dan mengembalikan hasil pertama yang ditemukan.

- `$allowEmptyInserts`: Menentukan apakah insert tanpa data diperbolehkan. Nilainya `false` berarti tidak diperbolehkan.
- `$updateOnlyChanged`: Menentukan apakah hanya kolom yang berubah saja yang akan diperbarui. Nilainya `true`.
- `$casts` dan `$castHandlers`: Digunakan untuk casting data, namun pada kasus ini tidak ada casting yang didefinisikan.

- `$useTimestamps`: Menentukan apakah timestamps digunakan atau tidak. Pada kasus ini, nilainya `true`.
- `$dateFormat`: Menentukan format tanggal yang digunakan, yaitu `'datetime'`.
- `$createdField` dan `$updatedAtField`: Menentukan nama kolom untuk timestamps `created_at` dan `updated_at`.
- `$deletedField`: Dalam kasus ini, nama kolom untuk soft deletes tidak didefinisikan karena `useSoftDeletes` bernilai `false`.

Properti seperti `$validationRules`, `$validationMessages`, `$skipValidation`, dan `$cleanValidationRules` digunakan untuk aturan validasi, pesan validasi, serta pengaturan validasi lainnya, namun dalam implementasi ini tidak ada aturan validasi yang ditentukan.

Callbacks seperti `$beforeInsert`, `$afterInsert`, `$beforeUpdate`, `$afterUpdate`, `$beforeFind`, `$afterFind`, `$beforeDelete`, dan `$afterDelete` dapat digunakan untuk menambahkan logika sebelum atau setelah operasi tertentu dieksekusi di database.

- **CreateView.php**

Dokumen HTML dimulai dengan deklarasi `<!DOCTYPE html>` untuk mendefinisikan tipe dokumen dan versi HTML yang digunakan. Elemen `<html>` menjadi pembungkus seluruh konten dokumen.

Bagian `<head>` berisi metadata dokumen seperti judul dan gaya. Tag `<title>` memberikan judul "Form Buku" pada halaman. Blok `<style>` di dalam `<head>` mendefinisikan berbagai gaya CSS yang digunakan untuk memformat tampilan form.

Kode CSS internal digunakan untuk mengatur tata letak dan tampilan visual dari halaman. Ini mencakup pengaturan seperti background gradient, warna teks, ukuran font, dan tata letak elemen-elemen seperti form dan tombol.

- `body`: Mengatur margin, padding, tampilan fleksibel, pusatkan konten secara horizontal dan vertikal, tinggi 100vh, latar belakang dengan gradien warna, dan font default Arial.
- `.container`: Kontainer utama dengan lebar 40%, latar belakang putih, border-radius 12px, bayangan, dan padding 20px.
- `.form-header`: Bagian header form dengan padding, latar belakang abu-abu terang, border bawah, dan teks tengah.
- `.form-header h3`: Mengatur margin 0, ukuran font 25px, dan warna teks hitam.
- `.form-container`: Memberikan padding 20px pada kontainer form.
- `form label`: Mengatur margin bawah, tampilan blok, ukuran font, dan berat font.
- `form input`: Mengatur lebar penuh, margin bawah, padding, border, border-radius, dan box-sizing.
- `.button-container`: Mengatur teks tengah.

- `.button-container input[type="submit"]`: Mengatur lebar penuh, padding, latar belakang biru, border none, border-radius, warna putih, ukuran font, dan gaya hover.
- `.error`: Mengatur warna merah, ukuran font, dan margin untuk pesan error.

Bagian `<body>` berisi elemen `<div class="container">` yang menjadi kontainer utama form. Di dalamnya terdapat dua div: `.form-header` untuk header dan `.form-container` untuk kontainer form.

- `<?php if (isset($validation)): ?>` Mengecek apakah variabel `$validation` ada, yang menunjukkan adanya validasi.
- `<div class="error"> <?= $validation->listErrors() ?> </div>` Menampilkan pesan error jika ada kesalahan validasi.

Formulir menggunakan metode POST dan diarahkan ke `<?= base_url('buku/store') ?>`. Fungsi `csrf_field()` menambahkan token CSRF untuk keamanan.

- Label dan Input: Setiap label dan input teks diberi atribut `name`, `id`, dan nilai default menggunakan fungsi `set_value()`.
- Validasi Input: Jika ada kesalahan, pesan error spesifik akan ditampilkan di bawah input terkait.

Tombol submit ditempatkan dalam `.button-container` untuk mengirimkan form. Tombol ini memiliki gaya yang konsisten dengan tema form.

- EditView.php

Dokumen HTML dimulai dengan deklarasi `<!DOCTYPE html>` untuk mendefinisikan tipe dokumen dan versi HTML yang digunakan. Elemen `<html>` menjadi pembungkus seluruh konten dokumen.

Bagian `<head>` berisi metadata dokumen seperti judul dan gaya. Tag `<title>` memberikan judul "Edit Book" pada halaman. Blok `<style>` di dalam `<head>` mendefinisikan berbagai gaya CSS yang digunakan untuk memformat tampilan form.

Kode CSS internal digunakan untuk mengatur tata letak dan tampilan visual dari halaman. Ini mencakup pengaturan seperti background gradient, warna teks, ukuran font, dan tata letak elemen-elemen seperti form dan tombol.

- `body`: Mengatur margin, padding, tampilan fleksibel, pusatkan konten secara horizontal dan vertikal, tinggi 100vh, latar belakang dengan gradien warna, dan font default Arial.
- `.container`: Kontainer utama dengan lebar 40%, latar belakang putih, border-radius 12px, bayangan, dan padding 20px.
- `.form-header`: Bagian header form dengan padding, latar belakang abu-abu terang, border bawah, dan teks tengah.
- `.form-header h3`: Mengatur margin 0, ukuran font 25px, dan warna teks hitam.
- `.form-container`: Memberikan padding 20px pada kontainer form.

- `form label`: Mengatur margin bawah, tampilan blok, ukuran font, dan berat font.
- `form input`: Mengatur lebar penuh, margin bawah, padding, border, border-radius, dan box-sizing.
- `.button-container`: Mengatur teks tengah.
- `.button-container input[type="submit"]`: Mengatur lebar penuh, padding, latar belakang biru, border none, border-radius, warna putih, ukuran font, dan gaya hover.
- `.error`: Mengatur warna merah, ukuran font, dan margin untuk pesan error.

Bagian `<body>` berisi elemen `<div class="container">` yang menjadi kontainer utama form. Di dalamnya terdapat dua div: `.form-header` untuk header dan `.form-container` untuk kontainer form.

Formulir menggunakan metode POST dan diarahkan ke `/buku/<?=$book['id']?>/update`. Fungsi `csrf_field()` menambahkan token CSRF untuk keamanan.

- Label dan Input: Setiap label dan input teks diberi atribut name dan nilai default menggunakan fungsi `set_value()` untuk mengisi input dengan data yang ada dalam variabel `$book`.
- Validasi Input: Jika ada kesalahan, pesan error spesifik akan ditampilkan di bawah input terkait.
- `<?php if (isset($validation) && $validation->hasError('judul')): ?>` Mengecek apakah variabel `$validation` ada dan apakah terdapat kesalahan pada input 'judul'.
- `<div class="error"><?=$validation->getError('judul')?></div>` Menampilkan pesan error jika ada kesalahan validasi.

Tombol submit ditempatkan dalam `.button-container` untuk mengirimkan form. Tombol ini memiliki gaya yang konsisten dengan tema form.

- IndexView.php

Dokumen HTML dimulai dengan deklarasi `<!DOCTYPE html>` untuk mendefinisikan tipe dokumen dan versi HTML yang digunakan. Elemen `<html>` menjadi pembungkus seluruh konten dokumen.

Bagian `<head>` berisi metadata dokumen seperti judul dan gaya. Tag `<title>` memberikan judul "Data Buku" pada halaman. Blok `<style>` di dalam `<head>` mendefinisikan berbagai gaya CSS yang digunakan untuk memformat tampilan halaman.

Kode CSS internal digunakan untuk mengatur tata letak dan tampilan visual dari halaman. Ini mencakup pengaturan seperti background gradient, warna teks, ukuran font, dan tata letak elemen-elemen seperti form dan tombol.

- `body`: Mengatur margin, padding, tampilan fleksibel, arah kolom, pusatkan konten secara horizontal dan vertikal, latar belakang dengan gradien warna, ukuran latar belakang, posisi latar belakang, font default Arial, dan tinggi minimum 100vh.

- `.navbar`: Mengatur lebar penuh, tampilan fleksibel, pusatkan konten di ujung kanan, padding, latar belakang putih, border-radius 5px, dan bayangan.
- `.navbar a`: Mengatur warna teks biru, font tebal, dan dekorasi teks garis bawah.
- `.container`: Kontainer utama dengan lebar 90%, maksimal lebar 1200px, latar belakang putih, border-radius 10px, bayangan, margin vertikal 20px, dan padding 20px.
- `.table-header`: Mengatur padding, tampilan fleksibel, pusatkan konten secara horizontal, latar belakang abu-abu terang, dan border bawah.
- `.table-header h2`: Mengatur margin 0, ukuran font 25px, dan warna teks hitam.
- `.table-header .add-button`: Mengatur padding, latar belakang biru, warna teks hitam, border none, border-radius 5px, ukuran font, dan dekorasi teks none.
- `table`: Mengatur lebar penuh, border-collapse, dan margin atas 20px.
- `table thead`: Mengatur latar belakang abu-abu terang.
- `table th, table td`: Mengatur padding, teks rata kiri, border samping dan bawah, dan ukuran font.
- `table th`: Mengatur latar belakang abu-abu lebih gelap dan font tebal.
- `table tbody tr:nth-child(even)`: Mengatur latar belakang abu-abu terang.
- `.action-buttons`: Mengatur tampilan fleksibel dan jarak antar elemen.
- `.action-buttons a`: Mengatur dekorasi teks none, warna teks biru, dan font tebal.
- `.action-buttons form`: Mengatur tampilan inline.
- `.action-buttons form button`: Mengatur latar belakang none, border none, warna teks biru, font tebal, dan cursor pointer.

Bagian `<body>` berisi dua elemen utama:

- `.navbar`: Elemen ini berisi tautan logout yang ditempatkan di ujung kanan.
- `.container`: Kontainer utama yang berisi tabel data buku.
- `.table-header`: Bagian header tabel dengan judul "Data Buku" dan tombol "Add New" untuk menambahkan data buku baru.
- `table`: Elemen tabel yang menampilkan data buku.
 - `thead`: Bagian header tabel yang berisi judul kolom: ID, Judul, Penulis, Penerbit, Tahun Terbit, dan Aksi.
 - `tbody`: Bagian isi tabel yang menampilkan data buku. Jika tidak ada data, baris "No records found" akan ditampilkan.
- `<?php if (!empty($bukus)) : ?>`: Mengecek apakah variabel `$bukus` tidak kosong.
- `<?php foreach ($bukus as $book) : ?>`: Melakukan iterasi untuk setiap buku dalam variabel `$bukus` dan menampilkan data buku dalam baris tabel.
 - `<?= $book['id'] ?>`: Menampilkan ID buku.
 - `<?= $book['judul'] ?>`: Menampilkan judul buku.

- o `<?= $book['penulis'] ?>`: Menampilkan penulis buku.
- o `<?= $book['penerbit'] ?>`: Menampilkan penerbit buku.
- o `<?= $book['tahun_terbit'] ?>`: Menampilkan tahun terbit buku.
- o `<?= base_url("buku/{$book['id']}/edit") ?>`: Tautan untuk mengedit data buku.
- o `<?= base_url("buku/{$book['id']}/delete") ?>`: Formulir untuk menghapus data buku dengan metode POST dan token CSRF untuk keamanan.

- LoginView.php

Dokumen HTML dimulai dengan deklarasi `<!DOCTYPE html>` untuk mendefinisikan tipe dokumen dan versi HTML yang digunakan. Elemen `<html>` membungkus seluruh konten dokumen.

Bagian `<head>` berisi metadata dokumen seperti judul dan gaya. Tag `<title>` memberikan judul "Login" pada halaman. Blok `<style>` di dalam `<head>` mendefinisikan berbagai gaya CSS yang digunakan untuk memformat tampilan halaman.

Kode CSS internal digunakan untuk mengatur tata letak dan tampilan visual dari halaman. Ini mencakup pengaturan seperti background gradient, warna teks, ukuran font, dan tata letak elemen-elemen seperti form dan tombol.

- `body`: Mengatur margin, padding, tampilan fleksibel, pusatkan konten secara horizontal dan vertikal, latar belakang dengan gradien warna, font default Arial, dan tinggi penuh layar (100vh).
- `.container`: Kontainer utama dengan lebar 30%, latar belakang putih, border-radius 12px, bayangan lembut, dan overflow tersembunyi.
- `.form-header`: Bagian header formulir dengan padding 20px, latar belakang abu-abu terang (#F7F7F7), border-bottom, dan teks tengah.
- `.form-header h3`: Mengatur margin 0, ukuran font 25px, dan warna teks hitam.
- `.form-container`: Kontainer formulir dengan padding 20px.
- `form label`: Mengatur margin bawah, tampilan blok, ukuran font 14px, dan font tebal.
- `form input[type="text"], form input[type="password"]`: Mengatur lebar penuh, margin bawah 20px, padding 10px, border 1px solid #CCCCCC, border-radius 5px, dan box-sizing border-box.
- `.button-container`: Kontainer untuk tombol dengan teks tengah.
- `.button-container input[type="submit"]`: Tombol submit dengan lebar penuh, padding 10px, latar belakang biru (#73AFF6), border none, border-radius 5px, warna teks putih, dan ukuran font 16px.
- `.button-container input[type="submit"]:hover`: Mengatur latar belakang biru lebih gelap (#5D9BE1) saat dihover.
- `.container`: Kontainer utama yang berisi formulir login.
- `.form-header`: Bagian header formulir dengan judul "Login".
- `.form-container`: Kontainer utama untuk formulir login.

- **form:** Elemen formulir dengan metode POST dan aksi menuju /login.
 - `<?= csrf_field() ?>`: PHP untuk menambahkan token CSRF untuk keamanan.
 - `label for="username"`: Label untuk input username.
 - `input type="text" name="username" id="username"`: Input teks untuk username dengan atribut `required` agar tidak boleh kosong.
 - `label for="password"`: Label untuk input password.
 - `input type="password" name="password" id="password"`: Input password dengan atribut `required` agar tidak boleh kosong.
 - `.button-container`: Kontainer untuk tombol submit.
 - `input type="submit" name="login" value="Login"`: Tombol submit dengan nilai "Login".

LINK GITHUB

<https://github.com/wandakarliyanti/Praktikum-Pemrograman-Web-II>