

**LAPORAN AKHIR PRAKTIKUM
PEMROGRAMAN WEB II**



Oleh:

Muh. Haekal Barera NIM. 2210817310008

**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 : Muh. Haekal Barera

NIM : 2210817310008

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 TABEL	6
DAFTAR GAMBAR.....	8
MODUL 1	10
Soal 1.....	10
A. Source Code	10
B. Output Program	10
C. Pembahasan	10
Soal 2.....	11
A. Source Code	12
B. Output Program	13
C. Pembahasan	13
Soal 3.....	14
A. Source Code	14
B. Output Program	15
C. Pembahasan	15
Soal 4.....	15
A. Source Code	16
B. Output Program	17
C. Pembahasan	17
Soal 5.....	18
A. Source Code	19
B. Output Program	21
C. Pembahasan	21
Tautan Git	22
MODUL 2	22
Soal 1.....	22
A. Source Code	22
B. Output Program	24
C. Pembahasan	24
Soal 2.....	25
A. Source Code	26
B. Output Program	28

C. Pembahasan	29
Soal 3.....	30
A. Source Code	30
B. Output Program	33
C. Pembahasan	33
Soal 4.....	34
A. Source Code	35
B. Output Program	36
C. Pembahasan	36
Tautan Git	37
MODUL 3	37
Soal 1.....	37
A. Source Code	38
B. Output Program	39
C. Pembahasan	40
Soal 2.....	41
A. Source Code	41
B. Output Program	43
C. Pembahasan	43
Soal 3.....	44
A. Source Code	45
B. Output Program	46
C. Pembahasan	46
Soal 4.....	47
A. Source Code	48
B. Output Program	49
C. Pembahasan	50
Soal 5.....	51
A. Source Code	52
B. Output Program	53
C. Pembahasan	53
Tautan Git	54
MODUL 4	55
Soal 1.....	55
A. Source Code	55
B. Output Program	57
C. Pembahasan	58
Soal 2.....	59

A.	Source Code	60
B.	Output Program	62
C.	Pembahasan	63
Soal 3.....		64
A.	Source Code	65
B.	Output Program	68
C.	Pembahasan	68
Tautan Git		69
MODUL 5		69
Soal 1.....		69
A.	Source Code	71
B.	Output Program	103
C.	Pembahasan	105
Tautan Git		114
MODUL 6		114
Soal 1.....		114
A.	Source Code	115
B.	Output Program	156
C.	Pembahasan	158
Tautan Git		174

DAFTAR TABEL

Tabel 1. Tabel Soal 1 Modul 1	10
Tabel 2. Source Code Soal 1 Modul 1	10
Tabel 3. Tabel Soal 2 Modul 1	11
Tabel 4. Source Code Soal 2 Modul 1	13
Tabel 5. Tabel Soal 3 Modul 1	14
Tabel 6. Source Code Soal 3 Modul 1	14
Tabel 7. Source Code Soal 4 Modul 1	17
Tabel 8. Source Code Soal 5 Modul 1	20
Tabel 9. Source Code Soal 1 Modul 2	24
Tabel 10. Source Code Soal 2 Modul 2	28
Tabel 11. Source Code Soal 3 Modul 2	32
Tabel 12. Source Code Soal 4 Modul 2	36
Tabel 13. Source Code Soal 1 Modul 3	39
Tabel 14. Source Code Soal 2 Modul 3	42
Tabel 15. Source Code Soal 3 Modul 3	46
Tabel 16. Source Code Soal 4 Modul 3	49
Tabel 17. Source Code Soal 5 Modul 3	53
Tabel 18. Source Code Soal 1 Modul 4	57
Tabel 19. Table Soal 2 Modul 4.....	59
Tabel 20. Tabel Soal 2 Modul 4.....	60
Tabel 21. Source Code Soal 2 Modul 4	62
Tabel 22. Tabel Soal 3 Modul 4.....	64
Tabel 23. Tabel Soal 3 Modul 4.....	65
Tabel 24. Source Code Soal 3 Modul 4	68
Tabel 25. Source Code Koneksi.php Soal 1 Modul 5	72
Tabel 26. Source Code Buku.php Soal 1 Modul 5.....	75
Tabel 27. Source Code Formbuku.php Soal 1 Modul 5.....	79
Tabel 28. Source Code Formmember.php Soal 1 Modul 5	83
Tabel 29. Source Code Formpeminjaman.php Soal 1 Modul 5.....	86
Tabel 30. Source Code Index.php Soal 1 Modul 5	88
Tabel 31. Source Code Member.php Soal 1 Modul 5.....	92
Tabel 32. Source Code Model.php Soal 1 Modul 5.....	98
Tabel 33. Source Code Peminjaman.php Soal 1 Modul 5	102
Tabel 34. Source Code Routes.php Soal 1 Modul 6	115
Tabel 35. Source Code Buku.php di Controller Soal 1 Modul 6.....	119
Tabel 36. Source Code Home.php Soal 1 Modul 6	120
Tabel 37. Source Code Login.php Soal 1 Modul 6.....	122

Tabel 38. Source Code Register.php Soal 1 Modul 6	124
Tabel 39. Source Code Authfilter.php Soal 1 Modul 6	125
Tabel 40. Source Code vw_add.php Soal 1 Modul 6.....	129
Tabel 41. Source Code vw_buku.php Soal 1 Modul 6	135
Tabel 42. Source Code vw_edit.php Soal 1 Modul 6	139
Tabel 43. Source Code vw_login.php Soal 1 Modul 6	143
Tabel 44. Source Code vw_register.php Soal 1 Modul 6	147
Tabel 45. Source Code Bukumodel.php Soal 1 Modul 6.....	149
Tabel 46. Source Code Usermodel.php Soal 1 Modul 6.....	149
Tabel 47. Source Code Buku.php di Seeds Soal 1 Modul 6	150
Tabel 48. Source Code Perpustakaan.sql Soal 1 Modul 6	155

DAFTAR GAMBAR

Gambar 1. Screenshot Hasil Akhir Soal 1 Modul 1	10
Gambar 2. Screenshot Hasil Akhir Soal 2 Modul 1	13
Gambar 3. Screenshot Hasil Akhir Soal 3 Modul 1	15
Gambar 4. Gambar Soal 4 Modul 1	16
Gambar 5. Screenshot Hasil Akhir Soal 4 Modul 1	17
Gambar 6. Gambar Soal 5 Modul 1	19
Gambar 7. Screenshot Hasil Akhir Soal 5 Modul 1	21
Gambar 8. Gambar Soal 1 Modul 2	22
Gambar 9. Gambar Soal 1 Modul 2	22
Gambar 10. Screenshot Hasil Akhir Soal 1 Modul 2	24
Gambar 11. Gambar Soal 2 Modul 2	26
Gambar 12. Gambar Soal 2 Modul 2	26
Gambar 13. Gambar Soal 2 Modul 2	26
Gambar 14. Screenshot Hasil Akhir Soal 2 Modul 2	28
Gambar 15. Screenshot Hasil Akhir Soal 2 Modul 2	28
Gambar 16. Screenshot Hasil Akhir Soal 2 Modul 2	29
Gambar 17. Gambar Soal 3 Modul 2	30
Gambar 18. Screenshot Hasil Akhir Soal 3 Modul 2	33
Gambar 19. Gambar Soal 4 Modul 2	34
Gambar 20. Gambar Soal 4 Modul 2	35
Gambar 21. Screenshot Hasil Akhir Soal 4 Modul 2	36
Gambar 22. Gambar Soal 1 Modul 3	38
Gambar 23. Gambar Soal 1 Modul 3	38
Gambar 24. Screenshot Hasil Akhir Soal 1 Modul 3	39
Gambar 25. Screenshot Hasil Akhir Soal 1 Modul 3	39
Gambar 26. Gambar Soal 2 Modul 3	41
Gambar 27. Gambar Soal 2 Modul 3	41
Gambar 28. Screenshot Hasil Akhir Soal 2 Modul 3	43
Gambar 29. Screenshot Hasil Akhir Soal 2 Modul 3	43
Gambar 30. Gambar Soal 3 Modul 3	45
Gambar 31. Gambar Soal 3 Modul 3	45
Gambar 32. Screenshot Hasil Akhir Soal 2 Modul 3	46
Gambar 33. Screenshot Hasil Akhir Soal 2 Modul 3	46
Gambar 34. Gambar Soal 4 Modul 3	48
Gambar 35. Gambar Soal 4 Modul 3	48
Gambar 36. Gambar Soal 4 Modul 3	48
Gambar 37. Screenshot Hasil Akhir Soal 4 Modul 3	49

Gambar 38. Screenshot Hasil Akhir Soal 4 Modul 3.....	50
Gambar 39. Screenshot Hasil Akhir Soal 4 Modul 3.....	50
Gambar 40. Gambar Soal 5 Modul 3	52
Gambar 41. Gambar Soal 5 Modul 3	52
Gambar 42. Screenshot Hasil Akhir Soal 5 Modul 3.....	53
Gambar 43. Screenshot Hasil Akhir Soal 5 Modul 3.....	53
Gambar 44. Gambar Soal 1 Modul 4	55
Gambar 45. Gambar Soal 1 Modul 4	55
Gambar 46. Screenshot Hasil Akhir Soal 1 Modul 4.....	57
Gambar 47. Gambar Soal 2 Modul 4	60
Gambar 48. Screenshot Hasil Akhir Soal 2 Modul 4.....	62
Gambar 49. Screenshot Hasil Akhir Soal 3 Modul 4.....	68
Gambar 50. Gambar Soal 1 Modul 5	70
Gambar 51. Screenshot Hasil Akhir Soal 1 Modul 5.....	103
Gambar 52. Screenshot Hasil Akhir Soal 1 Modul 5.....	103
Gambar 53. Screenshot Hasil Akhir Soal 1 Modul 5.....	104
Gambar 54. Screenshot Hasil Akhir Soal 1 Modul 5.....	104
Gambar 55. Screenshot Hasil Akhir Soal 1 Modul 5.....	104
Gambar 56. Screenshot Hasil Akhir Soal 1 Modul 5.....	105
Gambar 57. Gambar Soal 1 Modul 6	115
Gambar 58. Screenshot Hasil Akhir Soal 1 Modul 6.....	156
Gambar 59. Screenshot Hasil Akhir Soal 1 Modul 6.....	156
Gambar 60. Screenshot Hasil Akhir Soal 1 Modul 6.....	157
Gambar 61. Screenshot Hasil Akhir Soal 1 Modul 6.....	157
Gambar 62. Screenshot Hasil Akhir Soal 1 Modul 6.....	157

MODUL 1

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}

Tabel 1. Tabel Soal 1 Modul 1

Simpan dengan nama file: PRAK101.php

A. Source Code

1	<?php
2	echo "Hello World in PHP ";
3	echo "Nama Praktikan : Muh Haekal Barera ";
4	echo "NIM : 2210817310008";
5	?>

Tabel 2. Source Code Soal 1 Modul 1

B. Output Program

Hello World in PHP
Nama Praktikan : Muh Haekal Barera
NIM : 2210817310008

Gambar 1. Screenshot Hasil Akhir Soal 1 Modul 1

C. Pembahasan

1. <?php ?>: Ini adalah tag pembuka dan penutup untuk blok kode PHP. Semua kode PHP harus berada di antara tag ini.
2. echo "Hello World in PHP
";: Ini adalah perintah PHP yang digunakan untuk mencetak teks "Hello World in PHP" ke layar. Perintah echo digunakan

untuk menampilkan teks atau variabel ke dalam output, sedangkan `
` adalah tag HTML untuk membuat baris baru (line break) di dalam output.

3. `echo "Nama Praktikan : Muh Haekal Barera
";` Ini adalah perintah PHP lainnya yang mencetak teks "Nama Praktikan : Muh Haekal Barera" ke layar, diikuti dengan tag HTML `
` untuk membuat baris baru.
4. `echo "NIM : 2210817310008";` Perintah PHP ini mencetak teks "NIM : 2210817310008" ke layar tanpa menambahkan tag HTML untuk 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

Tabel 3. Tabel Soal 2 Modul 1

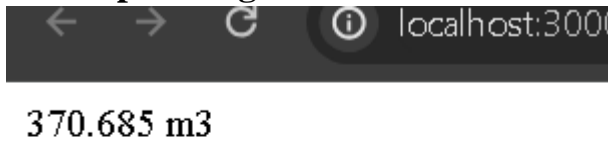
Simpan dengan nama file: PRAK102.php

A. Source Code

```
1 <?php
2
3 //variabel bangun ruang
4 $jarijari = 4.2;
5 $tinggi = 5.4;
6 $panjang = 8.9;
7 $lebar = 14.7;
8 $sisi = 7.9;
9 $alasPrisma = 5.5;
10 $tinggiPrisma = 7.5;
11 $tinggiLimas = 8.5;
12
13 //akhiran nim yang dimasukkan
14 $nim = 8;
15
16 //rumus bangun ruang
17 if( $nim == 0 || $nim == 1 ) {
18     $tabung = 3.14 * $jarijari * $jarijari * $tinggi;
19     echo number_format($tabung, 3);
20     echo " m3";
21 } else if ( $nim == 2 || $nim == 3 ) {
22     $kerucut = 1/3 * 3.14 * $jarijari * $jarijari * $tinggi;
23     echo number_format($kerucut, 3);
24     echo " m3";
25 } else if ( $nim == 4 || $nim == 5 ) {
26     $bola = 4/3 * 3.14 * $jarijari * $jarijari * $jarijari;
27     echo number_format($bola, 3);
28     echo " m3";
29 } else if ( $nim == 6 || $nim == 7 ) {
30     $prismaAlasSegitiga = 1/2 * $alasPrisma * $tinggiPrisma;
31     echo number_format($prismaAlasSegitiga, 3);
32     echo " m3";
33 } else if ( $nim == 8 || $nim == 9 ) {
34     $limasAlasPersegiPanjang = 1/3 * $panjang * $lebar * $tinggiLimas;
35     echo number_format($limasAlasPersegiPanjang, 3);
36     echo " m3";
37 }
38 ?>
```

Tabel 4. Source Code Soal 2 Modul 1

B. Output Program



Gambar 2. Screenshot Hasil Akhir Soal 2 Modul 1

C. Pembahasan

1. Variabel Bangun Ruang:
\$jarijari, \$tinggi, \$panjang, \$lebar, \$sisi, \$alasPrisma, dan \$tinggiPrisma adalah variabel yang menyimpan nilai-nilai yang akan digunakan dalam rumus perhitungan volume bangun ruang. Setiap variabel ini mewakili nilai jari-jari, tinggi, panjang, lebar, sisi, alas, dan tinggi dari bangun ruang yang sesuai.
2. Variabel NIM:
\$nim adalah variabel yang menyimpan dua digit terakhir dari NIM. Nilai ini digunakan untuk menentukan jenis bangun ruang yang akan dihitung volumenya.
3. Pengecekan NIM dan Perhitungan Volume:
Kode ini menggunakan struktur if-else if untuk mengecek nilai variabel \$nim dan kemudian menjalankan rumus perhitungan volume sesuai dengan jenis bangun ruang yang terkait.
Setiap blok if atau else if mengecek apakah nilai \$nim merupakan salah satu dari beberapa kategori. Jika iya, maka dilakukan perhitungan volume sesuai dengan rumus yang tepat untuk bangun ruang tersebut.
4. Rumus Perhitungan Volume:
Untuk setiap kategori bangun ruang, rumus perhitungan volume sesuai dengan bentuk bangun ruang yang diberikan.
Rumus-rumus untuk bangun ruang lainnya seperti kerucut, bola, prisma alas segitiga, dan limas alas persegi panjang dihitung secara serupa dengan menggantikan variabel yang sesuai.
5. Menampilkan Output:

Setiap hasil perhitungan volume ditampilkan menggunakan perintah echo, yang diikuti dengan hasil perhitungan volume yang telah diformat menggunakan fungsi number_format() untuk membatasi jumlah desimal menjadi tiga.

Setelah volume ditampilkan, satuan volume, yaitu meter kubik (m³), juga ditambahkan.

6. Penutup PHP:

Kode diakhiri dengan penutup PHP, yaitu ?>.

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

Tabel 5. Tabel Soal 3 Modul 1

Simpan dengan nama file: PRAK103.php

A. Source Code

1	<?php
2	
3	//variabel suhu
4	\$celcius = 37.841;
5	\$fahrenheit = (9/5) * \$celcius + 32;
6	\$reamur = (4/5) * \$celcius;
7	\$kelvin = \$celcius + 273.15;
8	
9	//output
10	echo "Fahrenheit (F) : \$fahrenheit ";
11	echo "Reamur (R) : \$reamur ";
12	echo "Kelvin (K): \$kelvin";
13	?>

Tabel 6. Source Code Soal 3 Modul 1

B. Output Program



Fahrenheit (F) : 100.1138

Reamur (R) : 30.2728

Kelvin (K): 310.991

Gambar 3. Screenshot Hasil Akhir Soal 3 Modul 1

C. Pembahasan

1. Variabel:

\$celcius: Menyimpan nilai suhu dalam Celcius.

\$fahrenheit: Menyimpan nilai suhu yang dikonversi ke Fahrenheit.

\$reamur: Menyimpan nilai suhu yang dikonversi ke Reamur.

\$kelvin: Menyimpan nilai suhu yang dikonversi ke Kelvin.

2. Perhitungan Konversi:

Rumus konversi suhu dari Celcius ke Fahrenheit, Reamur, dan Kelvin digunakan:

Fahrenheit: $(9/5) * \$celcius + 32$

Reamur: $(4/5) * \$celcius$

Kelvin: $\$celcius + 273.15$

3. Output:

Nilai suhu dalam Fahrenheit, Reamur, dan Kelvin ditampilkan di layar dengan echo.

Soal 4

Buatlah program dalam Bahasa pemrograman PHP yang menampilkan daftar data seperti ditunjukkan pada Gambar 1. 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

Gambar 4. Gambar Soal 4 Modul 1

Simpan dengan nama file: PRAK104.php

A. Source Code

```


1  <?php
2
3  //array
4  $daftarSmartphone = array("Samsung Galaxy S22", "Samsung Galaxy S22+",
5  "Samsung Galaxy A03", "Samsung Galaxy Xcover 5");
6  ?>
7
8  <!DOCTYPE html>
9  <html lang="en">
10 <head>
11   <meta charset="UTF-8">
12   <meta http-equiv="X-UA-Compatible" content="IE=edge">
13   <meta name="viewport" content="width=device-width, initial-scale=1.0">
14   <title>Prak104</title>
15   <style>
16     table {
17       font-family: sans-serif;
18       color: #232323;
19     }
20     table, th, td {
21       border: 1px solid;
22     }
23   </style>
24 </head>
25 <body>
26   <table>

```


27	<tr>
28	<th>Daftar Smartphone Samsung</th>
29	</tr>
30	<?php foreach(\$daftarSmartphone as \$ds) : ?>
31	<tr>
32	<td><?= \$ds; ?></td>
33	</tr>
34	<?php endforeach ?>
35	</table>
36	</body>
37	</html>

Tabel 7. Source Code Soal 4 Modul 1

B. Output Program



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

Gambar 5. Screenshot Hasil Akhir Soal 4 Modul 1

C. Pembahasan

1. Array Smartphone:
Variabel \$daftarSmartphone merupakan sebuah array yang berisi daftar smartphone merek Samsung. Setiap elemen array merupakan nama dari model smartphone Samsung.
2. Deklarasi Dokumen HTML:
Setelah deklarasi PHP, kode HTML dimulai dengan tag <!DOCTYPE html>, yang menandakan bahwa dokumen adalah dokumen HTML5.

Kemudian, terdapat tag `<html lang="en">` yang mendefinisikan bahasa dokumen sebagai bahasa Inggris.

3. Head Dokumen:

Bagian `<head>` dokumen HTML berisi meta tag dan title dari halaman.

Meta tag `charset` menentukan set karakter yang digunakan dalam dokumen sebagai UTF-8.

Meta tag `http-equiv="X-UA-Compatible"` mendefinisikan mode kompatibilitas untuk Internet Explorer.

Meta tag `viewport` mendefinisikan cara browser mengatur tampilan halaman pada perangkat yang berbeda.

Title dokumen adalah "Prak104".

4. CSS Styling:

Bagian `<style>` berisi aturan CSS untuk mengatur tampilan tabel.

CSS yang didefinisikan mengatur tampilan font dan warna teks, serta mengatur garis pembatas pada tabel.

5. Body Dokumen:

Bagian `<body>` dokumen HTML berisi struktur tabel yang akan ditampilkan.

Terdapat sebuah tabel dengan satu baris header dan baris-baris data yang diisi menggunakan data dari array `$daftarSmartphone`.

6. Penggunaan PHP dalam HTML:

Perulangan `foreach` digunakan untuk mengambil setiap elemen dari array `$daftarSmartphone`.

Setiap elemen dari array tersebut ditampilkan dalam sebuah baris tabel menggunakan tag HTML `<tr>` dan `<td>`.

Dalam baris tabel tersebut, data `smartphone` ditampilkan menggunakan sintaks `<?= $ds; ?>`, yang merupakan cara singkat untuk menampilkan

Soal 5

Buatlah program dalam Bahasa pemrograman PHP yang menampilkan daftar data seperti ditunjukkan pada Gambar 1. 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

Gambar 6. Gambar Soal 5 Modul 1

Simpan dengan nama file: PRAK105.php

A. Source Code

```

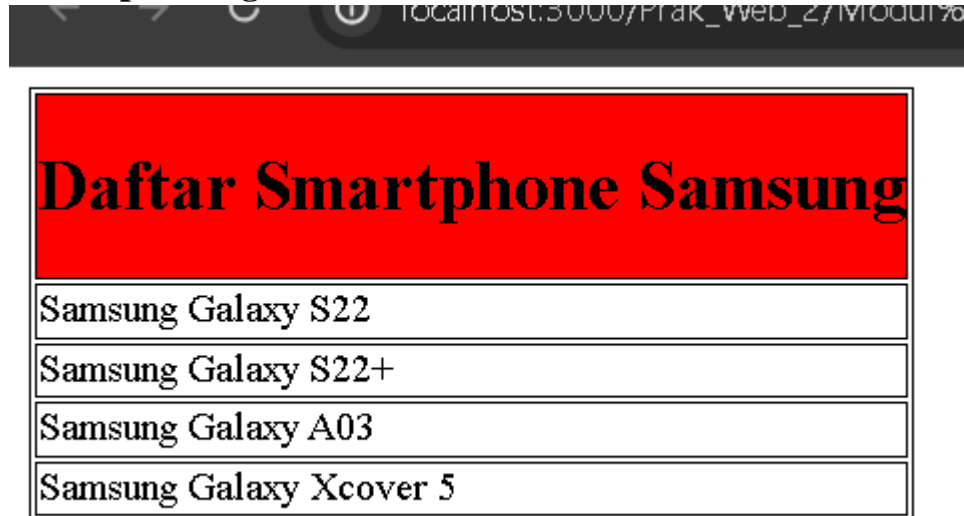
1  <?php
2
3  //array
4  $daftarSmartphone = array("S22" => "Samsung Galaxy S22", "S22+" =>
5  "Samsung Galaxy S22+", "A03" => "Samsung Galaxy A03", "Xcover5" =>
   "Samsung Galaxy Xcover 5");
6  ?>
7  <!DOCTYPE html>
8  <html lang="en">
9  <head>
10   <title>Prak105</title>
11   <style>
12     table {
13       color: black;
14     }
15     table, th, td {
16       border: 1px solid;
17     }
18     th {
19       background-color: red;
20       font-size: 28px;
21       padding: 20px 0px;
22     }

```

23	</style>
24	</head>
25	<body>
26	<table>
27	<tr>
28	<th>Daftar Smartphone Samsung</th>
29	</tr>
30	<tr>
31	<td><?= \$daftarSmartphone["S22"] ?></td>
32	</tr>
33	<tr>
34	<td><?= \$daftarSmartphone["S22+"] ?></td>
35	</tr>
36	<tr>
37	<td><?= \$daftarSmartphone["A03"] ?></td>
38	</tr>
39	<tr>
40	<td><?= \$daftarSmartphone["Xcover5"] ?></td>
41	</tr>
42	</table>
43	</body>
44	</html>

Tabel 8. Source Code Soal 5 Modul 1

B. Output Program



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

Gambar 7. Screenshot Hasil Akhir Soal 5 Modul 1

C. Pembahasan

1. Array Smartphone:
Variabel \$daftarSmartphone adalah sebuah array asosiatif yang berisi pasangan kunci-nilai. Kunci dari array ini adalah kode model (misalnya "S22", "S22+", "A03", "Xcover5"), sedangkan nilai-nilainya adalah nama model lengkap dari setiap smartphone.
2. Deklarasi Dokumen HTML:
Kode HTML dimulai setelah penutup PHP `?>` dengan deklarasi dokumen standar HTML5 (`<!DOCTYPE html>`).
Ada juga tag `<html lang="en">` yang menentukan bahasa dokumen sebagai bahasa Inggris.
3. Head Dokumen:
Bagian `<head>` dokumen HTML berisi judul halaman dan beberapa aturan CSS yang akan digunakan untuk styling tabel.
Title halaman diberi judul "Prak105".
Terdapat juga beberapa aturan CSS untuk mengatur tampilan tabel, seperti warna teks, border, dan latar belakang header.
4. Body Dokumen:
Bagian `<body>` dokumen HTML berisi struktur tabel yang akan ditampilkan.

Terdapat sebuah tabel dengan satu baris header yang menampilkan teks "Daftar Smartphone Samsung" dengan gaya yang telah ditentukan di CSS. Setiap elemen dari array \$daftarSmartphone ditampilkan dalam baris-baris tabel menggunakan tag HTML <tr> dan <td>. Dalam setiap baris tabel, nilai dari array \$daftarSmartphone diakses menggunakan kunci yang sesuai (misalnya, \$daftarSmartphone["S22"] untuk mendapatkan nama model "Samsung Galaxy S22").

Tautan Git

https://github.com/Vainn11/Prak_Web_2/blob/e7cd5a38b5d926f7405719cbf1ff37f0a8ab79a5/Modul%201/Prak105.php

MODUL 2

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	andry
Nama: 2	andre
Nama: 3	andra
Urutkan	

Gambar 8. Gambar Soal 1 Modul 2

Output
andra andre andry

Gambar 9. Gambar Soal 1 Modul 2

Simpan dengan nama file: PRAK201.php

A. Source Code

1	<!DOCTYPE html>
2	<html>
3	<head>
4	<title>Sorting Names</title>
5	</head>

6	<body>
7	
8	<h2>Masukkan Tiga Nama</h2>
9	
10	<form method="post" action="">
11	Nama 1: <input type="text" name="nama1">
12	Nama 2: <input type="text" name="nama2">
13	Nama 3: <input type="text" name="nama3">
14	<input type="submit" name="submit" value="Urutkan">
15	</form>
16	
17	<?php
18	// Cek apakah form telah disubmit
19	if(isset(\$_POST['submit'])) {
20	// Mengambil nilai dari inputan
21	\$nama1 = \$_POST['nama1'];
22	\$nama2 = \$_POST['nama2'];
23	\$nama3 = \$_POST['nama3'];
24	
25	// Memeriksa kondisi untuk mengurutkan nama
26	if(\$nama1 < \$nama2 && \$nama1 < \$nama3){
27	echo "Output: \$nama1 ";
28	if(\$nama2 < \$nama3){
29	echo "\$nama2 \$nama3 ";
30	} else {
31	echo "\$nama3 \$nama2 ";
32	}
33	} elseif(\$nama2 < \$nama1 && \$nama2 < \$nama3){
34	echo "Output: \$nama2 ";
35	if(\$nama1 < \$nama3){
36	echo "\$nama1 \$nama3 ";
37	} else {
38	echo "\$nama3 \$nama1 ";
39	}
40	} else {
41	echo "Output: \$nama3 ";
42	if(\$nama1 < \$nama2){
43	echo "\$nama1 \$nama2 ";
44	} else {

45	echo "\$nama2 \$nama1 ";
46	}
47	}
48	}
49	?>
50	
51	</body>
52	</html>

Tabel 9. Source Code Soal 1 Modul 2

B. Output Program

Masukkan Tiga Nama

Nama 1: andy
 Nama 2: andre
 Nama 3: anahy

Output:
 andy
 andre
 anahy

Gambar 10. Screenshot Hasil Akhir Soal 1 Modul 2

C. Pembahasan

1. <!DOCTYPE html>: Mendefinisikan tipe dokumen HTML yang digunakan.
2. <html>...</html>: Mengawali dan mengakhiri elemen HTML, yang berisi seluruh kode HTML.
3. <head>...</head>: Bagian kepala dokumen HTML, di mana informasi seperti judul halaman, meta tag, atau tautan stylesheet ditempatkan. Dalam hal ini, hanya judul halaman yang ditampilkan.
4. <title>Sorting Names</title>: Menentukan judul halaman yang akan ditampilkan di tab browser.
5. <body>...</body>: Bagian tubuh dokumen HTML, di mana konten yang dapat dilihat pengguna ditempatkan.
6. <h2>Masukkan Tiga Nama</h2>: Menampilkan judul "Masukkan Tiga Nama" sebagai heading level 2 di halaman web.
7. <form method="post" action="">...</form>: Membuat formulir dengan metode post yang akan mengirimkan data ke halaman itu sendiri ketika

formulir tersebut disubmit. Formulir ini memiliki tiga input untuk memasukkan tiga nama, dengan nama input berturut-turut nama1, nama2, dan nama3. Tombol "Urutkan" juga disediakan untuk mengirimkan formulir.

8. `<?php ... ?>`: Blok PHP yang dimulai setelah formulir. Kode PHP di dalamnya akan dieksekusi saat halaman dimuat kembali setelah formulir disubmit.
9. `if(isset($_POST['submit'])) { ... }`: Memeriksa apakah formulir telah disubmit dengan memeriksa apakah tombol submit dengan nama submit telah diklik.
10. `$nama1 = $_POST['nama1']; $nama2 = $_POST['nama2']; $nama3 = $_POST['nama3'];`: Mengambil nilai dari inputan formulir yang disubmit oleh pengguna.
11. Kode PHP selanjutnya memeriksa dan mengurutkan tiga nama yang dimasukkan pengguna. Setelah itu, menampilkan hasil pengurutan tersebut dengan menampilkan "Output:" diikuti dengan nama-nama yang sudah diurutkan sesuai dengan urutan abjad.
12. Struktur kondisional dalam kode PHP memeriksa setiap kemungkinan urutan nama, yaitu \$nama1, \$nama2, dan \$nama3, kemudian menampilkan nama-nama tersebut sesuai dengan urutan abjad.
13. Dalam kondisi pertama, kode memeriksa apakah \$nama1 merupakan nama terkecil. Jika ya, maka nama tersebut ditampilkan pertama dengan format tebal **Output:**, diikuti dengan \$nama2 dan \$nama3 yang diurutkan berdasarkan abjad.
14. Kondisi kedua memeriksa apakah \$nama2 merupakan nama terkecil. Jika ya, maka nama tersebut ditampilkan pertama dengan format tebal **Output:**, diikuti dengan \$nama1 dan \$nama3 yang diurutkan berdasarkan abjad.
15. Jika kondisi pertama dan kedua tidak terpenuhi, maka kondisi ketiga akan dijalankan, yang berarti \$nama3 merupakan nama terkecil di antara ketiga nama yang dimasukkan. Dalam hal ini, nama tersebut ditampilkan pertama dengan format tebal **Output:**, diikuti dengan \$nama1 dan \$nama2 yang diurutkan berdasarkan abjad.
16. Setelah tiga nama diurutkan dan ditampilkan, proses PHP selesai dan halaman HTML akan menampilkan hasil tersebut di bawah formulir.

Soal 2

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

Nama: *

Nim: *

Jenis Kelamin : *

☐ Laki-Laki

☐ Perempuan

Gambar 11. Gambar Soal 2 Modul 2

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

Gambar 12. Gambar Soal 2 Modul 2

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

Nama: *

Nim: *

Jenis Kelamin : *

☒ Laki-Laki

☐ Perempuan

Output:

Muhammad Fulan
123456
Laki-laki

Gambar 13. Gambar Soal 2 Modul 2

Simpan dengan nama file: PRAK202.php

A. Source Code

1	<!DOCTYPE html>
2	<html>
3	<head>
4	<style>
5	.error { color: red; }

6	</style>
7	<title>Soal_2_Modul_2</title>
8	</head>
9	<body>
10	<?php
11	\$namaError = \$nimError = \$genderError = "";
12	if(isset(\$_POST["submit"])){
13	if (empty(\$_POST["nama"])) {
14	\$namaError = "nama tidak boleh kosong";
15	}
16	if (empty(\$_POST["nim"])) {
17	\$nimError = "nim tidak boleh kosong";
18	}
19	if (empty(\$_POST["gender"])) {
20	\$genderError = "jenis kelamin tidak boleh kosong";
21	}
22	}?>
23	
24	<form action="" method="post">
25	Nama: <input type="text" name="nama"
26	value="<?=isset(\$_POST['nama']) ? \$_POST['nama'] : ">">
27	* <?php echo \$namaError;?>
28	Nim: <input type="text" name="nim" value="<?=isset(\$_POST['nim']) ?
29	\$_POST['nim'] : ">">
30	* <?php echo \$nimError;?>
31	Jenis Kelamin: * <?php echo
32	\$genderError;?>
33	<input type="radio" name="gender" value="Laki-laki" <?php if
34	(isset(\$_POST["gender"]) and \$_POST["gender"] == "Laki-laki") echo
35	"checked";?>>Laki-laki
36	<input type="radio" name="gender" value="Perempuan" <?php if
37	(isset(\$_POST["gender"]) and \$_POST["gender"] == "Perempuan") echo
38	"checked";?>>Perempuan
39	<button type="submit" name="submit">Submit</button>
40	</form>
41	<?php
42	if (isset(\$_POST["submit"])) {
43	if (!empty(\$_POST["nama"]) and !empty(\$_POST["nim"]) and
44	!empty(\$_POST["gender"])) {

37	echo "<h1>Output: </h1>";
38	if(!empty(\$_POST["nama"])){
39	echo \$_POST["nama"]." ";
40	}
41	if(!empty(\$_POST["nim"])){
42	echo \$_POST["nim"]." ";
43	}
44	if(!empty(\$_POST["gender"])) {
45	echo \$_POST["gender"];
46	}
47	}
48	}?>
49	</body>
50	</html>

Tabel 10. Source Code Soal 2 Modul 2

B. Output Program

Nama: *

Nim: *

Jenis Kelamin: *

☐ Laki-laki

☐ Perempuan

Gambar 14. Screenshot Hasil Akhir Soal 2 Modul 2

Nama: * nama tidak boleh kosong

Nim: * nim tidak boleh kosong

Jenis Kelamin: *

☒ Laki-laki

☐ Perempuan

Gambar 15. Screenshot Hasil Akhir Soal 2 Modul 2

Nama: ✖

Nim: ✖

Jenis Kelamin: ✖

☒ Laki-laki

☐ Perempuan

Output:

Muhammad Fulan
123456
Laki-laki

Gambar 16. Screenshot Hasil Akhir Soal 2 Modul 2

C. Pembahasan

1. `<!DOCTYPE html>`: Mendefinisikan jenis dokumen HTML.
2. `<html>`: Menandai awal dari dokumen HTML.
3. `<head>`: Bagian kepala dokumen HTML, di mana informasi-informasi meta dan pengaturan halaman web diletakkan.
4. `<style>`: Bagian untuk mendefinisikan gaya CSS.
5. `.error {color: red;}`: Mendefinisikan gaya CSS untuk kelas CSS `.error`, yang akan mewarnai teks dengan warna merah.
6. `<title>Soal_2_Modul_2</title>`: Menentukan judul halaman.
7. `<body>`: Bagian isi dari halaman web.
8. `<?php ?>`: Penanda bahwa kode PHP dimulai.
9. Variabel-variabel seperti `$namaError`, `$nimError`, dan `$genderError` digunakan untuk menyimpan pesan kesalahan jika input tidak valid.
10. `isset($_POST["submit"])`: Mengecek apakah tombol submit telah ditekan.
11. `empty($_POST["nama"])`, `empty($_POST["nim"])`, `empty($_POST["gender"])`: Memeriksa apakah input nama, NIM, dan jenis kelamin kosong atau tidak. Jika kosong, pesan kesalahan akan disimpan dalam variabel yang sesuai.
12. `<form action="" method="post">`: Membuat formulir dengan metode POST yang akan mengirimkan data ke halaman itu sendiri.
13. `<input type="text" name="nama" value="<?==isset($_POST['nama']) ? $_POST['nama'] : ">">`: Kotak input untuk nama dengan nilai awal yang diambil dari data yang sudah di-submit sebelumnya jika ada.
14. `* <?php echo $namaError;?>`: Menampilkan pesan kesalahan jika input nama kosong.

15. `<input type="radio" name="gender" value="Laki-laki" <?php if (isset($_POST["gender"]) and $_POST["gender"] == "Laki-laki") echo "checked";?>>Laki-laki: Tombol radio untuk memilih jenis kelamin Laki-laki dengan cek otomatis jika jenis kelamin tersebut telah dipilih sebelumnya.`
16. `<button type="submit" name="submit">Submit</button>`: Tombol submit untuk mengirimkan formulir.
17. Pada akhirnya, ada bagian PHP lagi untuk menampilkan output jika semua input telah diisi dengan benar. Jika tidak, maka pesan kesalahan akan muncul di sebelah field input yang sesuai.

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

Gambar 17. Gambar Soal 3 Modul 2

Simpan dengan nama file: PRAK203.php

A. Source Code

1	<code><!DOCTYPE html></code>
2	<code><html></code>
3	<code><head></code>
4	<code> <title>Konversi Suhu</title></code>
5	<code></head></code>
6	<code><body></code>
7	

8	<h2>Konversi Suhu</h2>
9	
10	<form method="post" action="">
11	Masukkan Suhu: <input type="number" name="suhu" required>
12	Dari :
13	<input type="radio" name="from_unit" value="C"> Celcius
14	<input type="radio" name="from_unit" value="F"> Fahrenheit
15	<input type="radio" name="from_unit" value="Re"> Reamur
16	<input type="radio" name="from_unit" value="K"> Kelvin
17	Ke :
18	<input type="radio" name="to_unit" value="C"> Celcius
19	<input type="radio" name="to_unit" value="F"> Fahrenheit
20	<input type="radio" name="to_unit" value="Re"> Reamur
21	<input type="radio" name="to_unit" value="K"> Kelvin
22	<input type="submit" name="submit" value="Konversi">
23	</form>
24	
25	<?php
26	if(isset(\$_POST['submit'])){
27	\$suhu = \$_POST['suhu'];
28	\$from_unit = \$_POST['from_unit'];
29	\$to_unit = \$_POST['to_unit'];
30	
31	// Konversi suhu
32	switch(\$from_unit){
33	case 'C':
34	\$C = \$suhu;
35	break;
36	case 'F':
37	\$C = (\$suhu - 32) * (5/9);
38	break;
39	case 'Re':
40	\$C = \$suhu * (5/4);
41	break;
42	case 'K':
43	\$C = \$suhu - 273.15;
44	break;
45	}
46	

47	// Konversi ke unit yang diinginkan
48	switch(\$to_unit){
49	case 'C':
50	\$hasil = \$C;
51	break;
52	case 'F':
53	\$hasil = (\$C * 9/5) + 32;
54	break;
55	case 'Re':
56	\$hasil = \$C * (4/5);
57	break;
58	case 'K':
59	\$hasil = \$C + 273.15;
60	break;
61	}
62	
63	// Tampilkan hasil konversi
64	echo "Hasil konversi: \$hasil °\$to_unit";
65	}
66	?>
67	
68	</body>
69	</html>

Tabel 11. Source Code Soal 3 Modul 2

B. Output Program



Konversi Suhu

Masukkan Suhu:

Dari :

☒ Celcius
☐ Fahrenheit
☐ Reamur
☐ Kelvin

Ke :

☐ Celcius
☒ Fahrenheit
☐ Reamur
☐ Kelvin

Hasil konversi: 136.4 °F

Gambar 18. Screenshot Hasil Akhir Soal 3 Modul 2

C. Pembahasan

1. `<!DOCTYPE html>`: Mendefinisikan tipe dokumen HTML yang digunakan.
2. `<html>...</html>`: Mengawali dan mengakhiri elemen HTML, yang berisi seluruh kode HTML.
3. `<head>...</head>`: Bagian kepala dokumen HTML, di mana informasi seperti judul halaman, meta tag, atau tautan stylesheet ditempatkan. Dalam hal ini, hanya judul halaman yang ditampilkan.
4. `<title>Konversi Suhu</title>`: Menentukan judul halaman yang akan ditampilkan di tab browser.
5. `<body>...</body>`: Bagian tubuh dokumen HTML, di mana konten yang dapat dilihat pengguna ditempatkan.
6. `<h2>Konversi Suhu</h2>`: Menampilkan judul "Konversi Suhu" sebagai heading level 2 di halaman web.
7. `<form method="post" action="">...</form>`: Membuat formulir dengan metode post yang akan mengirimkan data ke halaman itu sendiri ketika formulir tersebut disubmit. Formulir ini memiliki input untuk memasukkan suhu, pilihan unit suhu awal (from_unit), dan pilihan unit suhu tujuan (to_unit). Tombol "Konversi" juga disediakan untuk mengirimkan formulir.
8. Dalam formulir, terdapat radio button untuk memilih unit suhu awal (from_unit) dan tujuan (to_unit), serta input untuk memasukkan suhu yang akan dikonversi.
9. `<?php ... ?>`: Blok PHP yang dimulai setelah formulir. Kode PHP di dalamnya akan dieksekusi saat halaman dimuat kembali setelah formulir disubmit.

10. `if(isset($_POST['submit'])) { ... }`: Memeriksa apakah formulir telah disubmit dengan memeriksa apakah tombol submit dengan nama submit telah diklik.
11. `$suhu = $_POST['suhu'];, $from_unit = $_POST['from_unit'];, $to_unit = $_POST['to_unit'];`: Mengambil nilai dari inputan formulir yang disubmit oleh pengguna.
12. Kode PHP kemudian melakukan konversi suhu dari unit awal ke unit tujuan yang dipilih pengguna menggunakan struktur switch-case. Setiap case mengkonversi suhu ke unit Celcius terlebih dahulu (disebut sebagai \$C), dan kemudian konversi dilanjutkan ke unit tujuan yang diinginkan.
13. Hasil konversi suhu kemudian ditampilkan dalam elemen HTML dengan format tebal menggunakan tag ``, bersama dengan unit suhu tujuan.
14. Setelah melakukan konversi suhu, hasilnya disimpan dalam variabel \$hasil.
15. Kemudian, hasil konversi tersebut ditampilkan di dalam halaman HTML dengan menggunakan tag `` untuk memberikan efek tebal pada teks, dan ditambahkan unit suhu tujuan yang dipilih oleh pengguna.

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
100	Anda Menginput Melebihi Limit Bilangan
632	Ratusan
13	Belasan

Gambar 19. Gambar Soal 4 Modul 2

Output yang diinginkan :

Nilai :

Hasil: belasan

Gambar 20. Gambar Soal 4 Modul 2

Simpan dengan nama file: PRAK204.php

A. Source Code

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>Document</title>
7 </head>
8 <body>
9   <form method="post" action="">
10     Nilai : <input type="number" name="nilai"> <br>
11     <input type="submit" name="submit" value="Konversi">
12   </form>
13   <?php
14   if(isset($_POST['submit'])){
15     $nilai = $_POST['nilai'];
16
17     if ($nilai == 0) {
18       echo "<b>Hasil : Nol</b>";
19       return false;
20     }
21     elseif ($nilai >= 1 && $nilai < 10) {
22       echo "<b>Hasil : Satuan</b>";
23       return false;
24     }
25     elseif ($nilai >= 10 && $nilai < 100) {
26       echo "<b>Hasil : Belasan</b>";
27       return false;
28     }
29     elseif ($nilai >= 100 && $nilai < 1000) {
30       echo "<b>Hasil : Ratusan</b>";
```

31	return false;
32	}
33	else {
34	echo "Anda Menginput Melebihi Limit Bilangan";
35	return false;
36	}
37	}
38	?>
39	</body>
40	</html>

Tabel 12. Source Code Soal 4 Modul 2

B. Output Program

Nilai :

Hasil : Belasan

Gambar 21. Screenshot Hasil Akhir Soal 4 Modul 2

C. Pembahasan

1. <!DOCTYPE html>: Mendefinisikan tipe dokumen HTML yang digunakan.
2. <html lang="en">: Mengawali elemen HTML dengan menentukan bahasa (English) yang digunakan di dalam dokumen.
3. <head>...</head>: Bagian kepala dokumen HTML, di mana meta tag, judul halaman, atau tautan stylesheet ditempatkan.
4. Dalam bagian kepala, terdapat <meta charset="UTF-8"> yang menentukan pengkodean karakter UTF-8 untuk mendukung karakter internasional, dan <meta name="viewport" content="width=device-width, initial-scale=1.0"> yang mengatur tampilan responsif pada perangkat seluler dengan lebar yang sama dengan lebar perangkat dan skalanya 1:1.
5. <title>Document</title>: Menentukan judul halaman.
6. <body>...</body>: Bagian tubuh dokumen HTML, di mana konten yang dapat dilihat pengguna ditempatkan.
7. <form method="post" action="">...</form>: Membuat formulir dengan metode post yang akan mengirimkan data ke halaman itu sendiri ketika

formulir tersebut disubmit. Formulir ini memiliki input untuk memasukkan nilai dan tombol "Konversi" untuk mengirimkan formulir.

8. Input nilai diberi nama nilai dengan tipe data number, yang membatasi input hanya bisa menerima nilai numerik.
9. `<?php ... ?>`: Blok PHP yang dimulai setelah formulir. Kode PHP di dalamnya akan dieksekusi saat halaman dimuat kembali setelah formulir disubmit.
10. `if(isset($_POST['submit'])) { ... }`: Memeriksa apakah formulir telah disubmit dengan memeriksa apakah tombol submit dengan nama submit telah diklik.
11. `$nilai = $_POST['nilai'];`: Mengambil nilai dari inputan formulir yang disubmit oleh pengguna.
12. Setelah itu, dilakukan serangkaian kondisi if-elseif-else untuk mengecek kategori nilai yang dimasukkan oleh pengguna. Jika nilai adalah 0, 1-9, 10-99, atau 100-999, pesan yang sesuai akan ditampilkan. Jika nilai yang dimasukkan melebihi 999, maka akan muncul pesan bahwa nilai tersebut melebihi batas.
13. Pesan hasil kategori nilai tersebut kemudian ditampilkan di dalam halaman HTML dengan menggunakan tag `` untuk memberikan efek tebal pada teks.

Tautan Git

https://github.com/Vainn11/Prak_Web_2/blob/f1b03425110326d2b398b30b02ce6562a22462ac/Modul%202/Prak204.php

MODUL 3

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

Gambar 22. Gambar Soal 1 Modul 3

Contoh Output 2:

Jumlah Peserta :

Peserta ke-1

Peserta ke-2

Peserta ke-3

Gambar 23. Gambar Soal 1 Modul 3

Simpan dengan nama file: PRAK301.php

A. Source Code

```
1 <?php $i = 1; ?>
2 <html>
3 <head>
4   <title>Soal_1_Modul_3</title>
5 </head>
6 <body>
7   <form action="" method="post">
8     <label for="jumlah">Jumlah Peserta :</label>
9     <input type="text" name="jumlah"> <br>
10    <button type="submit" name="submit">Cetak</button>
11  </form>
12 <?php if( isset($_POST['submit']) ) : ?>
```

13	<?php while(\$i <= \$_POST['jumlah']) { ?>
14	<?php if(\$i % 2 == 1) { ?>
15	<h1 style="color: red">Peserta ke-<?= \$i; ?></h1>
16	<?php } else { ?>
17	<h1 style="color: green">Peserta ke-<?= \$i; ?></h1>
18	<?php } ?>
19	<?php \$i = \$i + 1; ?>
20	<?php } ?>
21	<?php endif; ?>
22	</body>
23	</html>

Tabel 13. Source Code Soal 1 Modul 3

B. Output Program

Jumlah Peserta :

Peserta ke-1

Peserta ke-2

Peserta ke-3

Peserta ke-4

Gambar 24. Screenshot Hasil Akhir Soal 1 Modul 3

Jumlah Peserta :

Peserta ke-1

Peserta ke-2

Peserta ke-3

Gambar 25. Screenshot Hasil Akhir Soal 1 Modul 3

C. Pembahasan

1. `<?php $i = 1; ?>`: Baris ini mendefinisikan variabel `$i` dengan nilai awal 1. Variabel ini akan digunakan untuk iterasi dalam menampilkan peserta.
2. `<html>`, `<head>`, dan `<title>`: Ini adalah bagian awal dari markup HTML yang mendefinisikan struktur halaman web dan judul halaman.
3. `<form action="" method="post">`: Ini adalah tag form HTML yang mengarah ke halaman itu sendiri dan menggunakan metode POST untuk mengirimkan data.
4. `<label for="jumlah">Jumlah Peserta </label>`: Label untuk input jumlah peserta.
5. `<input type="text" name="jumlah">`: Input untuk memasukkan jumlah peserta.
6. `<button type="submit" name="submit">Cetak</button>`: Tombol submit untuk mengirimkan form.
7. `<?php if(isset($_POST['submit'])) : ?>`: Memeriksa apakah tombol submit telah ditekan.
8. `<?php while($i <= $_POST['jumlah']) { ?>`: Memulai loop while. Ini akan berjalan selama nilai `$i` kurang dari atau sama dengan jumlah peserta yang dimasukkan oleh pengguna melalui form.
9. `<?php if($i % 2 == 1) { ?>`: Memeriksa apakah nilai `$i` adalah ganjil.
10. `<h1 style="color: red">Peserta ke-<?= $i; ?></h1>`: Jika nilai `$i` ganjil, maka teks "Peserta ke-" diikuti dengan nomor peserta (nilai `$i`) akan ditampilkan dengan warna merah.
11. `<?php } else { ?>`: Bagian alternatif dari kondisi jika nilai `$i` bukan ganjil.
12. `<h1 style="color: green">Peserta ke-<?= $i; ?></h1>`: Jika nilai `$i` bukan ganjil (genap), maka teks "Peserta ke-" diikuti dengan nomor peserta (nilai `$i`) akan ditampilkan dengan warna hijau.
13. `<?php $i = $i + 1; ?>`: Menaikkan nilai `$i` setiap kali loop dilakukan agar iterasi ke peserta berikutnya.
14. `<?php } ?>`: Menutup blok if-else.
15. `<?php } ?>`: Menutup loop while.
16. `<?php endif; ?>`: Menutup blok if untuk memeriksa apakah tombol submit telah ditekan.

Soal 2

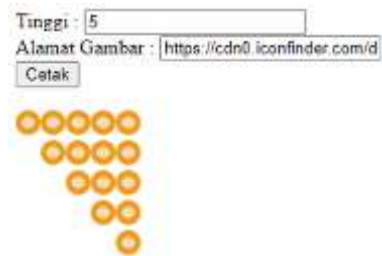
Buatlah program yang dapat menampilkan bentuk segitiga dari gambar yang dimasukkan. Input baris pertama adalah tinggi segitiga. Input berikutnya adalah alamat gambar yang akan digunakan untuk membangun segitiga. Gunakan link berikut sebagai inputan gambar : <https://cdn0.iconfinder.com/data/icons/web-and-mobile-icons-volume-2/128/52-512.png>.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

Contoh Output 1:



Gambar 26. Gambar Soal 2 Modul 3

Contoh Output 2:



Gambar 27. Gambar Soal 2 Modul 3

Simpan dengan nama file: PRAK302.php

A. Source Code

```
1 <html>
2 <head>
3   <title>Soal_2_Modul_3</title>
4 </head>
5 <body>
6   <form action="" method="post">
7     <label for="jumlah">Tinggi :</label>
8     <input type="text" name="jumlah"> <br>
9     <label for="alamat">Alamat Gambar :</label>
10    <input type="text" name="alamat"> <br>
```

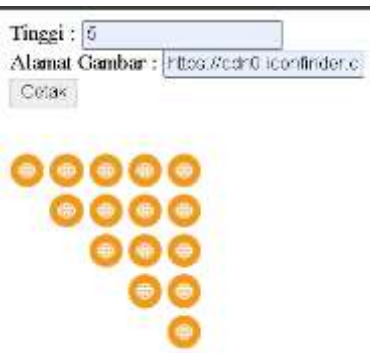
11	<button type="submit" name="submit">Cetak</button>
12	</form>
13	
14	<?php
15	if(isset(\$_POST['submit'])) {
16	\$max = \$_POST['jumlah'];
17	\$i = 1; \$j = 1; \$k = \$max;
18	\$gambar = \$_POST['alamat'];
19	} ?>
20	<?php if(isset(\$_POST['submit'])) : ?>
21	<?php while(\$i <= \$max) { ?>
22	<?php while(\$j < \$i) { ?>
23	<img style="width: 25px; opacity: 0;" src=<?= "\$gambar"; ?>
24	alt="">
25	<?php \$j = \$j + 1; ?>
26	<?php } ?>
27	<?php while(\$k >= \$i) { ?>
28	<img style="width : 25px" src=<?= "\$gambar"; ?> alt="">
29	<?php \$k = \$k - 1; ?>
30	<?php } ?>
31	
32	<?php
33	\$j = 1; \$k = \$max;
34	\$i = \$i + 1;
35	?>
36	<?php } ?>
37	<?php endif; ?>
38	</body>
	</html>

Tabel 14. Source Code Soal 2 Modul 3

B. Output Program



Gambar 28. Screenshot Hasil Akhir Soal 2 Modul 3



Gambar 29. Screenshot Hasil Akhir Soal 2 Modul 3

C. Pembahasan

1. `<html>`, `<head>`, dan `<title>`: Ini adalah bagian awal dari markup HTML yang mendefinisikan struktur halaman web dan judul halaman.
2. `<form action="" method="post">`: Ini adalah tag form HTML yang mengarah ke halaman itu sendiri dan menggunakan metode POST untuk mengirimkan data.
3. `<label for="jumlah">Tinggi :</label>` dan `<input type="text" name="jumlah">`: Label dan input untuk memasukkan tinggi pola gambar.
4. `<label for="alamat">Alamat Gambar :</label>` dan `<input type="text" name="alamat">`: Label dan input untuk memasukkan alamat gambar yang akan ditampilkan.
5. `<button type="submit" name="submit">Cetak</button>`: Tombol submit untuk mengirimkan form.
6. `<?php if(isset($_POST['submit'])) { ... } ?>`: Memeriksa apakah tombol submit telah ditekan. Jika ya, maka inisialisasi variabel `$max`, `$i`, `$j`, dan `$k`, serta mengambil nilai alamat gambar dari input.

7. `<?php while($i <= $max) { ?>`: Memulai loop while pertama. Ini akan berjalan selama nilai \$i kurang dari atau sama dengan nilai \$max, yang merupakan tinggi pola gambar.
8. `<?php while($j < $i) { ?>`: Memulai loop while kedua. Ini akan berjalan selama nilai \$j kurang dari \$i.
9. `<img style="width: 25px; opacity: 0;" src=<?=" $gambar"; ?> alt="">`: Tag `` untuk menampilkan gambar. Dalam kasus ini, gambar akan memiliki lebar 25px dan opacity (kejelasan) 0 (tidak terlihat), sesuai dengan pola yang dibuat.
10. `<?php $j = $j + 1; ?>`: Menaikkan nilai \$j setiap kali loop dilakukan.
11. `<?php while($k >= $i) { ?>`: Memulai loop while ketiga. Ini akan berjalan selama nilai \$k lebih besar dari atau sama dengan \$i.
12. `<img style="width : 25px" src=<?=" $gambar"; ?> alt="">`: Tag `` untuk menampilkan gambar. Dalam kasus ini, gambar akan memiliki lebar 25px.
13. `<?php $k = $k - 1; ?>`: Mengurangi nilai \$k setiap kali loop dilakukan.
14. `
`: Tag `
` untuk membuat baris baru setelah setiap baris gambar selesai.
15. `<?php $j = 1; $k = $max; $i = $i + 1; ?>`: Mengatur kembali nilai \$j dan \$k ke nilai awal, dan menaikkan nilai \$i untuk melanjutkan ke baris gambar berikutnya.
16. `<?php } ?>`: Menutup loop while ketiga.
17. `<?php } ?>`: Menutup loop while pertama.
18. `<?php endif; ?>`: Menutup blok if untuk memeriksa apakah tombol submit telah ditekan.

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

Gambar 30. Gambar Soal 3 Modul 3

Contoh Output 2:

Batas Bawah :

Batas Atas :

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

Gambar 31. Gambar Soal 3 Modul 3

Simpan dengan nama file: PRAK303.php

A. Source Code

```

1  <html>
2  <head>
3      <title>Soal_3_Mdul_3</title>
4  </head>
5  <body>
6      <form action="" method="post">
7          <label for="bawah">Batas Bawah :</label>
8          <input type="text" name="bawah"> </br>
9          <label for="atas">Batas Atas :</label>
10         <input type="text" name="atas"> </br>
11         <button type="submit" name="submit">Cetak</button>
12     </form>
13     <br>
14     <?php
15         $i; $j;
16     ?>
17     <?php if( isset($_POST['submit']) ) : ?>
18         <?php
19             $i = $_POST['bawah'];
20             $j = $_POST['atas'];
21         ?>

```

22	<?php do { ?>
23	<?php if((\$i + 7) % 5 == 0) { ?>
24	
25	<?php } else { ?>
26	<?= \$i; ?>
27	<?php } ?>
28	<?php \$i = \$i + 1; ?>
29	<?php } while(\$i <= \$j) ?>
30	<?php endif; ?>
31	</body>
32	</html>

Tabel 15. Source Code Soal 3 Modul 3

B. Output Program

Batas Bawah :

Batas Atas :

1 2 ★ 4 5 6 7 ★ 9 10

Gambar 32. Screenshot Hasil Akhir Soal 2 Modul 3

Batas Bawah :

Batas Atas :

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

Gambar 33. Screenshot Hasil Akhir Soal 2 Modul 3

C. Pembahasan

1. <html>, <head>, dan <title>: Bagian standar dari markup HTML yang mendefinisikan struktur halaman web dan judul halaman.
2. <form action="" method="post">: Tag form HTML untuk mengirimkan data ke halaman itu sendiri menggunakan metode POST.
3. <label for="bawah">Batas Bawah:</label> dan <input type="text" name="bawah">: Label dan input untuk memasukkan nilai batas bawah.

4. `<label for="atas">Batas Atas :</label>` dan `<input type="text" name="atas">`: Label dan input untuk memasukkan nilai batas atas.
5. `<button type="submit" name="submit">Cetak</button>`: Tombol submit untuk mengirimkan form.
6. `<?php $i; $j; ?>`: Ini adalah inisialisasi variabel \$i dan \$j, meskipun tidak diberikan nilai awal.
7. `<?php if(isset($_POST['submit'])) : ?>`: Memeriksa apakah tombol submit telah ditekan.
8. `<?php $i = $_POST['bawah']; $j = $_POST['atas']; ?>`: Mengambil nilai batas bawah dan batas atas yang dimasukkan oleh pengguna dari form.
9. `<?php do { ?>`: Memulai loop do-while. Ini akan terus dijalankan sekali, kemudian memeriksa kondisi di akhir iterasi.
10. `<?php if(($i + 7) % 5 == 0) { ?>`: Memeriksa apakah nilai \$i + 7 memiliki sisa bagi 5 sama dengan 0. Jika ya, maka akan menampilkan sebuah gambar bintang.
11. ``: Tag `` untuk menampilkan gambar bintang dengan lebar 20px.
12. `<?php } else { ?>`: Bagian alternatif dari kondisi jika nilai \$i + 7 tidak memenuhi syarat.
13. `<?= $i; ?>`: Ini akan menampilkan nilai \$i.
14. `<?php $i = $i + 1; ?>`: Menaikkan nilai \$i setiap kali loop dilakukan.
15. `<?php } while($i <= $j) ?>`: Menutup loop do-while dan memeriksa apakah nilai \$i masih kurang dari atau sama dengan nilai \$j.
16. `<?php endif; ?>`: Menutup blok if untuk memeriksa apakah tombol submit telah ditekan.

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 Ouput 1 (Sebelum disubmit)

Jumlah bintang

Gambar 34. Gambar Soal 4 Modul 3

Contoh Output 2 (Sesudah disubmit):

Jumlah bintang 5



Gambar 35. Gambar Soal 4 Modul 3

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

Jumlah bintang 6



Gambar 36. Gambar Soal 4 Modul 3

Simpan dengan nama file: PRAK304.php

A. Source Code

```

1 <html>
2 <head>
3   <title>Soal_4_Modul_3</title>
4 </head>
5 <body>
6   <?php
7     $bintang = NULL;
8     $picture = "<img style='width: 70px;'
src='https://www.freepnglogos.com/uploads/star-png/file-featured-article-star-
svg-wikimedia-commons-8.png' >";
9     if( $_SERVER["REQUEST_METHOD"] == "POST" ) {
10      $bintang = $_POST['bintang'];

```


11	}
12	if(isset(\$_POST['tambah'])) {
13	\$bintang += 1;
14	}
15	if(isset(\$_POST['kurang'])) {
16	\$bintang -= 1;
17	}
18	if(empty(\$bintang)) { ?>
19	<form action="" method="post">
20	<label for="bintang">Jumlah Bintang :</label>
21	<input type="text" name="bintang"> </br>
22	<button type="submit" name="submit">Submit</button>
23	</form>
24	<?php } ?>
25	<?php if(!empty(\$bintang)) {
26	echo "Jumlah Bintang : \$bintang "; ?>
27	
28	<?php for(\$i = 0; \$i < \$bintang; \$i++) {
29	echo "\$picture";
30	} ?>
31	<form action="" method="post">
32	<input type="text" name="bintang" value="<?= \$bintang ?>" hidden>
33	<button type="submit" name="tambah"
34	value="Rambah">Tambah</button>
35	<button type="submit" name="kurang"
36	value="kurang">Kurang</button>
37	</form>
38	<?php } ?>
	</body>
	</html>

Tabel 16. Source Code Soal 4 Modul 3

B. Output Program

Jumlah Bintang :

Gambar 37. Screenshot Hasil Akhir Soal 4 Modul 3



Gambar 38. Screenshot Hasil Akhir Soal 4 Modul 3



Gambar 39. Screenshot Hasil Akhir Soal 4 Modul 3

C. Pembahasan

1. `<?php $bintang = NULL; ?>`: Mendefinisikan variabel \$bintang dan menginisialisasinya dengan nilai NULL.
2. `<?php $picture = ""; ?>`: Mendefinisikan variabel \$picture yang berisi tag gambar bintang dengan lebar 70px.
3. `<?php if($_SERVER["REQUEST_METHOD"] == "POST") { ... } ?>`: Memeriksa apakah form dikirim menggunakan metode POST.
4. `<?php $bintang = $_POST['bintang']; ?>`: Mengambil nilai jumlah bintang yang dikirim melalui form jika form dikirimkan.
5. `<?php if(isset($_POST['tambah'])) { ... } ?>`: Memeriksa apakah tombol "Tambah" ditekan.
6. `<?php $bintang += 1; ?>`: Menambahkan 1 ke nilai \$bintang jika tombol "Tambah" ditekan.
7. `<?php if(isset($_POST['kurang'])) { ... } ?>`: Memeriksa apakah tombol "Kurang" ditekan.
8. `<?php $bintang -= 1; ?>`: Mengurangkan 1 dari nilai \$bintang jika tombol "Kurang" ditekan.
9. `<?php if(empty($bintang)) { ?>`: Memeriksa apakah nilai \$bintang kosong.
10. `<form action="" method="post">`: Tag form untuk mengirimkan data ke halaman itu sendiri menggunakan metode POST.

11. `<label for="bintang">Jumlah Bintang :</label>` dan `<input type="text" name="bintang">`: Label dan input untuk memasukkan jumlah bintang.
12. `<button type="submit" name="submit">Submit</button>`: Tombol submit untuk mengirim form.
13. `<?php } ?>`: Menutup blok if untuk memeriksa apakah nilai \$bintang kosong.
14. `<?php if(!empty($bintang)) { ?>`: Memeriksa apakah nilai \$bintang tidak kosong.
15. `<?php echo "Jumlah Bintang : $bintang "; ?>`: Menampilkan jumlah bintang yang telah dimasukkan oleh pengguna.
16. `

`: Menambahkan beberapa baris kosong untuk memberikan jarak antara jumlah bintang dan gambar bintang.
17. `<?php for($i = 0; $i < $bintang; $i++) { ... } ?>`: Looping untuk menampilkan gambar bintang sebanyak nilai \$bintang.
18. `<?php echo "$picture"; ?>`: Menampilkan gambar bintang sebanyak nilai \$bintang.
19. `<form action="" method="post">`: Tag form untuk mengirimkan data ke halaman itu sendiri menggunakan metode POST.
20. `<input type="text" name="bintang" value="<?=$bintang ?>" hidden>`: Input hidden untuk mengirim nilai \$bintang ke dalam form.
21. `<button type="submit" name="tambah" value="Rambah">Tambah</button>`: Tombol untuk menambah jumlah bintang.
22. `<button type="submit" name="kurang" value="kurang">Kurang</button>`: Tombol untuk mengurangi jumlah bintang.
23. `<?php } ?>`: Menutup blok if untuk memeriksa apakah nilai \$bintang tidak kosong.

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:

BbbbbbbAaaaaaNNnnnnnDdddddUuuuuuNnnnnnGggggg

Gambar 40. Gambar Soal 5 Modul 3

Contoh 2

Input:

BANJAR

Output:

BbbbbbbAaaaaaNNnnnnJjjjjAaaaaRrrrr

Gambar 41. Gambar Soal 5 Modul 3

Simpan dengan nama file: PRAK305.php'

A. Source Code

```

1 <html>
2 <head>
3   <title>Soal_5_Modul_3</title>
4 </head>
5 <body>
6   <form action="" method="post">
7     <input type="text" name="kata">
8     <button type="submit" name="submit">Submit</button>
9   </form>
10  <br>
11  <?php if( isset($_POST['submit']) ) : ?>
12    <?php
13      $kata = $_POST['kata'];
14      $arr = str_split($kata);
15      echo "<h2>Input:</h2>$kata<br><h2>Output:</h2>";
16      for( $i = 1; $i <= count($arr); $i = $i + 1 ) {
17        for( $j = 1; $j <= count($arr); $j = $j + 1 ) {
18          if( $j == 1 ) {

```

19	echo strtoupper(\$arr[\$i-1]);
20	} else {
21	echo strtolower(\$arr[\$i-1]);
22	}
23	}
24	} ?>
25	<?php endif; ?>
26	</body>
27	</html>

Tabel 17. Source Code Soal 5 Modul 3

B. Output Program

Input:

BanDunG

Output:

BbbbbbAaaaaaNnnnnDdddddUuuuuNnnnnGggggg

Gambar 42. Screenshot Hasil Akhir Soal 5 Modul 3

Input:

BANJAR

Output:

BbbbbbAaaaaaNnnnnJjjjjAaaaaRrrrr

Gambar 43. Screenshot Hasil Akhir Soal 5 Modul 3

C. Pembahasan

1. <form action="" method="post">: Tag form untuk mengirimkan data ke halaman itu sendiri menggunakan metode POST.

2. `<input type="text" name="kata">`: Input teks untuk memasukkan kata.
3. `<button type="submit" name="submit">Submit</button>`: Tombol submit untuk mengirimkan form.
4. `<?php if(isset($_POST['submit'])) : ?>`: Memeriksa apakah tombol submit telah ditekan.
5. `<?php $kata = $_POST['kata']; ?>`: Mengambil nilai kata yang dimasukkan oleh pengguna melalui form.
6. `<?php $arr = str_split($kata); ?>`: Membagi kata menjadi array karakter dengan menggunakan fungsi `str_split()`.
7. `<?php echo "<h2>Input:</h2>$kata
<h2>Output:</h2>"; ?>`: Menampilkan label "Input" diikuti dengan kata yang dimasukkan pengguna, kemudian label "Output".
8. `<?php for($i = 1; $i <= count($arr); $i = $i + 1) { ?>`: Memulai loop for untuk iterasi melalui setiap karakter dalam kata.
9. `<?php for($j = 1; $j <= count($arr); $j = $j + 1) { ?>`: Memulai loop for dalam loop pertama untuk mengulangi setiap karakter dalam kata.
10. `<?php if($j == 1) { ?>`: Memeriksa apakah ini adalah iterasi pertama dalam loop dalam kata.
11. `<?php echo strtoupper($arr[$i-1]); ?>`: Jika ya, maka karakter pertama akan ditampilkan dalam huruf kapital menggunakan fungsi `strtoupper()`.
12. `<?php } else { ?>`: Bagian alternatif dari kondisi jika ini bukan iterasi pertama dalam loop dalam kata.
13. `<?php echo strtolower($arr[$i-1]); ?>`: Jika bukan iterasi pertama, maka karakter akan ditampilkan dalam huruf kecil menggunakan fungsi `strtolower()`.
14. `<?php } ?>`: Menutup blok if-else.
15. `<?php } ?>`: Menutup loop kedua dalam loop pertama.
16. `<?php } ?>`: Menutup loop pertama.
17. `<?php endif; ?>`: Menutup blok if untuk memeriksa apakah tombol submit telah ditekan.

Tautan Git

https://github.com/Vainn11/Prak_Web_2/blob/5443d50ef2a555aef951064943f0633696764afc/Modul%203/Prak305.php

MODUL 4

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

Gambar 44. Gambar Soal 1 Modul 4

Contoh Ouput 2 :

Panjang :
Lebar :
Nilai :

Panjang nilai tidak sesuai dengan ukuran matriks

Gambar 45. Gambar Soal 1 Modul 4

Simpan dengan nama file: PRAK401.php

A. Source Code

1	<html>
2	<head>
3	<title>Soal_1_Modul_4</title>
4	<style>
5	table {
6	border-collapse: collapse;
7	}
8	td {
9	border: 1px solid;

10	width: 50px;
11	height: 50px;
12	}
13	</style>
14	</head>
15	<body>
16	<form action="" method="post">
17	Panjang : <input type="text" name="panjang">
18	Lebar : <input type="text" name="lebar">
19	Nilai : <input type="text" name="nilai">
20	<input type="submit" name="cetak" value="Cetak">
21	</form>
22	
23	<?php
24	\$panjang = \$lebar = \$nilai = "";
25	if(isset(\$_POST['cetak'])){
26	if(isset(\$_POST['panjang'])){
27	\$panjang = \$_POST['panjang'];
28	}
29	if(isset(\$_POST['lebar'])){
30	\$lebar = \$_POST['lebar'];
31	}
32	if(isset(\$_POST['nilai'])){
33	\$nilai = (String)\$_POST['nilai'];
34	}
35	\$a = explode(" ", \$nilai);
36	if(\$panjang * \$lebar < count(\$a)){
37	echo "Panjang nilai tidak sesuai dengan ukuran matriks";
38	} else { ?>
39	<table cellpadding="0" cellspacing="0">
40	<?php
41	for(\$i = 0 ; \$i < \$panjang;\$i++){
42	?>
43	<tr>
44	<?php for(\$j=0 ;\$j<\$lebar;\$j++){ ?>
45	<td style="text-align: center;">
46	<?php
47	if(empty(\$a[(\$i*\$panjang)+\$j])){
48	echo 0;


```

49         } else {
50             echo $a[($i*$panjang)+$j];
51         }
52     ?>
53 </td>
54 <?php } ?>
55 </tr>
56 <?php } ?>
57 </table>
58 <?php
59     }
60 }
61 ?>
62 </body>
63 </html>

```

Tabel 18. Source Code Soal 1 Modul 4

B. Output Program

Panjang :

Lebar :

Nilai :

34	56	41
45	36	37
51	32	46

Gambar 46. Screenshot Hasil Akhir Soal 1 Modul 4

Panjang :	<input type="text" value="3"/>
Lebar :	<input type="text" value="2"/>
Nilai :	<input type="text" value="34 56 41 45 36 37 51 32"/>
<input type="button" value="Cetak"/>	

Panjang nilai tidak sesuai dengan ukuran matriks

C. Pembahasan

1. <html>: Tag pembuka untuk menandakan awal dari dokumen HTML.
2. <head>: Bagian kepala dokumen HTML yang berisi metadata dan referensi ke file lain seperti CSS atau JavaScript.
3. <title>Soal_1_Modul_4</title>: Menetapkan judul halaman web.
4. <style>: Mulai tag untuk menulis CSS inline di dalam dokumen HTML.
5. table {...}: Mendefinisikan gaya untuk tabel. border-collapse: collapse; digunakan untuk menghilangkan celah antara sel-sel tabel.
6. td {...}: Mendefinisikan gaya untuk sel-sel tabel. Setiap sel memiliki border dengan ketebalan 1px, dan ukuran width dan height masing-masing 50px.
7. </head>: Penutup dari bagian kepala dokumen HTML.
8. <body>: Bagian tubuh dokumen HTML yang berisi konten yang akan ditampilkan kepada pengguna.
9. <form action="" method="post"> ... </form>: Mendefinisikan sebuah form yang menggunakan metode POST dan akan mengirimkan data ke halaman itu sendiri.
10. <input type="text" name="panjang">: Kotak input untuk memasukkan nilai panjang.
11. <input type="text" name="lebar">: Kotak input untuk memasukkan nilai lebar.
12. <input type="text" name="nilai">: Kotak input untuk memasukkan nilai-nilai matriks.
13. <input type="submit" name="cetak" value="Cetak">: Tombol submit yang akan mengirimkan data ke server.
14. <?php ... ?>: Mulai blok PHP di dalam HTML.
15. Variabel \$panjang, \$lebar, dan \$nilai diinisialisasi dengan string kosong.
16. if(isset(\$_POST['cetak'])) { ... }: Memeriksa apakah tombol "Cetak" telah ditekan.

17. `if(isset($_POST['panjang'])) { ... }, if(isset($_POST['lebar'])) { ... }, dan if(isset($_POST['nilai'])) { ... }`: Mengambil nilai panjang, lebar, dan nilai dari data yang dikirimkan melalui metode POST.
18. `$a = explode(" ", $nilai);`: Memecah nilai-nilai matriks menjadi array menggunakan spasi sebagai delimiter.
19. `if($panjang * $lebar < count($a)) { ... } else { ... }`: Memeriksa apakah jumlah elemen matriks lebih besar dari panjang kali lebar. Jika ya, maka menampilkan pesan kesalahan.
20. `<table cellpadding="0" cellspacing="0"> ... </table>`: Mulai dan akhiri tabel. `cellspacing="0"` dan `cellpadding="0"` digunakan untuk menghilangkan jarak antara sel-sel tabel.
21. `for($i = 0 ; $i < $panjang; $i++){ ... }`: Loop untuk baris tabel.
22. `<tr> ... </tr>`: Mulai dan akhiri baris tabel.
23. `for($j=0 ; $j<$lebar; $j++){ ... }`: Loop untuk kolom tabel.
24. `<td style="text-align: center;"> ... </td>`: Mulai dan akhiri sel tabel dengan teks yang terpusat.
25. `if(empty($a[($i*$panjang)+$j])) { ... } else { ... }`: Memeriksa apakah nilai dalam matriks adalah kosong atau tidak. Jika kosong, menampilkan "0", jika tidak, menampilkan nilai matriks.
26. `</body>`: Penutup dari bagian tubuh dokumen HTML.
27. `</html>`: Penutup dari dokumen HTML.

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

Tabel 19. Table Soal 2 Modul 4

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: $40\% \text{ UTS} + 60\% \text{ UAS}$. 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

Gambar 47. Gambar Soal 2 Modul 4

***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

Tabel 20. Tabel Soal 2 Modul 4

Simpan dengan nama file: PRAK402.php

A. Source Code

1	<html>
2	<head>
3	<title>Soal_2_Modul_4</title>
4	<style>
5	tr { border: black; }
6	table { border-collapse: collapse; }
7	th {
8	padding: 10px;
9	}
10	</style>
11	</head>
12	<body>
13	<?php
14	\$data = [
15	[

16	"nama" => "Andi",
17	"NIM"=>2101001,
18	"uts"=>87 ,
19	"uas"=>65
20],
21	[
22	"nama" => "Budi",
23	"NIM"=>2101002,
24	"uts"=>76 ,
25	"uas"=>79
26],
27	[
28	"nama" => "Tono",
29	"NIM"=>2101003,
30	"uts"=>50 ,
31	"uas"=>41
32],
33	[
34	"nama" => "Jessica",
35	"NIM"=>2101004,
36	"uts"=>60 ,
37	"uas"=>75
38],
39];
40	foreach (\$data as \$key => \$value) {
41	\$a = \$value['uts'];
42	\$b = \$value['uas'];
43	\$data[\$key]['rerata'] = (\$a * 0.4) + (\$b * 0.6);
44	if(\$data[\$key]['rerata']>= 80){
45	\$data[\$key]['huruf'] = 'A';
46	} elseif(\$data[\$key]['rerata']>= 70 and (\$data[\$key]['rerata'] <=
47	79)){
48	\$data[\$key]['huruf'] = 'B';
49	} elseif(\$data[\$key]['rerata']>= 60 and (\$data[\$key]['rerata'] <=
50	69)){
51	\$data[\$key]['huruf'] = 'C';
	} elseif(\$data[\$key]['rerata']>= 50 and (\$data[\$key]['rerata'] <=
	59)){
	\$data[\$key]['huruf'] = 'D';

```

52         } else {
53             $data[$key]['huruf'] = 'E';
54         }
55     }
56     ?>
57     <table border="1",cellspacing="0" cellpadding="5">
58         <tr bgcolor="CCCDCC">
59             <th>Nama</th>
60             <th>NIM</th>
61             <th>Nilai UTS</th>
62             <th>Nilai UAS</th>
63             <th>Nilai Akhir</th>
64             <th>Huruf</th>
65         </tr>
66         <?php foreach($data as $a) { ?>
67             <tr>
68                 <td><?php echo $a["nama"];?></td>
69                 <td><?php echo $a["NIM"];?></td>
70                 <td><?php echo $a["uts"];?></td>
71                 <td><?php echo $a["uas"];?></td>
72                 <td><?php echo $a["rerata"];?></td>
73                 <td><?php echo $a["huruf"];?></td>
74             </tr>
75             <?php } ?>
76         </table>
77     </body>
78 </html>

```

Tabel 21. Source Code Soal 2 Modul 4

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 48. Screenshot Hasil Akhir Soal 2 Modul 4

C. Pembahasan

1. `<html>`: Tag pembuka untuk menandakan awal dari dokumen HTML.
2. `<head>`: Bagian kepala dokumen HTML yang berisi metadata dan referensi ke file lain seperti CSS atau JavaScript.
3. `<title>Soal_2_Modul_4</title>`: Menetapkan judul halaman web.
4. `<style>`: Mulai tag untuk menulis CSS inline di dalam dokumen HTML.
5. `tr { border: black; }`: Mendefinisikan batas seluruh baris tabel dengan warna hitam.
6. `table { border-collapse: collapse; }`: Mengatur agar batas sel tabel menghilang dengan menggabungkan batas sel yang berdekatan.
7. `th { padding: 10px; }`: Memberikan padding 10px untuk sel header tabel.
8. `</head>`: Penutup dari bagian kepala dokumen HTML.
9. `<body>`: Bagian tubuh dokumen HTML yang berisi konten yang akan ditampilkan kepada pengguna.
10. `<?php ... ?>`: Mulai blok PHP di dalam HTML.
11. `$data`: Variabel yang berisi data mahasiswa dalam bentuk array asosiatif.
12. `foreach ($data as $key => $value) { ... }`: Melakukan iterasi melalui array `$data`.
13. Variabel `$a` dan `$b` digunakan untuk menyimpan nilai UTS dan UAS masing-masing.
14. `$data[$key]['rerata']`: Menghitung nilai akhir mahasiswa dengan bobot 40% UTS dan 60% UAS dan menyimpannya dalam array.
15. Kondisional `if...elseif...else` digunakan untuk menentukan huruf nilai berdasarkan nilai akhir mahasiswa.
16. `<table border="1",cellspacing="0" cellpadding="5">`: Mulai tag tabel dengan atribut `border`, `cellspacing`, dan `cellpadding` yang ditentukan.
17. `<tr bgcolor="CCCDCC">`: Mulai baris tabel dengan latar belakang warna abu-abu.
18. `<th>...</th>`: Header kolom tabel.
19. `<?php foreach($data as $a) { ?> ... <?php } ?>`: Melakukan iterasi melalui array `$data` untuk membuat baris-baris data tabel.
20. `<td>...</td>`: Sel-sel data tabel.
21. `</body>`: Penutup dari bagian tubuh dokumen HTML.
22. `</html>`: Penutup dari dokumen HTML.

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

Tabel 22. Tabel Soal 3 Modul 4

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		

Tabel 23. Tabel Soal 3 Modul 4

Simpan dengan nama file: PRAK403.php

A. Source Code

1	<html>
2	<head>
3	<title>Soal_3_Modul_4</title>
4	<style>
5	tr{ border: black; }
6	table{ border-collapse: collapse; }
7	</style>
8	</head>
9	<body>
10	<?php
11	\$data = [
12	

13	<pre> ["no"=> 1,"nama" => "Ridho","Matkul" => ["Pemrograman 1","Praktikum Pemrograman 1","pengantar Lingkungan Lahan Basah","Arsitektur Komputer"],"sks"=>[2,1,2,3]], ["no"=>2,"nama" => "Ratna","Matkul" => ["Basis Data 1","Praktikum Basis Data 1","Kalkulus"],"sks"=>[2,1,3]], ["no"=>3,"nama" => "Tono","Matkul" => ["Rekayasa Perangkat Lunak","Analisa dan Perancangan Sistem","Komputasi Awan","Kecerdasan Bisnis"],"sks"=>[3,3,3,3]],]; foreach (\$data as \$key => \$value) { \$data[\$key]['total'] = array_sum(\$value['sks']); if(\$data[\$key]['total'] < 7){ \$data[\$key]['ket'] = "Revisi KRS"; } else{ \$data[\$key]['ket'] = "Tidak Revisi"; } } ?> <table border="1",cellspacing="0" cellpadding ="5"> <tr bgcolor = "CCCDCC"> <th>No</th> <th>Nama</th> <th>Mata Kuliah</th> <th>SKS</th> <th>Total SKS</th> <th>Keterangan</th> </tr> <?php foreach(\$data as \$a){ ?><tr> <td><?php echo \$a["no"]; ?></td> <td><?php echo \$a["nama"]; ?></td><td><?php echo \$a["Matkul"][0];?> </td><td><?php </pre>
----	---

47	echo \$a["sks"][0];?>
48	</td><td><?php
49	echo \$a["total"];
50	?></td><?php
51	if(\$a['total'] < 7){
52	?> <td bgcolor = "red"><?php
53	echo \$a["ket"];
54	}else{
55	?> <td bgcolor = "green"><?php
56	echo \$a["ket"];
57	}?></td></tr>
58	<tr>
59	<td></td>
60	<td></td>
61	<td><?php
62	echo \$a["Matkul"][1];?>
63	</td><td><?php
64	echo \$a["sks"][1];?>
65	</td><td></td><td></td></tr>
66	<tr>
67	<td></td>
68	<td></td>
69	<td><?php
70	echo \$a["Matkul"][2];?>
71	</td><td><?php
72	echo \$a["sks"][2];?>
73	</td><td></td><td></td></tr>
74	<tr>
75	<?php
76	if(!empty(\$a['Matkul'][3])){ ?>
77	<td></td><td></td><td><?php
78	echo \$a["Matkul"][3];?>
79	</td><td><?php
80	}
81	if(!empty(\$a['sks'][3])){ ?>
82	<?php
83	echo \$a["sks"][3];?>
84	</td><td></td></tr><?php
85	}?>

86	<?php
87	} ?>
88	</table>
89	</body>
	</html>

Tabel 24. Source Code Soal 3 Modul 4

B. Output Program

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

Gambar 49. Screenshot Hasil Akhir Soal 3 Modul 4

C. Pembahasan

1. <html>: Tag pembuka untuk menandakan awal dari dokumen HTML.
2. <head>: Bagian kepala dokumen HTML yang berisi metadata dan referensi ke file lain seperti CSS atau JavaScript.
3. <title>Soal_3_Modul_4</title>: Menetapkan judul halaman web.
4. <style>: Mulai tag untuk menulis CSS inline di dalam dokumen HTML.
5. tr { border: black; }; Mendefinisikan batas seluruh baris tabel dengan warna hitam.
6. table { border-collapse: collapse; }; Mengatur agar batas sel tabel menghilang dengan menggabungkan batas sel yang berdekatan.
7. </head>: Penutup dari bagian kepala dokumen HTML.
8. <body>: Bagian tubuh dokumen HTML yang berisi konten yang akan ditampilkan kepada pengguna.
9. <?php ... ?>: Mulai blok PHP di dalam HTML.
10. \$data: Variabel yang berisi data mahasiswa dalam bentuk array multidimensi.

11. `foreach ($data as $key => $value) { ... }`: Melakukan iterasi melalui array `$data`.
12. `$data[$key]['total']`: Menghitung total SKS yang diambil oleh mahasiswa dan menyimpannya dalam array.
13. Kondisional `if...else` digunakan untuk menentukan keterangan apakah mahasiswa tersebut perlu melakukan revisi KRS atau tidak berdasarkan total SKS yang diambil.
14. `<table border="1",cellspacing="0" cellpadding="5">`: Mulai tag tabel dengan atribut `border`, `cellspacing`, dan `cellpadding` yang ditentukan.
15. `<tr bgcolor="CCCDCC">`: Mulai baris tabel dengan latar belakang warna abu-abu.
16. `<th>...</th>`: Header kolom tabel.
17. `<?php foreach($data as $a){ ... } ?>`: Melakukan iterasi melalui array `$data` untuk membuat baris-baris data tabel.
18. `<tr>`: Mulai baris baru dalam tabel.
19. `<td>...</td>`: Sel-sel data tabel.
20. `echo $a["no"]`, `echo $a["nama"]`: Menampilkan nomor dan nama mahasiswa.
21. `echo $a["Matkul"][0]`, `echo $a["sks"][0]`: Menampilkan nama dan SKS mata kuliah pertama.
22. `echo $a["total"]`: Menampilkan total SKS yang diambil.
23. Kondisional `if...else` digunakan untuk menentukan latar belakang sel berdasarkan keterangan revisi KRS atau tidak.
24. Untuk setiap mahasiswa, baris baru dibuat untuk setiap mata kuliah yang diambil. Setiap mata kuliah dan SKS ditampilkan dalam sel-sel yang sesuai.
25. `</body>`: Penutup dari bagian tubuh dokumen HTML.
26. `</html>`: Penutup dari dokumen HTML.

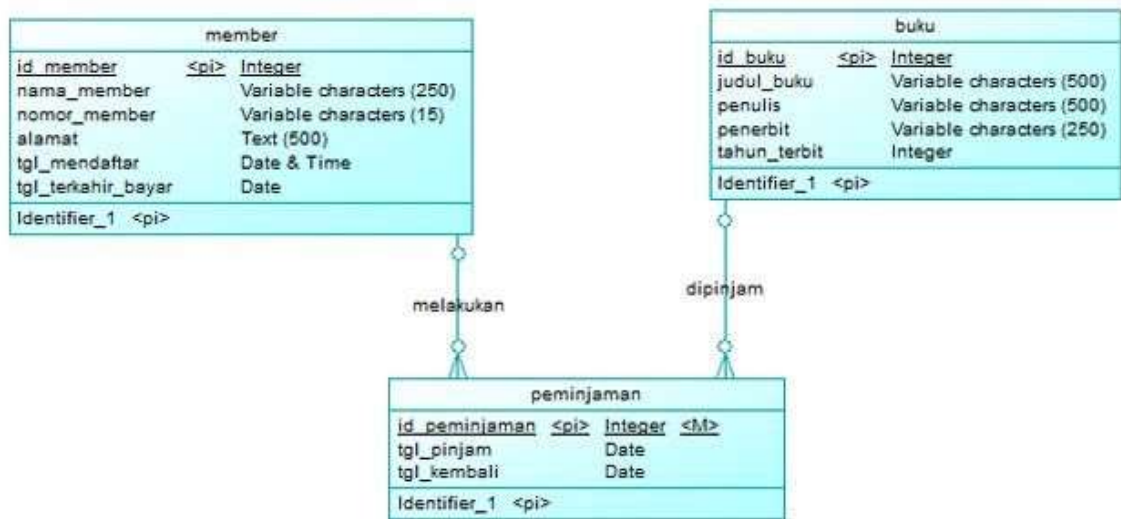
Tautan Git

https://github.com/Vainn11/Prak_Web_2/blob/652b93afd6df7b7958a6c369f8b1354b3349f778/Modul%204/Prak403.php

MODUL 5

Soal 1

Diberikan desain konseptual sebuah basis data seperti berikut



Gambar 50. Gambar Soal 1 Modul 5

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 ormular 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	function koneksi()
3	{
4	try{
5	\$pdo_conn = new PDO('mysql:host=localhost;dbname=perpus','root','',
6	array(PDO::ATTR_ERRMODE,PDO::ERRMODE_EXCEPTION,
	PDO::ATTR_PERSISTENT=>TRUE)

7);
8	}catch(PDOException \$e){
9	print "Koneksi atau query bermasalah: ".\$e->getMessage() . " ";
10	die();
11	}
12	return \$pdo_conn;
13	}?>

Tabel 25. Source Code Koneksi.php Soal 1 Modul 5

1	<!DOCTYPE html>
2	<html>
3	<head>
4	<title>Kale</title>
5	<link rel="stylesheet" type="text/css"
6	href="http://code.ionicframework.com/ionicons/2.0.1/css/ionicons.min.css"
	/>
	<link
	href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css"
	rel="stylesheet" integrity="sha384-
	EVSTQN3/azprG1Anm3QDgpJLIm9Nao0Yz1ztcQTwFspd3yD65Vohhpuu
7	COmLASjC" crossorigin="anonymous">
8	<style type="text/css">
9	* {
10	box-sizing: border-box;
11	}
12	body {
13	background-image: url();
14	background-size: cover;
15	font-family: "forte";
16	margin: 0;
17	padding: 0;
18	}
19	header {
20	text-align: center;
21	}
22	.link_none {
23	color: white;
24	}
25	button {

26	padding: 12.5px 30px;
27	border: 0;
28	border-radius: 100px;
29	background-color: #5F9EA0;
30	color: #ffffff;
31	font-weight: Bold;
32	transition: all 0.5s;
33	-webkit-transition: all 0.5s;
34	}
35	button:hover {
36	background-color: grey;
37	box-shadow: 0 0 20px white;
38	transform: scale(1.1);
39	}
40	button:active {
41	background-color: black;
42	transition: all 0.25s;
43	-webkit-transition: all 0.25s;
44	box-shadow: none;
45	transform: scale(0.98);
46	}
47	ul {
48	list-style-type: none;
49	overflow: hidden;
50	background-color: #5F9EA0;
51	background-repeat: no-repeat;
52	padding-left: 18px !important;
53	}
54	li a {
55	display: block;
56	color: white;
57	text-align: center;
58	padding: 20px 20px;
59	text-decoration: none;
60	}
61	li {
62	float: right;
63	}
64	li a:hover {

65	background-color: #272727;
66	}
67	.active {
68	background-color: #272727;
69	}
70	section:after {
71	content: "";
72	display: table;
73	clear: both;
74	}
75	nav {
76	float: left;
77	width: 20%;
78	padding: 10px;
79	}
80	article {
81	height: 650px;
82	width: 100%;
83	padding: 10px;
84	}
85	footer {
86	text-align: center;
87	color: white;
88	padding: 10px;
89	padding-left: 0px;
90	}
91	</style>
92	</head>
93	<body>
94	
95	<header>
96	<div>
97	
98	<a href="peminjaman.php"
99	class="link_none">Peminjaman
100	<a href="member.php"
101	class="link_none">Member
	<a href="buku.php"
	class="link_none">Buku

102	
103	
104	</div>
105	</header>
106	<section>
107	<article>
108	<h2 style="text-align:center;">Buku</h2>
109	<table border="1" class="table table-striped">
110	<thead>
111	<tr>
112	<th><p>ID Buku</p></th>
113	<th><p>Judul Buku</p></th>
114	<th><p>Penulis</p></th>
115	<th><p>Penerbit</p></th>
116	<th><p>Tahun Terbit</p></th>
117	<th><p>Aksi</p></th>
118	</tr>
119	</thead>
120	<tbody>
121	<?php
122	require('model.php');
123	if (isset(\$_GET['id_buku'])){
124	hapusbuku(\$_GET['id_buku']);
125	}
126	tampilbuku();
127	?>
128	</tbody>
129	</table>
130	 </br>
131	
132	<center>
133	<button>Tambah Buku</button> </br>
134	</article>
135	</center>
136	</section>
	</body>
	</html>

Tabel 26. Source Code Buku.php Soal 1 Modul 5

1	<?php require('model.php');
2	if (isset(\$_GET['id_buku'])) {
3	editbuku();
4	}
5	?>
6	<!DOCTYPE html>
7	<html>
8	<head>
9	<?php echo (isset(\$_GET['id_buku'])) ? "<title>Edit Buku</title>":
	"<title>Tambah Buku</title>" ?>
10	<style>
11	body {
12	font-family: Arial, sans-serif;
13	margin: 0;
14	padding: 20px;
15	}
16	
17	h2 {
18	margin-bottom: 20px;
19	}
20	
21	form {
22	width: 400px;
23	}
24	
25	table {
26	width: 100%;
27	}
28	
29	tr {
30	margin-bottom: 10px;
31	}
32	
33	td {
34	padding: 5px;
35	}
36	
37	input[type="text"] {
38	width: 100%;

```

39         padding: 5px;
40     }
41
42     button {
43         padding: 12.5px 30px;
44         border: 0;
45         border-radius: 100px;
46         background-color: #5F9EA0;
47         color: #ffffff;
48         font-weight: Bold;
49         transition: all 0.5s;
50         -webkit-transition: all 0.5s;
51     }
52
53     button:hover {
54         background-color: grey;
55         box-shadow: 0 0 20px white;
56         transform: scale(1.1);
57     }
58
59     button:active {
60         background-color: black;
61         transition: all 0.25s;
62         -webkit-transition: all 0.25s;
63         box-shadow: none;
64         transform: scale(0.98);
65     }
66 </style>
67 </head>
68 <body>
69     <center>
70     <h2>Form Buku</h2>
71     <form action="" method="post">
72         <table>
73             <tr>
74                 <td>ID Buku</td>
75                 <td><input type="text" name="id_buku" <?php echo
(isset($_GET['id_buku'])) ? "value = " . $result[0]['id_buku'] . "" : "value =
76 " "; ?> required> <br></td>

```

77	</tr>
78	<tr>
79	<td>Judul Buku</td> <td><input type="text" name="judul" <?php echo
80	(isset(\$_GET['id_buku'])) ? "value = " . \$result[0]['judul_buku'] . "" : "value
81	= " "; ?> required> </td>
82	</tr>
83	<tr>
	<td>Nama Penulis</td>
84	<td><input type="text" name="penulis" <?php echo
85	(isset(\$_GET['id_buku'])) ? "value = " . \$result[0]['penulis'] . "" : "value = "
86	"; ?> required> </td>
87	</tr>
	<tr>
88	<td>Penerbit</td>
89	<td><input type="text" name="penerbit" <?php echo
90	(isset(\$_GET['id_buku'])) ? "value = " . \$result[0]['penerbit'] . "" : "value = "
91	" "; ?> required> </td>
	</tr>
92	<tr>
93	<td>Tahun Terbit</td>
94	<td><input type="text" name="thnterbit" <?php echo
95	(isset(\$_GET['id_buku'])) ? "value = " . \$result[0]['tahun_terbit'] . "" :
96	"value = " "; ?> required> </td>
97	</tr>
98	<tr>
99	<td></td>
100	<td>
101	<?php
102	if (isset(\$_GET['id_buku'])) {
103	echo "<button type=\"submit\"
104	name=\"update\">Edit</button>";
105	}else {
106	echo "<button type=\"submit\"
107	name=\"submit\">Tambah</button>";
108	}
109	?>
110	</td>
111	</tr>

	<pre> </table> 112 </form> 113 </center> 114 <?php 115 if (isset(\$_POST['submit'])) { 116 tambahbuku(\$_POST['id_buku'],\$_POST['judul'], \$_POST['penulis'], 117 \$_POST['penerbit'], \$_POST['thnterbit']); 118 } if (isset(\$_POST['update'])) { updatebuku(\$_GET['id_buku'],\$_POST['judul'],\$_POST['penulis'],\$_PO ST['penerbit'],\$_POST['thnterbit']); } ?> </body> </html> </pre>
--	---

Tabel 27. Source Code Formbuku.php Soal 1 Modul 5

1	<?php require('model.php');
2	if (isset(\$_GET['id_member'])) {
3	editmember();
4	}
5	?>
6	<!DOCTYPE html>
7	<html>
8	<head>
9	<?php echo (isset(\$_GET['id_member'])) ? "<title>Edit Member</title>":
	"<title>Tambah Member</title>" ?>
10	<style>
11	body {
12	font-family: Arial, sans-serif;
13	margin: 0;
14	padding: 20px;
15	}
16	
17	h2 {
18	margin-bottom: 20px;
19	}
20	

21	form {
22	width: 400px;
23	}
24	
25	table {
26	width: 100%;
27	}
28	
29	tr {
30	margin-bottom: 10px;
31	}
32	
33	td {
34	padding: 5px;
35	}
36	
37	input[type="text"] {
38	width: 100%;
39	padding: 5px;
40	}
41	
42	button {
43	padding: 12.5px 30px;
44	border: 0;
45	border-radius: 100px;
46	background-color: #5F9EA0;
47	color: #ffffff;
48	font-weight: Bold;
49	transition: all 0.5s;
50	-webkit-transition: all 0.5s;
51	}
52	
53	button:hover {
54	background-color: grey;
55	box-shadow: 0 0 20px white;
56	transform: scale(1.1);
57	}
58	
59	button:active {

60	background-color: black ;
61	transition: all 0.25s;
62	-webkit-transition: all 0.25s;
63	box-shadow: none;
64	transform: scale(0.98);
65	}
66	</style>
67	</head>
68	<body>
69	<center>
70	<h2>Form Member</h2>
71	<form action="" method="post">
72	<table>
73	<tr>
74	<td>ID Member</td>
75	<td><input type="text" name="id_member" <?php echo (isset(\$_GET['id_member'])) ? "value = " . \$result[0]["id_member"] . "" : "value = " ; ?> required> </td>
76	</tr>
77	<tr>
78	<td>Nama Member</td>
79	<td><input type="text" name="Nama" <?php echo (isset(\$_GET['id_member'])) ? "value = " . \$result[0]["nama_member"] . "" : "value = " ; ?> required> </td>
80	</tr>
81	<tr>
82	<td>Nomor Member</td>
83	<td><input type="text" name="nomor" <?php echo (isset(\$_GET['id_member'])) ? "value = " . \$result[0]["nomor_member"] . "" : "value = " ; ?> required> </td>
84	</tr>
85	<tr>
86	<td>Alamat</td>
87	<td><input type="text" name="alamat" <?php echo (isset(\$_GET['id_member'])) ? "value = " . \$result[0]["alamat"] . "" : "value = " ; ?> required> </td>
88	</tr>
89	<tr>
90	<td>Tanggal Daftar</td>

```

91         <td><input type="datetime-local" name="daftar" <?php echo
(isset($_GET['id_member'])) ? "value = " . $result[0]["tgl_mendaftar"] . "" :
"value = " . ""; ?> required> <br></td>
92     </tr>
93     <tr>
94         <td>Tanggal bayar Terakhir</td>
95         <td><input type="date" name="bayar" <?php echo
(isset($_GET['id_member'])) ? "value = " . $result[0]["tgl_terakhir_bayar"] .
"" : "value = " . ""; ?> required> <br></td>
96     </tr>
97     <tr>
98         <td></td>
99         <td>
100             <?php
101                 if (isset($_GET['id_member'])) {
102                     echo "<button type=\"submit\"
103 name=\"update\">Edit</button>";
104                 }else {
105                     echo "<button type=\"submit\"
106 name=\"submit\">Tambah</button>";
107                 }
108             ?>
109         </td>
110     </tr>
111 </table>
112 </form>
113 </center>
114
115 <?php
if (isset($_POST['submit'])) {
116     tambahmember($_POST['id_member'],$_POST['Nama'],
117 $_POST['nomor'], $_POST['alamat'], $_POST['daftar'],$_POST['bayar']);
118 }
if (isset($_POST['update'])) {
119     updatemember($_POST['id_member'],$_POST['Nama'],
120 $_POST['nomor'], $_POST['alamat'], $_POST['daftar'],$_POST['bayar']);
121 }
122 ?>
</body>

```

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

Tabel 28. Source Code Formmember.php Soal 1 Modul 5

1	<?php
2	require('model.php');
3	if (isset(\$_GET['id_peminjaman'])) {
4	editpeminjaman();
5	}
6	?>
7	<!DOCTYPE html>
8	<html>
9	<head>
10	<?php echo (isset(\$_GET['id_peminjaman'])) ? "<title>Edit Data Peminjaman</title>": "<title>Tambah Data Peminjaman</title>" ?>
11	<style>
12	body {
13	font-family: Arial, sans-serif;
14	margin: 0;
15	padding: 20px;
16	}
17	
18	h2 {
19	margin-bottom: 20px;
20	}
21	
22	form {
23	width: 400px;
24	}
25	
26	table {
27	width: 100%;
28	}
29	
30	tr {
31	margin-bottom: 10px;
32	}
33	
34	td {
35	padding: 5px;

36	}
37	
38	input[type="text"] {
39	width: 100%;
40	padding: 5px;
41	}
42	
43	button {
44	padding: 12.5px 30px;
45	border: 0;
46	border-radius: 100px;
47	background-color: #5F9EA0;
48	color: #ffffff;
49	font-weight: Bold;
50	transition: all 0.5s;
51	-webkit-transition: all 0.5s;
52	}
53	
54	button:hover {
55	background-color: grey;
56	box-shadow: 0 0 20px white;
57	transform: scale(1.1);
58	}
59	
60	button:active {
61	background-color: black;
62	transition: all 0.25s;
63	-webkit-transition: all 0.25s;
64	box-shadow: none;
65	transform: scale(0.98);
66	}
67	</style>
68	</head>
69	<body>
70	<center>
71	<h2>Form Peminjaman</h2>
72	<form action="" method="post">
73	<table>
74	<tr>

75	<td>ID Peminjaman</td>
76	<td><input type="text" name="id_Peminjaman" <?php echo (isset(\$_GET['id_peminjaman'])) ? "value = " . \$result[0]['id_peminjaman'] . "" : "value = " . \$_GET['id_peminjaman']; ?> required> </td>
77	</tr>
78	<td>Tanggal Peminjaman</td>
79	<td><input type="date" name="pinjam" <?php echo (isset(\$_GET['id_peminjaman'])) ? "value = " . \$result[0]['tgl_peminjaman'] . "" : "value = " . \$_GET['id_peminjaman']; ?> required> </td>
80	</tr>
81	<tr>
82	<td>Tanggal Kembali</td>
83	<td><input type="date" name="kembali" <?php echo (isset(\$_GET['id_peminjaman'])) ? "value = " . \$result[0]['tgl_kembali'] . "" : "value = " . \$_GET['id_peminjaman']; ?> required> </td>
84	</tr>
85	<tr>
86	<td></td>
87	<td>
88	<?php
89	if (isset(\$_GET['id_peminjaman'])) {
90	echo "<button type='submit' name='update'>Edit</button>";
91	}else {
92	echo "<button type='submit' name='submit'>Tambah</button>";
93	}
94	?>
95	</td>
96	</tr>
97	</table>
98	</form>
99	</center>
100	<?php
101	if (isset(\$_POST['submit'])) {
102	tambahpeminjaman(\$_POST['id_Peminjaman'],\$_POST['pinjam'],\$_POST['kembali']);
103	}
104	if (isset(\$_POST['update'])) {
105	updatepeminjaman(\$_POST['id_Peminjaman'],\$_POST['pinjam'],\$_POST['kembali']);

106	}
107	?>
108	</body>
109	</html>

Tabel 29. Source Code Formpeminjaman.php Soal 1 Modul 5

1	<!DOCTYPE html>
2	<html>
3	<head>
4	<title>Kale</title>
5	<link rel="stylesheet" type="text/css"
6	href="http://code.ionicframework.com/ionicons/2.0.1/css/ionicons.min.css"
7	/>
8	<style type="text/css">
9	* {
10	box-sizing: border-box;
11	}
12	body {
13	background-image: url("asd.png");
14	background-size: 100%;
15	font-family: "forte";
	margin: 0;

16	padding: 0;
17	}
18	header {
19	text-align: center;
20	background-color: #737373;
21	}
22	.link_none {
23	color: white;
24	font-size: 18px;
25	}
26	ul {
27	list-style-type: none;
28	margin: 0;
29	padding: 0;
30	overflow: hidden;
31	background-color: #5F9EA0;
32	background-repeat: no-repeat;
33	padding-left: 16px !important;
34	background-size: 4%;
35	background-position: 4%;
36	}
37	li {
38	display: inline-block;
39	margin-right: 10px;
40	}
41	li a {
42	display: block;
43	color: white;
44	text-align: center;
45	padding: 20px 20px;
46	text-decoration: none;
47	}
48	li a:hover {
49	background-color: #272727;
50	}
51	.active {
52	background-color: #272727;
53	}
54	section:after {

55	content: "";
56	display: table;
57	clear: both;
58	}
59	nav {
60	float: left;
61	width: 20%;
62	padding: 10px;
63	}
64	article {
65	height: 650px;
66	width: 100%;
67	padding: 10px;
68	}
69	footer {
70	text-align: center;
71	color: white;
72	padding: 10px;
73	padding-left: 0px;
74	}
75	</style>
76	</head>
77	<body>
78	<header>
79	<div class="active">
80	
81	<a href="buku.php"
82	class="link_none">Buku
83	<a href="member.php"
84	class="link_none">Member
85	<a href="peminjaman.php"
86	class="link_none">Peminjaman
87	
	</div>
	</header>
	</body>
	</html>

Tabel 30. Source Code Index.php Soal 1 Modul 5

1	<!DOCTYPE html>
2	<html>
3	<head>
4	<title>Kale</title>
5	<link rel="stylesheet" type="text/css"
6	href="http://code.ionicframework.com/ionicons/2.0.1/css/ionicons.min.css" />
7	<link
8	href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css"
9	rel="stylesheet" integrity="sha384-
10	EVSTQN3/azprG1Anm3QDgpJLIm9Nao0Yz1ztcQTwFspd3yD65VohhpuuCOmLASjC"
11	crossorigin="anonymous">
12	<style type="text/css">
13	* {
14	box-sizing: border-box;
15	}
16	body {
17	background-image: url();
18	background-size: cover;
19	font-family: "forte";
20	margin: 0;
21	padding: 0;
22	}
23	header {
24	text-align: left;
25	color: yellow;
26	background-color: #5F9EA0;
27	}
28	section:after {
29	content: "";
30	display: table;
31	clear: both;
32	}
33	nav {
34	float: left;
35	width: 20%;
36	padding: 10px;
37	}
38	article {
39	height: 650px;

35	float: left;
36	width: 100%;
37	padding: 10px;
38	}
39	footer {
40	text-align: center;
41	color: white;
42	padding: 10px;
43	padding-left: 0px;
44	background: tr;
45	}
46	.link_none {
47	color: white;
48	}
49	button {
50	padding: 12.5px 30px;
51	border: 0;
52	border-radius: 100px;
53	background-color: #5F9EA0;
54	color: #ffffff;
55	font-weight: bold;
56	transition: all 0.5s;
57	-webkit-transition: all 0.5s;
58	}
59	button:hover {
60	background-color: grey;
61	box-shadow: 0 0 20px white;
62	transform: scale(1.1);
63	}
64	button:active {
65	background-color: black;
66	transition: all 0.25s;
67	-webkit-transition: all 0.25s;
68	box-shadow: none;
69	transform: scale(0.98);
70	}
71	ul {
72	list-style-type: none;
73	overflow: hidden;

74	background-color: #5F9EA0;
75	background-repeat: no-repeat;
76	padding-left: 18px !important;
77	}
78	li a {
79	display: block;
80	color: white;
81	text-align: center;
82	padding: 20px 20px;
83	text-decoration: none;
84	}
85	li {
86	float: right;
87	}
88	li a:hover {
89	background-color: #272727;
90	}
91	.active {
92	background-color: #272727;
93	}
94	</style>
95	</head>
96	<body>
97	<header>
98	<div class="active">
99	
100	
101	<a href="peminjaman.php"
102	class="link_none">Peminjaman
103	<a href="member.php"
104	class="link_none">Member
105	Buku
106	
107	
108	</div>
109	</header>
110	<section>
111	<article>
112	<h2 style="text-align:center;">Member</h2>

111	<center>
112	<table border="1" class="table table-striped">
113	<thead>
114	<tr>
115	<th><p>ID Member</p></th>
116	<th><p>Nama</p></th>
117	<th><p>Nomor</p></th>
118	<th><p>Alamat</p></th>
119	<th><p>Tanggal Daftar</p></th>
120	<th><p>Tanggal Terakhir Bayar</p></th>
121	<th><p>Aksi</p></th>
122	</tr>
123	</thead>
124	<tbody>
125	<?php
126	require('model.php');
127	if (isset(\$_GET['id_member'])) {
128	hapusmember(\$_GET['id_member']);
129	}
130	tampilmember();
131	?>
132	</tbody>
133	</table>
134	</center>
135	
136	<center><button>Tambah
137	Member</button></center>
138	
139	</article>
140	</section>
141	</body>
141	</html>

Tabel 31. Source Code Member.php Soal 1 Modul 5

1	<!DOCTYPE html>
2	<html lang="en">
3	<head>
4	<meta charset="UTF-8">
5	<meta http-equiv="X-UA-Compatible" content="IE=edge">

6	<meta name="viewport" content="width=device-width, initial-scale=1.0">
7	<link
	href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css
	" rel="stylesheet" integrity="sha384-
	EVSTQN3/azprG1Anm3QDgpJLIm9Nao0Yz1ztcQTwFspd3yD65Vohhpuu
8	COmLASjC" crossorigin="anonymous">
9	<title>Kale</title>
10	<style>
11	body {
12	text-align: center;
13	}
14	a:link {
15	color: white;
16	text-decoration: none;
17	}
18	
19	a:active {
20	color: white;
21	text-decoration: none;
22	}
23	
24	a:visited {
25	color: white;
26	text-decoration: none;
27	}
28	p {
29	text-align: center;
30	}
31	</style>
32	</head>
33	<body>
34	<?php
35	require("koneksi.php");
36	
37	function tampilbuku()
38	{
39	\$stmt = koneksi()->prepare("SELECT * FROM buku");
40	\$stmt->execute();
41	\$result = \$stmt -> fetchAll();

```

42     if(!empty($result)){
43         foreach ($result as $buku) {
44             echo "<tr>";
45             echo "<td><p>". $buku['id_buku']. "</p></td>";
46             echo "<td><p>". $buku['judul_buku']. "</p></td>";
47             echo "<td><p>". $buku['penulis']. "</p></td>";
48             echo "<td><p>". $buku['penerbit']. "</p></td>";
49             echo "<td><p>". $buku['tahun_terbit']. "</p></td>";
50             echo "<td>". "<button class=link_none> <a href =
'formbuku.php?id_buku=".$buku['id_buku']."'>Edit</a> </button>";
51             echo "<button class=link_none><a href='buku.php?id_buku=" .
52 $buku['id_buku'] . "' onclick=\"return confirm('Anda yakin ingin
menghapusnya?')\">Hapus</a> </button></td>";
53             echo "</tr>";
54         }
55     }
56 }
57
58 function tambahBuku($id,$judul,$penulis,$penerbit,$tahun)
59 {
60     $sql = "INSERT INTO
buku(id_buku,judul_buku,penulis,penerbit,tahun_terbit)values(:id_b,:judul_b
61 ,:penulis,:penerbit,:tahun)";
62     $stmt = koneksi()->prepare($sql);
63     $result = $stmt->execute(array(':id_b' => $id,
64 'judul_b'=>$judul,':penulis'=>$penulis,':penerbit'=>$penerbit,':tahun'=>$tahu
65 n));
66     if (!empty($result)) {
67         header('location:buku.php');
68     }
69 }
70 function editbuku(){
71     $stmt = koneksi()->prepare("SELECT * FROM buku WHERE id_buku
72 =".$_GET['id_buku']);
73     $stmt->execute();
74     $GLOBALS['result'] = $stmt->fetchAll();
75 }
76 function updatebuku($id,$judul,$penulis,$penerbit,$tahun)
77 {

```

	\$pdo_statement = koneksi()->prepare(
74	"update buku set judul_buku=" . \$judul . " , penulis=" . \$penulis . " ,
75	penerbit=" . \$penerbit . " , tahun_terbit=" . \$tahun . " where id_buku=" . \$id
76);
77	\$result = \$pdo_statement->execute();
78	if(\$result){
79	header('location:buku.php');
80	}
81	}
82	function hapusbuku(\$id_buku){
83	\$stmt = koneksi()->prepare("DELETE FROM buku where id_buku=" .
84	\$id_buku);
85	\$result = \$stmt->execute();
86	if (\$result) {
87	header('location:buku.php');
88	}
89	}
90	function tampilmember()
91	{
92	\$stmt = koneksi()->prepare("SELECT * FROM member");
93	\$stmt->execute();
94	\$result = \$stmt -> fetchAll();
95	if(!empty(\$result)){
96	foreach (\$result as \$member) {
97	echo "<tr>";
98	echo "<td><p>". \$member['id_member']. "</p></td>";
99	echo "<td><p>". \$member['nama_member']. "</p></td>";
100	echo "<td><p>". \$member['nomor_member']. "</p></td>";
101	echo "<td><p>". \$member['alamat']. "</p></td>";
	echo "<td><p>". \$member['tgl_mendaftar']. "</p></td>";
102	echo "<td><p>". \$member['tgl_terakhir_bayar']. "</p></td>";
	echo "<td><p>". "<button class=link_none><a href =
	'formmember.php?id_member=" . \$member['id_member'] . "'>Edit
103	</button>";
104	echo "<button class=link_none><a href='member.php?id_member="
105	. \$member['id_member'] . " onclick=\"return confirm('lanjut delete
106	?\")\">Hapus </button></td>";
107	echo "</tr>";
108	}

```

109     }
110 }

function tambahmember($id,$nama,$nomor,$alamat,$tgldaftar,$tglbayar)
111 {
    $sql = "INSERT INTO
member(id_member,nama_member,nomor_member,alamat,tgl_mendaftar,tg
112 l_terakhir_bayar)values(:id_m,:nama_m,:no_m,:alamat,:tgl_daf,:tgl_bay)";
    $stmt = koneksi()->prepare($sql);
    $result = $stmt->execute(array(':id_m' => $id,
113 'nama_m'=>$nama,'no_m'=>$nomor,'alamat'=>$alamat,'tgl_daf'=>$tgldaft
114 ar,'tgl_bay'=>$tglbayar));
    if (!empty($result)) {
115         header('location:member.php');
116     }
117 }
118 }

function editmember(){
119     $stmt = koneksi()->prepare("SELECT * FROM member WHERE
120 id_member = ".$_GET['id_member']);
121     $stmt->execute();
122     $GLOBALS['result'] = $stmt->fetchAll();
123 }
124 function updatemember($id,$nama,$no,$alamat,$tgldaftar,$tglbayar)
125 {
    $pdo_statement = koneksi()->prepare(
        "update member set nama_member=" . $nama . " , nomor_member =" .
126 $no . " , alamat=" . $alamat . " , tgl_mendaftar=" . $tgldaftar . " ,
127 tgl_terakhir_bayar=" . $tglbayar . " where id_member=" . $id
128 );
    $result = $pdo_statement->execute();
129     if($result){
130         header('location:member.php');
131     }
132 }
133 }

function hapusmember($id_member){
134     $stmt = koneksi()->prepare("DELETE FROM member WHERE
135 id_member=" . $id_member);
136     $result= $stmt->execute();
137     if($result){

```



```

138     header('location:member.php');
139 }
140 }
141 function tampilpeminjaman()
142 {
143     $stmt = koneksi()->prepare("SELECT * FROM peminjaman");
144     $stmt->execute();
145     $result = $stmt -> fetchAll();
146     if(!empty($result)){
147         foreach ($result as $peminjaman) {
148             echo "<tr>";
149             echo "<td><p>". $peminjaman['id_peminjaman']."</p></td>";
150             echo "<td><p>". $peminjaman['tgl_peminjaman']."</p></td>";
151             echo "<td><p>". $peminjaman['tgl_kembali']."</p></td>";
152             echo "<td><p>". "<button class=link_none><a href =
'form6?id_peminjaman=".$peminjaman['id_peminjaman']."'>Edit</a></butto
n>";
153             echo "<button class=link_none><a
154 href='peminjaman.php?id_peminjaman=" . $peminjaman['id_peminjaman'] .
155 "" onclick=\"return confirm('lanjut delete ?')\">Hapus</a></button> </td>";
156             echo "</tr>";
157         }
158     }
159
160     function tambahpeminjaman($id,$tglpinjam,$tglbalik)
161     {
162         $sql = "INSERT INTO
163 peminjaman(id_peminjaman,tgl_peminjaman,tgl_kembali)values(:id_p,:tgl_
164 pinjam,:tgl_balik)";
165         $stmt = koneksi()->prepare($sql);
166         $result = $stmt->execute(array(':id_p' =>
167 $id,':tgl_pinjam'=>$tglpinjam,':tgl_balik'=>$tglbalik));
168         if (!empty($result)) {
169             header('location:peminjaman.php');
170         }
171     }
172
173     function editpeminjaman(){
174

```

170	\$stmt = koneksi()->prepare("SELECT * FROM peminjaman WHERE
171	id_peminjaman = ".\$_GET['id_peminjaman']);
172	\$stmt->execute();
173	\$GLOBALS['result'] = \$stmt->fetchAll();
174	}
	function updatepeminjaman(\$id,\$tglpinjam,\$tglbalik)
175	{
176	\$pdo_statement = koneksi()->prepare(
177	"update peminjaman set tgl_peminjaman=" . \$tglpinjam . "',
178	tgl_kembali=" . \$tglbalik . "' where id_peminjaman=" . \$id
179);
180	\$result = \$pdo_statement->execute();
181	if(\$result){
182	header('location:peminjaman.php');
	}
183	}
184	function hapuspeminjaman(\$id_peminjaman){
185	\$stmt = koneksi()->prepare("DELETE FROM peminjaman WHERE
186	id_peminjaman=".\$id_peminjaman);
187	\$result= \$stmt->execute();
188	if(\$result){
189	header('location:peminjaman.php');
190	}
	}
	?>
	</body>
	</html>

Tabel 32. Source Code Model.php Soal 1 Modul 5

1	<!DOCTYPE html>
2	<html>
3	<head>
4	<title>Kale</title>
5	<link rel="stylesheet" type="text/css"
	href="http://code.ionicframework.com/ionicons/2.0.1/css/ionicons.min.css" />
6	<link
	href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css"
	rel="stylesheet" integrity="sha384-

	EVSTQN3/azprG1Anm3QDgpJLIm9Nao0Yz1ztcQTwFspd3yD65VohhpuuCOmLASjC"
7	crossorigin="anonymous">
8	<style type="text/css">
9	* {
10	box-sizing: border-box;
11	}
12	body {
13	background-image: url();
14	background-size: cover;
15	font-family: "forte";
16	margin: 0;
17	padding: 0;
18	}
19	header {
20	text-align: left;
21	color: yellow;
22	background-color: #5F9EA0;
23	}
24	section:after {
25	content: "";
26	display: table;
27	clear: both;
28	}
29	nav {
30	float: left;
31	width: 20%;
32	padding: 10px;
33	}
34	article {
35	height: 650px;
36	float: left;
37	width: 100%;
38	padding: 10px;
39	}
40	footer {
41	text-align: center;
42	color: white;
43	padding: 10px;
44	padding-left: 0px;

45	background: tr;
46	}
47	.link_none {
48	color: white;
49	}
50	button {
51	padding: 12.5px 30px;
52	border: 0;
53	border-radius: 100px;
54	background-color: #5F9EA0;
55	color: #ffffff;
56	font-weight: bold;
57	transition: all 0.5s;
58	-webkit-transition: all 0.5s;
59	}
60	button:hover {
61	background-color: grey;
62	box-shadow: 0 0 20px white;
63	transform: scale(1.1);
64	}
65	button:active {
66	background-color: black;
67	transition: all 0.25s;
68	-webkit-transition: all 0.25s;
69	box-shadow: none;
70	transform: scale(0.98);
71	}
72	ul {
73	list-style-type: none;
74	overflow: hidden;
75	background-color: #5F9EA0;
76	background-repeat: no-repeat;
77	padding-left: 18px !important;
78	}
79	li a {
80	display: block;
81	color: white;
82	text-align: center;
83	padding: 20px 20px;

84	text-decoration: none;
85	}
86	li {
87	float: right;
88	}
89	li a:hover {
90	background-color: #272727;
91	}
92	.active {
93	background-color: #272727;
94	}
95	</style>
96	</head>
97	<body>
98	<header>
99	<div class="active">
100	
101	<a href="peminjaman.php"
102	class="link_none">Peminjaman
103	Member
104	Buku
105	
106	
107	</div>
108	</header>
109	<section>
110	<article>
111	<h2 style="text-align:center;">Peminjaman</h2>
112	<center>
113	<table border="1" class="table table-striped">
114	<thead>
115	<tr>
116	<th><p>ID Peminjaman</p></th>
117	<th><p>Tanggal Peminjaman</p></th>
118	<th><p>Tanggal Kembali</p></th>
119	<th><p>Aksi</p></th>
120	</tr>
121	</thead>
	<tbody>

122	<?php
123	require('model.php');
	if (isset(\$_GET['id_peminjaman'])) {
124	
125	hapuspeminjaman(\$_GET['id_peminjaman']);
126	}
127	tampilpeminjaman();
128	?>
129	</tbody>
130	</table>
131	</center>
132	
	<center><button>Tambah Data
133	Peminjaman</button></center>
134	
135	</article>
136	</section>
137	</body>
	</html>

Tabel 33. Source Code Peminjaman.php Soal 1 Modul 5

B. Output Program

Form Buku

ID Buku	<input type="text"/>
Judul Buku	<input type="text"/>
Nama Penulis	<input type="text"/>
Penerbit	<input type="text"/>
Tahun Terbit	<input type="text"/>

Gambar 51. Screenshot Hasil Akhir Soal 1 Modul 5

Form Peminjaman

ID Peminjaman	<input type="text" value="1410"/>
Tanggal Peminjaman	<input type="text" value="dd/mm/yyyy"/> 
Tanggal Kembali	<input type="text" value="dd/mm/yyyy"/> 

Gambar 52. Screenshot Hasil Akhir Soal 1 Modul 5

Form Member

ID Member

Nama Member

Nomor Member

Alamat

Tanggal Daftar

Tanggal bayar Terakhir

Gambar 53. Screenshot Hasil Akhir Soal 1 Modul 5

Buku					
ID Buku	Judul Buku	Penulis	Penerbit	Tahun Terbit	Aksi
0	DI KAKI BUKIT CIBALAK	Ahmad Tohari	Pustaka Jaya	1986	<input type="button" value="Edit"/> <input type="button" value="Hapus"/>
213	and	and	sd	12	<input type="button" value="Edit"/> <input type="button" value="Hapus"/>

Gambar 54. Screenshot Hasil Akhir Soal 1 Modul 5

Member						
ID Member	Nama	Nomor	Alamat	Tanggal Daftar	Tanggal Terakhir Bayar	Aksi
1410	Isak	1	Jl. Tanggusi Baru	2024-06-22 22:45:00	2024-06-22	<input type="button" value="Edit"/> <input type="button" value="Hapus"/>

Gambar 55. Screenshot Hasil Akhir Soal 1 Modul 5

Buku

Member

Peminjaman

Peminjaman

ID Peminjaman	Tanggal Peminjaman	Tanggal Kembali	Aksi
1410	2024-06-22	2024-06-22	<div>Edit</div> <div>Hapus</div>

Tambah Data Peminjaman

Gambar 56. Screenshot Hasil Akhir Soal 1 Modul 5

C. Pembahasan

1) File Buku.php

- `<html>`: Menandakan awal dari halaman HTML.
- `<head>`: Bagian ini mengandung informasi tentang halaman seperti judul, stylesheet, dan meta tag.
- `<title>`: Menetapkan judul halaman web (dalam contoh ini, "Kale").
- `<link>`: Menghubungkan halaman dengan stylesheet eksternal (Ionic Icons dan Bootstrap).
- `<style>`: Menyediakan CSS internal untuk halaman. CSS ini menetapkan tata letak, gaya teks, warna latar belakang, dan animasi tombol.
- `<body>`: Bagian utama dari halaman HTML yang berisi konten yang akan ditampilkan kepada pengguna.
- `<header>`: Bagian ini biasanya berisi elemen yang menampilkan judul atau header utama.
- `` dan ``: Digunakan untuk membuat menu navigasi dengan tautan ke halaman Peminjaman, Member, dan Buku.
- `<a>`: Tautan untuk mengarahkan ke halaman terkait.
- `.active`: Menunjukkan tautan menu yang aktif dengan warna latar belakang yang berbeda.
- `<section>`: Bagian ini mengelompokkan konten halaman, dalam hal ini untuk konten utama.
- `<article>`: Menyediakan area utama yang berisi tabel daftar buku.
- `<h2>`: Menampilkan judul "Buku" dengan teks berpusat.
- `<table>`: Menampilkan tabel untuk menampilkan informasi buku.
- `<thead>` dan `<tbody>`: Bagian kepala dan tubuh tabel.
- `<th>`: Kolom judul tabel.

- `<td>`: Sel data dalam tabel.
- `<?php ... ?>`: Kode PHP untuk memproses logika bisnis seperti menampilkan dan menghapus buku.
- CSS: Digunakan untuk mengatur tata letak, gaya, dan responsivitas halaman.
- Mendefinisikan gaya untuk elemen seperti tombol, latar belakang, dan animasi hover.
- Bootstrap: Framework CSS untuk membangun tata letak responsif.
- Membantu dalam membangun tabel yang responsif dan tombol dengan desain yang konsisten.
- `<?php ... ?>`: Terdapat penggunaan PHP untuk memanggil fungsi dari file `model.php`.
- `require('model.php');`: Memasukkan file PHP eksternal yang berisi definisi fungsi untuk mengakses data buku.
- `tampilbuku()` dan `hapusbuku($_GET['id_buku'])`: Fungsi-fungsi ini digunakan untuk menampilkan daftar buku dan menghapus buku berdasarkan `id_buku` yang dikirim melalui parameter GET.

2) File FormBuku.php

- Require Model (`model.php`):
- Pada baris pertama (`require('model.php');`), kode ini mengimpor file `model.php` yang kemungkinan berisi fungsi-fungsi PHP untuk mengakses dan memanipulasi data buku dari database.
- Edit Buku (`editbuku()`):
- Kode ini mengecek apakah parameter `id_buku` ada pada `$_GET`. Jika ya, maka dipanggil fungsi `editbuku()`. Fungsi ini diharapkan mengambil data buku dari database berdasarkan `id_buku`.
- HTML Form untuk Tambah/Edit Buku:
- Kode HTML form ini digunakan untuk menampilkan form entri buku. Jika parameter `id_buku` ada pada `$_GET`, maka form ini akan diatur untuk mode "Edit". Jika tidak, form akan diatur untuk mode "Tambah".
- Pengaturan judul halaman (`<title>`) juga dinamis berdasarkan keberadaan parameter `id_buku`.
- Input Form Dinamis:
- Setiap input form (ID Buku, Judul Buku, Nama Penulis, Penerbit, Tahun Terbit) memiliki nilai awal yang diatur berdasarkan apakah `id_buku` sudah ada dalam `$_GET` dan data dari database (`$result`).

- Button Aksi (Tambah/Edit):
- Tombol submit form (<button>) akan menyesuaikan tindakan berdasarkan apakah id_buku sudah ada dalam \$_GET. Jika ada, tombol akan menampilkan "Edit" untuk memperbarui data buku. Jika tidak, tombol akan menampilkan "Tambah" untuk menambahkan buku baru.
- Pemrosesan Form:
- Di bagian akhir halaman, setelah form, terdapat kode PHP untuk memproses input dari form.
- Jika tombol "Tambah" (\$_POST['submit']) ditekan, fungsi tambahbuku() dipanggil untuk menambahkan data buku baru ke database.
- Jika tombol "Edit" (\$_POST['update']) ditekan, fungsi updatebuku() dipanggil untuk memperbarui data buku yang sudah ada di database.

3) File FormMember.php

- Require Model (model.php):
- Kode pertama (require('model.php');) mengimpor file model.php, yang diduga berisi definisi fungsi-fungsi untuk mengakses dan memanipulasi data member dari database.
- Edit Member (editmember()):
- Kode ini memeriksa apakah parameter id_member ada dalam \$_GET. Jika ada, maka dipanggil fungsi editmember(). Fungsi ini diharapkan mengambil data member dari database berdasarkan id_member.
- HTML Form untuk Tambah/Edit Member:
- Form HTML ini digunakan untuk menampilkan form entri member. Jika parameter id_member ada dalam \$_GET, form ini akan diatur untuk mode "Edit". Jika tidak, form akan diatur untuk mode "Tambah".
- Judul halaman (<title>) juga diatur secara dinamis berdasarkan keberadaan parameter id_member.
- Input Form Dinamis:
- Setiap input form (ID Member, Nama Member, Nomor Member, Alamat, Tanggal Daftar, Tanggal Bayar Terakhir) memiliki nilai awal yang diatur berdasarkan apakah id_member sudah ada dalam \$_GET dan data dari database (\$result).
- Button Aksi (Tambah/Edit):
- Tombol submit form (<button>) akan menyesuaikan tindakan berdasarkan apakah id_member sudah ada dalam \$_GET. Jika ada, tombol akan menampilkan "Edit" untuk memperbarui data member. Jika tidak, tombol akan menampilkan "Tambah" untuk menambahkan member baru.

- Pemrosesan Form:
- Di bagian akhir halaman, setelah form, terdapat kode PHP untuk memproses input dari form.
- Jika tombol "Tambah" (\$_POST['submit']) ditekan, fungsi tambahmember() dipanggil untuk menambahkan data member baru ke database.
- Jika tombol "Edit" (\$_POST['update']) ditekan, fungsi updatemember() dipanggil untuk memperbarui data member yang sudah ada di database.

4) File FormPeminjaman.php

- Require Model (model.php):
- Kode pertama (require('model.php');) mengimpor file model.php, yang seharusnya berisi definisi fungsi-fungsi untuk mengakses dan memanipulasi data peminjaman dari database.
- Edit Peminjaman (editpeminjaman()):
- Kode ini memeriksa apakah parameter id_peminjaman ada dalam \$_GET. Jika ada, maka dipanggil fungsi editpeminjaman(). Fungsi ini diharapkan mengambil data peminjaman dari database berdasarkan id_peminjaman.
- HTML Form untuk Tambah/Edit Peminjaman:
- Form HTML ini digunakan untuk menampilkan form entri peminjaman. Jika parameter id_peminjaman ada dalam \$_GET, form ini akan diatur untuk mode "Edit". Jika tidak, form akan diatur untuk mode "Tambah".
- Judul halaman (<title>) juga diatur secara dinamis berdasarkan keberadaan parameter id_peminjaman.
- Input Form Dinamis:
- Setiap input form (ID Peminjaman, Tanggal Peminjaman, Tanggal Kembali) memiliki nilai awal yang diatur berdasarkan apakah id_peminjaman sudah ada dalam \$_GET dan data dari database (\$result).
- Button Aksi (Tambah/Edit):
- Tombol submit form (<button>) akan menyesuaikan tindakan berdasarkan apakah id_peminjaman sudah ada dalam \$_GET. Jika ada, tombol akan menampilkan "Edit" untuk memperbarui data peminjaman. Jika tidak, tombol akan menampilkan "Tambah" untuk menambahkan peminjaman baru.
- Pemrosesan Form:
- Di bagian akhir halaman, setelah form, terdapat kode PHP untuk memproses input dari form.

- Jika tombol "Tambah" (\$_POST['submit']) ditekan, fungsi tambahpeminjaman() dipanggil untuk menambahkan data peminjaman baru ke database.
- Jika tombol "Edit" (\$_POST['update']) ditekan, fungsi updatepeminjaman() dipanggil untuk memperbarui data peminjaman yang sudah ada di database.

5) File Index.php

- HTML
- <!DOCTYPE html>: Mendefinisikan bahwa dokumen ini adalah dokumen HTML versi terbaru.
- <html>, <head>, <title>, <link>, dan <style>: Bagian standar dari sebuah dokumen HTML yang digunakan untuk mengatur struktur dan gaya halaman.
- <header>: Bagian ini berisi menu navigasi utama halaman.
- CSS
- * { box-sizing: border-box; }: Mengatur box-sizing secara global agar padding dan border tidak mempengaruhi ukuran total elemen.
- body: Mengatur font-family, background image, margin, dan padding secara global untuk halaman.
- header: Mendefinisikan gaya untuk bagian header, termasuk pengaturan teks tengah dan latar belakang warna abu-abu (#737373).
- .link_none: Menetapkan gaya untuk tautan di dalam menu navigasi, termasuk warna putih dan ukuran font 18px.
- ul: Menetapkan gaya untuk daftar tak berurutan (unordered list) yang berfungsi sebagai menu navigasi, termasuk latar belakang, padding, dan posisi background image.
- li: Menetapkan gaya untuk elemen daftar (list item) di dalam menu navigasi, termasuk tata letak inline-block dan margin kanan.
- li a: Menetapkan gaya untuk tautan di dalam elemen daftar, termasuk tata letak block, warna teks putih, padding, dan penghapusan dekorasi teks.
- li a:hover dan .active: Memberikan gaya interaksi saat tautan menu dinavigasi dengan kursor, seperti perubahan warna latar belakang.
- Pengaturan Layout
- nav: Ditetapkan untuk tata letak float kiri dengan lebar 20% dari lebar halaman, memberikan ruang untuk menu navigasi.
- article: Digunakan untuk menetapkan bagian konten utama dengan tinggi 650px dan padding 10px, mengambil sisa 80% lebar halaman.

- footer: Ditetapkan di bagian bawah halaman dengan teks berwarna putih dan padding.

6) File Koneksi.php

- `function koneksi()`
- Ini adalah deklarasi fungsi bernama `koneksi()` yang tidak menerima parameter apa pun.
- `try { ... } catch(PDOException $e) { ... }`
- Blok `try { ... } catch(PDOException $e) { ... }` digunakan untuk mencoba menjalankan kode yang mungkin dapat menghasilkan pengecualian (exception).
- PDO adalah kelas utama untuk koneksi dan operasi database dengan PDO.
- `new PDO('mysql:host=localhost;dbname=perpus','root','...')` adalah kode yang melakukan inisialisasi objek PDO untuk koneksi ke database MySQL dengan detail koneksi berikut:
 - `mysql:host=localhost`: Menunjukkan host database adalah localhost.
 - `dbname=perpus`: Menunjukkan database yang akan digunakan adalah perpus.
 - `'root'` dan `''`: Merupakan username dan password untuk mengakses database MySQL. Di sini, username adalah root dan password kosong.
- `array(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION, PDO::ATTR_PERSISTENT=>TRUE)`: Array ini memberikan opsi konfigurasi tambahan untuk PDO:
 - `PDO::ATTR_ERRMODE` dan `PDO::ERRMODE_EXCEPTION`: Mengatur mode error PDO ke `ERRMODE_EXCEPTION`, yang berarti PDO akan melemparkan pengecualian (exception) saat terjadi kesalahan.
 - `PDO::ATTR_PERSISTENT=>TRUE`: Membuat koneksi PDO menjadi persisten, yang berarti koneksi tidak akan ditutup setelah skrip PHP selesai dieksekusi.
- `catch(PDOException $e)`
- Ini menangkap pengecualian (exception) jika terjadi kesalahan saat menjalankan kode dalam blok `try`.
- `PDOException $e` adalah jenis pengecualian yang ditangkap. Jika terjadi kesalahan, pesan kesalahan ditampilkan dengan menggunakan `$e->getMessage()` yang akan menampilkan pesan kesalahan PDO dalam pesan.

- `print "Koneksi atau query bermasalah: ".$e->getMessage() . "
";`
Menampilkan pesan kesalahan koneksi atau query jika terjadi masalah, dan kemudian program dihentikan dengan `die()`.
- `return $pdo_conn;`
- Jika koneksi berhasil, objek PDO yang sudah terkoneksi akan dikembalikan (`return $pdo_conn;`) ke pemanggil fungsi koneksi().

7) File Member.php

- `<!DOCTYPE html>`
- Mendefinisikan bahwa dokumen ini adalah dokumen HTML versi terbaru.
- `<html>`, `<head>`, `<title>`
- Struktur dasar dokumen HTML dengan mendefinisikan judul halaman.
- `<link>` tags
- Menghubungkan CSS eksternal dari Ionic Icons dan Bootstrap untuk memperkaya gaya dan fungsi halaman.
- `<style>` tags
- CSS internal untuk menetapkan gaya halaman secara lebih spesifik.
- `<body>`
- Isi utama halaman web, termasuk header, section, dan artikel.
- `<header>`
- Bagian atas halaman yang menampilkan menu navigasi.
- `<section>` dan `<article>`
- `<section>` digunakan untuk menetapkan area konten utama.
- `<article>` digunakan untuk menempatkan konten spesifik, dalam hal ini tabel daftar member.
- `<table>`, `<thead>`, `<tbody>`, `<tr>`, `<th>`, `<td>`
- Struktur tabel HTML untuk menampilkan data member.
- `<thead>` untuk bagian header tabel.
- `<tbody>` untuk bagian isi tabel.
- `<tr>` untuk baris tabel.
- `<th>` untuk sel header.
- `<td>` untuk sel data.
- PHP Embedded
- Menggunakan PHP untuk memanggil fungsi dari `model.php`, seperti `hapusmember()` dan `tampilmember()` untuk mengelola dan menampilkan data member.

- `<button>Tambah Member</button>` digunakan untuk menambahkan tombol yang mengarahkan pengguna ke halaman form untuk menambah member baru.

8) File Model.php

- `<!DOCTYPE html>` dan tag `<html>` sebagai penanda bahwa ini adalah dokumen HTML dengan bahasa Inggris.
- `<head>` berisi meta tag untuk pengaturan karakter, kompatibilitas, viewport, dan referensi ke CSS Bootstrap.
- `<title>` menentukan judul halaman.
- CSS inline untuk menyesuaikan tata letak halaman.
- Koneksi ke Database: Menggunakan file koneksi.php untuk menghubungkan dan memulai koneksi PDO ke database MySQL.
- Fungsi CRUD (Create, Read, Update, Delete):
- Buku (tampilbuku, tambahBuku, editbuku, updatebuku, hapusbuku):
- tampilbuku(): Menampilkan semua buku dari database dalam bentuk tabel.
- tambahBuku(): Menambahkan buku baru ke database.
- editbuku(): Mengambil data buku berdasarkan id_buku untuk proses edit.
- updatebuku(): Mengupdate data buku berdasarkan id_buku.
- hapusbuku(): Menghapus buku berdasarkan id_buku.
- Member (tampilmember, tambahmember, editmember, updatemember, hapusmember):
- tampilmember(): Menampilkan semua member dari database dalam bentuk tabel.
- tambahmember(): Menambahkan member baru ke database.
- editmember(): Mengambil data member berdasarkan id_member untuk proses edit.
- updatemember(): Mengupdate data member berdasarkan id_member.
- hapusmember(): Menghapus member berdasarkan id_member.
- Peminjaman (tampilpeminjaman, tambahpeminjaman, editpeminjaman, updatepeminjaman, hapuspeminjaman):
- tampilpeminjaman(): Menampilkan semua data peminjaman dari database dalam bentuk tabel.
- tambahpeminjaman(): Menambahkan data peminjaman baru ke database.
- editpeminjaman(): Mengambil data peminjaman berdasarkan id_peminjaman untuk proses edit.
- updatepeminjaman(): Mengupdate data peminjaman berdasarkan id_peminjaman.

- hapuspeminjaman(): Menghapus data peminjaman berdasarkan id_peminjaman.

9) File Peminjaman.php

- DOCTYPE dan HTML: Deklarasi tipe dokumen dan awal tag HTML.
- Head Section: Berisi judul halaman, referensi ke CSS (Bootstrap dan Ionic framework), serta sedikit CSS internal untuk styling tambahan.
- Body Section: Memuat konten utama halaman.
- CSS Styling
- Universal Styling: Menggunakan box-sizing: border-box untuk mengatur model kotak secara global.
- Body Styling: Menentukan background image (walaupun dalam kode tidak ada URL tertentu yang diberikan), menggunakan font family "forte", serta mengatur margin dan padding menjadi nol.
- Header, Section, dan Article: Menyusun tata letak halaman dengan header yang berisi menu navigasi, section untuk konten utama, dan article untuk menampilkan data peminjaman.
- Footer: Menentukan tata letak dan styling untuk footer, meskipun pada kode tersebut footer tidak ditampilkan secara eksplisit.
- PHP Integration
- PHP Embedding: Terdapat skrip PHP yang digunakan untuk memproses data peminjaman.
- Include Model: Menggunakan require('model.php') untuk menyertakan file PHP yang mungkin berisi definisi fungsi atau koneksi ke database.
- Pemanggilan Fungsi:
- tampilpeminjaman(): Memanggil fungsi ini untuk menampilkan semua data peminjaman dari database dalam bentuk tabel. Setiap baris memiliki tombol untuk mengedit atau menghapus data.
- hapuspeminjaman(\$_GET['id_peminjaman']): Memanggil fungsi hapuspeminjaman() jika ada parameter id_peminjaman yang dikirimkan melalui URL untuk menghapus data peminjaman tersebut dari database.
- Interaksi dengan Pengguna
- Tabel Data: Menampilkan data peminjaman dalam tabel dengan kolom ID Peminjaman, Tanggal Peminjaman, Tanggal Kembali, dan Aksi (untuk mengedit atau menghapus).
- Tombol Aksi:

- Tambah Data Peminjaman: Tombol ini mengarahkan pengguna ke halaman formpeminjaman.php untuk menambahkan data peminjaman baru.
- Edit dan Hapus: Masing-masing baris pada tabel memiliki tombol untuk mengedit dan menghapus data peminjaman. Tombol hapus mengkonfirmasi pengguna sebelum menghapus data.

Tautan Git

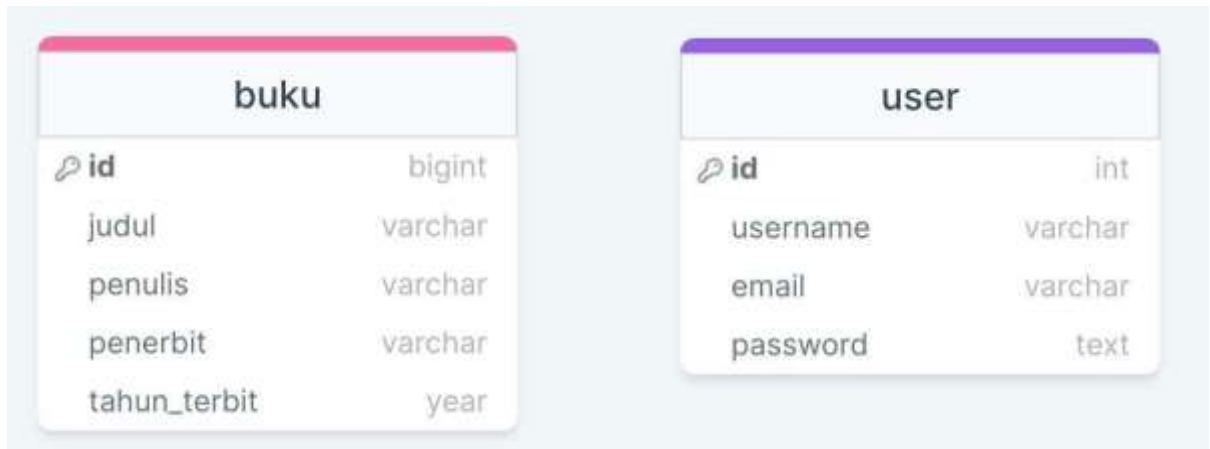
https://github.com/Vainn11/Prak_Web_2/tree/da21a5fbff792bc5e8002f4279d0bfb62e867a8e/Modul%205

MODUL 6

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:



Gambar 57. Gambar Soal 1 Modul 6

Simpan dengan nama projek di Github: PRAK601

A. Source Code

```

1  <?php
2
3  use CodeIgniter\Router\RouteCollection;
4
5  /**
6   * @var RouteCollection $routes
7   */
8  $routes->setAutoRoute(true);
9  $routes->get('/', 'Buku::login');
10 $routes->get('home', 'Buku::index');
    $routes->get('/register', 'Register::index');
    $routes->post('/register/process', 'Register::process');
    $routes->get('/login', 'Login::index');
    $routes->post('/login/process', 'Login::process');
    $routes->get('/logout', 'Login::logout');
    $routes->get('/buku/create', 'Buku::create');
    $routes->post('buku/store', 'Buku::store');
    $routes->post('buku/update/', 'Buku::update', ['filter' => 'auth']);
    $routes->get('buku/edit/', 'Buku::edit', ['filter' => 'auth']);
    $routes->get('buku/delete/', 'Buku::delete', ['filter' => 'auth']);

```

Tabel 34. Source Code Routes.php Soal 1 Modul 6

1	<?php namespace App\Controllers;
2	
3	use App\Models\BukuModel;
4	use CodeIgniter\Controller;
5	use App\Models\PostModel;
6	
7	class Buku extends Controller
8	{
9	/**
10	*
11	* @var Model
12	*/
13	protected \$model;
14	
15	public function __construct()
16	{
17	\$this->model = new BukuModel();
18	\$this->helpers = ['form', 'url'];
19	
20	}
21	
22	public function login()
23	{
24	\$data = [
25	'posts' => \$this->model->paginate(10),
26	'pager' => \$this->model->pager,
27	'title' => 'POST LIST'
28];
29	
30	return view('vw_login', \$data);
31	}
32	
33	public function index()
34	{
35	if (!session()->get('logged_in')) {
36	return redirect()->to(base_url('/login'));
37	}
38	\$data = [
39	'posts' => \$this->model->paginate(10),
40	'pager' => \$this->model->pager,
41	'title' => 'POST LIST'
42];

```

43
44     return view('vw_buku', $data);
45 }
46
47 public function create()
48 {
49     // Check if user is authenticated
50     if (!session()->get('logged_in')) {
51         return redirect()->to(base_url('/login'));
52     }
53     $data = ['title' => 'Create new post'];
54
55     return view('vw_add', $data);
56 }
57
58 public function store()
59 {
60     $data = $this->request->getPost(['judul', 'penulis', 'penerbit',
61 'tahun_terbit']);
62
63     if (! $this->validateData($data, $this->model->validationRules)) {
64         return $this->create();
65     }
66
67     $post = $this->validator->getValidated();
68
69     $save = $this->model->save([
70         'judul' => $post['judul'],
71         'penulis' => $post['penulis'],
72         'penerbit' => $post['penerbit'],
73         'tahun_terbit' => $post['tahun_terbit'],
74     ]);
75
76     session()->setFlashdata('success', 'Post has been added successfully.');
```

```

77     return redirect()->to(base_url('home'));
78 }
79
80 public function edit($id)
81 {
82     // Check if user is authenticated
83     if (!session()->get('logged_in')) {
84         return redirect()->to(base_url('/login'));

```

```

85     }
86     $post = $this->model->find($id);
87     if (empty($post)) {
88         session()->setFlashdata('error','Post not found');
89         return redirect()->back();
90     }
91     $data = [
92         'title' => 'Edit Post',
93         'post' => $post
94     ];
95
96     return view('vw_edit', $data);
97 }
98
99 public function update($id)
100 {
101     // Check if user is authenticated
102     if (!session()->get('logged_in')) {
103         return redirect()->to(base_url('/login'));
104     }
105     $post = $this->model->find($id);
106
107     if (empty($post)) {
108         session()->setFlashdata('error','Post not found');
109         return redirect()->back();
110     }
111
112     $data = $this->request->getPost(['judul', 'penulis', 'penerbit',
113 'tahun_terbit']);
114
115     if (!$this->validateData($data, $this->model->validationRules)) {
116         return $this->create();
117     }
118
119     $updatedPost = $this->validator->getValidated();
120
121     $update = $this->model->update($post['id'], $updatedPost);
122
123     if ($update) {
124         session()->setFlashdata('success', 'Post has been updated successfully');
125         return redirect()->to(base_url('home'));
126     } else {

```

127	session()->setFlashdata('error', 'Some problems occurred, please try
128	again.');
129	return redirect()->back();
130	}
131	}
132	
133	public function delete(\$id)
134	{
135	// Check if user is authenticated
136	if (!session()->get('logged_in')) {
137	return redirect()->to(base_url('home'));
138	}
139	
140	if (empty(\$id)) {
141	return redirect()->to(base_url('home'));
142	}
143	
144	\$delete = \$this->model->delete(\$id);
145	
146	if (\$delete) {
147	session()->setFlashdata('success', 'Post has been removed
148	successfully.');
149	return redirect()->to(base_url('home'));
150	} else {
151	session()->setFlashdata('error', 'Some problems occurred, please try
152	again.');
153	return redirect()->to(base_url('home'));
154	}
155	}
	}

Tabel 35. Source Code Buku.php di Controller Soal 1 Modul 6

1	<?php
2	
3	namespace App\Controllers;
4	
5	class Home extends BaseController
6	{
7	public function index(): string
8	{
9	return view('vw_buku');
10	}
11	}

Tabel 36. Source Code Home.php Soal 1 Modul 6

1	<?php
2	
3	namespace App\Controllers;
4	
5	use App\Models\UsersModel;
6	
7	class Login extends BaseController
8	{
9	public function index()
10	{
11	return view('vw_login');
12	}
13	
14	public function process()
15	{
16	\$users = new UsersModel();
17	\$username = \$this->request->getVar('username');
18	\$password = \$this->request->getVar('password');
19	\$dataUser = \$users->where('username', \$username)->first();
20	
21	if (\$dataUser) {
22	// Check the password using object property syntax
23	if (password_verify(\$password, \$dataUser->password)) {
24	session()->set([
25	'username' => \$dataUser->username,
26	'logged_in' => true
27]);
28	return redirect()->to(base_url('home'));
29	} else {
30	session()->setFlashdata('error', 'Username & Password Salah');
31	return redirect()->to(base_url('login'));
32	}
33	} else {
34	session()->setFlashdata('error', 'Username & Password Salah');
35	return redirect()->to(base_url('login'));
36	}
37	}
38	
39	public function logout()
40	{
41	session()->destroy();
42	return redirect()->to(base_url('login'));

43	}
44	}
45	

Tabel 37. Source Code Login.php Soal 1 Modul 6

1	<?php	
2		
3	namespace App\Controllers;	
4		
5	use App\Models\UsersModel;	
6		
7	class Register extends BaseController	
8	{	
9	public function index()	
10	{	
11	return view('vw_register');	
12	}	
13		
14	public function process()	
15	{	
16	// Validate input	
17	if (!\$this->validate([
18	'username' => [
19		'rules' =>
20	'required min_length[4] max_length[20] is_unique[users.username]',	
21	'errors' => [
22	'required' => '{field} Harus diisi',	
23	'min_length' => '{field} Minimal 4 Karakter',	
24	'max_length' => '{field} Maksimal 20 Karakter',	
25	'is_unique' => 'Username sudah digunakan sebelumnya'	
26]	
27],	
28	'password' => [
29	'rules' => 'required min_length[0] max_length[50]',	
30	'errors' => [
31	'required' => '{field} Harus diisi',	
32	'min_length' => '{field} Minimal 0 Karakter',	
33	'max_length' => '{field} Maksimal 50 Karakter',	
34]	
35],	
36	'password_conf' => [
37	'rules' => 'matches[password]',	
38	'errors' => [
39	'matches' => 'Konfirmasi Password tidak sesuai dengan password',	
40]	
41],	
42	'email' => [

43	'rules' => 'required min_length[4] max_length[100] valid_email',
44	'errors' => [
45	'required' => '{field} Harus diisi',
46	'min_length' => '{field} Minimal 4 Karakter',
47	'max_length' => '{field} Maksimal 100 Karakter',
48	'valid_email' => 'Email tidak valid'
49]
50],
51)) {
52	session()->setFlashdata('error', \$this->validator->listErrors());
53	return redirect()->back()->withInput();
54	}
55	
56	\$users = new UsersModel();
57	
58	// Hash the password before storing it
59	\$hashedPassword = password_hash(\$this->request->getVar('password'),
60	PASSWORD_DEFAULT);
61	
62	\$users->insert([
63	'username' => \$this->request->getVar('username'),
64	'password' => \$hashedPassword,
65	'email' => \$this->request->getVar('email')
66]);
67	
68	return redirect()->to('/login');
	}
	}

Tabel 38. Source Code Register.php Soal 1 Modul 6

1	<?php
2	// app/Filters/AuthFilter.php
3	
4	namespace App\Filters;
5	
6	use CodeIgniter\HTTP\RequestInterface;
7	use CodeIgniter\HTTP\ResponseInterface;
8	use CodeIgniter\Filters\FilterInterface;
9	
10	class AuthFilter implements FilterInterface
11	{
12	public function before(RequestInterface \$request, \$arguments = null)
13	{
14	// Check if the user is logged in
15	if (!session()->has('user_id')) {
16	// If not logged in, redirect to the login page
17	return redirect()->to('/login');
18	}
19	}
20	
21	public function after(RequestInterface \$request, ResponseInterface
22	\$response, \$arguments = null)
23	{
24	// Do nothing after the request
25	}
	}

Tabel 39. Source Code Authfilter.php Soal 1 Modul 6

1	<?= \$this->extend('layouts/main') ?>
2	
3	<?= \$this->section('content') ?>
4	<style>
5	body {
6	background-color: #121212;
7	background-repeat: no-repeat;
8	background-attachment: fixed;
9	background-position: center;
10	background-size: cover;
11	color: #ffffff;
12	}
13	
14	.card {
15	background-color: #1e1e1e;
16	color: #ffffff;
17	}
18	
19	.card-header {
20	background-color: #6200ea;
21	}
22	
23	.form-control {
24	background-color: #2c2c2c;
25	color: #ffffff;
26	border: 1px solid #444;
27	}
28	
29	.form-control::placeholder {
30	color: #888;
31	}
32	
33	.form-group label {
34	color: #ffffff;
35	}
36	
37	.btn-success {
38	background-color: #bb86fc;
39	border-color: #bb86fc;
40	color: #000000;
41	}
42	

43	.btn-success:hover {
44	background-color: #3700b3;
45	border-color: #3700b3;
46	color: #ffffff;
47	}
48	
49	.btn-secondary {
50	background-color: #333;
51	border-color: #333;
52	color: #ffffff;
53	}
54	
55	.btn-secondary:hover {
56	background-color: #555;
57	border-color: #555;
58	color: #ffffff;
59	}
60	
61	.alert-danger {
62	background-color: #ff4444;
63	color: #ffffff;
64	border: none;
65	}
66	</style>
67	
68	<div class="container mt-5">
69	<div class="row justify-content-center">
70	<div class="col-md-8">
71	<div class="card shadow-sm">
72	<div class="card-header text-white">
73	<h4>Tambah</h4>
74	</div>
75	<div class="card-body">
76	<?php if (session()->getFlashdata('error')) : ?>
77	<div class="alert alert-danger">
78	<?php echo session()->getFlashdata('error'); ?>
79	</div>
80	<?php endif; ?>
81	<?= validation_list_errors() ?>
82	<?= form_open('buku/store'); ?>

```

83         <div class="form-group">
84             <label for="judul">Judul</label>
85             <input type="text" class="form-control" id="judul"
86 name="judul" placeholder="Masukkan Judul">
87         </div>
88         <div class="form-group">
89             <label for="penulis">Penulis</label>
90             <input type="text" class="form-control" id="penulis"
91 name="penulis" placeholder="Masukkan Penulis">
92         </div>
93         <div class="form-group">
94             <label for="penerbit">Penerbit</label>
95             <input type="text" class="form-control" id="penerbit"
96 name="penerbit" placeholder="Masukkan Penerbit">
97         </div>
98         <div class="form-group">
99             <label for="tahun_terbit">Tahun Terbit</label>
100             <input type="number" class="form-control" id="tahun_terbit"
101 name="tahun_terbit" placeholder="Masukkan Tahun Terbit">
102         </div>
103         <div class="form-group text-right">
104             <button class="btn btn-success">Simpan</button>
105             <a href="<?= base_url('home') ?>" class="btn btn-
106 secondary">Kembali</a>
107         </div>
108         <?= form_close(); ?>
109     </div>
110 <?= $this->endSection() ?>
111
112 <?= $this->section('extra-js') ?>
113 <!-- include summernote css/js -->
114 <link
115 href="https://cdn.jsdelivr.net/npm/summernote@0.8.15/dist/summernote-
116 bs4.min.css" rel="stylesheet">
117 <script
118 src="https://cdn.jsdelivr.net/npm/summernote@0.8.15/dist/summernote-
119 bs4.min.js"></script>

```


120	<script>
121	\$(document).ready(function() {
122	\$('#post_content').summernote({
123	tabsize: 2,
124	height: 500
125	});
126	});
127	</script>
128	<?= \$this->endSection() ?>
129	
130	
131	
132	
133	
134	
135	
136	
137	
138	

Tabel 40. Source Code vw_add.php Soal 1 Modul 6

1	<?= \$this->extend('layouts/main') ?>
2	
3	<?= \$this->section('content') ?>
4	<style>
5	body {
6	background-color: #121212;
7	background-repeat: no-repeat;
8	background-attachment: fixed;
9	background-position: center;
10	background-size: cover;
11	color: #ffffff;
12	}
13	
14	.navbar-light {
15	background-color: #1f1f1f;
16	}
17	
18	.navbar-light .navbar-brand,
19	.navbar-light .navbar-nav .nav-link {
20	color: #ffffff;
21	}
22	
23	.navbar-toggler {
24	border-color: rgba(255, 255, 255, 0.1);
25	}
26	
27	.navbar-toggler-icon {
28	background-image: url("data:image/svg+xml;charset=utf8,%3Csvg
29	viewBox='0 0 30 30' xmlns='http://www.w3.org/2000/svg'%3E%3Cpath
30	
31	
32	
33	
34	
35	
36	
37	
38	
39	
40	
41	
42	

43	stroke='rgba%2855, 55, 55, 0.7%29' stroke-width='2' stroke-linecap='round'
44	stroke-miterlimit='10' d='M4 7h22M4 15h22M4 23h22'/%3E%3C/svg%3E");
45	}
46	
47	.card {
48	background-color: #1e1e1e;
49	color: #ffffff;
50	}
51	
52	.card-header {
53	background-color: #6200ea;
54	}
55	
56	.btn {
57	border: none;
58	}
59	
60	.btn-sm {
61	background-color: #bb86fc;
62	color: #000000;
63	}
64	
65	.btn-sm:hover {
66	background-color: #3700b3;
67	color: #ffffff;
68	}
69	
70	.alert-success {
71	background-color: #00c851;
72	color: #ffffff;
73	border: none;
74	}
75	
76	.alert-danger {
77	background-color: #ff4444;
78	color: #ffffff;
79	border: none;
80	}
81	
82	.table {
83	background-color: #1e1e1e;
84	color: #ffffff;

```

85     }
86
87     .table thead {
88         background-color: #6200ea;
89         color: #ffffff;
90     }
91
92     .table-bordered,
93     .table-bordered td,
94     .table-bordered th {
95         border-color: #444;
96     }
97
98     .pagination .page-item .page-link {
99         background-color: #1e1e1e;
100        color: #ffffff;
101        border-color: #6200ea;
102    }
103
104    .pagination .page-item .page-link:hover {
105        background-color: #6200ea;
106        color: #ffffff;
107        border-color: #3700b3;
108    }
109 </style>
110
111 <!-- Navbar -->
112 <nav class="navbar navbar-expand-lg navbar-light bg-light shadow-sm">
113     <div class="container">
114         <a class="navbar-brand" href="<?= base_url('/') ?>">My Library</a>
115         <button class="navbar-toggler" type="button" data-toggle="collapse"
116 data-target="#navbarNav" aria-controls="navbarNav" aria-expanded="false"
117 aria-label="Toggle navigation">
118             <span class="navbar-toggler-icon"></span>
119         </button>
120         <div class="collapse navbar-collapse" id="navbarNav">
121             <ul class="navbar-nav ml-auto">
122                 <?php if (session()->get('logged_in')) : ?>
123                     <li class="nav-item">
124                         <a class="nav-link" href="#">Welcome, <?= session()-
125 >get('username'); ?></a>
126                     </li>

```

```

127         <li class="nav-item">
128             <a class="nav-link" href="<?= base_url('login/logout');
129 ?>">Logout</a>
130         </li>
131         <?php else : ?>
132             <li class="nav-item">
133                 <a class="nav-link" href="<?= base_url('login');
134 ?>">Login</a>
135             </li>
136             <li class="nav-item">
137                 <a class="nav-link" href="<?= base_url('register');
138 ?>">Register</a>
139             </li>
140         <?php endif; ?>
141     </ul>
142 </div>
143 </div>
144 </nav>
145
146 <div class="container mt-5">
147     <div class="row justify-content-center">
148         <div class="col-md-10">
149             <div class="card shadow-sm">
150                 <div class="card-header text-white">
151                     <h5 class="mb-0">List Buku</h5>
152                     <a href="<?= base_url('buku/create'); ?>" class="btn btn-sm
153 float-right">Tambah Data</a>
154                 </div>
155                 <div class="card-body">
156                     <?php if (session()->getFlashdata('success')) : ?>
157                         <div class="alert alert-success">
158                             <?= session()->getFlashdata('success'); ?>
159                         </div>
160                     <?php endif; ?>
161
162                     <?php if (session()->getFlashdata('error')) : ?>
163                         <div class="alert alert-danger">
164                             <?= session()->getFlashdata('error'); ?>
165                         </div>
166                     <?php endif; ?>
167
168                     <table class="table table-bordered table-striped table-hover">

```

169	<thead class="text-center">
170	<tr>
171	<th scope="col">Judul</th>
172	<th scope="col">Penulis</th>
173	<th scope="col">Penerbit</th>
174	<th scope="col">Tahun Terbit</th>
175	<th scope="col">Action</th>
176	</tr>
177	</thead>
178	<tbody>
179	<?php if (!empty(\$posts) && is_array(\$posts)) : ?>
180	<?php foreach (\$posts as \$row) : ?>
181	<tr>
182	<td><?= \$row['judul']; ?></td>
183	<td><?= \$row['penulis']; ?></td>
184	<td><?= \$row['penerbit']; ?></td>
185	<td><?= \$row['tahun_terbit']; ?></td>
186	<td class="text-center">
187	<form onsubmit="return confirm('Apakah Anda
188	Yakin ?');;" action="<?= base_url('buku/delete/' . \$row['id']); ?>"
189	method="POST">
190	<input type="hidden" name="{csrf_token}"
191	value="{csrf_hash}">
192	<input type="hidden" name="_method"
193	value="DELETE">
194	<a href="<?= base_url('buku/edit/' . \$row['id']);
195	?>" class="btn btn-warning btn-sm">Edit
196	<button type="submit" class="btn btn-danger
197	btn-sm">Delete</button>
198	</form>
199	</td>
200	</tr>
201	<?php endforeach; ?>
202	<?php else : ?>
203	<tr>
204	<td colspan="5" class="text-center">No post
205	found.</td>
	</tr>
	<?php endif; ?>
	</tbody>
	</table>

	<pre> <div class="d-flex justify-content-center"> <?= \$pager->links(); ?> </div> </div> </div> </div> </div> </div> </div> </div> <?= \$this->endSection() ?> <?= \$this->section('extra-js') ?> <script> \$(document).ready(function() { \$('.pagination li').addClass('page-item'); \$('.pagination li a').addClass('page-link'); }); </script> <?= \$this->endSection() ?> </pre>
--	---

Tabel 41. Source Code vw_buku.php Soal 1 Modul 6

1	<?= \$this->extend('layouts/main') ?>
2	
3	<?= \$this->section('content') ?>
4	<style>
5	body {
6	background-color: #121212;
7	background-repeat: no-repeat;
8	background-attachment: fixed;
9	background-position: center;
10	background-size: cover;
11	color: #ffffff;
12	}
13	
14	.card {
15	background-color: #1e1e1e;
16	color: #ffffff;
17	}
18	
19	.card-header {
20	background-color: #6200ea;
21	}
22	
23	.form-control {
24	background-color: #2c2c2c;
25	color: #ffffff;
26	border: 1px solid #444;
27	}
28	
29	.form-control::placeholder {
30	color: #888;
31	}
32	
33	.form-group label {
34	color: #ffffff;
35	}
36	
37	.btn-success {
38	background-color: #bb86fc;
39	border-color: #bb86fc;
40	color: #000000;
41	}
42	

43	.btn-success:hover {
44	background-color: #3700b3;
45	border-color: #3700b3;
46	color: #ffffff;
47	}
48	
49	.btn-secondary {
50	background-color: #333;
51	border-color: #333;
52	color: #ffffff;
53	}
54	
55	.btn-secondary:hover {
56	background-color: #555;
57	border-color: #555;
58	color: #ffffff;
59	}
60	
61	.alert-danger {
62	background-color: #ff4444;
63	color: #ffffff;
64	border: none;
65	}
66	</style>
67	
68	<div class="container mt-5">
69	<div class="row justify-content-center">
70	<div class="col-md-8">
71	<div class="card shadow-sm">
72	<div class="card-header text-white">
73	<h4>Edit</h4>
74	</div>
75	<div class="card-body">
76	
77	<?php if (session()->getFlashdata('error')) : ?>
78	<div class="alert alert-danger">
79	<?= session()->getFlashdata('error'); ?>
80	</div>
81	<?php endif; ?>
82	
83	<?= validation_list_errors() ?>
84	

```

85      <?= form_open('buku/update/' . $post['id']); ?>
86      <input type="hidden" name="_method" value="PUT">
87
88      <div class="form-group">
89          <label for="judul">Judul</label>
90          <input type="text" class="form-control" id="judul"
91 name="judul" placeholder="Masukkan Judul" value="<?= $post['judul'] ?>">
92      </div>
93      <div class="form-group">
94          <label for="penulis">Penulis</label>
95          <input type="text" class="form-control" id="penulis"
96 name="penulis" placeholder="Masukkan Penulis" value="<?=
97 $post['penulis'] ?>">
98      </div>
99      <div class="form-group">
100          <label for="penerbit">Penerbit</label>
101          <input type="text" class="form-control" id="penerbit"
102 name="penerbit" placeholder="Masukkan Penerbit" value="<?=
103 $post['penerbit'] ?>">
104      </div>
105      <div class="form-group">
106          <label for="tahun_terbit">Tahun Terbit</label>
107          <input type="number" class="form-control" id="tahun_terbit"
108 name="tahun_terbit" placeholder="Masukkan Tahun Terbit" value="<?=
109 $post['tahun_terbit'] ?>">
110      </div>
111      <div class="form-group text-right">
112          <button class="btn btn-success">Update</button>
113          <a href="<?= base_url('home') ?>" class="btn btn-
114 secondary">Kembali</a>
115      </div>
116      <?= form_close(); ?>
117
118      </div>
119      </div>
120      </div>
121      </div>
122      </div>
123
124      <?= $this->endSection() ?>
125
126      <?= $this->section('extra-js') ?>

```

127	<!-- include summernote css/js -->
128	<link
129	href="https://cdn.jsdelivr.net/npm/summernote@0.8.15/dist/summernote-
130	bs4.min.css" rel="stylesheet">
131	<script
132	src="https://cdn.jsdelivr.net/npm/summernote@0.8.15/dist/summernote-
133	bs4.min.js"></script>
134	<script>
135	\$(document).ready(function() {
136	\$('#post_content').summernote({
137	tabsize: 2,
138	height: 500,
139	styleTags: [
140	'p', 'blockquote', 'pre', 'h1', 'h2', 'h3', 'h4', 'h5', 'h6'
141],
142	toolbar: [
143	['style', ['style']],
144	['font', ['bold', 'italic', 'underline', 'clear']],
145	['fontname', ['fontname']],
146	['color', ['color']],
147	['para', ['ul', 'ol', 'paragraph']],
148	['table', ['table']],
149	['insert', ['link', 'picture', 'video']],
150	['view', ['fullscreen', 'codeview', 'help']]
151]
152	});
153	});
154	</script>
155	<?= \$this->endSection() ?>
156	
157	
158	

Tabel 42. Source Code vw_edit.php Soal 1 Modul 6

1	<!DOCTYPE html>
2	<html lang="en">
3	
4	<head>
5	<meta charset="UTF-8">
6	<meta name="viewport" content="width=device-width, initial-scale=1.0">
7	<title>Login</title>
8	<style>
9	body {
10	display: flex;
11	justify-content: center;
12	align-items: center;
13	min-height: 100vh;
14	padding: 0;
15	margin: 0;
16	font-family: Arial, sans-serif;
17	background-color: #121212;
18	background-repeat: no-repeat;
19	background-attachment: fixed;
20	background-position: center;
21	background-size: cover;
22	}
23	
24	.form-signin {
25	width: 100%;
26	max-width: 330px;
27	padding: 15px;
28	margin: auto;
29	background: #1e1e1e;
30	border-radius: 8px;
31	box-shadow: 0 4px 6px rgba(0, 0, 0, 1);
32	position: relative;
33	color: #ffffff;
34	text-align: center;
35	}
36	
37	.form-signin img {
38	margin-bottom: 20px;
39	}
40	
41	.form-signin h1 {
42	margin-bottom: 10px;

43	font-size: 24px;
44	font-weight: normal;
45	color: #ffffff;
46	}
47	
48	.form-signin p {
49	margin-bottom: 20px;
50	font-size: 14px;
51	color: #b3b3b3;
52	}
53	
54	.form-signin input[type="text"],
55	.form-signin input[type="password"] {
56	width: 100%;
57	padding: 10px;
58	margin-bottom: 10px;
59	border: 1px solid #333;
60	border-radius: 4px;
61	box-sizing: border-box;
62	background-color: #333;
63	color: #ffffff;
64	}
65	
66	.form-signin button {
67	width: 100%;
68	padding: 10px;
69	background-color: #6200ea;
70	color: white;
71	border: none;
72	border-radius: 4px;
73	cursor: pointer;
74	margin-bottom: 10px;
75	transition: background-color 0.3s ease;
76	}
77	
78	.form-signin button:hover {
79	background-color: #3700b3;
80	}
81	
82	.btn-register {
83	display: block;
84	width: 100%;

85	padding: 10px;
86	font-size: 14px;
87	color: white;
88	cursor: pointer;
89	background: #03dac6;
90	border: none;
91	border-radius: 4px;
92	margin-top: 10px;
93	transition: background-color 0.3s ease;
94	}
95	
96	.btn-register:hover {
97	background-color: #018786;
98	}
99	
100	.form-signin .alert {
101	padding: 10px;
102	margin-bottom: 10px;
103	border-radius: 4px;
104	color: #856404;
105	background-color: rgba(255, 253, 220, 0.1);
106	border: 1px solid #ffeeba;
107	display: none;
108	position: relative;
109	}
110	
111	.form-signin .text-muted {
112	color: #6c757d;
113	margin-top: 20px;
114	}
115	</style>
116	</head>
117	
118	<body>
119	
120	<main class="form-signin">
121	<!-- Alert untuk informasi saat halaman dimuat -->
122	<div class="alert" id="initial-alert" style="display: block;">
123	Login terlebih dahulu!
124	</div>
125	
126	<!-- Alert untuk kesalahan login -->

127	<div class="alert" style="display:	<?php echo (session()-
128	>getFlashdata('error')) ? 'block' : 'none'; ?>;">	
129	<?= session()->getFlashdata('error'); ?>	
130	</div>	
131		
132	<form method="post" action="<?= base_url(); ?>/login/process">	
133	<?= csrf_field(); ?>	
134	<h1 class="h3 mb-3 fw-normal">Login</h1>	
135	<p>Register dulu apabila belum memiliki akun</p>	
136		
137	<input type="text" name="username" id="username"	
138	placeholder="Username" required autofocus>	
139	<input type="password" name="password" id="password"	
	placeholder="Password" required>	
	<button type="submit">Login</button>	
	<button type="button"	class="btn-register"
	onclick="window.location.href='<?php echo	base_url('register');
	?>'>Register</button>	
	</form>	
	</main>	
	</body>	
	</html>	

Tabel 43. Source Code vw_login.php Soal 1 Modul 6

1	<!doctype html>
2	<html lang="en" class="h-100">
3	
4	<head>
5	<meta charset="utf-8">
6	<meta name="viewport" content="width=device-width, initial-scale=1">
7	<title>Register</title>
8	
9	<!-- Custom styles for this template -->
10	<style>
11	body,
12	html {
13	margin: 0;
14	padding: 0;
15	font-family: Arial, sans-serif;
16	height: 100%;
17	display: flex;
18	flex-direction: column;
19	background-color: black;
20	background-repeat: no-repeat;
21	background-attachment: fixed;
22	background-position: center;
23	background-size: cover;
24	background-color: #121212;
25	padding: 20px; /* Add padding to body */
26	color: #ffffff; /* Default text color */
27	}
28	
29	.container {
30	max-width: 600px;
31	margin: auto;
32	padding: 20px;
33	background: #1e1e1e;
34	border-radius: 8px;
35	box-shadow: 0 4px 6px rgba(0, 0, 0, 0.1);
36	margin-top: 40px;
37	color: #ffffff;
38	}
39	
40	h1 {
41	font-size: 24px;
42	margin-bottom: 20px;

43	text-align: center;
44	color: #ffffff;
45	}
46	
47	.form-label {
48	display: block;
49	margin-bottom: 5px;
50	font-weight: bold;
51	color: #ffffff;
52	}
53	
54	.form-control {
55	width: 100%;
56	padding: 10px;
57	margin-bottom: 10px;
58	border: 1px solid #333;
59	border-radius: 4px;
60	box-sizing: border-box; /* Ensure padding is included in width
61	calculation */
62	background-color: #333;
63	color: #ffffff;
64	}
65	
66	.btn {
67	width: 100%;
68	padding: 10px;
69	background-color: #6200ea;
70	color: white;
71	border: none;
72	border-radius: 4px;
73	cursor: pointer;
74	transition: background-color 0.3s ease;
75	}
76	
77	.btn:hover {
78	background-color: #3700b3;
79	}
80	
81	.alert {
82	padding: 10px;
83	margin-bottom: 10px;
84	border-radius: 4px;

```

85         color: #856404;
86         background-color: rgba(255, 253, 220, 0.1);
87         border: 1px solid #ffeeba;
88     }
89
90     .footer {
91         margin-top: auto;
92         padding: 10px 0;
93         text-align: center;
94         background-color: #1e1e1e;
95         color: #ffffff;
96     }
97
98     .footer .container {
99         max-width: none;
100     }
101 </style>
102 </head>
103
104 <body>
105
106     <main class="flex-shrink-0">
107         <div class="container">
108             <h1>Register Form</h1>
109             <hr />
110             <?php if (!empty(session()->getFlashdata('error'))): ?>
111                 <div class="alert">
112                     <h4>Periksa Entrian Form</h4>
113                     <?= session()->getFlashdata('error'); ?>
114                 </div>
115             <?php endif; ?>
116             <form method="post" action="<?= base_url(); ?>/register/process">
117                 <?= csrf_field(); ?>
118                 <div class="mb-3">
119                     <label for="username" class="form-label">Username</label>
120                     <input type="text" class="form-control" id="username"
121 name="username" required>
122                 </div>
123                 <div class="mb-3">
124                     <label for="password" class="form-label">Password</label>
125                     <input type="password" class="form-control" id="password"
126 name="password" required>

```

127	</div>
128	<div class="mb-3">
129	<label for="password_conf" class="form-label">Confirm
130	Password</label>
131	<input type="password" class="form-control"
132	id="password_conf" name="password_conf" required>
133	</div>
134	<div class="mb-3">
135	<label for="email" class="form-label">Email</label>
136	<input type="email" class="form-control" id="email"
137	name="email" required>
	</div>
	<div class="mb-3">
	<button type="submit" class="btn">Register</button>
	</div>
	</form>
	<hr />
	</div>
	</main>
	<footer class="footer">
	<div class="container">
	© 2023 Your Company
	</div>
	</footer>
	</body>
	</html>

Tabel 44. Source Code vw_register.php Soal 1 Modul 6

```

1  <?php
2
3  namespace App\Models;
4
5  use CodeIgniter\Model;
6
7  class BukuModel extends Model
8  {
9      protected $table = 'buku';
10     protected $allowedFields = ['judul', 'penulis', 'penerbit', 'tahun_terbit'];
11     protected $createdField = 'created_at';
12     protected $updatedField = 'updated_at';
13     protected $deletedField = 'deleted_at';
14
15     protected $validationRules = ([
16         'judul' => [
17             'rules' => 'required|string',
18             'errors' => [
19                 'required' => 'Judul harus diisi.',
20                 'string' => 'Judul harus berupa string.'
21             ]
22         ],
23         'penulis' => [
24             'rules' => 'required|string',
25             'errors' => [
26                 'required' => 'Penulis harus diisi.',
27                 'string' => 'Penulis harus berupa string.'
28             ]
29         ],
30         'penerbit' => [
31             'rules' => 'required|string',
32             'errors' => [
33                 'required' => 'Penerbit harus diisi.',
34                 'string' => 'Penerbit harus berupa string.'
35             ]
36         ],
37         'tahun_terbit' => [
38             'rules' => 'required|integer|greater_than[1800]|less_than[2024]',
39             'errors' => [
40                 'required' => 'Tahun terbit harus diisi.',
41                 'integer' => 'Tahun terbit harus berupa angka.',
42                 'greater_than' => 'Tahun terbit harus lebih besar dari 1800.'

```

43	'less_than' => 'Tahun terbit harus lebih kecil dari 2024.'
44]
45],
46]);
47	
48	protected \$skipValidation = false;
49	
50	}

Tabel 45. Source Code Bukumodel.php Soal 1 Modul 6

1	<?php
2	
3	namespace App\Models;
4	
5	use CodeIgniter\Model;
6	
7	class UsersModel extends Model
8	{
9	protected \$table = "users";
10	protected \$returnType = "object";
11	protected \$useTimestamps = true;
12	protected \$allowedFields = ['username', 'password', 'email'];
13	}

Tabel 46. Source Code Usermodel.php Soal 1 Modul 6

1	<?php
2	
3	namespace App\Database\Seeds;
4	
5	use CodeIgniter\Database\Seeder;
6	
7	class Buku extends Seeder
8	{
9	public function run()
10	{
11	\$this->db->table('buku')->insert([
12	'judul' => 'Laut Bercerita',
13	'penulis' => 'Leila S. Chudori',
14	'penerbit' => 'Gramedia',
15	'tahun_terbit' => '2017'
16]);
17	
18	\$this->db->table('buku')->insert([
19	'judul' => 'Tentang Kamu',
20	'penulis' => 'Tere Liye',
21	'penerbit' => 'Republika',
22	'tahun_terbit' => '2016'
23]);
24	}
25	}

Tabel 47. Source Code Buku.php di Seeds Soal 1 Modul 6

1	<?php
2	
3	namespace App\Database\Seeds;
4	
5	use CodeIgniter\Database\Seeder;
6	
7	class UserSeeder extends Seeder
8	{
9	public function run()
10	{
11	\$this->db->table('users')->insert([
12	'username' => 'Farly',
13	'email' => 'farlyhaydy123@gmail.com',
14	'password' => password_hash('123', PASSWORD_DEFAULT),
15]);
16	}
17	}

1	-- phpMyAdmin SQL Dump
2	-- version 5.2.1
3	-- https://www.phpmyadmin.net/
4	--
5	-- Host: 127.0.0.1
6	-- Waktu pembuatan: 11 Jun 2024 pada 19.57
7	-- Versi server: 10.4.32-MariaDB
8	-- Versi PHP: 8.2.12
9	
10	SET SQL_MODE = "NO_AUTO_VALUE_ON_ZERO";
11	START TRANSACTION;
12	SET time_zone = "+00:00";
13	
14	/*!40101 SET
15	@OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
16	/*!40101 SET
17	@OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS
18	*/;
19	/*!40101 SET
20	@OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
21	/*!40101 SET NAMES utf8mb4 */;
22	
23	--
24	-- Database: `perpustakaan`
25	--
26	
27	-- -----
28	
29	--
30	-- Struktur dari tabel `buku`
31	--
32	
33	CREATE TABLE `buku` (
34	`id` bigint(20) UNSIGNED NOT NULL,
35	`judul` varchar(255) NOT NULL,
36	`penulis` varchar(255) NOT NULL,
37	`penerbit` varchar(255) NOT NULL,
38	`tahun_terbit` year(4) DEFAULT NULL
39) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4
40	COLLATE=utf8mb4_general_ci;
41	
42	--


```

43 -- Dumping data untuk tabel `buku`
44 --
45
46 INSERT INTO `buku` (`id`, `judul`, `penulis`, `penerbit`, `tahun_terbit`)
47 VALUES
48 (15, 'Laut Bercerita', 'Leila S. Chudori', 'Gramedia', '2017'),
49 (16, 'Tentang Kamu', 'Tere Liye', 'Republika', '2016');
50
51 -----
52
53 --
54 -- Struktur dari tabel `migrations`
55 --
56
57 CREATE TABLE `migrations` (
58   `id` bigint(20) UNSIGNED NOT NULL,
59   `version` varchar(255) NOT NULL,
60   `class` varchar(255) NOT NULL,
61   `group` varchar(255) NOT NULL,
62   `namespace` varchar(255) NOT NULL,
63   `time` int(11) NOT NULL,
64   `batch` int(11) UNSIGNED NOT NULL
65 ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4
66 COLLATE=utf8mb4_general_ci;
67
68 --
69 -- Dumping data untuk tabel `migrations`
70 --
71
72 INSERT INTO `migrations` (`id`, `version`, `class`, `group`, `namespace`, `time`,
73 `batch`) VALUES
74 (1, '2024-06-03-103310', 'App\\Database\\Migrations\\Users', 'default', 'App',
75 1718081782, 1),
76 (2, '2024-06-03-122804', 'App\\Database\\Migrations\\Buku', 'default', 'App',
77 1718081782, 1),
78 (4, '2024-06-11-173406', 'App\\Database\\Migrations\\Users', 'default', 'App',
79 1718127395, 3);
80
81 -----
82
83 --
84 -- Struktur dari tabel `users`

```

```

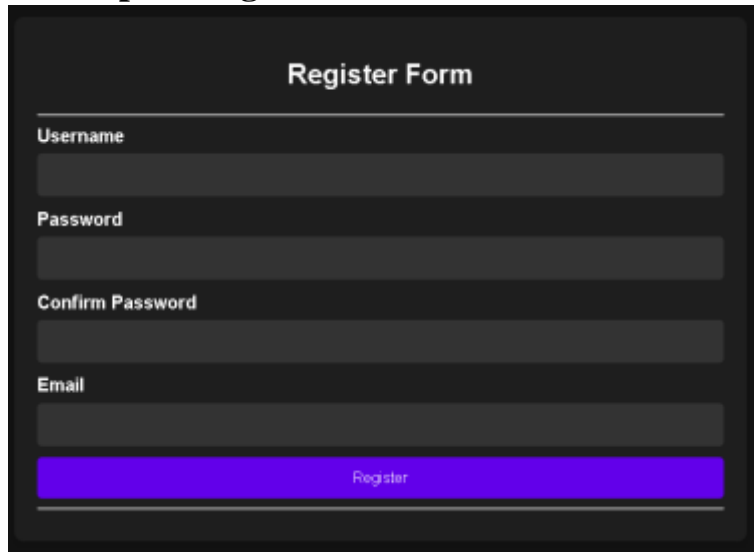
85 --
86
87 CREATE TABLE `users` (
88   `id` int(10) UNSIGNED NOT NULL,
89   `username` varchar(255) NOT NULL,
90   `email` varchar(255) NOT NULL,
91   `password` text NOT NULL,
92   `created_at` datetime DEFAULT NULL,
93   `updated_at` datetime DEFAULT NULL
94 ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4
95 COLLATE=utf8mb4_general_ci;
96
97 --
98 -- Dumping data untuk tabel `users`
99 --
100
101 INSERT INTO `users` (`id`, `username`, `email`, `password`, `created_at`,
102 `updated_at`) VALUES
103 (4, 'Farly', 'farlyhaydy123@gmail.com',
104 '$2y$10$xYhsftGqi132/1Y8Qrypge2VGyR2bS0TEBFpXTa4sXUKI4GwnEoh2',
105 NULL, NULL);
106
107 --
108 -- Indexes for dumped tables
109 --
110
111 --
112 -- Indeks untuk tabel `buku`
113 --
114 ALTER TABLE `buku`
115   ADD PRIMARY KEY (`id`);
116
117 --
118 -- Indeks untuk tabel `migrations`
119 --
120 ALTER TABLE `migrations`
121   ADD PRIMARY KEY (`id`);
122
123 --
124 -- Indeks untuk tabel `users`
125 --
126 ALTER TABLE `users`

```

127	ADD PRIMARY KEY (`id`),
128	ADD UNIQUE KEY `email` (`email`);
129	
130	--
131	-- AUTO_INCREMENT untuk tabel yang dibuang
132	--
133	
134	--
135	-- AUTO_INCREMENT untuk tabel `buku`
136	--
137	ALTER TABLE `buku`
138	MODIFY `id` bigint(20) UNSIGNED NOT NULL AUTO_INCREMENT,
139	AUTO_INCREMENT=19;
140	
141	--
	-- AUTO_INCREMENT untuk tabel `migrations`
	--
	ALTER TABLE `migrations`
	MODIFY `id` bigint(20) UNSIGNED NOT NULL AUTO_INCREMENT,
	AUTO_INCREMENT=5;
	--
	-- AUTO_INCREMENT untuk tabel `users`
	--
	ALTER TABLE `users`
	MODIFY `id` int(10) UNSIGNED NOT NULL AUTO_INCREMENT,
	AUTO_INCREMENT=5;
	COMMIT;
	/*!40101 SET
	CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;
	/*!40101 SET
	CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;
	/*!40101 SET
	COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;

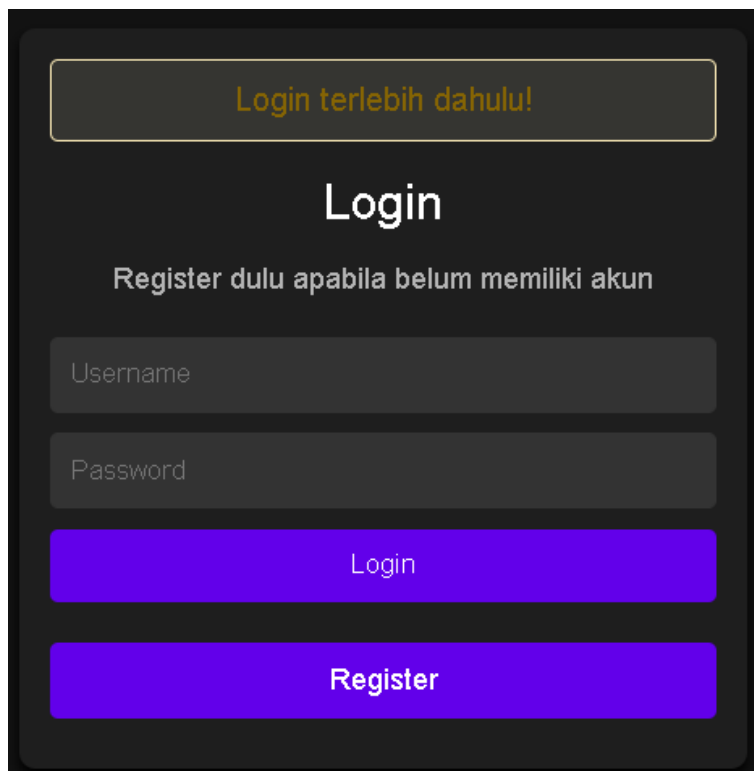
Tabel 48. Source Code Perpustakaan.sql Soal 1 Modul 6

B. Output Program



A screenshot of a 'Register Form' with a dark background. The form contains four input fields: 'Username', 'Password', 'Confirm Password', and 'Email'. Each field is represented by a dark gray rectangular box with a light gray border. Below the 'Email' field is a red button with the text 'Register' in white. The title 'Register Form' is centered at the top of the form area.

Gambar 58. Screenshot Hasil Akhir Soal 1 Modul 6



A screenshot of a 'Login' page with a dark background. At the top, there is a red rectangular box with the text 'Login terlebih dahulu!' in white. Below this, the word 'Login' is centered in a large white font. Underneath, the text 'Register dulu apabila belum memiliki akun' is displayed in a smaller white font. There are two input fields: 'Username' and 'Password', each with a dark gray box and a light gray border. At the bottom, there are two red buttons: 'Login' and 'Register', both with white text.

Gambar 59. Screenshot Hasil Akhir Soal 1 Modul 6

List Buku					Tambah Data
Judul	Penulis	Penerbit	Tahun Terbit	Action	
Tentang Kamu	Tere Liye	Republika	2016	Edit	Hapus
1					

Gambar 60. Screenshot Hasil Akhir Soal 1 Modul 6

Tambah	
Judul	<input type="text" value="Masukkan Judul"/>
Penulis	<input type="text" value="Masukkan Penulis"/>
Penerbit	<input type="text" value="Masukkan Penerbit"/>
Tahun Terbit	<input type="text" value="Masukkan Tahun Terbit"/>
<input type="button" value="Simpan"/> <input type="button" value="Kembali"/>	

Gambar 61. Screenshot Hasil Akhir Soal 1 Modul 6

Edit	
Judul	<input type="text" value="Tentang Kamu"/>
Penulis	<input type="text" value="Tere Liye"/>
Penerbit	<input type="text" value="Republika"/>
Tahun Terbit	<input type="text" value="2016"/>
<input type="button" value="Update"/> <input type="button" value="Kembali"/>	

Gambar 62. Screenshot Hasil Akhir Soal 1 Modul 6

C. Pembahasan

- Routes.php

Kode tersebut ditulis menggunakan framework CodeIgniter untuk mengatur routing aplikasi web. Mari kita bahas fungsinya secara umum:

1. Auto Routing (`$routes->setAutoRoute(true);`)

Baris ini secara opsional mengaktifkan auto routing. Artinya, CodeIgniter akan mencoba mengenali controller dan method berdasarkan struktur URL. Namun, pada kode ini terlihat pengaturan manual dilakukan pada baris selanjutnya, jadi fungsinya kemungkinan dinonaktifkan.

2. Route Definisi

- `$routes->get('/', 'Buku::login');` o Ini mengatur routing untuk path /. Ketika pengguna mengakses halaman utama (/), aplikasi akan memanggil controller Buku dengan method login.
- `$routes->get('home', 'Buku::index');`
 - o Routing untuk path /home. Controller Buku dengan method index akan dijalankan. Biasanya method index digunakan untuk menampilkan halaman utama setelah login.
- `$routes->get('/register', 'Register::index');`
 - o Routing untuk path /register. Controller Register dengan method index akan dijalankan.
- Baris selanjutnya mengatur routing terkait proses registrasi dan login dengan controller Register dan Login.
- `$routes->get('/logout', 'Login::logout');`
 - o Routing untuk path /logout. Controller Login dengan method logout akan dijalankan untuk proses keluar (logout).

3. Routing Buku • `$routes->get('/buku/create', 'Buku::create');`

- o Routing untuk path /buku/create. Controller Buku dengan method create akan dijalankan, kemungkinan untuk menampilkan form pembuatan buku baru.

- **`$routes->post('buku/store', 'Buku::store');`**
 - Routing untuk method POST ke path `/buku/store`. Controller Buku dengan method `store` kemungkinan digunakan untuk memproses data submit form pembuatan buku baru.
- **`$routes->post('buku/update/', 'Buku::update', ['filter' => 'auth']);`**
 - Routing untuk method POST ke path dengan format `/buku/update/` diikuti angka (ID buku). Method `update` pada controller Buku akan dijalankan. Perhatikan bahwa route ini dilindungi dengan filter `auth`, artinya pengguna harus login terlebih dahulu untuk mengaksesnya.
- **`$routes->get('buku/edit/', 'Buku::edit', ['filter' => 'auth']);`**
 - Routing untuk path dengan format `/buku/edit/` diikuti angka (ID buku). Method `edit` pada controller Buku akan dijalankan untuk menampilkan form edit buku. Route ini juga dilindungi dengan filter `auth`.
- **`$routes->get('buku/delete/', 'Buku::delete', ['filter' => 'auth']);`**
 - Routing untuk path dengan format `/buku/delete/` diikuti angka (ID buku). Method `delete` pada controller Buku akan dijalankan untuk menghapus buku. Route ini juga dilindungi dengan filter `auth`.

- **Buku.php di controller**

Kode tersebut merupakan class controller bernama Buku di aplikasi My Library yang ditulis dengan framework CodeIgniter. Class ini menangani operasi terkait manajemen buku. Mari kita bahas fungsinya secara umum

1. Struktur Class

- Class Buku merupakan child class dari class Controller bawaan CodeIgniter.
- Properti `$model` dideklarasikan untuk menampung objek model yang digunakan untuk interaksi dengan database.
- Konstruktor (`__construct`) menginisialisasi objek model `BukuModel` dan helper yang dibutuhkan (form dan url).

2. Method-method

- **login():** Method ini menampilkan halaman login. Sepertinya ada kesalahan pemanggilan model BukuModel untuk menampilkan daftar post (mungkin seharusnya menggunakan model lain).
- **index():** Method ini menampilkan halaman daftar buku. Method ini melakukan pengecekan apakah user sudah login (menggunakan session), jika belum akan di redirect ke halaman login. Kemudian method ini mengambil data buku dengan pagination (10 data per halaman) dan mengirimkannya ke view vw_buku bersama dengan judul halaman.
- **create():** Method ini menampilkan halaman create buku baru. Method ini juga melakukan pengecekan apakah user sudah login, jika belum akan di redirect ke halaman login. Terakhir, method ini mengirimkan judul halaman ke view vw_add.
- **store():** Method ini menangani proses penyimpanan data buku baru. Method ini mengambil data buku yang di submit dari form dan melakukan validasi data. Jika validasi gagal, user akan dikembalikan ke halaman create(). Jika validasi sukses, data buku akan disimpan ke database menggunakan model BukuModel. Setelah berhasil disimpan, method ini menampilkan flashdata sukses dan redirect ke halaman home.
- **edit(\$id):** Method ini menampilkan halaman edit buku. Method ini mengambil data buku berdasarkan id dari database dan melakukan pengecekan apakah data buku ditemukan. Jika tidak ditemukan, method ini menampilkan flashdata error dan redirect ke halaman sebelumnya. Jika data ditemukan, method ini mengirimkan data buku dan judul halaman ke view vw_edit.
- **update(\$id):** Method ini menangani proses update data buku. Method ini mengambil data buku yang di submit dari form dan melakukan validasi data. Jika validasi gagal, user akan dikembalikan ke halaman edit. Jika validasi sukses, data buku akan di update di database menggunakan model BukuModel. Method ini kemudian mengecek apakah update berhasil dan menampilkan flashdata sesuai hasil update (berhasil/gagal) dan redirect ke halaman home.
- **delete(\$id):** Method ini menangani proses delete buku. Method ini melakukan pengecekan apakah user sudah login dan id buku tersedia. Jika tidak valid, method ini akan redirect ke halaman home. Selanjutnya, method ini menghapus data buku berdasarkan id dari database menggunakan model BukuModel. Terakhir, method ini mengecek apakah delete berhasil dan menampilkan flashdata sesuai hasil delete (berhasil/gagal) dan redirect ke halaman home.

- **Home.php**

Kode tersebut menunjukkan controller Home yang memperluas class BaseController dari framework CodeIgniter.

Fungsi Utama:

- Controller ini memiliki satu method publik yaitu index().
- Method index() tidak memiliki parameter dan mengembalikan string.
- Di dalam method index(), terdapat baris return view('vw_buku');.
 - Baris ini menunjukkan bahwa controller ini akan menampilkan view dengan nama vw_buku.

- **Login.php**

Kode tersebut merupakan class controller bernama Login di aplikasi MyLibrary yang ditulis dengan framework CodeIgniter. Class ini menangani proses login dan logout user.

1. Struktur Class

- Class Login merupakan child class dari class BaseController bawaan CodeIgniter.

2. Method-method

- **index():** Method ini menampilkan halaman login. User akan diarahkan ke view vw_login.
- **process():** Method ini menangani proses login user.
 - Method ini pertama-tama membuat objek model UsersModel untuk interaksi dengan database.
 - Kemudian, method ini mengambil username dan password yang di submit dari form login.
 - Selanjutnya, method ini mencari data user berdasarkan username dari database.
 - Jika data user ditemukan:
 - ✦ Method ini memverifikasi password yang di submit dengan password yang tersimpan di database menggunakan fungsi password_verify.

- ✦ Jika password valid, method ini menyimpan username dan flag `logged_in` ke dalam session.
- ✦ Terakhir, user akan di redirect ke halaman home.
- Jika data user tidak ditemukan atau password salah, method ini akan menampilkanflashdata error "Username & Password Salah" dan redirect user kembali ke halaman login.
- **logout():** Method ini menangani proses logout user.
 - Method ini memanggil fungsi `session()->destroy()` untuk menghapus semua data session yang terkait dengan user.
 - Terakhir, user akan di redirect ke halaman login.

- Register.php

Kode tersebut ditulis menggunakan framework CodeIgniter untuk menangani proses registrasi pengguna. Mari kita bahas fungsinya secara umum:

1. Method Index (Form Registrasi)

- Method index kemungkinan digunakan untuk menampilkan halaman registrasi.
- Method ini mengembalikan view `vw_register`.

2. Method Process (Registrasi Akun Baru)

- Method process menangani proses submit form registrasi.
- Diawali dengan validasi form input menggunakan library validate.
 - Username:
 - ✦ Diwajibkan diisi (required).
 - ✦ Minimal 4 karakter (`min_length[4]`).
 - ✦ Maksimal 20 karakter (`max_length[20]`).
 - ✦ Harus unik (`is_unique[users.username]`), tidak boleh ada user lain dengan username yang sama.
 - Password:
 - ✦ Diwajibkan diisi (required).
 - ✦ Minimal 0 karakter (`min_length[0]`).

- ✦ Maksimal 50 karakter (max_length[50]).
- Konfirmasi Password:
 - ✦ Harus sama dengan password (matches[password]).
- Email:
 - ✦ Diwajibkan diisi (required).
 - ✦ Minimal 4 karakter (min_length[4]).
 - ✦ Maksimal 100 karakter (max_length[100]).
 - ✦ Harus berupa email yang valid (valid_email).
- Jika validasi gagal:
 - Flashdata session untuk menampilkan pesan error berisi daftar error validasi dibuat dengan \$this->validator->listErrors().
 - Pengguna dialihkan kembali ke halaman registrasi (redirect()->back()) beserta data form yang sudah diisi sebelumnya (withInput()).
- Jika validasi berhasil:
 - Model UsersModel diinisiasi untuk interaksi dengan database.
 - Password yang diinputkan diubah menjadi hash menggunakan function password_hash sebelum disimpan ke database. Teknik hashing disarankan untuk keamanan password.
 - Data user baru (username, password hash, dan email) disimpan ke database menggunakan method insert pada model UsersModel.
- Setelah registrasi berhasil, pengguna akan dialihkan ke halaman login (redirect()->to('/login')).

- AuthFilter.php

Kode tersebut ditulis menggunakan framework CodeIgniter untuk membuat filter autentikasi pengguna. Mari kita bahas fungsinya secara umum:

1. Class dan Interface

- Class ini bernama AuthFilter dan mengimplementasi interface FilterInterface dari CodeIgniter.

2. Method before

- Method before dieksekusi sebelum request controller dijalankan.
- Di dalam method ini dilakukan pengecekan apakah user sudah login.
- Pengecekan dilakukan dengan memeriksa keberadaan data user_id di session.
- Jika user_id tidak ditemukan di session (artinya user belum login):
 - Filter akan menghentikan proses controller dan mengalihkan pengguna ke halaman login (redirect()->to('/login')).

3. Method after

- Method after dieksekusi setelah request controller dijalankan dan menghasilkan response.
- Namun pada kode ini, method after terlihat kosong. Artinya filter tidak melakukan tindakan apapun setelah proses controller selesai.

- vw_add.php

Kode tersebut tampaknya merupakan bagian dari halaman web untuk menambah data buku baru. Mari kita bahas komponennya secara umum:

1. Layout dan Konten

- File ini menggunakan layout layouts/main yang kemungkinan berisi struktur dasar halaman web seperti header, footer, dan navigasi.
- Konten utama ditempatkan di dalam section content.

2. Styling

- Terdapat style CSS di bagian awal untuk mengatur background image pada seluruh halaman.

3. Form Tambah Buku

- Terdapat formulir untuk menambahkan data buku baru.
- Formulir ini menggunakan library form helper dari CodeIgniter.
- Formulir di submit ke URL buku/store menggunakan method form_open.

- Elemen formulir meliputi:
 - o Judul buku (text input).
 - o Penulis buku (text input).
 - o Penerbit buku (text input).
 - o Tahun terbit buku (number input).
 - o Tombol submit "Simpan" dan tombol kembali "Kembali" ke halaman home.

4. Pesan Error

- Kode ini mengecek keberadaanflashdata session dengan key "error".
- Jika ada, maka akan menampilkan pesan error di dalam alert berwarna bahaya (danger).

5. Validasi Error

- Kode ini menggunakan function `validation_list_errors` untuk menampilkan pesan error validasi jika ada.

6. Summernote Text Editor

- Bagian akhir kode memuat script untuk menyertakan library Summernote.
- Summernote digunakan sebagai text editor WYSIWYG untuk mengisi konten buku.
- Script ini menginisialisasi Summernote pada element dengan ID `post_content` dan mengatur beberapa konfigurasi seperti ukuran `の高さ` (takasa - height) menjadi 500px.

- vw_buku.php

Struktur dan Konten HTML

Extend dan Section: Menggunakan `<?= $this->extend('layouts/main') ?>` dan `<?= $this->section('content') ?>` untuk memanfaatkan layout utama dan menetapkan bagian konten pada layout tersebut.

Navbar: Menampilkan navbar dengan logo dan menu navigasi. Jika pengguna sudah login (`session()->get('logged_in')`), maka menampilkan nama pengguna dan opsi logout. Jika belum login, menampilkan opsi login dan register.

CSS Styling

Body dan Komponen: Mengatur latar belakang, warna teks, dan beberapa komponen seperti navbar (navbar-light) dan tombol (btn, btn-sm).

PHP Integration

Flashdata: Menampilkan pesan success atau error menggunakan session()->getFlashdata('success') dan session()->getFlashdata('error').

Data Buku: Menampilkan data buku dari variabel \$posts dalam tabel dengan judul, penulis, penerbit, tahun terbit, dan opsi edit serta delete. Menggunakan foreach untuk iterasi data.

JavaScript Section

Pagination Styling: Menggunakan JavaScript untuk menambahkan kelas Bootstrap pada elemen pagination agar tampilan sesuai dengan tema (.pagination li dan .pagination li a).

Interaksi dengan Pengguna

Edit dan Delete: Setiap baris pada tabel memiliki tombol Edit dan Delete yang mengarahkan ke halaman edit buku atau menghapus data buku. Tombol Delete memiliki konfirmasi sebelum mengirimkan form.

- **vw_edit.php**

Struktur HTML dan PHP

Extend dan Section: Menggunakan <?= \$this->extend('layouts/main') ?> dan <?= \$this->section('content') ?> untuk memperluas dan menetapkan bagian konten pada layout utama.

Card Container: Menggunakan Bootstrap untuk merancang tata letak yang responsif. Card ini digunakan untuk mengelompokkan elemen-elemen terkait dengan edit form.

Form Edit: Menggunakan form_open('buku/update/' . \$post['id']) untuk membuka form yang mengarah ke endpoint update buku dengan method PUT. Input-input seperti judul, penulis, penerbit, dan tahun terbit diisi dengan nilai yang sudah ada (value="<?= \$post['judul'] ?>") untuk menampilkan nilai yang sudah ada sebelumnya.

Flashdata dan Validasi: Menggunakan session()->getFlashdata('error') untuk menampilkan pesan error jika ada, dan validation_list_errors() untuk menampilkan daftar error validasi jika ada.

Styling dengan CSS

Body dan Komponen: Mengatur latar belakang, warna teks, dan tata letak menggunakan CSS untuk memastikan tampilan yang sesuai dengan tema aplikasi perpustakaan.

JavaScript untuk WYSIWYG Editor

Summernote Editor: Menggunakan Summernote untuk menyediakan editor teks yang kaya fitur. Ini termasuk mengatur ukuran tab, toolbar dengan pilihan gaya teks, penambahan gambar, dan fitur-fitur lainnya untuk memudahkan pengguna dalam mengedit konten.

Interaksi dengan Pengguna

Button Actions: Tombol "Update" digunakan untuk mengirimkan form yang telah diisi untuk menyimpan perubahan data buku. Tombol "Kembali" mengarahkan pengguna kembali ke halaman utama setelah melakukan proses edit.

- **vw_login.php**

HTML Structure

DOCTYPE dan Head Section: Mendefinisikan DOCTYPE HTML5 dan bagian `<head>` yang berisi meta tag untuk pengaturan karakter set, viewport, dan judul halaman.

Body: Bagian utama dari halaman yang berisi semua konten yang akan ditampilkan kepada pengguna. Diatur untuk berada di tengah-tengah layar dengan background berupa gambar yang diatur agar rapi dan sesuai dengan tema.

CSS Styling

Body Styling: Menggunakan CSS untuk mengatur tata letak body dengan `display: flex`, `justify-content: center`, dan `align-items: center` untuk memposisikan elemen-elemen secara tengah-tengah di layar.

Form .form-signin: Mengatur form login yang dibungkus dalam sebuah card dengan latar belakang hitam dan `border-radius` untuk membuat sudut melengkung. Shadow dan warna teks putih untuk menonjolkan konten form.

Alerts .alert: Dua bagian alert yang ditampilkan secara kondisional:

#initial-alert: Alert yang selalu ditampilkan saat halaman dimuat.

Alert untuk kesalahan login (`session()->getFlashdata('error')`), yang ditampilkan jika terdapat kesalahan saat login.

Form Inputs and Buttons: Mengatur input teks (username dan password) dengan background berwarna gelap dan border yang halus. Tombol login dengan warna latar belakang ungu tua (#6200ea) dan transisi warna saat hover. Tombol register dengan warna latar belakang hijau muda (#03dac6) yang berubah warna saat dihover.

Interaksi JavaScript

Register Button: Digunakan onclick untuk mengarahkan pengguna ke halaman register dengan `window.location.href`.

- **vw_register.php**

HTML Structure

DOCTYPE dan Head Section: Mendefinisikan DOCTYPE HTML5 dan bagian `<head>` yang berisi meta tag untuk pengaturan karakter set, viewport, dan judul halaman.

Body: Bagian utama dari halaman yang berisi semua konten yang akan ditampilkan kepada pengguna. Diatur untuk berada di tengah-tengah layar dengan background yang rapi dan sesuai dengan tema.

CSS Styling

Body Styling: Menggunakan CSS untuk mengatur tata letak body dengan `display: flex`, `flex-direction: column`, dan `height: 100%` untuk mengisi tinggi layar penuh. Background menggunakan gambar yang diatur agar rapi dan sesuai dengan tema aplikasi.

Container `.container`: Mengatur container untuk form registrasi dengan maksimum lebar 600px, padding, background berwarna gelap (#1e1e1e), dan border-radius untuk membuat sudut melengkung. Shadow dan warna teks putih untuk menonjolkan konten form.

Form Styling: Mengatur input teks (username, password, confirm password, email) dengan background berwarna gelap (#333), border yang halus, dan padding yang sesuai. Tombol "Register" dengan latar belakang ungu tua (#6200ea) dan transisi warna saat hover untuk efek visual yang lebih interaktif.

Alert Styling `.alert`: Bagian alert yang ditampilkan jika ada kesalahan dalam pengisian formulir. Memiliki padding, border-radius, dan warna latar belakang yang sesuai dengan tema aplikasi.

Footer .footer: Bagian footer dengan teks yang berada di bagian bawah halaman, menggunakan background dan warna teks yang konsisten dengan tema aplikasi.

Interaksi PHP

Flashdata Display: PHP digunakan untuk menampilkan pesan error jika terdapat kesalahan dalam pengisian form. Menggunakan `session()->getFlashdata('error')` untuk menampilkan pesan yang relevan di dalam alert.

- **BukuModel.php**

Kode tersebut mendefinisikan model `BukuModel` yang mewakili tabel buku di database. Model ini mempermudah interaksi dengan data buku di aplikasi. Mari kita bahas komponennya secara umum:

1. Class dan Namespace

- Kode diawali dengan deklarasi namespace `App\Models` yang menunjukkan lokasi class model.
- Deklarasi class `BukuModel extends Model` mendefinisikan class `BukuModel` yang mewarisi class `Model` dari `CodeIgniter`.

2. Atribut Model

- `protected $table = 'buku';` Menentukan nama tabel di database yang diwakili oleh model ini, yaitu buku.
- `protected $allowedFields = ['judul', 'penulis', 'penerbit', 'tahun_terbit'];` Menentukan daftar nama kolom yang diizinkan untuk diakses dan dimanipulasi oleh model.
- `protected $createdField = 'created_at';` Menentukan nama kolom yang menyimpan waktu pembuatan data.
- `protected $updatedField = 'updated_at';` Menentukan nama kolom yang menyimpan waktu pembaruan data.
- `protected $deletedField = 'deleted_at';` Menentukan nama kolom yang menyimpan waktu penghapusan data (jika soft delete diaktifkan).
- `protected $validationRules = [...];` Menentukan aturan validasi untuk masing-masing kolom.
 - Setiap kolom memiliki array aturan yang terdiri dari rules dan errors.

- rules berisi daftar aturan validasi yang dipisahkan dengan |.
 - errors berisi pesan error untuk setiap aturan validasi.
- protected \$skipValidation = false;; Menentukan apakah validasi data otomatis harus dilewati.

3. Aturan Validasi

Model ini mendefinisikan aturan validasi untuk kolom judul, penulis, penerbit, dan tahun_terbit:

- **Judul:**
 - Harus diisi (required).
 - Harus berupa string (string).
- **Penulis:**
 - Harus diisi (required).
 - Harus berupa string (string).
- **Penerbit:**
 - Harus diisi (required).
 - Harus berupa string (string).
- **Tahun Terbit:**
 - Harus diisi (required).
 - Harus berupa bilangan bulat (integer).
 - Harus lebih besar dari 1800 (greater_than[1800]).
 - Harus lebih kecil dari 2024 (less_than[2024]).

- UsersModel.php

Kode tersebut mendefinisikan model UsersModel yang mewakili tabel users di database. Model ini mempermudah interaksi dengan data pengguna di aplikasi. Mari kita bahas komponennya secara umum:

1. Class dan Namespace

- Kode diawali dengan deklarasi namespace App\Models yang menunjukkan lokasi class model.
- Deklarasi class UsersModel extends Model mendefinisikan class UsersModel yang mewarisi class Model dari CodeIgniter.

2. Atribut Model

- `protected $table = "users";`: Menentukan nama tabel di database yang diwakili oleh model ini, yaitu users.
- `protected $returnType = "object";`: Menentukan tipe data yang dikembalikan oleh metode model. Dalam hal ini, model akan mengembalikan objek.
- `protected $useTimestamps = true;`: Mengaktifkan penandaan waktu `created_at` dan `updated_at` secara otomatis untuk setiap data yang ditambahkan atau diperbarui.

protected \$allowedFields = ['username', 'password', 'email']; Menentukan daftar nama kolom yang diizinkan untuk diakses dan dimanipulasi oleh model.

- **Buku.php di Seeds**

Kode tersebut mendefinisikan class seeder bernama Buku yang digunakan untuk mengisi data awal ke dalam tabel buku di database. Class seeder ini mewarisi class Seeder dari CodeIgniter.

1. Struktur Class

- Deklarasi namespace App\Database\Seeds; menunjukkan lokasi class seeder.
- Deklarasi class Buku extends Seeder mendefinisikan class Buku yang mewarisi class Seeder.
- Method run() berisi kode yang akan dijalankan untuk mengisi data ke tabel.

2. Memasukkan Data ke Tabel Buku

- Method run() menggunakan method insert() dari class DB untuk memasukkan data ke tabel buku.
- Dua baris insert() digunakan untuk memasukkan dua data buku yang berbeda.
- Setiap baris insert() berisi array asosiatif yang memetakan nama kolom dengan nilai data yang ingin dimasukkan. o Kolom judul, penulis, penerbit, dan tahun_terbit diisi dengan data yang sesuai.

- **UserSeeder.php**

Kode tersebut mendefinisikan class seeder bernama UserSeeder yang digunakan untuk mengisi data awal ke dalam tabel users di database. Class seeder ini mewarisi class Seeder dari CodeIgniter.

1. Struktur Class

- Deklarasi namespace App\Database\Seeds; menunjukkan lokasi class seeder.
- Deklarasi class UserSeeder extends Seeder mendefinisikan class UserSeeder yang mewarisi class Seeder.

Method run() berisi kode yang akan dijalankan untuk mengisi data ke tabel.

2. Memasukkan Data ke Tabel Users

- Method run() menggunakan method insert() dari class DB untuk memasukkan data ke tabel users.
- Satu baris insert() digunakan untuk memasukkan satu data pengguna.
- Baris insert() berisi array asosiatif yang memetakan nama kolom dengan nilai data yang ingin dimasukkan:
 - Kolom username diisi dengan nilai "Farly".
 - Kolom email diisi dengan nilai "farlyhaydy123@gmail.com".
 - Kolom password diisi dengan nilai hash dari string "123" menggunakan fungsi password_hash(). Algoritma hash yang digunakan adalah PASSWORD_DEFAULT.

- perpustakaan.sql

File tersebut berisi kode untuk membuat struktur dan mengisi data awal ke dalam sebuah database perpustakaan. Mari kita bahas bagian-bagiannya:

1. Informasi Header

--- phpMyAdmin SQL Dump - Menunjukkan ini adalah file dump database yang dibuat oleh phpMyAdmin. --- Host: 127.0.0.1 - Alamat server database (localhost). --- Waktu pembuatan - Tanggal dan waktu pembuatan file dump. --- Versi server - Versi server database (MariaDB). --- Versi PHP - Versi interpreter PHP yang digunakan.

2. Perintah SQL

-SET SQL_MODE - Mengatur mode operasi untuk server database. -START TRANSACTION - Memulai transaksi database (perubahan data bersifat sementara sampai commit). -SET time_zone - Mengatur zona waktu database.

3. Dump Struktur Database perpustakaan

-CREATE TABLE - Perintah untuk membuat tabel-tabel di database perpustakaan. - Tabel buku menyimpan informasi buku (id, judul, penulis, penerbit, tahun_terbit). - Tabel migrations menyimpan history migrasi database (versi, class, group, namespace, time, batch). - Tabel users menyimpan informasi pengguna (id, username, email, password, created_at, updated_at).

4. Dump Data Tabel

-INSERT INTO - Perintah untuk memasukkan data awal ke dalam tabel. - Tabel buku berisi data awal 2 buah buku. - Tabel migrations berisi history migrasi yang pernah

dilakukan (3 record). - Tabel users berisi data awal 1 pengguna dengan username "Farly". Password disimpan dalam bentuk hash.

5. Indexes dan Auto Increment

-ALTER TABLE - Perintah untuk mengatur indeks dan auto increment pada tabel. - Setiap tabel memiliki primary key untuk identifikasi data. - Tabel users memiliki unique key pada kolom email untuk memastikan email unik. - Perintah AUTO_INCREMENT mengatur nilai awal dan penambahan otomatis untuk kolom id pada setiap tabel.

6. Commit Transaksi

-COMMIT - Menyimpan perubahan data yang dilakukan pada transaksi sebelumnya.

Tautan Git

https://github.com/Vainn11/Prak_Web_2/tree/b4af3f7197d0a76126fb49215269dcde20945c63/Modul%206

