

LATIHAN PRAKTIKUM PENGEMBANGAN APLIKASI WEB

BAB : PEMROGRAMAN BERORIENTASI OBJEK DAN POLA MVC
NAMA : DANI ADRIAN
NIM : 225150201111009
ASISTEN : - IBAR HUTTAQI SULTHON
 : - MUHAMMAD AMMAR
TANGGAL PRAKTIKUM : 25/04/2024

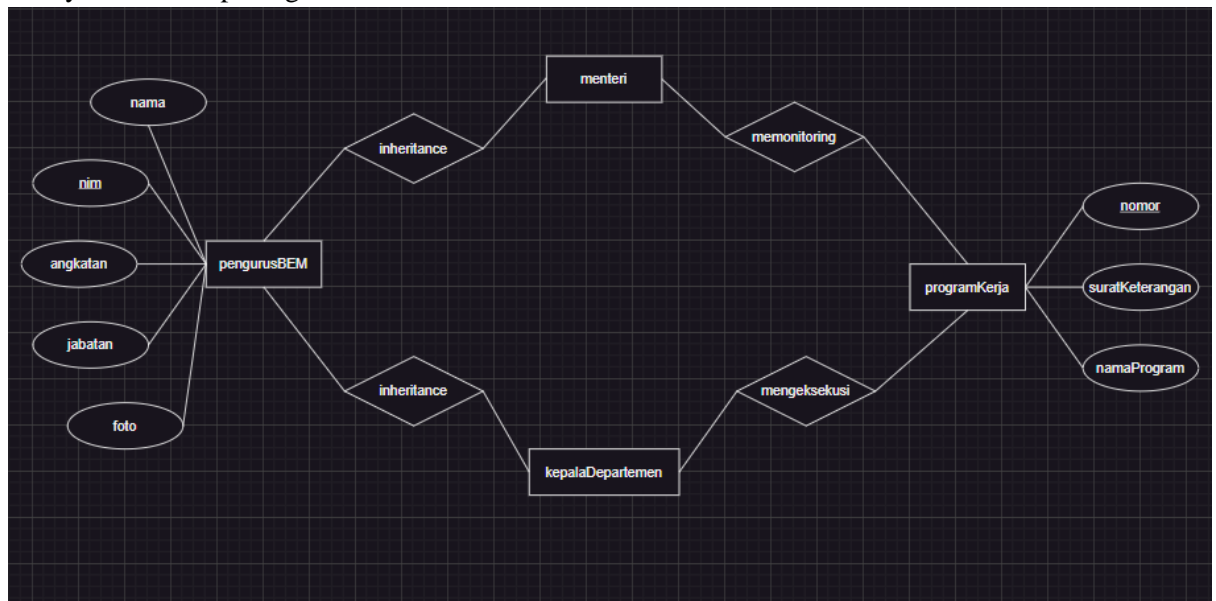
LATIHAN 1

A. Soal

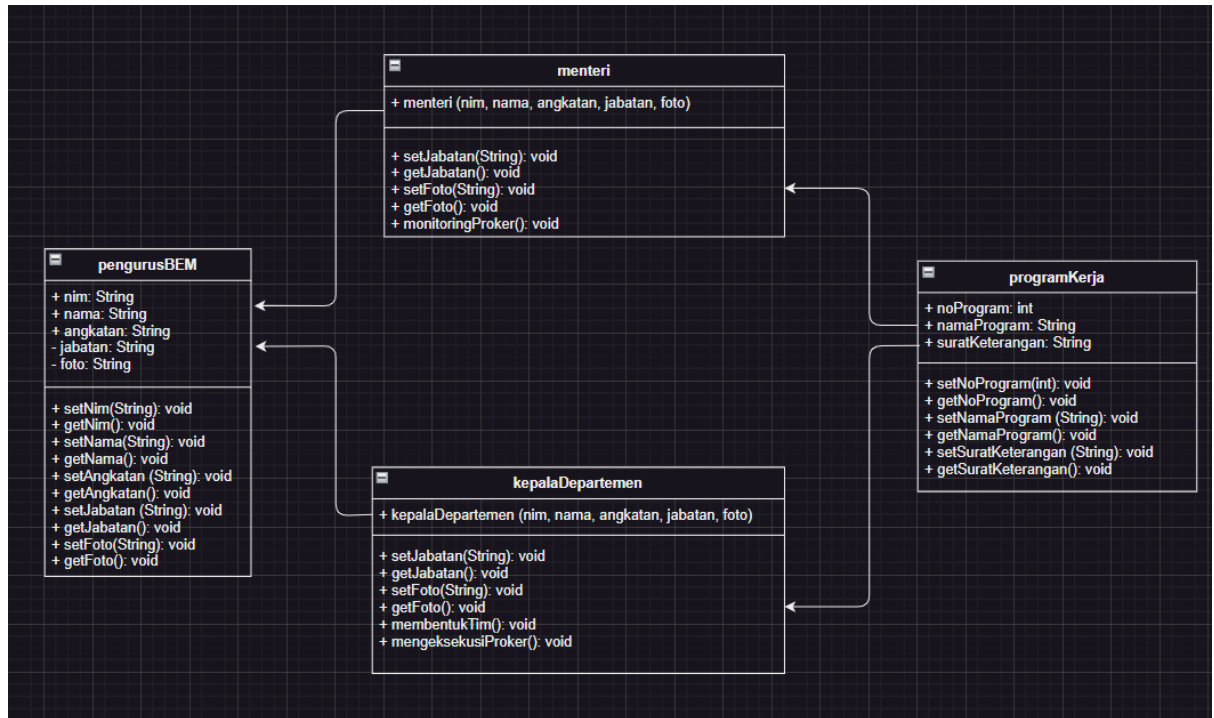
Buatlah pemodelan class diagram dan *entity relationship diagram* sesuai dengan analisis dan perancangan sistem yang sudah dibuat.

B. Screenshoot

Entity Relationship Diagram



Class Diagram



C. Syntax

D. Penjelasan

Berdasarkan deskripsi spesifikasi kebutuhan tersebut, dapat disusun sebuah Diagram Kelas dan Diagram Hubungan Entitas (ERD). Diagram Kelas mencakup empat kelas utama, yaitu pengurusBEM, kepalaDepartemen, menteri, dan programKerja, yang masing-masing memiliki atribut dan metode. Kelas pengurusBEM, misalnya, memiliki atribut seperti nama, nim, angkatan, jabatan, dan foto, dengan metode untuk mengakses dan menginisialisasi atribut tersebut. Kelas programKerja memiliki atribut nomor, suratKeterangan, dan namaProker. Kelas menteri dan kepalaDepartemen juga memiliki metode khusus.

Pada ERD, terdapat empat entitas, yaitu pengurusBEM, kepalaDepartemen, menteri, dan programKerja, yang masing-masing memiliki atribut. Entitas pengurusBEM, sebagai contoh, memiliki atribut seperti nama, nim, angkatan, jabatan, dan foto, dengan nim sebagai kunci primer. Entitas programKerja memiliki atribut nomor, suratKeterangan, dan namaProker, dengan nomor sebagai kunci primer. Sedangkan entitas menteri dan kepalaDepartemen tidak memiliki atribut sendiri karena merupakan turunan dari entitas pengurusBEM. Hubungan antara entitas tersebut adalah many-to-one.

LATIHAN 2

A. Soal

Buatlah folder baru dalam folder htdocs dan beri nama oop, seluruh kode Latihan 2 ini disimpan dalam folder oop

Membuat pemodelan struktur pengguna sistem dengan menuliskan kode di bawah ini dan simpan dalam file pengurusBEM.php

pengurusBEM.php	
1.	<?php
2.	class pengurusBEM
3.	{
4.	public \$nama;
5.	public \$nim;
6.	public \$angkatan;
7.	private \$jabatan;
8.	private \$foto;
9.	public function setName(\$nama)
10.	{
11.	\$this->nama = \$nama;
12.	}
13.	public function getName()
14.	{
15.	echo "\$this->nama";
16.	}
17.	private function setJabatan(\$jabatan)
18.	{
19.	\$this->jabatan = \$jabatan;
20.	}
21.	public function getJabatan()
22.	{
23.	echo "\$this->jabatan";
24.	}
25.	}

Tulis kode di bawah ini dan simpan dalam file menteri.php

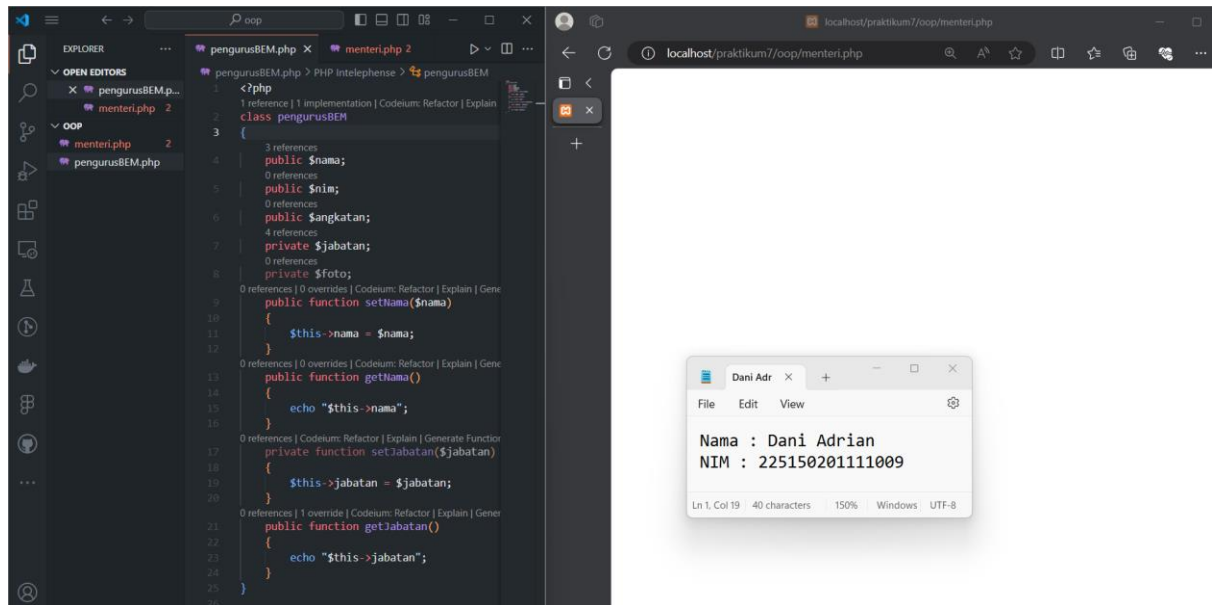
menteri.php	
1.	<?php
2.	include "pengurusBEM.php";
3.	
4.	class menteri extends pengurusBEM {
5.	public function __construct(\$nama) {
6.	\$this->nama = \$nama;
7.	}
8.	
9.	function setJabatan(\$jabatan) { // method
10.	\$this->jabatan = \$jabatan;
11.	}
12.	
13.	public function getJabatan() {
14.	echo "\$this->jabatan";
15.	}
16.	}

- Apa maksud dari penulisan public dan private pada class pengurusBEM?
- Modifikasilah kode diatas sehingga kita dapat melakukan manipulasi attribut dari pengurus BEM dengan lengkap.

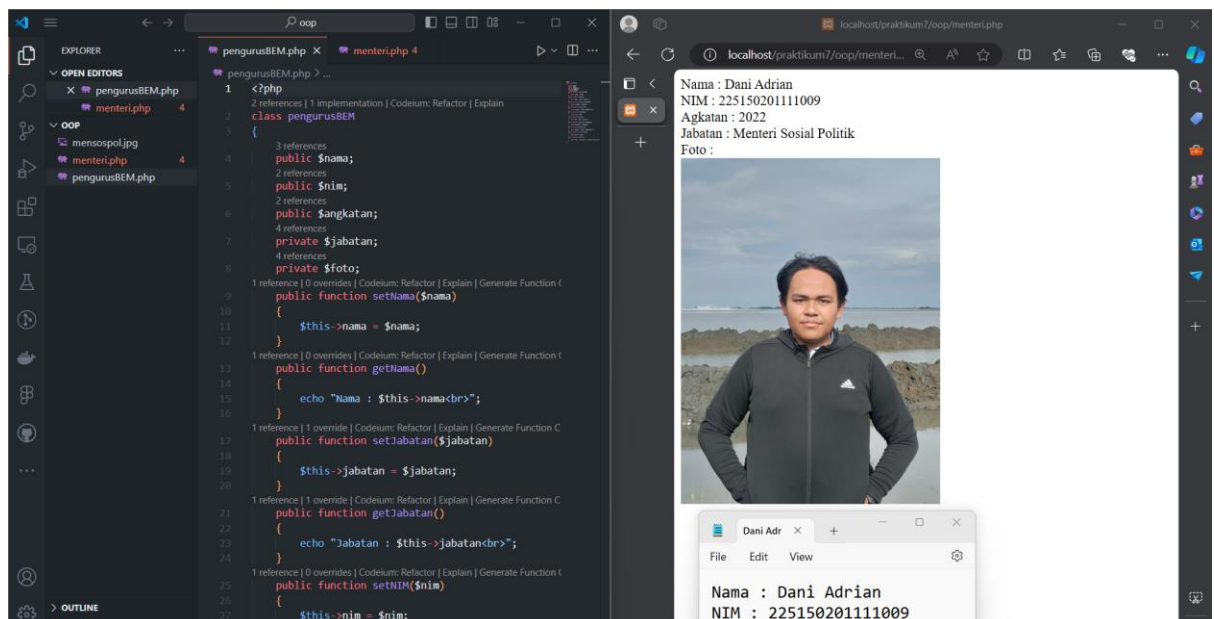
- c. Apa hubungan antara method `setJabatan` pada class `pengurusBEM` dan class `Menteri`? Apakah method yang sama karena konsep turunan?
- d. Apa yang terjadi saat dalam pembuatan class kita tidak menuliskan tingkat *visibility*? Apakah method atau attribute dengan semua tingkat *visibility* itu diturunkan?

B. Screenshoot

Sebelum Diubah



Setelah Diubah



C. Syntax

pengurusBEM.php	
1.	<?php
2.	class pengurusBEM
3.	{
4.	public \$nama;
5.	public \$nim;
6.	public \$angkatan;
7.	private \$jabatan;
8.	private \$foto;
9.	public function setName(\$nama)
10.	{
11.	\$this->nama = \$nama;
12.	}
13.	public function getName()
14.	{
15.	echo "Nama : \$this->nama ";
16.	}
17.	public function setJabatan(\$jabatan)
18.	{
19.	\$this->jabatan = \$jabatan;
20.	}
21.	public function getJabatan()
22.	{
23.	echo "Jabatan : \$this->jabatan ";
24.	}
25.	public function setNIM(\$nim)
26.	{
27.	\$this->nim = \$nim;
28.	}
29.	public function getNIM()
30.	{
31.	echo "NIM : \$this->nim ";
32.	}
33.	public function setAngkatan(\$angkatan)
34.	{
35.	\$this->angkatan = \$angkatan;
36.	}
37.	public function getAgkatan()
38.	{
39.	echo "Agkatan : \$this->angkatan ";
40.	}
41.	public function setFoto(\$foto)
42.	{
43.	\$this->foto = \$foto;
44.	}
45.	public function getFoto()
46.	{
47.	echo "Foto : foto
48.	width='300'>";
49.	}
	}

menteri.php	
1.	<?php
2.	include "pengurusBEM.php";
3.	class menteri extends pengurusBEM
4.	{
5.	public function __construct(\$nama)
6.	{
7.	\$this->nama = \$nama;
8.	}
9.	function setJabatan(\$jabatan)
10.	{ // method
11.	\$this->jabatan = \$jabatan;
12.	}
13.	public function getJabatan()
14.	{
15.	echo "\$this->jabatan";
16.	}
17.	function setFoto(\$foto)
18.	{ // method
19.	\$this->foto = \$foto;
20.	}
21.	public function getFoto()
22.	{
23.	echo "\$this->foto";
24.	}
25.	}
26.	\$mahasiswa = new pengurusBEM;
27.	\$mahasiswa->setNama("Dani Adrian");
28.	\$mahasiswa->getNama();
29.	\$mahasiswa->setNIM("225150201111009");
30.	\$mahasiswa->getNIM();
31.	\$mahasiswa->setAngkatan(2022);
32.	\$mahasiswa->getAgkatan();
33.	\$mahasiswa->setJabatan("Menteri Sosial Politik");
34.	\$mahasiswa->getJabatan();
35.	\$mahasiswa->setFoto("mensospol.jpg");
36.	\$mahasiswa->getFoto();

D. Penjelasan

Penulisan public dan private pada class pengurusBEM merupakan representasi dari konsep enkapsulasi objek, yang artinya adalah pengaturan hak akses pada class itu sendiri dan class lainnya. Terdapat tiga jenis hak akses: private, public, dan protected. Private membatasi akses hanya pada class tersebut, public memungkinkan akses dari semua class pada program, dan protected membolehkan akses dari class itu sendiri dan class turunannya.

File pengurusBEM.php berisi class pengurusBEM dengan atribut seperti nama, nim, angkatan, jabatan, dan foto, serta method setter dan getter. Setter digunakan untuk mengatur nilai atribut class, sementara getter mengambil nilai yang telah diatur. File menteri.php menggunakan include untuk memasukkan pengurusBEM.php dan extend untuk mewarisi sifat serta menghubungkan class menteri dengan pengurusBEM. Dalam menteri.php, terdapat nama dan jabatan menteri yang diset melalui constructor dan method overriding. Objek mahasiswa dibuat dan nilainya diatur serta dicetak.

Hubungan antara method `setJabatan` dalam class `pengurusBEM` dan `menteri` adalah bahwa `setJabatan` dalam `menteri` merupakan penggantian dari `setJabatan` dalam `pengurusBEM`. Ini dilakukan untuk memungkinkan akses ke `setJabatan` yang bersifat `private` oleh class lain, khususnya `menteri`.

Apabila pengguna tidak menentukan tingkat visibilitas, secara default akan menjadi `public`, sehingga class tersebut dapat diakses oleh class lain dalam paket yang sama. Penerusan (inheritance) method dan atribut hanya bisa dilakukan pada tingkat visibilitas `public` dan `protected`.

LATIHAN 3

A. Soal

Buatlah tabel proker dalam database untuk menyimpan data program kerja, dengan struktur sebagai berikut.

No	Nama Field	Tipe Data	Ukuran	Keterangan
1	nomorProgram	Integer	default	Primary key
2	namaProgram	Varchar	255	
3	suratKeterangan	Varchar	100	

Buatlah folder baru dalam folder `htdocs` dan beri nama `mvc`, seluruh kode Latihan 3 ini disimpan dalam folder `mvc`.

Tulis kode di bawah ini dan simpan dalam *file* `koneksiMVC.php`

koneksiMVC.php	
1.	<?php
2.	class koneksiMVC
3.	{
4.	public \$mysqli;
5.	public function __construct()
6.	{
7.	\$this->mysqli = mysqli_connect("localhost",
8.	"root", "", "praktikum12");
9.	}
	}

Tulis kode dibawah ini dan simpan dalam file “`m_programKerja.php`”

m_programKerja.php	
1.	<?php
2.	require "../koneksiMVC.php";
3.	class m_programKerja
4.	{
5.	private \$database;
6.	protected \$tablename = "program_kerja";
7.	public function __construct()
8.	{
9.	\$this->database = new koneksiMVC();

10.	\$this->database = \$this->database->mysqli;
11.	}
12.	// Add Program Kerja
13.	public function setProgramKerja(\$nomorProgram, \$namaProgram, \$suratKeterangan)
14.	{
15.	\$this->database->query("INSERT INTO \$this->tablename (nomorProgram, namaProgram, suratKeterangan) VALUES (\$nomorProgram, '\$namaProgram', '\$suratKeterangan')");
16.	}
17.	// Get All ProgramKerja
18.	public function getAll()
19.	{
20.	return \$this->database->query("SELECT * FROM \$this->tablename");
21.	}
22.	}

Tulis kode di bawah ini dan simpan dalam file “c_programKerja.php”

c_programKerja.php	
1.	<?php
2.	include_once("m_programKerja");
3.	
4.	class c_programKerja {
5.	public \$model;
6.	
7.	public function __construct(\$nama) {
8.	\$this->model = new m_programKerja();
9.	}
10.	
11.	public function invoke() {
12.	\$proker = \$this->model->getAll();
13.	include 'v_programKerja.php';
14.	}
15.	}

Tulis kode di bawah ini dan simpan dalam file “v_programKerja.php”

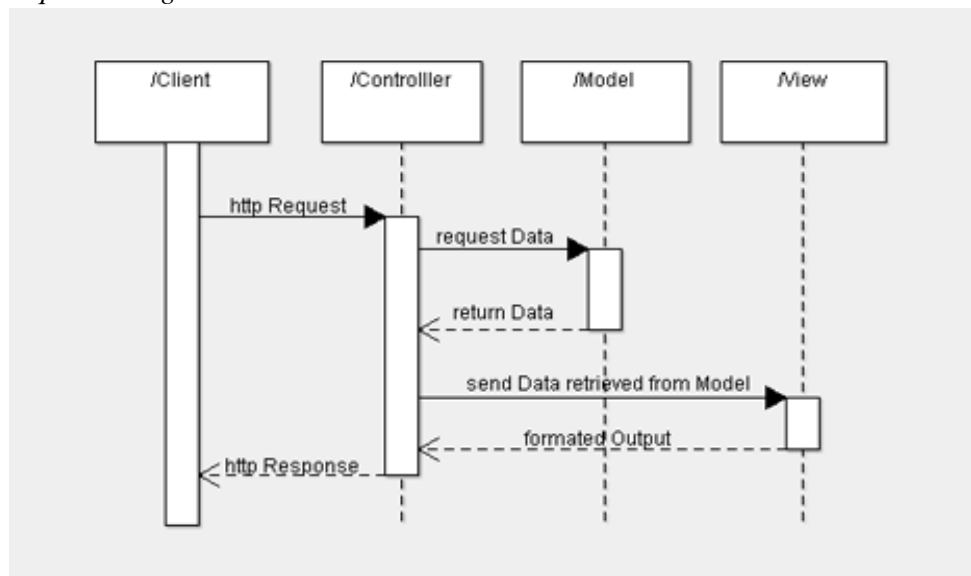
v_programKerja.php	
1.	<html>
2.	<head></head>
3.	<body>
4.	<h2>Daftar Program Kerja BEM</h2>
5.	<table>
6.	<tbody>
7.	<tr>
8.	<td>No</td>
9.	<td>Nama Program Kerja</td>
10.	<td>Surat Keterangan</td>
11.	</tr>
12.	</tbody>

13.	<?php
14.	
15.	echo '<table><tr>','
16.	'<td>\$proker[nomorProgram]</td>','
17.	'<td>\$proker[namaProgram]</td>','
18.	'<td>\$proker[suratKeterangan]</td>','
19.	'</tr></table>';
20.	?>
21.	</body>
22.	</html>

Tulis kode di bawah ini dan simpan dalam file “index.php”

index.php	
1.	<?php
2.	include_once("c_programKerja.php");
3.	
4.	\$controller = new c_programKerja();
5.	\$controller->invoke();

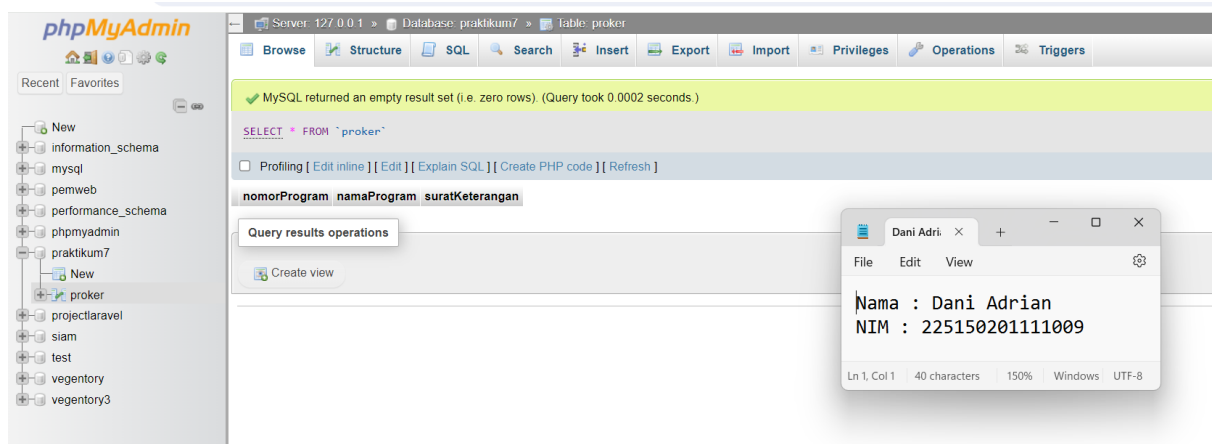
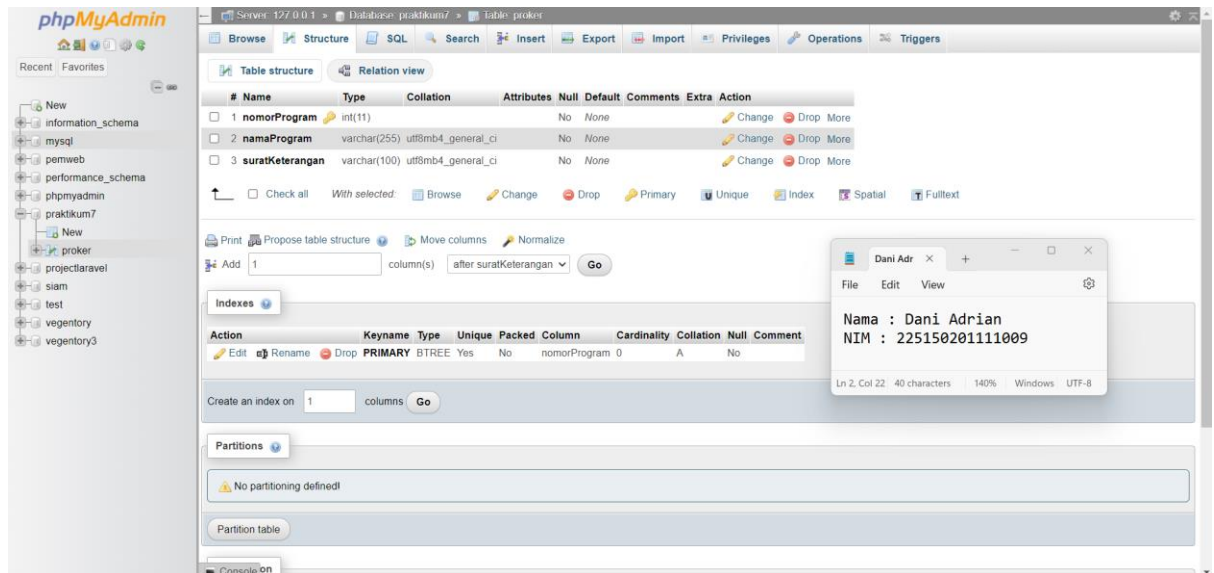
- a. Jalankan file index.php dan amati yang terjadi. Apa hubungannya dengan *sequence diagram* di bawah ini?



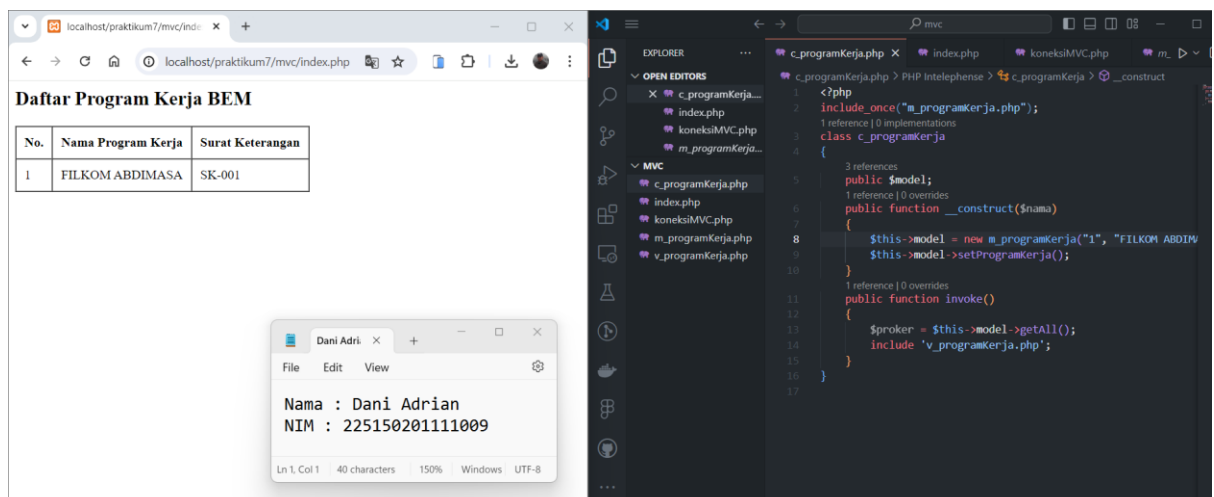
- b. Apa maksud dari pembuatan kode “m_programKerja.php”?
- c. Apa maksud dari pembuatan kode “c_programKerja.php”?
- d. Apa maksud dari pembuatan kode “v_programKerja.php”?
- e. Apa maksud dari pembuatan kode “index.php”?

B. Screenshoot

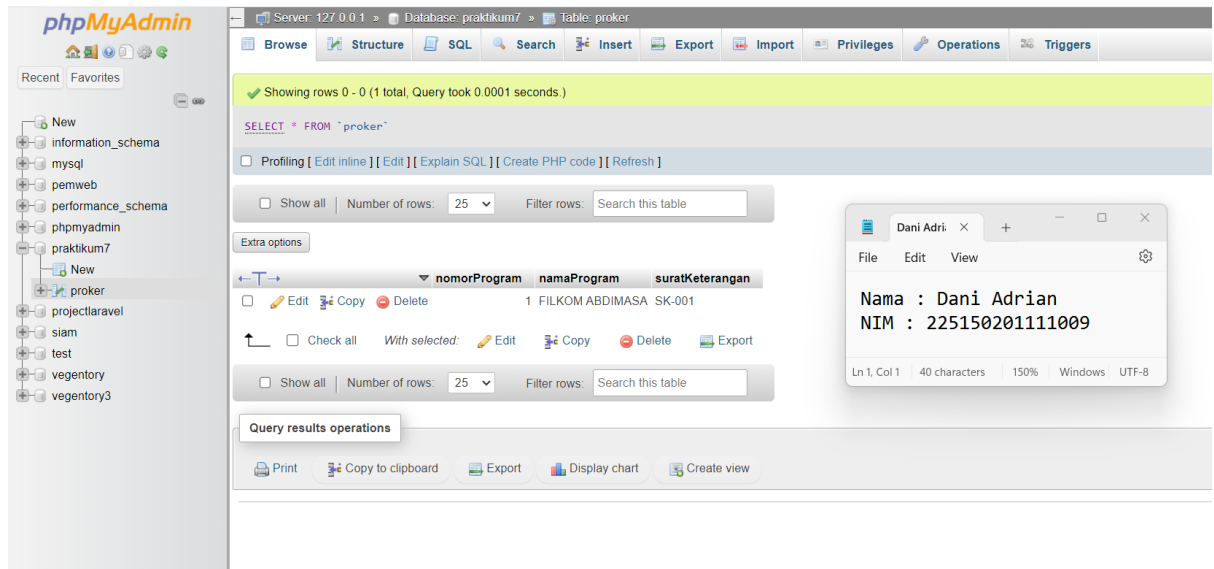
Database sebelum:



index.php



Database sesudah :



C. Syntax

koneksiMVC.php

```
1. <?php
2. $mysqli = mysqli_connect('localhost', 'root',
   '02030507', 'praktikum7');
```

m_programKerja.php

```
1. <?php
2. class m_programKerja
3. {
4.     private $nomorProgram;
5.     private $namaProgram;
6.     private $suratKeterangan;
7.     public $hasil = array();
8.     public function __construct($nomorProgram,
   $namaProgram, $suratKeterangan)
9.     {
10.         $this->nomorProgram = $nomorProgram;
11.         $this->namaProgram = $namaProgram;
12.         $this->suratKeterangan = $suratKeterangan;
13.     }
14.     // Add Program Kerja
15.     public function setProgramKerja()
16.     {
17.         require "koneksiMVC.php";
18.         $rs = $mysqli->query("INSERT INTO proker
   VALUES ('$this->nomorProgram', '$this-
   >namaProgram', '$this->suratKeterangan')");
19.     }
20.     // Get All ProgramKerja
21.     public function getAll()
22.     {
23.         require "koneksiMVC.php";
24.         $sql = "SELECT * FROM proker";
25.         $rs = $mysqli->query($sql);
26.         $rows = array();
```

27.	while (\$row = \$rs->fetch_assoc()) {
28.	\$rows[] = \$row;
29.	}
30.	\$this->hasil = \$rows;
31.	return \$this->hasil;
32.	}
33.	}

c_programKerja.php	
1.	<?php
2.	include_once("m_programKerja.php");
3.	class c_programKerja
4.	{
5.	public \$model;
6.	public function __construct(\$nama)
7.	{
8.	\$this->model = new m_programKerja("1",
9.	"FILKOM ABDIMASA", "SK-001");
10.	\$this->model->setProgramKerja();
11.	}
12.	public function invoke()
13.	{
14.	\$proker = \$this->model->getAll();
15.	include 'v_programKerja.php';
16.	}

v_programKerja.php	
1.	<html>
2.	
3.	<head></head>
4.	
5.	<body>
6.	<h2>Daftar Program Kerja BEM</h2>
7.	<table border="1" cellpadding='10'
8.	cellspacing='0'>
9.	<tbody>
10.	<tr>
11.	<th>No.</th>
12.	<th>Nama Program Kerja</th>
13.	<th>Surat Keterangan</th>
14.	</tr>
15.	</tbody>
16.	<?php
17.	foreach (\$proker as \$row) {
18.	echo "<tr>
19.	<td> \$row[nomorProgram]</td>
20.	<td> \$row[namaProgram]</td>
21.	<td> \$row[suratKeterangan]</td>
22.	</tr> ";
23.	}
24.	?>

25.	</body>
26.	
27.	

index.php	
1.	<?php include_once("c_programKerja.php"); \$controller = new c_programKerja("Program Kerja"); \$controller->invoke();
2.	
3.	
4.	

D. Penjelasan

Hubungan antara file index.php dan gambar diagram urutan tersebut mencerminkan serangkaian proses yang serupa. Di dalam index.php, user membuat permintaan yang kemudian ditangani oleh kontroler. Kontroler ini kemudian meminta data dari model dan menerima kembali hasilnya untuk kemudian diteruskan ke tampilan (view). Akhirnya, hasil gabungan dari kontroler, model, dan tampilan dikirim sebagai respons kepada pengguna, menciptakan tampilan yang mirip dengan yang ditunjukkan dalam tangkapan layar.

Dalam konteks program, m_programKerja bertindak sebagai model yang bertanggung jawab atas semua operasi yang terkait dengan basis data. Ini menyediakan fungsi setter dan getter untuk menambahkan atau mengambil nilai-nilai dalam basis data.

c_programKerja, sebagai kontroler, berperan dalam mengelola permintaan dari pengguna dan menghubungkannya dengan model dan tampilannya. Metode invoke digunakan untuk mengambil data dari model dan menampilkannya dalam tampilan.

v_programKerja bertindak sebagai tampilan yang bertugas menampilkan tabel dan konten dari basis data. Tampilan ini disesuaikan berdasarkan permintaan yang diterima dari kontroler.

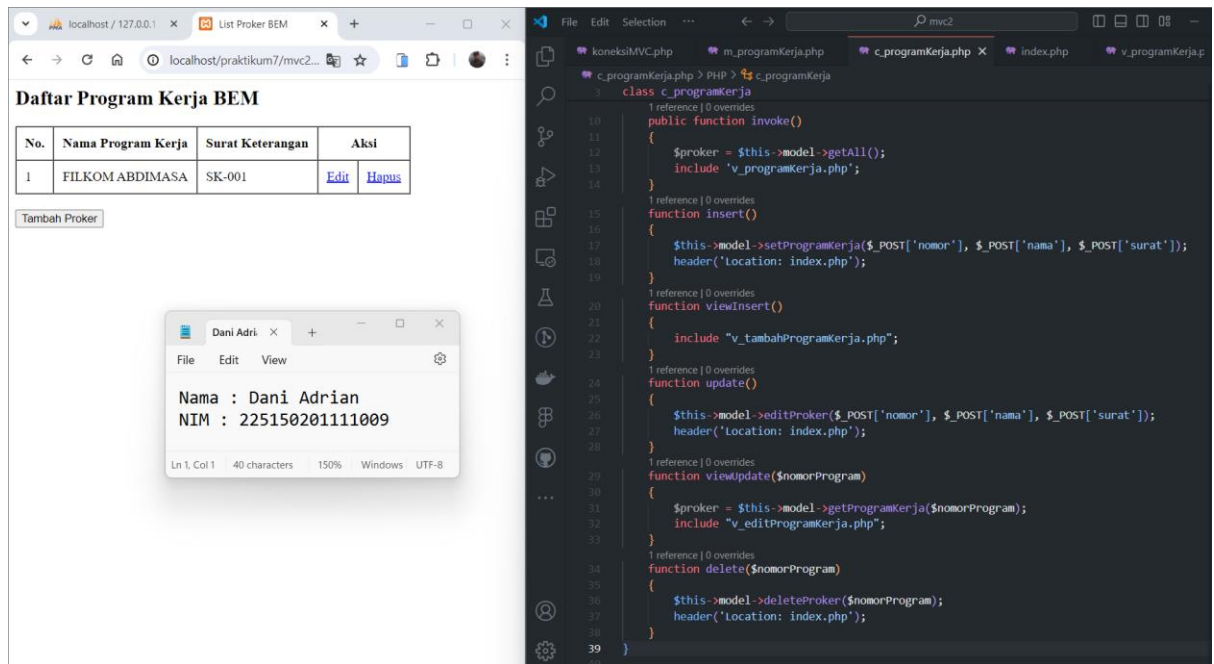
Secara keseluruhan, index.php berperan sebagai file utama yang dieksekusi pertama kali dan bertanggung jawab untuk menampilkan hasil dari permintaan pengguna.

LATIHAN 4

A. Soal

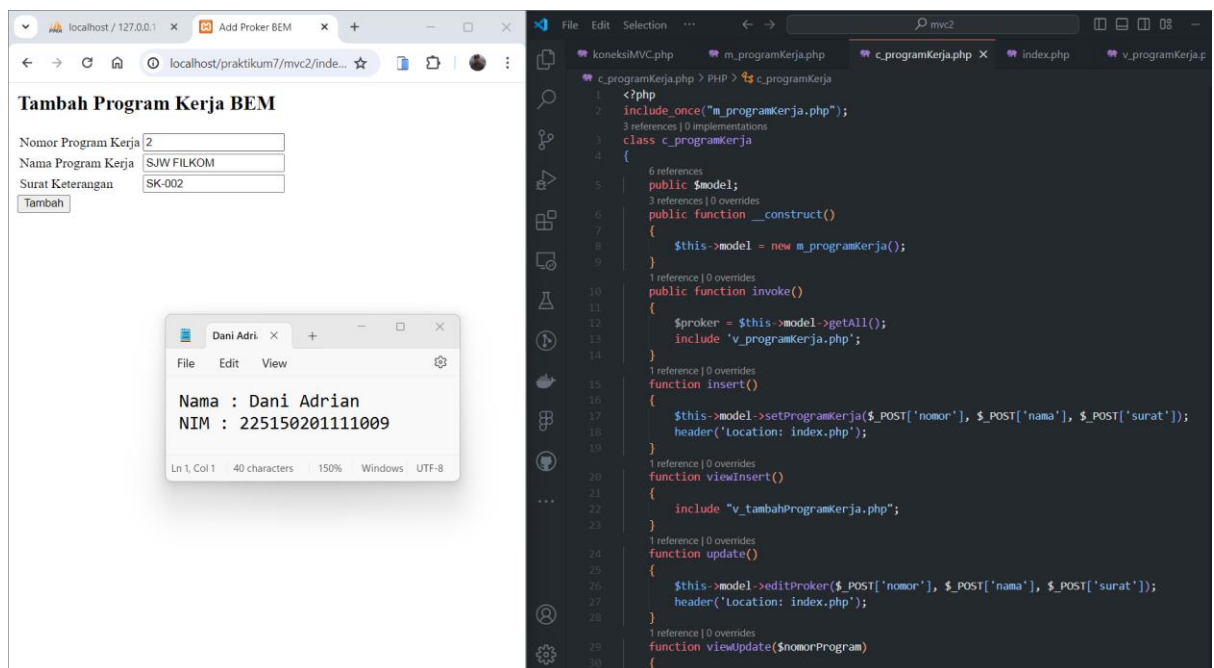
Modifikasilah Latihan 3 sehingga dapat dibuat program dengan kemampuan CRUD secara lengkap!

B. Screenshoot

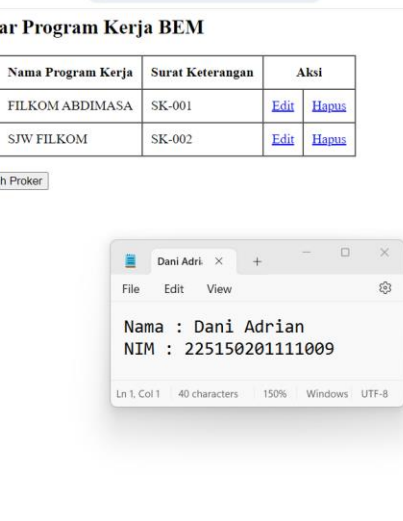


Form tambah

- Sebelum ditambah :



- Sesudah ditambah



The screenshot shows a web browser at localhost / 127.0.0.1 with the title 'List Proker BEM'. The main content is a table titled 'Daftar Program Kerja BEM' with the following data:

No.	Nama Program Kerja	Surat Keterangan	Aksi
1	FILKOM ABDIMASA	SK-001	Edit Hapus
2	SJW FILKOM	SK-002	Edit Hapus

Below the table is a button labeled 'Tambah Proker'.

A modal window is open, displaying the following information:

Nama : Dani Adrian
NIM : 225150201111009

The modal window has a title bar with 'Dani Adri' and standard window controls. It also includes a menu bar with 'File', 'Edit', and 'View', and a status bar at the bottom showing 'Ln 1, Col 1', '40 characters', '150%', 'Windows', and 'UTF-8'.

```
File Edit Selection ... mv2 ...
koneksimVC.php m_programKerja.php c_programKerja.php index.php v_programKerja.php
c_programKerja.php > PHP > c_programKerja
1 <?php
2 include_once("m_programKerja.php");
3 references | 0 implementations
3 class c_programKerja
4 {
5     6 references
6     public $model;
7     3 references | 0 overrides
8     public function __construct()
9     {
10         $this->model = new m_programKerja();
11     }
12     1 reference | 0 overrides
13     public function invoke()
14     {
15         $proker = $this->model->getall();
16         include 'v_programKerja.php';
17     }
18     1 reference | 0 overrides
19     function insert()
20     {
21         $this->model->setProgramKerja($_POST['nomor'], $_POST['nama'], $_POST['surat']);
22         header('Location: index.php');
23     }
24     1 reference | 0 overrides
25     function viewInsert()
26     {
27         include "v_tambahProgramKerja.php";
28     }
29     1 reference | 0 overrides
30     function update()
31     {
32         $this->model->editProker($_POST['nomor'], $_POST['nama'], $_POST['surat']);
33         header('Location: index.php');
34     }
35     1 reference | 0 overrides
36     function viewUpdate($nomorProgram)
37     {
38     }
```

Form Edit

The screenshot shows a web browser at localhost/127.0.0.1 displaying a list of programs. The table has four columns: No., Nama Program Kerja, Surat Keterangan, and Aksi. There are six rows of data. Each row has 'Edit' and 'Hapus' links in the 'Aksi' column. Below the table is a button labeled 'Tambah Proker'. In the foreground, a text editor window titled 'Dani Adri...' is open, showing the text 'Nama : Dani Adrian' and 'NIM : 225150201111009'.

No.	Nama Program Kerja	Surat Keterangan	Aksi
1	FILKOM ABDIMASA	SK-001	Edit Hapus
2	SJW FILKOM	SK-002	Edit Hapus
3	ENVIROLEARN	SK-003	Edit Hapus
4	FILKOM GREENERATION	SK-004	Edit Hapus
5	FILKOM PEDULI	SK-005	Edit Hapus
6	EDULIS	SK-006	Edit Hapus

Tambah Proker

Dani Adri. x + - □ x

File Edit View ⚙

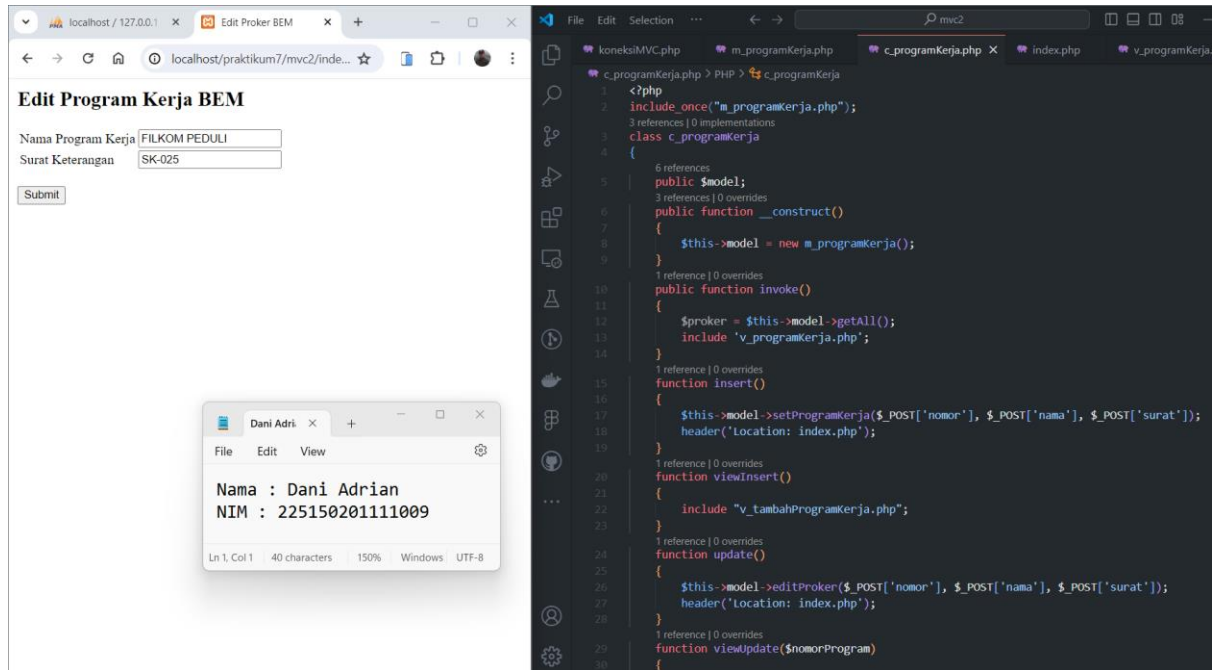
Nama : Dani Adrian
NIM : 225150201111009

Ln 1, Col 1 40 characters 150% Windows UTF-8

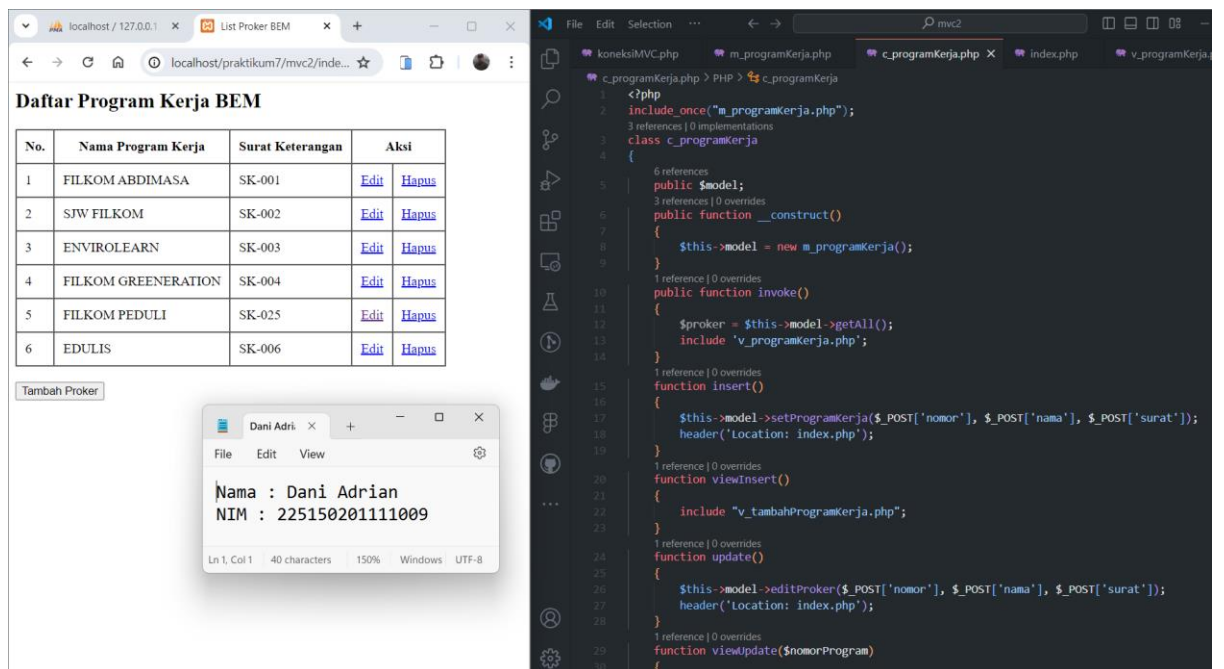
```

1  <?php
2  include_once("m_programKerja.php");
3  // references | 0 implementations
4  class c_programKerja
5  {
6      // 6 references
7      public $model;
8      // 3 references | 0 overrides
9      public function __construct()
10     {
11         $this->model = new m_programKerja();
12     }
13     // 1 reference | 0 overrides
14     public function invoke()
15     {
16         $proker = $this->model->getall();
17         include 'v_programKerja.php';
18     }
19     // 1 reference | 0 overrides
20     function insert()
21     {
22         $this->model->setProgramKerja($_POST['nomor'], $_POST['nama'], $_POST['surat']);
23         header('location: index.php');
24     }
25     // 1 reference | 0 overrides
26     function viewInsert()
27     {
28         include "v_tambahProgramKerja.php";
29     }
30     // 1 reference | 0 overrides
31     function update()
32     {
33         $this->model->editProker($_POST['nomor'], $_POST['nama'], $_POST['surat']);
34         header('location: index.php');
35     }
36     // 1 reference | 0 overrides
37     function viewUpdate($nomorProgram)
38     {

```

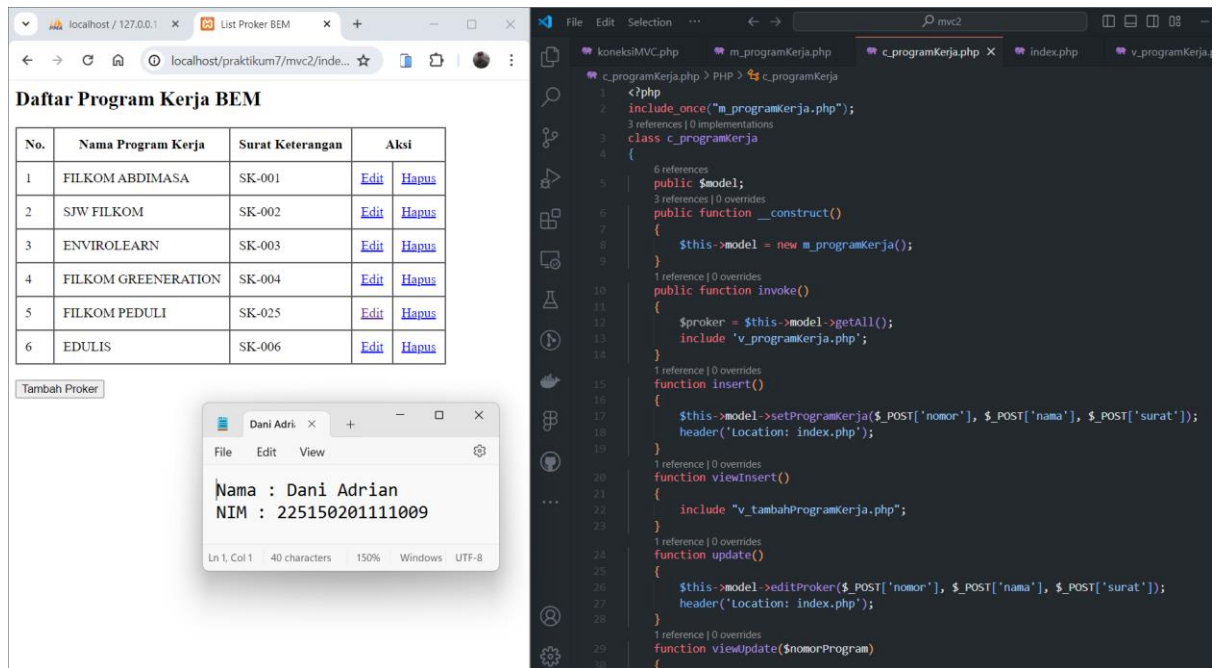


- Sesudah diedit

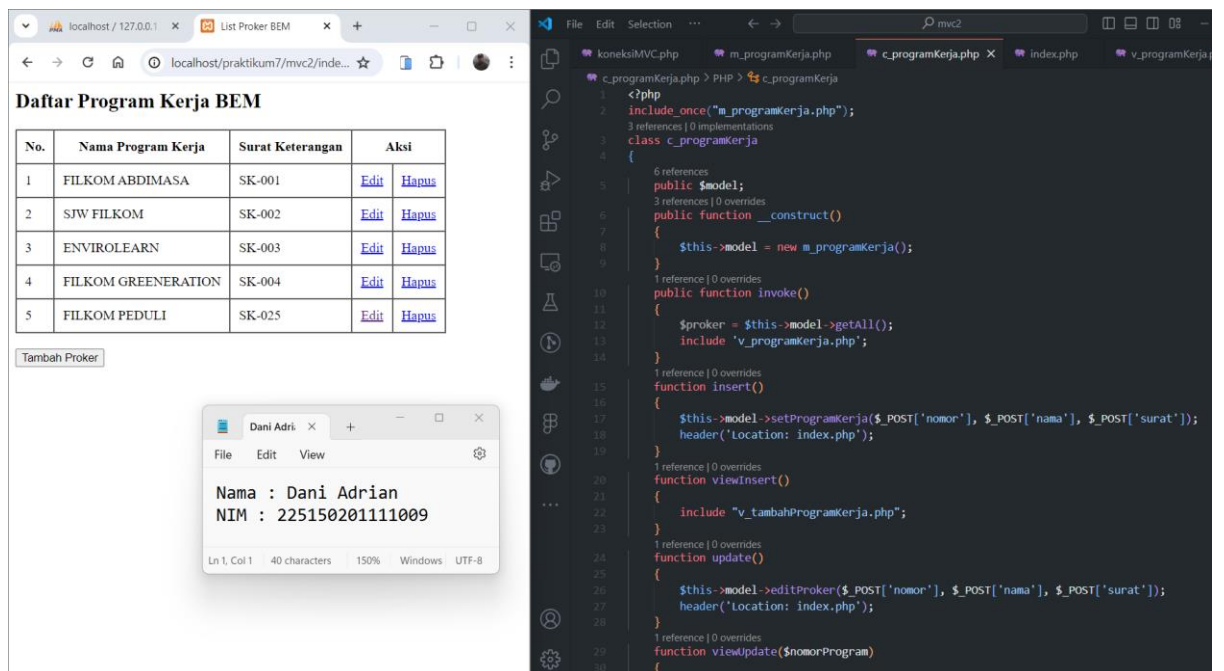


Hapus

- Sebelum dihapus



- Sesudah dihapus



C. Syntax

koneksiMVC.php	
1.	<?php
2.	\$mysqli = mysqli_connect('localhost', 'root', '02030507', 'praktikum7');

m_programKerja.php	
1.	<?php
2.	class m_programKerja
3.	{
4.	public \$hasil = array();

5.	private \$mysqli;
6.	public function __construct()
7.	{
8.	require 'koneksiMVC.php';
9.	\$this->mysqli = \$mysqli;
10.	}
11.	public function setProgramKerja(\$nomorProker, \$namaProker, \$suratKet)
12.	{
13.	\$rs = \$this->mysqli->query("INSERT INTO proker VALUES ('\$nomorProker', '\$namaProker', '\$suratKet')");
14.	}
15.	public function getProgramKerja(\$nomorProker)
16.	{
17.	\$rs = \$this->mysqli->query("SELECT * FROM proker WHERE nomorProgram = \$nomorProker");
18.	\$this->hasil = \$rs->fetch_array();
19.	return \$this->hasil;
20.	}
21.	public function deleteProker(\$nomorProker)
22.	{
23.	\$rs = \$this->mysqli->query("DELETE from proker WHERE nomorProgram = \$nomorProker");
24.	}
25.	public function editProker(\$nomorProker, \$namaProker, \$suratKet)
26.	{
27.	\$rs = \$this->mysqli->query("UPDATE proker SET namaProgram = '\$namaProker', suratKeterangan= '\$suratKet' WHERE nomorProgram = \$nomorProker");
28.	}
29.	public function getAll()
30.	{
31.	\$rs = \$this->mysqli->query("SELECT * FROM proker");
32.	while (\$row = \$rs->fetch_assoc()) {
33.	\$this->hasil[] = \$row;
34.	}
35.	return \$this->hasil;
36.	}
37.	}

c programKerja.php	
1.	<?php
2.	include_once("m_programKerja.php");
3.	class c_programKerja
4.	{
5.	public \$model;
6.	public function __construct()
7.	{
8.	\$this->model = new m_programKerja();
9.	}
10.	public function invoke()
11.	{

12.	\$proker = \$this->model->getAll();
13.	include 'v_programKerja.php';
14.	}
15.	function insert()
16.	{
17.	\$this->model->setProgramKerja(\$_POST['nomor'], \$_POST['nama'], \$_POST['surat']);
18.	header('Location: index.php');
19.	}
20.	function viewInsert()
21.	{
22.	include "v_tambahProgramKerja.php";
23.	}
24.	function update()
25.	{
26.	\$this->model->editProker(\$_POST['nomor'], \$_POST['nama'], \$_POST['surat']);
27.	header('Location: index.php');
28.	}
29.	function viewUpdate(\$nomorProgram)
30.	{
31.	\$proker = \$this->model->getProgramKerja(\$nomorProgram);
32.	include "v_editProgramKerja.php";
33.	}
34.	function delete(\$nomorProgram)
35.	{
36.	\$this->model->deleteProker(\$nomorProgram);
37.	header('Location: index.php');
38.	}
39.	}

v_programKerja.php	
1.	<html>
2.	
3.	<head>
4.	<title>List Proker BEM</title>
5.	</head>
6.	
7.	<body>
8.	<h2>Daftar Program Kerja BEM</h2>
9.	<table border="1" cellpadding="10" cellspacing="0">
10.	<tbody>
11.	<tr>
12.	<th>No.</th>
13.	<th>Nama Program Kerja</th>
14.	<th>Surat Keterangan</th>
15.	<th colspan="2">Aksi</th>
16.	</tr>
17.	</tbody>
18.	<?php
19.	foreach (\$proker as \$row) {

20.	echo " <tr>
21.	<td> \$row[nomorProgram]</td>
22.	<td> \$row[namaProgram]</td>
23.	<td> \$row[suratKeterangan]</td>
24.	<td>Edit </td>
25.	<td>Hapus </td>
26.	</tr> ";
27.	}
28.	?>
29.	</table>
30.	<h3><input type="button" value="Tambah Proker" onclick="window.location.href='?i=1' "></h3>
31.	</body>
32.	
33.	</html>

v_editprogramKerja.php	
1.	<html>
2.	
3.	<head>
4.	<title>Edit Proker BEM</title>
5.	</head>
6.	
7.	<body>
8.	<table>
9.	<tbody>
10.	<tr>
11.	<td>
12.	<h2>Edit Program Kerja BEM</h2>
13.	<form method="post" action="#">
14.	<input type="hidden" name="nomor" value="<?php echo \$proker['nomorProgram'] ?>">
15.	<table>
16.	<tr>
17.	<td><label>Nama Program Kerja</label></td>
18.	<td><input type="text" name="nama" value="<?php echo \$proker['namaProgram'] ?>"></td>
19.	</tr>
20.	<tr>
21.	<td><label>Surat Keterangan</label></td>
22.	<td><input type="text" name="surat" value="<?php echo \$proker['suratKeterangan'] ?>">
23.	</td>
24.	</tr>
25.	</table>
26.	<input type="submit" name="edit" value="Submit">
27.	</form>
28.	</td>
29.	</tr>

30.	</table>
31.	</body>
32.	
33.	</html>
34.	<?php
35.	if (isset(\$_POST['edit'])) {
36.	\$control = new c_programKerja();
37.	\$control->update();
38.	}
39.	?>

v_tambahprogramKerja.php	
1.	<html>
2.	
3.	<head>
4.	<title>Add Proker BEM</title>
5.	</head>
6.	
7.	<body>
8.	<table>
9.	<tbody>
10.	<tr>
11.	<td>
12.	<h2>Tambah Program Kerja
13.	BEM</h2>
14.	<form method="post" action="">
15.	<table>
16.	<tr>
17.	<td><label>Nomor
18.	Program Kerja</label></td>
19.	<td><input
20.	type="text" name="nomor" value=""></td>
21.	</tr>
22.	<tr>
23.	<td><label>Nama
24.	Program Kerja</label></td>
25.	<td><input
26.	type="text" name="nama" value="">
27.	</tr>
28.	<tr>
29.	<td><label>Surat
30.	Keterangan</label></td>
31.	<td><input
32.	type="text" name="surat" value=""></td>
33.	</tr>
34.	</table>
35.	<input type="submit"
36.	name="ins" value="Tambah">
37.	</form>
38.	</td>
39.	</tr>
40.	</table>
41.	</body>
42.	

35.	</html>
36.	<?php
37.	if (isset(\$_POST['ins'])) {
38.	\$control = new c_programKerja();
39.	\$control->insert();
40.	}
41.	?>

index.php	
1.	<?php
2.	include_once("c_programKerja.php");
3.	\$controller = new c_programKerja();
4.	if (isset(\$_GET['d'])) {
5.	\$controller->delete(\$_GET['d']);
6.	} else if (isset(\$_GET['u'])) {
7.	\$controller->viewUpdate(\$_GET['u']);
8.	} else if (isset(\$_GET['i'])) {
9.	\$controller->viewInsert();
10.	} else {
11.	\$controller->invoke();
12.	}

D. Penjelasan

Ketika kode program dijalankan, akan muncul daftar program kerja BEM. Di dalam kode program tersebut, pengguna dapat mengelola program kerja dengan melakukan tindakan menambahkan, mengedit, atau menghapus program kerja. Jika tombol "Tambah Proker" ditekan, pengguna akan dialihkan ke halaman untuk menambahkan program kerja dengan mengisi formulir yang disediakan. Jika tombol "Edit" di kolom Aksi ditekan, pengguna akan diarahkan ke halaman yang sudah berisi nilai dari program kerja yang dipilih, yang kemudian dapat diedit kecuali nomor program kerja tersebut. Jika tombol "Hapus" di kolom Aksi ditekan, program kerja yang dipilih akan secara otomatis terhapus seperti yang terlihat dalam tangkapan layar.

Dikarenakan file c_programKerja.php berfungsi sebagai kontroler, file tersebut berisi beberapa metode kontrol untuk daftar program kerja BEM. Ini termasuk fungsi invoke untuk mengambil data dari kelas model dan menampilkannya dalam tampilan, fungsi insert untuk menambahkan program kerja BEM, fungsi viewUpdate untuk melakukan pengeditan program kerja, dan fungsi delete untuk menghapus program kerja.

Berbeda dengan file c_programKerja, dalam file m_programKerja.php fokus pada operasi-operasi atau modifikasi pada basis datanya, sehingga terdapat query MySQL dalam sintaksis file PHP tersebut. Di dalam file c_programKerja.php ini terdapat fungsi setProgramKerja yang digunakan untuk memberikan nilai atribut program kerja seperti nomor program, nama program, dan keterangan surat pada basis data. Selain itu, terdapat juga fungsi deleteProker untuk menghapus nilai variabel program kerja pada basis data, fungsi editProker untuk mengupdate atau membuat nilai baru untuk variabel program kerja di basis data, dan fungsi getAll yang mengambil nilai atribut program kerja dari basis data.

File v_tambahProgramKerja.php merupakan formulir untuk menambahkan program kerja BEM dengan menggunakan kontrol insert. Sedangkan file v_editProgramKerja.php

adalah formulir untuk mengedit program kerja BEM dengan menggunakan kontrol update. Penggunaan kontrol tersebut merupakan turunan dari file c_programKerja.php.

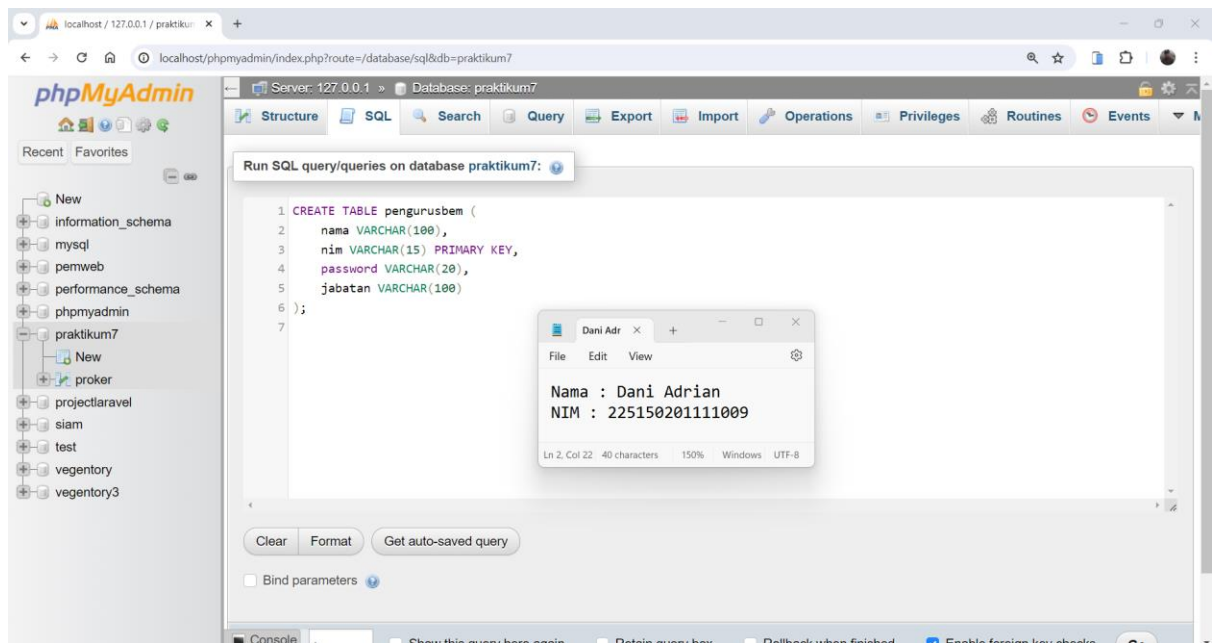
LATIHAN 5

A. Soal

Modifikasi sesuai kebutuhan terlebih dahulu, lengkapi dengan halaman login dan halaman administrasi dan gabungkan Latihan 2 dan Latihan 3 sehingga didapat skenario Menteri hanya dapat melihat daftar program kerja saja, sedangkan Kepala Departemen dapat melakukan proses manajemen program kerja (CRUD). Lakukan modifikasi pada database dan struktur class model, controller dan view jika diperlukan untuk mengakomodasi skenario tersebut.

B. Screenshoot

- Membuat Database pengurusbem



localhost / 127.0.0.1 / praktikum7

localhost/phpmyadmin/index.php?route=/table/sql&db=praktikum7&table=pengurusbem

phpMyAdmin

Recent Favorites

New

- information_schema
- mysql
- pemweb
- performance_schema
- phpmyadmin
- praktikum7
 - New
 - pengurusbem
 - proker
 - projectlaravel
 - siam
 - test
 - vegetory
 - vegetory3

Server: 127.0.0.1 > Database: praktikum7 > Table: pengurusbem

Browse Structure SQL Search Insert Export Import Privileges Operations Triggers

Run SQL query/queries on table praktikum7.pengurusbem:

```
1 INSERT INTO pengurusbem (nama, nim, password, jabatan) VALUES
2 ('Dani Adrian', '225150201111009', '1234', 'Menteri'),
3 ('Zakki', '225150201111010', '505', 'Kepala Departemen');
```

nama
nim
password
jabatan

Dani Adr x + - □ ×
File Edit View
Nama : Dani Adrian
NIM : 225150201111009
Ln 2, Col 22: 40 characters 150% Windows UTF-8

SELECT * SELECT INSERT UPDATE DELETE Clear Format Get auto-saved query

☐ Bind parameters

Console Show this query here again Retain query box Rollback when finished ☒ Enable foreign key checks Go

phpMyAdmin

Recent Favorites

New

- information_schema
- mysql
- pemweb
- performance_schema
- phpmyadmin
- praktikum7
 - New
 - pengurusbem
 - proker
 - projectlaravel
 - siam
 - test
 - vegetory
 - vegetory3

Server: 127.0.0.1 > Database: praktikum7

Structure SQL Search Query Export Import Operations Privileges Routines Events

Run SQL query/queries on database praktikum7:

```
1 DELIMITER $$
2
3 CREATE PROCEDURE pengurusbem(IN nim_param VARCHAR(20), IN password_param VARCHAR(20))
4 BEGIN
5     SELECT * FROM pengurusbem WHERE nim = nim_param AND password = password_param;
6 END$$
7
8 DELIMITER ;
```

Dani Adr x + - □ ×
File Edit View
Nama : Dani Adrian
NIM : 225150201111009
Ln 2, Col 22: 40 characters 150% Windows UTF-8

Clear Format Get auto-saved query

☐ Bind parameters

Console Show this query here again Retain query box Rollback when finished ☒ Enable foreign key checks Go

phpMyAdmin

Recent Favorites

New

- information_schema
- mysql
- pemweb
- performance_schema
- phpmyadmin
- praktikum7
 - Procedures
 - Tables
 - New
 - pengurusbem
 - proker
 - projectlaravel
 - siam
 - test
 - vegetory
 - vegetory3

Server: 127.0.0.1 > Database: praktikum7 > Table: pengurusbem

Browse Structure SQL Search Insert Export Import Privileges Operations Triggers

Showing rows 0 - 1 (2 total, Query took 0.0002 seconds.)

SELECT * FROM `pengurusbem`

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

☐ Show all Number of rows: 25 Filter rows: Search this table Sort by key: None

Extra options

	nama	nim	password	jabatan
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	Dani Adrian	225150201111009	1234	Menteri
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	Zakki	225150201111010	505	Kepala Departemen

☐ Check all With selected: ☐ Edit ☐ Copy ☐ Delete ☐ Export

☐ Show all Number of rows: 25 Filter rows: Search this table Sort by key: None

Query results operations

Print Copy to clipboard Export Display chart Create view

Console

Dani Adr x + - □ ×
File Edit View
Nama : Dani Adrian
NIM : 225150201111009
Ln 2, Col 22: 40 characters 150% Windows UTF-8

- Form login sebagai Kepala Departemen

The screenshot displays a web browser window at `localhost/praktikum7/mvc3/login.php` showing a login form. The form has two input fields: "NIM" with the value "225150201111010" and "Password" with masked characters. A "Login" button is at the bottom. An overlay window shows the user's name "Dani Adrian" and NIM "225150201111009". To the right, the VS Code editor shows the `login.php` file with the following HTML and PHP code:

```

1 <html>
2
3 <head>
4 <title>Login</title>
5 </head>
6
7 <body>
8 <form action="admin.php" method="post">
9 <table>
10 <tr>
11 <td>
12 <p1>NIM</p1>
13 </td>
14 <td>
15 <input type="text" id="nim" name="nim" placeholder="nim">
16 </td>
17 </tr>
18 <tr>
19 <td>
20 <p2>Password</p2>
21 </td>
22 <td>
23 <input type="password" id="pass" name="password" placeholder="password">
24 </td>
25 </tr>
26 <tr>
27 <td colspan="2">
28 <input type="submit" id="tombol" name="tombol" value="Login">
29 </td>
30 </tr>
31 </table>
32 </form>
33 </body>
34 </html>
35
36 <?php
37 $agagal = "";
38 if (isset($_GET['e'])) {
39     if ($_GET['e'] == 1) {
40         $agagal = "NIM atau password salah !";
41         echo $agagal;
42     }
43 }

```

- Daftar program kerja BEM sebagai Kepala Departemen

The screenshot displays a web browser window at `localhost/praktikum7/mvc3/index.php` showing a "Daftar Program Kerja BEM" page. The page has a "Tambah Proker" button and a table with 5 rows of program data. Each row has "Edit" and "Hapus" links. A "Logout" button is at the bottom. An overlay window shows the user's name "Dani Adrian" and NIM "225150201111009". To the right, the VS Code editor shows the `c_programKerja.php` file with the following PHP code:

```

1 <?php
2 include_once("m_programKerja.php");
3
4 class c_programKerja
5 {
6     public $model;
7     public function __construct()
8     {
9         $this->model = new m_programKerja();
10    }
11    public function invoke($nama, $nim, $jabatan)
12    {
13        $proker = $this->model->getAll();
14        $n = $nama;
15        $ni = $nim;
16        $j = $jabatan;
17        include 'v_programKerja.php';
18    }
19    public function insert()
20    {
21        $this->model->setProgramKerja(
22            $_POST['nomor'],
23            $_POST['nama'],
24            $_POST['surat']
25        );
26        header('Location: index.php');
27    }
28    public function viewInsert()
29    {
30        include "v_tambahProgramKerja.php";
31    }
32    public function update()
33    {
34    }
35 }

```

- Form tambah program kerja BEM

localhost / 127.0.0.1 / praktikum7 / mvc3 / index.php?i=1

Tambah Program Kerja BEM

Nomor Program Kerja

Nama Program Kerja

Surat Keterangan

```

1 <html>
2
3 <head>
4 <title>Tambah Program Kerja BEM</title>
5 </head>
6
7 <body>
8 <table>
9 <tbody>
10 <tr>
11 <td>
12 <h2> Tambah Program Kerja BEM </h2>
13 <form method="post" action="">
14 <table>
15 <tr>
16 <td><label>Nomor
17 Program Kerja</label></td>
18 <td><input type="text" name="nomor" value=
19
20 </tr>
21 <tr>
22 <td><label>Nama Program
23 Kerja</label></td>
24 <td><input type="text" name="nama" value=
25
26 </tr>
27 <tr>
28 <td><label>Surat
29 Keterangan</label></td>
30 <td><input type="text" name="surat" value=
31
32 </tr>
33 </table>
34 <input type="submit" name="ins" value="Tambah">
35 </form>
36 </td>
37 </tr>
38 </tbody>
39 </table>
40 </body>

```

Dani Adr

Nama : Dani Adrian
NIM : 225150201111009

localhost / 127.0.0.1 / praktikum7 / mvc3 / index.php?name=Zakki...

Daftar Program Kerja BEM

No.	Nama Program Kerja	Surat Keterangan	Aksi
1	FILKOM ABDIMASA	SK-001	Edit Hapus
2	SJW FILKOM	SK-002	Edit Hapus
3	ENVIROLEARN	SK-003	Edit Hapus
4	FILKOM GREENERATION	SK-004	Edit Hapus
5	FILKOM PEDULI	SK-025	Edit Hapus
6	EDULIS	SK-006	Edit Hapus

```

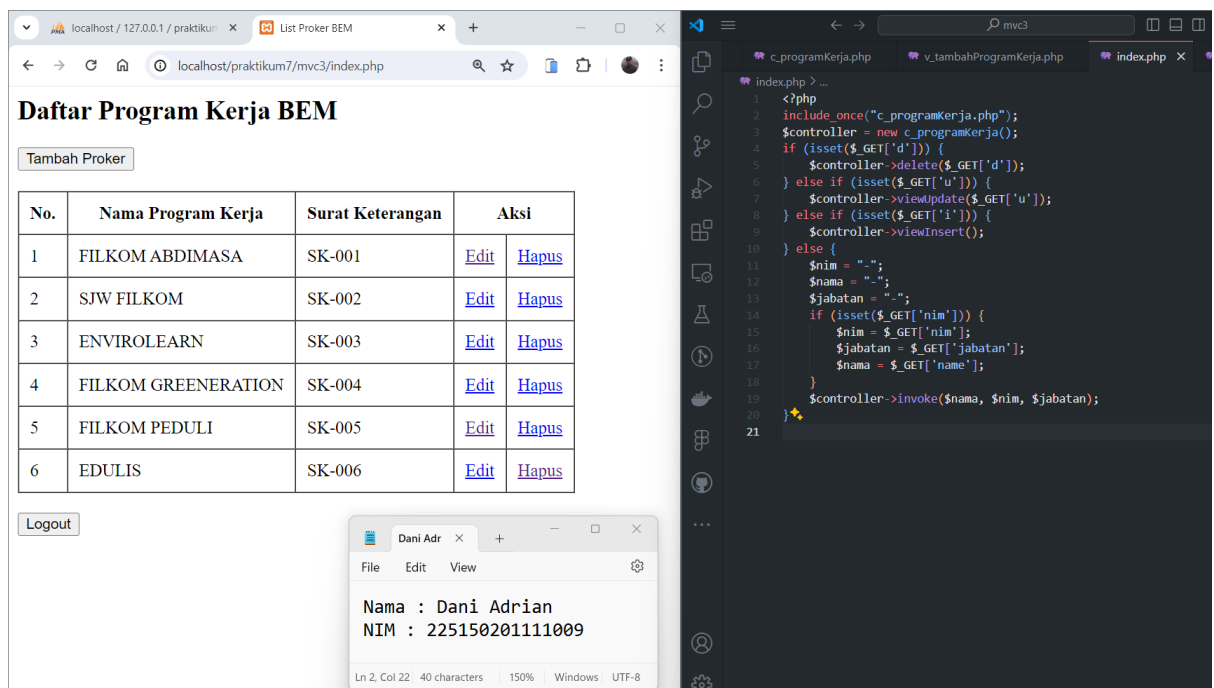
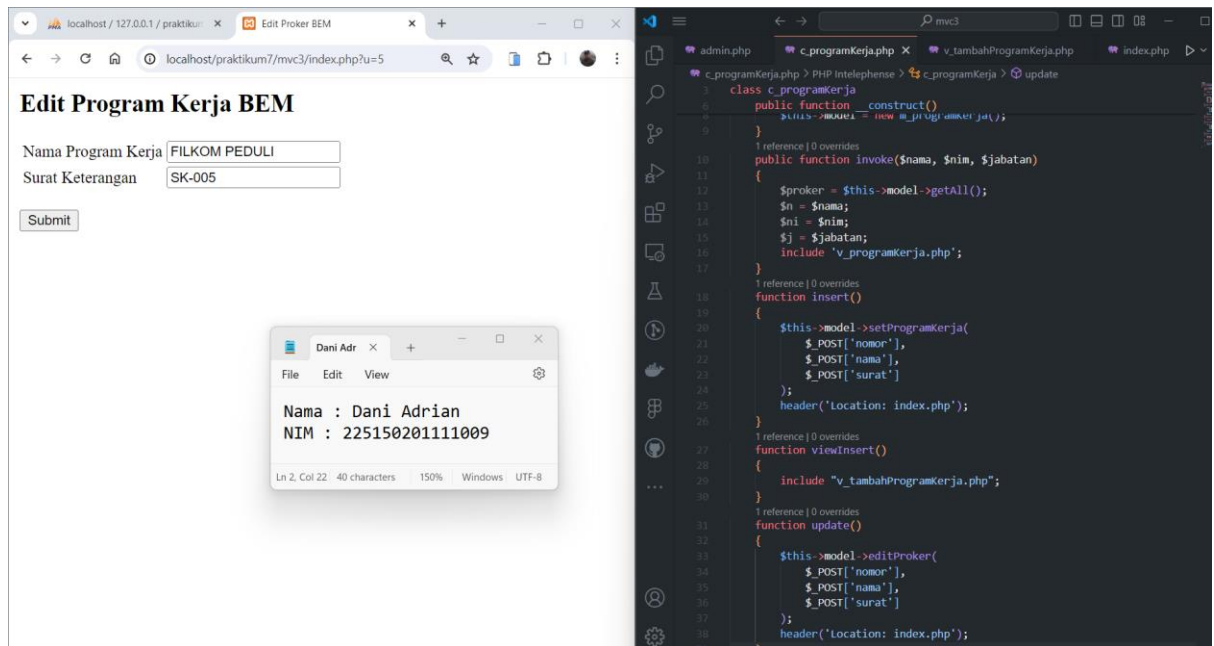
1 <?php
2 include_once("c_programKerja.php");
3 $controller = new c_programKerja();
4 if (isset($_GET['d'])) {
5     $controller->delete($_GET['d']);
6 } else if (isset($_GET['u'])) {
7     $controller->viewupdate($_GET['u']);
8 } else if (isset($_GET['i'])) {
9     $controller->viewinsert();
10 } else {
11     $nim = "-";
12     $nama = "-";
13     $jabatan = "-";
14     if (isset($_GET['nim'])) {
15         $nim = $_GET['nim'];
16         $jabatan = $_GET['jabatan'];
17         $nama = $_GET['name'];
18     }
19     $controller->invoke($nama, $nim, $jabatan);
20 }
21

```

Dani Adr

Nama : Dani Adrian
NIM : 225150201111009

- Form edit program kerja BEM



- Hapus salah satu program kerja BEM

localhost / 127.0.0.1 / praktikum7 / mvc3 / index.php

Daftar Program Kerja BEM

Tambah Proker

No.	Nama Program Kerja	Surat Keterangan	Aksi
1	FILKOM ABDIMASA	SK-001	Edit Hapus
2	SJW FILKOM	SK-002	Edit Hapus
3	ENVIROLEARN	SK-003	Edit Hapus
4	FILKOM GREENERATION	SK-004	Edit Hapus
5	FILKOM PEDULI	SK-005	Edit Hapus

Logout

Dani Adr

Nama : Dani Adrian
NIM : 225150201111009

```

class c_programKerja
{
    function insert()
    {
        $this->model->setProgramKerja(
            $_POST['nomor'],
            $_POST['nama'],
            $_POST['surat']
        );
        header('Location: index.php');
    }

    1 reference | 0 overrides
    function viewInsert()
    {
        include "v_tambahProgramKerja.php";
    }

    1 reference | 0 overrides
    function update()
    {
        $this->model->editProker(
            $_POST['nomor'],
            $_POST['nama'],
            $_POST['surat']
        );
        header('Location: index.php');
    }

    1 reference | 0 overrides
    function viewUpdate($nomorProgram)
    {
        $proker = $this->model->getProgramKerja($nomorProgram);
        include "v_editProgramKerja.php";
    }

    1 reference | 0 overrides
    function delete($nomorProgram)
    {
        $this->model->deleteProker($nomorProgram);
        header('Location: index.php');
    }
}

```

- Form login sebagai Menteri

localhost / 127.0.0.1 / praktikum7 / mvc3 / login.php

Login

NIM :

Password :

Login

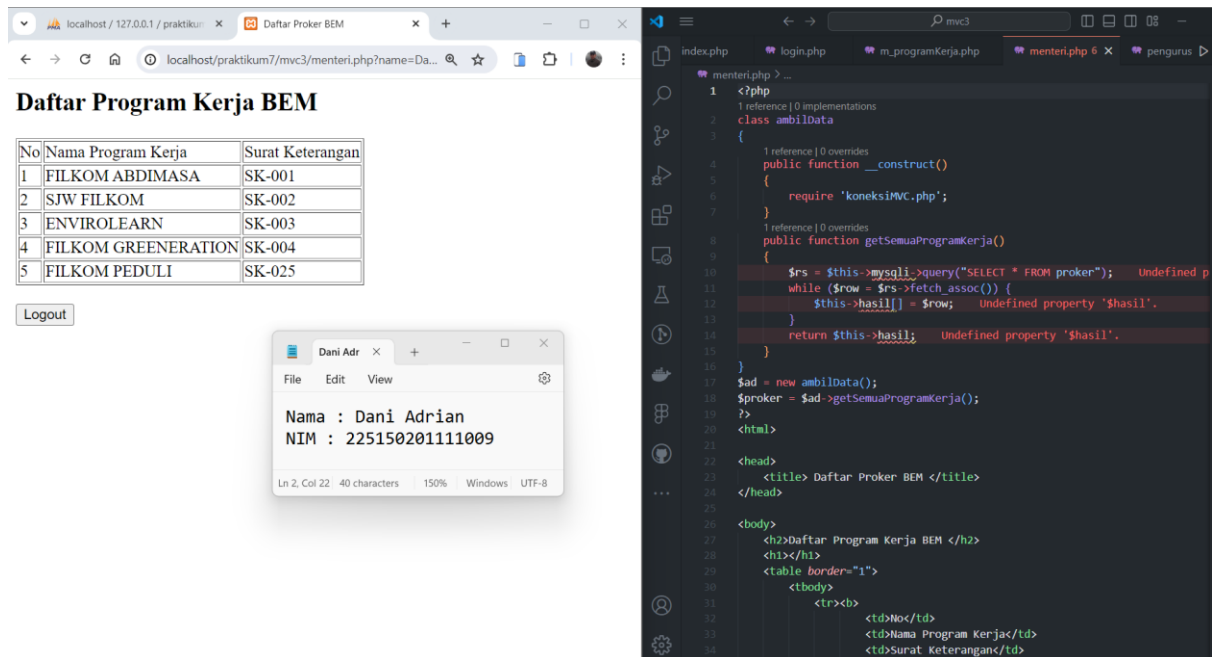
Dani Adr

Nama : Dani Adrian
NIM : 225150201111009

```

<html>
<head>
<h1>login</h1>
</head>
<body>
<form action="admin.php" method="post">
<table>
<tr>
<td>
<p1>NIM</p1>
</td>
<td> : </td>
<td><input type="text" id="nim" name="nim" placeholder="nim">
</tr>
<tr>
<td>
<p2>Password</p2>
</td>
<td> : </td>
<td><input type="password" id="pass" name="password" place
</tr>
<tr>
<td><input type="submit" id="tombol" name="tombol" value="
</td></td>
</tr>
</table>
</form>
</body>
<?php
$agagal = "";
if (isset($_GET['e'])) {
    if ($_GET['e'] == 1) {
        $agagal = "NIM atau password salah !";
        echo $agagal;
    }
}

```



C. Syntax

CREATE TABLE penguruBEM	
1.	CREATE TABLE pengurusbem (
2.	nama VARCHAR(100),
3.	nim VARCHAR(15) PRIMARY KEY,
4.	password VARCHAR(20),
5.	jabatan VARCHAR(100)
6.);

1.	DELIMITER \$\$
2.	
3.	CREATE PROCEDURE pengurusbem(IN nim_param
4.	VARCHAR(20), IN password_param VARCHAR(20))
5.	BEGIN
6.	SELECT * FROM pengurusbem WHERE nim = nim_param
7.	AND password = password_param;
8.	END\$\$
	DELIMITER ;

koneksiMVC.php	
1.	<?php
2.	\$this->mysqli = new mysqli('localhost', 'root',
	'02030507', 'praktikum7');

admin.php	
1.	<?php
2.	\$conn = new mysqli("localhost", "root", "02030507",
	"praktikum7");
3.	\$nim = \$_POST['nim'];

4.	\$password = \$_POST['password'];
5.	\$query = "CALL pengurusbem('\$nim','\$password')";
6.	\$hasil = \$conn->query(\$query) or die('query gagal');
7.	if (\$hasil->num_rows > 0) {
8.	\$nama = "";
9.	\$nim = "";
10.	\$password = "";
11.	\$jabatan = "";
12.	while (\$data = \$hasil->fetch_object()) {
13.	\$nama = \$data->nama;
14.	\$nim = \$data->nim;
15.	\$password = \$data->password;
16.	\$jabatan = \$data->jabatan;
17.	}
18.	require 'pengurusBEM.php';
19.	\$pb = new pengurusBEM();
20.	\$pb->setJabatan(\$jabatan);
21.	\$pb->setNama(\$nama);
22.	\$pb->setNim(\$nim);
23.	\$pb->pilih();
24.	} else {
25.	header('location: login.php?e=1');
26.	}
27.	?>

index.php	
1.	<?php
2.	include_once("c_programKerja.php");
3.	\$controller = new c_programKerja();
4.	if (isset(\$_GET['d'])) {
5.	\$controller->delete(\$_GET['d']);
6.	} else if (isset(\$_GET['u'])) {
7.	\$controller->viewUpdate(\$_GET['u']);
8.	} else if (isset(\$_GET['i'])) {
9.	\$controller->viewInsert();
10.	} else {
11.	\$nim = "-";
12.	\$nama = "-";
13.	\$jabatan = "-";
14.	if (isset(\$_GET['nim'])) {
15.	\$nim = \$_GET['nim'];
16.	\$jabatan = \$_GET['jabatan'];
17.	\$nama = \$_GET['name'];
18.	}
19.	\$controller->invoke(\$nama, \$nim, \$jabatan);
20.	}

login.php	
1.	<html>
2.	
3.	<head>
4.	<h1>Login</h1>
5.	</head>

6.	
7.	<body>
8.	<form action="admin.php" method="post">
9.	<table>
10.	<tr>
11.	<td>
12.	<p1>NIM</p1>
13.	</td>
14.	<td> : </td>
15.	<td><input type="text" id="nim"
16.	name="nim" placeholder="nim"> </td>
17.	</tr>
18.	<tr>
19.	<td>
20.	<p2>Password</p2>
21.	</td>
22.	<td> : </td>
23.	<td><input type="password"
24.	id="pass" name="password"
25.	placeholder="password"> </td>
26.	</tr>
27.	<tr>
28.	<td><input type="submit"
29.	id="tombol" name="tombol" value="Login"></td>
30.	<td></td>
31.	<td></td>
32.	</tr>
33.	</table>
34.	</form>
35.	</body>
36.	<?php
37.	\$gagal = "";
38.	if (isset(\$_GET['e'])) {
39.	if (\$_GET['e'] == 1) {
40.	\$gagal = "NIM atau password salah !";
41.	echo \$gagal;
42.	} else if (\$_GET['e'] == 2) {
43.	echo "Terjadi Kesalahan, Login Kembali !";
44.	}
	}
	?>
	</html>

c_programKerja.php	
1.	<?php
2.	include_once("m_programKerja.php");
3.	class c_programKerja
4.	{
5.	public \$model;
6.	public function __construct()
7.	{
8.	\$this->model = new m_programKerja();
9.	}

10.	public function invoke(\$nama, \$nim, \$jabatan)
11.	{
12.	\$proker = \$this->model->getAll();
13.	\$n = \$nama;
14.	\$ni = \$nim;
15.	\$j = \$jabatan;
16.	include 'v_programKerja.php';
17.	}
18.	function insert()
19.	{
20.	\$this->model->setProgramKerja(
21.	\$_POST['nomor'],
22.	\$_POST['nama'],
23.	\$_POST['surat']
24.);
25.	header('Location: index.php');
26.	}
27.	function viewInsert()
28.	{
29.	include "v_tambahProgramKerja.php";
30.	}
31.	function update()
32.	{
33.	\$this->model->editProker(
34.	\$_POST['nomor'],
35.	\$_POST['nama'],
36.	\$_POST['surat']
37.);
38.	header('Location: index.php');
39.	}
40.	function viewUpdate(\$nomorProgram)
41.	{
42.	\$proker = \$this->model->
43.	getProgramKerja(\$nomorProgram);
44.	include "v_editProgramKerja.php";
45.	}
46.	function delete(\$nomorProgram)
47.	{
48.	\$this->model->deleteProker(\$nomorProgram);
49.	header('Location: index.php');
50.	}

m_programKerja.php	
1.	<?php
2.	class m_programKerja
3.	{
4.	public \$hasil = array();
5.	private \$mysqli;
6.	public function __construct()
7.	{
8.	require 'koneksiMVC.php';
9.	}
10.	public function setProgramKerja(

11.	\$nomorProker,
12.	\$namaProker,
13.	\$suratKet
14.) {
15.	require 'koneksiMVC.php';
16.	\$rs = \$this->mysql->query("INSERT INTO
17.	proker
18.	VALUES ('\$nomorProker',
19.	'\$namaProker', '\$suratKet')");
20.	}
21.	public function getProgramKerja(\$nomorProker)
22.	{
23.	require 'koneksiMVC.php';
24.	\$rs = \$this->mysql->query("SELECT * FROM
25.	proker WHERE nomorProgram = \$nomorProker");
26.	\$this->hasil = \$rs->fetch_array();
27.	return \$this->hasil;
28.	}
29.	public function deleteProker(\$nomorProker)
30.	{
31.	require 'koneksiMVC.php';
32.	\$rs = \$this->mysql->query("DELETE from
33.	proker
34.	WHERE nomorProgram = \$nomorProker");
35.	}
36.	public function editProker(
37.	\$nomorProker,
38.	\$namaProker,
39.	\$suratKet
40.) {
41.	require 'koneksiMVC.php';
42.	\$rs = \$this->mysql->query("UPDATE proker
43.	SET
44.	namaProgram = '\$namaProker', suratKeterangan=
45.	'\$suratKet' WHERE nomorProgram = \$nomorProker");
46.	}
47.	public function getAll()
48.	{
49.	require 'koneksiMVC.php';
50.	\$rs = \$this->mysql->query("SELECT * FROM
51.	proker");
52.	while (\$row = \$rs->fetch_assoc()) {
53.	\$this->hasil[] = \$row;
54.	}
55.	return \$this->hasil;
56.	}
57.	}

pengurusBEM.php	
1.	<?php
2.	class pengurusBEM
3.	{
4.	private \$nama;
5.	private \$nim;

6.	private \$angkatan;
7.	private \$jabatan;
8.	public function setName(\$nama)
9.	{
10.	\$this->nama = \$nama;
11.	}
12.	public function getName()
13.	{
14.	return \$this->nama;
15.	}
16.	public function setJabatan(\$jabatan)
17.	{
18.	\$this->jabatan = \$jabatan;
19.	}
20.	public function getJabatan()
21.	{
22.	return \$this->jabatan;
23.	}
24.	public function setNim(\$nim)
25.	{
26.	\$this->nim = \$nim;
27.	}
28.	public function getNim()
29.	{
30.	return \$this->nim;
31.	}
32.	public function pilih()
33.	{
34.	\$jabat = \$this->getJabatan();
35.	\$nama = \$this->getName();
36.	\$nim = \$this->getNim();
37.	if (\$jabat == "Kepala Departemen") {
38.	echo "<meta http-equiv=\"refresh\"
39.	content=\"0;index.php?name=\$nama&nim=\$nim&jabatan=\$
40.	jabat\">";
41.	} else if (\$jabat = "Menteri") {
42.	echo "<meta http-equiv=\"refresh\"
43.	content=\"0;menteri.php?name=\$nama&nim=\$nim&jabatan
44.	= \$jabat\">";
45.	} else {
46.	header('location: login.php?e=2');
47.	}

menteri.php	
1.	<?php
2.	class ambilData
3.	{
4.	public function __construct()
5.	{
6.	require 'koneksiMVC.php';
7.	}
8.	public function getAllProgramKerja()

9.	{
10.	\$rs = \$this->mysql->query("SELECT * FROM proker");
11.	while (\$row = \$rs->fetch_assoc()) {
12.	\$this->hasil[] = \$row;
13.	}
14.	return \$this->hasil;
15.	}
16.	}
17.	\$ad = new ambilData();
18.	\$proker = \$ad->getSemuaProgramKerja();
19.	?>
20.	<html>
21.	
22.	<head>
23.	<title> Daftar Proker BEM </title>
24.	</head>
25.	
26.	<body>
27.	<h2>Daftar Program Kerja BEM </h2>
28.	<h1></h1>
29.	<table border="1">
30.	<tbody>
31.	<tr>
32.	<td>No</td>
33.	<td>Nama Program Kerja</td>
34.	<td>Surat Keterangan</td>
35.	</tr>
36.	</tbody>
37.	<?php
38.	foreach (\$proker as \$row) {
39.	echo " <tr>
40.	<td> \$row[nomorProgram]</td>
41.	<td> \$row[namaProgram]</td>
42.	<td> \$row[suratKeterangan]</td>
43.	</tr> ";
44.	}
45.	?>
46.	</table>
47.	
48.	<form action="login.php" method="post">
49.	<input type="submit" value="Logout">
50.	</form>
51.	</body>
52.	
53.	</html>

v_programKerja.php	
1.	<html>
2.	
3.	<head>
4.	<title>List Proker BEM</title>
5.	</head>
6.	

7.	<body>
8.	<h2>Daftar Program Kerja BEM</h2>
9.	<h3><input type="button" value="Tambah Proker" onclick="window.location.href='?i=1'"></h3>
10.	<table border="1" cellpadding="10" cellspacing="0">
11.	<tbody>
12.	<tr>
13.	<th>No.</th>
14.	<th>Nama Program Kerja</th>
15.	<th>Surat Keterangan</th>
16.	<th colspan="2">Aksi</th>
17.	</tr>
18.	</tbody>
19.	<?php
20.	foreach (\$proker as \$row) {
21.	echo " <tr>
22.	<td> \$row[nomorProgram]</td>
23.	<td> \$row[namaProgram]</td>
24.	<td> \$row[suratKeterangan]</td>
25.	<td>Edit
26.	</td>
27.	<td>Hapus
28.	</td>
29.	</tr> ";
30.	}
31.	?>
32.	</table>
33.	
34.	<form action="login.php" method="post">
35.	<input type="submit" value="Logout">
36.	</form>
37.	</body>
38.	
39.	</html>

v_editProgramKerja.php	
1.	<html>
2.	
3.	<head>
4.	<title>Edit Proker BEM</title>
5.	</head>
6.	
7.	<body>
8.	<table>
9.	<tbody>
10.	<tr>
11.	<td>
12.	<h2> Edit Program Kerja BEM
13.	</h2>
14.	<form method="post" action="#">
15.	<input type="hidden" name="nomor" value="<?php echo \$proker['nomorProgram'] ?>">

16.	<table>
17.	<tr>
18.	<td><label>Nama
19.	Program
20.	Kerja</label></td>
21.	<td><input
22.	type="text" name="nama" value="<?php echo
23.	\$proker['namaProgram'] ?>"></td>
24.	</tr>
25.	<tr>
26.	<td><label>Surat
27.	Keterangan</label></td>
28.	<td><input
29.	type="text" name="surat" value="<?php echo
30.	\$proker['suratKeterangan'] ?>">
31.	</td>
32.	</tr>
33.	</table>
34.	<input type="submit"
35.	name="edit" value="Submit">
36.	</form>
37.	</td>
38.	</tr>
39.	</table>
40.	</body>
41.	</html>
42.	<?php
43.	if (isset(\$_POST['edit'])) {
44.	\$control = new c_programKerja();
	\$control->update();
	}
	?>

v_tambahProgramKerja.php	
1.	<html>
2.	
3.	<head>
4.	<title>Tambah Program Kerja BEM</title>
5.	</head>
6.	
7.	<body>
8.	<table>
9.	<tbody>
10.	<tr>
11.	<td>
12.	<h2> Tambah Program Kerja BEM
13.	</h2>
14.	<form method="post" action="">
15.	<table>
	<tr>

16.		<td><label>Nomor
17.		Program
18.	Kerja</label></td>	<td><input
19.		type="text" name="nomor" value=""></td>
20.		</tr>
21.		<tr>
22.	Program	<td><label>Nama
23.	Kerja</label></td>	<td><input
24.		type="text" name="nama" value="">
25.		</tr>
26.		<tr>
27.		<td><label>Surat
28.	Keterangan</label></td>	<td><input
29.		type="text" name="surat" value=""></td>
30.		</tr>
31.		</table>
32.		<input type="submit"
33.		name="ins" value="Tambah">
34.		</form>
35.		</td>
36.		</tr>
37.		</table>
38.	</body>	
39.	</html>	
40.	<?php	
41.	if (isset(\$_POST['ins'])) {	
42.	\$control = new c_programKerja();	
43.	\$control->insert();	
44.	}	
	?>	

D. Penjelasan

Setelah mengisi NIM dan password serta menekan tombol login, pengguna akan diarahkan ke file admin.php untuk memeriksa apakah akun yang digunakan telah terdaftar. Jika tidak terdaftar, pengguna akan kembali ke halaman login.php. Namun, jika akun sudah terdaftar, data pengguna akan dimasukkan ke dalam file pengurusBEM.php. Selanjutnya, fungsi pilih dalam file tersebut akan menentukan tampilan selanjutnya. Jika akun yang dimasukkan adalah kepala departemen, maka akan dipanggil file index.php dan kemudian mengarahkan ke controller file c_programKerja.php, yang akan mengambil data dari database melalui file m_programKerja.php dan menampilkannya di file v_programKerja.php. Jika jabatan yang dimasukkan adalah menteri, maka akan dipanggil file menteri.php dan menampilkan program kerja yang ada. Ketika tombol "Tambah Proker" ditekan, akan memanggil file v_tambahProgramKerja.php untuk menambahkan program kerja baru. Jika tombol "Hapus" ditekan, akan memanggil file index.php dan kemudian memanggil fungsi delete dalam file c_programKerja.php untuk menghapus data. Jika tombol "Edit" ditekan, akan memanggil file v_editProgramKerja.php untuk

mengedit program kerja yang ada, kemudian tekan tombol "Submit" untuk menyimpannya ke dalam database.