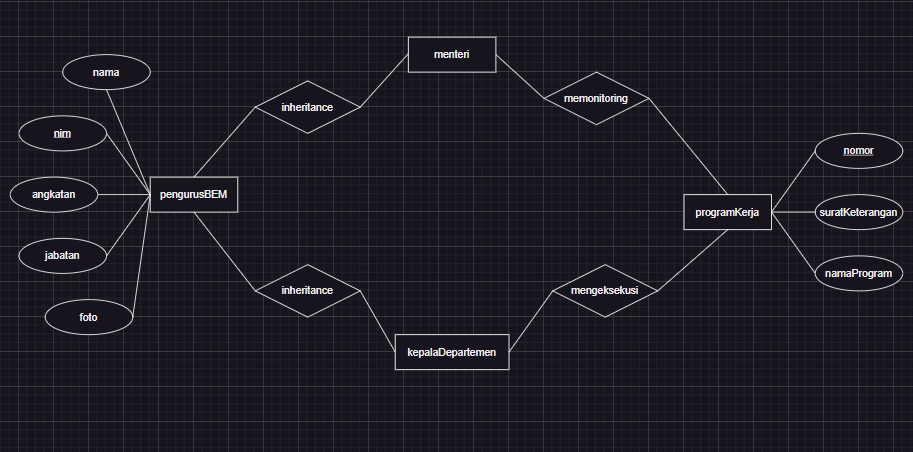
**LATIHAN 1**

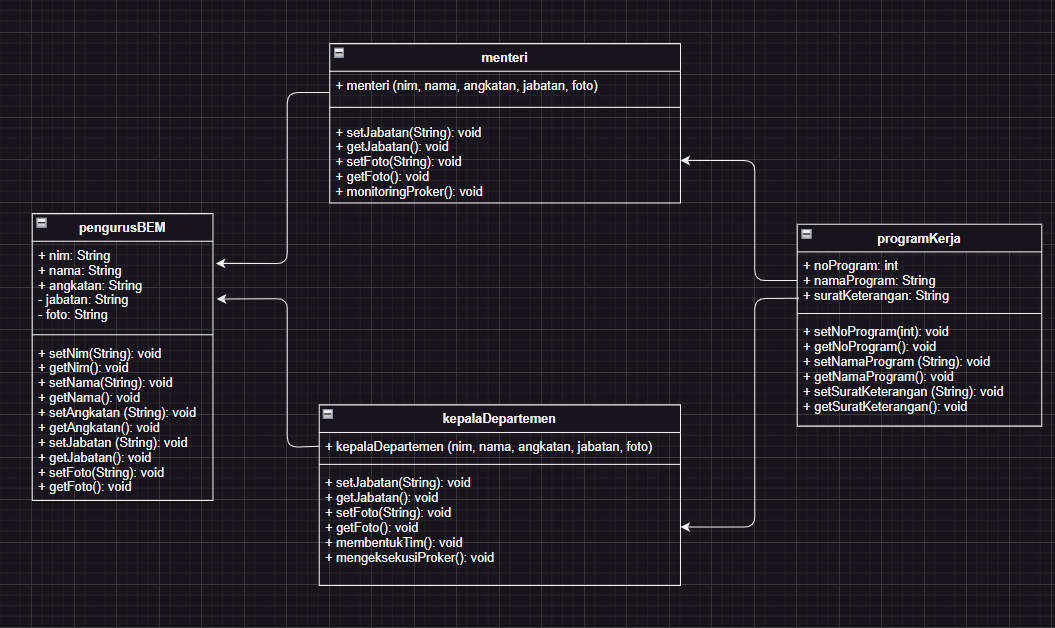
1. **Soal**

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

1. **Screenshoot**

Entity Relationship Diagram  


Class Diagram



1. **Syntax**
2. **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**

1. **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 | |
|  | <?php |
|  | class pengurusBEM |
|  | { |
|  | public $nama; |
|  | public $nim; |
|  | public $angkatan; |
|  | private $jabatan; |
|  | private $foto; |
|  | public function setNama($nama) |
|  | { |
|  | $this->nama = $nama; |
|  | } |
|  | public function getNama() |
|  | { |
|  | echo "$this->nama"; |
|  | } |
|  | private function setJabatan($jabatan) |
|  | { |
|  | $this->jabatan = $jabatan; |
|  | } |
|  | public function getJabatan() |
|  | { |
|  | echo "$this->jabatan"; |
|  | } |
|  | } |

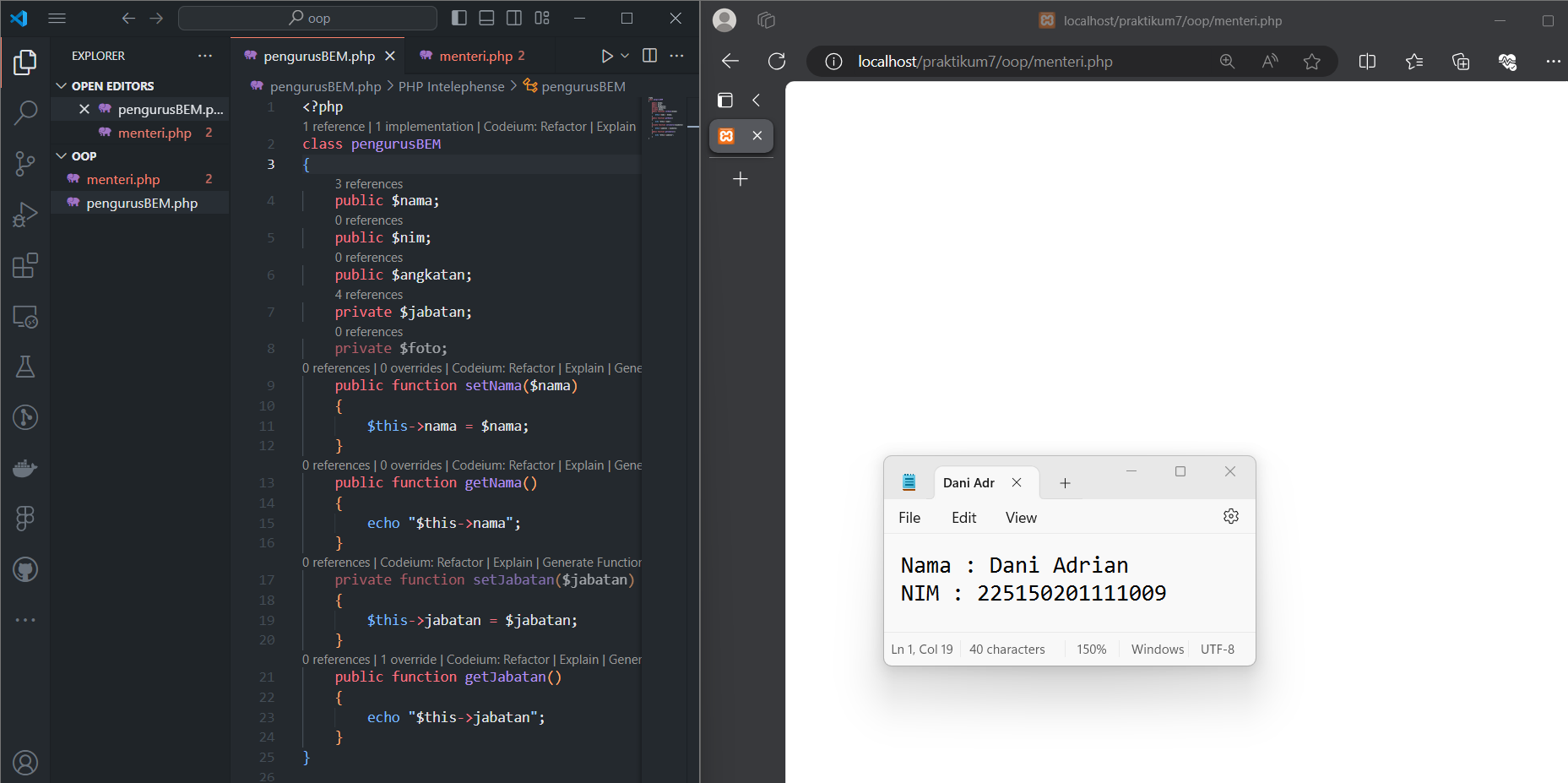
Tulis kode di bawah ini dan simpan dalam file menteri.php

|  |  |
| --- | --- |
| menteri.php | |
|  | <?php |
|  | include "pengurusBEM.php"; |
|  |  |
|  | class menteri extends pengurusBEM { |
|  | public function \_\_construct($nama) { |
|  | $this->nama = $nama; |
|  | } |
|  |  |
|  | function setJabatan($jabatan) { // method |
|  | $this->jabatan = $jabatan; |
|  | } |
|  |  |
|  | public function getJabatan() { |
|  | echo "$this->jabatan"; |
|  | } |
|  | } |

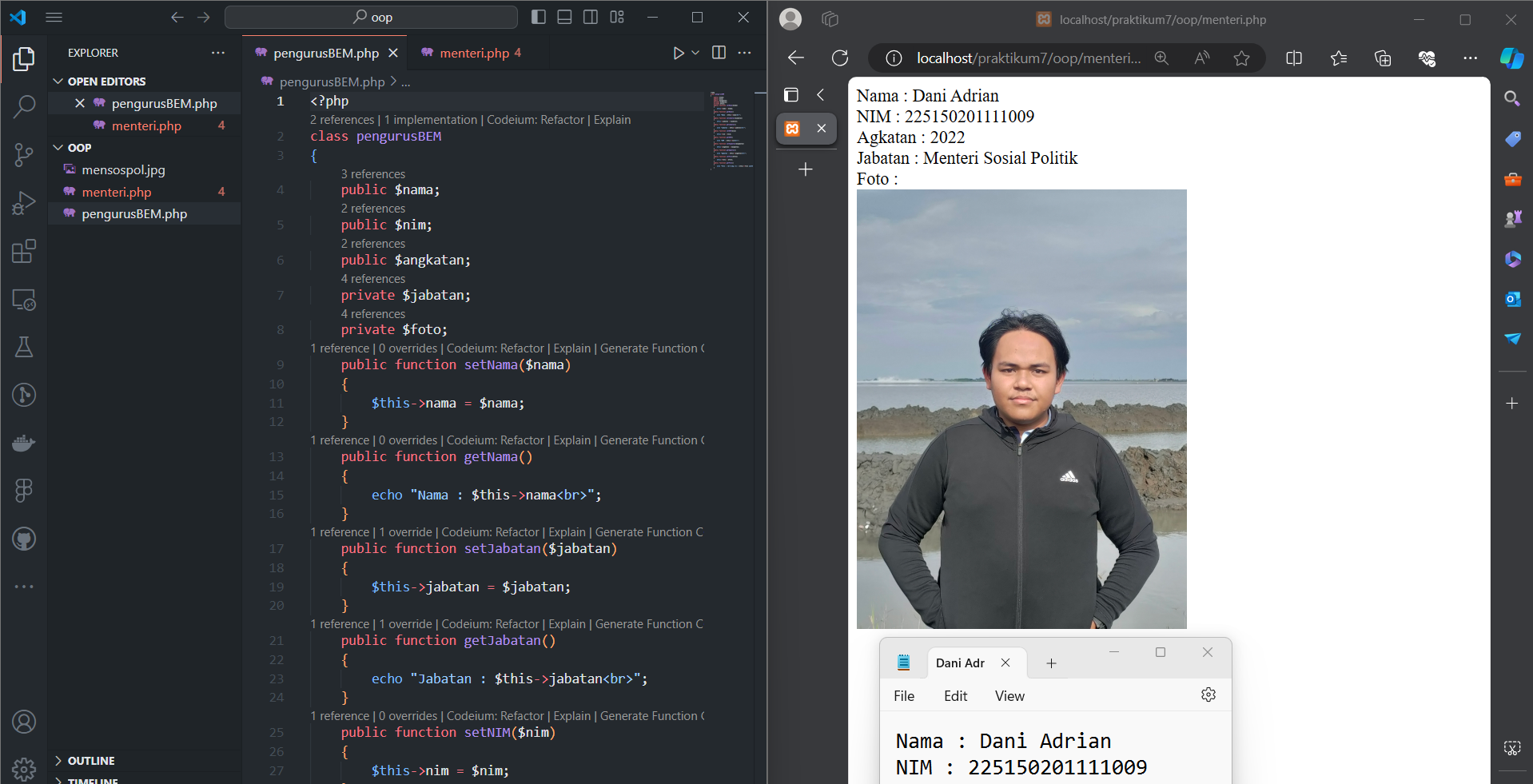
* 1. Apa maksud dari penulisan public dan private pada class pengurusBEM?
  2. Modifikasilah kode diatas sehingga kita dapat melakukan manipulasi attribut dari pengurus BEM dengan lengkap.
  3. Apa hubungan antara method setJabatan pada class pengurusBEM dan class Menteri? Apakah method yang sama karena konsep turunan?
  4. Apa yang terjadi saat dalam pembuatan class kita tidak menuliskan tingkat *visibility*? Apakah method atau atribute dengan semua tingkat *visibility* itu diturunkan?

1. **Screenshoot**

Sebelum Diubah



Setelah Diubah



1. **Syntax**

|  |  |
| --- | --- |
| pengurusBEM.php | |
|  | <?php |
|  | class pengurusBEM |
|  | { |
|  | public $nama; |
|  | public $nim; |
|  | public $angkatan; |
|  | private $jabatan; |
|  | private $foto; |
|  | public function setNama($nama) |
|  | { |
|  | $this->nama = $nama; |
|  | } |
|  | public function getNama() |
|  | { |
|  | echo "Nama : $this->nama<br>"; |
|  | } |
|  | public function setJabatan($jabatan) |
|  | { |
|  | $this->jabatan = $jabatan; |
|  | } |
|  | public function getJabatan() |
|  | { |
|  | echo "Jabatan : $this->jabatan<br>"; |
|  | } |
|  | public function setNIM($nim) |
|  | { |
|  | $this->nim = $nim; |
|  | } |
|  | public function getNIM() |
|  | { |
|  | echo "NIM : $this->nim<br>"; |
|  | } |
|  | public function setAngkatan($angkatan) |
|  | { |
|  | $this->angkatan = $angkatan; |
|  | } |
|  | public function getAgkatan() |
|  | { |
|  | echo "Agkatan : $this->angkatan<br>"; |
|  | } |
|  | public function setFoto($foto) |
|  | { |
|  | $this->foto = $foto; |
|  | } |
|  | public function getFoto() |
|  | { |
|  | echo "Foto : <br><img src = $this->foto width='300'>"; |
|  | } |
|  | } |

|  |  |
| --- | --- |
| menteri.php | |
|  | <?php |
|  | include "pengurusBEM.php"; |
|  | class menteri extends pengurusBEM |
|  | { |
|  | public function \_\_construct($nama) |
|  | { |
|  | $this->nama = $nama; |
|  | } |
|  | function setJabatan($jabatan) |
|  | { // method |
|  | $this->jabatan = $jabatan; |
|  | } |
|  | public function getJabatan() |
|  | { |
|  | echo "$this->jabatan"; |
|  | } |
|  | function setFoto($foto) |
|  | { // method |
|  | $this->foto = $foto; |
|  | } |
|  | public function getFoto() |
|  | { |
|  | echo "$this->foto"; |
|  | } |
|  | } |
|  | $mahasiswa = new pengurusBEM; |
|  | $mahasiswa->setNama("Dani Adrian"); |
|  | $mahasiswa->getNama(); |
|  | $mahasiswa->setNIM("225150201111009"); |
|  | $mahasiswa->getNIM(); |
|  | $mahasiswa->setAngkatan(2022); |
|  | $mahasiswa->getAgkatan(); |
|  | $mahasiswa->setJabatan("Menteri Sosial Politik"); |
|  | $mahasiswa->getJabatan(); |
|  | $mahasiswa->setFoto("mensospol.jpg"); |
|  | $mahasiswa->getFoto(); |

1. **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**

1. **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 | |
|  | <?php |
|  | class koneksiMVC |
|  | { |
|  | public $mysqli; |
|  | public function \_\_construct() |
|  | { |
|  | $this->mysqli = mysqli\_connect("localhost", "root", "", "praktikum12"); |
|  | } |
|  | } |

Tulis kode dibawah ini dan simpan dalam file “m\_programKerja.php”

|  |  |
| --- | --- |
| m\_programKerja.php | |
|  | <?php |
|  | require "./koneksiMVC.php"; |
|  | class m\_programKerja |
|  | { |
|  | private $database; |
|  | protected $tablename = "program\_kerja"; |
|  | public function \_\_construct() |
|  | { |
|  | $this->database = new koneksiMVC(); |
|  | $this->database = $this->database->mysqli; |
|  | } |
|  | // Add Program Kerja |
|  | public function setProgramKerja($nomorProgram, $namaProgram, $suratKeterangan) |
|  | { |
|  | $this->database->query("INSERT INTO $this->tablename (nomorProgram, namaProgram, suratKeterangan) VALUES ($nomorProgram, '$namaProgram', '$suratKeterangan')"); |
|  | } |
|  | // Get All ProgramKerja |
|  | public function getAll() |
|  | { |
|  | return $this->database->query("SELECT \* FROM $this->tablename"); |
|  | } |
|  | } |

Tulis kode di bawah ini dan simpan dalam file “c\_programKerja.php”

|  |  |
| --- | --- |
| c\_programKerja.php | |
|  | <?php |
|  | include\_once("m\_programKerja"); |
|  |  |
|  | class c\_programKerja { |
|  | public $model; |
|  |  |
|  | public function \_\_construct($nama) { |
|  | $this->model = new m\_programKerja(); |
|  | } |
|  |  |
|  | public function invoke() { |
|  | $proker = $this->model->getAll(); |
|  | include 'v\_programKerja.php'; |
|  | } |
|  | } |

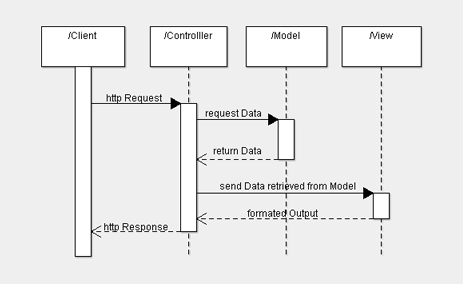
Tulis kode di bawah ini dan simpan dalam file “v\_programKerja.php”

|  |  |
| --- | --- |
| v\_programKerja.php | |
|  | <html> |
|  | <head></head> |
|  | <body> |
|  | <h2>Daftar Program Kerja BEM</h2> |
|  | <table> |
|  | <tbody> |
|  | <tr> |
|  | <td>No</td> |
|  | <td>Nama Program Kerja</td> |
|  | <td>Surat Keterangan</td> |
|  | </tr> |
|  | </tbody> |
|  | <?php |
|  |  |
|  | echo '<table><tr>', |
|  | '<td>$proker[nomorProgram]</td>', |
|  | '<td>$proker[namaProgram]</td>', |
|  | '<td>$proker[suratKeterangan]</td>', |
|  | '</tr></table>'; |
|  | ?> |
|  | </body> |
|  | </html> |

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

|  |  |
| --- | --- |
| index.php | |
|  | <?php |
|  | include\_once("c\_programKerja.php"); |
|  |  |
|  | $controller = new c\_programKerja(); |
|  | $controller->invoke(); |

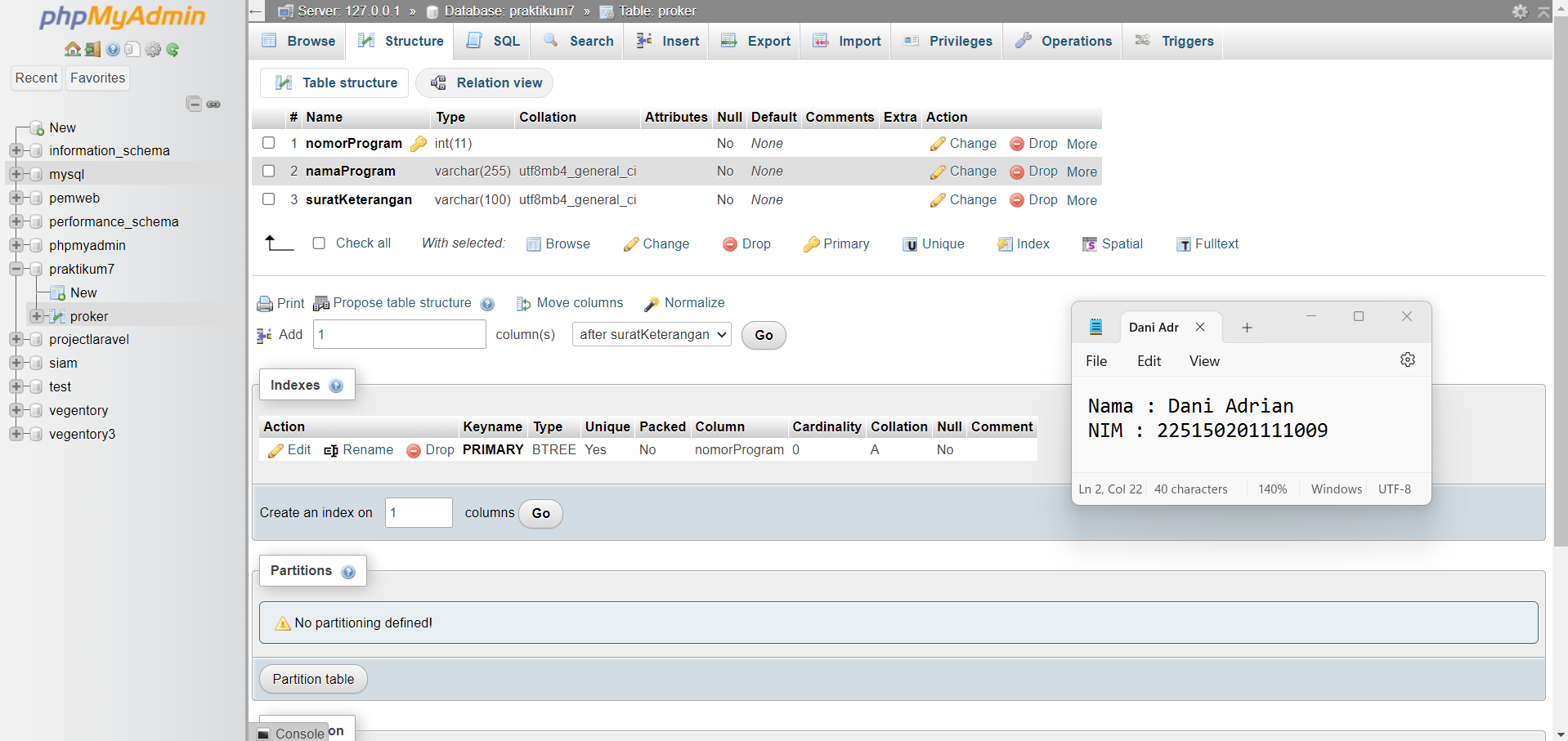
* 1. Jalankan file index.php dan amati yang terjadi. Apa hubungannya dengan *sequence diagram* di bawah ini?

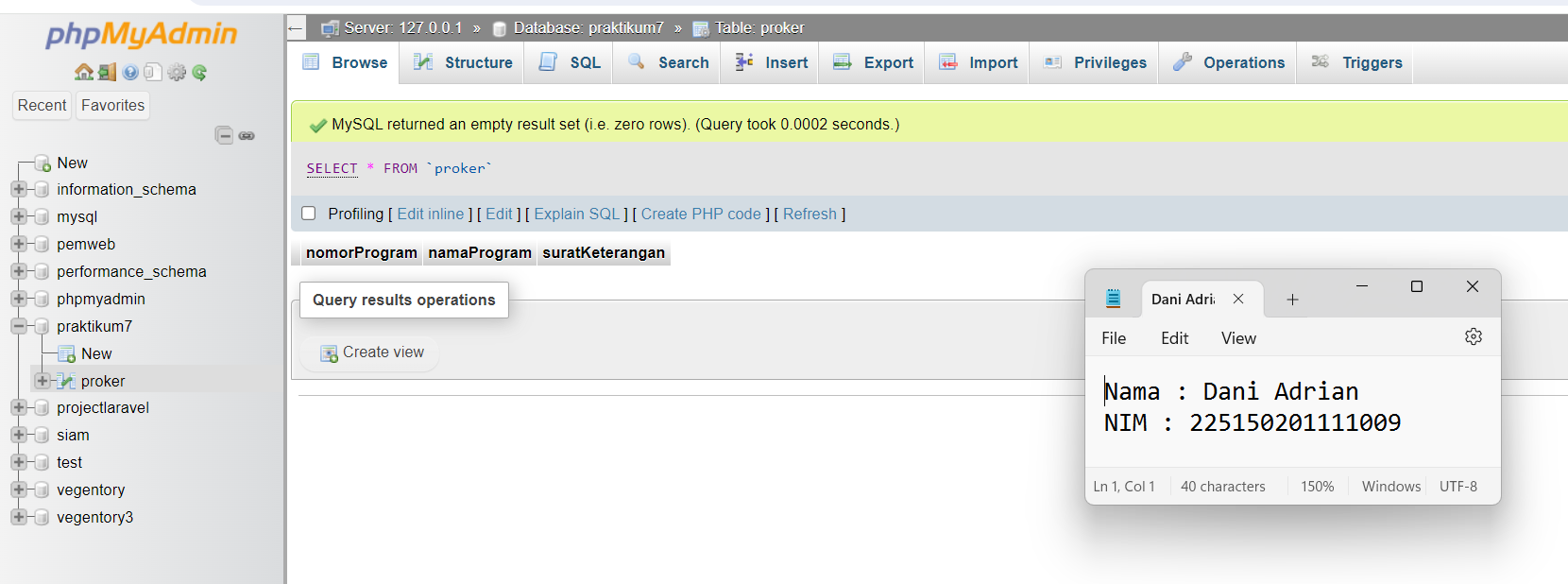


* 1. Apa maksud dari pembuatan kode “m\_programKerja.php”?
  2. Apa maksud dari pembuatan kode “c\_programKerja.php”?
  3. Apa maksud dari pembuatan kode “v\_programKerja.php”?
  4. Apa maksud dari pembuatan kode “index.php”?

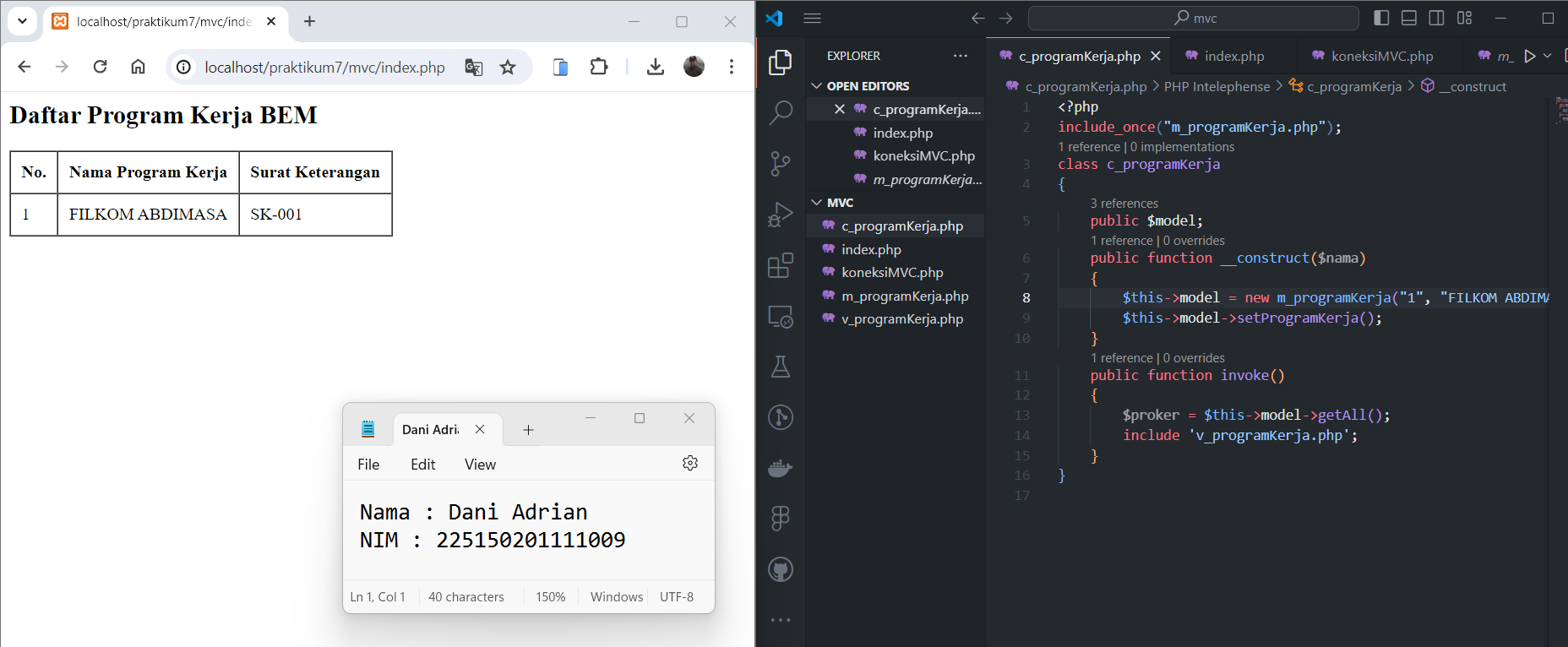
1. **Screenshoot**

Database sebelum:

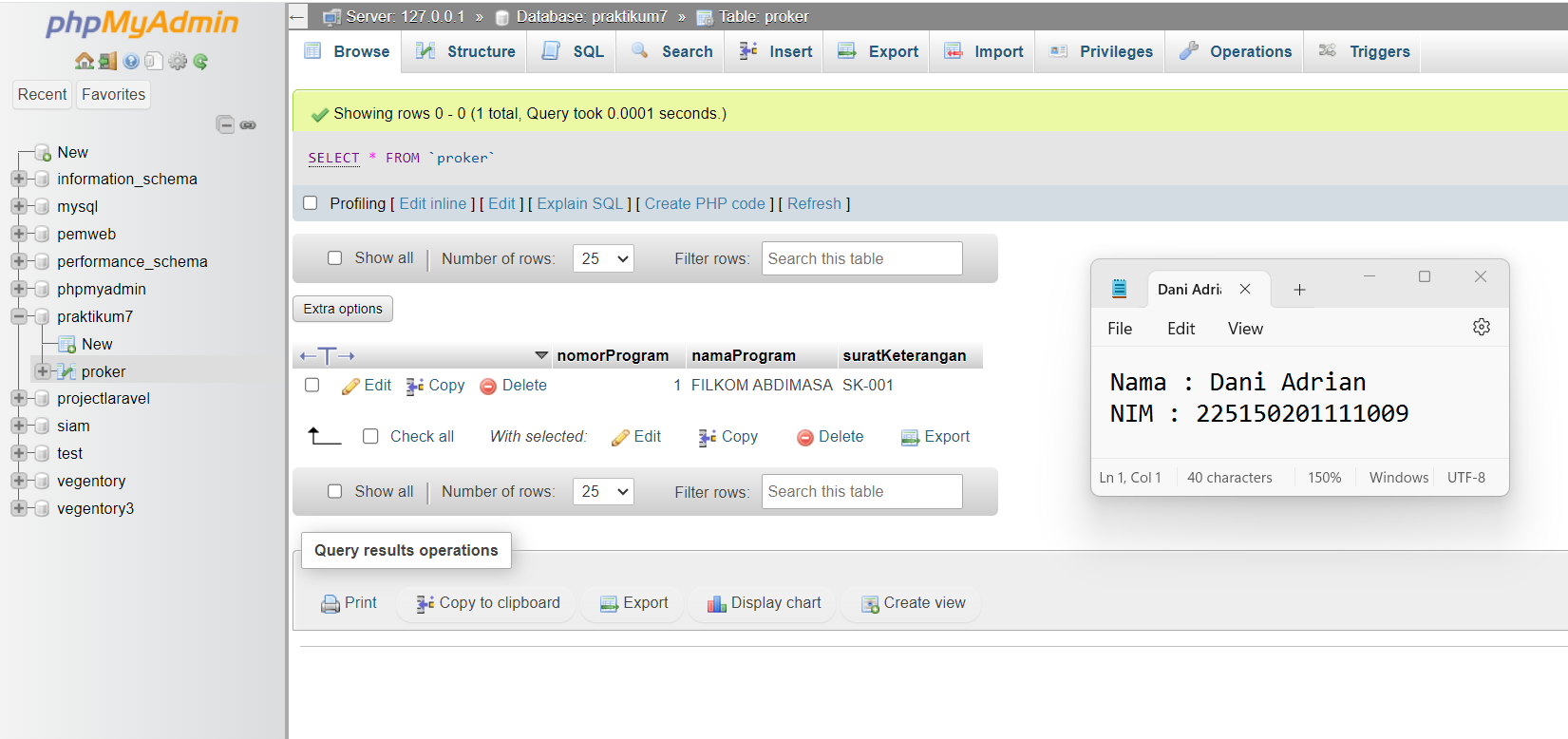




index.php



Database sesudah :



1. **Syntax**

|  |  |
| --- | --- |
| koneksiMVC.php | |
|  | <?php |
|  | $mysqli = mysqli\_connect('localhost', 'root', '02030507', 'praktikum7'); |

|  |  |
| --- | --- |
| m\_programKerja.php | |
|  | <?php |
|  | class m\_programKerja |
|  | { |
|  | private $nomorProgram; |
|  | private $namaProgram; |
|  | private $suratKeterangan; |
|  | public $hasil = array(); |
|  | public function \_\_construct($nomorProgram, $namaProgram, $suratKeterangan) |
|  | { |
|  | $this->nomorProgram = $nomorProgram; |
|  | $this->namaProgram = $namaProgram; |
|  | $this->suratKeterangan = $suratKeterangan; |
|  | } |
|  | // Add Program Kerja |
|  | public function setProgramKerja() |
|  | { |
|  | require "koneksiMVC.php"; |
|  | $rs = $mysqli->query("INSERT INTO proker VALUES ('$this->nomorProgram','$this->namaProgram','$this->suratKeterangan')"); |
|  | } |
|  | // Get All ProgramKerja |
|  | public function getAll() |
|  | { |
|  | require "koneksiMVC.php"; |
|  | $sql = "SELECT \* FROM proker"; |
|  | $rs = $mysqli->query($sql); |
|  | $rows = array(); |
|  | while ($row = $rs->fetch\_assoc()) { |
|  | $rows[] = $row; |
|  | } |
|  | $this->hasil = $rows; |
|  | return $this->hasil; |
|  | } |
|  | } |

|  |  |
| --- | --- |
| c\_programKerja.php | |
|  | <?php |
|  | include\_once("m\_programKerja.php"); |
|  | class c\_programKerja |
|  | { |
|  | public $model; |
|  | public function \_\_construct($nama) |
|  | { |
|  | $this->model = new m\_programKerja("1", "FILKOM ABDIMASA", "SK-001"); |
|  | $this->model->setProgramKerja(); |
|  | } |
|  | public function invoke() |
|  | { |
|  | $proker = $this->model->getAll(); |
|  | include 'v\_programKerja.php'; |
|  | } |
|  | } |

|  |  |
| --- | --- |
| v\_programKerja.php | |
|  | <html> |
|  |  |
|  | <head></head> |
|  |  |
|  | <body> |
|  | <h2>Daftar Program Kerja BEM</h2> |
|  | <table border="1" cellpadding='10' cellspacing='0'> |
|  | <tbody> |
|  | <tr> |
|  | <th>No.</th> |
|  | <th>Nama Program Kerja</th> |
|  | <th>Surat Keterangan</th> |
|  | </tr> |
|  | </tbody> |
|  | <?php |
|  | foreach ($proker as $row) { |
|  | echo "<tr> |
|  | <td> $row[nomorProgram]</td> |
|  | <td> $row[namaProgram]</td> |
|  | <td> $row[suratKeterangan]</td> |
|  | </tr> "; |
|  | } |
|  | ?> |
|  | </table> |
|  | </body> |
|  |  |
|  | </html> |

|  |  |
| --- | --- |
| index.php | |
|  | <?php |
|  | include\_once("c\_programKerja.php"); |
|  | $controller = new c\_programKerja("Program Kerja"); |
|  | $controller->invoke(); |

1. **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.

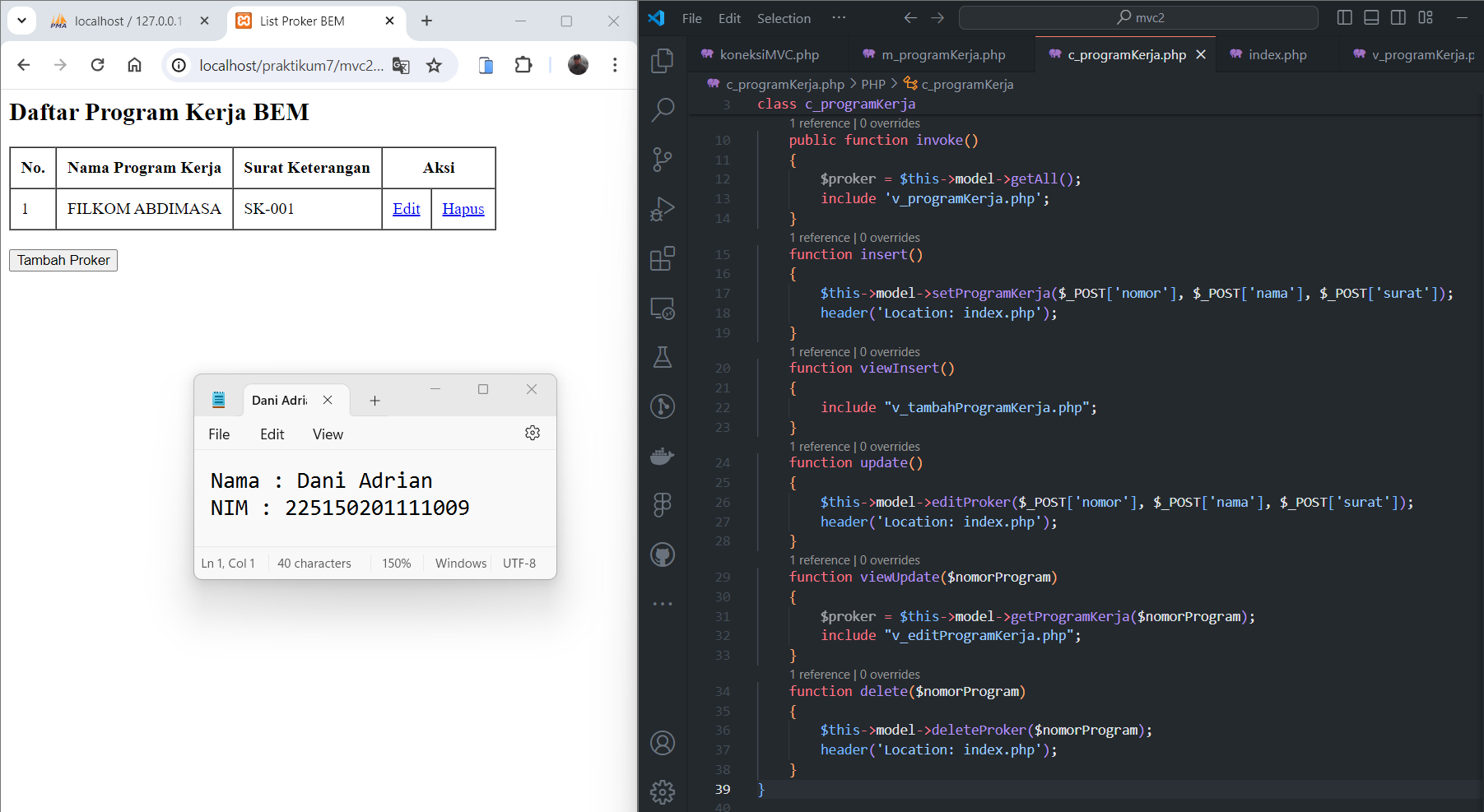
Secara keseluruhan, index.php berperan sebagai file utama yang dieksekusi pertama kali dan bertanggung jawab untuk menampilkan hasil dari permintaan pengguna.

**LATIHAN 4**

1. **Soal**

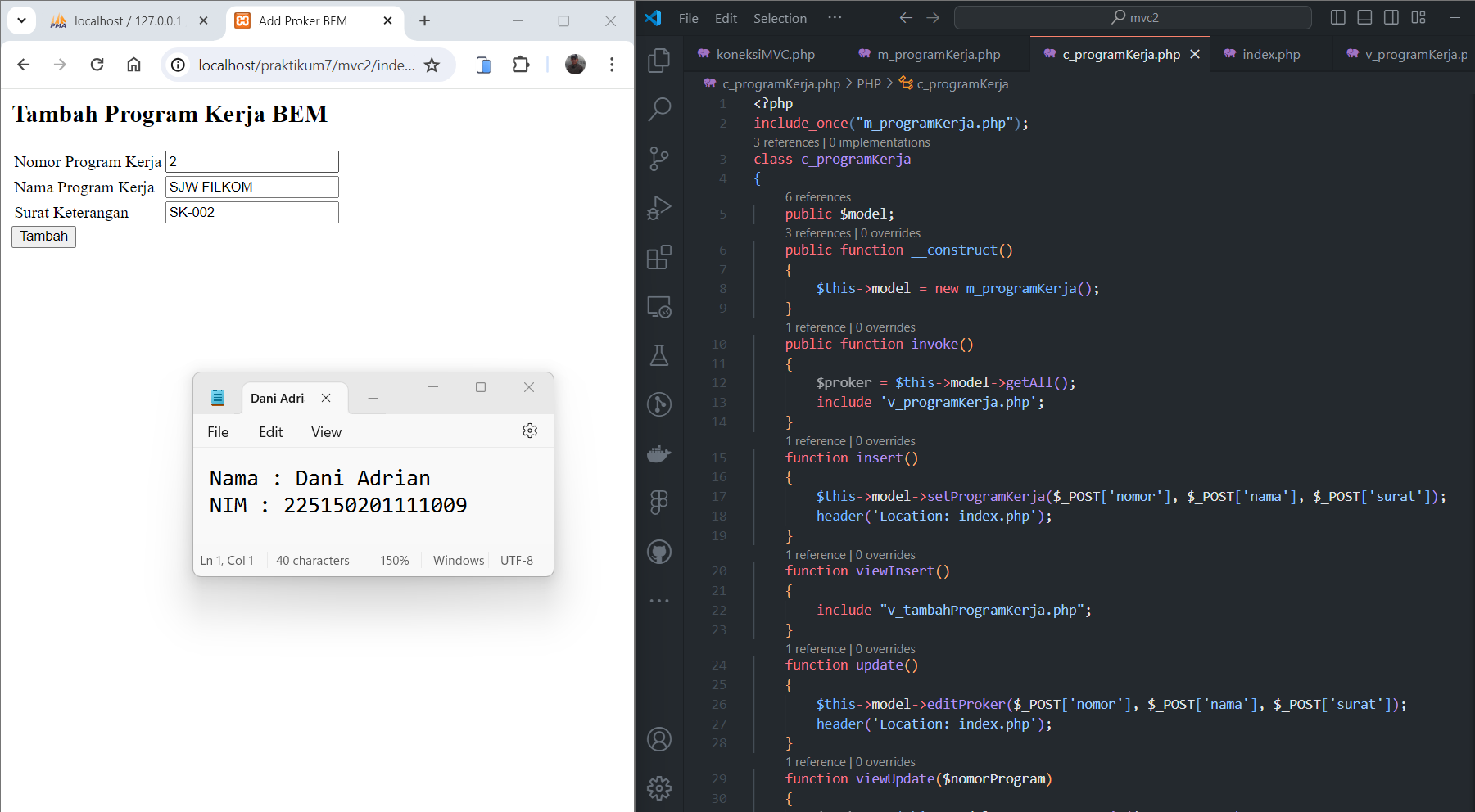
Modifikasilah Latihan 3 sehingga dapat dibuat program dengan kemampuan CRUD secara lengkap!

1. **Screenshoot**

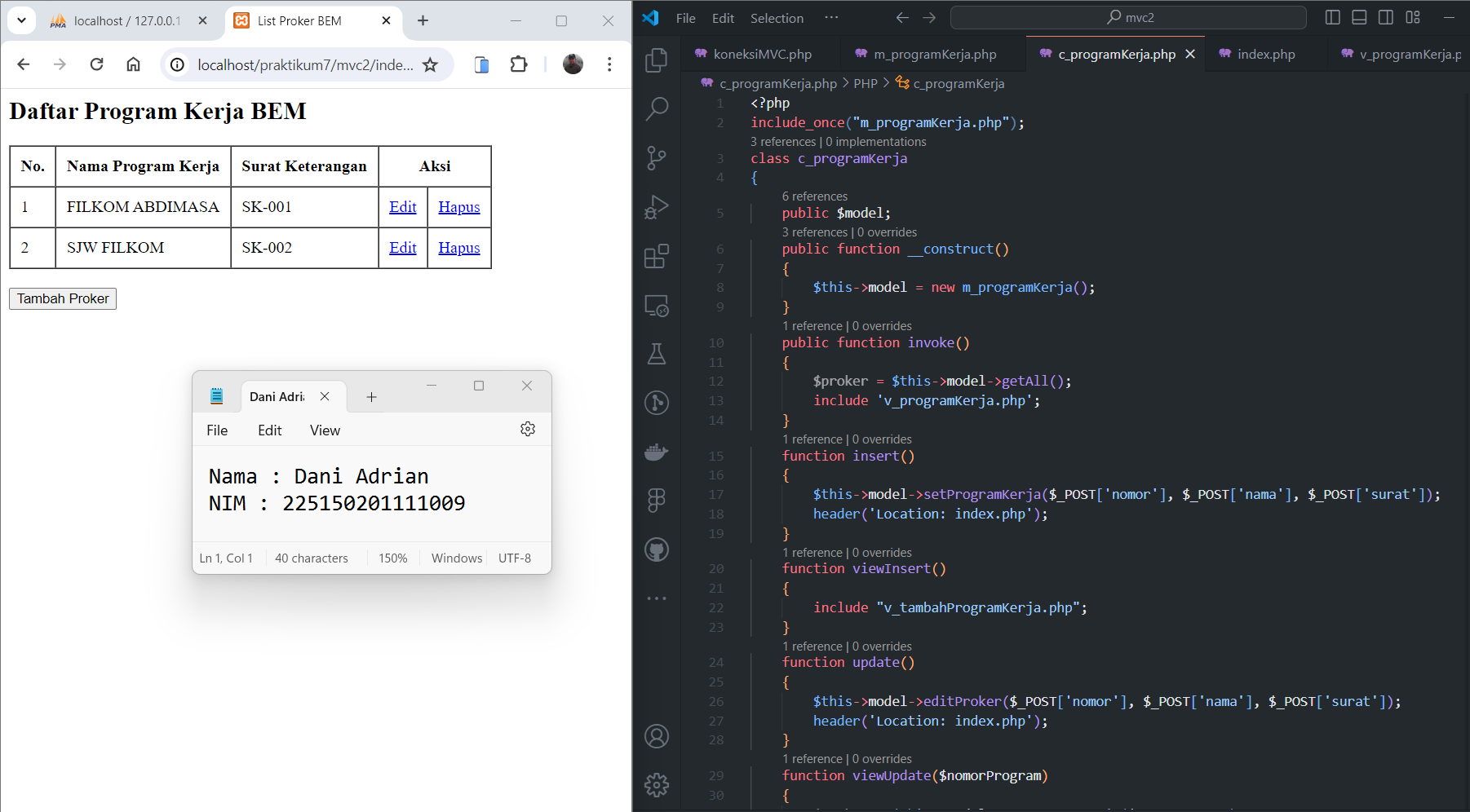


**Form tambah**

* Sebelum ditambah :

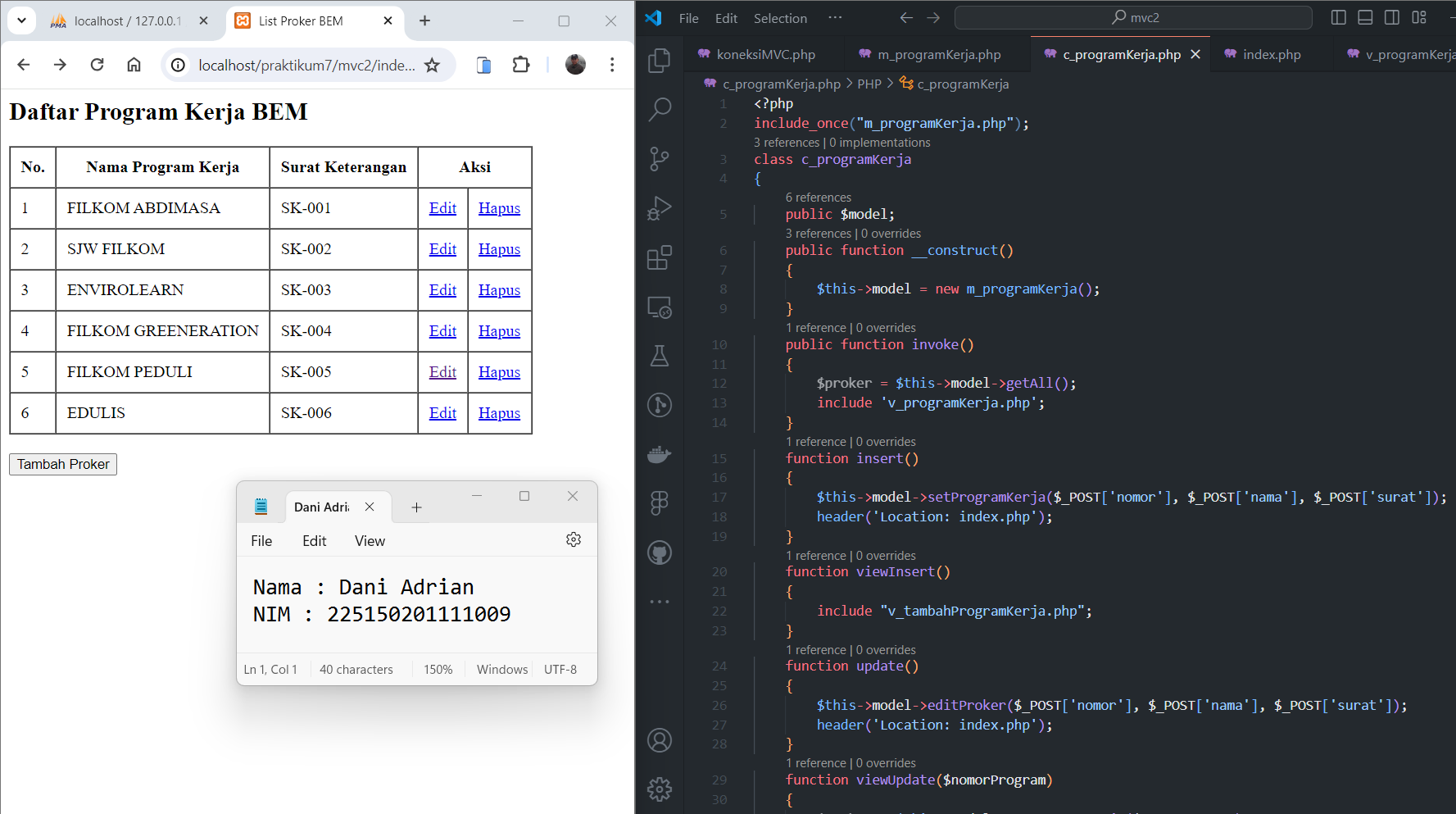


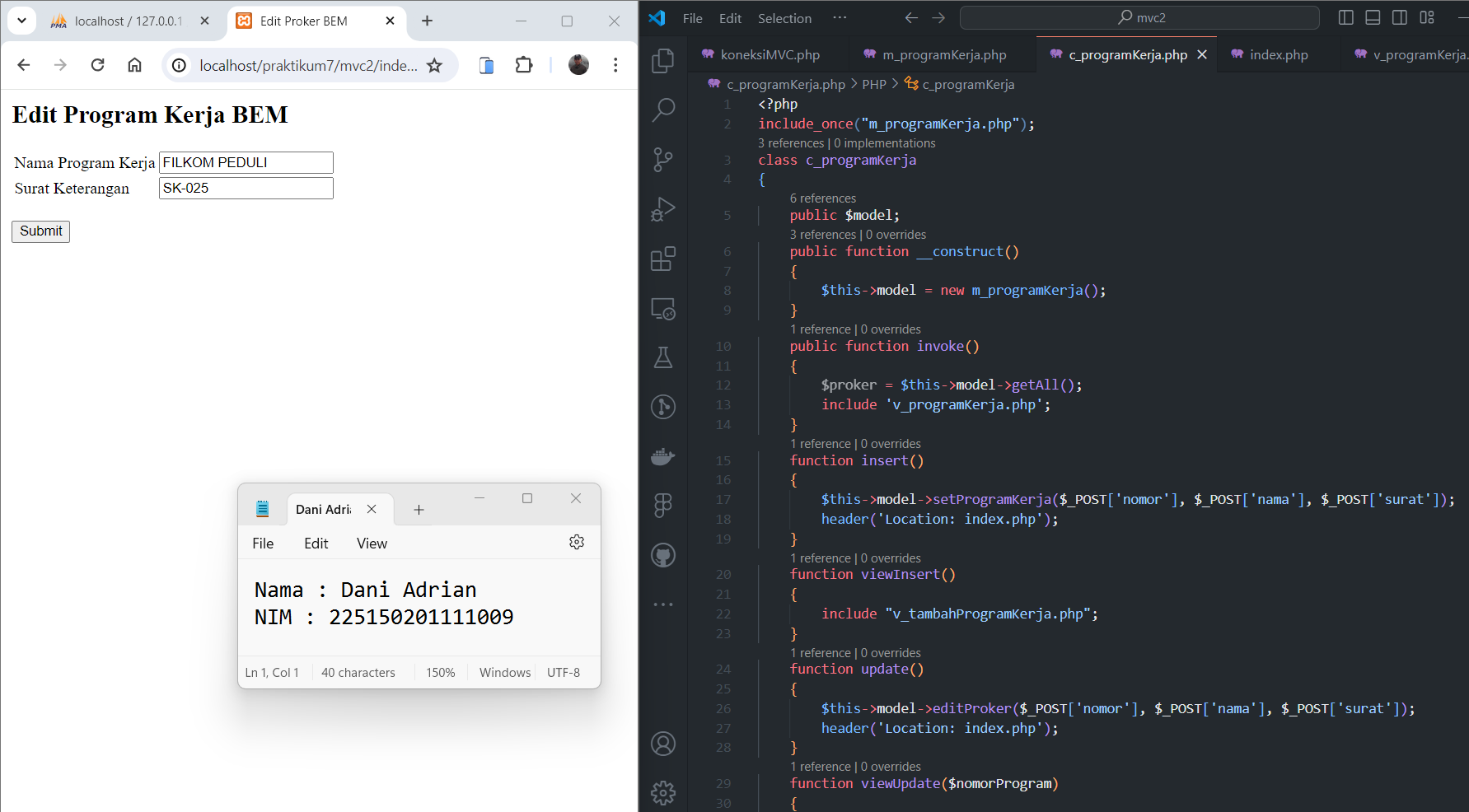
* Sesudah ditambah



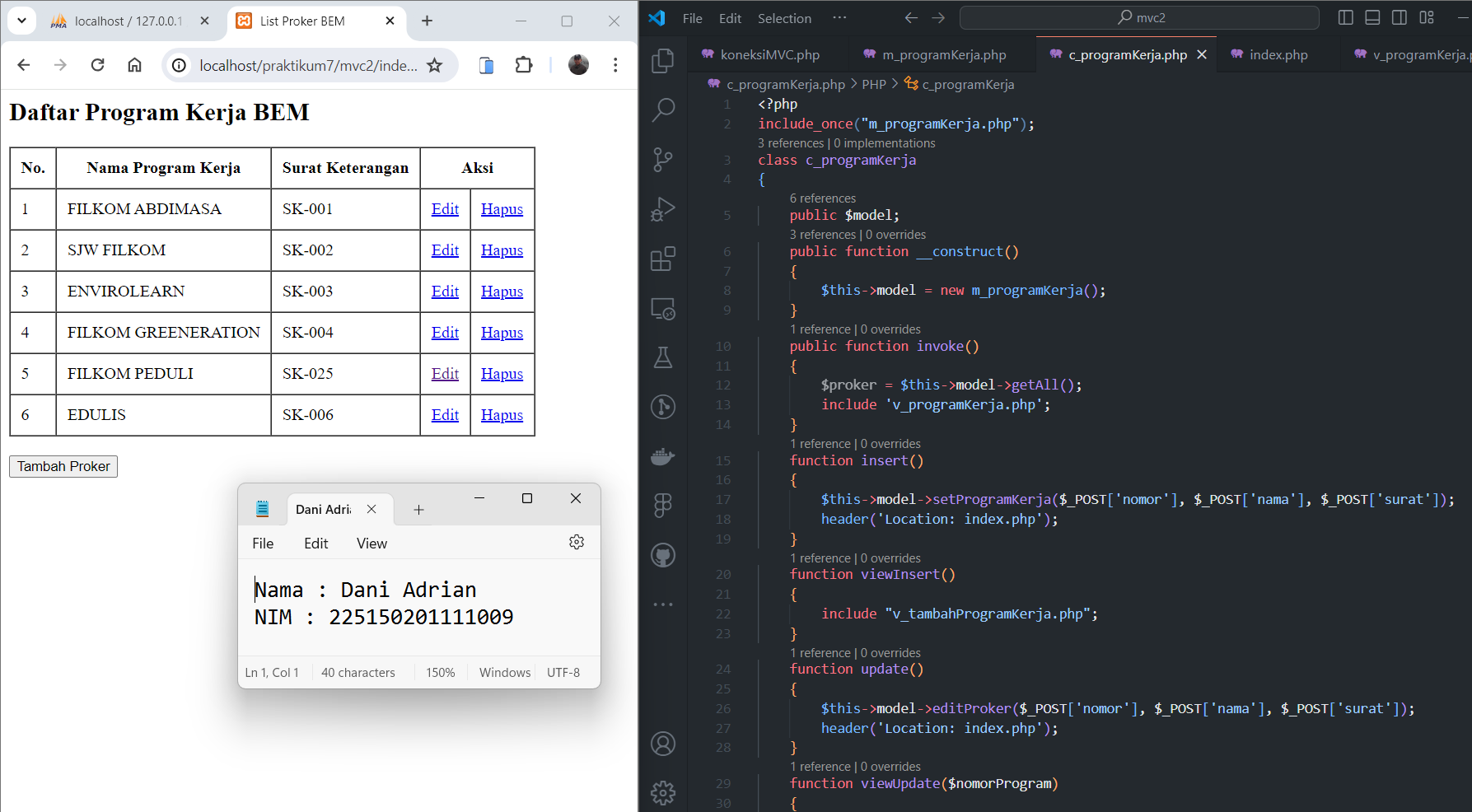
**Form Edit**

* Sebelum diedit



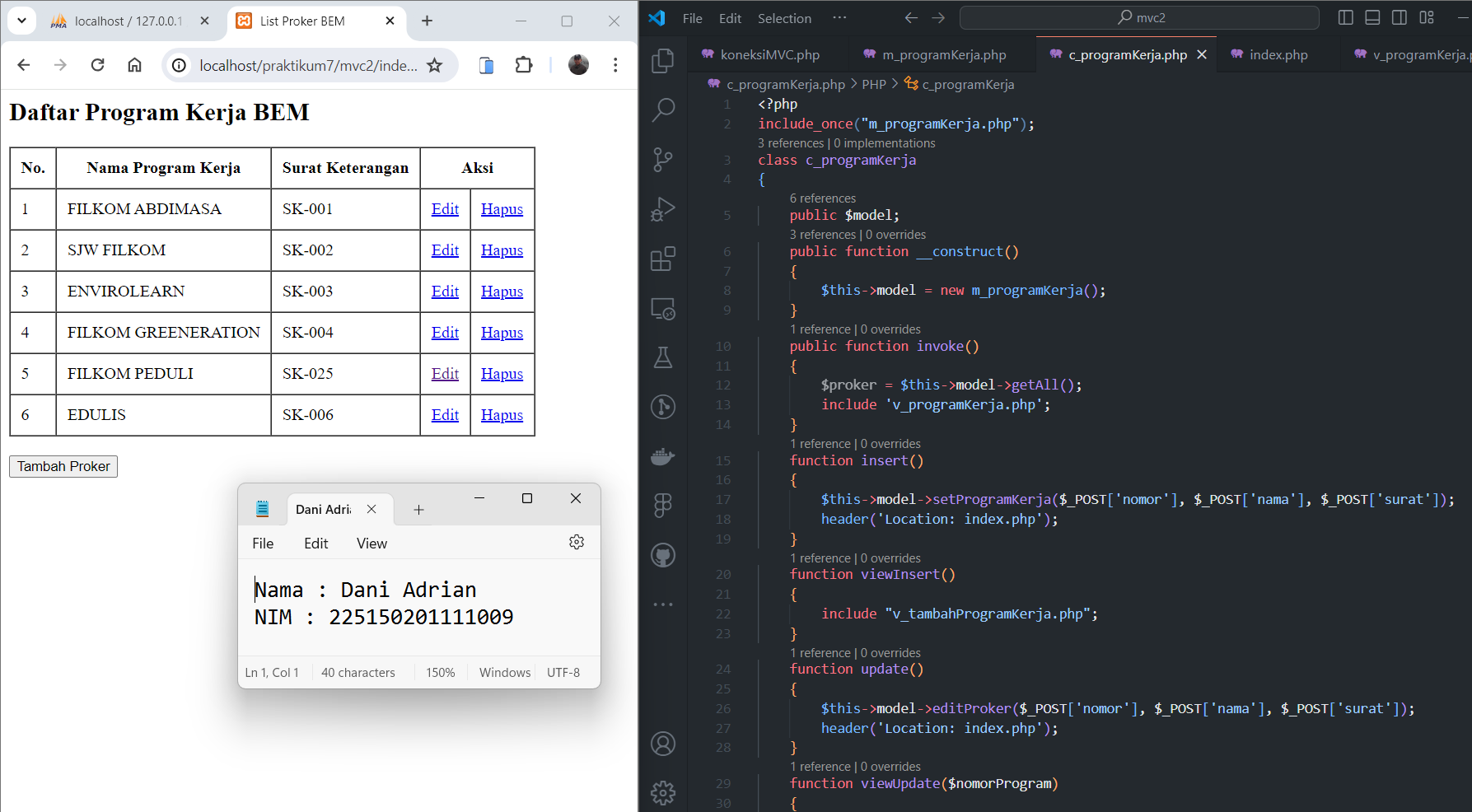


* Sesudah diedit

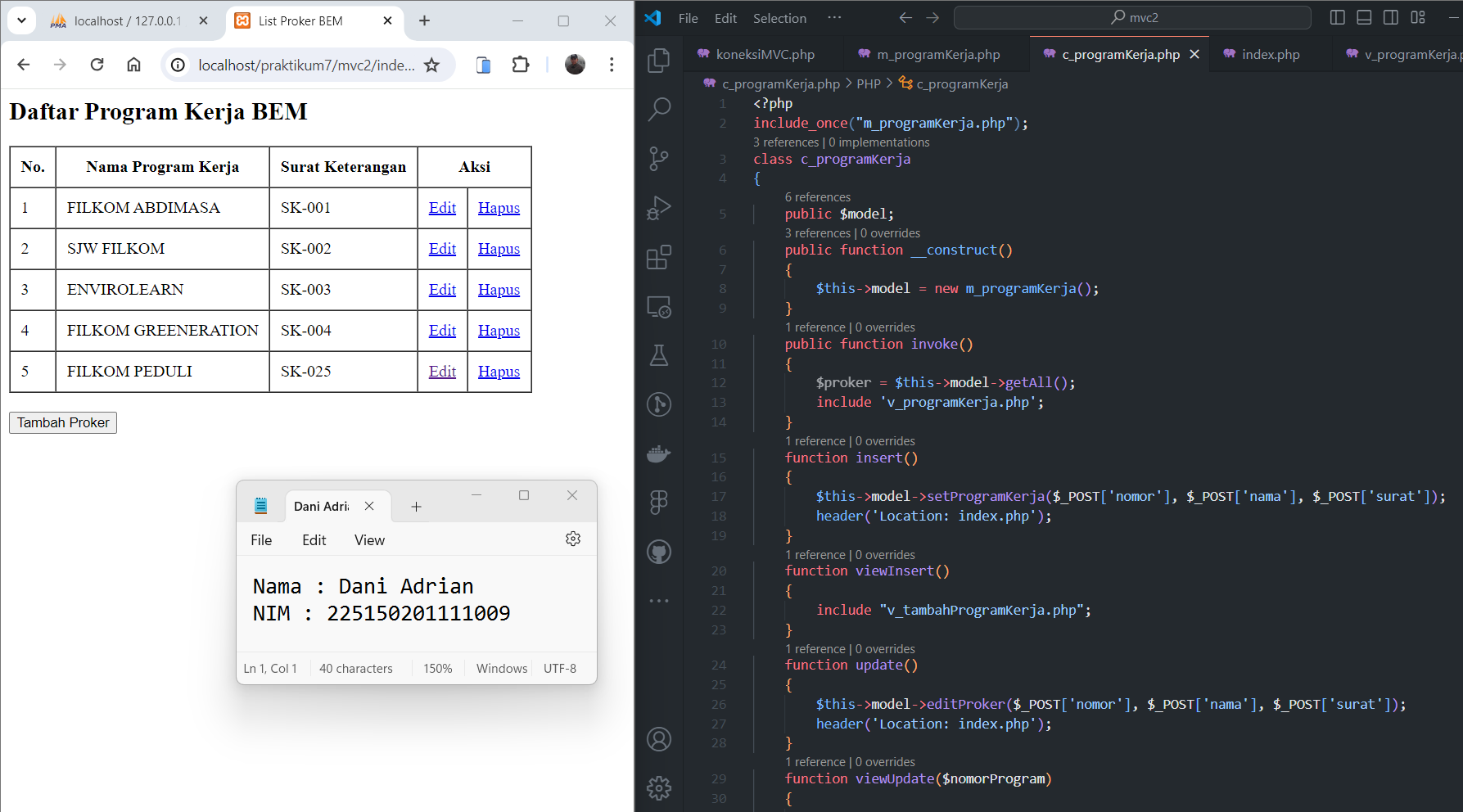


**Hapus**

* Sebelum dihapus



* Sesudah dihapus



1. **Syntax**

|  |  |
| --- | --- |
| koneksiMVC.php | |
|  | <?php |
|  | $mysqli = mysqli\_connect('localhost', 'root', '02030507', 'praktikum7'); |

|  |  |
| --- | --- |
| m\_programKerja.php | |
|  | <?php |
|  | class m\_programKerja |
|  | { |
|  | public $hasil = array(); |
|  | private $mysqli; |
|  | public function \_\_construct() |
|  | { |
|  | require 'koneksiMVC.php'; |
|  | $this->mysqli = $mysqli; |
|  | } |
|  | public function setProgramKerja($nomorProker, $namaProker, $suratKet) |
|  | { |
|  | $rs = $this->mysqli->query("INSERT INTO proker VALUES ('$nomorProker', '$namaProker','$suratKet')"); |
|  | } |
|  | public function getProgramKerja($nomorProker) |
|  | { |
|  | $rs = $this->mysqli->query("SELECT \* FROM proker WHERE nomorProgram = $nomorProker"); |
|  | $this->hasil = $rs->fetch\_array(); |
|  | return $this->hasil; |
|  | } |
|  | public function deleteProker($nomorProker) |
|  | { |
|  | $rs = $this->mysqli->query("DELETE from proker WHERE nomorProgram = $nomorProker"); |
|  | } |
|  | public function editProker($nomorProker, $namaProker, $suratKet) |
|  | { |
|  | $rs = $this->mysqli->query("UPDATE proker SET namaProgram = '$namaProker',suratKeterangan= '$suratKet' WHERE nomorProgram = $nomorProker"); |
|  | } |
|  | public function getAll() |
|  | { |
|  | $rs = $this->mysqli->query("SELECT \* FROM proker"); |
|  | while ($row = $rs->fetch\_assoc()) { |
|  | $this->hasil[] = $row; |
|  | } |
|  | return $this->hasil; |
|  | } |
|  | } |

|  |  |
| --- | --- |
| c\_programKerja.php | |
|  | <?php |
|  | include\_once("m\_programKerja.php"); |
|  | class c\_programKerja |
|  | { |
|  | public $model; |
|  | public function \_\_construct() |
|  | { |
|  | $this->model = new m\_programKerja(); |
|  | } |
|  | public function invoke() |
|  | { |
|  | $proker = $this->model->getAll(); |
|  | include 'v\_programKerja.php'; |
|  | } |
|  | function insert() |
|  | { |
|  | $this->model->setProgramKerja($\_POST['nomor'], $\_POST['nama'], $\_POST['surat']); |
|  | header('Location: index.php'); |
|  | } |
|  | function viewInsert() |
|  | { |
|  | include "v\_tambahProgramKerja.php"; |
|  | } |
|  | function update() |
|  | { |
|  | $this->model->editProker($\_POST['nomor'], $\_POST['nama'], $\_POST['surat']); |
|  | header('Location: index.php'); |
|  | } |
|  | function viewUpdate($nomorProgram) |
|  | { |
|  | $proker = $this->model->getProgramKerja($nomorProgram); |
|  | include "v\_editProgramKerja.php"; |
|  | } |
|  | function delete($nomorProgram) |
|  | { |
|  | $this->model->deleteProker($nomorProgram); |
|  | header('Location: index.php'); |
|  | } |
|  | } |

|  |  |
| --- | --- |
| v\_programKerja.php | |
|  | <html> |
|  |  |
|  | <head> |
|  | <title>List Proker BEM</title> |
|  | </head> |
|  |  |
|  | <body> |
|  | <h2>Daftar Program Kerja BEM</h2> |
|  | <table border="1" cellpadding='10' cellspacing='0'> |
|  | <tbody> |
|  | <tr> |
|  | <th>No.</th> |
|  | <th>Nama Program Kerja</th> |
|  | <th>Surat Keterangan</th> |
|  | <th colspan="2">Aksi</th> |
|  | </tr> |
|  | </tbody> |
|  | <?php |
|  | foreach ($proker as $row) { |
|  | echo " <tr> |
|  | <td> $row[nomorProgram]</td> |
|  | <td> $row[namaProgram]</td> |
|  | <td> $row[suratKeterangan]</td> |
|  | <td><a href='?u=$row[nomorProgram]'>Edit </a></td> |
|  | <td><a href='?d=$row[nomorProgram]'>Hapus </a></td> |
|  | </tr> "; |
|  | } |
|  | ?> |
|  | </table> |
|  | <h3><input type="button" value="Tambah Proker" onclick="window.location.href='?i=1'"></h3> |
|  | </body> |
|  |  |
|  | </html> |

|  |  |
| --- | --- |
| v\_editprogramKerja.php | |
|  | <html> |
|  |  |
|  | <head> |
|  | <title>Edit Proker BEM</title> |
|  | </head> |
|  |  |
|  | <body> |
|  | <table> |
|  | <tbody> |
|  | <tr> |
|  | <td> |
|  | <h2>Edit Program Kerja BEM</h2> |
|  | <form method="post" action="#"> |
|  | <input type="hidden" name="nomor" value="<?php echo $proker['nomorProgram'] ?>"> |
|  | <table> |
|  | <tr> |
|  | <td><label>Nama Program Kerja</label></td> |
|  | <td><input type="text" name="nama" value="<?php echo $proker['namaProgram'] ?>"></td> |
|  | </tr> |
|  | <tr> |
|  | <td><label>Surat Keterangan</label></td> |
|  | <td><input type="text" name="surat" value="<?php echo $proker['suratKeterangan'] ?>"> |
|  | </td> |
|  | </tr> |
|  | </table><br> |
|  | <input type="submit" name="edit" value="Submit"> |
|  | </form> |
|  | </td> |
|  | </tr> |
|  | </table> |
|  | </body> |
|  |  |
|  | </html> |
|  | <?php |
|  | if (isset($\_POST['edit'])) { |
|  | $control = new c\_programKerja(); |
|  | $control->update(); |
|  | } |
|  | ?> |

|  |  |
| --- | --- |
| v\_tambahprogramKerja.php | |
|  | <html> |
|  |  |
|  | <head> |
|  | <title>Add Proker BEM</title> |
|  | </head> |
|  |  |
|  | <body> |
|  | <table> |
|  | <tbody> |
|  | <tr> |
|  | <td> |
|  | <h2>Tambah Program Kerja BEM</h2> |
|  | <form method="post" action=""> |
|  | <table> |
|  | <tr> |
|  | <td><label>Nomor Program Kerja</label></td> |
|  | <td><input type="text" name="nomor" value=""></td> |
|  | </tr> |
|  | <tr> |
|  | <td><label>Nama Program Kerja</label></td> |
|  | <td><input type="text" name="nama" value=""> |
|  | </tr> |
|  | <tr> |
|  | <td><label>Surat Keterangan</label></td> |
|  | <td><input type="text" name="surat" value=""></td> |
|  | </tr> |
|  | </table> |
|  | <input type="submit" name="ins" value="Tambah"> |
|  | </form> |
|  | </td> |
|  | </tr> |
|  | </table> |
|  | </body> |
|  |  |
|  | </html> |
|  | <?php |
|  | if (isset($\_POST['ins'])) { |
|  | $control = new c\_programKerja(); |
|  | $control->insert(); |
|  | } |
|  | ?> |

|  |  |
| --- | --- |
| index.php | |
|  | <?php |
|  | include\_once("c\_programKerja.php"); |
|  | $controller = new c\_programKerja(); |
|  | if (isset($\_GET['d'])) { |
|  | $controller->delete($\_GET['d']); |
|  | } else if (isset($\_GET['u'])) { |
|  | $controller->viewUpdate($\_GET['u']); |
|  | } else if (isset($\_GET['i'])) { |
|  | $controller->viewInsert(); |
|  | } else { |
|  | $controller->invoke(); |
|  | } |

1. **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.

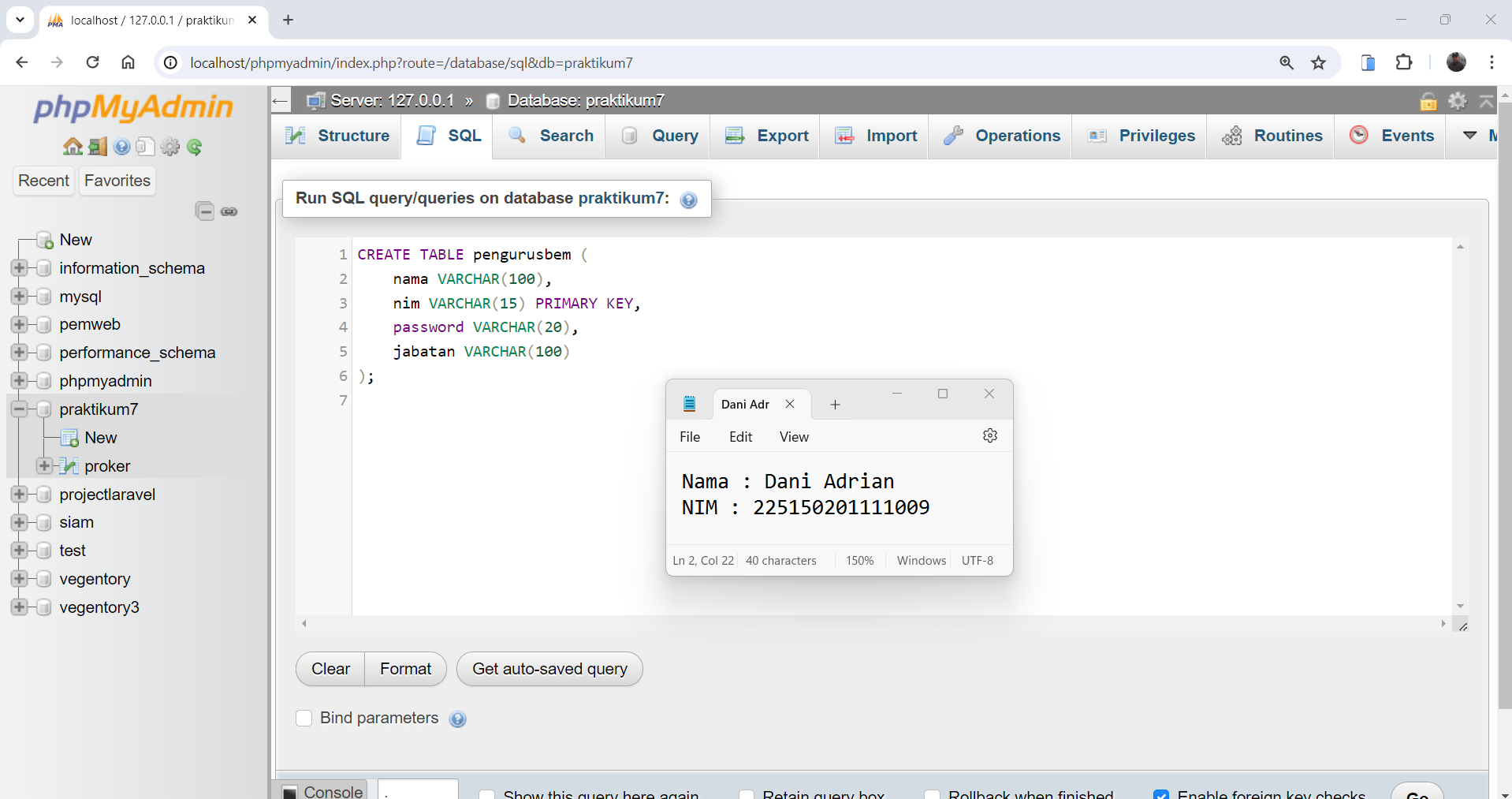
**LATIHAN 5**

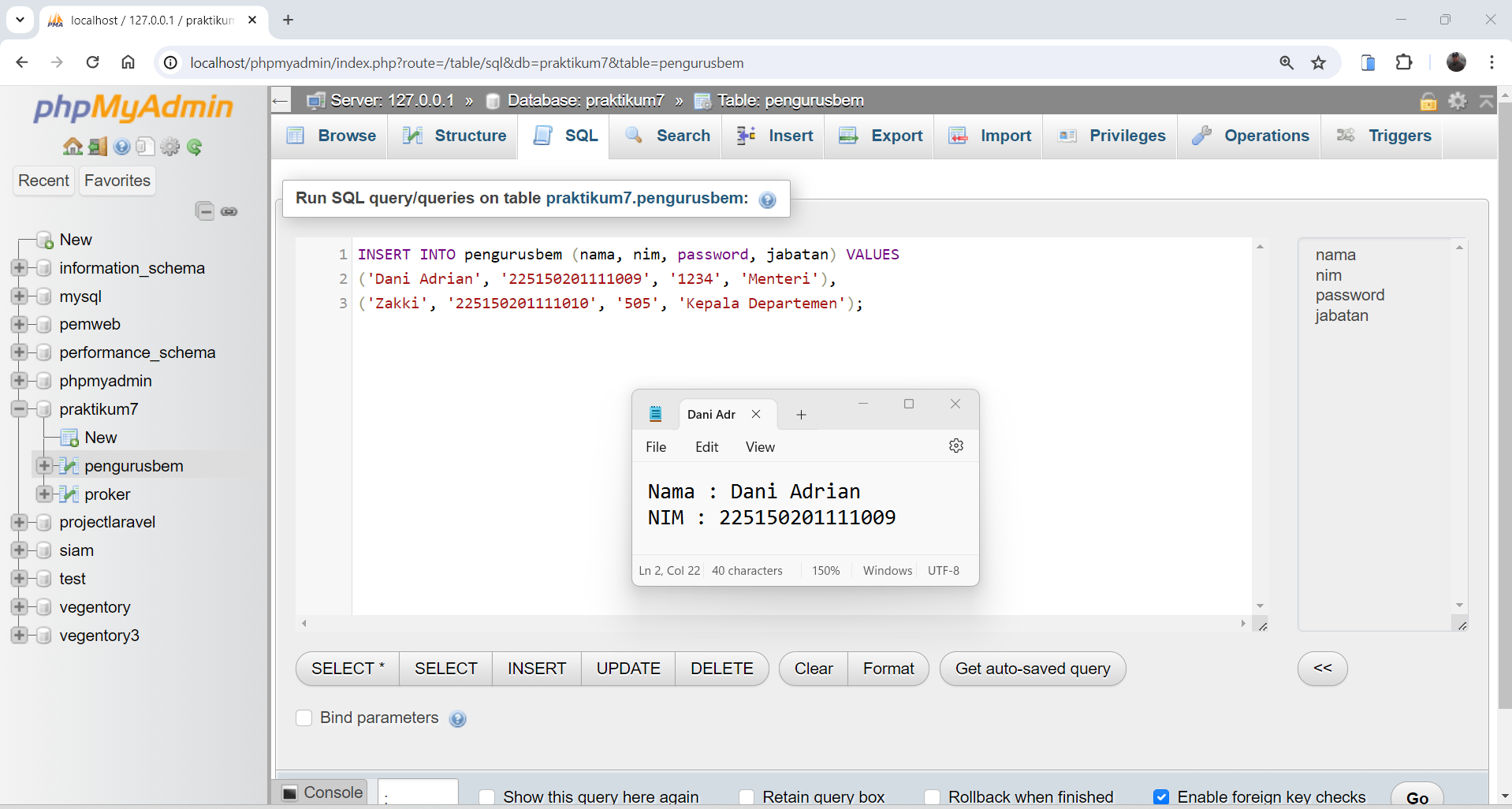
1. **Soal**

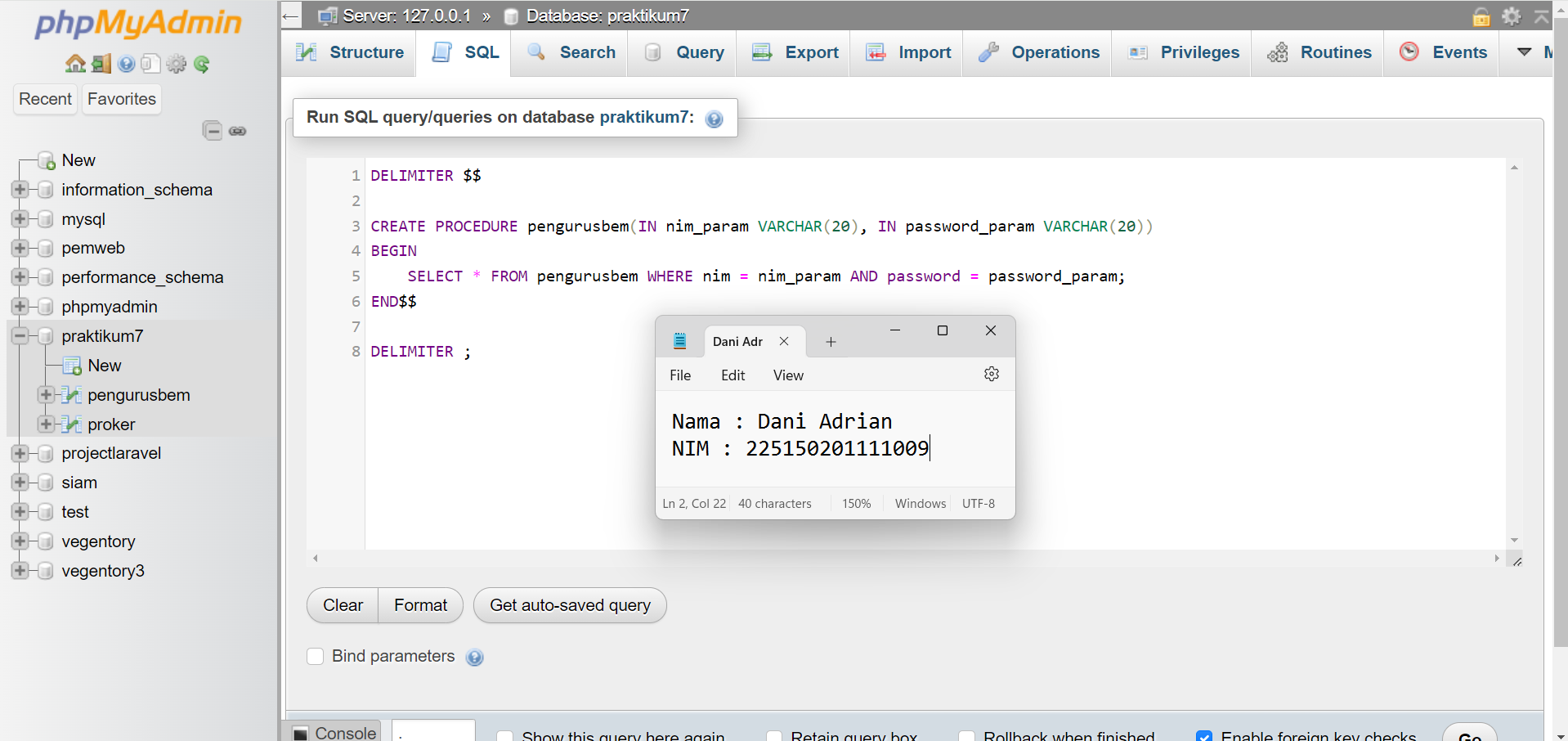
Modifikasi sesuai kebutuhan terlebih dahulu, lengkapi dengan halaman login dan halaman administrasi dan gabungkan Latihan 2 dan Latihan 3 sehingga didapat skenario Menteri hanya dapat melihat daftar program kerja saja, sedangkan Kepala Departemen dapat melakukan proses manajemen program kerja (CRUD).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

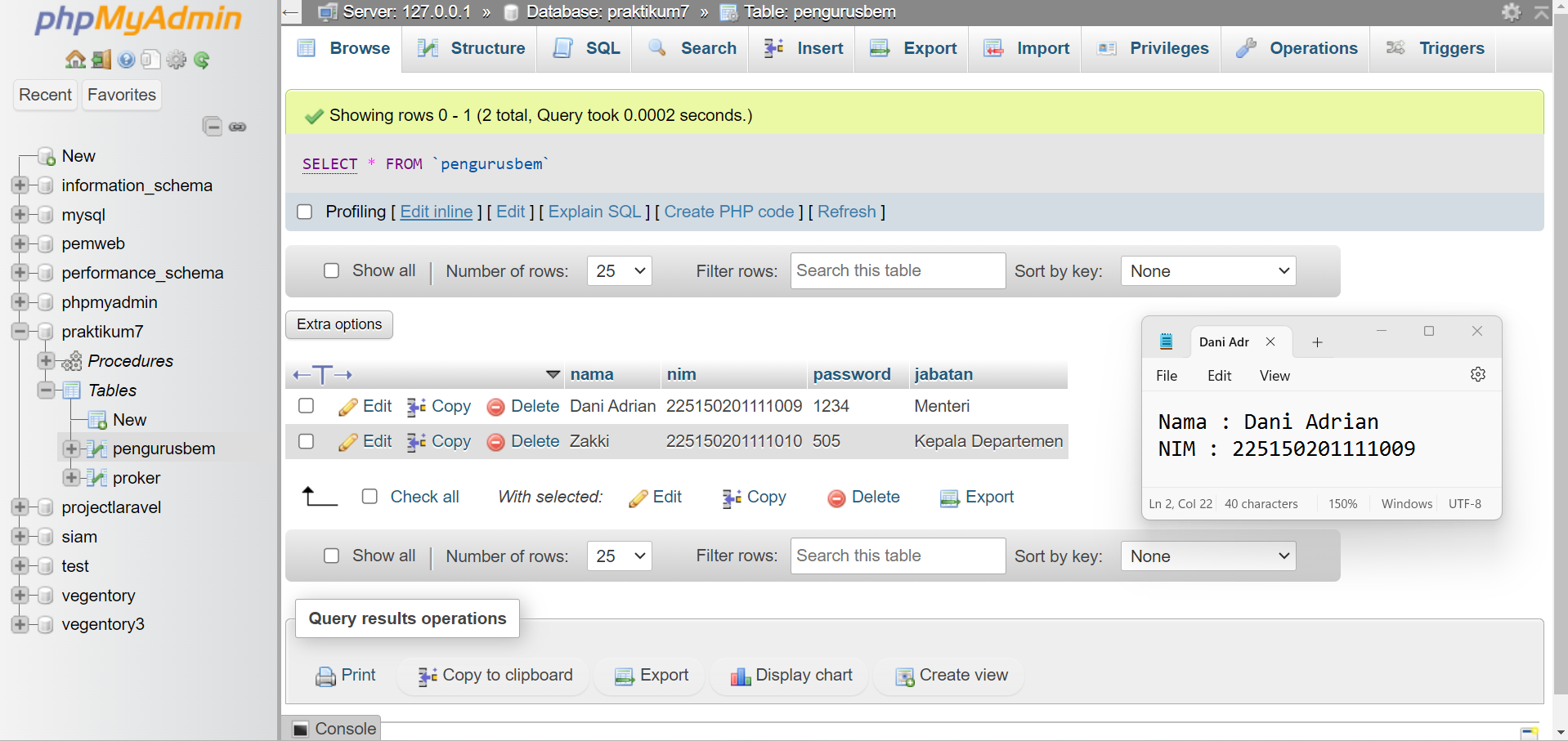
1. **Screenshoot**

* Membuat Database pengurusbem

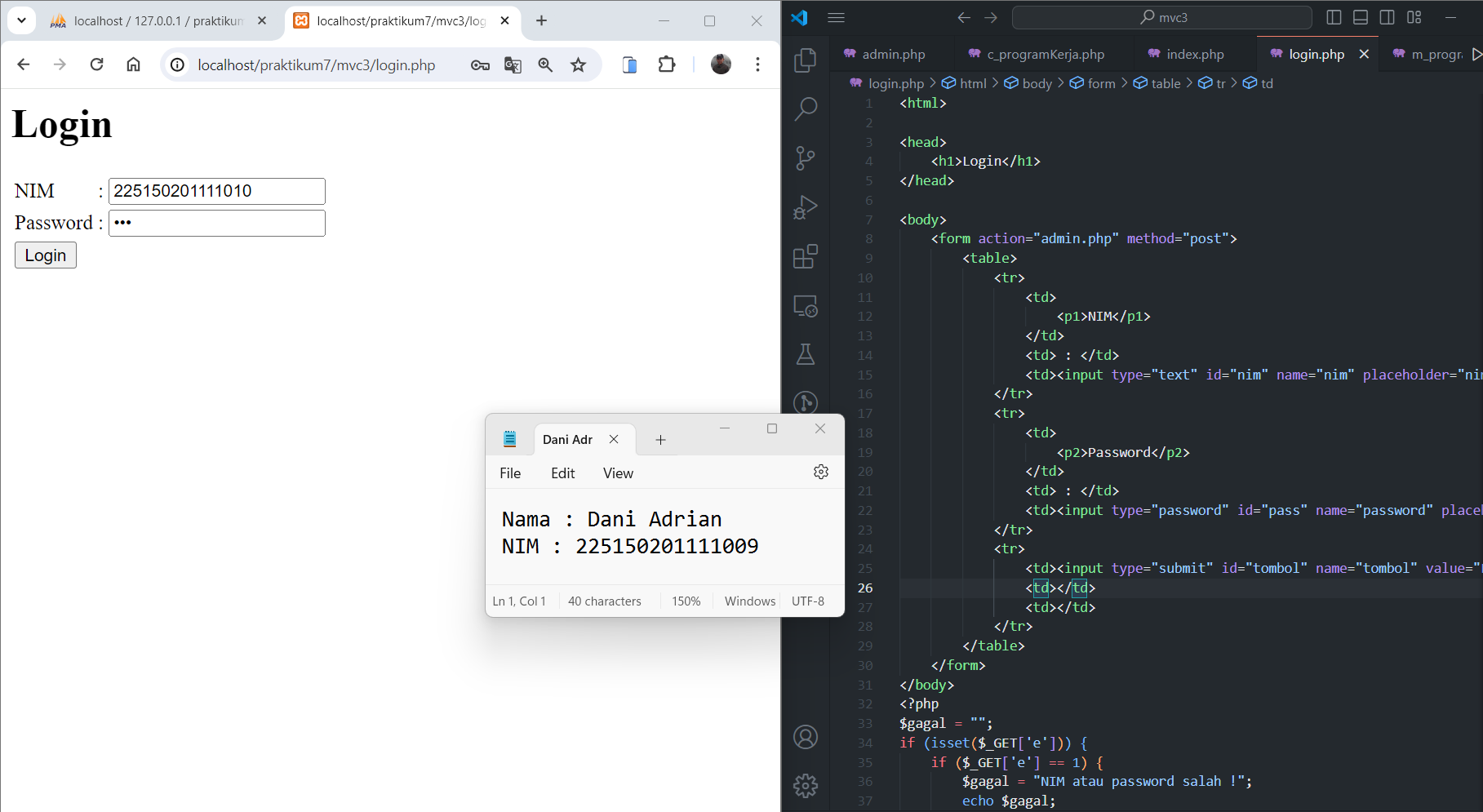




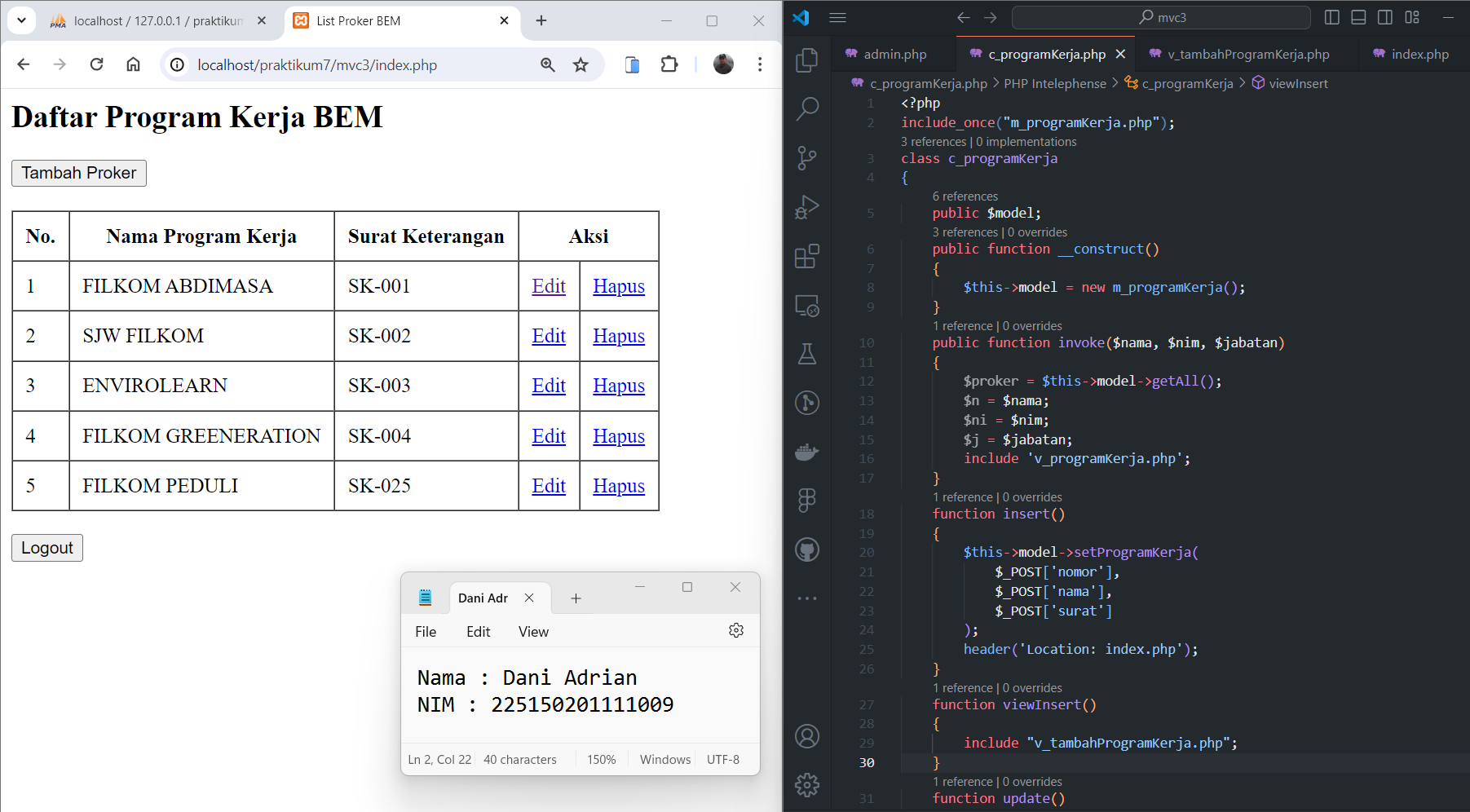




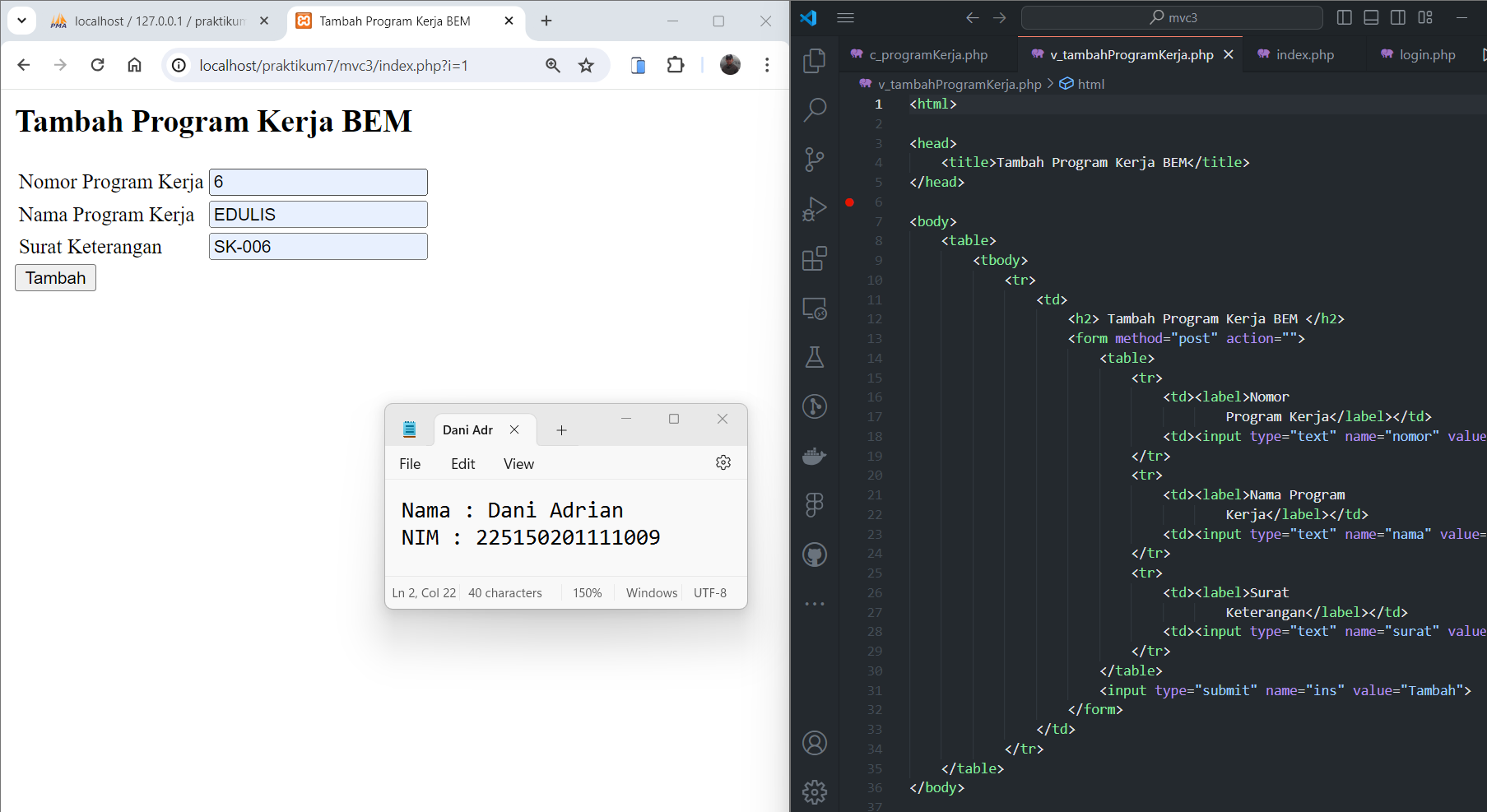
* Form login sebagai Kepala Departemen

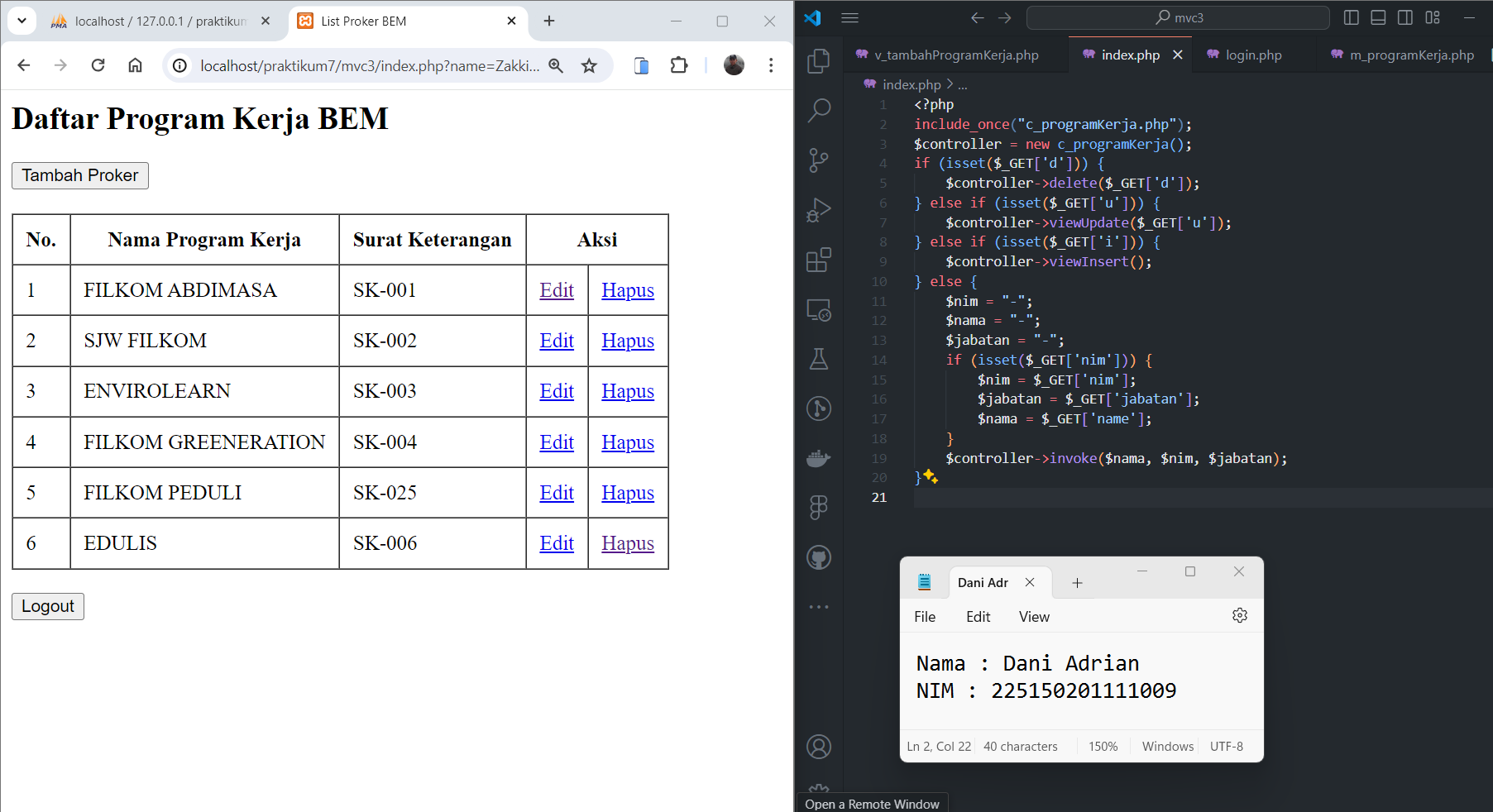


* Daftar program kerja BEM sebagai Kepala Departemen

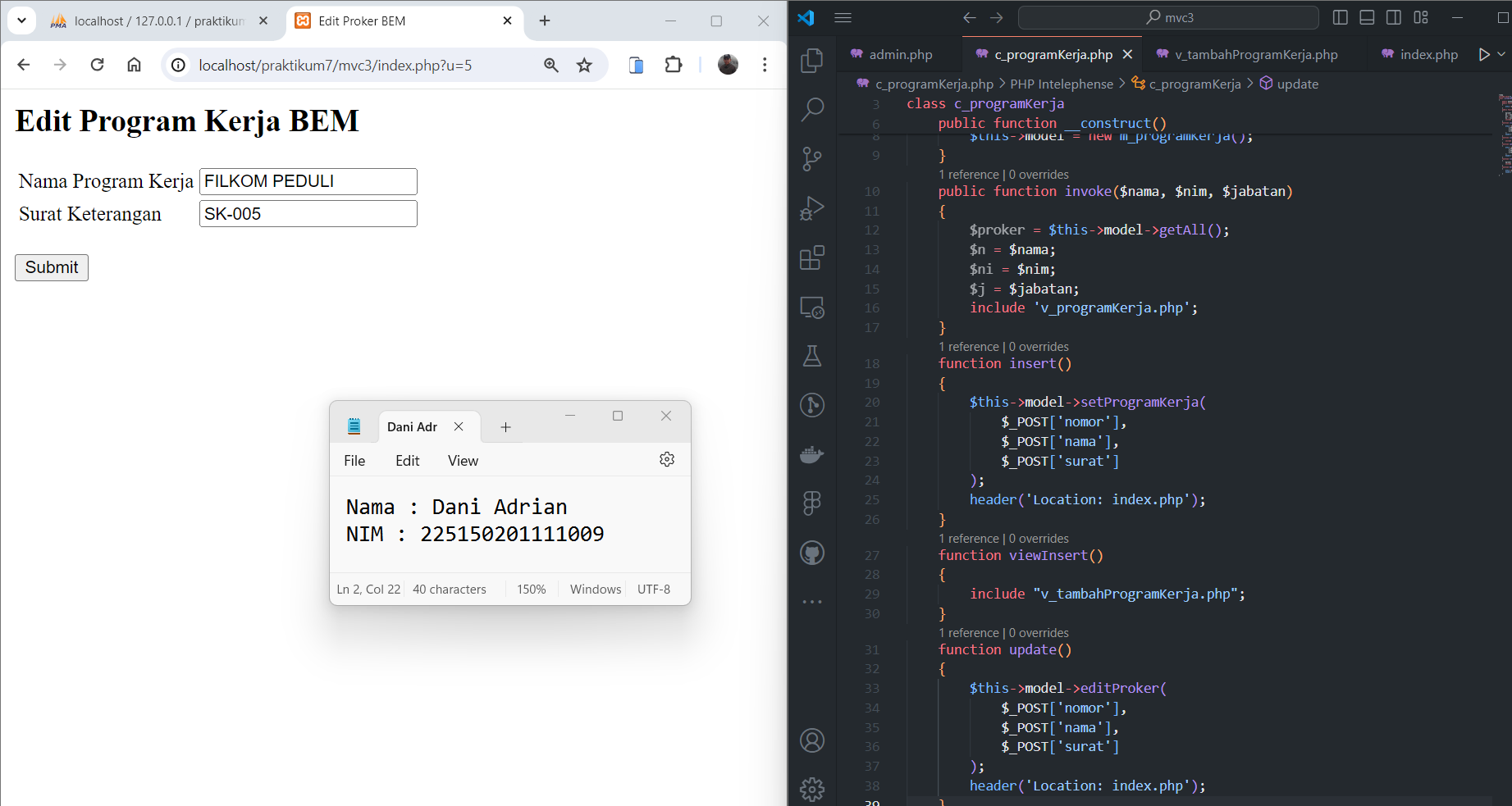


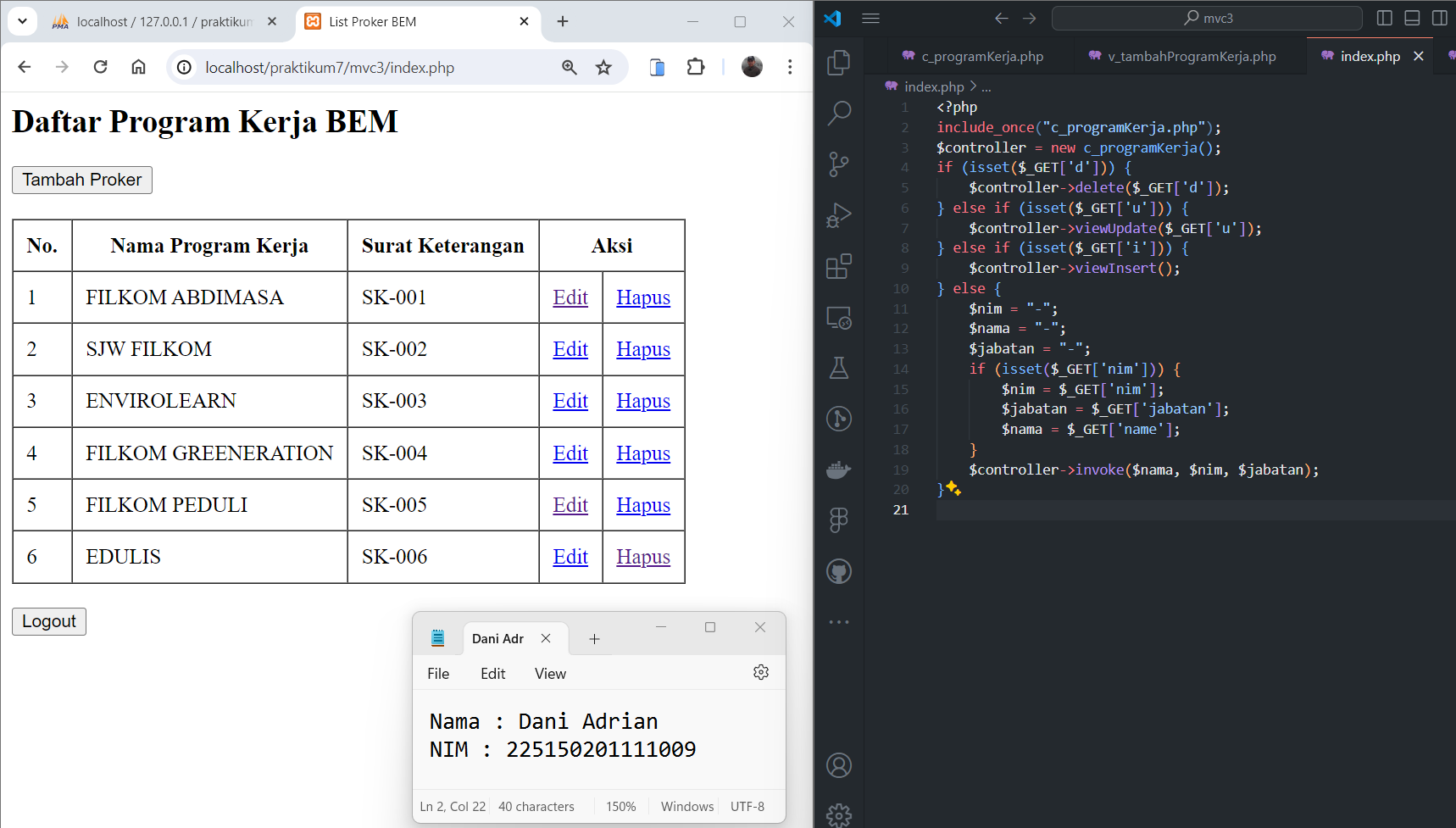
* Form tambah program kerja BEM



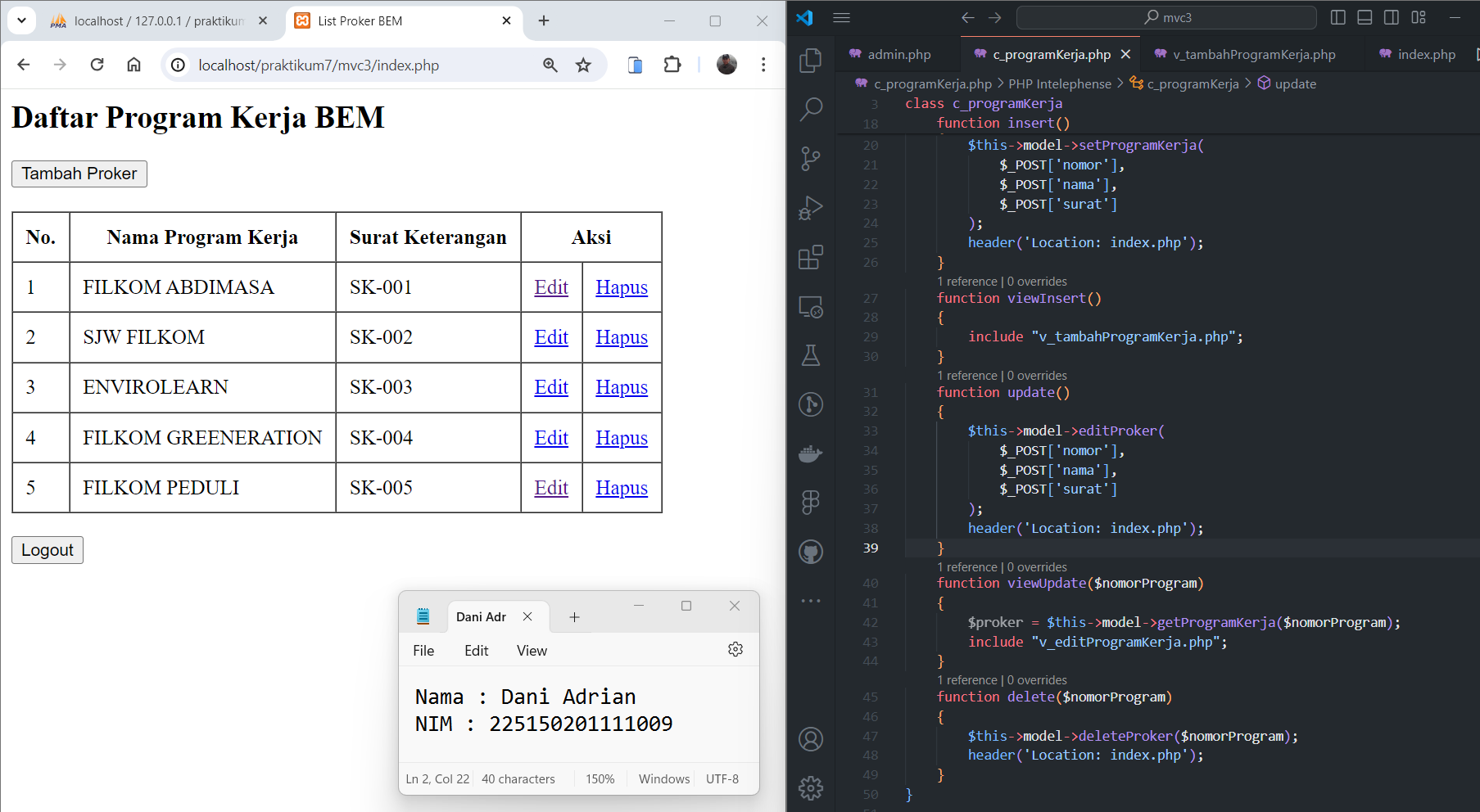


* Form edit program kerja BEM

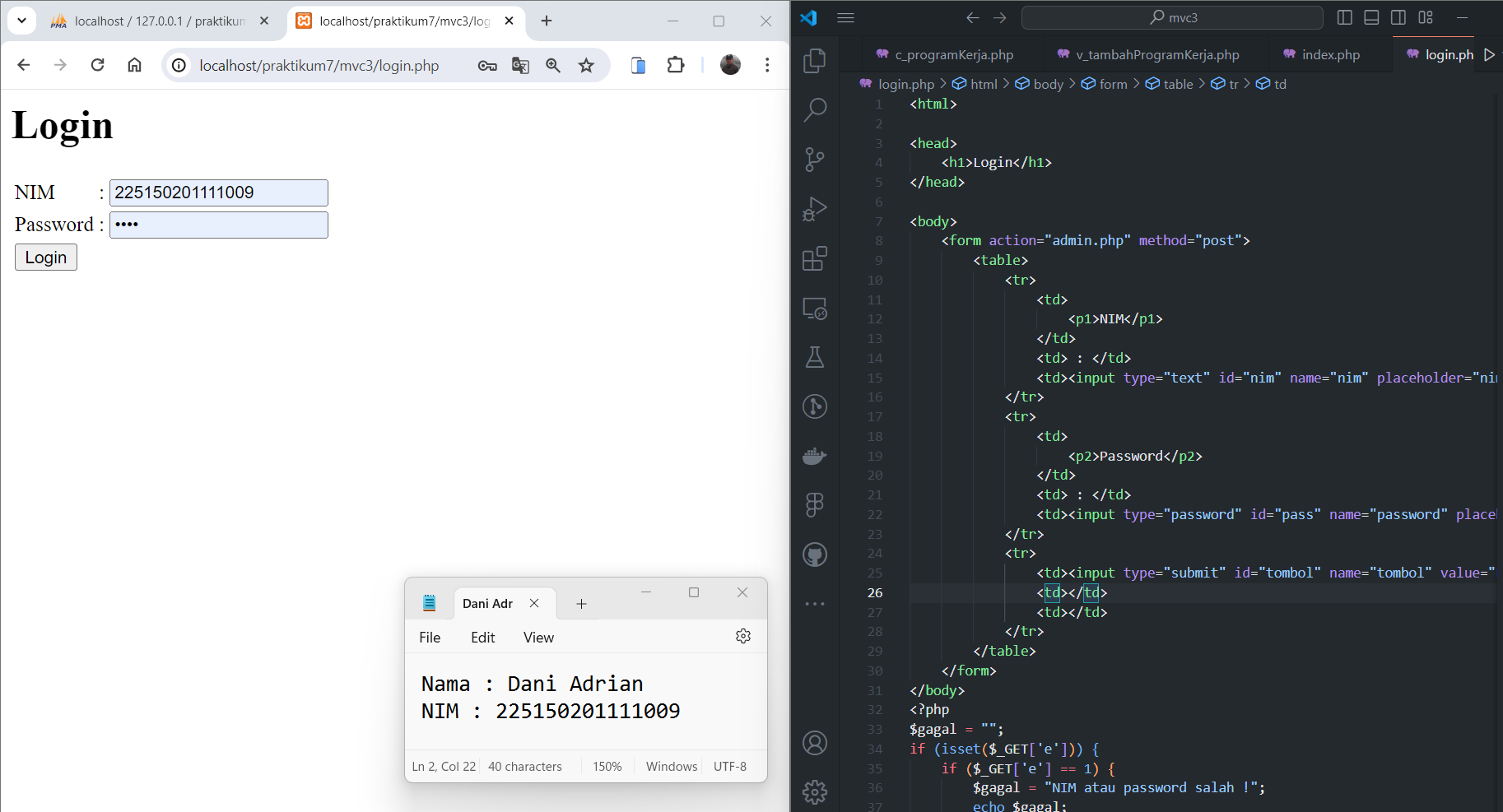




* Hapus salah satu program kerja BEM



* Form login sebagai Menteri





1. **Syntax**

|  |  |
| --- | --- |
| CREATE TABLE penguruBEM | |
|  | CREATE TABLE pengurusbem ( |
|  | nama VARCHAR(100), |
|  | nim VARCHAR(15) PRIMARY KEY, |
|  | password VARCHAR(20), |
|  | jabatan VARCHAR(100) |
|  | ); |

|  |  |
| --- | --- |
|  | DELIMITER $$ |
|  |  |
|  | CREATE PROCEDURE pengurusbem(IN nim\_param VARCHAR(20), IN password\_param VARCHAR(20)) |
|  | BEGIN |
|  | SELECT \* FROM pengurus\_bem WHERE nim = nim\_param AND password = password\_param; |
|  | END$$ |
|  |  |
|  | DELIMITER ; |

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

|  |  |
| --- | --- |
| admin.php | |
|  | <?php |
|  | $conn = new mysqli("localhost", "root", "02030507", "praktikum7"); |
|  | $nim = $\_POST['nim']; |
|  | $password = $\_POST['password']; |
|  | $query = "CALL pengurusbem('$nim','$password')"; |
|  | $hasil = $conn->query($query) or die('query gagal'); |
|  | if ($hasil->num\_rows > 0) { |
|  | $nama = ""; |
|  | $nim = ""; |
|  | $password = ""; |
|  | $jabatan = ""; |
|  | while ($data = $hasil->fetch\_object()) { |
|  | $nama = $data->nama; |
|  | $nim = $data->nim; |
|  | $password = $data->password; |
|  | $jabatan = $data->jabatan; |
|  | } |
|  | require 'pengurusBEM.php'; |
|  | $pb = new pengurusBEM(); |
|  | $pb->setJabatan($jabatan); |
|  | $pb->setNama($nama); |
|  | $pb->setNim($nim); |
|  | $pb->pilih(); |
|  | } else { |
|  | header('location: login.php?e=1'); |
|  | } |
|  | ?> |

|  |  |
| --- | --- |
| index.php | |
|  | <?php |
|  | include\_once("c\_programKerja.php"); |
|  | $controller = new c\_programKerja(); |
|  | if (isset($\_GET['d'])) { |
|  | $controller->delete($\_GET['d']); |
|  | } else if (isset($\_GET['u'])) { |
|  | $controller->viewUpdate($\_GET['u']); |
|  | } else if (isset($\_GET['i'])) { |
|  | $controller->viewInsert(); |
|  | } else { |
|  | $nim = "-"; |
|  | $nama = "-"; |
|  | $jabatan = "-"; |
|  | if (isset($\_GET['nim'])) { |
|  | $nim = $\_GET['nim']; |
|  | $jabatan = $\_GET['jabatan']; |
|  | $nama = $\_GET['name']; |
|  | } |
|  | $controller->invoke($nama, $nim, $jabatan); |
|  | } |

|  |  |
| --- | --- |
| login.php | |
|  | <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"><br /></td> |
|  | </tr> |
|  | <tr> |
|  | <td> |
|  | <p2>Password</p2> |
|  | </td> |
|  | <td> : </td> |
|  | <td><input type="password" id="pass" name="password" placeholder="password"><br /></td> |
|  | </tr> |
|  | <tr> |
|  | <td><input type="submit" id="tombol" name="tombol" value="Login"></td> |
|  | <td></td> |
|  | <td></td> |
|  | </tr> |
|  | </table> |
|  | </form> |
|  | </body> |
|  | <?php |
|  | $gagal = ""; |
|  | if (isset($\_GET['e'])) { |
|  | if ($\_GET['e'] == 1) { |
|  | $gagal = "NIM atau password salah !"; |
|  | echo $gagal; |
|  | } else if ($\_GET['e'] == 2) { |
|  | echo "Terjadi Kesalahan, Login Kembali !"; |
|  | } |
|  | } |
|  | ?> |
|  |  |
|  | </html> |

|  |  |
| --- | --- |
| c\_programKerja.php | |
|  | <?php |
|  | include\_once("m\_programKerja.php"); |
|  | class c\_programKerja |
|  | { |
|  | public $model; |
|  | public function \_\_construct() |
|  | { |
|  | $this->model = new m\_programKerja(); |
|  | } |
|  | public function invoke($nama, $nim, $jabatan) |
|  | { |
|  | $proker = $this->model->getAll(); |
|  | $n = $nama; |
|  | $ni = $nim; |
|  | $j = $jabatan; |
|  | include 'v\_programKerja.php'; |
|  | } |
|  | function insert() |
|  | { |
|  | $this->model->setProgramKerja( |
|  | $\_POST['nomor'], |
|  | $\_POST['nama'], |
|  | $\_POST['surat'] |
|  | ); |
|  | header('Location: index.php'); |
|  | } |
|  | function viewInsert() |
|  | { |
|  | include "v\_tambahProgramKerja.php"; |
|  | } |
|  | function update() |
|  | { |
|  | $this->model->editProker( |
|  | $\_POST['nomor'], |
|  | $\_POST['nama'], |
|  | $\_POST['surat'] |
|  | ); |
|  | header('Location: index.php'); |
|  | } |
|  | function viewUpdate($nomorProgram) |
|  | { |
|  | $proker = $this->model->getProgramKerja($nomorProgram); |
|  | include "v\_editProgramKerja.php"; |
|  | } |
|  | function delete($nomorProgram) |
|  | { |
|  | $this->model->deleteProker($nomorProgram); |
|  | header('Location: index.php'); |
|  | } |
|  | } |

|  |  |
| --- | --- |
| m\_programKerja.php | |
|  | <?php |
|  | class m\_programKerja |
|  | { |
|  | public $hasil = array(); |
|  | private $mysqli; |
|  | public function \_\_construct() |
|  | { |
|  | require 'koneksiMVC.php'; |
|  | } |
|  | public function setProgramKerja( |
|  | $nomorProker, |
|  | $namaProker, |
|  | $suratKet |
|  | ) { |
|  | require 'koneksiMVC.php'; |
|  | $rs = $this->mysqli->query("INSERT INTO proker |
|  | VALUES ('$nomorProker', '$namaProker','$suratKet')"); |
|  | } |
|  | public function getProgramKerja($nomorProker) |
|  | { |
|  | require 'koneksiMVC.php'; |
|  | $rs = $this->mysqli->query("SELECT \* FROM |
|  | proker WHERE nomorProgram = $nomorProker"); |
|  | $this->hasil = $rs->fetch\_array(); |
|  | return $this->hasil; |
|  | } |
|  | public function deleteProker($nomorProker) |
|  | { |
|  | require 'koneksiMVC.php'; |
|  | $rs = $this->mysqli->query("DELETE from proker |
|  | WHERE nomorProgram = $nomorProker"); |
|  | } |
|  | public function editProker( |
|  | $nomorProker, |
|  | $namaProker, |
|  | $suratKet |
|  | ) { |
|  | require 'koneksiMVC.php'; |
|  | $rs = $this->mysqli->query("UPDATE proker SET |
|  | namaProgram = '$namaProker',suratKeterangan= |
|  | '$suratKet' WHERE nomorProgram = $nomorProker"); |
|  | } |
|  | public function getAll() |
|  | { |
|  | require 'koneksiMVC.php'; |
|  | $rs = $this->mysqli->query("SELECT \* FROM |
|  | proker"); |
|  | while ($row = $rs->fetch\_assoc()) { |
|  | $this->hasil[] = $row; |
|  | } |
|  | return $this->hasil; |
|  | } |
|  | } |

|  |  |
| --- | --- |
| pengurusBEM.php | |
|  | <?php |
|  | class pengurusBEM |
|  | { |
|  | private $nama; |
|  | private $nim; |
|  | private $angkatan; |
|  | private $jabatan; |
|  | public function setNama($nama) |
|  | { |
|  | $this->nama = $nama; |
|  | } |
|  | public function getNama() |
|  | { |
|  | return $this->nama; |
|  | } |
|  | public function setJabatan($jabatan) |
|  | { |
|  | $this->jabatan = $jabatan; |
|  | } |
|  | public function getJabatan() |
|  | { |
|  | return $this->jabatan; |
|  | } |
|  | public function setNim($nim) |
|  | { |
|  | $this->nim = $nim; |
|  | } |
|  | public function getNim() |
|  | { |
|  | return $this->nim; |
|  | } |
|  | public function pilih() |
|  | { |
|  | $jabat = $this->getJabatan(); |
|  | $nama = $this->getNama(); |
|  | $nim = $this->getNim(); |
|  | if ($jabat == "Kepala Departemen") { |
|  | echo "<meta http-equiv=\"refresh\" |
|  | content=\"0;index.php?name=$nama&nim=$nim&jabatan=$jabat\">"; |
|  | } else if ($jabat = "Menteri") { |
|  | echo "<meta http-equiv=\"refresh\" |
|  | content=\"0;menteri.php?name=$nama&nim=$nim&jabatan=$jabat\">"; |
|  | } else { |
|  | header('location: login.php?e=2'); |
|  | } |
|  | } |
|  | } |

|  |  |
| --- | --- |
| menteri.php | |
|  | <?php |
|  | class ambilData |
|  | { |
|  | public function \_\_construct() |
|  | { |
|  | require 'koneksiMVC.php'; |
|  | } |
|  | public function getSemuaProgramKerja() |
|  | { |
|  | $rs = $this->mysqli->query("SELECT \* FROM proker"); |
|  | while ($row = $rs->fetch\_assoc()) { |
|  | $this->hasil[] = $row; |
|  | } |
|  | return $this->hasil; |
|  | } |
|  | } |
|  | $ad = new ambilData(); |
|  | $proker = $ad->getSemuaProgramKerja(); |
|  | ?> |
|  | <html> |
|  |  |
|  | <head> |
|  | <title> Daftar Proker BEM </title> |
|  | </head> |
|  |  |
|  | <body> |
|  | <h2>Daftar Program Kerja BEM </h2> |
|  | <h1></h1> |
|  | <table border="1"> |
|  | <tbody> |
|  | <tr><b> |
|  | <td>No</td> |
|  | <td>Nama Program Kerja</td> |
|  | <td>Surat Keterangan</td> |
|  | </b></tr> |
|  | </tbody> |
|  | <?php |
|  | foreach ($proker as $row) { |
|  | echo " <tr> |
|  | <td> $row[nomorProgram]</td> |
|  | <td> $row[namaProgram]</td> |
|  | <td> $row[suratKeterangan]</td> |
|  | </tr> "; |
|  | } |
|  | ?> |
|  | </table> |
|  | <br /> |
|  | <form action="login.php" method="post"> |
|  | <input type="submit" value="Logout"> |
|  | </form> |
|  | </body> |
|  |  |
|  | </html> |

|  |  |
| --- | --- |
| v\_programKerja.php | |
|  | <html> |
|  |  |
|  | <head> |
|  | <title>List Proker BEM</title> |
|  | </head> |
|  |  |
|  | <body> |
|  | <h2>Daftar Program Kerja BEM</h2> |
|  | <h3><input type="button" value="Tambah Proker" onclick="window.location.href='?i=1'"></h3> |
|  | <table border="1" cellpadding='10' cellspacing='0'> |
|  | <tbody> |
|  | <tr> |
|  | <th>No.</th> |
|  | <th>Nama Program Kerja</th> |
|  | <th>Surat Keterangan</th> |
|  | <th colspan="2">Aksi</th> |
|  | </tr> |
|  | </tbody> |
|  | <?php |
|  | foreach ($proker as $row) { |
|  | echo " <tr> |
|  | <td> $row[nomorProgram]</td> |
|  | <td> $row[namaProgram]</td> |
|  | <td> $row[suratKeterangan]</td> |
|  | <td><a href='?u=$row[nomorProgram]'>Edit |
|  | </a></td> |
|  | <td><a href='?d=$row[nomorProgram]'>Hapus |
|  | </a></td> |
|  | </tr> "; |
|  | } |
|  | ?> |
|  | </table> |
|  | <br /> |
|  | <form action="login.php" method="post"> |
|  | <input type="submit" value="Logout"> |
|  | </form> |
|  | </body> |
|  |  |
|  | </html> |

|  |  |
| --- | --- |
| v\_editProgramKerja.php | |
|  | <html> |
|  |  |
|  | <head> |
|  | <title>Edit Proker BEM</title> |
|  | </head> |
|  |  |
|  | <body> |
|  | <table> |
|  | <tbody> |
|  | <tr> |
|  | <td> |
|  | <h2> Edit Program Kerja BEM </h2> |
|  | <form method="post" action="#"> |
|  | <input type="hidden" name="nomor" value="<?php echo |
|  | $proker['nomorProgram'] ?>"> |
|  | <table> |
|  | <tr> |
|  | <td><label>Nama Program |
|  | Kerja</label></td> |
|  | <td><input type="text" name="nama" value="<?php echo |
|  | $proker['namaProgram'] ?>"></td> |
|  | </tr> |
|  | <tr> |
|  | <td><label>Surat |
|  | Keterangan</label></td> |
|  | <td><input type="text" name="surat" value="<?php echo |
|  | $proker['suratKeterangan'] ?>"> |
|  | </td> |
|  | </tr> |
|  | </table><br> |
|  | <input type="submit" name="edit" value="Submit"> |
|  | </form> |
|  | </td> |
|  | </tr> |
|  | </table> |
|  | </body> |
|  |  |
|  | </html> |
|  | <?php |
|  | if (isset($\_POST['edit'])) { |
|  | $control = new c\_programKerja(); |
|  | $control->update(); |
|  | } |
|  | ?> |

|  |  |
| --- | --- |
| v\_tambahProgramKerja.php | |
|  | <html> |
|  |  |
|  | <head> |
|  | <title>Tambah Program Kerja BEM</title> |
|  | </head> |
|  |  |
|  | <body> |
|  | <table> |
|  | <tbody> |
|  | <tr> |
|  | <td> |
|  | <h2> Tambah Program Kerja BEM </h2> |
|  | <form method="post" action=""> |
|  | <table> |
|  | <tr> |
|  | <td><label>Nomor |
|  | Program Kerja</label></td> |
|  | <td><input type="text" name="nomor" value=""></td> |
|  | </tr> |
|  | <tr> |
|  | <td><label>Nama Program |
|  | Kerja</label></td> |
|  | <td><input type="text" name="nama" value=""> |
|  | </tr> |
|  | <tr> |
|  | <td><label>Surat |
|  | Keterangan</label></td> |
|  | <td><input type="text" name="surat" value=""></td> |
|  | </tr> |
|  | </table> |
|  | <input type="submit" name="ins" value="Tambah"> |
|  | </form> |
|  | </td> |
|  | </tr> |
|  | </table> |
|  | </body> |
|  |  |
|  | </html> |
|  | <?php |
|  | if (isset($\_POST['ins'])) { |
|  | $control = new c\_programKerja(); |
|  | $control->insert(); |
|  | } |
|  | ?> |

1. **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.