

**TUGAS**  
**WEB PROGRAMMING 2**

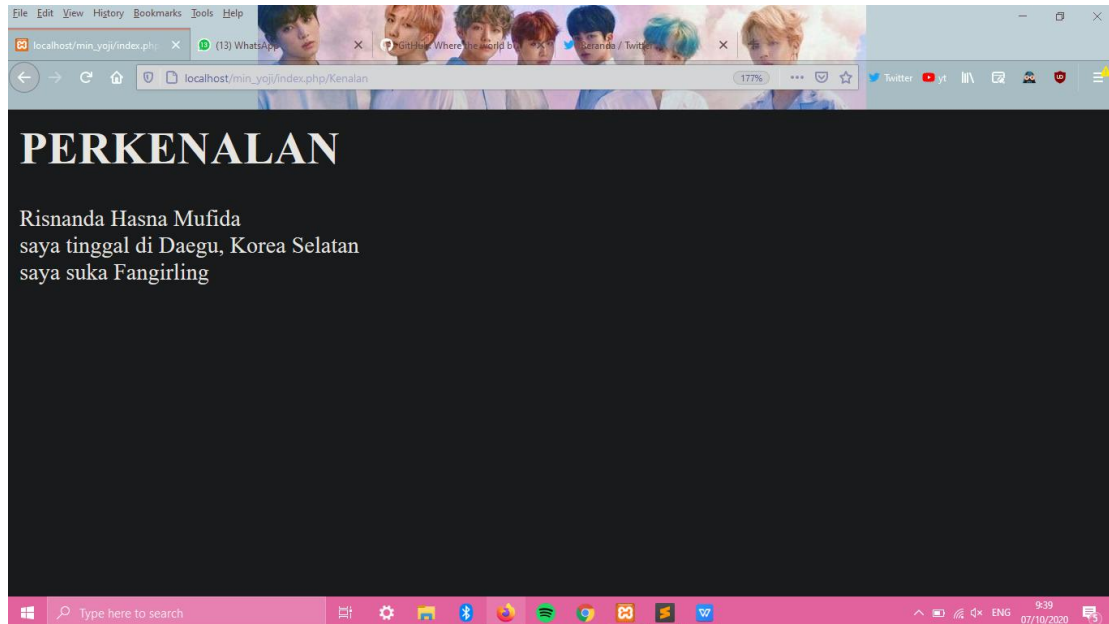


**Disusun Oleh:**  
**Risnanda Hasna Mufida      (18090092)**

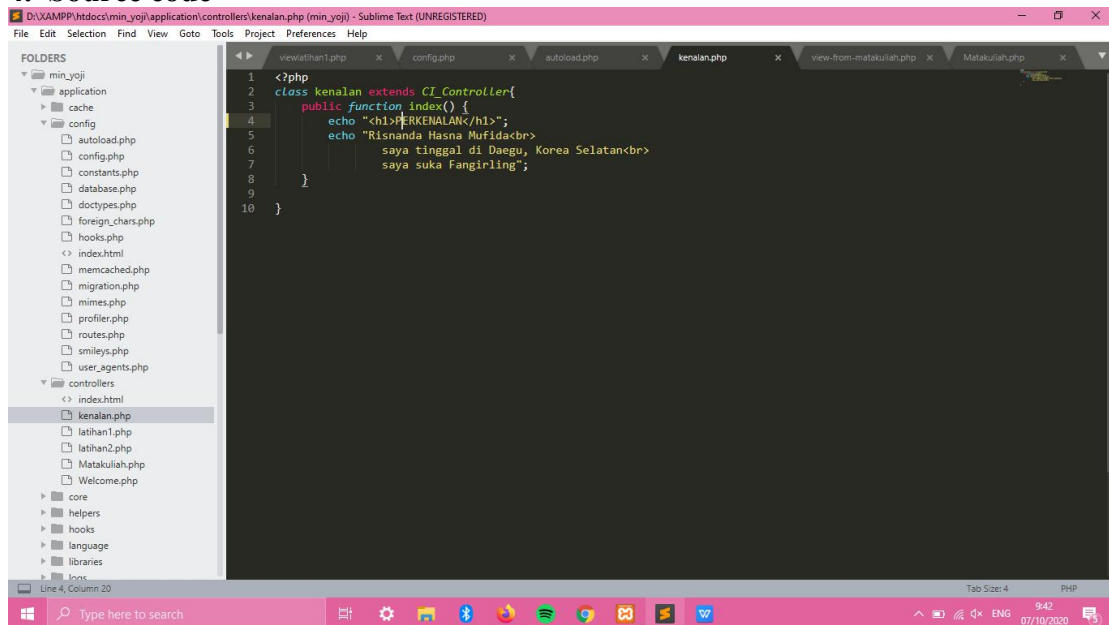
**PROGRAM STUDI SARJANA SAINS TERAPAN**  
**DIV TEKNIK INFORMATIKA**  
**POLITEKNIK HARAPAN BERSAMA TEGAL**  
**2020**

# PRAKTIKUM 1

1. Perkenalan.php -> application/controller
2. Hasil Url -> [http://localhost/min\\_yoji/index.php/kenalan](http://localhost/min_yoji/index.php/kenalan)
3. Tampilan



## 4. Source code



Perkenalan.php -> application/controller

## PRAKTIKUM 2

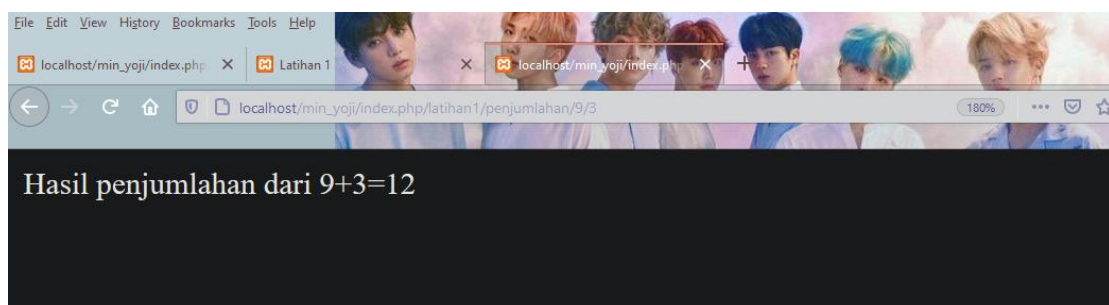
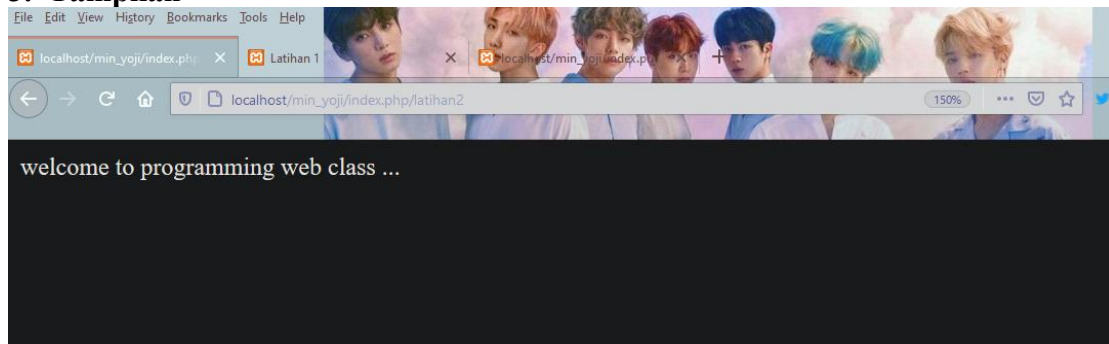
### 1. Lokasi :

- View-latihan1.php -> application/view
- Model1.php -> application/models
- Latihan1.php -> application/controllers
- Latihan2.php -> application/controllers

### 2. Hasil Url :

- [http://localhost/min\\_yoji/index.php/latihan2](http://localhost/min_yoji/index.php/latihan2)
- [http://localhost/min\\_yoji/index.php/latihan2/penjumlahan/1/6](http://localhost/min_yoji/index.php/latihan2/penjumlahan/1/6)
- [http://localhost/min\\_yoji/index.php/latihan1/penjumlahan/9/3](http://localhost/min_yoji/index.php/latihan1/penjumlahan/9/3)

### 3. Tampilan



#### 4. Source Code

```
viewlatihan1.php
1 <html>
2
3 <head>
4   <title>Latihan 1</title>
5 </head>
6
7 <body>
8
9   <h1>Hello friends ... lets learn web programming II.</h1><br>
10   Nilai 1 = <?= $nilai1 ?>
11   Nilai 2 = <?= $nilai2 ?>
12   ini adalah hasil dari permodelan penjumlahan <?= $nilai1."<+>".$nilai2."<=>".$hasil; ?>
13
14 </body>
15 </html>
```

View-latihan1.php -> application/view

```
model1.php
1 <?php
2 class model1 extends CI_Model{
3
4   //membuat variabel yang menampung nilai
5   public $nilai1, $nilai2, $hasil;
6
7   public function jumlah($nil1=null, $nil2=null)
8   {
9
10    $this->nilai1 = $nil1;
11    $this->nilai2 = $nil2;
12    $this->hasil = $this->nilai1 + $this->nilai2;
13    return $this->hasil;
14  }
15 }
```

Model1.php -> application/models

```
latihan1.php
1 <?php
2 class latihan1 extends CI_Controller{
3
4   public function index() {
5     echo " welcome to programming web class ...";
6   }
7
8   public function penjumlahan($n1, $n2){
9     $this->load->model('model1');
10    $hasil = $this->model1->jumlah($n1, $n2);
11    echo "Hasil penjumlahan dari " . $n1."<+>".$n2."<=>".$hasil;
12  }
13
14 }
```

Latihan1.php -> application/controllers

```
latihan2.php x
1 <?php
2 class latihan2 extends CI_Controller{
3
4     //mengkombinasikan view controller model
5
6     public function index() {
7         echo " welcome to programming web class ...";
8     }
9
10    public function penjumlahan($n1, $n2){
11        $this->load->model('model1');
12
13        $data['nilai1'] = $n1;
14        $data['nilai2'] = $n2;
15        $data['hasil'] = $this->model1->jumlah($n1,$n2);
16
17        $this->load->view('viewlatihan1', $data);
18    }
19
20 }
```

Latihan2.php -> application/controllers

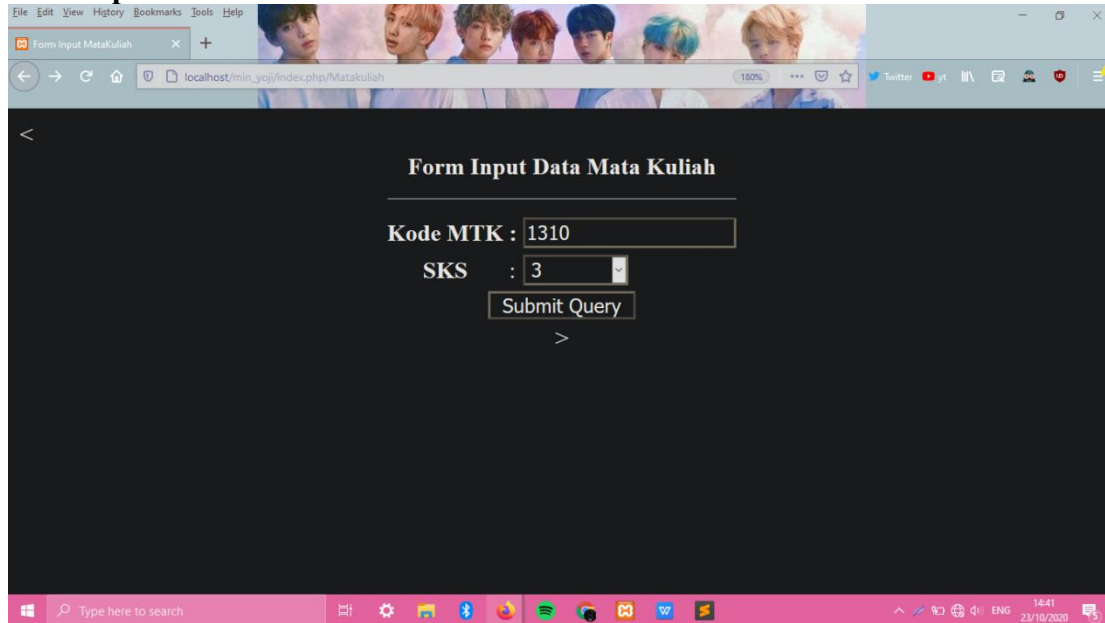
## PRAKTIKUM 3

### 1. Lokasi

- A. Matakuliah.php-> application/controller
- B. View-from-matakuliah.php->view/errors

### 2. Hasil URL-> [http://localhost/min\\_yoji/index.php/Matakuliah](http://localhost/min_yoji/index.php/Matakuliah)

### 3. Tampilan



### 4. Source Code

```
view-from-matakuliah.php x Matakuliah.php x
1 <<?php
2
3 class Matakuliah extends CI_controller
4 {
5
6     public function index()
7     {
8         $this->load->view('view-from-matakuliah');
9     }
10
11 }
```

Matakuliah.php-> application/controller

```
view-from-matakuliah.php x viewlatihan1.php x
1 <head>
2   <title>Form Input Matakuliah</title>
3 </head>
4
5 <body>
6   <center>
7     <form action="<?= base_url('matakuliah'); ?>"
8       " method="post">
9       <table>
10        <tr>
11          <th colspan="3">
12            Form Input Data Mata Kuliah
13          </th>
14        </tr>
15
16        <tr>
17          <td colspan="3">
18            <hr>
19          </td>
20        </tr>
21
22        <tr>
23          <th>Kode MTK</th>
24          <th>:</th>
25          <td>
26            <input type="text" name="nama" id="nama">
27          </td>
28        </tr>
```

View-from-matakuliah.php->view/errors

```
view-from-matakuliah.php x viewlatihan1.php x
29
30   <tr>
31     <th>SKS</th>
32     <td>:</td>
33     <td>
34       <select name="SKS" id="sks">
35         <option value=""> Pilih SKS </option>
36         <option value="2"> 2 </option>
37         <option value="3"> 3 </option>
38         <option value="4"> 4 </option>
39       </select>
40     </td>
41   </tr>
42
43   <tr>
44     <th colspan="3" align="center">
45       <input type="submit" name="submit">
46     </th>
47   </tr>
48
49 </table>
50 ></form>
51 </center>
52 </body>
```

View-from-matakuliah.php->view/errors

## **PRAKTIKUM 4**



## PRAKTIKUM 5

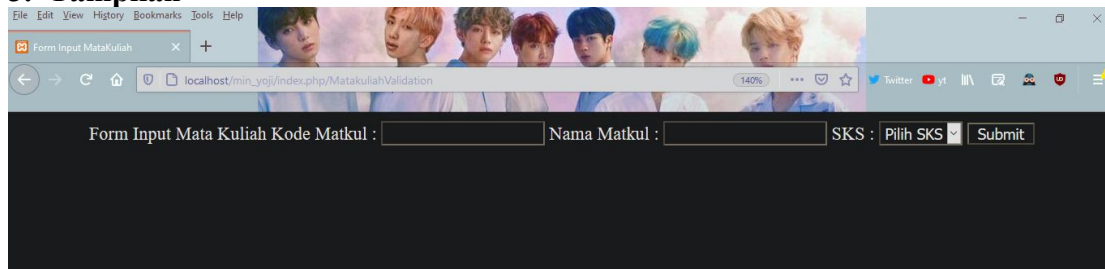
### 1. Lokasi

- A. autoload.php -> application/config
- B. MatakuliahValidation.php -> application/controllers
- C. Form-matakuliah.php/application/views

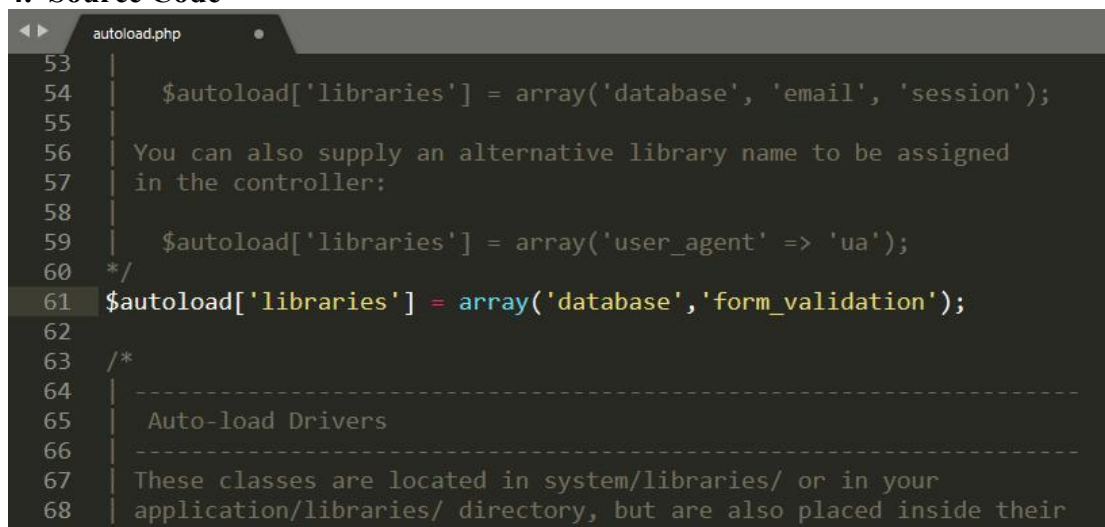
### 2. URL

[http://localhost/min\\_yoji/index.php/MatakuliahValidation](http://localhost/min_yoji/index.php/MatakuliahValidation)

### 3. Tampilan



### 4. Source Code



autoload.php -> application/config

```

1 <?php
2 class MatakuliahValidation extends CI_Controller
3 {
4     public function index()
5     {
6         $this->load->view('form-matakuliah');
7     }
8 }
9
10
11 function cetak()
12 {
13     $this->form_validation->set_rules('kode', 'Kode Matakuliah', 'required|min_length[3]', [
14         'required' => 'Kode Matakuliah Harus diisi', 'min_lenght' => 'Kode terlalu pendek'
15     ]);
16
17     $this->form_validation->set_rules('nama', 'Nama Matakuliah', 'required|min_length[3]', [
18         'required' => 'Nama Matakuliah Harus diisi', 'min_lenght' => 'Nama terlalu pendek'
19     ]);
20
21     if ($this->form_validation->run() != true) {
22         $this->load->view('view-form-matakuliah');
23     } else {

```

MatakuliahValidation.php-> application/controllers

```

34 } else {
35     $data = [
36         'kode' => $this->input->post('kode'), 'nama' => $this->input->post('nama'), 'sks' => $this->input->post('sks')
37     ];
38     $this->load->view('view-data-matakuliah', $data);
39 }
40 }
41 }
42 }
43 }
44 }
45 }
46 }
47 }
48 }

```

MatakuliahValidation.php-> application/controllers

```

1 <html>
2
3 <head>
4   <title>Form Input MataKuliah</title>
5 </head>
6
7 <body>
8   <center>
9     <form action="<?= base_url("matakuliah/cetak"); ?>" method="post"
10     <table>
11       <tr>
12         <th colspan="3">
13           Form Input Mata Kuliah
14         </th>
15       </tr>
16
17       <tr>
18         <th colspan="3">
19         </th>
20       </tr>
21
22       <tr>
23         <th> Kode Matkul</th>
24         <th> : </th>
25         <td>
26           <input type="text" name="kode" id="kode">
27         </td>
28       </tr>
29
30       <tr>
31         <th> Nama Matkul</th>
32         <td> : </td>
33         <td>
34           <input type="text" name="kode" id="kode">
35         </td>

```

Form-matakuliah.php/application/views

```

35 </td>
36 </tr>
37
38 <tr>
39   <th> SKS </th>
40   <td> : </td>
41   <td>
42     <select name="SKS" id="sks">
43       <option value="">Pilih SKS</option>
44       <option value="">2</option>
45       <option value="">3</option>
46       <option value="">4</option>
47     </select>
48   </td>
49 </tr>
50
51 <tr>
52   <th colspan="3" align="center">
53     <button type="sumbit" value="sumbit">Submit
54   </th>
55 </tr>
56
57 </table>
58 </form>
59 </center>
60 </body>
61
62 </html>

```

Form-matakuliah.php/application/views

## PRAKTIKUM 6

### Lokasi

autoload.php-> application/config  
database.php->application/config  
modelUser.php->application/model  
modelBuku.php->application/model

### Source Code

```
57 | in the controller:
58 |
59 |     $autoload['libraries'] = array('user_agent' => 'ua');
60 | */
61 | $autoload['libraries'] = array('database', 'form_validation');
62 |
63 | /*
64 | -----
65 | Auto-load Drivers
66 | -----
67 | These classes are located in system/libraries/ or in your
68 | application/libraries/ directory, but are also placed inside their
69 | own subdirectory and they extend the CI_Driver_Library class. They
70 | offer multiple interchangeable driver options.
71 |
72 | Prototype:
73 |
74 |     $autoload['drivers'] = array('cache');
75 |
76 | You can also supply an alternative property name to be assigned in
77 | the controller:
78 |
79 |     $autoload['drivers'] = array('cache' => 'cch');
80 | */
81 | $autoload['drivers'] = array();
82 |
83 | /*
84 | -----
85 | Auto-load Helper Files
86 | -----
87 | Prototype:
88 |
89 |     $autoload['helper'] = array('url', 'file');
90 | */
91 | $autoload['helper'] = array('url');
92 |
93 | /*
94 | -----
95 | Auto-load Config files
96 | -----
```

autoload.php-> application/config

```
69 |
70 | The $query_builder variable lets you determine whether or not to load
71 | the query builder class.
72 | */
73 | $active_group = 'default';
74 | $query_builder = TRUE;
75 |
76 | $db['default'] = array(
77 |     'dsn' => '',
78 |     'hostname' => 'localhost',
79 |     'username' => '',
80 |     'password' => '',
81 |     'database' => '',
82 |     'dbdriver' => 'mysqli',
83 |     'dbprefix' => '',
84 |     'pconnect' => FALSE,
85 |     'db_debug' => (ENVIRONMENT !== 'production'),
86 |     'cache_on' => FALSE,
87 |     'cachedir' => '',
88 |     'char_set' => 'utf8',
89 |     'dbcollat' => 'utf8_general_ci',
90 |     'swap_pre' => '',
91 |     'encrypt' => FALSE,
92 |     'compress' => FALSE,
93 |     'stricton' => FALSE,
94 |     'failover' => array(),
95 |     'save_queries' => TRUE
96 | );
97 |
98 |
```

database.php->application/config

```

1 <?php
2 defined('BASEPATH') or exit('No direct script access allowed');
3 class ModelUser extends CI_Model
4 {
5     public function simpanData($data = null)
6     {
7         $this->db->insert('user', $data);
8     }
9     public function cekData($where = null)
10    {
11        return $this->db->get_where('user', $where);
12    }
13    public function getUserWhere($where = null)
14    {
15        return $this->db->get_where('user', $where);
16    }
17    public function cekUserAccess($where = null)
18    {
19        $this->db->select('*');
20        $this->db->from('access_menu');
21        $this->db->where($where);
22        return $this->db->get();
23    }
24    public function getUserLimit()
25    {
26        $this->db->select('*');
27        $this->db->from('user');
28        $this->db->limit(10, 0);
29        return $this->db->get();
30    }
31 }
32

```

modelUser.php->application/model

```

1 <?php
2 defined('BASEPATH') or exit('No direct script access allowed');
3 class ModelBuku extends CI_Model
4 {
5     //manajemen buku
6     public function getBuku()
7     {
8         return $this->db->get('buku');
9     }
10    public function bukuWhere($where)
11    {
12        return $this->db->get_where('buku', $where);
13    }
14    public function simpanBuku($data = null)
15    {
16        $this->db->insert('buku', $data);
17    }
18    public function updateBuku($data = null, $where = null)
19    {
20        $this->db->update('buku', $data, $where);
21    }
22
23    public function hapusBuku($where = null)
24    {
25        $this->db->delete('buku', $where);
26    }
27    public function total($field, $where)
28    {
29        $this->db->select_sum($field);
30        if (!empty($where) && count($where) > 0) {
31            $this->db->where($where);
32        }
33        $this->db->from('buku');
34        return $this->db->get()->row($field);
35    }
36 }
37

```

modelBuku.php->application/model

```

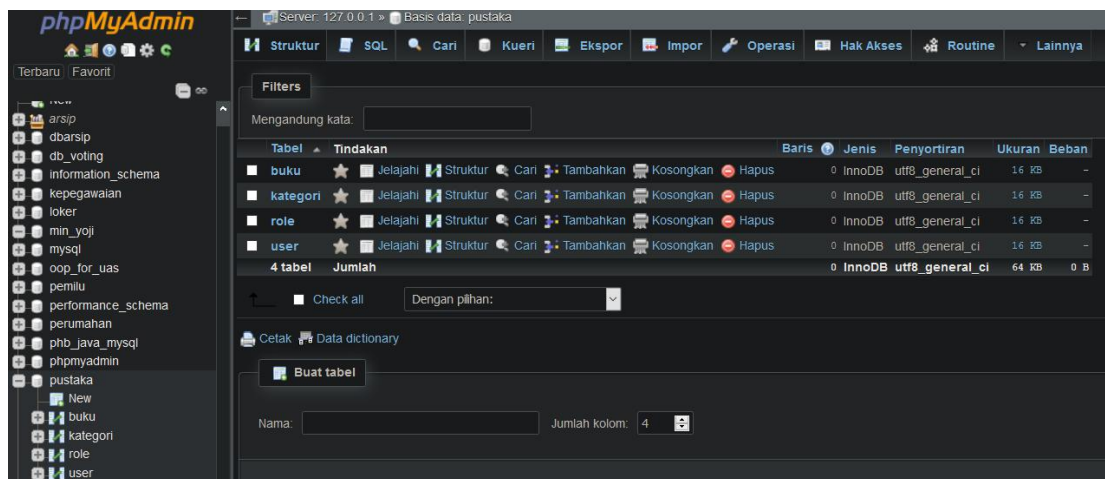
35 }
36 //manajemen kategori
37 public function getKategori()
38 {
39     return $this->db->get('kategori');
40 }
41 public function kategoriWhere($where)
42 {
43     return $this->db->get_where('kategori', $where);
44 }
45 public function simpanKategori($data = null)
46 {
47     $this->db->insert('kategori', $data);
48 }
49 public function hapusKategori($where = null)
50 {
51     $this->db->delete('kategori', $where);
52 }
53 public function updateKategori($where = null, $data = null)
54 {
55     $this->db->update('kategori', $data, $where);
56 }
57 //join
58 public function joinKategoriBuku($where)
59 {
60     $this->db->select('buku.id_kategori,kategori.kategori');
61     $this->db->from('buku');
62     $this->db->join('kategori', 'kategori.id = buku.id_kategori');
63     $this->db->where($where);
64     return $this->db->get();
65 }
66 }
67

```

modelBuku.php->application/model

## Membuat database

- pustaka
- Membuat table user
- Membuat table buku
- Membuat table role
- Membuat table katogori



The screenshot shows the phpMyAdmin interface for a database named 'pustaka'. The left sidebar lists various databases, with 'pustaka' selected. The main area displays the 'Tabel' (Tables) section, showing a list of four tables: 'buku', 'kategori', 'role', and 'user'. Each table entry includes icons for viewing, structure, search, and actions like 'Tambahkan' (Add), 'Kosongkan' (Empty), and 'Hapus' (Delete). The 'user' table is highlighted. Below the table list, there is a 'Cetak' (Print) button and a 'Data dictionary' link. At the bottom, there is a 'Buat tabel' (Create table) button and a form to create a new table with fields for 'Nama' (Name) and 'Jumlah kolom' (Number of columns).