



Mata Kuliah : Pemrograman Web (PW)
Program Studi : D4 – Teknik Informatika / D4 – Sistem Informasi Bisnis
Semester : 3 (tiga)
Pertemuan ke- : 13 (tiga belas)

JOBSHEET 13

Koneksi PHP ke SQL Server (Tambahan)

INFO

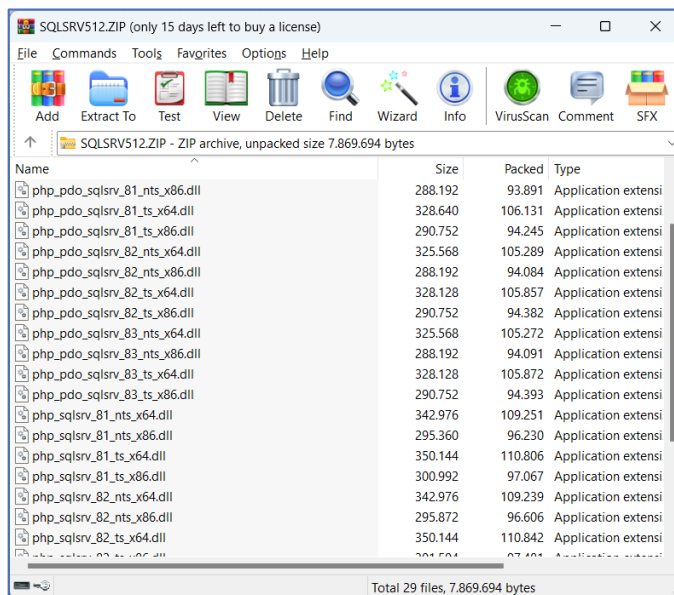
Materi ini adalah materi tambahan dari Jobsheet 13 untuk menghubungkan PHP dengan Database SQL Server.

Jangan melakukan konversi materi PDF ini ke Docx, karena kode program hasil konversi ke Docx bisa tidak sama dengan kode program di materi PDF ini.

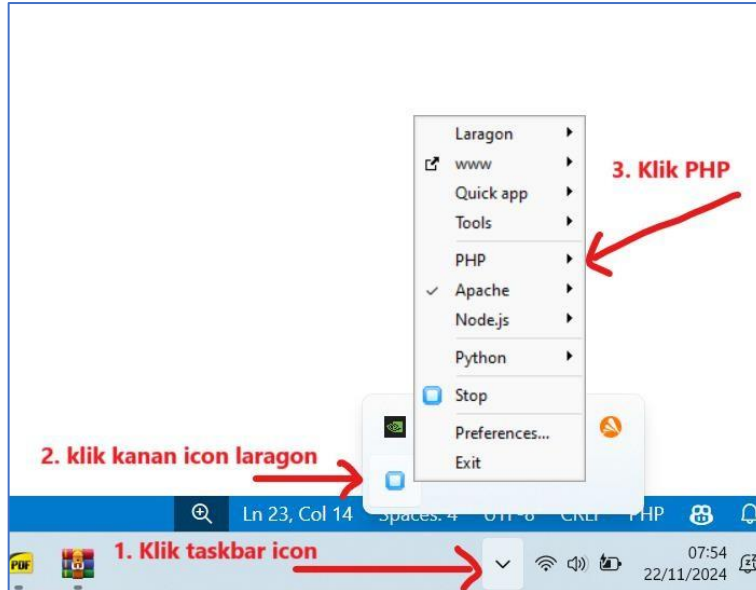
Untuk tabel contoh, kita bisa menggunakan DB dasar_web.sql (MySQL) yang sudah dikonversi ke SQL Server

Praktikum 1 – Instalasi Ekstensi SQL Server pada PHP:

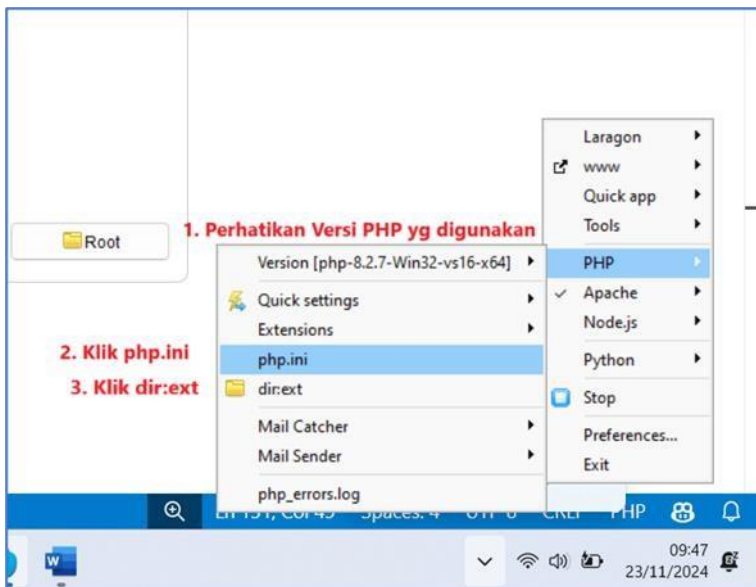
1. Kita download ekstensi **Microsoft Drivers for PHP for SQL Server** disini <https://go.microsoft.com/fwlink/?linkid=2258816>
2. Kemudian kita buka file **SQLSRV512.zip** yang telah kita download tadi



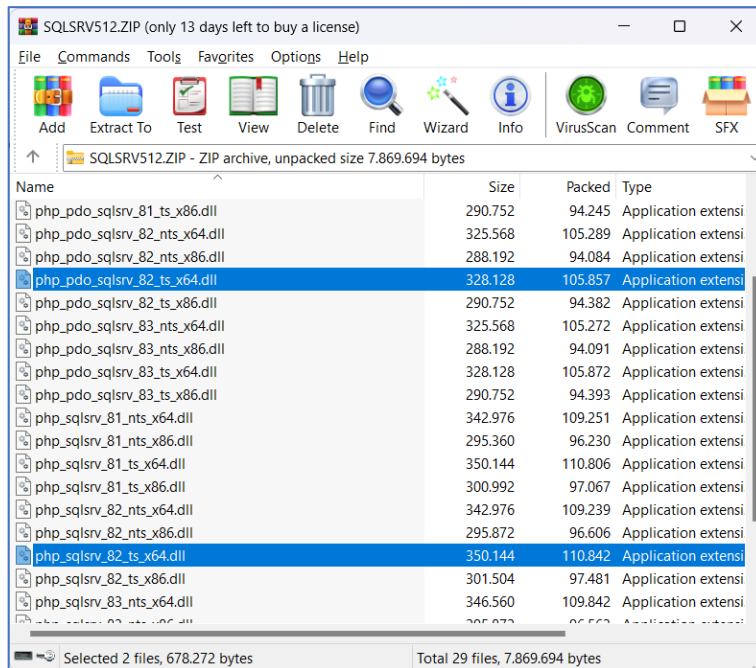
3. Setelah itu, kita cek dulu versi PHP yang kita gunakan di laragon



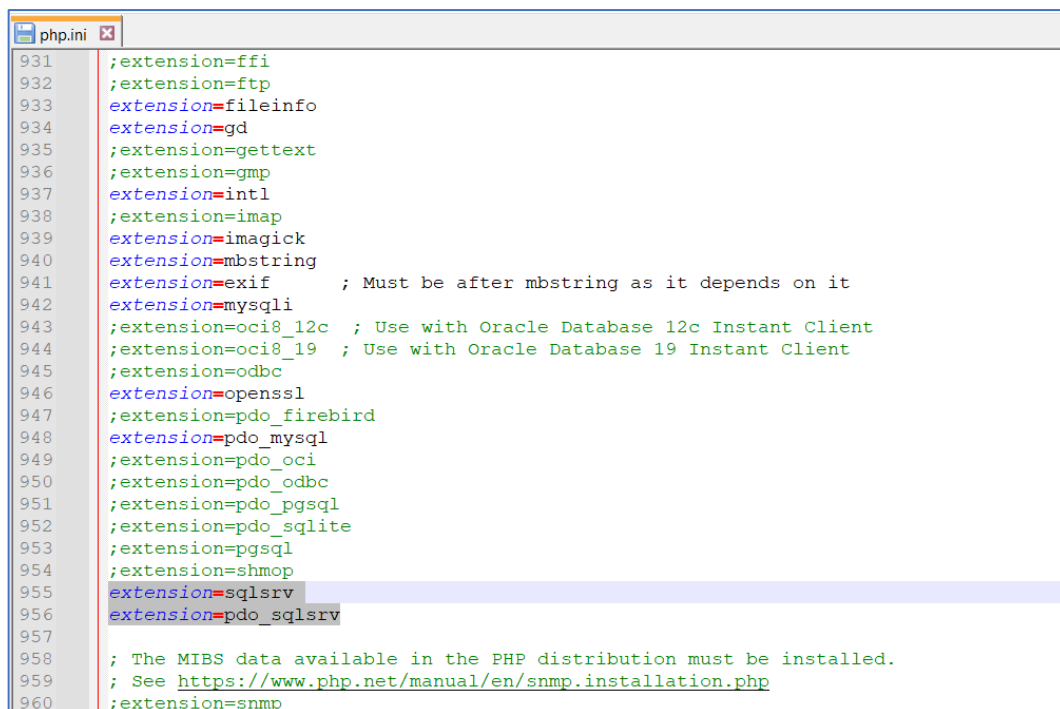
4. Kemudian akan muncul tampilan seperti berikut



5. Dari gambar tahap 4 kita mengetahui bahwa versi PHP yang kita gunakan adalah versi PHP 8.2.7. Sekarang kita copy ekstensi PHP SQL Server sesuai versi PHP ke direktori ekstensi PHP di laragon di `C:\laragon\bin\php\php-8.2.7-Win32-vs16-x64\ext` (direktori bisa berbeda, sesuai dari masing-masing laptop dan versi PHP).
6. Kita pilih `php_sqlsrv_82_ts_x64.dll` dan `php_pdo_sqlsrv_82_ts_x64.dll` yang akan kita masukkan ke ekstensi PHP.



7. Kemudian kita ke file **php.ini** yang sudah kita buka pada tahap 4 tadi. Kemudian kita tambahkan load library ekstensi PHP SQL Server tadi pada baris 955 (atau pada group load ekstensi lain di **php.ini**)



8. Setelah itu, kita restart laragon kita. Selesai



Praktikum 2 – Cek Koneksi ke SQL Server dengan PHP:

1. Setelah kita berhasil melakukan proses instalasi ekstensi PHP SQL Server, kita buat koneksi PHP ke SQL Server
2. Kemudian kita modifikasi file **Connection.php** pada **C:\laragon\www\dasarWeb\JS13\lib\Connection.php**, seperti berikut

```
<?php
use_driver = 'sqlsrv'; // mysql atau sqlsrv

$host      = 'localhost';
$username  = ''; // 'sa';
$password  = '';
$database  = 'master';
$db;

if($use_driver == 'mysql'){
    try{
        $db = new mysqli('localhost', $username, $password, $database);
        if($db->connect_error){
            die('Connection DB failed: ' . $db->connect_error);
        }
    }catch(Exception $e){
        die($e->getMessage());
    }
} else if($use_driver == 'sqlsrv'){
    $credential = [
        'Database' => $database,
        'UID' => $username,
        'PWD' => $password
    ];

    try{
        $db = sqlsrv_connect($host, $credential);

        if (!$db){
            $msg = sqlsrv_errors();
            die($msg[0]['message']);
        }
    }catch(Exception $e){
        die($e->getMessage());
    }
}
```

Kode di atas memungkinkan kita untuk mengubah koneksi ke database MySQL atau SQL Server, sesuai dengan konfigurasi.

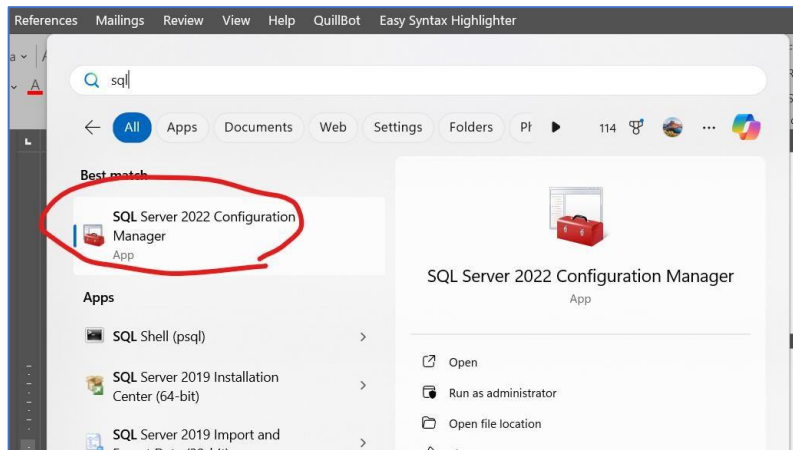
3. Silahkan kalian sesuaikan dengan konfigurasi dengan SQL Server kemudian silahkan dicoba untuk mengakses file Connection.php dari browser

<http://localhost/dasarWeb/JS13/lib/Connection.php>

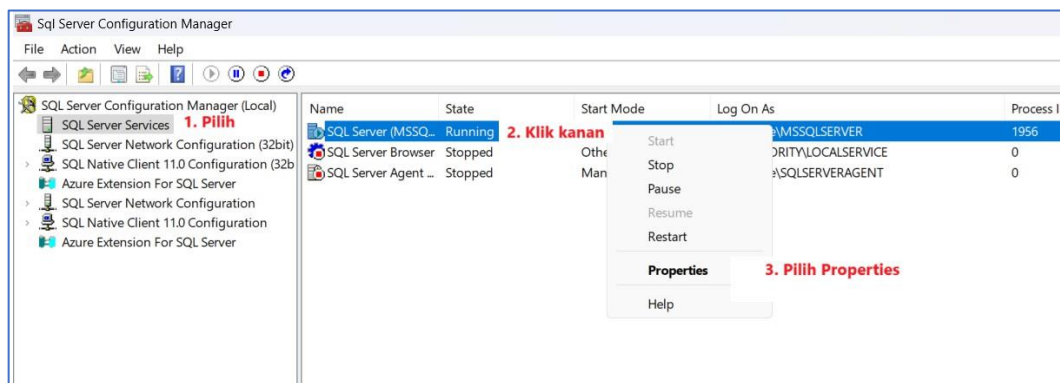
jika tampilan browser adalah **blank**, maka koneksi kita sudah sesuai dan **berhasil**.

Praktikum 2 selesai dan lanjut ke praktikum 3.

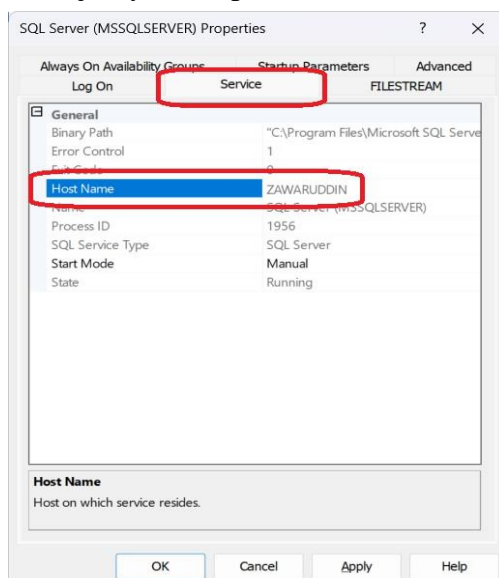
4. Jika koneksi masih tidak berhasil, maka perlu kita ganti konfigurasinya.
5. Kita buka SQL Server Configuration Manager pada Start Windows



6. Setelah terbuka, kita pilih sesuai dengan gambar



7. Selanjutnya, kita pilih tab **Service**, dan kita lihat nilai dari **Host Name**-nya



8. Host Name tersebut yang akan kita gunakan untuk konfigurasi **\$host** pada file **Connection.php** kita



```
Connection.php X
1  <?php
2      $use_driver = 'sqlsrv'; // mysql atau sqlsrv
3
4      $host      = "ZAWARUDDIN"; //'localhost';
5      $username  = ''; //'sa';
6      $password  = '';
7      $database  = 'master';
8      $db;
9
10
11     if($use_driver == 'mysql'){
```

9. Sekarang kita coba lagi untuk cek koneksi ke DB SQL Server dengan mengakses
<http://localhost/dasarWeb/JS13/lib/Connection.php>

Praktikum 3 – Modifikasi Source Code:

1. Untuk bisa mencoba proses CRUD (Create, Read, Update, Delete) pada database SQL Server dengan PHP, kita perlu memodifikasi Source Code dari pertemuan 13 kemarin.
2. Pertama kita buat file baru di

<C:\laragon\www\dasarWeb\JS13\model\UserModel.php>

```
<?php
include('Model.php');

class UserModel extends Model{
    protected $db;
    protected $table = 'm_user';
    protected $driver;

    public function __construct(){
        include('../lib/Connection.php');

        $this->db = $db;
        $this->driver = $use_driver;
    }

    public function insertData($data){
        if($this->driver == 'mysql'){
            // prepare statement untuk query insert
            $query = $this->db->prepare("insert into {$this->table} (username, nama, level,
password) values(?,?,?,?)");

            // binding parameter ke query, "s" berarti string, "ss" berarti dua string
            $query->bind_param('ssss', $data['username'], $data['nama'], $data['level'],
password_hash($data['password'], PASSWORD_DEFAULT));

            // eksekusi query untuk menyimpan ke database
            $query->execute();
        } else {
            // eksekusi query untuk menyimpan ke database
            sqlsrv_query($this->db, "insert into {$this->table} (username, nama, level,
password) values(?,?,?,?)", array($data['username'], $data['nama'], $data['level'],
password_hash($data['password'], PASSWORD_DEFAULT)));
        }
    }
}
```



```
public function getData(){
    if($this->driver == 'mysql'){
        // query untuk mengambil data dari tabel
        return $this->db->query("select * from {$this->table} ")-
>fetch_all(MYSQLI_ASSOC);
    } else {
        // query untuk mengambil data dari tabel
        $query = sqlsrv_query($this->db, "select * from {$this->table}");

        $data = [];
        while($row = sqlsrv_fetch_array($query, SQLSRV_FETCH_ASSOC)){
            $data[] = $row;
        }

        return $data;
    }
}

public function getDataById($id){
    if($this->driver == 'mysql'){
        // query untuk mengambil data berdasarkan id
        $query = $this->db->prepare("select * from {$this->table} where user_id = ?");

        // binding parameter ke query "i" berarti integer. Biar tidak kena SQL Injection
        $query->bind_param('i', $id);

        // eksekusi query
        $query->execute();

        // ambil hasil query
        return $query->get_result()->fetch_assoc();
    } else {
        // query untuk mengambil data berdasarkan id
        $query = sqlsrv_query($this->db, "select * from {$this->table} where user_id =
?", [$id]);

        // ambil hasil query
        return sqlsrv_fetch_array($query, SQLSRV_FETCH_ASSOC);
    }
}

public function updateData($id, $data){
    if($this->driver == 'mysql'){
        // query untuk update data
        $query = $this->db->prepare("update {$this->table} set username = ?, nama = ?,
level = ?, password = ? where user_id = ?");

        // binding parameter ke query
        $query->bind_param('ssssi', $data['username'], $data['nama'], $data['level'],
password_hash($data['password'], PASSWORD_DEFAULT), $id);

        // eksekusi query
        $query->execute();
    } else {
        // query untuk update data
        sqlsrv_query($this->db, "update {$this->table} set username = ?, nama = ?, level
= ?, password = ? where user_id = ?", [$data['username'], $data['nama'], $data['level'],
password_hash($data['password'], PASSWORD_DEFAULT), $id]);
    }
}

public function deleteData($id){
    if($this->driver == 'mysql'){
        // query untuk delete data
        $query = $this->db->prepare("delete from {$this->table} where user_id = ?");
```




```
// binding parameter ke query
$query->bind_param('i', $id);

// eksekusi query
$query->execute();
} else {
    // query untuk delete data
    sqlsrv_query($this->db, "delete from {$this->table} where user_id = ?", [$id]);
}
}

public function getSingleDataByKeyword($column, $keyword){
    if($this->driver == 'mysql'){
        // query untuk mengambil data berdasarkan id
        $query = $this->db->prepare("select * from {$this->table} where {$column} = ?");

        // binding parameter ke query "i" berarti integer. Biar tidak kena SQL Injection
        $query->bind_param('s', $keyword);

        // eksekusi query
        $query->execute();

        return $query->get_result()->fetch_assoc();
    } else {
        // query untuk mengambil data berdasarkan id
        $query = sqlsrv_query($this->db, "select * from {$this->table} where {$column} = ?", [$keyword]);

        // ambil hasil query
        return sqlsrv_fetch_array($query, SQLSRV_FETCH_ASSOC);
    }
}
```

3. Selanjutnya, kita modifikasi **KategoriModel.php** seperti berikut

```
<?php
include('Model.php');

class KategoriModel extends Model{
    protected $db;
    protected $table = 'm_kategori';
    protected $driver;

    public function __construct(){
        include_once('../lib/Connection.php');

        $this->db = $db;
        $this->driver = $use_driver;
    }

    public function insertData($data){
        if($this->driver == 'mysql'){
            // prepare statement untuk query insert
            $query = $this->db->prepare("insert into {$this->table} (kategori_kode, kategori_nama) values(?,?)");

            // binding parameter ke query, "s" berarti string, "ss" berarti dua string
            $query->bind_param('ss', $data['kategori_kode'], $data['kategori_nama']);

            // eksekusi query untuk menyimpan ke database
            $query->execute();
        } else {
            // eksekusi query untuk menyimpan ke database
        }
    }
}
```




```
        sqlsrv_query($this->db, "insert into {$this->table} (kategori_kode,
kategori_nama) values(?,?)", array($data['kategori_kode'], $data['kategori_nama']));
    }
}

public function getData(){
    if($this->driver == 'mysql'){
        // query untuk mengambil data dari tabel
        return $this->db->query("select * from {$this->table} ")-
>fetch_all(MYSQLI_ASSOC);
    } else {
        // query untuk mengambil data dari tabel
        $query = sqlsrv_query($this->db, "select * from {$this->table}");

        $data = [];
        while($row = sqlsrv_fetch_array($query, SQLSRV_FETCH_ASSOC)){
            $data[] = $row;
        }

        return $data;
    }
}

public function getDataById($id){
    if($this->driver == 'mysql'){
        // query untuk mengambil data berdasarkan id
        $query = $this->db->prepare("select * from {$this->table} where kategori_id =
?");

        // binding parameter ke query "i" berarti integer. Biar tidak kena SQL Injection
        $query->bind_param('i', $id);

        // eksekusi query
        $query->execute();

        // ambil hasil query
        return $query->get_result()->fetch_assoc();
    } else {
        // query untuk mengambil data berdasarkan id
        $query = sqlsrv_query($this->db, "select * from {$this->table} where kategori_id
= ?", [$id]);

        // ambil hasil query
        return sqlsrv_fetch_array($query, SQLSRV_FETCH_ASSOC);
    }
}

public function updateData($id, $data){
    if($this->driver == 'mysql'){
        // query untuk update data
        $query = $this->db->prepare("update {$this->table} set kategori_kode = ?,
kategori_nama = ? where kategori_id = ?");

        // binding parameter ke query
        $query->bind_param('ssi', $data['kategori_kode'], $data['kategori_nama'], $id);

        // eksekusi query
        $query->execute();
    } else {
        // query untuk update data
        sqlsrv_query($this->db, "update {$this->table} set kategori_kode = ?,
kategori_nama = ? where kategori_id = ?", [$data['kategori_kode'], $data['kategori_nama'],
$id]);
    }
}
```



```
public function deleteData($id){
    if($this->driver == 'mysql'){
        // query untuk delete data
        $query = $this->db->prepare("delete from {$this->table} where kategori_id = ?");

        // binding parameter ke query
        $query->bind_param('i', $id);

        // eksekusi query
        $query->execute();
    } else {
        // query untuk delete data
        $sqlsrv_query($this->db, "delete from {$this->table} where kategori_id = ?",
[$id]);
    }
}
```

4. Kemudian kita modifikasi file `C:\laragon\www\dasarWeb\JS13\action\auth.php` kita seperti berikut

```
<?php
include('../lib/Session.php');
include('../lib/Connection.php');

$session = new Session();

$act = isset($_GET['act'])? strtolower($_GET['act']) : '';

if($act == 'login'){

    $username = $_POST['username'];
    $password = $_POST['password'];

    include('../model/UserModel.php');
    // digunakan untuk query user
    $user = new UserModel();
    $data = $user->getSingleDataByKeyword('username', $username);

    // jika password sesuai
    if(password_verify($password, $data['password'])){
        $session->set('is_login', true);
        $session->set('username', $data['username']);
        $session->set('name', $data['nama']);
        $session->set('level', $data['level']);
        $session->commit();

        header('Location: ../index.php', false);
    }else{
        $session->setFlash('status', false);
        $session->setFlash('message', 'Username dan password salah. ');
        $session->commit();
        header('Location: ../login.php', false);
    }
}else if($act == 'logout'){

    $session->deleteAll();

    header('Location: ../login.php', false);
}
```



Pertanyaan 1

Kenapa kita perlu memodifikasi file `auth.php`? apa perbedaan dengan yang sebelumnya?

Jawaban:

Modifikasi dilakukan dengan memindahkan proses interaksi dengan database ke dalam UserModel melalui metode `getSingleDataByKeyword` yang sebelumnya tidak ada. Metode ini digunakan untuk mengambil satu baris data dari tabel database berdasarkan nilai tertentu pada kolom yang ditentukan. Pendekatan ini membuat kode lebih terstruktur, modular, dan mudah dikelola.

Pertanyaan 2

Kenapa kita perlu menambahkan file baru `UserModel.php`? apa fungsinya?

Jawaban:

File `UserModel.php` digunakan untuk melakukan manajemen data pengguna yang berada dalam tabel `m_user`. Didalamnya terdapat method seperti menambahkan, mengedit, menghapus, dan mengambil data pengguna dari database. Data ini nantinya dapat digunakan untuk keperluan dalam website, termasuk login. File ini berinteraksi dengan database melalui konfigurasi yang diatur dalam file `Connection.php`.

5. Yang terakhir, kita modifikasi file

`C:\laragon\www\dasarWeb\JS13\action\kategoriAction.php` seperti berikut



```
<?php
include('../lib/Session.php');

$session = new Session();

if($session->get('is_login') !== true){
    header('Location: login.php');
}

include_once('../model/KategoriModel.php');
include_once('../lib/Secure.php');

$act = isset($_GET['act'])? strtolower($_GET['act']) : '';

if($act == 'load'){
    $kategori = new KategoriModel();
    $data = $kategori->getData();
    $result = [];
    $i = 1;
    foreach($data as $row){
        $result['data'][] = [
            $i,
            $row['kategori_kode'],
            $row['kategori_nama'],
            '<button class="btn btn-sm btn-warning"
onclick="editData('.$row['kategori_id'].')"><i class="fa fa-edit"></i></button>
            <button class="btn btn-sm btn-danger"
onclick="deleteData('.$row['kategori_id'].')"><i class="fa fa-trash"></i></button>'
        ];
        $i++;
    }
    echo json_encode($result);
}

if($act == 'get'){
    $id = (isset($_GET['id']) && ctype_digit($_GET['id']))? $_GET['id'] : 0;

    $kategori = new KategoriModel();
    $data = $kategori->getDataById($id);
    echo json_encode($data);
}

if($act == 'save'){
    $data = [
        'kategori_kode' => antiSqlInjection($_POST['kategori_kode']),
        'kategori_nama' => antiSqlInjection($_POST['kategori_nama'])
    ];
}
```



```
$kategori = new KategoriModel();
$kategori->insertData($data);

echo json_encode([
    'status' => true,
    'message' => 'Data berhasil disimpan.'
]);
}

if($act == 'update'){
    $id = (isset($_GET['id']) && ctype_digit($_GET['id']))? $_GET['id'] : 0;
    $data = [
        'kategori_kode' => antiSqlInjection($_POST['kategori_kode']),
        'kategori_nama' => antiSqlInjection($_POST['kategori_nama'])
    ];

    $kategori = new KategoriModel();
    $kategori->updateData($id, $data);

    echo json_encode([
        'status' => true,
        'message' => 'Data berhasil diupdate.'
    ]);
}

if($act == 'delete'){
    $id = (isset($_GET['id']) && ctype_digit($_GET['id']))? $_GET['id'] : 0;

    $kategori = new KategoriModel();
    $kategori->deleteData($id);

    echo json_encode([
        'status' => true,
        'message' => 'Data berhasil dihapus.'
    ]);
}
```

6. Selesai, sekarang kita coba testing aplikasi yang sudah kita modifikasi di

<http://localhost/dasarWeb/JS13>

Pertanyaan 3

Apa perbedaan dengan kode program sebelumnya untuk **KategoriModel.php** dan **kategoriAction.php**? Jelaskan.

Jawaban:

Perbedaan antara kode program sebelumnya dan setelah modifikasi pada KategoriModel.php dan kategoriAction.php terletak pada pemisahan antara logika pengelolaan data dan kontrol aplikasi. Pada KategoriModel.php, setelah modifikasi, kode lebih terstruktur dengan memisahkan operasi CRUD (Create, Read, Update, Delete) ke dalam metode yang jelas dan mengatur koneksi database sesuai dengan driver yang digunakan. Sementara itu, pada kategoriAction.php, file ini lebih fokus pada logika kontrol, seperti mengambil, menyimpan, memperbarui, atau menghapus data, dengan memanfaatkan KategoriModel untuk interaksi dengan database.

Pertanyaan 4

Silahkan implementasikan juga untuk menu Buku.



Jawaban:
BukuAction

```
Connection.php M X UserModel.php kategoriAction.php bukuAction.php X KategoriModel.php buku.php M auth.php
JS13_AdminLTE > action > bukuAction.php > ...
1 <?php
2
3 include('../lib/Session.php');
4 $session = new Session();
5 if ($session->get(key: 'is_login') != true) {
6     header(header: 'Location: login.php');
7 }
8 include_once('../model/BukuModel.php');
9 include_once('../lib/Secure.php');
10 $act = isset($_GET['act']) ? strtolower(string: $_GET['act']) : '';
11
12 if ($act == 'load') {
13     $buku = new BukuModel();
14     $data = $buku->getData();
15     $result = [];
16     $i = 1;
17     foreach ($data as $row) {
18         $result['data'][] = [
19             $i,
20             $row['kategori_id'],
21             $row['buku_kode'],
22             $row['buku_nama'],
23             $row['jumlah'],
24             $row['deskripsi'],
25             'img src="'. htmlspecialchars(string: $row['gambar']) . '" alt="gambar" style="width: 100px; height: auto;"',
26             '<button class="btn btn-sm btn-warning"
27             onclick="editData(' . $row['buku_id'] . ') "><i class="fa fa-edit"></i></button>
28             <button class="btn btn-sm btn-danger"
29             onclick="deleteData(' . $row['buku_id'] . ') "><i class="fa fa-trash"></i></button>'
30         ];
31         $i++;
32     }
33     echo json_encode(value: $result);
34 }
35
36 if ($act == 'get') {
37     $id = (isset($_GET['id']) && ctype_digit(text: $_GET['id'])) ? $_GET['id'] : 0;
38     $buku = new BukuModel();
39     $data = $buku->getDataById(id: $id);
40
41     if (empty($data['deskripsi'])) {
42         $data['deskripsi'] = 'Belum ada deskripsi';
43     }
44     echo json_encode(value: $data);
45 }
46
47 if ($act == 'save') {
48     $data = [
49         'kategori_id' => (int)antiSqlInjection(data: $_POST['kategori_id']),
50         'buku_kode' => antiSqlInjection(data: $_POST['buku_kode']),
51         'buku_nama' => antiSqlInjection(data: $_POST['buku_nama']),
52         'jumlah' => (int)antiSqlInjection(data: $_POST['jumlah']),
53         'deskripsi' => antiSqlInjection(data: $_POST['deskripsi']),
54         'gambar' => antiSqlInjection(data: $_POST['gambar'])
55     ];
56     $buku = new BukuModel();
57     $buku->insertData(data: $data);
58     echo json_encode(value: [
59         'status' => true,
60         'message' => 'Data berhasil disimpan.'
61     ]);
62 }
63
64 if ($act == 'update') {
65     $id = (isset($_GET['id']) && ctype_digit(text: $_GET['id'])) ? $_GET['id'] : 0;
66     $data = [
67         'kategori_id' => (empty($_POST['kategori_id']) ? (int)antiSqlInjection(data: $_POST['kategori_id']) : null,
68         'buku_kode' => antiSqlInjection(data: $_POST['buku_kode']),
69         'buku_nama' => antiSqlInjection(data: $_POST['buku_nama']),
70         'jumlah' => (empty($_POST['jumlah']) ? (int)antiSqlInjection(data: $_POST['jumlah']) : 0,
71         'deskripsi' => (empty($_POST['deskripsi']) ? antiSqlInjection(data: $_POST['deskripsi']) : 'Tidak ada deskripsi',
72         'gambar' => antiSqlInjection(data: $_POST['gambar'])
73     ];
74     if (is_null(value: $data['kategori_id'])) {
75         echo json_encode(value: [
76             'status' => false,
77             'message' => 'Harus diisi.'
78         ]);
79     }
80     exit;
81 }
```



```
Connection.php M UserModel.php kategoriAction.php bukuAction.php M X KategoriModel.php
JS13_AdminLTE > action > bukuAction.php > ...
79 |
80 | $buku = new BukuModel();
81 | $buku->updateData(id: $id, data: $data);
82 | echo json_encode(value: [
83 |     'status' => true,
84 |     'message' => 'Data berhasil diupdate.'
85 | ]);
86 | }
87 |
88 | if ($sact == 'delete') {
89 |     $id = (isset($_GET['id']) && ctype_digit(text: $_GET['id'])) ? $_GET['id'] : 0;
90 |     $buku = new BukuModel();
91 |     $buku->deleteData(id: $id);
92 |     echo json_encode(value: [
93 |         'status' => true,
94 |         'message' => 'Data berhasil dihapus.'
95 |     ]);
96 | }
```

BukuModel

```
Connection.php M UserModel.php kategoriAction.php BukuModel.php M X KategoriModel.php buku.php M auth.php
JS13_AdminLTE > model > BukuModel.php > PHP Intelephense > BukuModel
1 <?php
2 include('Model.php');
3 class BukuModel extends Model
4 {
5     11 references
6     private $db;
7     10 references
8     private $table = 'm_buku';
9     7 references
10    protected $driver;
11
12    5 references | 0 overrides
13    public function __construct()
14    {
15        include_once('../lib/Connection.php');
16        $this->db = $db;
17        // $this->db->set_charset('utf8');
18        $this->driver = $use_driver;
19    }
20    2 references | 0 overrides | prototype
21    public function insertData($data): void
22    {
23        if ($this->driver == 'mysql') {
24            // Menggunakan MySQLi
25            $query = $this->db->prepare("INSERT INTO {$this->table} (kategori_id, buku_kode, buku_nama, jumlah, deskripsi, gambar)
26            VALUES (?, ?, ?, ?, ?, ?)");
27            // Binding parameter ke query
28            $query->bind_param('ississ',
29            $data['kategori_id'],
30            $data['buku_kode'],
31            $data['buku_nama'],
32            $data['jumlah'],
33            $data['deskripsi'],
34            $data['gambar']
35        );
36    }
```




KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI
POLITEKNIK NEGERI MALANG
JURUSAN TEKNOLOGI INFORMASI
Jl. Soekarno Hatta No. 9, Jatimulyo, Lowokwaru, Malang 65141
Telp. (0341) 404424 – 404425, Fax (0341) 404420
<http://www.polinema.ac.id>

```
JS13_AdminLTE > model > BukuModel.php > PHP Intelephense > BukuModel
3  class BukuModel extends Model
17  public function insertData($data): void
35      // Eksekusi query
36      if (!$query->execute()) {
37          die('MySQL Insert Error: ' . $query->error);
38      }
39
40      } else if ($this->driver == 'sqlsrv') {
41          // Menggunakan SQLSRV
42          $sql = "INSERT INTO {$this->table} (kategori_id, buku_kode, buku_nama, jumlah, deskripsi, gamb
43              VALUES (?, ?, ?, ?, ?, ?)";
44          $params = [
45              $data['kategori_id'],
46              $data['buku_kode'],
47              $data['buku_nama'],
48              $data['jumlah'],
49              $data['deskripsi'],
50              $data['gambar']
51          ];
52
53          // Eksekusi query
54          $stmt = sqlsrv_query(conn: $this->db, sql: $sql, params: $params);
55
56          if ($stmt === false) {
57              die(print_r(value: sqlsrv_errors(), return: true));
58          }
59      } else {
60          die('Unsupported database driver.');
```

2 references | 0 overrides | prototype

```
63  public function getData(): mixed{
64      if ($this->driver == 'mysql') {
65          // fetch data
66          return $this->db->query("SELECT * FROM {$this->table}")->fetch_all(MYSQLI_ASSOC);
67      } else {
68          // fetch data
69          $query = sqlsrv_query(conn: $this->db, sql: "SELECT * FROM {$this->table}");
70          $data = [];
```



```
Connection.php M UserModel.php kategoriAction.php BukuModel.php M X KategoriModel.php buku.php M auth
JS13_AdminLTE > model > BukuModel.php > PHP Intelephense > BukuModel > getData
3 class BukuModel extends Model
63 public function getData(): mixed{
71     while ($row = sqlsrv_fetch_array($stmt: $query, fetchType: SQLSRV_FETCH_ASSOC)) {
72         $data[] = $row;
73     }
74     return $data;
75 }
76
77
2 references | 0 overrides | prototype
78 public function getDataById($id): mixed{
79     if ($this->driver == 'mysql') {
80         // query untuk mengambil data berdasarkan id
81         $query = $this->db->prepare("select * from {$this->table} where buku_id = ?");
82         // binding parameter ke query "i" berarti integer. Biar tidak kena SQL Injection
83         $query->bind_param('i', $id);
84         // eksekusi query
85         $query->execute();
86         // ambil hasil query
87         return $query->get_result()->fetch_assoc();
88     } else {
89         $query = sqlsrv_query($conn: $this->db, sql: "SELECT * FROM {$this->table} WHERE buku_id = ?", params: [$id]);
90         return sqlsrv_fetch_array($stmt: $query, fetchType: SQLSRV_FETCH_ASSOC);
91     }
92 }
93
2 references | 0 overrides | prototype
94 public function updateData($id, $data): void
95 {
96     if ($this->driver == 'mysql') {
97         // query untuk update data
98         $query = $this->db->prepare("UPDATE {$this->table} SET
99             kategori_id = ?,
100             buku_kode = ?,
101             buku_nama = ?,
102             jumlah = ?,
103             deskripsi = ?,
104             gambar = ?
105             WHERE buku_id = ?");
106
107         // binding parameter ke query
108         $query->bind_param('ississi',
109             $data['kategori_id'],
110             $data['buku_kode'],
111             $data['buku_nama'],
112             $data['jumlah'],
113             $data['deskripsi'],
114             $data['gambar'],
115             $id
116         );
117
118         // eksekusi query
119         $query->execute();
120     } else {
121         // query untuk update data untuk sqlsrv
122         $sql = "UPDATE {$this->table} SET
123             kategori_id = ?,
124             buku_kode = ?,
125             buku_nama = ?,
126             jumlah = ?,
127             deskripsi = ?,
128             gambar = ?
129             WHERE buku_id = ?";
130
131         $params = [
132             $data['kategori_id'],
133             $data['buku_kode'],
134             $data['buku_nama'],
135             $data['jumlah'],
136             $data['deskripsi'],
137             $data['gambar'],
138             $id
139         ];
140
141         // eksekusi query menggunakan sqlsrv_query
142         $stmt = sqlsrv_query($conn: $this->db, sql: $sql, params: $params);
143     }
```



```
JS13_AdminLTE > model > BukuModel.php > PHP Intelephense > BukuModel > updateData
3 class BukuModel extends Model
94 public function updateData($id, $data): void
144     // cek jika query gagal
145     if ($stmt === false) {
146         die(print_r(value: sqlsrv_errors(), return: true));
147     }
148 }
149 }
150
2 references | 0 overrides | prototype
151 public function deleteData($id): void{
152     if ($this->driver == 'mysql') {
153         // MySQL: Prepare delete query
154         $query = $this->db->prepare("DELETE FROM {$this->table} WHERE buku_id = ?");
155         // Bind parameter
156         $query->bind_param('i', $id);
157         // Execute the query
158         $query->execute();
159     } else {
160         // SQL Server: Execute delete query
161         sqlsrv_query(conn: $this->db, sql: "DELETE FROM {$this->table} WHERE buku_id = ?", params: [$id]);
162     }
163 }
164 }
```

Buku

```
JS13_AdminLTE > pages > buku.php > PHP Intelephense > script
1 <?php
2 require_once __DIR__ . '/../lib/Connection.php';
3 function getKategori(): array
4 {
5     global $db;
6     $query = "SELECT * FROM m_kategori ORDER BY kategori_nama ASC";
7     $result = sqlsrv_query(conn: $db, sql: $query);
8     $kategori = [];
9     if ($result) {
10         while ($row = sqlsrv_fetch_array(stmt: $result, fetchType: SQLSRV_FETCH_ASSOC)) {
11             $kategori[] = $row;
12         }
13     } else {
14         die(print_r(value: sqlsrv_errors(), return: true));
15     }
16     return $kategori;
17 }
18 $kategori = getKategori();
19 >>
20
21 <section class="content-header">
22     <div class="container-fluid">
23         <div class="row mb-2">
24             <div class="col-sm-6">
25                 <h1>Buku</h1>
26             </div>
27             <div class="col-sm-6">
28                 <ol class="breadcrumb float-sm-right">
29                     <li class="breadcrumb-item"><a href="#">Home</a></li>
30                     <li class="breadcrumb-item active">Buku</li>
31                 </ol>
32             </div>
33         </div>
34     </div>
35 </section>
36 <!-- Main content -->
37 <section class="content">
38     <div class="card">
```



KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI
POLITEKNIK NEGERI MALANG
JURUSAN TEKNOLOGI INFORMASI
Jl. Soekarno Hatta No. 9, Jatimulyo, Lowokwaru, Malang 65141
Telp. (0341) 404424 – 404425, Fax (0341) 404420
<http://www.polinema.ac.id>

```
<div class="card-header">
  <h3 class="card-title">Daftar Buku</h3>
  <div class="card-tools">
    <button type="button" class="btn btn-md btn-primary" onclick="tambahData()">
      Tambah
    </button>
  </div>
</div>
<div>
  <div class="card-body">
    <table class="table table-sm table-bordered table-striped" id="table-data">
      <thead>
        <tr>
          <th style="width: 5%;>No</th>
          <th style="width: 5%;>Kategori Buku</th>
          <th style="width: 10%;>Kode Buku</th>
          <th style="width: 15%;>Nama Buku</th>
          <th style="width: 5%;>Jumlah</th>
          <th style="width: 38%;>Deskripsi</th>
          <th style="width: 5%;>Gambar</th>
          <th>Aksi</th>
        </tr>
      </thead>
      <tbody>
      </tbody>
    </table>
  </div>
</div>
</section>
```



```
67 <div class="modal fade" id="form-data" style="display: none;" aria-hidden="true">
68   <form action="action/BukuAction.php?act=save" method="post" id="form-tambah">
69     <!-- Ukuran Modal
70     modal-sm : Modal ukuran kecil
71     modal-md : Modal ukuran sedang
72     modal-lg : Modal ukuran besar
73     modal-xl : Modal ukuran sangat besar
74     penerapan setelah class modal-dialog seperti di bawah
75   -->
76   <div class="modal-dialog modal-md">
77     <div class="modal-content">
78       <div class="modal-header">
79         <h4 class="modal-title">Tambah Buku</h4>
80       </div>
81       <div class="modal-body">
82         <div class="form-group">
83           <label>Kategori Buku</label>
84           <select id="kategori_id" name="kategori_id" class="form-control">
85             <?php if (empty($kategori)): ?>
86               <?php foreach ($kategori as $id): ?>
87                 <option value="<?> htmlspecialchars(string: $id['kategori_id']); ?>"
88                   <?> htmlspecialchars(string: $id['kategori_nama']); ?>
89                 </option>
90               <?php endforeach; ?>
91             <?php else: ?>
92               <option value=""></option>
93             <?php endif; ?>
94           </select>
95         </div>
96         <div class="form-group">
97           <label>Kode Buku</label>
98           <input type="text" class="form-control" name="buku_kode"
99             id="buku_kode">
100         </div>
101         <div class="form-group">
102           <label>Nama Buku</label>
103           <input type="text" class="form-control" name="buku_nama"
104             id="buku_nama">
105         </div>
106         <div class="form-group">
107           <label>Jumlah</label>
108           <input type="text" class="form-control" name="jumlah"
109             id="jumlah">
110         </div>
111         <div class="form-group">
112           <label>Deskripsi</label>
113           <input type="text" class="form-control" name="deskripsi"
114             id="deskripsi">
115         </div>
116         <div class="form-group">
117           <label>Gambar</label>
118           <input type="text" class="form-control" name="gambar"
119             id="gambar">
120         </div>
121       </div>
122       <div class="modal-footer justify-content-between">
123         <button type="button" class="btn btn-default" data-dismiss="modal">Batal</button>
124         <button type="submit" class="btn btn-primary">Simpan</button>
125       </div>
126     </div>
127   </div>
128 </form>
129 </div>
130 </script>
```



KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI
POLITEKNIK NEGERI MALANG
JURUSAN TEKNOLOGI INFORMASI
Jl. Soekarno Hatta No. 9, Jatimulyo, Lowokwaru, Malang 65141
Telp. (0341) 404424 – 404425, Fax (0341) 404420
<http://www.polinema.ac.id>

```
131 function tambahData() {
132     $('#form-data').modal('show');
133     $('#form-tambah').attr('action', 'action/BukuAction.php?act=save');
134     $('#kategori_id').val('');
135     $('#buku_kode').val('');
136     $('#buku_nama').val('');
137     $('#jumlah').val('');
138     $('#deskripsi').val('');
139     $('#gambar').val('');
140 }
141
142 function editData(id) {
143     $.ajax({
144         url: 'action/BukuAction.php?act=get&id=' + id,
145         method: 'post',
146         success: function(response) {
147             var data = JSON.parse(response);
148             $('#form-data').modal('show');
149             $('#form-tambah').attr('action',
150                 'action/BukuAction.php?act=update&id=' + id);
151             $('#kategori_id').val(data.kategori_id).trigger('change');
152             $('#buku_kode').val(data.buku_kode);
153             $('#buku_nama').val(data.buku_nama);
154             $('#jumlah').val(data.jumlah);
155             $('#deskripsi').val(data.deskripsi);
156             $('#gambar').val(data.gambar);
157         }
158     });
159 }
160
161 function deleteData(id) {
162     if (confirm('Apakah anda yakin?')) {
163         $.ajax({
164             url: 'action/BukuAction.php?act=delete&id=' + id,
165             method: 'post',
166             success: function(response) {
167                 var result = JSON.parse(response);
168                 if (result.status) {
169                     tabelData.ajax.reload();
170                 } else {
171                     alert(result.message);
172                 }
173             }
174         });
175     }
176 }
177 var tabelData;
178 $(document).ready(function() {
179     tabelData = $('#table-data').DataTable({
180         ajax: 'action/BukuAction.php?act=load',
181     });
182     $('#form-tambah').validate({
183         rules: {
184             kategori_id: {
185                 required: true,
186                 minlength: 1
187             },
188             buku_kode: {
189                 required: true,
190                 minlength: 2
191             },
192             buku_nama: {
193                 required: true,
194                 minlength: 5
195             },
196         }
197     });
198 }
```



```
196     jumlah: {  
197         required: true,  
198         minlength: 1  
199     },  
200     deskripsi: {  
201         required: true,  
202         minlength: 5  
203     },  
204     gambar: {  
205         required: true,  
206         minlength: 5  
207     },  
208 },  
209 errorElement: 'span',  
210 errorPlacement: function(error, element) {  
211     error.addClass('invalid-feedback');  
212     element.closest('.form-group').append(error);  
213 },  
214 highlight: function(element, errorClass, validClass) {  
215     $(element).addClass('is-invalid');  
216 },  
217 unhighlight: function(element, errorClass, validClass) {  
218     $(element).removeClass('is-invalid');  
219 },  
220 submitHandler: function(form) {  
221     $.ajax({  
222         url: $(form).attr('action'),  
223         method: 'post',  
224         data: $(form).serialize(),  
225         success: function(response) {  
226             var result = JSON.parse(response);  
227             if (result.status) {  
228                 $('#form-data').modal('hide');  
229                 tabelData.ajax.reload(); // reload data tabel  
230             } else {  
231                 alert(result.message);  
232             }  
233         }  
234     });  
235 }  
236 });  
237 });  
238 </script>
```

Hasil running



KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI
POLITEKNIK NEGERI MALANG
JURUSAN TEKNOLOGI INFORMASI
Jl. Soekarno Hatta No. 9, Jatimulyo, Lowokwaru, Malang 65141
Telp. (0341) 404424 – 404425, Fax (0341) 404420
<http://www.polinema.ac.id>

localhost/Jobsheet_Web/JS13_AdminLTE/index.php?page=buku

AdminLTE 3
Alexander Pierce

Search

Dashboard
Kategori Buku
Buku
Logout

Home Contact

Buku

Daftar Buku Tambah

Show 10 entries Search:

No	Kategori Buku	Kode Buku	Nama Buku	Jumlah	Deskripsi	Gambar	Aksi
1	3	FISIKA	Fisika Quantum	3	Buku Fisika Quantum		
2	8	MTK	Matematika Dasar	3	Dasar belajar matematika, panduan		

Showing 1 to 2 of 2 entries

Previous 1 Next

*** Sekian, dan selamat belajar ***