



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

JOBSHEET 13

Web Templating (AdminLTE) dan CRUD

INFO

Materi ini adalah materi tambahan, jadi perhatikan dan ikuti tahap demi tahap dari materi ini.

A. Web Template

Merupakan suatu file yang sudah memiliki desain tertentu sehingga, kita tinggal memodifikasi bagian-bagian tertentu saja untuk mendapatkan hasil yang sudah bagus. Dengan menggunakan template yang sesuai dengan kebutuhan, kita bisa mengurangi waktu pembuatan suatu website secara drastis.

Secara umum kita bisa menggunakan template website yang sudah ada di internet untuk mempermudah kita dalam membuat website. Ada banyak sekali template yang ada di internet dan kita ambil satu contoh template untuk halaman administrator, seperti **AdminLTE** template. **AdminLTE** adalah salah satu template yang sering digunakan oleh web developer sebagai template backend/admin pada proyek yang sering dikerjakan. Template ini dibuat menggunakan framework bootstrap yang merupakan framework CSS yang paling banyak digunakan di kalangan web desainer

Template AdminLTE dapat di download di


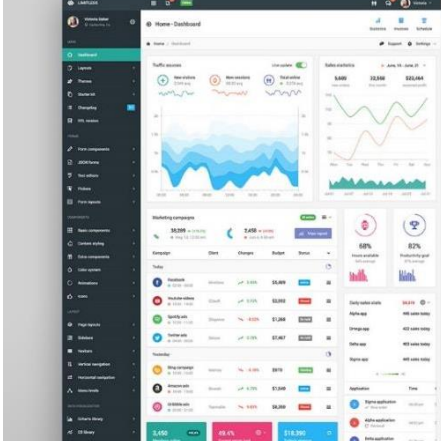


<https://github.com/ColorlibHQ/AdminLTE/archive/refs/tags/v3.2.0.zip>

diakses pada (17 November 2024)

Kita pakai **AdminLTE Versi 3.2.0**



Contoh Web template

Frontend Template	Backend Template
	
	

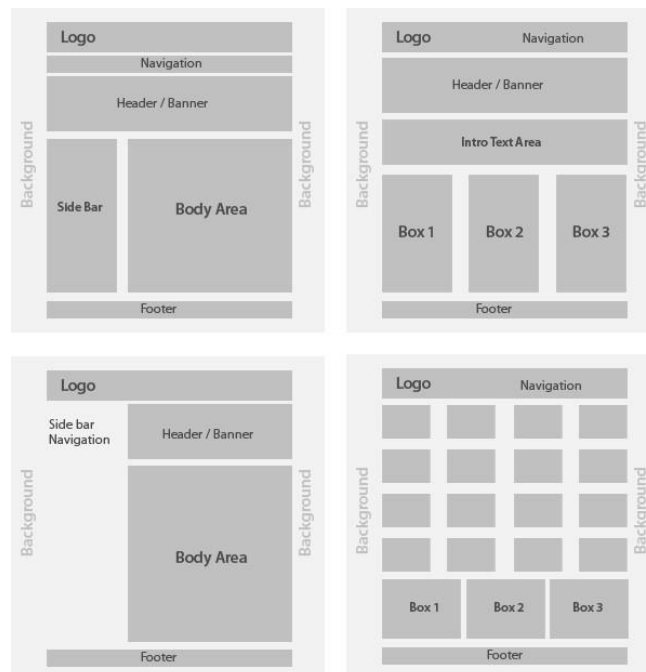
B. Web Layout

Merupakan sekumpulan elemen atau variabel desain yang terkait dengan media tertentu untuk mendukung konsep pembuatan website. Pada dasarnya tujuan penggunaan layout website adalah untuk menggabungkan elemen yang digunakan untuk membangun website. Selain elemen di dalamnya, layout website juga harus memuat beberapa elemen, mulai dari header, navigasi, sidebar, hingga footer. Untuk memahami perbedaan elemen sebuah layout website, berikut penjelasan singkatnya

- **Header** → Pada bagian header, layout designer dapat mengisi bagian ini dengan logo website, navigasi situs, ikon media sosial, dan menu pencarian.



- **Navigasi** → Navigasi dapat dipahami sebagai panduan. Di website, navigasi dapat berkisar dari menu yang muncul di bagian atas halaman web hingga menu pendukung lainnya yang biasa ditemukan di bawah situs.
- **Body/Konten** → Selain itu, terdapat elemen body/konten yang biasanya diisi dengan informasi produk, fitur produk, dan deskripsi produk yang dijual.
- **Sidebar** → elemen ini merupakan elemen yang berada di samping kanan/kiri sebuah website. Element ini kadang sudah tidak dipakai, terutama pada Web Frontend. Namun, sidebar masih digunakan di banyak halaman artikel, yang kemudian akan diisi dengan informasi produk, produk terpopuler, dan navigasi tambahan.
- **Footer** → Elemen terakhir adalah footer. Footer merupakan elemen yang berada dibawah sendiri sebagai menu pendukung atau sebagai identitas dari sebuah website.



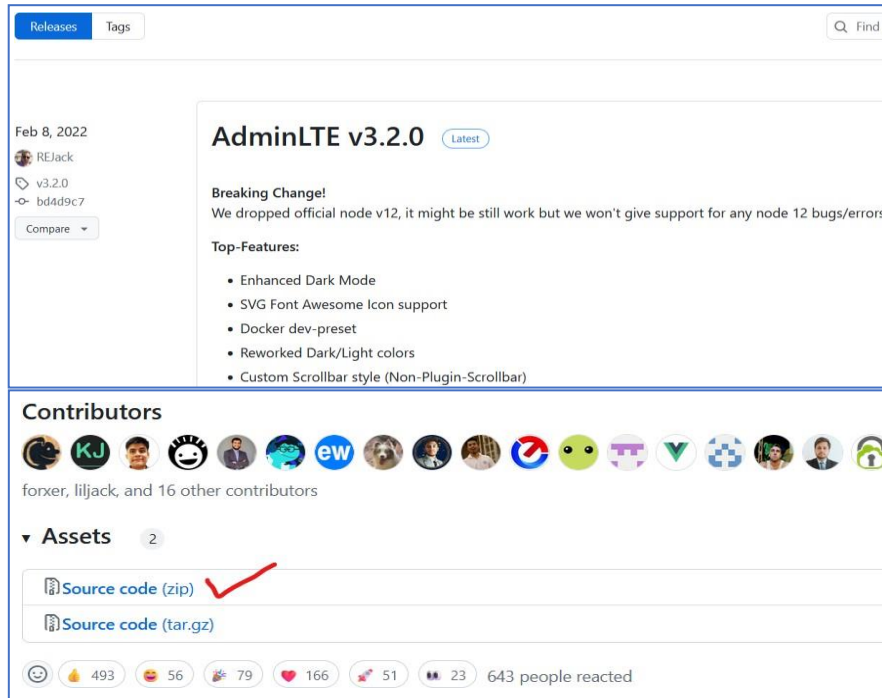
C. **Layouting AdminLTE**

Pada kali ini kita akan melakukan *layouting* template AdminLTE. Tujuan dari *layouting* ini adalah untuk memecah bagian-bagian dari template adminLTE menjadi berbagai elemen. Hal ini dapat mempermudah kita dalam membuat website karena kita bisa menggunakan berulang-ulang elemen yang sudah kita buat nantinya.

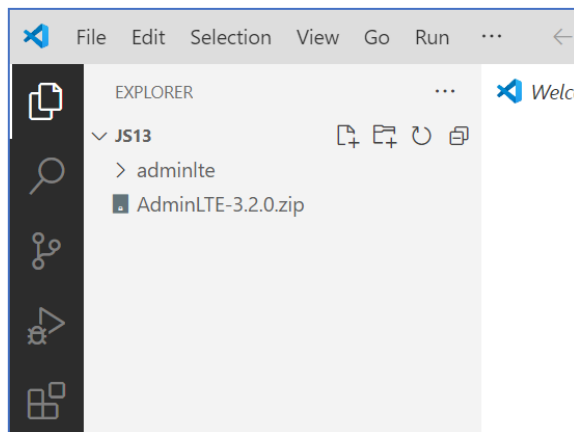


Praktikum 1 - Penerapan AdminLTE (Login):

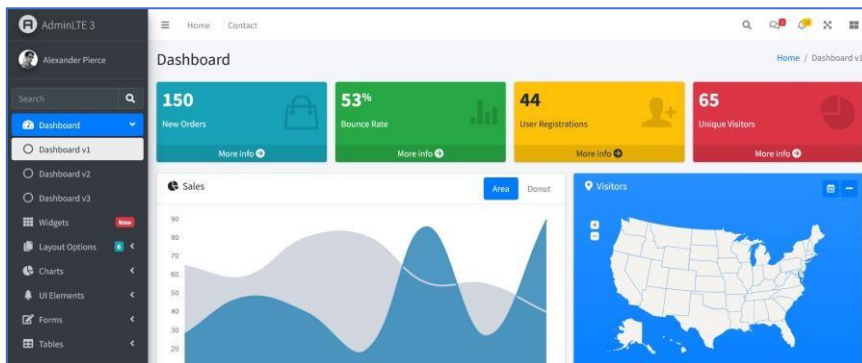
1. Kita download **AdminLTE v3.2.0** yang rilis pada 8 Feb 2022



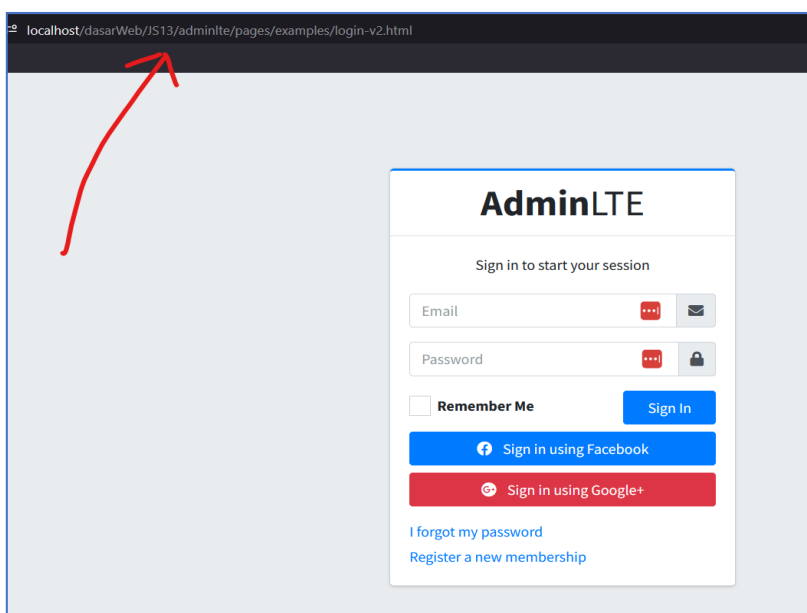
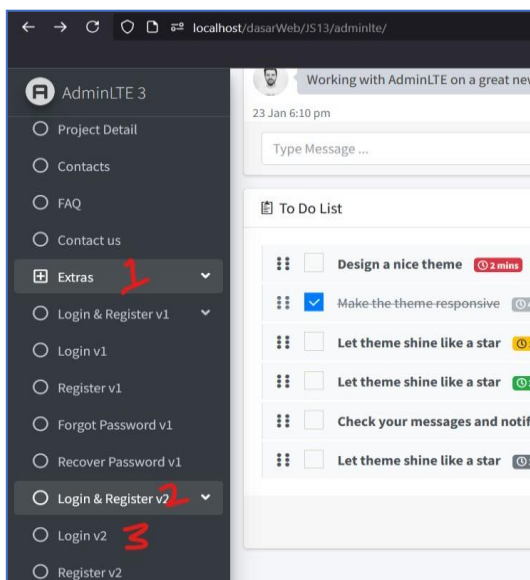
2. Setelah kita berhasil download, kita ekstrak file yang sudah di download ke folder project `C:\laragon\www\dasarWeb\JS13`, kemudian kita **rename** folder cukup menjadi **adminlte**



3. Kita buat database baru pada phpmyadmin dengan nama **dasar_web**. Kemudian kita **import** file database **dasar_web.sql** pada phpmyadmin.
4. Selanjutnya kita buka di browser dengan alamat <http://localhost/dasarWeb/adminlte> maka akan muncul tampilan seperti berikut

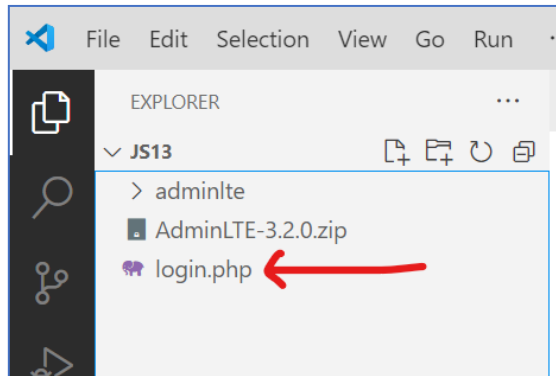


5. Kita klik menu **Extras > Login & Register V2 > Login V2**, kita gunakan template ini untuk halaman login





6. Kita copy file
`C:\laragon\www\dasarWeb\JS13\adminlte\pages\examples\login-v2.html`,
dan kita paste ke `C:\laragon\www\dasarWeb\JS13`. Kemudian kita rename menjadi
`login.php`



7. Kemudian kita edit file `login.php` tersebut seperti berikut

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

$session = new Session();

if($session->get('is_login') === true){
    header('Location: index.php');
}
?>
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <title>AdminLTE 3 | Log in (v2)</title>

    <!-- Google Font: Source Sans Pro -->
    <link rel="stylesheet"
href="https://fonts.googleapis.com/css?family=Source+Sans+Pro:300,400,400i,700&display=fallb
ack">
    <!-- Font Awesome -->
    <link rel="stylesheet" href="adminlte/plugins/fontawesome-free/css/all.min.css">
    <!-- icheck bootstrap -->
    <link rel="stylesheet" href="adminlte/plugins/icheck-bootstrap/icheck-bootstrap.min.css">
    <!-- Theme style -->
    <link rel="stylesheet" href="adminlte/dist/css/adminlte.min.css">
</head>
<body class="hold-transition login-page">
<div class="login-box">
    <!-- /.login-logo -->
    <div class="card card-outline card-primary">
        <div class="card-header text-center"><a href="/" class="h1"><b>Admin</b>LTE</a></div>
        <div class="card-body">
            <p class="login-box-msg">Sign in to start your session</p>
            <?php
                $status = $session->getFlash('status');

                if($status === false){
                    $message = $session->getFlash('message');
                    echo '<div class="alert alert-warning">'. $message .
```



```
'<button type="button" class="close" data-dismiss="alert" aria-  
label="Close"><span aria-hidden="true">&times;</span></div>';  
}  
>?  
<form action="action/auth.php?act=login" method="post" id="form-login">  
  <div class="input-group mb-3">  
    <input type="text" class="form-control" name="username" placeholder="Username">  
    <div class="input-group-append">  
      <div class="input-group-text">  
        <span class="fas fa-user"></span>  
      </div>  
    </div>  
  </div>  
  <div class="input-group mb-3">  
    <input type="password" class="form-control" name="password"  
placeholder="Password">  
    <div class="input-group-append">  
      <div class="input-group-text">  
        <span class="fas fa-lock"></span>  
      </div>  
    </div>  
  </div>  
  <div class="row">  
    <div class="col-8">  
      <div class="icheck-primary">  
        <input type="checkbox" id="remember">  
        <label for="remember">Remember Me</label>  
      </div>  
    </div>  
    <!-- /.col -->  
    <div class="col-4">  
      <button type="submit" class="btn btn-primary btn-block">Sign In</button>  
    </div>  
    <!-- /.col -->  
  </div>  
</form>  
</div>  
<!-- /.card-body -->  
</div>  
<!-- /.card -->  
</div>  
<!-- /.login-box -->  
  
<!-- jQuery -->  
<script src="adminlte/plugins/jquery/jquery.min.js"></script>  
<!-- Bootstrap 4 -->  
<script src="adminlte/plugins/bootstrap/js/bootstrap.bundle.min.js"></script>  
<!-- AdminLTE App -->  
<script src="adminlte/dist/js/adminlte.min.js"></script>  
<script src="adminlte/plugins/jquery-validation/jquery.validate.min.js"></script>  
<script src="adminlte/plugins/jquery-validation/additional-methods.min.js"></script>  
<script src="adminlte/plugins/jquery-validation/localization/messages_id.min.js"></script>  
  
<script>  
  // maksud nya adl ketika dokumen sudah siap (html telah d render oleh browser) maka  
  jalankan fungsi berikut ini  
  $(document).ready(function(){  
    $('#form-login').validate({  
      rules: {  
        username: {  
          required: true,  
          minlength: 3,  
          maxlength: 20  
        },  
        password: {  
          required: true,
```




```
        minlength: 5,  
        maxlength: 255  
    },  
    },  
    errorElement: 'span',  
    errorPlacement: function (error, element) {  
        error.addClass('invalid-feedback');  
        element.closest('.input-group').append(error);  
    },  
    highlight: function (element, errorClass, validClass) {  
        $(element).addClass('is-invalid');  
    },  
    unhighlight: function (element, errorClass, validClass) {  
        $(element).removeClass('is-invalid');  
    }  
    }  
});  
});  
</script>  
</body>  
</html>
```

Pertanyaan 1

Apa saja perbedaan kode program antara kode program pada
[C:\laragon\www\dasarWeb\JS13\adminlte\pages\examples\login-v2.html](#)
dengan
[C:\laragon\www\dasarWeb\JS13\login.php](#) ?

Jelaskan pendapatmu

- login-v2.html adalah template statis tanpa fungsi backend atau validasi.
- login.php adalah implementasi dinamis dengan backend untuk autentikasi pengguna, validasi input, dan manajemen sesi.

8. Selanjutnya kita buat folder dan file pada
[C:\laragon\www\dasarWeb\JS13\lib\Session.php](#). Kita akan membuat class
Session untuk menangani session PHP. Berikut class Session yang akan kita buat



```
<?php
class Session{
    public function __construct(){
        if (session_status() === PHP_SESSION_NONE) {
            session_start();
        }
    }

    public function set($key, $value){
        $_SESSION[$key] = $value;
    }

    public function get($key){
        return (isset($_SESSION[$key]))? $_SESSION[$key] : null;
    }

    public function exist($key){
        return (isset($_SESSION[$key]))? true : false;
    }

    public function delete($key){
        if(isset($_SESSION[$key])){
            unset($_SESSION[$key]);
        }
    }
}
```



```
}

public function setFlash($key, $value){
    $_SESSION['flash'][$key] = $value;
}

public function getFlash($key){
    $value = (isset($_SESSION['flash'][$key]))? $_SESSION['flash'][$key] : null;

    $this->deleteFlash($key);
    return $value;
}

public function deleteFlash($key){
    if(isset($_SESSION['flash'][$key])){
        unset($_SESSION['flash'][$key]);
    }
}

public function deleteAllFlash(){
    if(isset($_SESSION['flash'])){
        unset($_SESSION['flash']);
    }
}

public function deleteAll(){
    session_destroy();
}

public function commit(){
    session_commit();
}
}
```

9. Kita buat folder dan file di [C:\laragon\www\dasarWeb\JS13\lib\Connection.php](#).
Kemudia kita tulis kode pada file [Connection.php](#) seperti berikut

```
<?php
$username = 'root';
$password = '';
$database = 'dasar_web';

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());
}
?>
```

10. Selanjutnya, kita buat kode program untuk menangani proses login dari file [login.php](#) yang sudah kita buat. Kita buat folder dan file baru pada [C:\laragon\www\dasarWeb\JS13\action\auth.php](#). Folder **action** nanti berfungsi sebagai tempat penyimpanan kode program untuk memproses data. Sekarang kita tulis kode program [auth.php](#) 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'];

    // digunakan untuk query user
    $query = $db->prepare('select * from m_user where username = ?');
    $query->bind_param('s', $username);
    $query->execute();

    // untuk ambil datanya
    $data = $query->get_result()->fetch_assoc();

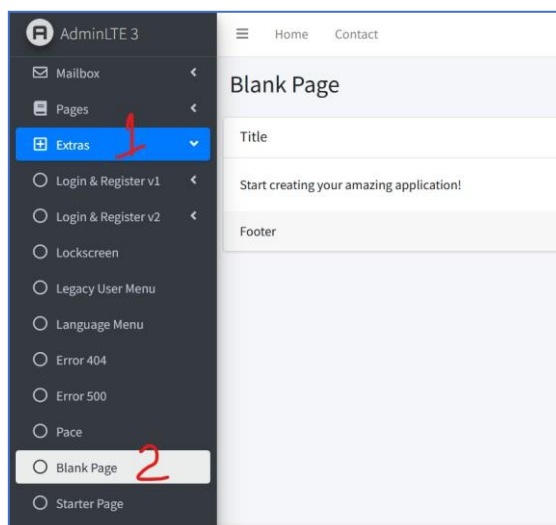
    // jika password sesuai
    if(password_verify($password, $data['password'])){
        $session->set('is_login', true);
        $session->set('username', $data['username']);
        $session->set('name', $data['nama']);
        $session->set('level', $data['level']);
        $session->commit();

        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);
}
```

11. Selanjutnya kita lihat pada template adminlte kita di <localhost/dasarWeb/JS13/adminlte>. Kita pilih menu Extras > Blank Page. **Page inilah yang akan menjadi dasar web template**





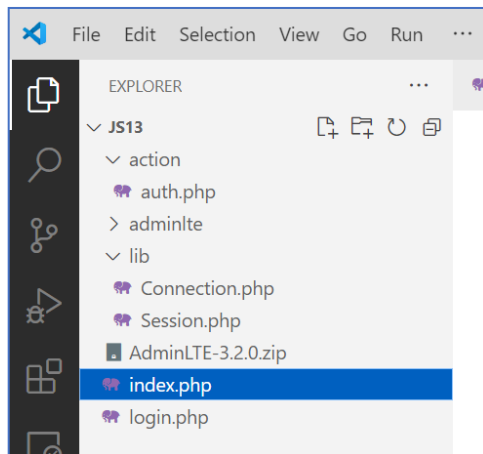
12. Kita *copy* file yang ada di

`C:\laragon\www\dasarWeb\JS13\adminlte\pages\examples\blank.html`

sesuai dengan alamat url

`localhost/dasarWeb/JS13/adminlte/pages/examples/blank.html`.

Kemudian kita *paste* di `C:\laragon\www\dasarWeb\JS13`. Selanjutnya kita *rename* file tersebut jadi `index.php`



13. Selanjutnya kita modifikasi sedikit untuk file `index.php` yang sudah kita buat. Kita tambahkan kode berikut di awal file `index.php`

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

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

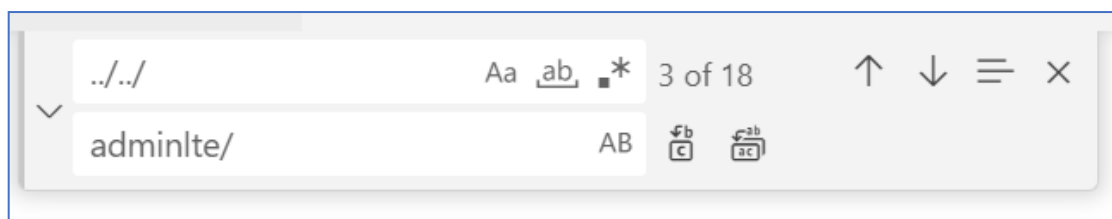
Kemudian kita *find* and *replace* bagian berikut pada file `index.php` di VS Code

Find

`../..`

Dan kita *replace* menjadi

`adminlte/`



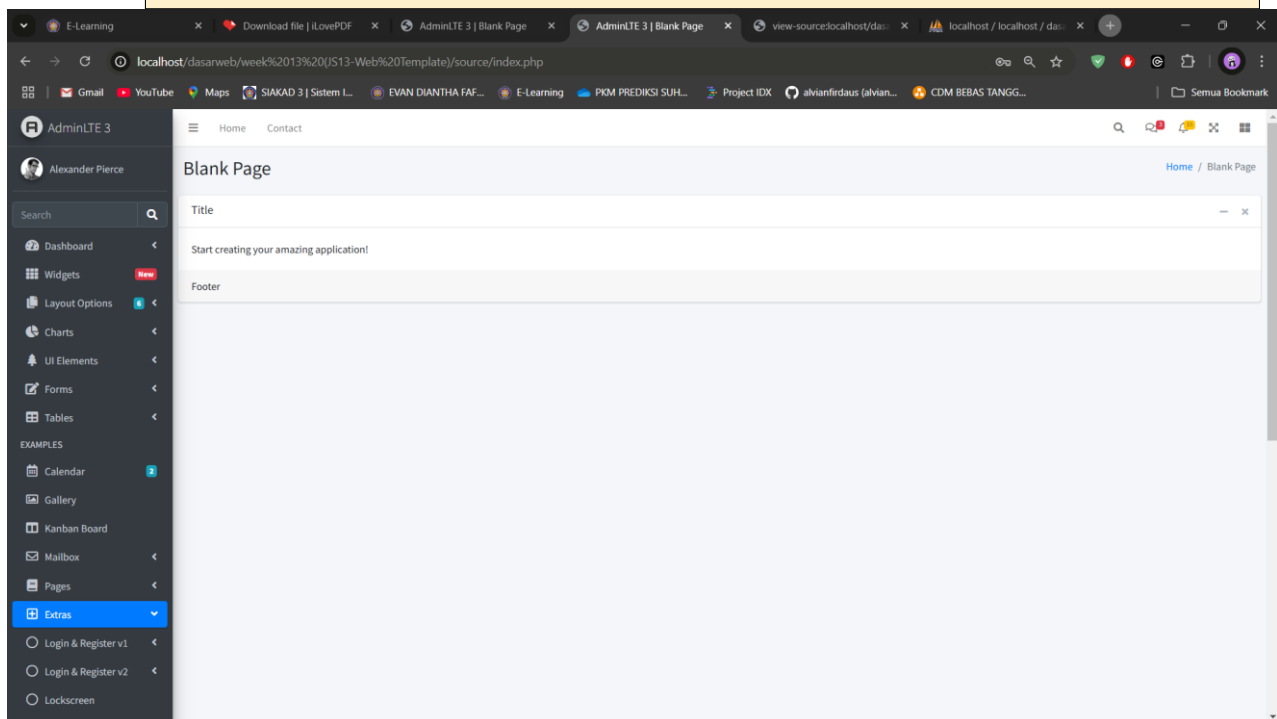


14. Selanjutnya kita coba jalankan pada browser dengan mengetikkan url <localhost/dasarWeb/JS13>.
15. Silahkan lakukan proses login dengan akun
username : admin
password : 12345
16. Amati apa yang terjadi, dan laporkan dalam laporan.

Pertanyaan 2

Apa yang terjadi ketika menjalankan browser pada kode program yang sudah dibuat?

Screenshot hasil ujicoba pada browser dan jelaskan pendapatmu.

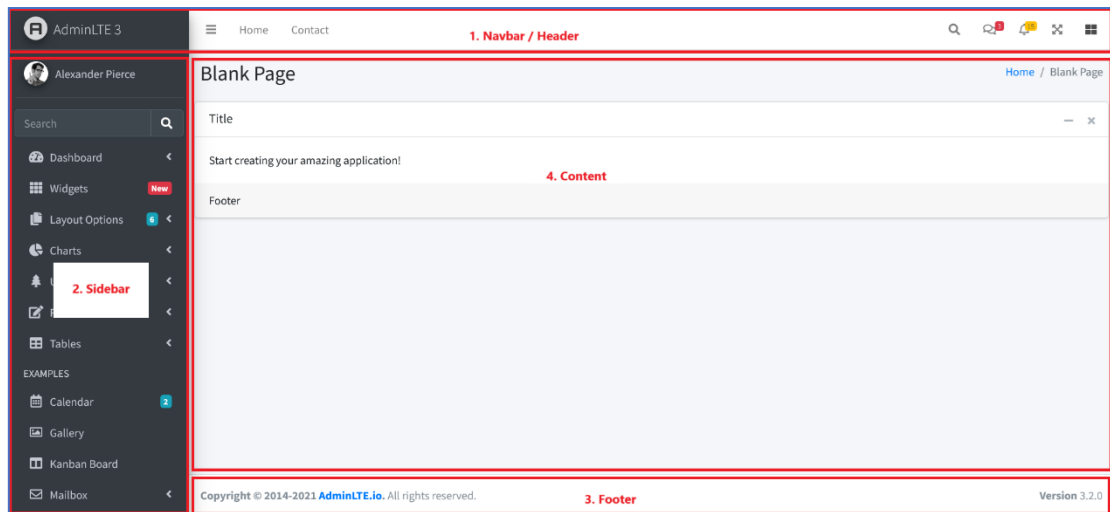


- Ketika kode program dijalankan di browser, pengguna pertama kali diarahkan ke halaman login ('login.php') untuk memasukkan username dan password. Jika login berhasil, sistem memverifikasi kredensial ke database melalui 'auth.php' dan menyimpan data sesi pengguna, kemudian mengarahkan mereka ke dashboard ('index.php'). Jika login gagal, pengguna tetap berada di halaman login dengan pesan error. Dashboard hanya dapat diakses oleh pengguna yang telah login; jika tidak, pengguna secara otomatis diarahkan kembali ke halaman login. Proses logout menghapus semua data sesi pengguna dan mengembalikan mereka ke halaman login. Validasi input dilakukan di sisi klien menggunakan jQuery sebelum data dikirim ke server. Dengan pengaturan yang benar, aplikasi berjalan lancar sesuai alur autentikasi dan kontrol akses.



Praktikum 2 - Layouting AdminLTE:

1. Dari file [index.php](#) yang sudah kita buat, akan kita gunakan untuk proses layouting web
2. Kita lakukan layouting halaman [index.php](#) ini menjadi 5 element seperti pada gambar berikut



3. Kita edit file [index.php](#) dan kita modifikasi untuk memecah file tersebut menjadi 4 elemen, yaitu *header*, *sidebar*, *breadcrumb*, *content*, dan *footer*.
4. Kita buat elemen pertama, yaitu **header**.
5. Kita **cut** kode diantara tanda `<!-- /.navbar -->` pada file [index.php](#), kemudian kita **paste**-kan pada file baru
[C:\laragon\www\dasarWeb\JS13\layout\header.php](#)



6. Selanjutnya kita load file `layouts/header.php` pada `index.php` seperti berikut

```
24 <body class="hold-transition sidebar-mini">
25 <!-- Site wrapper -->
26 <div class="wrapper">
27   <!-- Navbar -->
28   <?php include('layouts/header.php');?>
29   <!-- /.navbar -->
30
31   <!-- Main Sidebar Container -->
32   <aside class="main-sidebar sidebar-dark-primary elevation-4">
```

7. Selanjutnya, kita buat elemen kedua yaitu *sidebar*.
8. Kita *cut* kode diantara tanda `<!-- Sidebar -->` dan `<!-- /.sidebar -->` pada file `index.php`, kemudian kita *paste*-kan pada file baru `C:\laragon\www\dasarWeb\JS13\layout\sidebar.php`
9. Selanjutnya kita load file `layouts/sidebar.php` pada `index.php` seperti berikut

```
31 <!-- Main Sidebar Container -->
32 <aside class="main-sidebar sidebar-dark-primary elevation-4">
33   <!-- Brand Logo -->
34   <a href="adminlte/index3.html" class="brand-link">
35     AdminLTE 3</span>
37   </a>
38
39   <!-- Sidebar -->
40   <?php include('layouts/sidebar.php');?>
41   <!-- /.sidebar -->
42 </aside>
```

10. Selanjutnya, kita buat elemen ketiga yaitu *footer*.
11. Kita pindahkan kode pada tag `<footer></footer>` pada `index.php` ke file baru `C:\laragon\www\dasarWeb\JS13\layout\footer.php`

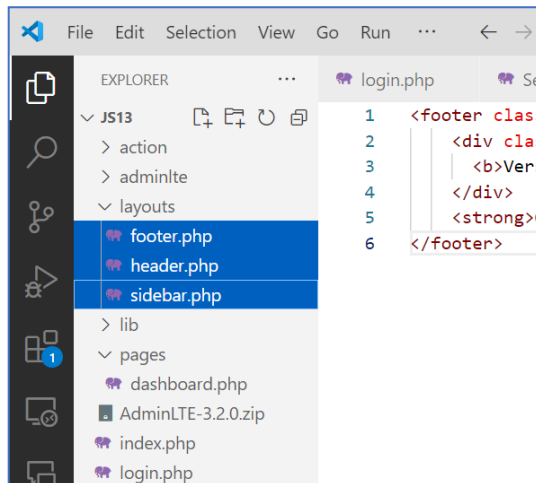
```
<footer class="main-footer">
  <div class="float-right d-none d-sm-block">
    <b>Version</b> 3.2.0
  </div>
  <strong>Copyright &copy; 2014-2021 <a href="https://adminlte.io">AdminLTE.io</a>. All rights reserved.
</footer>
```

12. Selanjutnya kita load file `layouts/footer.php` pada `index.php` seperti berikut

```
80 <!-- /.content-wrapper -->
81
82 <?php include('layouts/footer.php');?>
83
84 <!-- Control Sidebar -->
85 <aside class="control-sidebar control-sidebar-dark">
86   <!-- Control sidebar content goes here -->
87 </aside>
88 <!-- /.control-sidebar -->
89 </div>
```




13. Sehingga dalam folder **layouts**, terdapat 4 file yaitu **header.php**, **sidebar.php**, dan **footer.php**.



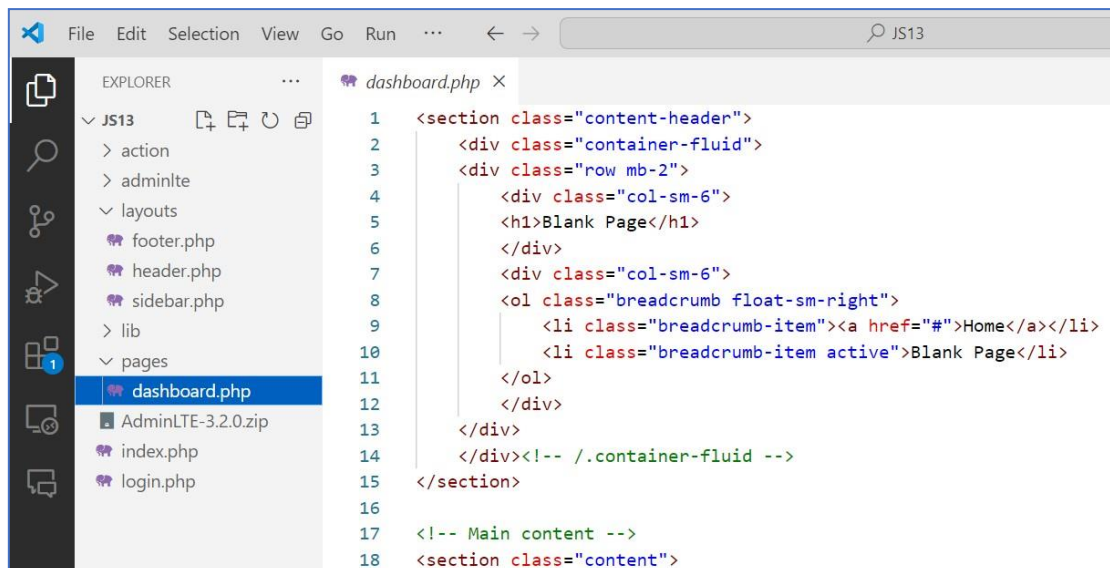
14. Sekarang kita buat layouts untuk content yang nantinya akan sering berganti-ganti sesuai dengan menu yang akses.

15. Kita modif file **index.php**, kita *cut* kode di antara tag

`<!-- Content Header (Page header) -->` dan `<!-- /.content -->`

Lalu kita paste-kan di file dan folder baru pada

C:\laragon\www\dasarWeb\JS13\pages\dashboard.php



16. Kemudian di antara tag

`<!-- Content Header (Page header) -->` dan `<!-- /.content -->`

Pada **index.php**, kita ubah menjadi seperti berikut



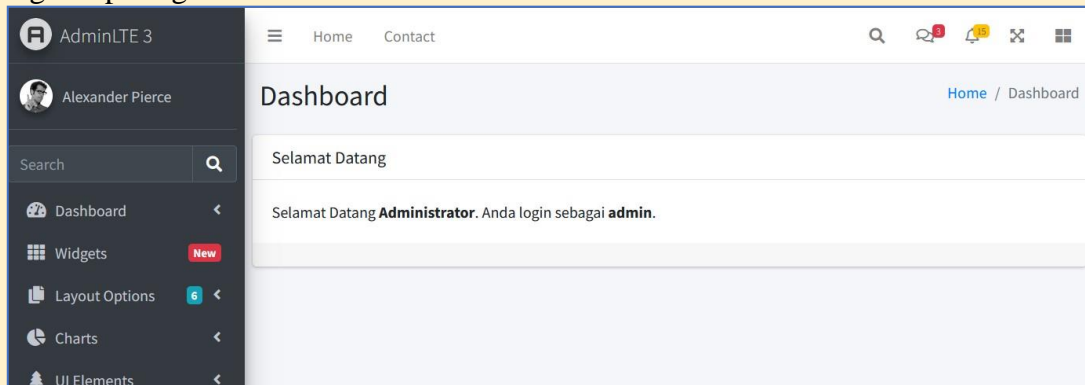
```
44 <!-- Content Wrapper. Contains page content -->
45 <div class="content-wrapper">
46   <!-- Content Header (Page header) -->
47
48   <?php
49     $page = isset($_GET['page']) ? $_GET['page'] : 'dashboard';
50     switch(strtolower($page)){
51       case 'dashboard': include('pages/dashboard.php'); break;
52       case 'buku'      : include('pages/buku.php'); break;
53       case 'kategori' : include('pages/kategori.php'); break;
54       case 'user'     : include('pages/user.php'); break;
55       default         : include('pages/404.php'); break;
56     }
57   ?>
58
59   <!-- /.content -->
60 </div>
61 <!-- /.content-wrapper -->
```

17. Ok, kita sudah berhasil melakukan layouting web template AdminLTE

18. Sekarang silahkan dicoba dijalankan di browser dan amati apa yang terjadi

Pertanyaan 3

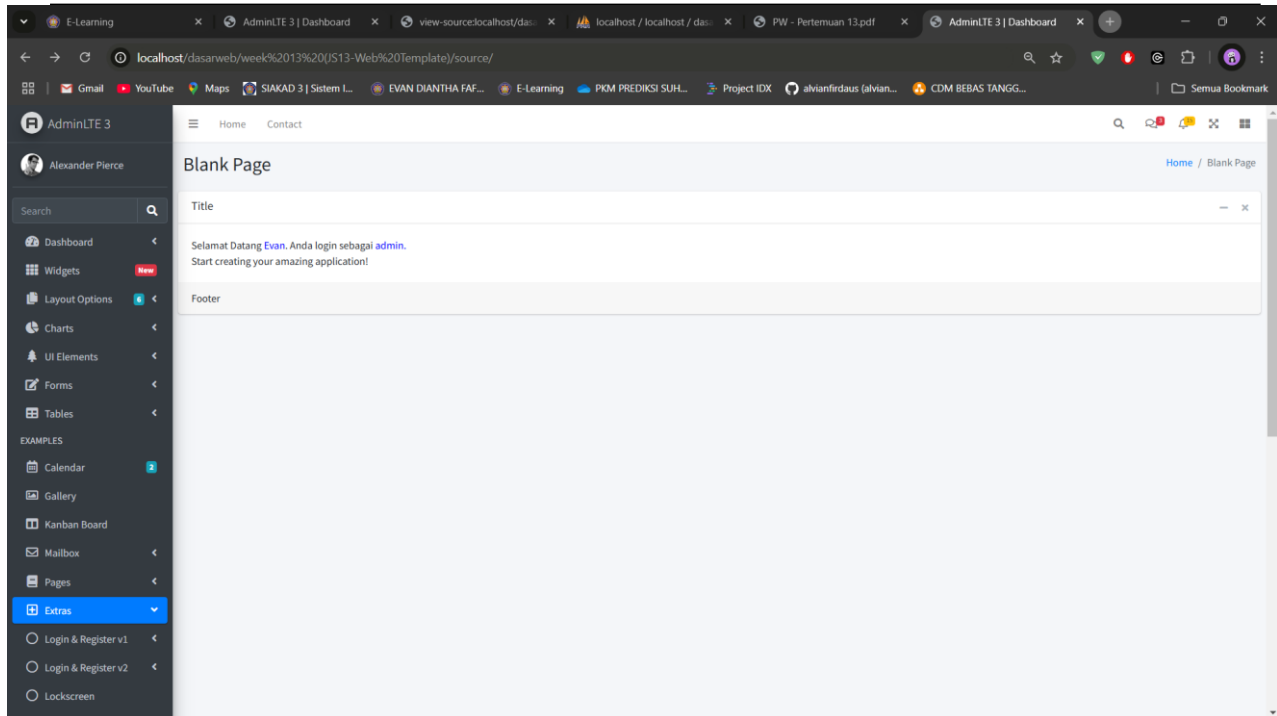
Silahkan ubah tampilan dashboard agar menampilkan nama pengguna yang telah login seperti gambar berikut



Screenshot hasil ujicoba pada browser dan jelaskan pendapatmu.



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>



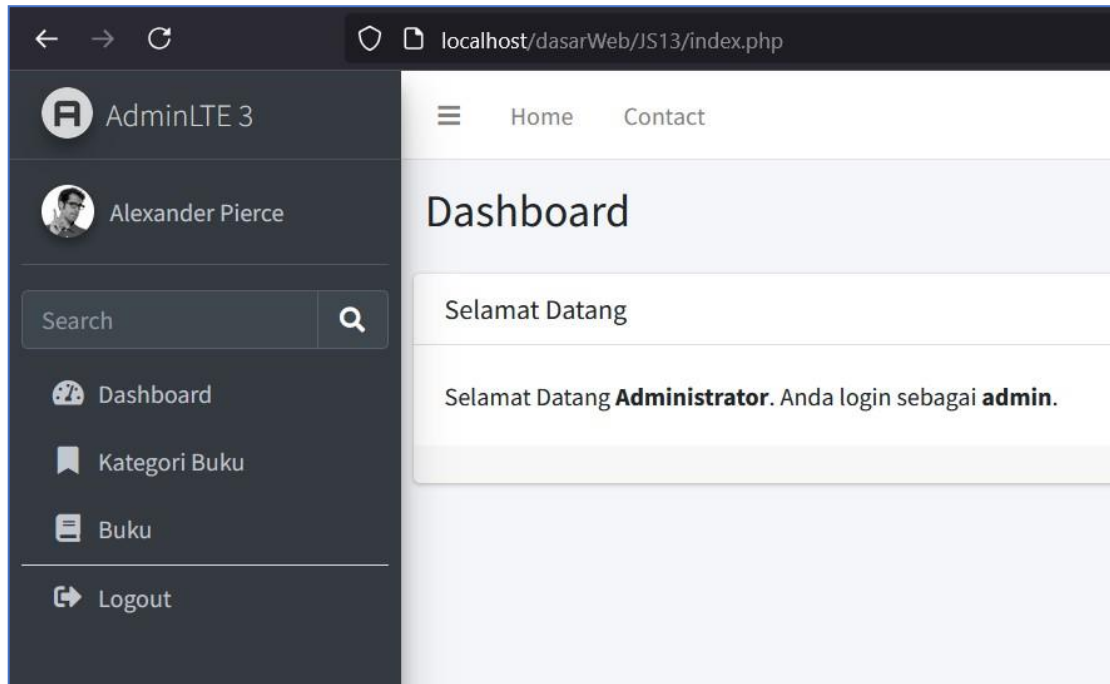
- Di bagian folder pages dan masuk files dashboard lalu ganti bagian title lalu bisa di modify di gantikan oleh font dan bold font



Praktikum 3 - Penerapan Layouting:

Sekarang kita akan mencoba melakukan penerapan terhadap layouting yang sudah kita lakukan.

1. Kita buat tampilan menu sidebar sesuai seperti pada gambar berikut



2. Kita modifikasi file [layouts/sidebar.php](#)



```
<div class="sidebar">
  <!-- Sidebar user (optional) -->
  <div class="user-panel mt-3 pb-3 mb-3 d-flex">
    <div class="image">
      
    </div>
    <div class="info">
      <a href="#" class="d-block">Alexander Pierce</a>
    </div>
  </div>

  <!-- SidebarSearch Form -->
  <div class="form-inline">
    <div class="input-group" data-widget="sidebar-search">
      <input class="form-control form-control-sidebar" type="search"
placeholder="Search" aria-label="Search">
      <div class="input-group-append"><button class="btn btn-sidebar"><i class="fas
fa-search fa-fw"></i></button></div>
    </div>
  </div>

  <!-- Sidebar Menu -->
  <nav class="mt-2">
    <ul class="nav nav-pills nav-sidebar flex-column" data-widget="treeview" role="menu"
data-accordion="false">
      <!-- Add icons to the links using the .nav-icon class
      with font-awesome or any other icon font library -->
      <li class="nav-item">
        <a href="index.php" class="nav-link">
          <i class="nav-icon fas fa-tachometer-alt"></i>
          <p>Dashboard</p>
        </a>
      </li>
      <li class="nav-item">
```



```
<a href="index.php?page=kategori" class="nav-link">
  <i class="nav-icon fas fa-bookmark"></i>
  <p>Kategori Buku</p>
</a>
</li>
<li class="nav-item border-bottom">
  <a href="index.php?page=buku" class="nav-link">
    <i class="nav-icon fas fa-book"></i>
    <p>Buku</p>
  </a>
</li>
<li class="nav-item">
  <a href="action/auth.php?act=logout" class="nav-link">
    <i class="nav-icon fas fa-sign-out-alt"></i>
    <p>Logout</p>
  </a>
</li>
</ul>
</nav>
<!-- /.sidebar-menu -->
</div>
```

3. Kita akan menerapkan jQuery Datatables seperti yang ada pada contoh di template AdminLTE <localhost/dasarWeb/JS13/adminlte/pages/tables/data.html>, maka kita perlu memodifikasi file [index.php](#).
4. Kita tambahkan kode untuk load css eksternal pada [index.php](#). Kita tambahkan kode berikut, sebelum baris load css eksternal [adminlte.min.css](#)

```
<!-- DataTables -->
<link rel="stylesheet" href="adminlte/plugins/datatables-
bs4/css/dataTables.bootstrap4.min.css">
<link rel="stylesheet" href="adminlte/plugins/datatables-
responsive/css/responsive.bootstrap4.min.css">
<link rel="stylesheet" href="adminlte/plugins/datatables-
buttons/css/buttons.bootstrap4.min.css">
```

Selanjutnya, kita pindahkan load library javascript jQuery yang ada di posisi bawah, ke posisi atas setelah load css eksternal [adminlte.min.css](#) (khusus untuk materi pembelajaran kali ini)

```
<!-- Control Sidebar -->
<aside class="control-sidebar control-sidebar-dark">
  <!-- Control sidebar content goes here -->
</aside>
<!-- /.control-sidebar -->
</div>
<!-- ./wrapper -->

<!-- jQuery -->
<script src="adminlte/plugins/jquery/jquery.min.js"></script>
<!-- Bootstrap 4 -->
<script src="adminlte/plugins/bootstrap/js/bootstrap.bundle.min.js"></script>
```

Kita pindah ke atas

Maka tampilannya untuk bagian atas [index.php](#) akan seperti berikut



```
<title>AdminLTE 3 | Blank Page</title>

<!-- Google Font: Source Sans Pro -->
<link rel="stylesheet" href="https://fonts.googleapis.com/css?family=Source+Sans+Pro:300,400,400i,700&displaysubset=latin">
<!-- Font Awesome -->
<link rel="stylesheet" href="adminlte/plugins/fontawesome-free/css/all.min.css">
<!-- DataTables -->
<link rel="stylesheet" href="adminlte/plugins/datatables-bs4/css/dataTables.bootstrap4.min.css">
<link rel="stylesheet" href="adminlte/plugins/datatables-responsive/css/responsive.bootstrap4.min.css">
<link rel="stylesheet" href="adminlte/plugins/datatables-buttons/css/buttons.bootstrap4.min.css">
<!-- Theme style -->
<link rel="stylesheet" href="adminlte/dist/css/adminlte.min.css">
<!-- jQuery -->
<script src="adminlte/plugins/jquery/jquery.min.js"></script>
</head>
<body class="hold-transition sidebar-mini">
```

5. Kemudian, kita tambahkan load library javascript eksternal pada `index.php` (baris-baris terakhir) sebelum baris load javascript `adminlte.min.js`.

```
<!-- jQuery Validation -->
<script src="adminlte/plugins/jquery-validation/jquery.validate.min.js"></script>
<script src="adminlte/plugins/jquery-validation/additional-methods.min.js"></script>
<script src="adminlte/plugins/jquery-validation/localization/messages_id.min.js"></script>
<!-- DataTables & Plugins -->
<script src="adminlte/plugins/datatables/jquery.dataTables.min.js"></script>
<script src="adminlte/plugins/datatables-bs4/js/dataTables.bootstrap4.min.js"></script>
<script src="adminlte/plugins/datatables-responsive/js/dataTables.responsive.min.js"></script>
<script src="adminlte/plugins/datatables-responsive/js/responsive.bootstrap4.min.js"></script>
<script src="adminlte/plugins/datatables-buttons/js/dataTables.buttons.min.js"></script>
<script src="adminlte/plugins/datatables-buttons/js/buttons.bootstrap4.min.js"></script>
<script src="adminlte/plugins/jszip/jszip.min.js"></script>
<script src="adminlte/plugins/pdfmake/pdfmake.min.js"></script>
<script src="adminlte/plugins/pdfmake/vfs_fonts.js"></script>
<script src="adminlte/plugins/datatables-buttons/js/buttons.html5.min.js"></script>
<script src="adminlte/plugins/datatables-buttons/js/buttons.print.min.js"></script>
<script src="adminlte/plugins/datatables-buttons/js/buttons.colVis.min.js"></script>
```

Maka akan tampil kode seperti gambar berikut



```
<!-- ./wrapper -->

<!-- Bootstrap 4 -->
<script src="adminlte/plugins/bootstrap/js/bootstrap.bundle.min.js"></script>
<!-- jQuery Validation -->
<script src="adminlte/plugins/jquery-validation/jquery.validate.min.js"></script>
<script src="adminlte/plugins/jquery-validation/additional-methods.min.js"></script>
<script src="adminlte/plugins/jquery-validation/localization/messages_id.min.js"></script>
<!-- DataTables & Plugins -->
<script src="adminlte/plugins/datatables/jquery.dataTables.min.js"></script>
<script src="adminlte/plugins/datatables-bs4/js/dataTables.bootstrap4.min.js"></script>
<script src="adminlte/plugins/datatables-responsive/js/dataTables.responsive.min.js"></script>
<script src="adminlte/plugins/datatables-responsive/js/responsive.bootstrap4.min.js"></script>
<script src="adminlte/plugins/datatables-buttons/js/dataTables.buttons.min.js"></script>
<script src="adminlte/plugins/datatables-buttons/js/buttons.bootstrap4.min.js"></script>
<script src="adminlte/plugins/jszip/jszip.min.js"></script>
<script src="adminlte/plugins/pdfmake/pdfmake.min.js"></script>
<script src="adminlte/plugins/pdfmake/vfs_fonts.js"></script>
<script src="adminlte/plugins/datatables-buttons/js/buttons.html5.min.js"></script>
<script src="adminlte/plugins/datatables-buttons/js/buttons.print.min.js"></script>
<script src="adminlte/plugins/datatables-buttons/js/buttons.colVis.min.js"></script>
<!-- AdminLTE App -->
<script src="adminlte/dist/js/adminlte.min.js"></script>
</body>
</html>
```

6. Kemudian, kita buat file baru di

`C:\laragon\www\dasarWeb\JS13\pages\kategori.php` dan kita ketikkan kode berikut

```
<section class="content-header">
  <div class="container-fluid">
    <div class="row mb-2">
      <div class="col-sm-6">
        <h1>Kategori Buku</h1>
      </div>
      <div class="col-sm-6">
        <ol class="breadcrumb float-sm-right">
          <li class="breadcrumb-item"><a href="#">Home</a></li>
          <li class="breadcrumb-item active">Kategori</li>
        </ol>
      </div>
    </div>
  </div>
</section>

<!-- Main content -->
<section class="content">
  <div class="card">
    <div class="card-header">
      <h3 class="card-title">Daftar Kategori Buku</h3>
      <div class="card-tools">
        <button type="button" class="btn btn-md btn-primary" onclick="tambahData()">
          Tambah
        </button>
      </div>
    </div>
    <div class="card-body">
      <table class="table table-sm table-bordered table-striped" id="table-data">
        <thead>
          <tr>
            <th>No</th>
          </tr>
        </thead>
      </table>
    </div>
  </div>
```



```
                <th>Kode Kategori</th>
                <th>Nama Kategori</th>
                <th>Aksi</th>
            </tr>
        </thead>
        <tbody>
        </tbody>
    </table>
</div>
</div>
</section>
<div class="modal fade" id="form-data" style="display: none;" aria-hidden="true">
    <form action="action/kategoriAction.php?act=save" method="post" id="form-tambah">
        <!-- Ukuran Modal
            modal-sm : Modal ukuran kecil
            modal-md : Modal ukuran sedang
            modal-lg : Modal ukuran besar
            modal-xl : Modal ukuran sangat besar
        penerapan setelah class modal-dialog seperti di bawah
        -->
        <div class="modal-dialog modal-md">
            <div class="modal-content">
                <div class="modal-header">
                    <h4 class="modal-title">Tambah Kategori</h4>
                </div>
                <div class="modal-body">
                    <div class="form-group">
                        <label>Kode Kategori</label>
                        <input type="text" class="form-control" name="kategori_kode"
id="kategori_kode">
                    </div>
                    <div class="form-group">
                        <label>Nama Kategori</label>
                        <input type="text" class="form-control" name="kategori_nama"
id="kategori_nama">
                    </div>
                </div>
                <div class="modal-footer justify-content-between">
                    <button type="button" class="btn btn-default" data-
dismiss="modal">Batal</button>
                    <button type="submit" class="btn btn-primary">Simpan</button>
                </div>
            </div>
        </form>
    </div>

    <script>
        function tambahData(){
            $('#form-data').modal('show');
            $('#form-tambah').attr('action', 'action/kategoriAction.php?act=save');
            $('#kategori_kode').val('');
            $('#kategori_nama').val('');
        }

        function editData(id){
            $.ajax({
                url: 'action/kategoriAction.php?act=get&id='+id,
                method: 'post',
                success: function(response){
                    var data = JSON.parse(response);
                    $('#form-data').modal('show');
                    $('#form-tambah').attr('action',
'action/kategoriAction.php?act=update&id='+id);
                    $('#kategori_kode').val(data.kategori_kode);
                    $('#kategori_nama').val(data.kategori_nama);
                }
            });
        }
    </script>
</div>
```



```
    });  
  }  
  
  function deleteData(id){  
    if(confirm('Apakah anda yakin?')){  
      $.ajax({  
        url: 'action/kategoriAction.php?act=delete&id='+id,  
        method: 'post',  
        success: function(response){  
          var result = JSON.parse(response);  
          if(result.status){  
            tabelData.ajax.reload();  
          }else{  
            alert(result.message);  
          }  
        }  
      });  
    }  
  }  
  
  var tabelData;  
  $(document).ready(function(){  
    tabelData = $('#table-data').DataTable({  
      ajax: 'action/kategoriAction.php?act=load',  
    });  
  
    $('#form-tambah').validate({  
      rules: {  
        kategori_kode: {  
          required: true,  
          minlength: 3  
        },  
        kategori_nama: {  
          required: true,  
          minlength: 5  
        }  
      },  
      errorElement: 'span',  
      errorPlacement: function(error, element){  
        error.addClass('invalid-feedback');  
        element.closest('.form-group').append(error);  
      },  
      highlight: function(element, errorClass, validClass){  
        $(element).addClass('is-invalid');  
      },  
      unhighlight: function(element, errorClass, validClass){  
        $(element).removeClass('is-invalid');  
      },  
      submitHandler: function(form){  
        $.ajax({  
          url: $(form).attr('action'),  
          method: 'post',  
          data: $(form).serialize(),  
          success: function(response){  
            var result = JSON.parse(response);  
            if(result.status){  
              $('#form-data').modal('hide');  
              tabelData.ajax.reload(); // reload data tabel  
            }else{  
              alert(result.message);  
            }  
          }  
        });  
      }  
    });  
  });  
});
```



```
});  
</script>
```

7. Selanjutnya kita buat file baru di `C:\laragon\www\dasarWeb\JS13\lib\Secure.php`
Kemudian kita ketikkan kode berikut

```
<?php  
  
function antiSqlInjection($data){  
    $data = stripslashes($data);  
    $data = strip_tags($data);  
    $data = htmlentities($data);  
    $data = htmlspecialchars($data);  
    $data = addslashes($data);  
    return $data;  
}
```

8. Selanjutnya kita buat folder baru model, dan kita buat file baru di dalamnya dengan nama `C:\laragon\www\dasarWeb\JS13\model\Model.php`. File `Model.php` ini adalah abstract class untuk class model yang lain.

```
<?php  
abstract class Model{  
    public abstract function insertData($id);  
    public abstract function getData();  
    public abstract function getDataById($id);  
    public abstract function updateData($id, $data);  
    public abstract function deleteData($id);  
}
```

9. Kemudian kita juga buat file `KategoriModel.php` di dalam folder `model`

```
<?php  
include('Model.php');  
  
class KategoriModel extends Model{  
    private $db;  
    private $table = 'm_kategori';  
  
    public function __construct(){  
        include_once('..\lib/Connection.php');  
  
        $this->db = $db;  
        $this->db->set_charset('utf8');  
    }  
  
    public function insertData($data){  
        // 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();  
    }  
}
```



```
public function getData(){
    // query untuk mengambil data dari tabel bank_soal
    return $this->db->query("select * from {$this->table} ");
}

public function getDataById($id){

    // 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();
}

public function updateData($id, $data){
    // 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();
}

public function deleteData($id){
    // 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();
}
}
```

10. Bagian akhir, kita bisa membuat file controller yang memproses CRUD data. Kita buat file `kategoriAction.php` pada folder `action`.

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

```
<?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 = [];
$i = 1;
while($row = $data->fetch_assoc()){
    $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.'  
    });  
}
```

11. Sekarang kita coba jalankan di browser, dan mengakses menu **Kategori Buku**. Kita coba lakukan proses CRUD

<http://localhost/dasarWeb/JS13>

Pertanyaan 4

Silahkan lakukan analisis terhadap file-file yang telah dibuat pada Praktikum 3 ini.

Jabarkan masing-masing file yang dibuat, dan jelaskan maksud dari file-file tersebut.

1. kategori.php:
 - Tampilan utama untuk mengelola kategori buku (tabel data, modal form tambah/edit, dan tombol aksi).
 - Menggunakan AJAX untuk komunikasi server.
2. Secure.php:
 - Melindungi aplikasi dari SQL Injection dengan membersihkan data input pengguna.
3. Model.php:
 - Kelas abstrak sebagai kontrak standar untuk operasi CRUD pada semua model.
4. KategoriModel.php:
 - Implementasi CRUD untuk kategori buku, termasuk fungsi tambah, ambil, ubah, dan hapus data dari database.
5. kategoriAction.php:
 - Controller yang menangani permintaan (load, get, save, update, delete) antara frontend dan database menggunakan KategoriModel.

Pertanyaan 5

Silahkan implementasikan kode yang sudah dibuat untuk membuat CRUD menu Buku.

- Buku.php



```
1 <section class="content-header">
2   <div class="container-fluid">
3     <div class="row mb-2">
4       <div class="col-sm-6">
5         <h1>Daftar Buku</h1>
6       </div>
7       <div class="col-sm-6">
8         <ol class="breadcrumb float-sm-right">
9           <li class="breadcrumb-item"><a href="#">Home</a></li>
10          <li class="breadcrumb-item active">Buku</li>
11        </ol>
12      </div>
13    </div>
14  </div>
15 </section>
16
17 <section class="content">
18   <div class="card">
19     <div class="card-header">
20       <h3 class="card-title">Daftar Buku</h3>
21     </div>
22     <div class="card-tools">
23       <button type="button" class="btn btn-md btn-primary" onclick="tambahData()">
24         Tambah
25       </button>
26     </div>
27     <div class="card-body">
28       <table class="table table-sm table-bordered table-striped" id="table-data">
29         <thead>
30           <tr>
31             <th>No</th>
32             <th>Kode Buku</th>
33             <th>Judul Buku</th>
34             <th>Pengarang</th>
35             <th>Aksi</th>
36           </tr>
37         </thead>
38         <tbody></tbody>
39       </table>
40     </div>
41   </div>
42 </section>
43
44 <div class="modal fade" id="form-data" style="display: none;" aria-hidden="true">
45   <form action="action/bukuAction.php?act=save" method="post" id="form-tambah">
46     <div class="modal-dialog modal-md">
47       <div class="modal-content">
48         <div class="modal-header">
49           <h4 class="modal-title">Tambah Buku</h4>
50         </div>
51         <div class="modal-body">
52           <div class="form-group">
53             <label>Kode Buku</label>
54             <input type="text" class="form-control" name="buku_kode" id="buku_kode" required>
55           </div>
56           <div class="form-group">
57             <label>Judul Buku</label>
58             <input type="text" class="form-control" name="buku_judul" id="buku_judul" required>
59           </div>
60           <div class="form-group">
61             <label>Pengarang</label>
62             <input type="text" class="form-control" name="buku_pengarang" id="buku_pengarang" required>
63           </div>
64         </div>
65         <div class="modal-footer justify-content-between">
66           <button type="button" class="btn btn-default" data-dismiss="modal">Batal</button>
67           <button type="submit" class="btn btn-primary">Simpan</button>
68         </div>
69       </div>
70     </div>
71   </form>
72 </div>
73
74 <script>
75   function tambahData() {
76     $('#form-data').modal('show');
77     $('#form-tambah').attr('action', 'action/bukuAction.php?act=save');
78     $('#buku_kode').val('');
79     $('#buku_judul').val('');
80     $('#buku_pengarang').val('');
81   }
82
83   function editData(id) {
84     $.ajax({
85       url: 'action/bukuAction.php?act=get&id=' + id,
86       method: 'post',
87       success: function(response) {
88         var data = JSON.parse(response);
89         $('#form-data').modal('show');
90         $('#form-tambah').attr('action', 'action/bukuAction.php?act=update&id=' + id);
91         $('#buku_kode').val(data.buku_kode);
92         $('#buku_judul').val(data.buku_judul);
93         $('#buku_pengarang').val(data.buku_pengarang);
94       }
95     });
96   }
97
98   function deleteData(id) {
99     if (confirm("Apakah anda yakin?")) {
100       $.ajax({
101         url: 'action/bukuAction.php?act=delete&id=' + id,
102         method: 'post',
103         success: function(response) {
104           var result = JSON.parse(response);
105           if (result.status) {
106             tabelData.ajax.reload();
107           } else {
108             alert(result.message);
109           }
110         }
111       });
112     }
113   }
114
115   var tabelData;
116   $(document).ready(function() {
117     tabelData = $('#table-data').DataTable({
118       ajax: 'action/bukuAction.php?act=load',
119     });
120   });
121 </script>
```



- BukuModel.php

```
1  <?php
2  include('Model.php');
3
4  class BukuModel extends Model
5  {
6      private $db;
7      private $table = 'm_buku';
8
9      public function __construct()
10     {
11         include_once('../lib/Connection.php');
12         $this->db = $db;
13         $this->db->set_charset('utf8');
14     }
15
16     public function insertData($data)
17     {
18         $query = $this->db->prepare("INSERT INTO {$this->table} (buku_kode, buku_judul, buku_pengarang) VALUES (?, ?, ?)");
19         $query->bind_param('sss', $data['buku_kode'], $data['buku_judul'], $data['buku_pengarang']);
20         $query->execute();
21     }
22
23     public function getData()
24     {
25         return $this->db->query("SELECT * FROM {$this->table}");
26     }
27
28     public function getDataById($id)
29     {
30         $query = $this->db->prepare("SELECT * FROM {$this->table} WHERE buku_id = ?");
31         $query->bind_param('i', $id);
32         $query->execute();
33         return $query->get_result()->fetch_assoc();
34     }
35
36     public function updateData($id, $data)
37     {
38         $query = $this->db->prepare("UPDATE {$this->table} SET buku_kode = ?, buku_judul = ?, buku_pengarang = ? WHERE buku_id = ?");
39         $query->bind_param('sssi', $data['buku_kode'], $data['buku_judul'], $data['buku_pengarang'], $id);
40         $query->execute();
41     }
42
43     public function deleteData($id)
44     {
45         $query = $this->db->prepare("DELETE FROM {$this->table} WHERE buku_id = ?");
46         $query->bind_param('i', $id);
47         $query->execute();
48     }
49 }
50
```

- bukuAction.php



```
1 <?php
2 include('../lib/Session.php');
3 $session = new Session();
4
5 if ($session->get('is_login') != true) {
6     header('Location: ../login.php');
7     exit;
8 }
9
10 include_once('../model/BukuModel.php');
11 include_once('../lib/Secure.php');
12
13 $sact = isset($_GET['act']) ? strtolower($_GET['act']) : '';
14
15 if ($sact == 'load') {
16     $buku = new BukuModel();
17     $data = $buku->getData();
18     $result = [];
19     $i = 1;
20
21     while ($row = $data->fetch_assoc()) {
22         $result['data'][] = [
23             $i,
24             $row['buku_kode'],
25             $row['buku_judul'],
26             $row['buku_pengarang'],
27             '<button class="btn btn-sm btn-warning" onclick="editData(' . $row['buku_id'] . ') "><i class="fa fa-edit"></i></button>
28             <button class="btn btn-sm btn-danger" onclick="deleteData(' . $row['buku_id'] . ') "><i class="fa fa-trash"></i></button>'
29         ];
30         $i++;
31     }
32
33     echo json_encode($result);
34 }
35
36 if ($sact == 'get') {
37     $id = (isset($_GET['id']) && ctype_digit($_GET['id'])) ? intval($_GET['id']) : 0;
38     $buku = new BukuModel();
39     $data = $buku->getDataById($id);
40     echo json_encode($data);
41 }
42
43 if ($sact == 'save') {
44     $data = [
45         'buku_kode' => antiSqlInjection($_POST['buku_kode']),
46         'buku_judul' => antiSqlInjection($_POST['buku_judul']),
47         'buku_pengarang' => antiSqlInjection($_POST['buku_pengarang']),
48     ];
49
50     $buku = new BukuModel();
51     $buku->insertData($data);
52
53     echo json_encode([
54         'status' => true,
55         'message' => 'Data buku berhasil disimpan.'
56     ]);
57 }
58
59 if ($sact == 'update') {
60     $id = (isset($_GET['id']) && ctype_digit($_GET['id'])) ? intval($_GET['id']) : 0;
61
62     $data = [
63         'buku_kode' => antiSqlInjection($_POST['buku_kode']),
64         'buku_judul' => antiSqlInjection($_POST['buku_judul']),
65         'buku_pengarang' => antiSqlInjection($_POST['buku_pengarang']),
66     ];
67
68     $buku = new BukuModel();
69     $buku->updateData($id, $data);
70
71     echo json_encode([
72         'status' => true,
73         'message' => 'Data buku berhasil diperbarui.'
74     ]);
75 }
76
77 if ($sact == 'delete') {
78     $id = (isset($_GET['id']) && ctype_digit($_GET['id'])) ? intval($_GET['id']) : 0;
79
80     $buku = new BukuModel();
81     $buku->deleteData($id);
82
83     echo json_encode([
84         'status' => true,
85         'message' => 'Data buku berhasil dihapus.'
86     ]);
87 }
88 }
```



- update index.php

```
1  <?php
2      $page = isset($_GET['page']) ? strtolower($_GET['page']) : 'dashboard';
3      switch ($page) {
4          case 'dashboard':
5              include('pages/dashboard.php');
6              break;
7          case 'buku':
8              include('model/Buku.php');
9              break;
10         case 'kategori':
11             include('pages/kategori.php');
12             break;
13         case 'user':
14             include('pages/user.php');
15             break;
16         default:
17             include('pages/404.php');
18             break;
19     }
20     ?>
```

*** Sekian, dan selamat belajar ***