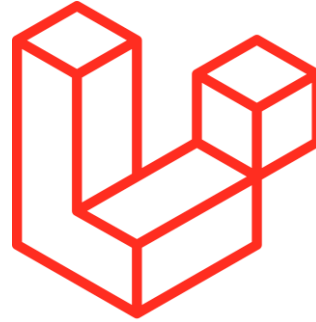


# Pemrograman Backend



## Pesantren PeTIK II YBM PLN

Jl. KH. Bisri Syansuri RT/01 RW/05, Plosogeneng,  
Kec. Jombang, Kabupaten Jombang, Jawa Timur

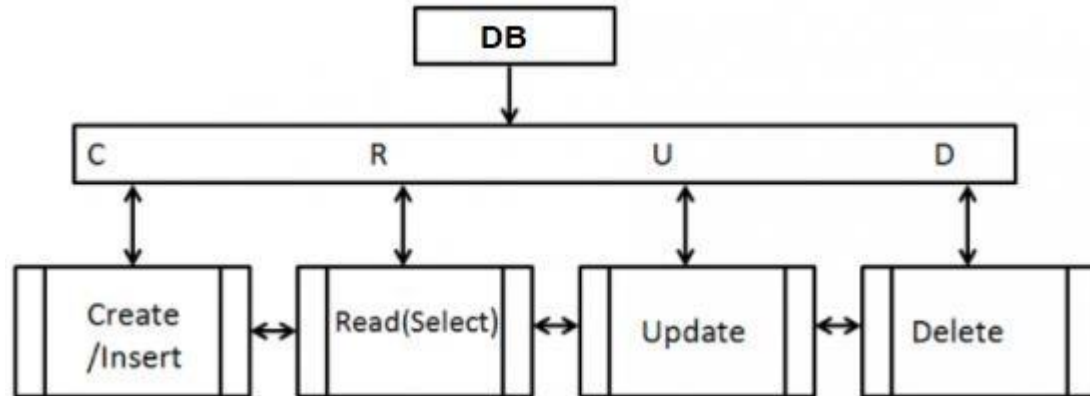


## 9. Operasi CRUD



# Definisi CRUD

- CRUD adalah singkatan dari create, read, update, and delete. Keempat poin ini merupakan fungsi-fungsi utama yang diimplementasikan dalam aplikasi yang terkoneksi ke sebuah database.
- Akronim CRUD mengidentifikasi semua fungsi utama yang melekat pada database relasional dan aplikasi yang digunakan untuk mengelolanya, termasuk Oracle Database, Microsoft SQL Server, MySQL, PostgreSQL dan database lainnya.





# Langkah-Langkah Operasi CRUD

Membuat Model Eloquent

Membuat Controller

Menambahkan Route

Membuat View



# Mendefinisikan Model Eloquent

Untuk studi kasus ini, Kita menggunakan metode Database First. Setelah membuat ER multi tabel, selanjutnya kita definisikan model eloquent, buka terminal dan masuk ke root path aplikasi Laravel kita, jalankan perintah di bawah ini:

```
E:\xampp\htdocs\example-app>php artisan make:model Pengarang  
Model created successfully.
```

```
E:\xampp\htdocs\example-app>php artisan make:model Penerbit  
Model created successfully.
```

```
E:\xampp\htdocs\example-app>php artisan make:model Kategori  
Model created successfully.
```

```
E:\xampp\htdocs\example-app>php artisan make:model Buku  
Model created successfully.
```



# Konfigurasi Model Eloquent Pengarang

```
<?php
namespace App\Models;
use Illuminate\Database\Eloquent\Factories\HasFactory;
use Illuminate\Database\Eloquent\Model;

class Pengarang extends Model
{
    use HasFactory;
    //tambahkan kode untuk mapping ke tabel pengarang
    protected $table = 'pengarang';
}
```



# Konfigurasi Model Eloquent Penerbit

```
<?php
namespace App\Models;
use Illuminate\Database\Eloquent\Factories\HasFactory;
use Illuminate\Database\Eloquent\Model;

class Penerbit extends Model
{
    use HasFactory;
    //tambahkan kode untuk mapping ke tabel penerbit
    protected $table = 'penerbit';
}
```





# Konfigurasi Model Eloquent Kategori

```
<?php
namespace App\Models;
use Illuminate\Database\Eloquent\Factories\HasFactory;
use Illuminate\Database\Eloquent\Model;

class Kategori extends Model
{
    use HasFactory;
    //tambahkan kode untuk mapping ke tabel kategori
    protected $table = 'kategori';
}
```







# Konfigurasi Model Eloquent Buku

```
<?php
namespace App\Models;
use Illuminate\Database\Eloquent\Factories\HasFactory;
use Illuminate\Database\Eloquent\Model;

class Buku extends Model
{
    use HasFactory;
    //tambahkan kode untuk mapping ke tabel buku
    protected $table = 'buku';
}
```



# Membuat Controller

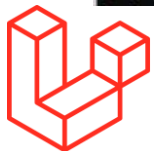
Setelah membuat model-model eloquent, langkah selanjutnya adalah membuat controller di masing-masing tabel, buka terminal dan masuk ke root path aplikasi Laravel kita, jalankan perintah di bawah ini:

```
E:\xampp\htdocs\example-app>php artisan make:controller PengarangController --resource
Controller created successfully.

E:\xampp\htdocs\example-app>php artisan make:controller PenerbitController --resource
Controller created successfully.

E:\xampp\htdocs\example-app>php artisan make:controller KategoriController --resource
Controller created successfully.

E:\xampp\htdocs\example-app>php artisan make:controller BukuController --resource
```



# Menambahkan Routing Baru

Selanjutnya tambahkanlah routing baru untuk setiap controller yang sudah dibuat, buka file **routes/web.php** lalu tambahkan kode di bawah ini:

```
use App\Http\Controllers\PengarangController;  
use App\Http\Controllers\PenerbitController;  
use App\Http\Controllers\KategoriController;  
use App\Http\Controllers\BukuController;
```

```
Route::resource('pengarang', PengarangController::class);  
Route::resource('penerbit', PenerbitController::class);  
Route::resource('kategori', KategoriController::class);  
Route::resource('buku', BukuController::class);
```



# Fungsi Index untuk Menampilkan Data Relasi Buku

<?php

//file BukuController.php yang berada di folder app/Http/Controllers

```
namespace App\Http\Controllers;
```

```
use Illuminate\Http\Request;
```

```
use DB; //tambahan
```

```
class BukuController extends Controller
```

```
{
```

```
    public function index()
```

```
    {
```

```
        $ar_buku = DB::table('buku') //join tabel dengan Query Builder Laravel
```

```
            ->join('pengarang', 'pengarang.id', '=', 'buku.idpengarang')
```

```
            ->join('penerbit', 'penerbit.id', '=', 'buku.idpenerbit')
```

```
            ->join('kategori', 'kategori.id', '=', 'buku.idkategori')
```

```
            ->select('buku.*', 'pengarang.nama', 'penerbit.nama AS pen',  
                'kategori.nama AS kat')->get();
```

```
        return view('buku.index', compact('ar_buku'));
```

```
    }
```



# Fungsi Create untuk Menampilkan Form Buku

```
<?php
//file BukuController.php yang berada di folder app/Http/Controllers
namespace App\Http\Controllers;
use Illuminate\Http\Request;
use DB; //tambahan
class BukuController extends Controller
{
    //kode sebelumnya
    public function create{
        //mengarahkan ke hal form input buku
        return view('buku.form');
    }
    //kode sesudahnya
}
```



# Fungsi Store untuk Proses Input Buku

```
app > Http > Controllers >  BukuController.php >  BukuController >  store  
42     public function store(Request $request)|  
43     {  
44         //proses input data, tangkap request dari form buku  
45         DB::table('buku')->insert(  
46             [  
47                 'isbn'=>$request->isbn,  
48                 'judul'=>$request->judul,  
49                 'tahun_cetak'=>$request->tahun_cetak,  
50                 'stok'=>$request->stok,  
51                 'idpengarang'=>$request->idpengarang,  
52                 'idpenerbit'=>$request->idpenerbit,  
53                 'idkategori'=>$request->idkategori,  
54                 'cover'=>$request->cover,  
55             ]  
56         );  
57         //landing page  
58         return redirect('/buku');  
59     }
```

# Fungsi Show untuk Menampilkan Detail Data Buku

app > Http > Controllers >  BukuController.php >  BukuController >  show

```
67 public function show($id)|
68 {
69     //menampilkan detail buku
70     $ar_buku = DB::table('buku')
71         ->join('pengarang', 'pengarang.id', '=', 'buku.idpengarang')
72         ->join('penerbit', 'penerbit.id', '=', 'buku.idpenerbit')
73         ->join('kategori', 'kategori.id', '=', 'buku.idkategori')
74         ->select('buku.*', 'pengarang.nama', 'penerbit.nama AS pen',
75             'kategori.nama AS kat')
76         ->where('buku.id', '=', $id)->get();
77     return view('buku.show', compact('ar_buku'));
78 }
```



# Fungsi Edit untuk Menampilkan Form Edit Buku



```
<?php
//file BukuController.php yang berada di folder app/Http/Controllers
namespace App\Http\Controllers;
use Illuminate\Http\Request;
use DB; //tambahan
class BukuController extends Controller
{
    //kode sebelumnya
    public function edit($id)
    {
        //mengarahkan ke halaman form edit buku
        $data = DB::table('buku')
                    ->where('id','=',$id)->get();
        return view('buku.form_edit',compact('data'));
    }
    //kode sesudahnya
}
```







# Fungsi Update untuk Ubah Data Buku

```
app > Http > Controllers >  BukuController.php >  BukuController >  update

102     public function update(Request $request, $id)|
103     {
104         //proses edit data lama, tangkap request dari form edit buku
105         DB::table('buku')->where('id','=', $id)->update(
106             [
107                 'isbn'=>$request->isbn,
108                 'judul'=>$request->judul,
109                 'tahun_cetak'=>$request->tahun_cetak,
110                 'stok'=>$request->stok,
111                 'idpengarang'=>$request->idpengarang,
112                 'idpenerbit'=>$request->idpenerbit,
113                 'idkategori'=>$request->idkategori,
114             ]
115         );
116         //landing page
117         return redirect('/buku'.'/'.$id);
118     }
```

# Fungsi Destroy untuk Hapus Data Buku

```
<?php
//file BukuController.php yang berada di folder app/Http/Controllers
namespace App\Http\Controllers;
use Illuminate\Http\Request;
use DB; //tambahan
class BukuController extends Controller
{
    //kode sebelumnya
    public function destroy($id)
    {
        //menghapus data
        DB::table('buku')->where('id',$id)->delete();
        return redirect('/buku');
    }
    //kode sesudahnya
}
```



# Membuat View Data Buku

Buatlah file `index.blade.php` untuk menampilkan data buku, di folder `resources/views/buku`, lalu tambahkan kode di bawah ini:

```
@extends('adminlte::page')
@section('title', 'Data Buku')
@section('content_header')
    <h1>Data Buku</h1>
@stop
@section('content')
    {{-- Ini Konten Data Buku --}}
@stop
@section('css')
    <link rel="stylesheet" href="/css/admin_custom.css">
@stop
@section('js')
    <script> console.log('Hi!'); </script>
@stop
```



# Isi Konten Data Buku(1)

```
@section('content')
@php
$ar_judul = ['No', 'ISBN', 'Judul', 'Stok', 'Pengarang', 'Penerbit',
            'Kategori', 'Action'];

$no = 1;
@endphp
<h3>Daftar Buku</h3>
<a class="btn btn-primary" href="{{ route('buku.create') }}"
    role="button">Tambah</a><br/><br/>
<table class="table table-striped">
    <thead>
        <tr>
            @foreach($ar_judul as $jdl)
                <th>{{ $jdl }}</th>
            @endforeach
        </tr>
    </thead>
    {{-- <tbody> lanjut ke slide berikutnya --}}
```

# Isi Konten Data Buku(2)

```
{{-- Lanjutan Isi Konten Data Buku dari slide sebelumnya --}}
```

```
<tbody>
```

```
  @foreach($ar_buku as $b)
```

```
    <tr>
```

```
      <td>{{ $no++ }}</td>
```

```
      <td>{{ $b->isbn }}</td>
```

```
      <td>{{ $b->judul }}</td>
```

```
      <td>{{ $b->stok }}</td>
```

```
      <td>{{ $b->nama }}</td>
```

```
      <td>{{ $b->pen }}</td>
```

```
      <td>{{ $b->kat }}</td>
```

```
{{-- <td> untuk tombol detail, ubah, hapus lanjut ke slide berikutnya --}}
```



# Isi Konten Data Buku(3)

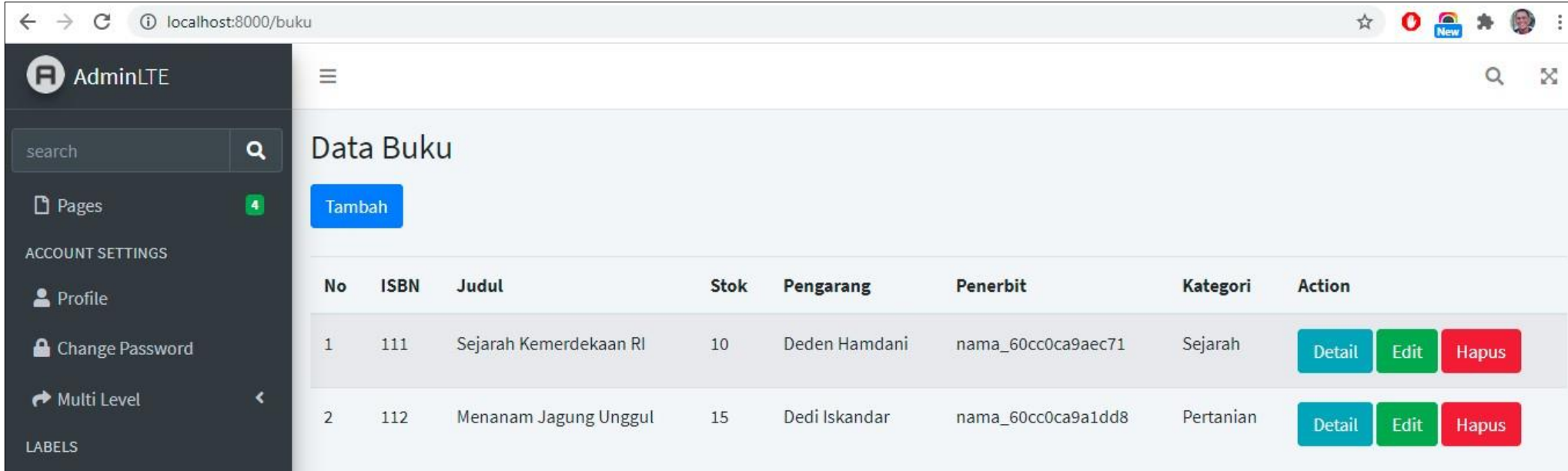
{{-- Membuat tombol Detail, Edit dan Hapus --}}

resources > views > buku > index.blade.php > table.table.table-striped > tbody > tr > td

```
30      <td>
31          <form method="POST" action="{{ route('buku.destroy',$b->id) }}">
32              @csrf {{--security untuk menghindari dari serangan pada saat input form--}}
33              @method('delete') {{-- method delete digunakan untuk menghapus data --}}
34              <a class="btn btn-info"
35                  href="{{ route('buku.show',$b->id) }}">Detail</a>
36              <a class="btn btn-success"
37                  href="{{ route('buku.edit',$b->id) }}">Edit</a>
38              <button class="btn btn-danger"
39                  onclick="return confirm('Anda Yakin Data diHapus?')">Hapus</button>
40          </form>
41      </td>
42  </tr>
43  @endforeach
44 </tbody>
45 </table>
46 @stop
```

# Tampilan Data Relasi Buku

Setelah membuat halaman view relasi buku, selanjutnya buka halaman data relasi buku, di web browser, pada url ketikkan: `http://localhost:8000/buku`. Maka tampililah halaman relasi buku, seperti gambar di bawah ini:



The screenshot shows a web browser window with the URL `localhost:8000/buku`. The page is titled "Data Buku" and features a sidebar with navigation links. The main content area displays a table of book data with columns: No, ISBN, Judul, Stok, Pengarang, Penerbit, Kategori, and Action. There are two rows of data, each with buttons for Detail, Edit, and Hapus.

| No | ISBN | Judul                  | Stok | Pengarang     | Penerbit           | Kategori  | Action  |
|----|------|------------------------|------|---------------|--------------------|-----------|---|
| 1  | 111  | Sejarah Kemerdekaan RI | 10   | Deden Hamdani | nama_60cc0ca9aec71 | Sejarah   | <a href="#">Detail</a> <a href="#">Edit</a> <a href="#">Hapus</a> |
| 2  | 112  | Menanam Jagung Unggul  | 15   | Dedi Iskandar | nama_60cc0ca9a1dd8 | Pertanian | <a href="#">Detail</a> <a href="#">Edit</a> <a href="#">Hapus</a> |

# Membuat View Form Input Buku

Buatlah file `form.blade.php` untuk menampilkan form input data buku, di folder `resources/views/buku`, lalu tambahkan kode di bawah ini:

```
@extends('adminlte::page')
@section('title', 'Form Buku')
@section('content_header')
    <h1>Data Buku</h1>
@stop
@section('content')
    {{-- Ini Konten Form Input Buku --}}
@stop
@section('css')
    <link rel="stylesheet" href="/css/admin_custom.css">
@stop
@section('js')
    <script> console.log('Hi!'); </script>
@stop
```





# Isi Konten Form Input Buku

```
@section('content')
{{-- Panggil master data pengarang, penerbit dan kategori untuk
    ditampilkan di element form --}}

@php
$rs1 = App\Models\Pengarang::all();
$rs2 = App\Models\Penerbit::all();
$rs3 = App\Models\Kategori::all();
@endphp
<form method="POST" action="{ route('buku.store') }">
@csrf {{-- security untuk menghindari dari serangan pada saat input form --}}

<div class="form-group">
    <label>ISBN</label>
    <input type="text" name="isbn" class="form-control"/>
</div>
{{-- Lanjutan ke element-element form berikutnya --}}
```



# Element Form Select Option(1)

```
{{-- Panggil master data pengarang untuk ditampilkan di element form select --}}
```

```
<div class="form-group">
  <label>Pengarang</label>
  <select class="form-control" name="idpengarang">
    <option value="">-- Pilih Pengarang --</option>
    @foreach($rs1 as $p)
      <option value="{{ $p->id }}">{{ $p->nama }}</option>
    @endforeach
  </select>
</div>
```



## Element Form Select Option(2)

```
{{-- Panggil master data penerbit untuk ditampilkan di element form select --}}
```

```
<div class="form-group">
  <label>Penerbit</label>
  <select class="form-control" name="idpenerbit">
    <option value="">-- Pilih Penerbit --</option>
    @foreach($rs2 as $pen)
      <option value="{{ $pen->id }}">{{ $pen->nama }}</option>
    @endforeach
  </select>
</div>
```



# Element Form Radio Button

```
{{-- Panggil master data kategori untuk ditampilkan  
di element radio button--}}
```

```
<div class="form-group">  
  <label>Kategori</label><br/>  
  @foreach($rs3 as $k)  
    <input type="radio" name="idkategori"  
      value="{{ $k->id }}" />{{ $k->nama }} &nbsp; &nbsp;  
  @endforeach  
</div>
```





# Membuat Tombol Simpan dan Batal

```
<button type="submit" class="btn btn-primary"  
name="proses">Simpan</button>
```

```
<button type="reset" class="btn btn-warning"  
name="unproses">Batal</button>
```



# Tampilan Form Input Buku

AdminLTE

search

Pages

ACCOUNT SETTINGS

Profile

Change Password

Multi Level

LABELS

Important

Warning

Information

Form Buku

ISBN

Judul

Tahun Cetak

Stok

Pengarang

Penerbit

Kategori

Cover

Simpan

Batal

# Membuat View Form Edit Buku

Buatlah file `form_edit.blade.php` untuk menampilkan form edit data buku, di folder `resources/views/buku`, lalu tambahkan kode di bawah ini:

```
@extends('adminlte::page')
@section('title', 'Form Buku')
@section('content_header')
    <h1>Data Buku</h1>
@stop
@section('content')
    {{-- Ini Konten Form Edit Buku --}}
@stop
@section('css')
    <link rel="stylesheet" href="/css/admin_custom.css">
@stop
@section('js')
    <script> console.log('Hi!'); </script>
@stop
```



# Isi Konten Form Edit Buku

```
@section('content')
```

```
{{-- Panggil master data pengarang, penerbit dan kategori untuk  
    ditampilkan di element form edit buku --}}
```

```
@php
```

```
$rs1 = App\Models\Pengarang::all();
```

```
$rs2 = App\Models\Penerbit::all();
```

```
$rs3 = App\Models\Kategori::all();
```

```
@endphp
```

```
@foreach($data as $rs)
```

```
<form method="POST" action="{{ route('buku.update',$rs->id) }}">
```

```
@csrf {{-- security untuk menghindari dari serangan pada saat input form --}}
```

```
@method('put') {{-- method put digunakan untuk meletakkan data yang akan diubah  
    disetiap element form edit buku --}}
```

```
<div class="form-group">
```

```
    <label>ISBN</label>
```

```
    <input type="text" name="isbn" value="{{ $rs->isbn }}" class="form-control"/>
```

```
</div>
```

```
{{-- Lanjutan ke element-element form berikutnya --}}
```





# Proses Edit Data pada Element Form Select(1)

```
<div class="form-group">
  <label>Pengarang</label>
  <select class="form-control" name="idpengarang">
    <option value="">-- Pilih Pengarang --</option>
    @foreach($rs1 as $p)
      {{-- edit pengarang --}}
      @php
        $sel1 = ($p->id == $rs->idpengarang) ? 'selected' : '';
      @endphp
      <option value="{{ $p->id }}" {{ $sel1 }}>{{ $p->nama }}</option>
    @endforeach
  </select>
</div>
```



## Proses Edit Data pada Element Form Select(2)

```
<div class="form-group">
  <label>Penerbit</label>
  <select class="form-control" name="idpenerbit">
    <option value="">-- Pilih Penerbit --</option>
    @foreach($rs2 as $pen)
      {{-- edit penerbit --}}
      @php
        $sel2 = ($pen->id == $rs->idpenerbit) ? 'selected' : '';
      @endphp
      <option value="{{ $pen->id }}" {{ $sel2 }}>{{ $pen-
>nama }}</option>
    @endforeach
  </select>
</div>
```



# Proses Edit Data pada Element Form Radio Button

```
<div class="form-group">
  <label>Kategori</label><br/>
  @foreach($rs3 as $k)
    {{-- edit kategori --}}
    @php
      $cek = ($k->id == $rs->idkategori) ? 'checked' : '';
    @endphp
    <input type="radio" name="idkategori"
      value="{{ $k->id }}" {{ $cek }}/>{{ $k->nama }} &nbsp;
  @endforeach
</div>
```





# Membuat Tombol Ubah dan Batal

```
<button type="submit" class="btn btn-primary"  
name="proses">Ubah</button>
```

```
<button type="reset" class="btn btn-warning"  
name="unproses">Batal</button>
```



# Tampilan Form Edit Buku

← → ↻

localhost:8000/buku/2/edit

🔍 ⚙️ 🌐 📱 🧑

AdminLTE

search 🔍

📄 Pages 4

ACCOUNT SETTINGS

👤 Profile

🔒 Change Password

🔄 Multi Level

LABELS

🔴 Important

🟡 Warning

🟢 Information

☰

Form Edit Buku

ISBN

112

Judul

Menanam Jagung Unggul

Tahun Cetak

2021

Stok

15

Pengarang

Dedi Iskandar

Penerbit

nama\_60cc0ca9a1dd8

Kategori

☒ Pertanian ☐ Sejarah ☐ Sosial & Politik

Ubah Batal



# Membuat View Detail Data Buku

Buatlah file `show.blade.php` untuk menampilkan detail data buku, di folder `resources/views/buku`, lalu tambahkan kode di bawah ini:

```
@extends('adminlte::page')
@section('title', 'Form Buku')
@section('content_header')
    <h1>Data Buku</h1>
@stop
@section('content')
    {{-- Ini Konten Detail Data Buku --}}
@stop
@section('css')
    <link rel="stylesheet" href="/css/admin_custom.css">
@stop
@section('js')
    <script> console.log('Hi!'); </script>
@stop
```



# Isi Konten Detail Data Buku

resources > views > buku > show.blade.php > ...

```
6  @section('content')
7      @foreach($ar_buku as $b)
8          <div class="media">
9              @php
10                  if(!empty($b->cover)){
11                      @endphp
12                      
13                      @php
14                  }else{
15                      @endphp
16                      
17                      @php
18                  }
19                  @endphp
20          <div class="media-body">
21              <h3><u>{{ $b->judul }}</u></h3>
22              <p>
23                  ISBN : {{ $b->isbn }}
24                  <br/>Tahun Cetak : {{ $b->tahun_cetak }} <br/>Stok : {{ $b->stok }}
25                  <br/>Pengarang : {{ $b->nama }} <br/>Penerbit : {{ $b->pen }} <br/>Kategori : {{ $b->kat }}
26              </p>
27              <hr/><a href="{{ url('/buku') }}" class="btn btn-primary">Go Back</a>
28          </div>
29      </div>
30  @endforeach
31  @stop
```

# Tampilan Detail Data Buku

AdminLTE

search

Pages

4

ACCOUNT SETTINGS

Profile

Change Password

Multi Level

LABELS

Important

Warning

Information

Detail Data Buku

Book Cover  
Not  
Available

Menanam Jagung Unggul

ISBN : 112  
Tahun Cetak : 2021  
Stok : 15  
Pengarang : Dedi Iskandar  
Penerbit : nama\_60cc0ca9a1dd8  
Kategori : Pertanian

Go Back



**TERIMA KASIH  
ATAS SEGALA PERHATIAN  
SEMOGA BERMANFAAT...**

