

Modul 2

Database – Konfigurasi – Model buku

Buat Database

- Buat database: perpustakaan
- Buat table: mst_buku
- Struktur table mst_buku

Field	Type Data	Ukuran
ID_Buku	Integer	8, auto increment
Judul	Varchar	100
Pengarang	Varchar	150
Kategori	Varchar	10

Konfigurasi Database (.env)

- buka file **.env** di folder C:\framework\perpus\.env
- Tentukan hostname, username, password, dan nama database.
Contoh:

```
11    DB_CONNECTION=mysql
12    DB_HOST=127.0.0.1
13    DB_PORT=3306
14    DB_DATABASE=perpus
15    DB_USERNAME=root
16    DB_PASSWORD=
```

Konfigurasi Database (config/database.php)

- Masuk ke direktori config/, lalu buka file **database.php**. Tentukan hostname, username, password, dan nama database. Contoh:

```
46      'mysql' => [  
47          'driver' => 'mysql',  
48          'url' => env('DATABASE_URL'),  
49          'host' => env('DB_HOST', '127.0.0.1'),  
50          'port' => env('DB_PORT', '3306'),  
51          'database' => env('DB_DATABASE', 'perpus'),  
52          'username' => env('DB_USERNAME', 'root'),  
53          'password' => env('DB_PASSWORD', ''),  
54          'unix_socket' => env('DB_SOCKET', ''),  
55          'charset' => 'utf8mb4',  
56          'collation' => 'utf8mb4_unicode_ci',  
57          'prefix' => '',  
58          'prefix_indexes' => true,  
59          'strict' => true,  
60          'engine' => null,  
61          'options' => extension_loaded('pdo_mysql') ? array_filter([  
62              PDO::MYSQL_ATTR_SSL_CA => env('MYSQL_ATTR_SSL_CA'),  
63          ]) : [],  
64      ],  
--
```

Buat Eloquent Model Buku (1)

- Buat file model **Buku_m.php** menggunakan artisan:

```
PS C:\framework\perpus> C:\xampp\php\php artisan make:model Buku_m
Model created successfully.
PS C:\framework\perpus> 
```

- Hasil generate menggunakan artisan:
 - C:\framework\perpus\app\Models\Buku_m.php

```
app > Models > Buku_m.php
1  <?php
2
3  namespace App\Models;
4
5  use Illuminate\Database\Eloquent\Factories\HasFactory;
6  use Illuminate\Database\Eloquent\Model;
7
8  class Buku_m extends Model
9  {
10     use HasFactory;
11 }
12
```

Edit Eloquent Model “Buku_m”

- Edit isi file model “Buku_m.php” sbb:

```
app > Models >  Buku_m.php
```

```
1  <?php
2
3  namespace App\Models;
4
5  use Illuminate\Database\Eloquent\Factories\HasFactory;
6  use Illuminate\Database\Eloquent\Model;
7
8  class Buku_m extends Model
9  {
10     use HasFactory;
11
12     protected $table = 'mst_buku';
13     protected $primaryKey = 'ID_Buku';
14     public $timestamps = false;
15 }
--
```

Menguji Eloquent Model “Buku_m” menggunakan tinker

- Jalankan tinker menggunakan artisan

```
PS C:\framework\perpus> C:\xampp\php\php artisan tinker
Psy Shell v0.11.8 (PHP 7.4.29 - cli) by Justin Hileman
>>> █
```

- Beberapa fungsi Eloquent model laravel yang dapat kita coba sbb:
 - all()
 - first()
 - find()
 - where() → get()

Menguji Eloquent Model “Buku_m” menggunakan tinker

- Fungsi all() untuk mendapatkan semua record

```
>>> App\Models\Buku_m::all()
```

- Hasil

```
>>> App\Models\Buku_m::all()
=> Illuminate\Database\Eloquent\Collection {#3632
  all: [
    App\Models\Buku_m {#3633
      ID_Buku: 1,
      Judul: "Senja Kelabu",
      Pengarang: "Mutia",
      Kategori: "komik",
    },
    App\Models\Buku_m {#3634
      ID_Buku: 2,
      Judul: "Donald Bebek",
      Pengarang: "disney",
      Kategori: "komik",
    },
  ],
}
```

```
App\Models\Buku_m {#3615
  ID_Buku: 2,
  Judul: "Donald Bebek",
  Pengarang: "disney",
  Kategori: "komik",
},
App\Models\Buku_m {#3616
  ID_Buku: 5,
  Judul: "UML",
  Pengarang: "rosa",
  Kategori: "pemrograman",
},
App\Models\Buku_m {#3617
  ID_Buku: 6,
  Judul: "Pemrograman Web",
}
```

Dst....

Menguji Eloquent Model “Buku_m” menggunakan tinker

- Fungsi first() untuk mendapatkan record pertama

```
>>> App\Models\Buku_m::first()
```

- Hasil

```
>>> App\Models\Buku_m::first()  
=> App\Models\Buku_m {#3604  
    ID_Buku: 1,  
    Judul: "Senja Kelabu",  
    Pengarang: "Mutia",  
    Kategori: "komik",  
}
```

Menguji Eloquent Model “Buku_m” menggunakan tinker

- Fungsi find() untuk mendapatkan record sesuai ID_Buku

```
>>> App\Models\Buku_m::find(6)
```

- Hasil

```
>>> App\Models\Buku_m::find(6)
=> App\Models\Buku_m {#3647
  ID_Buku: 6,
  Judul: "Pemrograman Web",
  Pengarang: "April",
  Kategori: "novel",
}
```

Menguji Eloquent Model “Buku_m” menggunakan tinker

- Fungsi where()->get() untuk mendapatkan record sesuai kriteria

```
>>> App\Models\Buku_m::where('Kategori','Novel')->get()
```

- Hasil

```
>>> App\Models\Buku_m::where('Kategori','Novel')->get()
=> Illuminate\Database\Eloquent\Collection {#3593
  all: [
    App\Models\Buku_m {#3598
      ID_Buku: 6,
      Judul: "Pemrograman Web",
      Pengarang: "April",
      Kategori: "novel",
    },
    App\Models\Buku_m {#3603
      ID_Buku: 7,
      Judul: "Pemrograman Web",
      Pengarang: "April",
      Kategori: "novel",
    },
    App\Models\Buku_m {#3647
      ID_Buku: 11,
      Judul: "Mereka",
      Pengarang: "aku",
      Kategori: "novel",
    },
  ],
}
```

Uji Insert data Eloquent Model "Buku_m"

- Ikuti perintah berikut:

```
PS C:\framework\perpus> c:\xampp\php\php artisan tinker
Psy Shell v0.11.8 (PHP 7.4.29 - cli) by Justin Hileman
>>> $buku = new App\Models\Buku_m
=> App\Models\Buku_m {#3587}

>>> $buku->Judul = "Buku 1"
=> "Buku 1"

>>> $buku->Pengarang = "Pengarang 1"
=> "Pengarang 1"

>>> $buku->Kategori = "Pemrograman"
=> "Pemrograman"

>>> $buku->save()
=> true

>>> █
```

- Hasil

<input type="checkbox"/>	 Edit	 Copy	 Delete	31 judul 1	pengarang 1	pemrograman
<input type="checkbox"/>	 Edit	 Copy	 Delete	32 judul 2	pengarang 3	komik
<input type="checkbox"/>	 Edit	 Copy	 Delete	33 Buku 1	Pengarang 1	Pemrograman



Uji Update data Eloquent Model "Buku_m"

- Ikuti perintah berikut:

```
PS C:\framework\perpus> c:\xampp\php\php artisan tinker
Psy Shell v0.11.8 (PHP 7.4.29 - cli) by Justin Hileman
>>> $buku = App\Models\Buku_m::find(33)
=> App\Models\Buku_m {#3597
    ID_Buku: 33,
    Judul: "Buku 1",
    Pengarang: "Pengarang 1",
    Kategori: "Pemrograman",
}

>>> $buku->Judul = "Buku 11111"
=> "Buku 11111"

>>> $buku->save()
=> true

>>> █
```

- Hasil

<input type="checkbox"/>	 Edit	 Copy	 Delete	31 judul 1	pengarang 1	pemrograman
<input type="checkbox"/>	 Edit	 Copy	 Delete	32 judul 2	pengarang 3	komik
<input type="checkbox"/>	 Edit	 Copy	 Delete	33 Buku 11111	Pengarang 1	Pemrograman

Uji Delete data Eloquent Model "Buku_m"

- Ikuti perintah berikut:

```
PS C:\framework\perpus> c:\xampp\php\php artisan tinker
Psy Shell v0.11.8 (PHP 7.4.29 - cli) by Justin Hileman
>>> $buku = App\Models\Buku_m::find(33)
=> App\Models\Buku_m {#3597
    ID_Buku: 33,
    Judul: "Buku 11111",
    Pengarang: "Pengarang 1",
    Kategori: "Pemrograman",
}

>>> $buku->delete()
=> true
```

- Hasil

<input type="checkbox"/>	 Edit	 Copy	 Delete	31 judul 1	pengarang 1	pemrograman
<input type="checkbox"/>	 Edit	 Copy	 Delete	32 judul 2	pengarang 3	komik

Edit file model “Buku_m.php”

- Tambahkan method berikut:
 - `get_records()` → untuk mengambil data
 - `insert_record()` → untuk menambah data
 - `update_by_id()` → untuk mengupdate data
 - `delete_by_id()` → untuk menghapus data

get_records()

```
16 function get_records($criteria='')
17 {
18     $result = self::select('*')
19         ->when($criteria, function ($query, $criteria) {
20             return $query->where('ID_Buku',$criteria);
21         })
22         ->get();
23     return $result;
24 }
```


insert_record(), update_by_id(), delete_by_id()

```
35 function insert_record($data)
36 {
37     $result = self::insert($data);
38     return $result;
39 }
40
41 function update_by_id($data, $id)
42 {
43     $result = self::where('ID_Buku', $id)
44     |>update($data);
45     return $result;
46 }
47
48 function delete_by_id($id)
49 {
50     $result = self::where('ID_Buku', $id)
51     |>delete();
52     return $result;
53 }
```


Uji insert_record()

```
PS C:\framework\perpus> c:\xampp\php\php artisan tinker
Psy Shell v0.11.8 (PHP 7.4.29 - cli) by Justin Hileman
>>> $buku = new App\Models\Buku_m
=> App\Models\Buku_m {#3587}

>>> $data = [
...     'Judul' => 'Doraemon',
...     'Pengarang' => 'Dino',
...     'Kategori' => 'Komik'
... ]
=> [
    "Judul" => "Doraemon",
    "Pengarang" => "Dino",
    "Kategori" => "Komik",
]

>>> $buku->insert_record($data)
=> true
```

Hasil:

<input type="checkbox"/>	 Edit	 Copy	 Delete	31 judul 1	pengarang 1	pemrograman
<input type="checkbox"/>	 Edit	 Copy	 Delete	35 Doraemon 	Dino	Komik







Uji get_records()

```
>>> $buku->get_records()
=> Illuminate\Database\Eloquent\Collection {#3614
  all: [
    App\Models\Buku_m {#3615
      ID_Buku: 1,
      Judul: "Senja Kelabu",
      Pengarang: "Mutia",
      Kategori: "komik",
    },
    App\Models\Buku_m {#3616
      ID_Buku: 2,
      Judul: "Donald Bebek",
      Pengarang: "disney",
      Kategori: "komik",
    },
    App\Models\Buku_m {#3617
      ID_Buku: 5,
      Judul: "UML",
      Pengarang: "rosa",
      Kategori: "pemrograman",
    },
  ],
}
```

Uji update_by_id()

```
>>> $data = [  
... 'Judul' => 'Petualangan Doraemon',  
... 'Pengarang' => 'Fujiko Fujio'  
... ]  
=> [  
  "Judul" => "Petualangan Doraemon",  
  "Pengarang" => "Fujiko Fujio",  
]  
  
>>> $buku->update_by_id($data,35)  
=> 1
```

Hasil:

<input type="checkbox"/>	 Edit	 Copy	 Delete	31 judul 1	pengarang 1	pemrograman
<input type="checkbox"/>	 Edit	 Copy	 Delete	35 Petualangan Doraemon	Fujiko Fujio	Komik







Uji delete_by_id()

```
>>> $buku->delete_by_id(35)
```

```
=> 1
```

```
>>> █
```

Hasil:

<input type="checkbox"/>	 Edit	 Copy	 Delete	30	aku dan kamu	chairil anwar	novel
<input type="checkbox"/>	 Edit	 Copy	 Delete	31	judul 1	pengarang 1	pemrograman

