
LAPORAN PRAKTIKUM MAHASISWA PEMROGRAMAN WEB FRAMEWORK

DATABASE



Oleh:

Nama : Mukaromah

Npm : 21753059

Kelas : Manajemen Informatika 4 B

Dosen:

Oki Arifin, S.Kom., M.Cs.

**D3 – MANAJEMEN INFORMATIKA
JURUSAN EKONOMI DAN BISNIS
POLITEKNIK NEGERI LAMPUNG**

2023

A. Tujuan

Setelah praktikum ini, praktikan diharapkan dapat:

1. Mampu memahami tentang framework php
2. Mampu mengimplementasikan perintah pada framework php
3. Mahasiswa dapat Menggunakan framework php

B. Alat dan Bahan

1. Visual studio code
2. MS. Word
3. Codeignite
4. Gitbash
5. Composer

C. Hasil dan Pembahasan

1. Membuat database baru dengan nama db_film



2. membuat tabel baru dengan nama film dan colom sesuai dengan kebutuhan

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id	int(11)			No	None		AUTO_INCREMENT	Change Drop More
2	nama_film	varchar(200)	utf8mb4_general_ci		No	None			Change Drop More
3	genre	varchar(200)	utf8mb4_general_ci		No	None			Change Drop More
4	duration	varchar(50)	utf8mb4_general_ci		No	None			Change Drop More
5	cover	varchar(200)	utf8mb4_general_ci		No	None			Change Drop More
6	created_at	timestamp			No	current_timestamp()			Change Drop More
7	updated_at	timestamp			No	current_timestamp()			Change Drop More

3. menambahkan data pada tabel film

id	id_genre	nama_film	duration	cover
1	0	Sewu Dino	2 jam 1 menit	sewu-dino.jpg
2	0	KKN Didesa Penari	2 jam 1 menit	kkn.jpg
3	0	Ivanna	1 jam 43 menit	ivana.jpg
4	0	Mencuri Raden Saleh	2 jam 43 menit	mencuri.jpg
5	0	Akhirat	2 jam 43 menit	akhirat.jpg
6	0	Keluarga Cemara	1 jam 30 menit	cemara.jpg
7	0	Mumun Pocong	1 jam 28 menit	mumun.jpg
8	0	Warkop DKI	1 jam 15 menit	dki.jpg
9	0	Ayat-ayat Cinta	2 jam 30 menit	ayat.jpg
10	0	Laskar Pelangi	1 jam 43 menit	laskar.jpg

4. menghubungkan codeigniter dengan database dengan mengedit file .env

```

database.default.hostname = localhost
database.default.database = db_film
database.default.username = root
database.default.password =
database.default.DBDriver = MySQLi
database.default.DBPrefix =
database.default.port = 3306

```

5. membuat file pada controller/film.php

```

<?php

namespace App\Controllers;
use App\Controllers\BaseController;
use App\Models\FilmModel;

class Film extends BaseController
{
    //step 2
    protected $Film;

    public function __construct()
    {
        //step 4 memanggil
        $this -> Film = new FilmModel();
    }

    //fungsi yang akan diakses pada url nanti
    public function all(){
        $data['semuaFilm'] = $this -> Film -> getAllData();
        return view("Film/semuaFilm", $data);
    }
}

```

pada bagian function all, digunakan untuk pengaksesan pada url nanti dan untuk bagian getAllData digunakan untuk memanggil function pada bagian model

6. membuat file pada Model/FilmModel.php

```
<?php

namespace App\Models;

use CodeIgniter\Model;

class FilmModel extends Model
{
    //menyambungkan ke dalam tabel film secara langsung
    protected $table = 'film';
    //apabila akan insert di database maka di secara langsung
    akan memberikan id dengan melanjutkan id yang sudah ada
    protected $primaryKey = 'id';
    //ketika insert menggunakan fungsi ci di akan melakukan
    autoincrement pada data yg kita tambahkan
    protected $useAutoIncrement = true;
    //mengisinkan apakah data dapat di update, delete dan insert
    pada kolom
    protected $allowField = [];

    //fungsi untuk menampilkan seluruh data
    public function getAllData(){
        //return mengembalikan data
        return $this -> FindAll();
    }
}
```

7. membuat file pada view/semuaFilm.php yang digunakan untuk menampilkan hasil yang telah di kerjakan

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-
scale=1.0">
    <title>data film</title>
</head>
<body>
    <h1>Data Film</h1>
    <table border="1" cellspacing="2" cellpadding="5">
```

```

<tr>
    <th>No</th>
    <th>Cover</th>
    <th>Nama Film</th>
    <th>Genre</th>
    <th>Durasi</th>

</tr>
<?php $i = 1 ;?>
<?php foreach($semuaFilm as $film): ?>
    <tr>
        <td><?= $i++;?></td>
        <td >
            
            </td>
        <td><?php echo $film['nama_film']?></td>
        <td><?= $film['genre']?></td>
        <td><?= $film['duration']?></td>
    </tr>
<?php endforeach;?>
</table>
</body>
</html>

```

8. menampilkan data yang telah di buat dengan perintah localhost:8080/film/all

Data Film

No	Cover	Nama Film	Genre	Durasi
1		Sewu Dino	Horor	2 jam 1 menit
2		KKN Didesa Penari	Horor	2 jam 1 menit
3		Ivanna	Horor	1 jam 43 menit
4		Mencuri Raden Saleh	Laga	2 jam 43 menit
5		Akhirat	Drama	2 jam 43 menit

9. menampilkan data dengan kondisi sesuai dengan id, dengan menggunakan find (\$id)

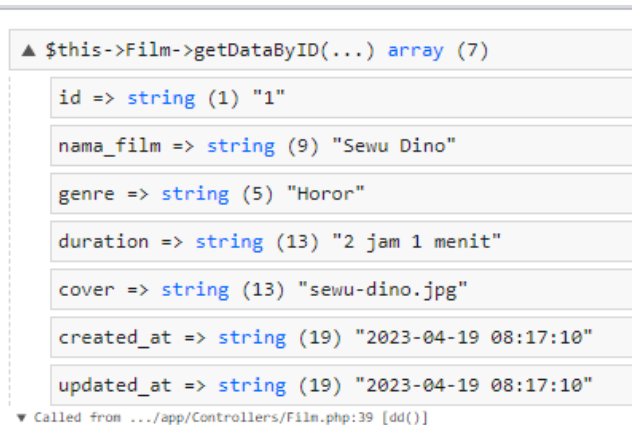
tambahan fungsi pada FilmModel.php

```
//fungsi untuk mencari data sesuai dengan id
public function getDataById($id){
    return $this->find ($id);
}
```

selanjutnya panggil pada controller/Film.php

```
//memanggil fungsi getdatabyid yang akan diakses pada url nanti
public function find_byid(){
    //memanggil fungsi getDataByID pada model
    dd($this->Film -> getDataByID(1));
}
```

hasilnya :



```
▲ $this->Film->getDataByID(...) array (7)
  id => string (1) "1"
  nama_film => string (9) "Sewu Dino"
  genre => string (5) "Horor"
  duration => string (13) "2 jam 1 menit"
  cover => string (13) "sewu-dino.jpg"
  created_at => string (19) "2023-04-19 08:17:10"
  updated_at => string (19) "2023-04-19 08:17:10"
▼ Called from .../app/Controllers/Film.php:39 [dd()]
```

10. menampilkan data dengan kondisi sesuai dengan judul, dengan menggunakan where

tambahan fungsi pada FilmModel.php

```
//fungsi untuk mencari data sesuai dengan where/kondisi yang di cari
public function getDataBy($data){
    return $this->where ('nama_film',| $data)-> findAll();
}
```

selanjutnya panggil pada controller/Film.php

```
//memanggil fungsi getdataby yang akan diakses pada url nanti
public function find_bywhere(){
    //memanggil fungsi getDataBy pada model
    dd($this->Film -> getDataBy('mumun pocong'));
}
```

hasilnya :

▲ \$this->Film->getDataBy(...) array (1)

▲ 0 => array (7)
id => string (1) "7"
nama_film => string (12) "Mumun Pocong"
genre => string (5) "Horor"
duration => string (14) "1 jam 28 menit"
cover => string (9) "mumun.jpg"
created_at => string (19) "2023-06-07 08:28:01"
updated_at => string (19) "2023-06-07 08:28:01"

▼ Called from .../app/Controllers/Film.php:45 [dd()]

11. menampilkan data dengan fungsi orderby, dengan menggunakan orderby pada bagian ini data created_at akan di urutkan.

tambahan fungsi pada FilmModel.php

```
//fungsi untuk mencari data sesuai dengan order by dan akan di urutkan
public function getOrderby(){
    return $this -> orderBy ('created_at', 'desc')-> findAll();
}
```

selanjutnya panggil pada controller/Film.php

```
//memanggil fungsi getorderby yang akan diakses pada url nanti
public function find_ordeby(){
    //memanggil fungsi getOrderby pada model
    dd($this->Film -> getOrderby());
}
```

hasilnya :

▲ \$this->Film->getOrderBy() array (10)

Table (10)

Contents (10)

	id	nama_film	genre	duration	cover	created_at	updated_at
0	10	Laskar Pelangi	Drama	1 jam 43 menit	laskar.jpg	2023-06-07 08:31:23	2023-06-07 08:31:23
1	9	Ayat-ayat Cinta	Romansa	2 jam 30 menit	ayat.jpg	2023-06-07 08:28:06	2023-06-07 08:28:06
2	5	Akhirat	Drama	2 jam 43 menit	akhirat.jpg	2023-06-07 08:28:01	2023-06-07 08:28:01
3	6	Keluarga Cemara	Drama	1 jam 30 menit	cemara.jpg	2023-06-07 08:28:01	2023-06-07 08:28:01
4	7	Mumun Pocong	Horor	1 jam 28 menit	mumun.jpg	2023-06-07 08:28:01	2023-06-07 08:28:01
5	8	Warkop DKI	Comedy	1 jam 15 menit	dki.jpg	2023-06-07 08:28:01	2023-06-07 08:28:01
6	1	Sewu Dino	Horor	2 jam 1 menit	sewu-dino.jpg	2023-04-19 08:17:10	2023-04-19 08:17:10
7	2	KKN Didesa Penari	Horor	2 jam 1 menit	kkn.jpg	2022-12-29 08:17:10	2022-12-29 08:17:10
8	4	Mencuri Raden Saleh	Laga	2 jam 43 menit	mencuri.jpg	2022-08-25 08:21:10	2022-08-25 08:21:10
9	3	Ivanna	Horor	1 jam 43 menit	ivana.jpg	2022-07-14 08:21:10	2022-07-14 08:21:10

▼ Called from .../app/Controllers/Film.php:51 [dd()]

12. menampilkan data dengan fungsi limit, dengan menggunakan limit untuk menampilkan data sesuai dengan limit yang telah di tentukan
tambahan fungsi pada FilmModel.php

```
//memanggil fungsi limit yang akan diakses pada url nanti
public function find_limit(){
    //memanggil fungsi getLimit data pada model
    dd($this->Film -> getLimit());
}
```

selanjutnya panggil pada controller/Film.php

```
//fungsi mengambil data sebanyak limit yang dimasukkan
public function getLimit(){
    return $this -> limit (5) -> get() -> getResultArray();
}
```

hasilnya :

▲ \$this->Film->getLimit() array (5)

Table (5)		Contents (5)					
	id	nama_film	genre	duration	cover	created_at	updated_at
0	1	Sewu Dino	Horor	2 jam 1 menit	sewu-dino.jpg	2023-04-19 08:17:10	2023-04-19 08:17:10
1	2	KKN Didesa Penari	Horor	2 jam 1 menit	kkn.jpg	2022-12-29 08:17:10	2022-12-29 08:17:10
2	3	Ivanna	Horor	1 jam 43 menit	ivana.jpg	2022-07-14 08:21:10	2022-07-14 08:21:10
3	4	Mencuri Raden Saleh	Laga	2 jam 43 menit	mencuri.jpg	2022-08-25 08:21:10	2022-08-25 08:21:10
4	5	Akhirat	Drama	2 jam 43 menit	akhirat.jpg	2023-06-07 08:28:01	2023-06-07 08:28:01

13. menampilkan data dengan fungsi builder, dengan menggunakan builder maka yang di tampilkan hanya kolom tertentu pada tabel
tambahan fungsi pada FilmModel.php

```
//memanggil fungsi sesuai dengan kolom yang akan diakses pada url nanti
public function find_kolom(){
    //memanggil fungsi getFilmAll pada model
    dd($this->Film -> getFilmAll());
}
```

selanjutnya panggil pada controller/Film.php

```
//fungsi yang di gunakan untuk menampilkan kolom tertentu pada tabel
public function getFilmAll(){
    $builder = $this->db->table('film');
    $builder ->select('film.nama_film');
    $query = $builder -> get ();
    return $query -> getResult();
}
```


hasilnya :



D. Tugas

1. Menambhakan tabel pada database db_film dengan nama tabel genre

id_genre	nama_genre	created_at	updated_at
1	Horor	2023-06-13 00:46:51	2023-06-14 00:46:51
2	Comedy	2023-06-04 00:46:51	2023-06-08 00:46:51
3	Romantis	2021-05-01 00:49:40	2022-06-30 00:49:40

2. Menampilkan data genre d ihalaman html, jika untuk di menampilkan data film kita film/all untuk menampilkan data genre aksesnya adalah genre/all

- membuat file controller/genre

```
<?php
namespace App\Controllers;
use App\Controllers\BaseController;
```

```
//step 1
use App\Models\GenreModel;

class Genre extends BaseController
{
    //step 2
    protected $Genre;
    //step 3 membuat fungsi construct untuk inisiasi class
    model(filmmodel)
    public function __construct()
    {
        //step 4 memanggil
        $this -> Genre = new GenreModel();
    }

    $data['semuaGenre'] = $this -> Genre -> getAllData();
    return view("Genre/semuaGenre", $data);
}
}
```

- membuat file model/GenreModel.php

```
<?php

namespace App\Models;

use CodeIgniter\Model;

class GenreModel extends Model
{
    //menyambungkan ke dalam tabel film secara langsung
    protected $table = 'genre';
    //apabila akan insert di database maka di secara
    langsung akan memberikan id dengan melanjutkan id yang
    sudah ada
    protected $primaryKey = 'id';
    //ketika insert menggunakan fungsi ci di akan melakukan
    autoincrement pada data yg kita tambahkan
    protected $useAutoIncrement = true;
    //mengisinkan apakah data dapat di update, delete dan
    insert pada kolom
    protected $allowField = [];

    //fungsi untuk menampilkan seluruh data
```

```

    public function getAllData(){
        //return mengembalikan data
        return $this -> FindAll();
    }
}

```

- membuat file view/semuagenre.php

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <title>data film</title>
</head>
<body>
    <h1>Data Genre</h1>
    <table border ="1" cellspacing="2" cellpadding="5">
        <tr>
            <th>No</th>
            <th>Nama genre</th>
            <th>Dibuat</th>
            <th>Diupdate</th>

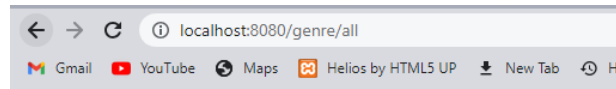
        </tr>

        <?php $i = 1 ;?>
        <?php foreach($semuaGenre as $Genre): ?>
            <tr>
                <td><?= $i++;?></td>
                <td><?php echo $Genre['nama_genre']?></td>
                <td><?php echo $Genre['created_at']?></td>
                <td><?= $Genre['updated_at']?></td>

            </tr>
        <?php endforeach;?>
    </table>
</body>
</html>

```

Hasilnya :



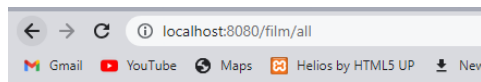
Data Genre

No	Nama genre	Dibuat	Diupdate
1	Horor	2023-06-13 00:46:51	2023-06-14 00:46:51
2	Comedy	2023-06-04 00:46:51	2023-06-08 00:46:51
3	Romantis	2021-05-01 00:49:40	2022-06-30 00:49:40

3. ubah tabel film pada kolom genre menjadi id_genre,, isi datanya sesuai dengan data pada tabel genre

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id	int(11)			No	None		AUTO_INCREMENT
2	nama_film	varchar(200)	utf8mb4_general_ci		No	None		
3	id_genre	int(11)			No	None		
4	duration	varchar(50)	utf8mb4_general_ci		No	None		
5	cover	varchar(200)	utf8mb4_general_ci		No	None		
6	created_at	timestamp			No	current_timestamp()		
7	updated_at	timestamp			No	current_timestamp()		

4. saat menjalankan film/all pastikan tidak ada error, artinya codingnya harus disesuaikan karena mengganti genre menjadi id genre.



Data Film

No	Cover	Nama Film	Id genre	Durasi
1		sewu Dino	1	2 jam 1 menit
2		Ayat - ayat Cinta	3	2 jam 30 menit
3		Warkop DKI	2	1jam 15 menit

link github :

<https://github.com/roma84011/Pemrograman-Framework/tree/82cc8806cca9dcf116ac61d3b912fed6ecda1fc9/Database-film-genre>

E. Daftar Pustaka

Daqiqil, Ibnu. Framework Codeigniter : Sebuah Panduan dan Best Practice. 2011.

Tutorislpoint. Codeigniter. 2015. www.tutorialspoint.com