

Kelompok : A2

```
C:\xampp\htdocs>composer create-project --prefer-dist laravel/laravel:^12.0 laravel-12
```

Langkah 3: Jalankan Server Lokal

A.)Masuk ke folder proyek:

```
C:\xampp\htdocs>cd laravel-12
```

B.)Lalu jalankan server lokal:

```
C:\xampp\htdocs\laravel-12>php artisan serve
```

Nanti akan muncul URL seperti:

<http://127.0.0.1:8000/>

Buka URL itu di browser, dan... selamat! Laravel kamu sudah berhasil diinstall!

Tutorial Setup Laravel: Dari Konfigurasi Sampai Migration

Halo teman-teman!

Di sesi ini kita akan belajar langkah-langkah penting setelah install Laravel, yaitu:

- Mengatur koneksi database
- Membuat storage link
- Membuat model + migration
- Menjalankan migration

Langsung aja, kita mulai!

1. Konfigurasi Koneksi Database di .env

Pertama-tama, kita harus menghubungkan Laravel ke database.

Buka file `.env` di project kamu, lalu cari bagian ini dan sesuaikan:

```
FILESYSTEM_DISK=local
```

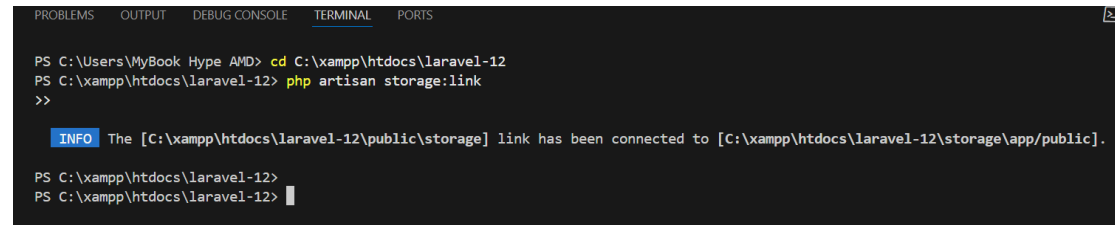
Ganti local dengan public lalu save filenya

```
FILESYSTEM_DISK=public
```

2. Membuat Storage Link

Laravel menyimpan file upload di folder storage, tapi supaya file tersebut bisa diakses dari browser, kita harus membuat symbolic link dulu.

Kita buka terminal di VSCode lalu ketikkan



```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\Users\MyBook Hype AMD> cd C:\xampp\htdocs\laravel-12
PS C:\xampp\htdocs\laravel-12> php artisan storage:link
>>

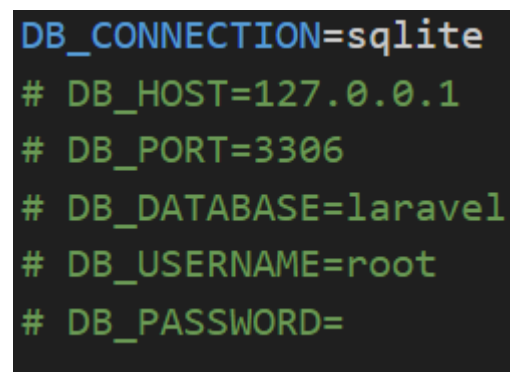
[INFO] The [C:\xampp\htdocs\laravel-12\public\storage] link has been connected to [C:\xampp\htdocs\laravel-12\storage\app/public].
PS C:\xampp\htdocs\laravel-12>
PS C:\xampp\htdocs\laravel-12>
```

3. Membuat Model dan Migration

Sekarang kita buat model dan migration secara bersamaan.

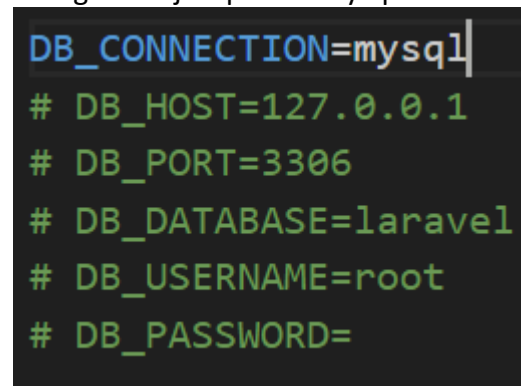
Konfigurasi Koneksi Database :

Masuk ke file .env lalu cari sqlite



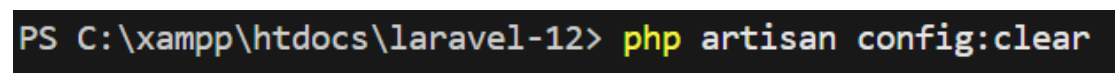
```
DB_CONNECTION=sqlite
# DB_HOST=127.0.0.1
# DB_PORT=3306
# DB_DATABASE=laravel
# DB_USERNAME=root
# DB_PASSWORD=
```

Lalu ganti saja sqlite ke mysql dan save filenya:



```
DB_CONNECTION=mysql
# DB_HOST=127.0.0.1
# DB_PORT=3306
# DB_DATABASE=laravel
# DB_USERNAME=root
# DB_PASSWORD=
```

Setelah itu masuk ke terminal lagi yang ada di filr .env dan ketikkan :



```
PS C:\xampp\htdocs\laravel-12> php artisan config:clear
```

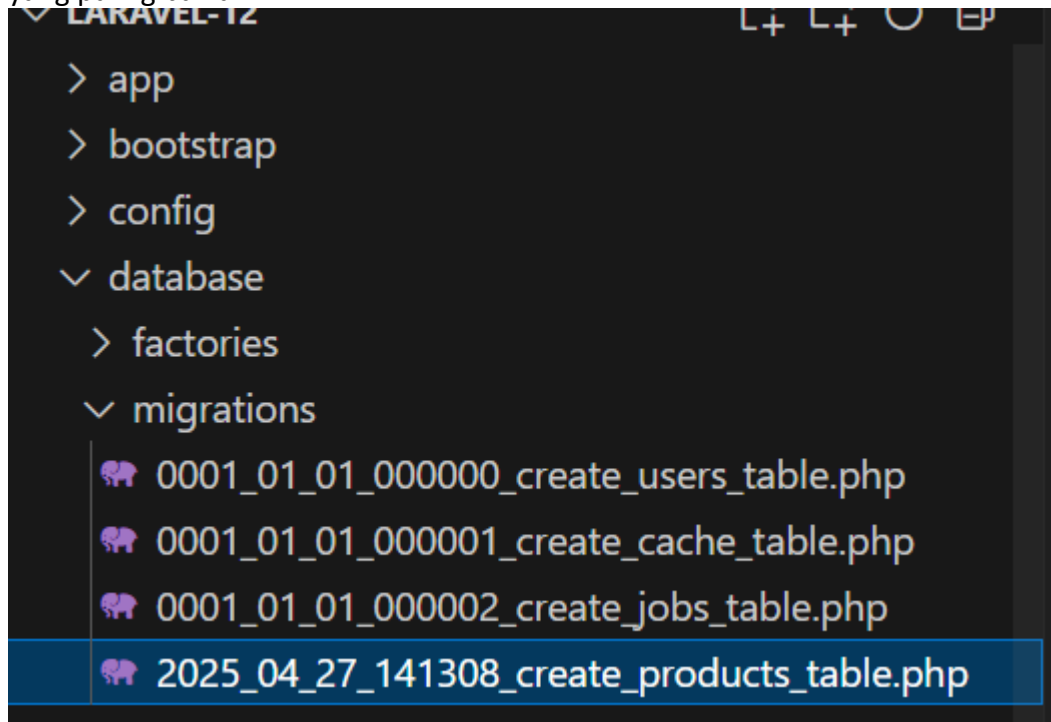
Membuat Model dan Migration

Sekarang kita buat model dan migration secara bersamaan.

Misalnya, kita mau buat model Produk, ketik perintah ini:

```
PS C:\xampp\htdocs\laravel-12> php artisan make:model Product -m  
INFO Model [C:\xampp\htdocs\laravel-12\app\Models\Product.php] created successfully.  
INFO Migration [C:\xampp\htdocs\laravel-12\database\Migrations\2025_04_27_141308_create_products_table.php] created successfully.  
PS C:\xampp\htdocs\laravel-12>
```

Setelah itu kita buka file lewat VSCode dari database/migrations dan pilih file php yang paling bawah :



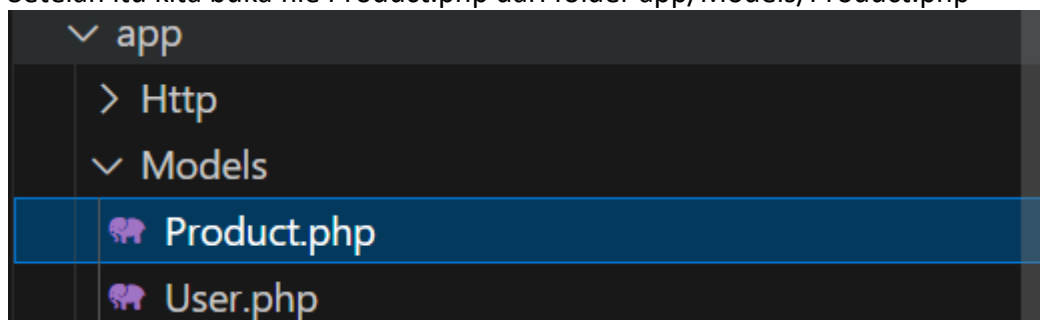
Dan replace isi codingnya dengan ini :

```
public function up(): void  
{  
    Schema::create('products', function (Blueprint $table) {  
        $table->id();  
        $table->string('image');  
        $table->string('title');  
        $table->text('description');  
        $table->bigInteger('price');  
        $table->integer('stock')->default(0);  
        $table->timestamps();  
    });  
}
```

Pastikan hanya yang fungsi up saja

```
public function up(): void
{
    Schema::create('products', function (Blueprint $table) {
        $table->id();
        $table->string('image');
        $table->string('title');
        $table->text('description');
        $table->bigInteger('price');
        $table->integer('stock')->default(0);
        $table->timestamps();
    });
}
```

Setelah itu kita buka file Product.php dari folder app/Models/Product.php



Setelahnya replace codingannya dengan ini dan save filenya :

```
<?php
```

```
namespace App\Models;
```

```
use Illuminate\Database\Eloquent\Factories\HasFactory;
```

```
use Illuminate\Database\Eloquent\Model;
```

```
class Product extends Model
```

```
{
```

```
    use HasFactory;
```

```
    /**
```

```
     * fillable
```

```
     *
```

```
     * @var array
```

```
     */
```

```
    protected $fillable = [
```

```
        'image',
```

```
        'title',
        'description',
        'price',
        'stock',
    ];
}
```

```
app > Models > Product.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 Product extends Model
9  {
10     use HasFactory;
11
12     /**
13      * fillable
14      *
15      * @var array
16      */
17     protected $fillable = [
18         'image',
19         'title',
20         'description',
21         'price',
22         'stock',
23     ];
24 }
```

5. Menjalankan Migration

Kita ketik saja lewat terminal di VSCode :

```
PS C:\xampp\htdocs\laravel-12> php artisan migrate
```

Nanti akan ada konfirmasi , ketik saja yes

```
WARN The database 'db_laravel_12' does not exist on the 'mysql' connection.

┌ Would you like to create it? _____
│ ● Yes / ○ No
└
```

```
php artisan migrate

WARN The database 'db_laravel_12' does not exist on the 'mysql' connection.

Would you like to create it?
Yes

INFO Preparing database.
Creating migration table ..... 25.48ms DONE
INFO Running migrations.
0001_01_01_000000_create_users_table ..... 28.28ms DONE
0001_01_01_000001_create_cache_table ..... 6.52ms DONE
0001_01_01_000002_create_jobs_table ..... 19.23ms DONE
2025_02_25_011257_create_products_table ..... 3.07ms DONE
```

Terakhir, cek pada PhpMyAdmin di <http://localhost/phpmyadmin/>

The screenshot shows the PhpMyAdmin web interface. The left sidebar displays a tree view of the database structure, including 'information_schema', 'laravel', 'mysql', 'performance_schema', 'phpmyadmin', and 'test'. The 'laravel' database is selected, and its tables are listed: 'cache', 'cache_locks', 'failed_jobs', 'jobs', 'job_batches', 'migrations', 'password_reset_tokens', 'products', 'sessions', and 'users'. The main panel shows the 'Structure' tab for the 'laravel' database. It displays a table of 10 tables with columns for 'Table', 'Action', 'Rows', 'Type', 'Collation', 'Size', and 'Overhead'. The 'Sum' row at the bottom indicates a total of 4 InnoDB tables with a total size of 240.0 KiB.

Table	Action	Rows	Type	Collation	Size	Overhead
cache	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	16.0 KiB	-
cache_locks	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	16.0 KiB	-
failed_jobs	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	32.0 KiB	-
jobs	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	32.0 KiB	-
job_batches	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	16.0 KiB	-
migrations	Browse Structure Search Insert Empty Drop	4	InnoDB	utf8mb4_unicode_ci	16.0 KiB	-
password_reset_tokens	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	16.0 KiB	-
products	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	16.0 KiB	-
sessions	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	48.0 KiB	-
users	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	32.0 KiB	-
10 tables	Sum	4	InnoDB	utf8mb4_general_ci	240.0 KiB	0 B