

Laravel Day 5

模型 (Model) 與遷移 (Migration)

這次我們要完成什麼？

今天我們將認識 Laravel 資料層操作的核心技術：

- 建立 `Task` 模型（代表一筆任務）
- 使用 migration 建立 `tasks` 資料表
- 練習欄位設計、執行 migrate、回滑 rollback
- 了解 Laravel 對命名與資料結構的規範

1. 什麼是 Model ?

- **Model（模型）** 是 Laravel 用來表示資料物件的層級
- 每個 model 預設會對應資料表

Task 模型 → tasks 資料表

✓ Model 將商業邏輯與資料儲存分層處理，維持程式架構整潔

2. 命名規則：Task vs. tasks

Laravel 使用 Eloquent ORM，強調「慣例優於設定」：

類型	命名	說明
Model	Task	使用單數，代表一筆資料
Table	tasks	使用複數，表示儲存多筆資料的表格

✅ 你只要命名正確，Laravel 自動幫你找對應關係！

3. 建立 Model + Migration

```
php artisan make:model Task -m
```


- `-m`：自動生成對應的 migration 檔

 成功後會產生：

- `app/Models/Task.php` → 模型類別
- `database/migrations/xxxx_create_tasks_table.php` → 建表設定檔

4. 編輯 Migration：定義資料表欄位

```
public function up(): void
{
    Schema::create('tasks', function (Blueprint $table) {
        $table->id();
        $table->string('title');
        $table->text('description');
        $table->text('long_description')->nullable();
        $table->boolean('completed')->default(false);
        $table->timestamps();
    });
}
```

 `nullable()` 表示允許空值；`timestamps()` 會自動建立 `created_at` 與 `updated_at` 欄位

5. 執行 Migration：建立資料表

```
php artisan migrate
```

🎉 資料表建立成功，會看到 `tasks` 表出現在資料庫中！

語言: 繁體中文

MariaDB » mysql » 資料庫: task-list

Adminer 5.3.0

資料庫: task-list

[SQL 命令](#) [匯入](#) [匯出](#)
[建立資料表](#)

[選擇 cache](#)
[選擇 cache_locks](#)
[選擇 failed_jobs](#)
[選擇 jobs](#)
[選擇 job_batches](#)
[選擇 migrations](#)
[選擇 password_reset_tokens](#)
[選擇 sessions](#)
[選擇 tasks](#)
[選擇 users](#)

資料庫: task-list

[修改資料庫](#) [資料庫結構](#) [權限](#)

資料表和檢視表

在資料庫搜尋 (10)

搜尋

<input type="checkbox"/>	資料表	引擎?	校對?	資料長度?	索引長度?	資料空間?	自動遞增?	行數?	註解?
<input type="checkbox"/>	cache	InnoDB	utf8mb4_unicode_ci	16,384	0	0	?	0	
<input type="checkbox"/>	cache_locks	InnoDB	utf8mb4_unicode_ci	16,384	0	0	?	0	
<input type="checkbox"/>	failed_jobs	InnoDB	utf8mb4_unicode_ci	16,384	16,384	0	1	0	
<input type="checkbox"/>	jobs	InnoDB	utf8mb4_unicode_ci	16,384	16,384	0	1	0	
<input type="checkbox"/>	job_batches	InnoDB	utf8mb4_unicode_ci	16,384	0	0	?	0	
<input type="checkbox"/>	migrations	InnoDB	utf8mb4_unicode_ci	16,384	0	0	5	~ 4	
<input type="checkbox"/>	password_reset_tokens	InnoDB	utf8mb4_unicode_ci	16,384	0	0	?	0	
<input type="checkbox"/>	sessions	InnoDB	utf8mb4_unicode_ci	16,384	32,768	0	?	0	
<input type="checkbox"/>	tasks	InnoDB	utf8mb4_unicode_ci	16,384	0	0	1	0	
<input type="checkbox"/>	users	InnoDB	utf8mb4_unicode_ci	16,384	16,384	0	1	0	
	總共 10 個	InnoDB		163,840	81,920	0			

延伸指令補充

指令	說明
<code>php artisan migrate</code>	執行全部尚未執行的 migration
<code>php artisan migrate:rollback</code>	回滾上次執行的 migration
<code>php artisan migrate:rollback --step=1</code>	回滾指定批次數
<code>php artisan migrate:refresh</code>	回滾 + 重新全部執行（重設資料表內容）
<code>php artisan migrate:reset</code>	回滾所有 migration，等同資料表清空

小結

- Model 是 Laravel 中與資料表對應的「資料代表」
- migration 是表結構的建造工具，可以回滾／刷新
- 遵守命名慣例（單數 model、複數 table）讓 Laravel 更好用！

 接下來我們會學 Seeder，自動產生假資料讓你方便測試