

# 使用 Laravel

如果你正在進入網頁開發領域，並希望找到一個既強大又易於上手的框架，Laravel 就是你的不二選擇。它是一個 PHP 框架，因其優雅的語法、強大的功能和支持你的社群而迅速風靡開發者世界。在這篇部落格中，我將帶你了解 Laravel 的基本知識，並告訴你為什麼值得你花時間學習。

**第 1 步：設置你的環境** 在開始使用 Laravel 之前，你需要一些工具。以下是你需要的：  
- **PHP**：版本 8.1 或更高  
(Laravel 發展迅速，所以保持更新！)。  
- **Composer**：這是 PHP 的依賴管理工具。從 [getcomposer.org](https://getcomposer.org) 下載。  
- **本地伺服器**：像 XAMPP、WAMP 或 Laravel 的內建伺服器都很棒。  
- **終端機**：你將運行命令，所以熟悉你的命令行。

安裝這些工具後，打開你的終端機並全域安裝 Laravel：

```
composer global require laravel/installer
```

這樣你就可以用一個命令創建新的 Laravel 專案。

**第 2 步：創建你的第一個 Laravel 專案** 準備好創建一些東西了嗎？在你的終端機中，導航到你想要專案所在的資料夾（例如 `cd /path/to/your/folder`），然後輸入：

```
laravel new my-first-app
```

幾分鐘後，Composer 將設置一個新的 Laravel 專案 `my-first-app`。導航到它：

```
cd my-first-app
```

要看到它的效果，啟動 Laravel 的內建伺服器：

```
php artisan serve
```

打開你的瀏覽器並前往 `http://localhost:8000`。啟動—你會看到一個歡迎頁面！這是 Laravel 在向你問好。

**第 3 步：了解基本知識** Laravel 遵循 MVC（模型-視圖-控制器）結構，這使你的程式碼保持整潔和有組織：  
- **模型**：處理你的數據（例如數據庫表）。  
- **視圖**：用戶看到的前端內容（HTML、CSS 等）。  
- **控制器**：連接模型和視圖的黏合劑。

你可以在 `app/` 資料夾中找到它們。例如，控制器位於 `app/Http/Controllers`，視圖位於 `resources/views`。

**第 4 步：建立一個簡單的頁面** 讓我們創建一個快速的「Hello, World」頁面。打開 `routes/web.php`—這是你定義應用程式 URL 的地方。添加這一行：

```
Route::get('/hello', function () {
    return view('hello');
});
```

現在，在 resources/views 中創建一個名為 hello.blade.php 的文件。添加這些內容：

```
<!DOCTYPE html>
<html>
<head>
    <title>Hello, Laravel</title>
</head>
<body>
    <h1>Hello, World!</h1>
</body>
</html>
```

重新啟動你的伺服器（或保持運行），然後訪問 `http://localhost:8000/hello`。你會看到你的「Hello, World！」頁面。.blade.php 文件擴展名意味著你正在使用 Laravel 的 Blade 模板引擎—非常方便用於動態內容。

**第 5 步：操作數據庫** Laravel 使用其 Eloquent ORM（對象關聯映射）使數據庫操作變得輕而易舉。首先，在 .env 文件中設置你的數據庫（例如 MySQL、SQLite）：

```
DB_CONNECTION=mysql
DB_HOST=127.0.0.1
DB_PORT=3306
DB_DATABASE=your_database_name
DB_USERNAME=your_username
DB_PASSWORD=your_password
```

創建一個遷移來設置一個表。運行：

```
php artisan make:migration create_posts_table
```

在 database/migrations 下的新文件中定義你的表：

```
public function up()
{
    Schema::create('posts', function (Blueprint $table) {
        $table->id();
        $table->string('title');
        $table->text('body');
        $table->timestamps();
    });
}
```

運行遷移：

```
php artisan migrate
```

現在你有一個 posts 表！你可以創建一個 Post 模型：

```
php artisan make:model Post
```

這將你的表與你可以在程式碼中使用的模型綁定。

**第 6 步：繼續探索** Laravel 還有更多功能—身份驗證、中介軟體、路由和通過 Composer 的套件。想要用戶登錄？運行：

```
php artisan make:auth
```

需要前端提升？使用 Laravel Breeze 或 Jetstream。官方文檔 是你最好的朋友。

**為什麼使用 Laravel？** 它快速、安全，並且節省了你重新發明輪子的時間。無論你是建立部落格、電子商務網站還是 API，Laravel 的工具—如 artisan 命令、Blade 模板和 Eloquent—都能讓生活變得更輕鬆。

那麼，你還在等什麼？從小處著手，實驗，很快你就會像專業人士一樣創建網頁應用程式。快樂編碼！