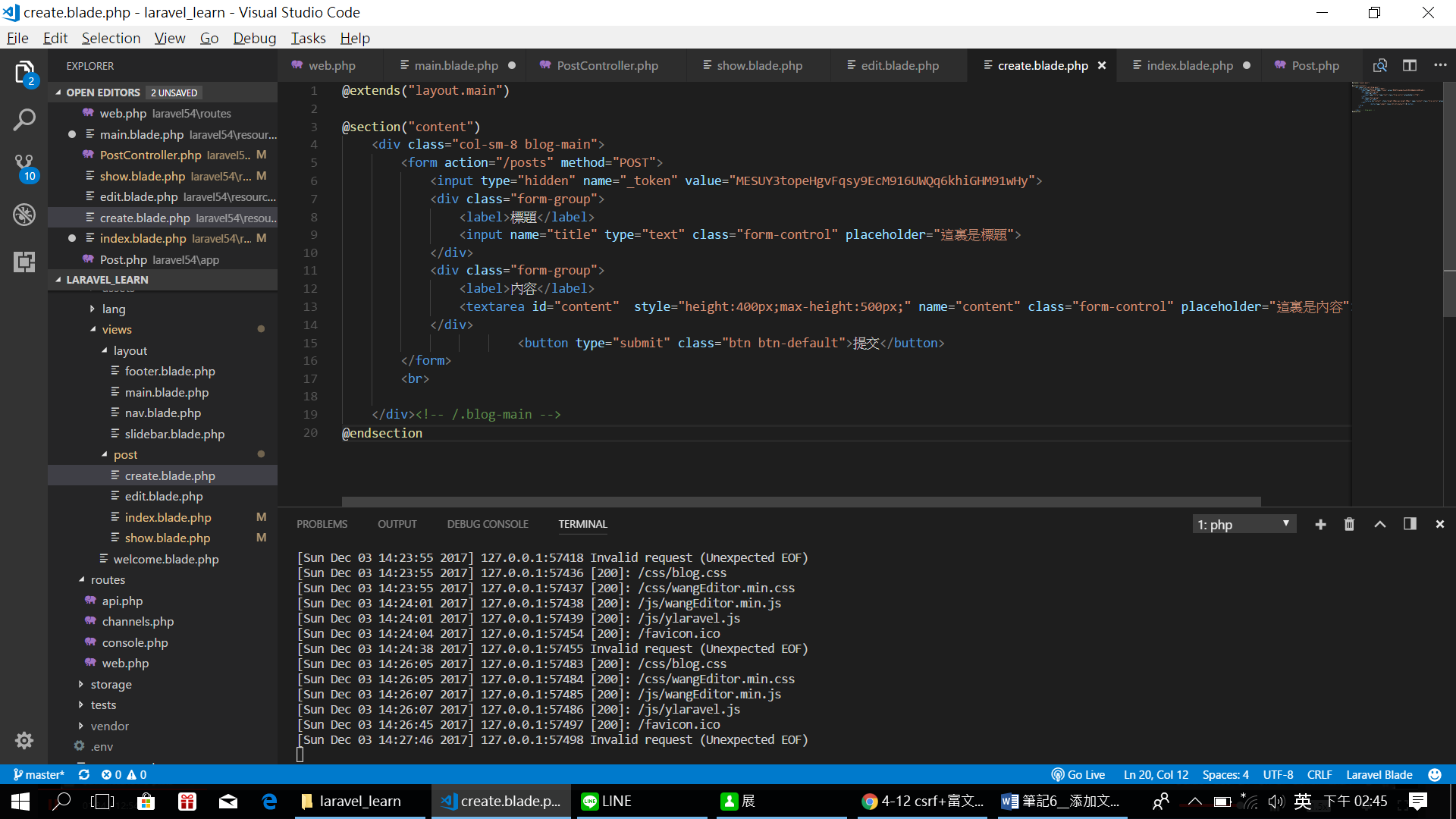


1. 創建文章頁面就是個模板:
2. 

可以看到就是個form，這個表單送出就是走post方法，對應到routes的

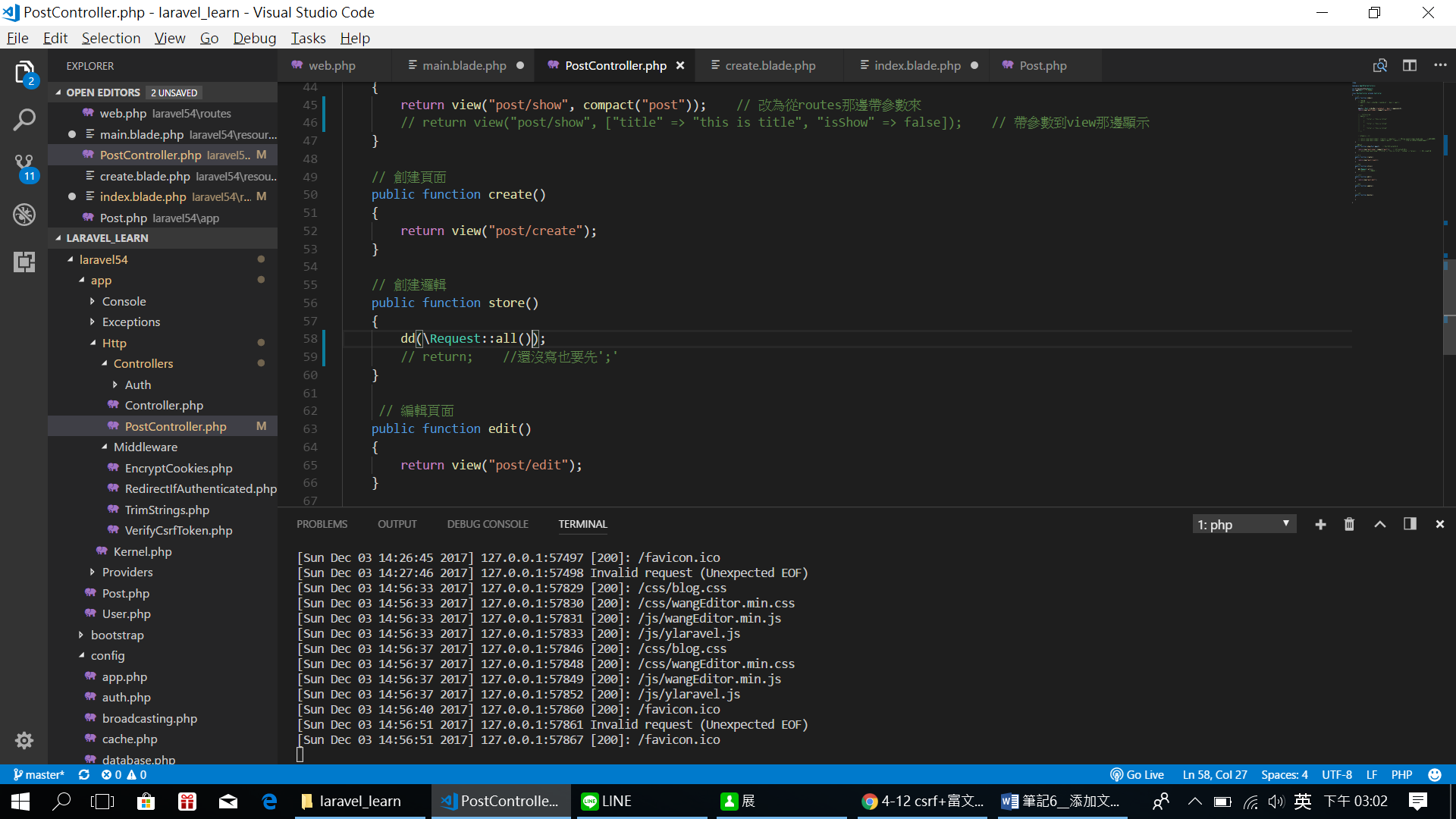
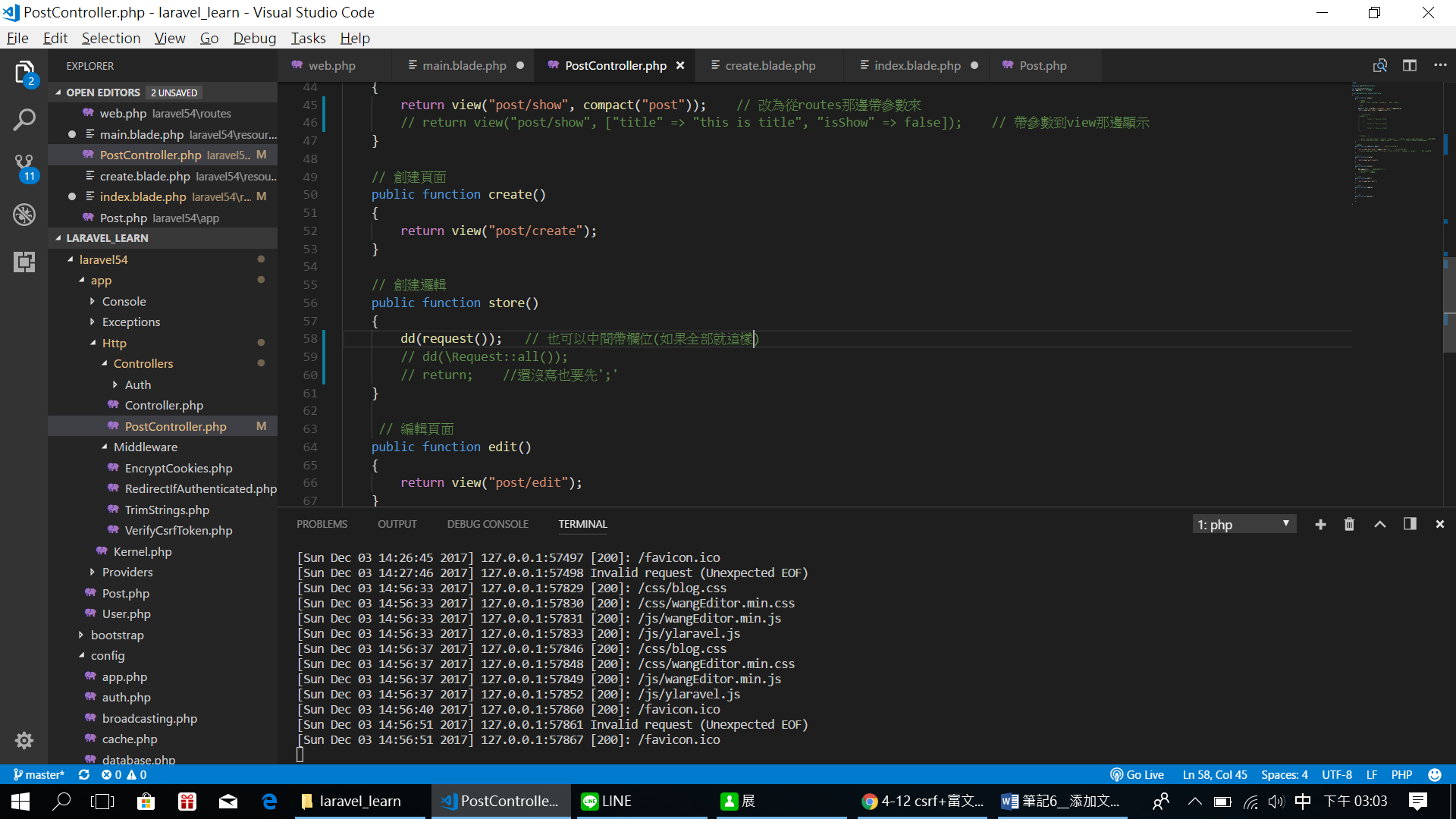
Route::post('/posts', '\App\Http\Controllers\PostController@store'); // 方法和最上面不衝突，因為方法不同，這是創建頁面點提交時的具體行為

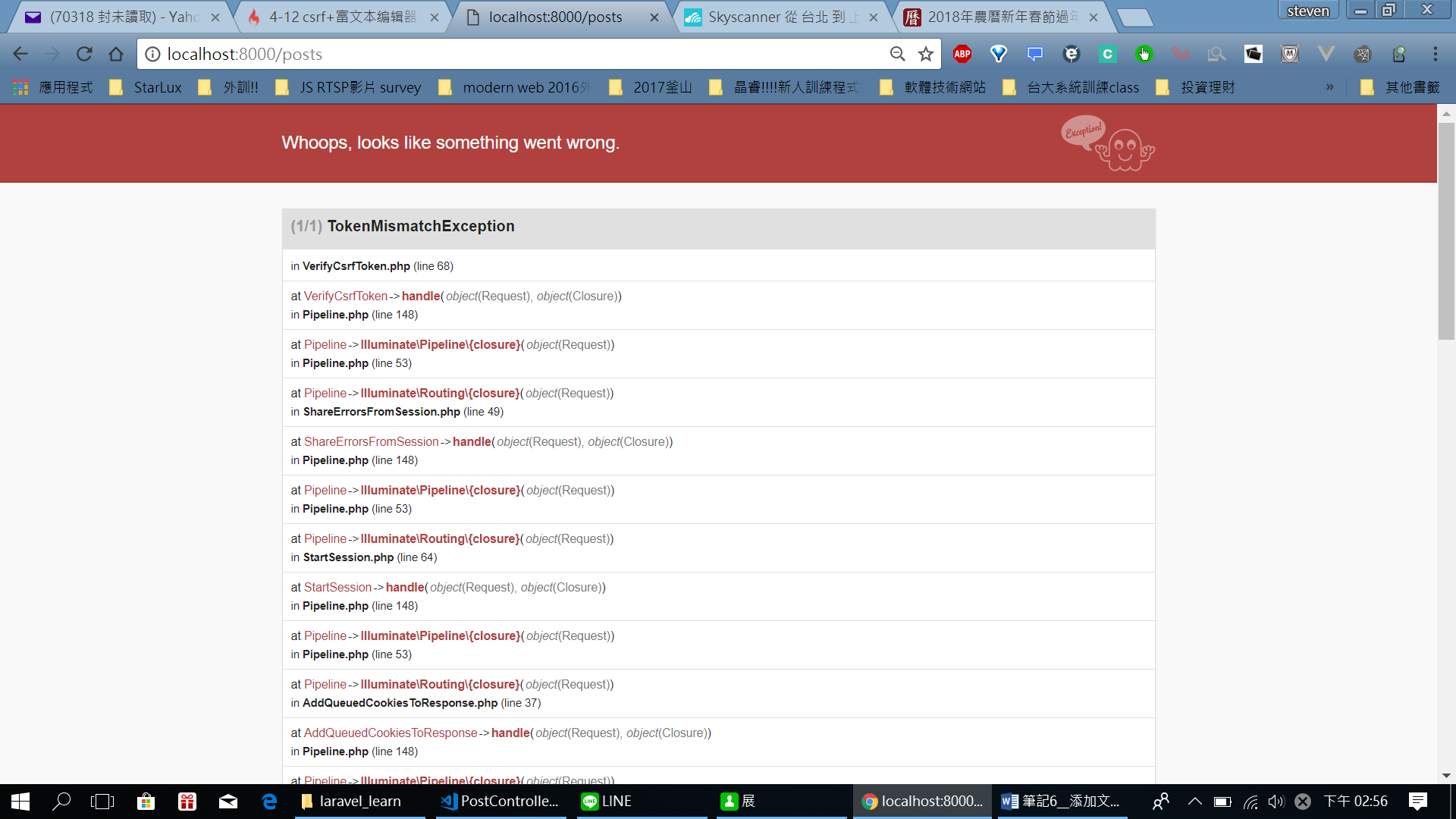
所以看得出來

送出表單會走到store

所以我們在store使laravel自帶的dd來顯示(不建議用php內建的var\_dump)

因為dd有經過格式化!



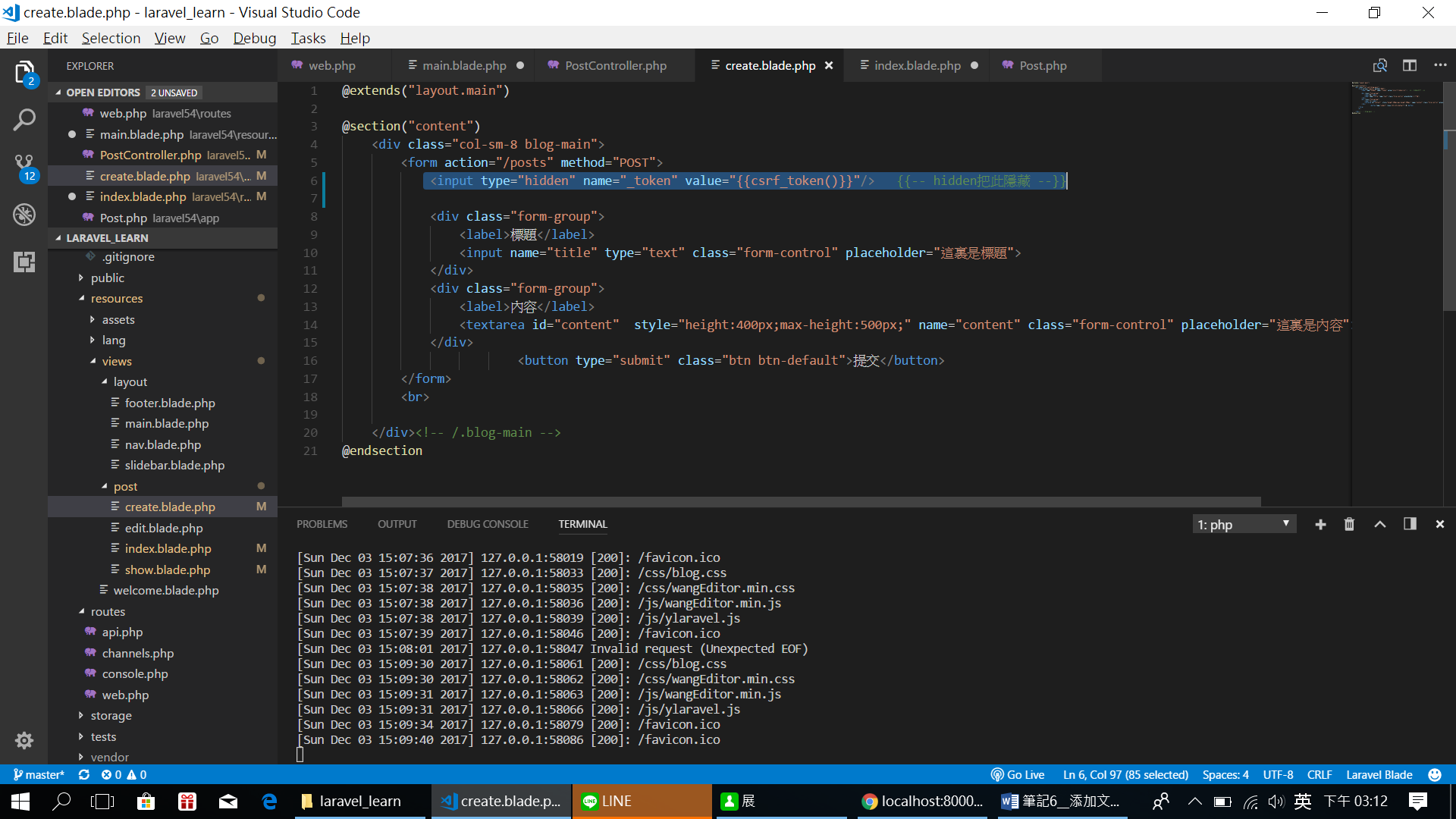
1. 結果發現

***送出表單發現>>>>>>>>> TokenMismatchException:***

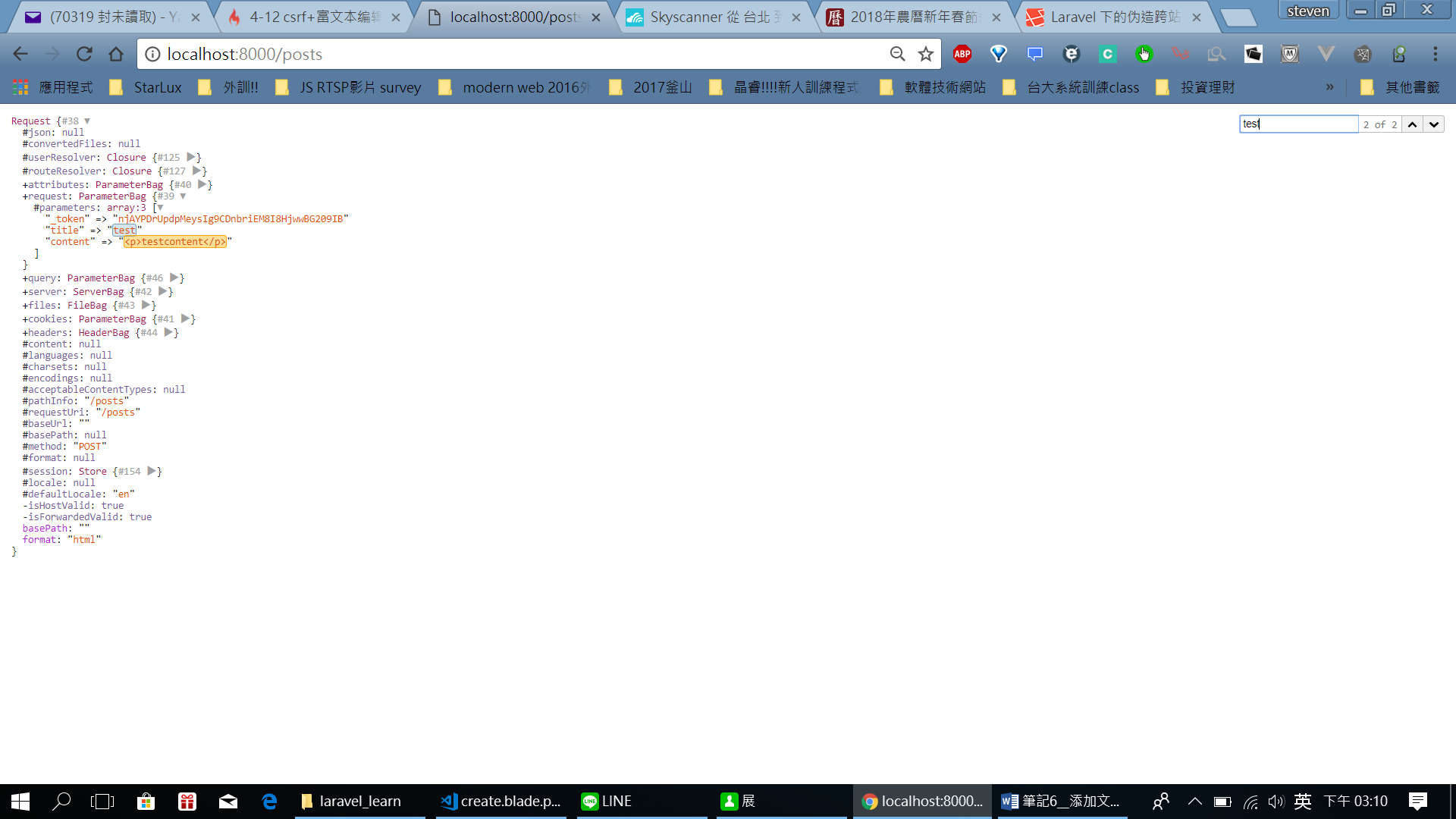
***這個就是防止CSRF跨站式請求攻擊!!!!!***

這邊可以這樣改:

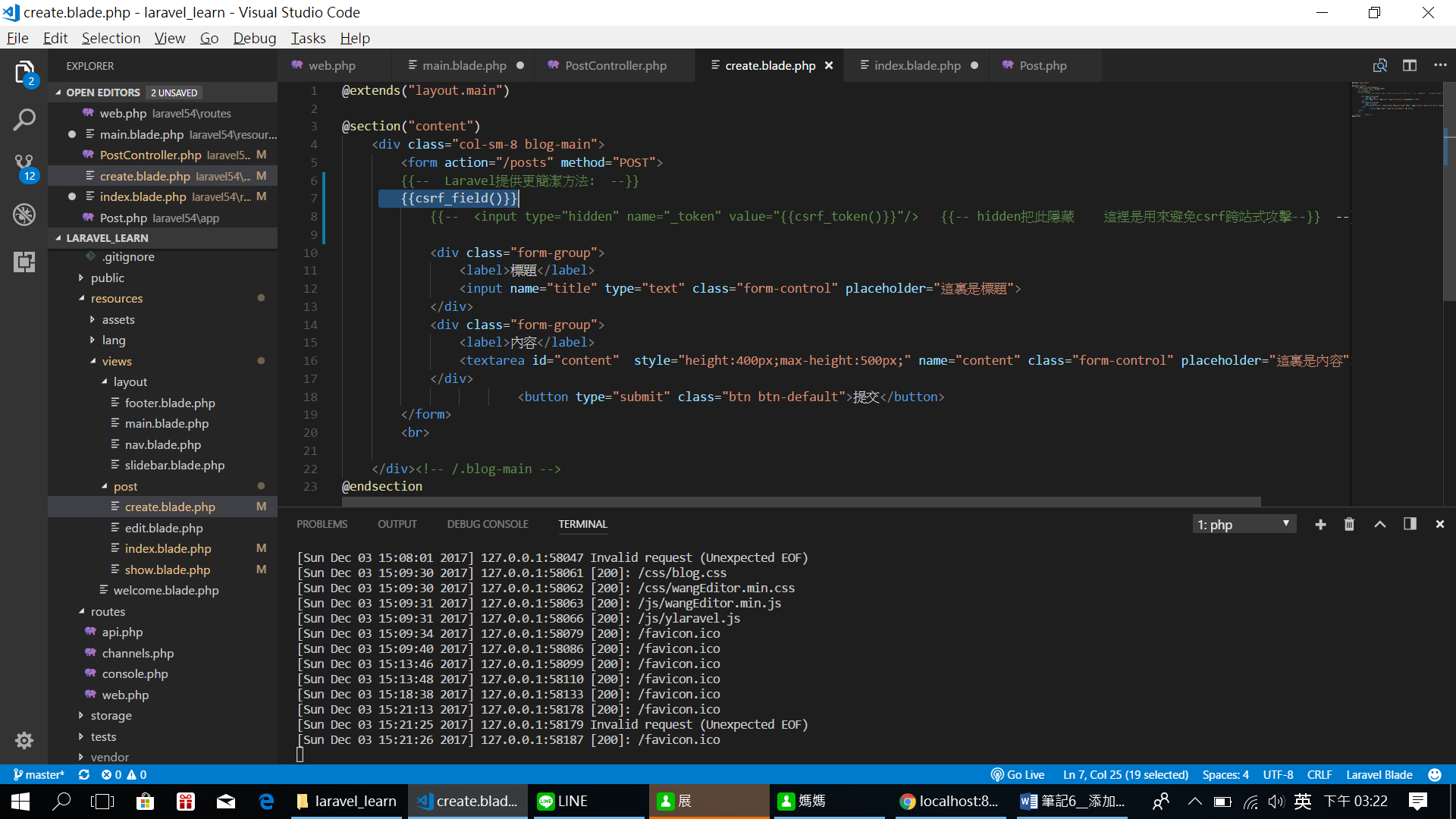
在form表單新增:



就成功看到request了



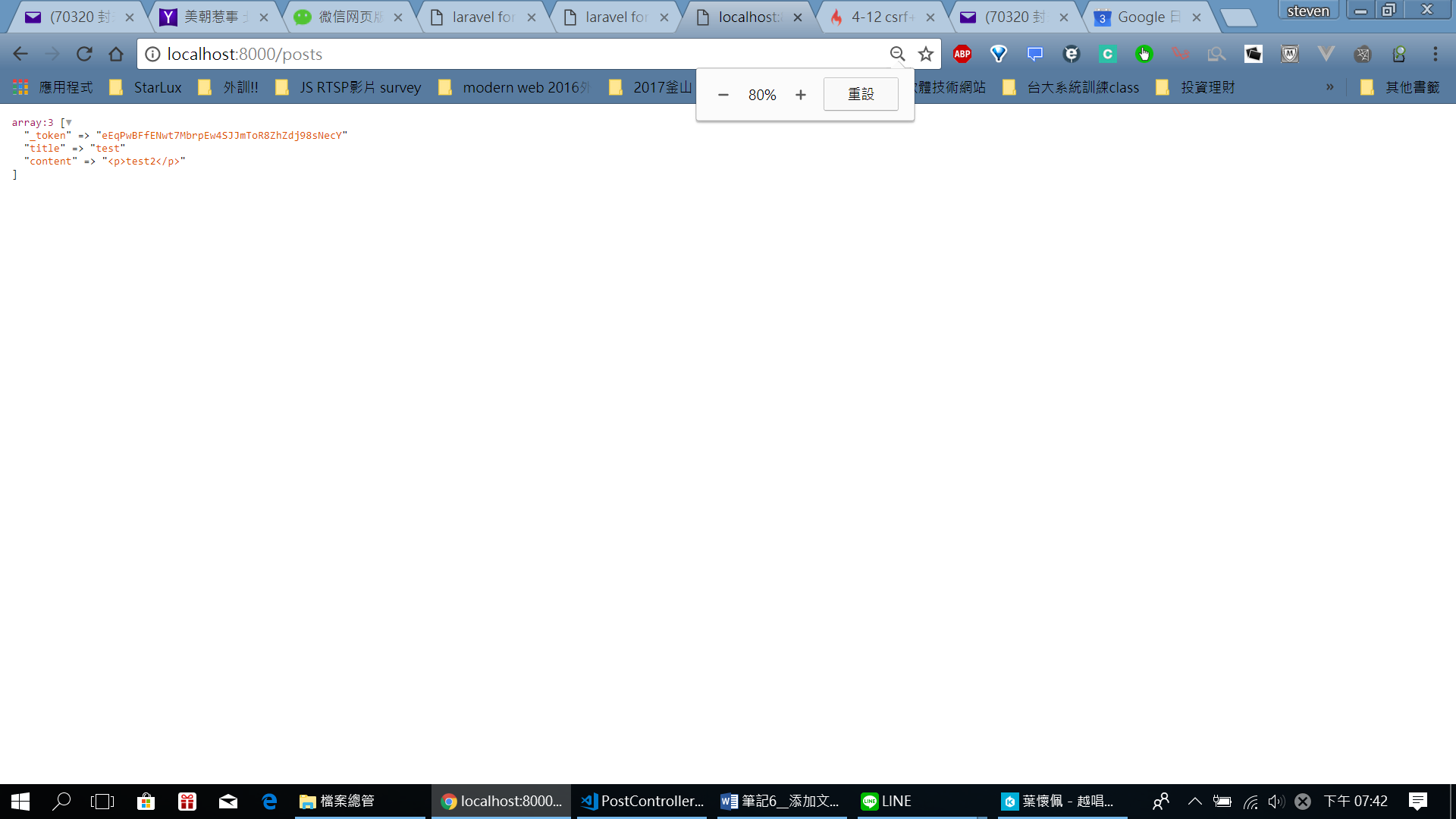
而 其實Laravel提供更簡單的方法:



只要輸入{{csrf\_field()}}即可

如果剛剛改為request()->all():

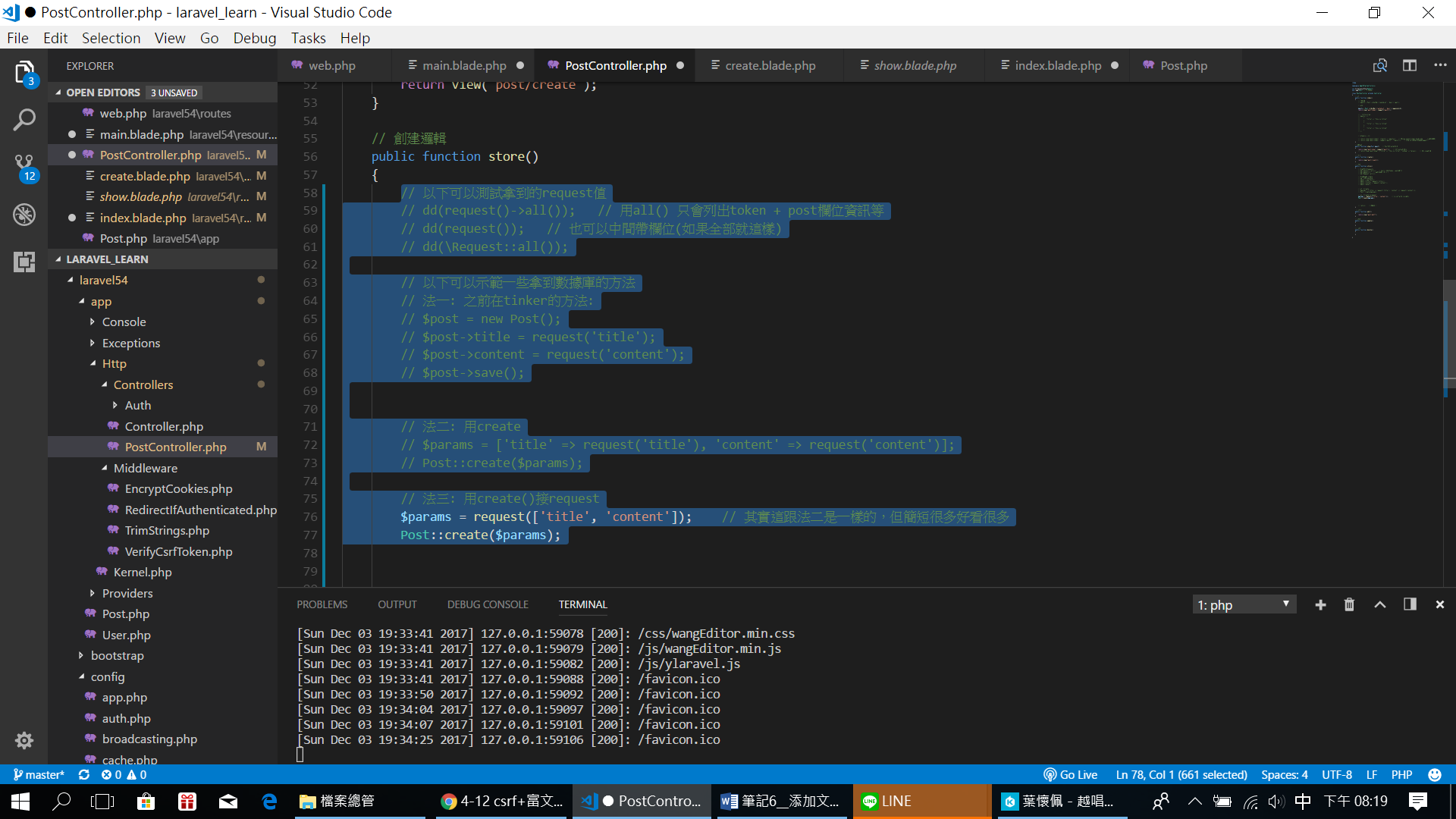
會顯示token + post那些欄位等而已:



>>>>>>>> save到DB 繼續!!!

1. 保存到數據庫

法一: 使用前面的tinker的方式:

// 以下可以測試拿到的request值

// dd(request()->all()); // 用all() 只會列出token + post欄位資訊等

// dd(request()); // 也可以中間帶欄位(如果全部就這樣)

// dd(\Request::all());

// 以下可以示範一些拿到數據庫的方法

// 法一: 之前在tinker的方法:

// $post = new Post();

// $post->title = request('title');

// $post->content = request('content');

// $post->save();

// 法二: 用create

// $params = ['title' => request('title'), 'content' => request('content')];

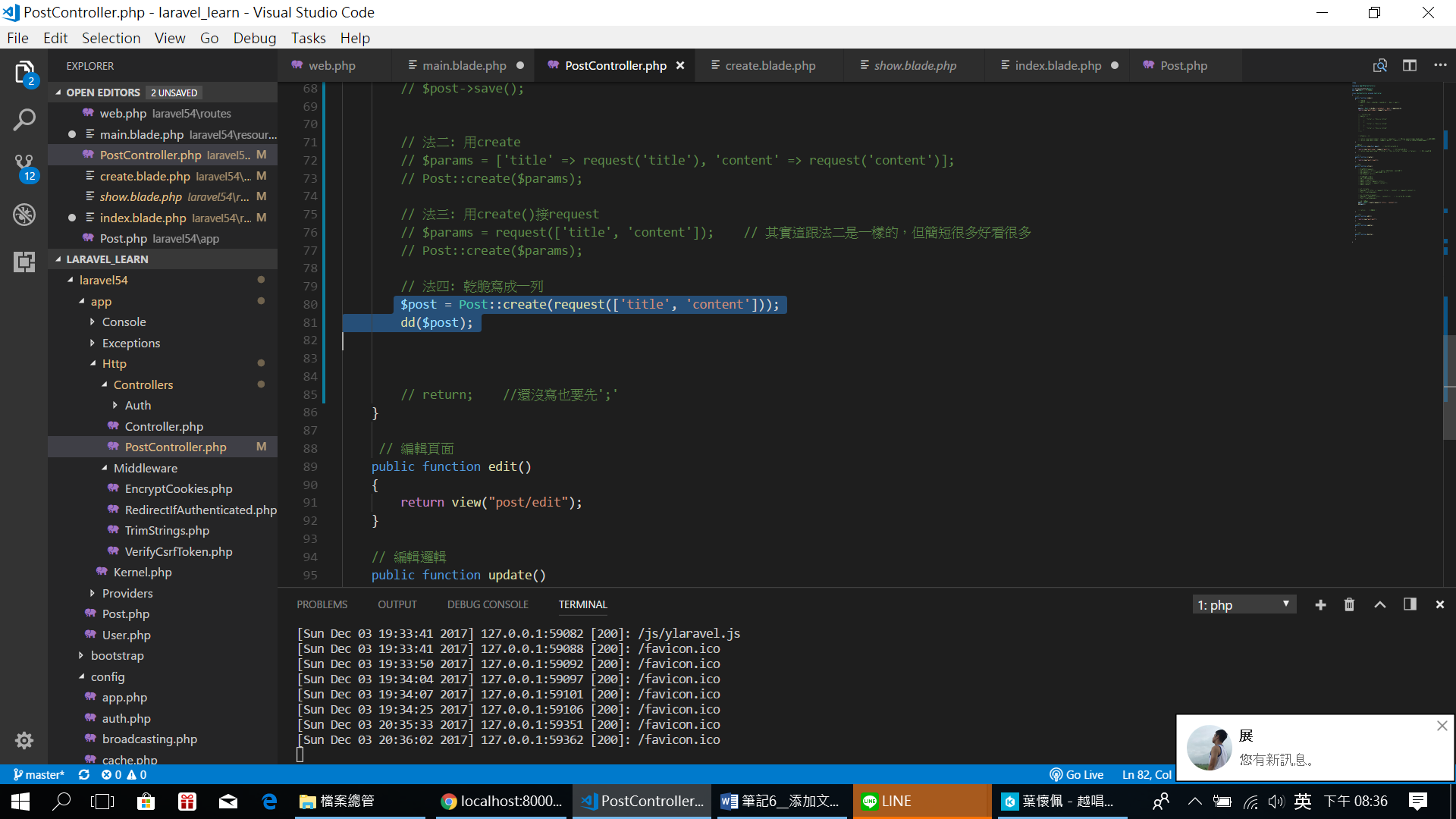
// Post::create($params);

// 法三: 用create()接request

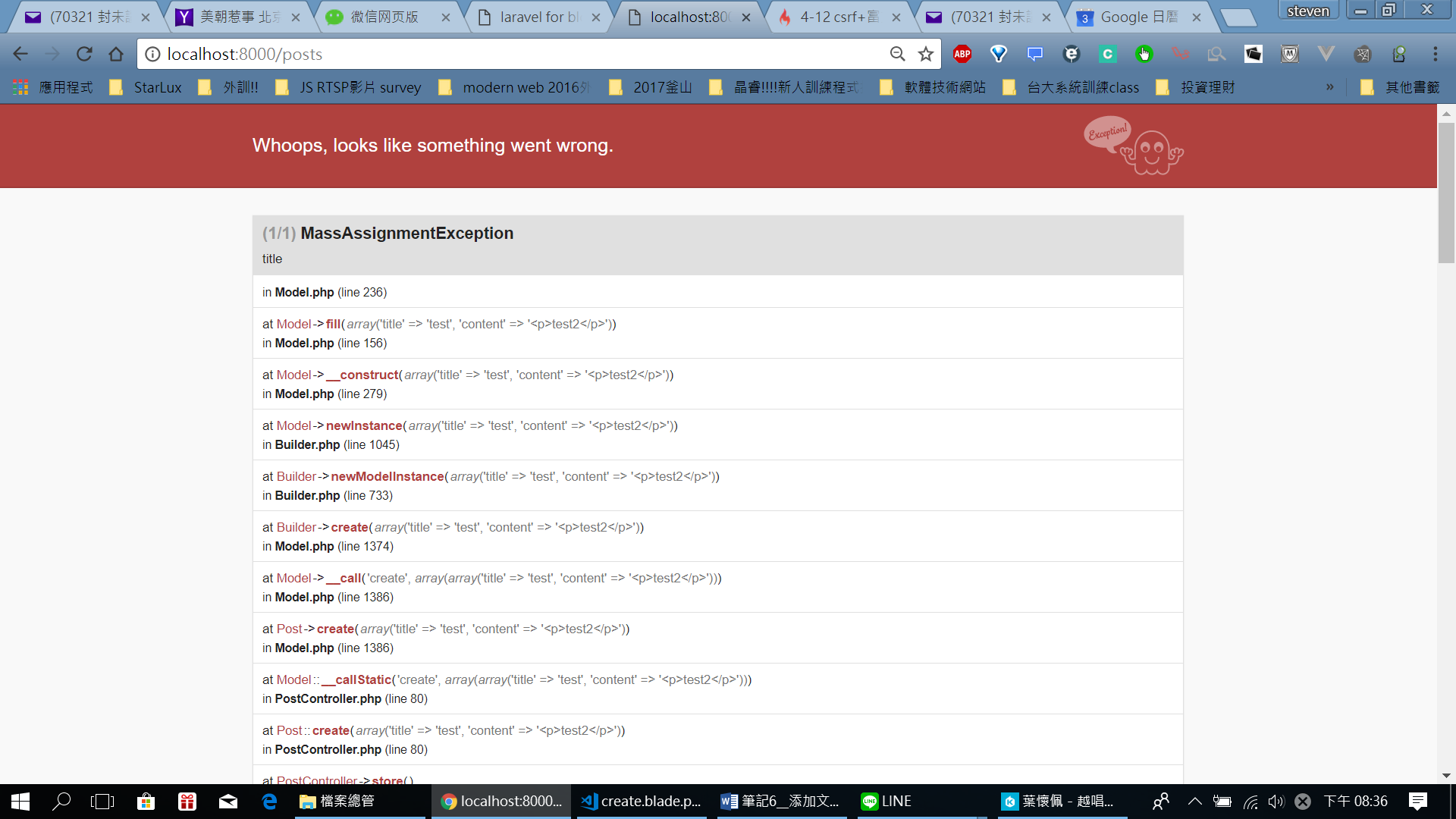
$params = request(['title', 'content']); // 其實這跟法二是一樣的，但簡短很多好看很多

Post::create($params);

再來我們試著印一下回傳值:

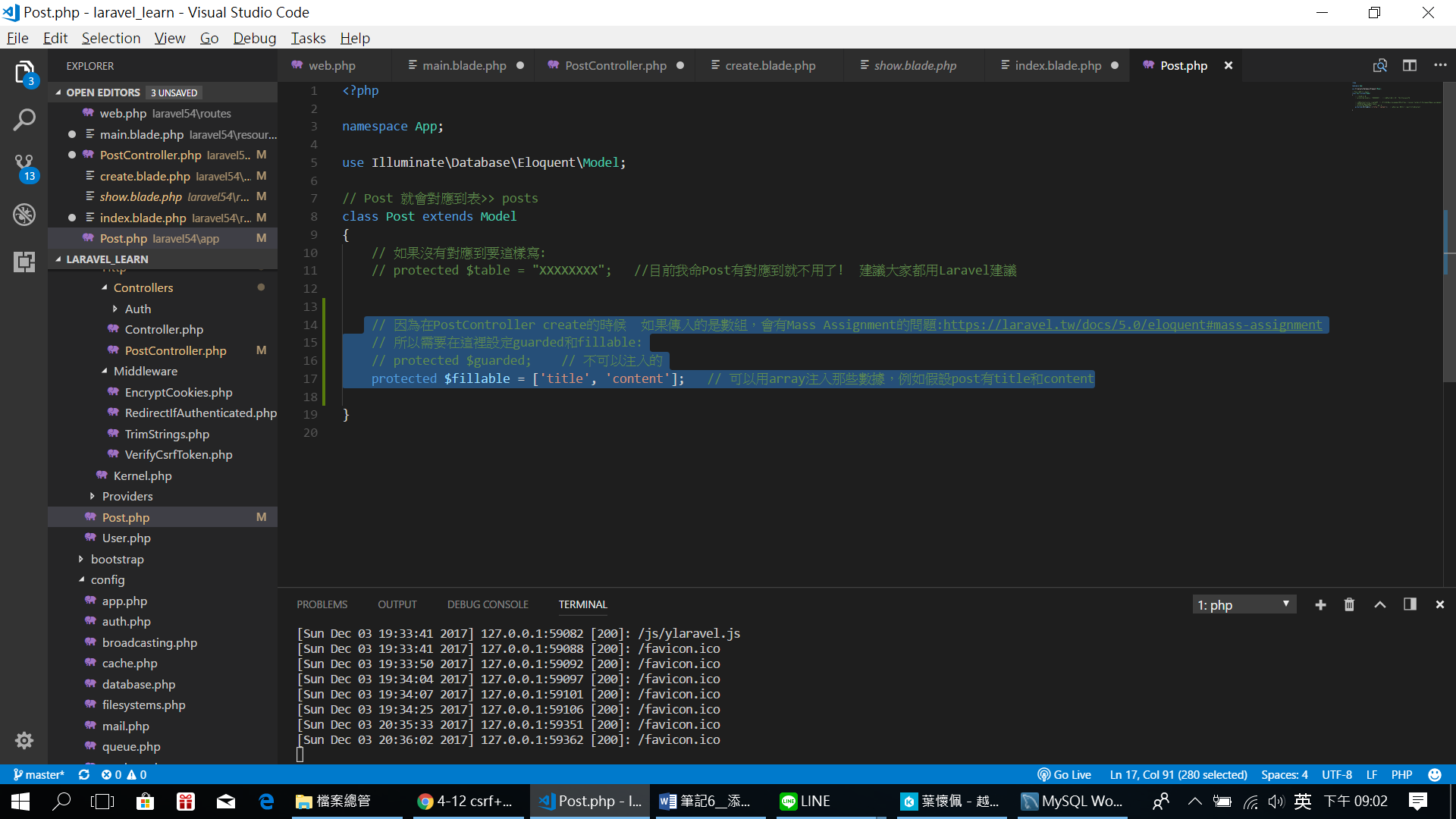


發現>>>>>>>>

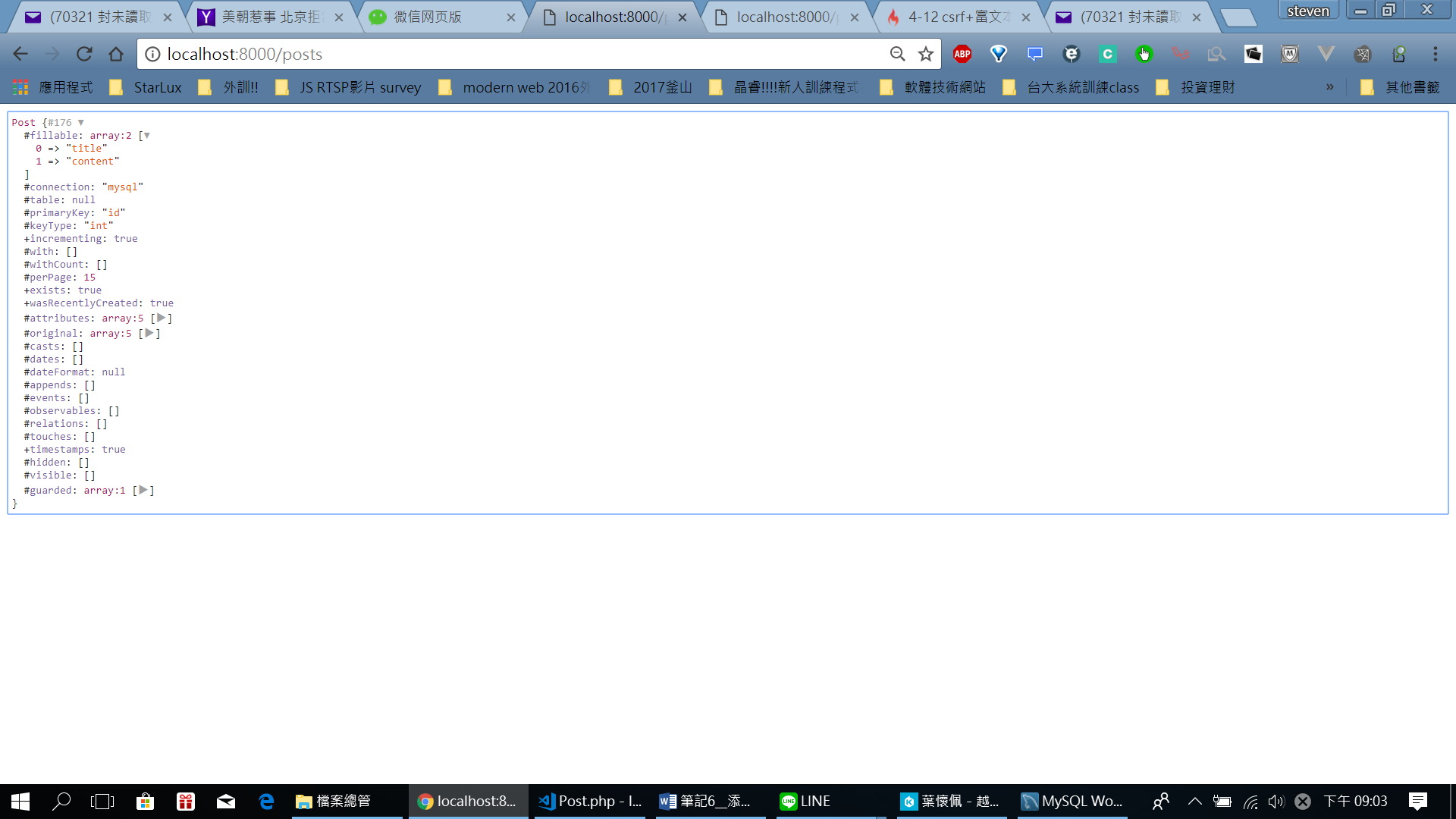


這是因為是用array進行複值的>>> <https://laravel.tw/docs/5.0/eloquent#mass-assignment>

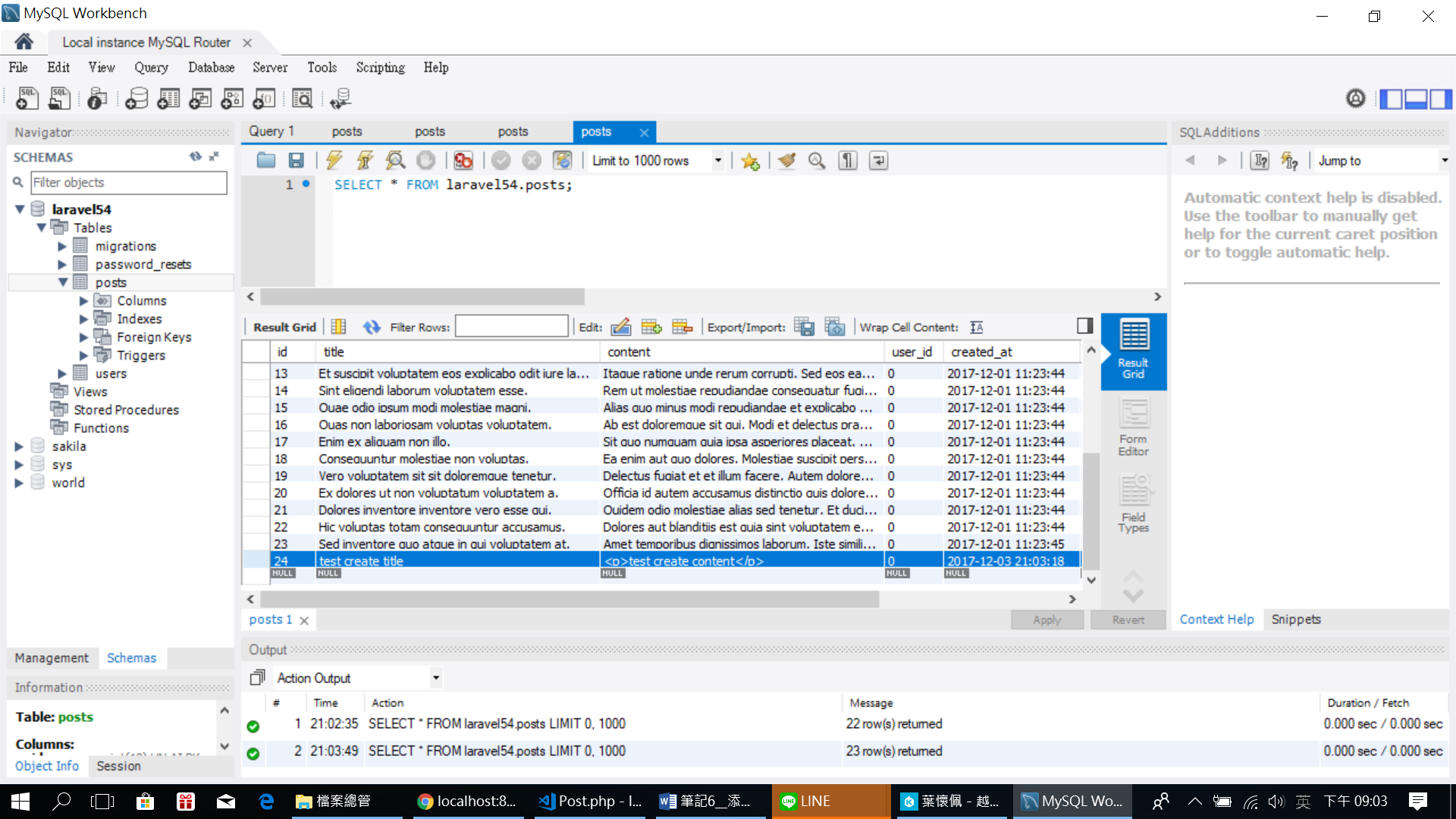
這時就需要在model 放上 protected $guarded和protected $filllable



就正常了:



成功多了一筆:



>>>> 也可以不寫$fillable 但把$guarded設為空[]

這樣等於全打開!! 也可以成功注入

***但是>>> 這樣不覺得有點麻煩嗎***

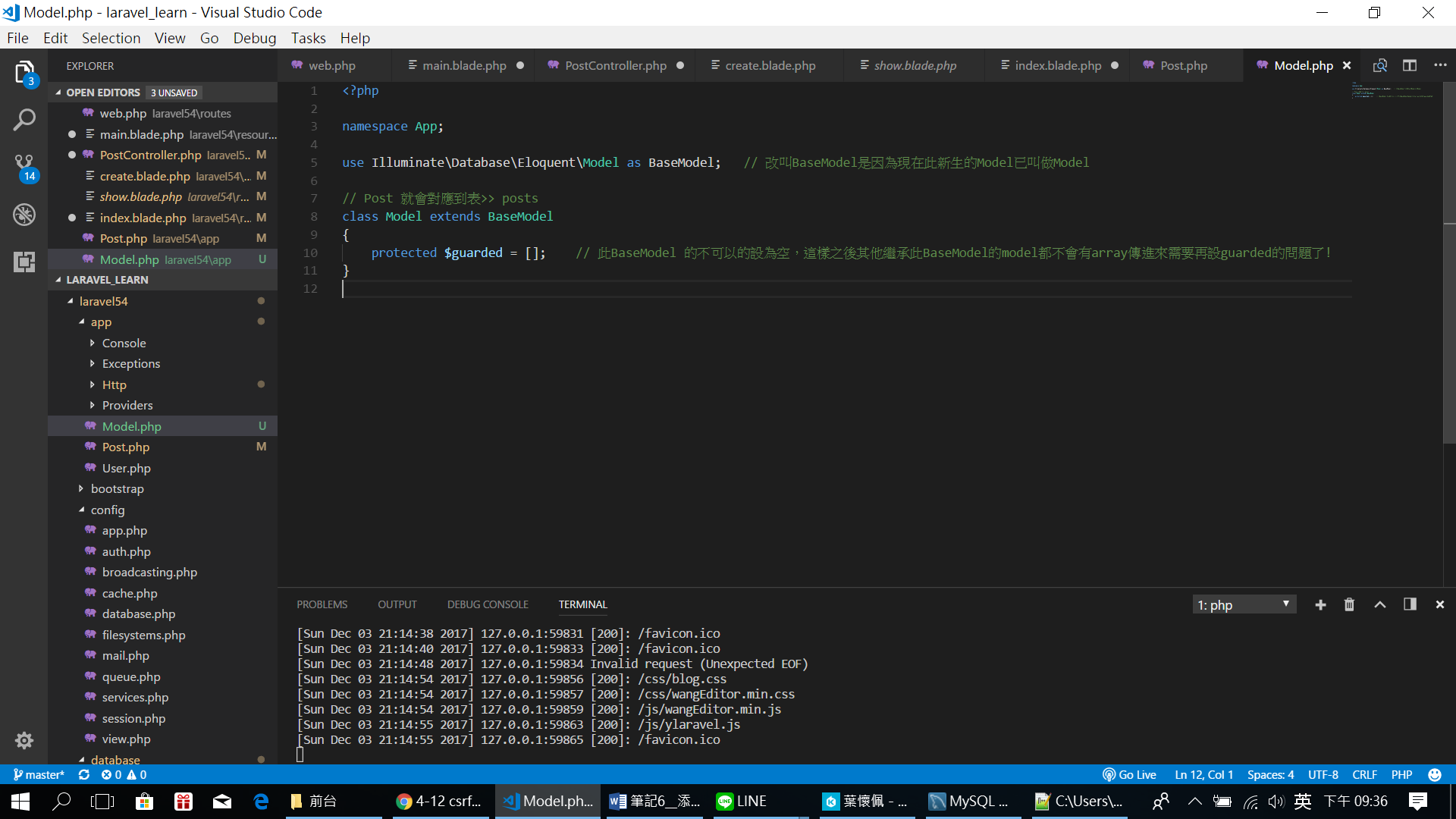
***如果每一次每一個model傳array近來都還要再來弄個$guarded = [];***

***乾脆我們抽出來一個BaseModel, 讓BaseModel都是$guarded = [];***

***之後的Model都改為繼承此model***

首先:

先生一個Model.php 作為讓其他Model繼承的model然後讓他的$guarded = []

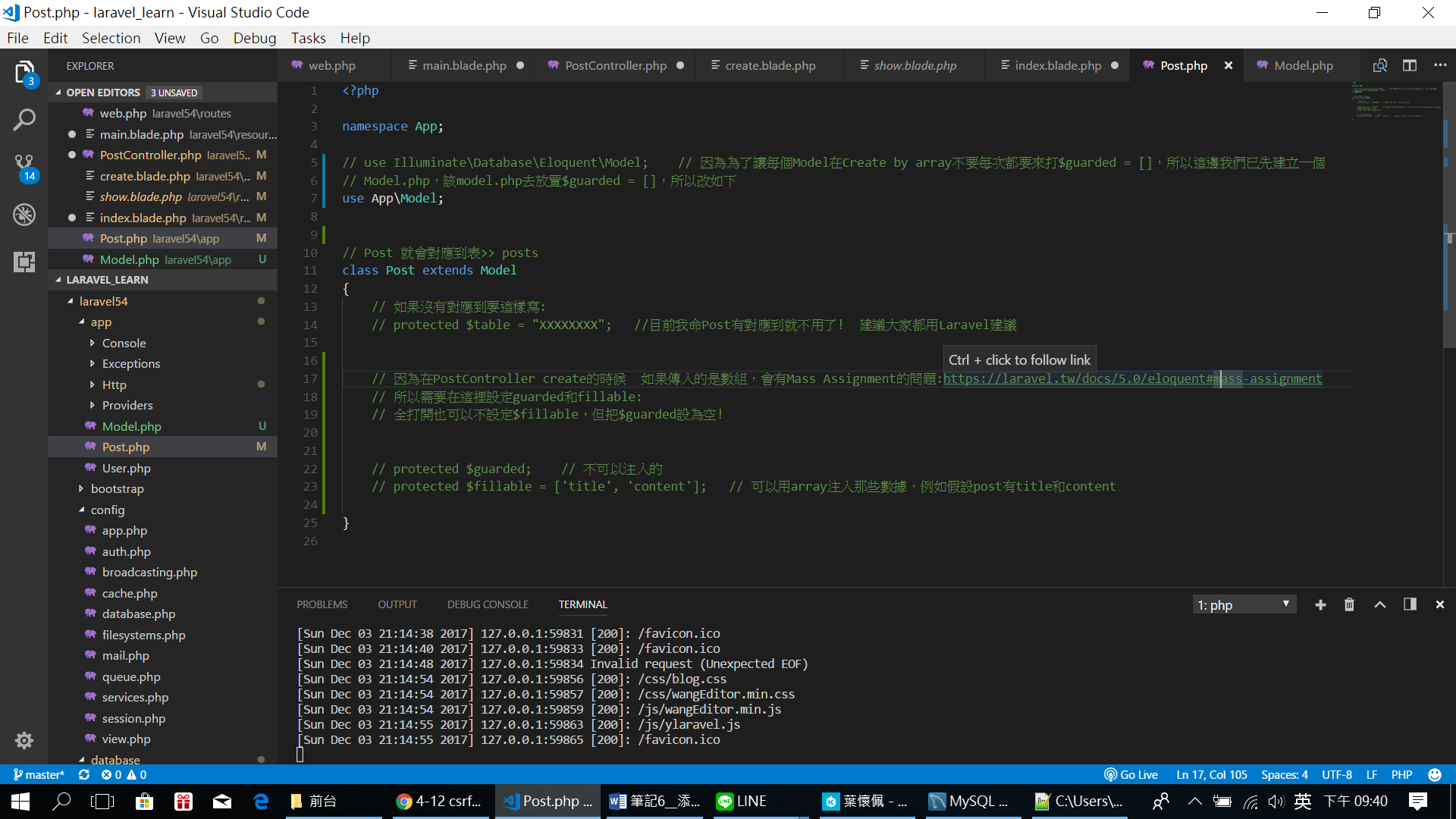


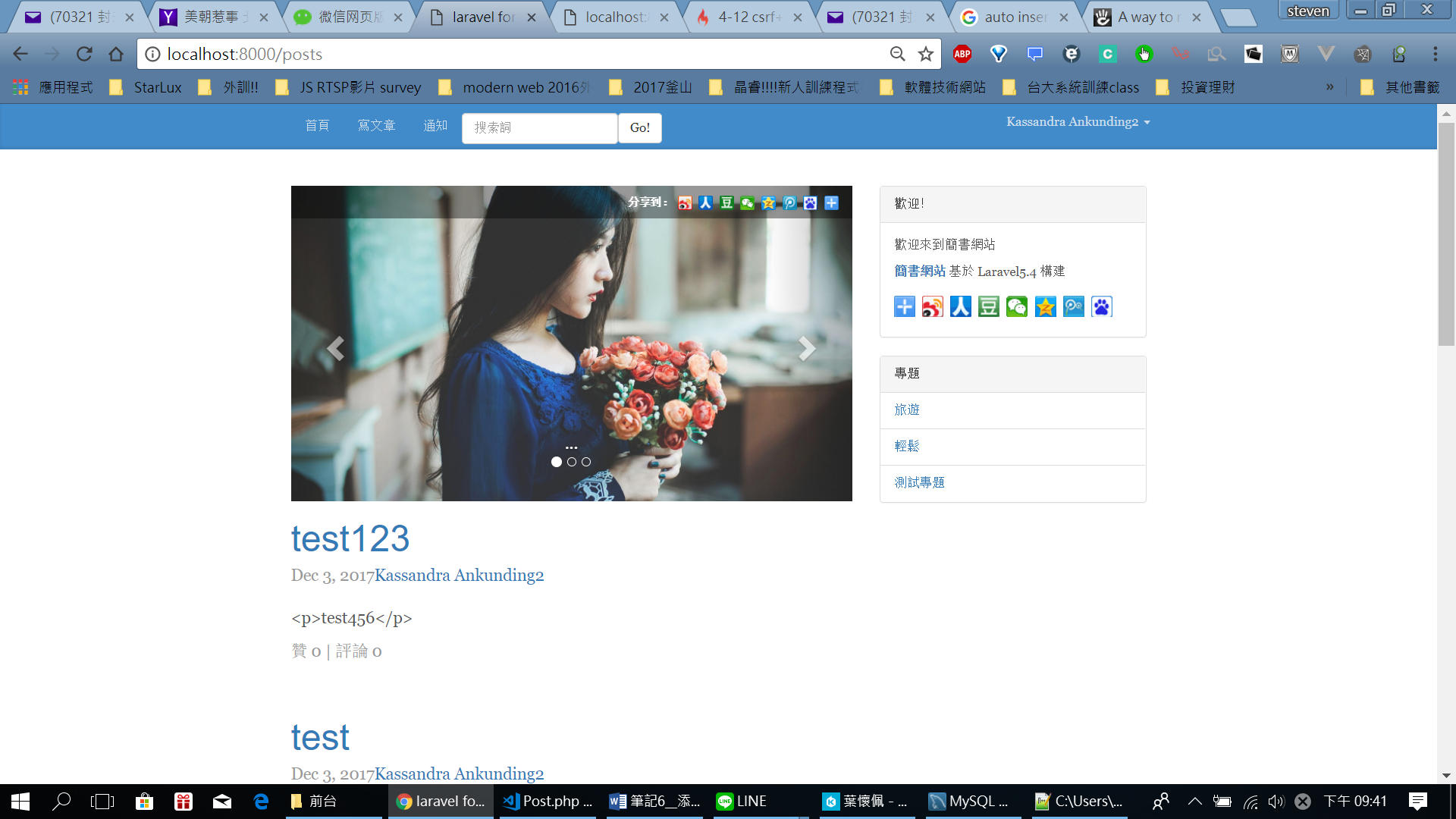
再把Post Model繼承此Model!!!!

// use Illuminate\Database\Eloquent\Model; // 因為為了讓每個Model在Create by array不要每次都要來打$guarded = []，所以這邊我們已先建立一個

// Model.php，該model.php去放置$guarded = []，所以改如下

use App\Model;



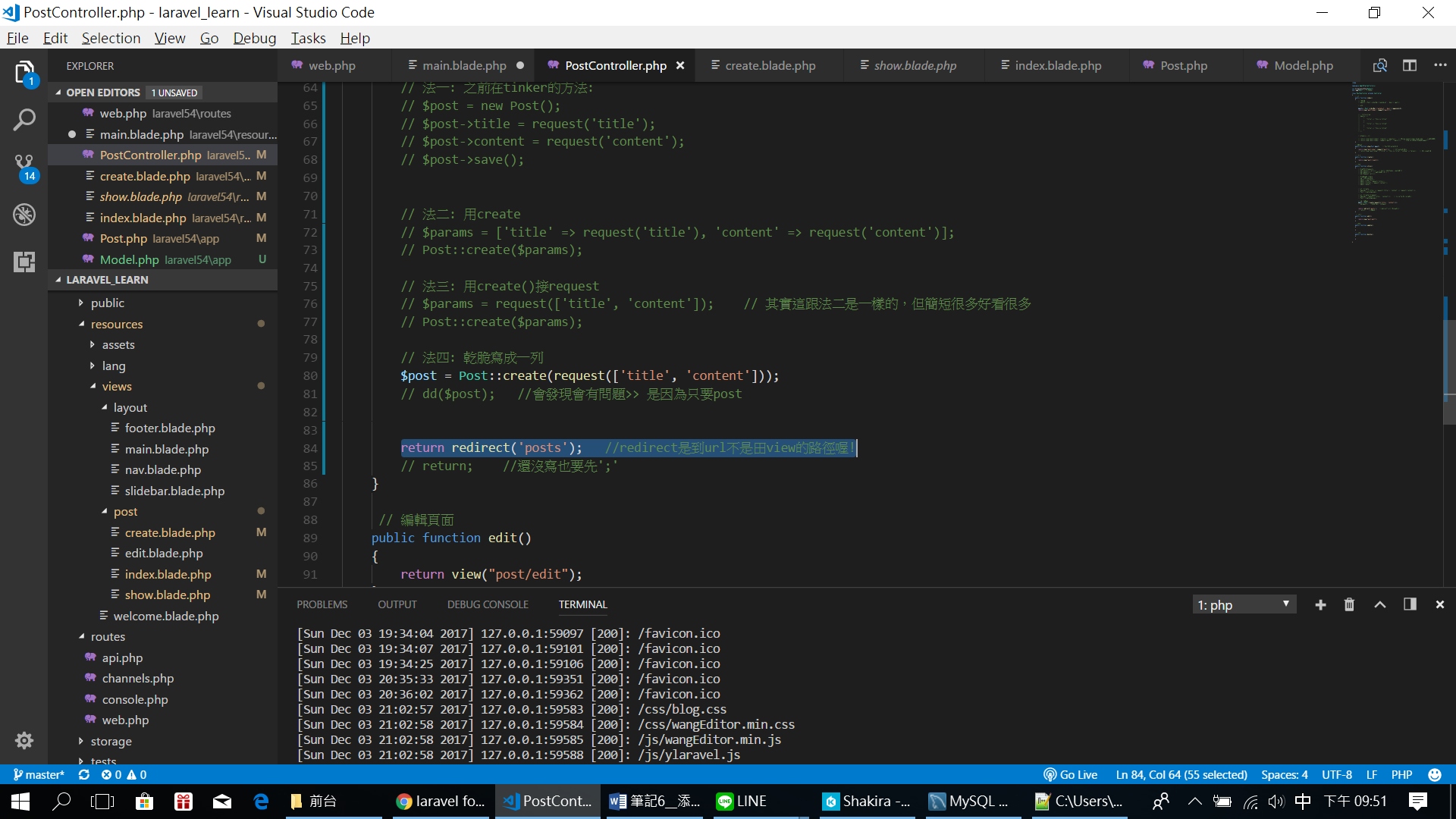


依樣成功!!!

<目前遇到content 都會自動塞一個<p> 原因我目前不明!

最後:

在把store 改為Return到index:



這樣Create完文章後就可以導回index囉!