

Separate Logic From the Template

□ مدت: حدود ۸ دقیقه و ۳۵ ثانیه

۱. مساله اصلی 🔍

تا این مرحله، منطق PHP مثل تعریف آرایه‌ها، فیلترها و توابع در همان فایل HTML قرار داشت.

مشکل اینه: کد بهم‌ریخته و سخت برای نگهداری، مخصوصاً وقتی پروژه بزرگ میشه.

۲. اهمیت «مختص کردن مسئولیت‌ها» Separation of Concerns – »

برای خوانایی بیشتر، Laracasts توصیه می‌کنه کد (PHP منطق) جدا از کد (HTML قالب نمایشی) باشه:

- `index.php` فقط مسئول آماده‌سازی داده‌ها؛
- `index.view.php` فقط وظیفه نمایش HTML داره

۳. اجرای ساختار پیشنهادی (مثال واقعی)

`index.php:`

```
<?php
$books = [
    ['name'=>'Do Androids Dream...', 'author'=>'Philip K.
    Dick', 'releaseYear'=>1968, 'purchaseUrl'=>'...'],
    ['name'=>'Project Hail Mary', 'author'=>'Andy
    Weir', 'releaseYear'=>2021, 'purchaseUrl'=>'...'],
    ['name'=>'The Martian', 'author'=>'Andy Weir', 'releaseYear'=>2011, 'purchaseUrl'=>'...'],
];

$filteredBooks = array_filter($books, function($book) {
```

```
return $book['releaseYear'] >= 1950 && $book['releaseYear'] <= 2020;
});
```

```
require "index.view.php";
```

index.view.php:

```
<!doctype html>
<html>
<head><meta charset="utf-8"><title>Demo</title></head>
<body>
<ul>
<?php foreach ($filteredBooks as $book): ?>
<li>
<a href="<?=$book['purchaseUrl'] ?>">
<?=$book['name'] ?> (<?=$book['releaseYear'] ?>) - By <?=$book['author'] ?>
</a>
</li>
<?php endforeach; ?>
</ul>
</body>
</html>
```

این ساختار رسمی Laracasts هست

۴. چرا این روش مفیده؟ ★

- خوانایی بیشتر View: فقط مسئول نمایشه، و logic فقط آماده‌سازی داده؛
- نگهداری ساده‌تر: تغییر در منطق بدون دست‌بردن تو HTML؛
- قابلیت تست و گسترش آسان‌تر: مخصوصاً برای پروژه‌های بزرگ یا استفاده از فریم‌ورک.

۵. نکات حرفه‌ای

نکته

توضیح

View

باید

‘dumb

’باشه

فایل

PHP

بدون

?>

بسته

می‌شه

Editor

Warni

ng

هیچ منطق پیچیده‌ای داخلش نباشه؛ فقط `display data`

laracasts.com+3laracasts.com+3laracasts.com+3laracasts.com+2github.com+2laracasts.com+2

خطای فضای خروجی تصادفی حذف می‌شه؛ اما اگر `view` دارید، نیاز نیست

اگر IDE مثل PHPStorm خطای “undefined variable” نشون داد، تنظیماتش رو برای `recognize include` ها فعال کن

۶. جمع‌بندی اصلی ✓

۱. منطق + داده `index.php` →

۲. نمایش `index.view.php` → `HTML` +

۳. `index.php` در نهایت `require 'index.view.php';` رو صدا می‌زنه

۴. `View` به سادگی داده فیلتر شده `$filteredBooks` رو `render` می‌کنه

این ساختار پایه‌ای برای رسیدن به معماری‌های مانند `MVC` هست.