

جلسه ۱۵ "Basic Routing"

🎯 هدف جلسه:

آشنایی با مسیریابی (Routing) در پروژه‌های PHP بدون استفاده از فریمورک‌ها.

تا الان ما مستقیماً فایل‌های `index.view.php`, `about.view.php`,

`contact.view.php` رو لود می‌کردیم.

الان می‌خوایم یک سیستم مسیریابی دستی بنویسیم که بسته به آدرس URL، مشخص کنه کدوم کنترلر (Controller) اجرا بشه.

📌 مراحل و نکات مهم جلسه:

✓ 1. استفاده از `$_SERVER['REQUEST_URI']`

در این جلسه، جفری وی نشان می‌دهد که چطور می‌توان مسیر فعلی صفحه (URL) را با استفاده از `$_SERVER['REQUEST_URI']` دریافت کرد:

```
$uri = $_SERVER['REQUEST_URI'];
```

مثلاً اگر URL این باشد:

```
/laracast-php/about
```

مقدار `$uri` خواهد شد:

```
/laracast-php/about
```

✓ 2. استفاده از ساختار شرطی برای کنترل مسیرها:

کدی شبیه زیر نوشته می‌شه:

```

if ($uri === '/laracast-php/') {
    require 'controllers/index.php';
} elseif ($uri === '/laracast-php/about') {
    require 'controllers/about.php';
} elseif ($uri === '/laracast-php/contact') {
    require 'controllers/contact.php';
} else {
    صفحه ۴۰۴ //
    require 'controllers/404.php';
}

```

! به این روش، پروژه با توجه به آدرس URL تصمیم می‌گیرد که کدام فایل PHP را اجرا کند.

3. نکته مهم درباره پوشه پروژه ⚠

در ویدیو جفری می‌گوید اگر پروژه‌تان داخل یک زیرپوشه مثل /laracast-php/ است، باید این مسیر را هم در شرط‌ها لحاظ کنید. اگر پروژه در ریشه (localhost) بود، فقط /about, /contact و ... کافی بود.

4. استفاده از کنترلرها ✓

به جای اینکه مستقیم فایل HTML را بارگذاری کند، حالا از کنترلرهایی مثل:

- controllers/index.php
- controllers/about.php
- controllers/contact.php

استفاده می‌شود، که داخل آن‌ها دستور view() برای نمایش صفحه است.

5. مدیریت خطاها (404) ✓

اگر مسیر URL هیچ کدام از موارد بالا نبود:

```
require 'controllers/404.php';
```

در اینجا می توان صفحه ای سفارشی برای خطای ۴۰۴ ساخت.
