

Laravel 1

1. Why Laravel?

- *Why Use a Framework?*
- *“I’ll Just Build It Myself”*
- *Consistency and Flexibility*
- *A Short History of Web and PHP Frameworks*
- *Ruby on Rails*
- *The Influx of PHP Frameworks*
- *The Good and the Bad of CodeIgniter*
- *Laravel 1, 2, and 3*
- *Laravel 4*
- *Laravel 5*
- *What’s So Special About Laravel?*
- *The Philosophy of Laravel*
- *How Laravel Achieves Developer Happiness*
- *The Laravel Community*
- *How It Works*
- *Why Laravel?*

2. Setting Up a Laravel Development Environment.

- *System Requirements*
- *Composer*
- *Local Development Environments*
- *Laravel Valet*
- *Laravel Homestead*
- *Creating a New Laravel Project*
- *Installing Laravel with the Laravel Installer Tool*
- *Installing Laravel with Composer's create-project Feature*
- *Lambo: Super-Powered "Laravel New"*
- *Laravel's Directory Structure*
- *The Folders*
- *The Loose Files*
- *Configuration*
- *The .env File*
- *Up and Running*
- *Testing*
- *TL;DR*

3. Routing and Controllers.

- *A Quick Introduction to MVC, the HTTP Verbs, and REST*
- *What Is MVC?*
- *The HTTP Verbs*
- *What Is REST?*
- *Route Definitions*
- *Route Verbs*
- *Route Handling*
- *Route Parameters*
- *Route Names*
- *Route Groups*
- *Middleware34*
- *Path Prefixes*
- *Fallback Routes*
- *Subdomain Routing*
- *Namespace Prefixes*
- *Name Prefixes*
- *Signed Routes*
- *Signing a Route*
- *Modifying Routes to Allow Signed Links*
- *Views*

- *Returning Simple Routes Directly with Route::view()*
- *Using View Composers to Share Variables with Every View*
- *Controllers*
- *Getting User Input*
- *Injecting Dependencies into Controllers*
- *Resource Controllers*
- *API Resource Controllers*
- *Single Action Controllers*
- *Route Model Binding*
- *Implicit Route Model Binding*
- *Custom Route Model Binding*
- *Route Caching*
- *Form Method Spoofing*
- *HTTP Verbs in Laravel*
- *HTTP Method Spoofing in HTML Forms*
- *CSRF Protection*
- *Redirects*
- *redirect()->to()*
- *redirect()->route()*
- *redirect()->back()*

- *Other Redirect Methods*
- *redirect()->with()*
- *Aborting the Request*
- *Custom Responses*
- *response()->make()*
- *response()->json() and ->jsonp()*
- *response()->download(), ->streamDownload(), and ->file()*
- *Testing*
- *TL;DR*

4. Blade Templating

- *Echoing Data*
- *Control Structures*
- *Conditionals*
- *Loops*
- *Template Inheritance*
- *Defining Sections with @section/@show and @yield*
- *Including View Partial*
- *Using Stacks*
- *Using Components and Slots*

- *View Composers and Service Injection*
- *Binding Data to Views Using View Composers*
- *Blade Service Injection*
- *Custom Blade Directives*
- *Parameters in Custom Blade Directives*
- *Example: Using Custom Blade Directives for a Multitenant App*
- *Easier Custom Directives for “if ” Statements*
- *Testing*
- *TL;DR*

5. Databases and Eloquent.

- *Configuration*
- *Database Connections*
- *Other Database Configuration Options*
- *Defining Migrations*
- *Running Migrations*
- *Seeding*
- *Creating a Seeder*
- *Model Factories*
- *Query Builder*

- *Basic Usage of the DB Facade*
- *Raw SQL*
- *Chaining with the Query Builder*
- *Transactions*
- *Introduction to Eloquent*
- *Creating and Defining Eloquent Models*
- *Retrieving Data with Eloquent*
- *Inserts and Updates with Eloquent*
- *Deleting with Eloquent*
- *Scopes*
- *Customizing Field Interactions with Accessors, Mutators, and Attribute*
- *Casting*
- *Eloquent Collections*
- *Eloquent Serialization*
- *Eloquent Relationships*
- *Child Records Updating Parent Record Timestamps*
- *Eloquent Events*
- *Testing*
- *TL;DR*

6. Frontend Components.

- *Laravel Mix*
- *Mix Folder Structure*
- *Running Mix*
- *What Does Mix Provide?*
- *Frontend Presets and Auth Scaffolding*
- *Frontend Presets*
- *Auth Scaffolding*
- *Pagination*
- *Paginating Database Results*
- *Manually Creating Paginators*
- *Message Bags*
- *Named Error Bags*
- *String Helpers, Pluralization, and Localization*
- *The String Helpers and Pluralization*
- *Localization*
- *Testing*
- *Testing Message and Error Bags*
- *Translation and Localization*
- *TL;DR*

