






LARAVEL FRAMEWORK

BÀI 3: BLADE TEMPLATE

- ⊙ Hiểu về Template.
- ⊙ Sử dụng được Blade



Phần I: Giới thiệu về Template và Blade

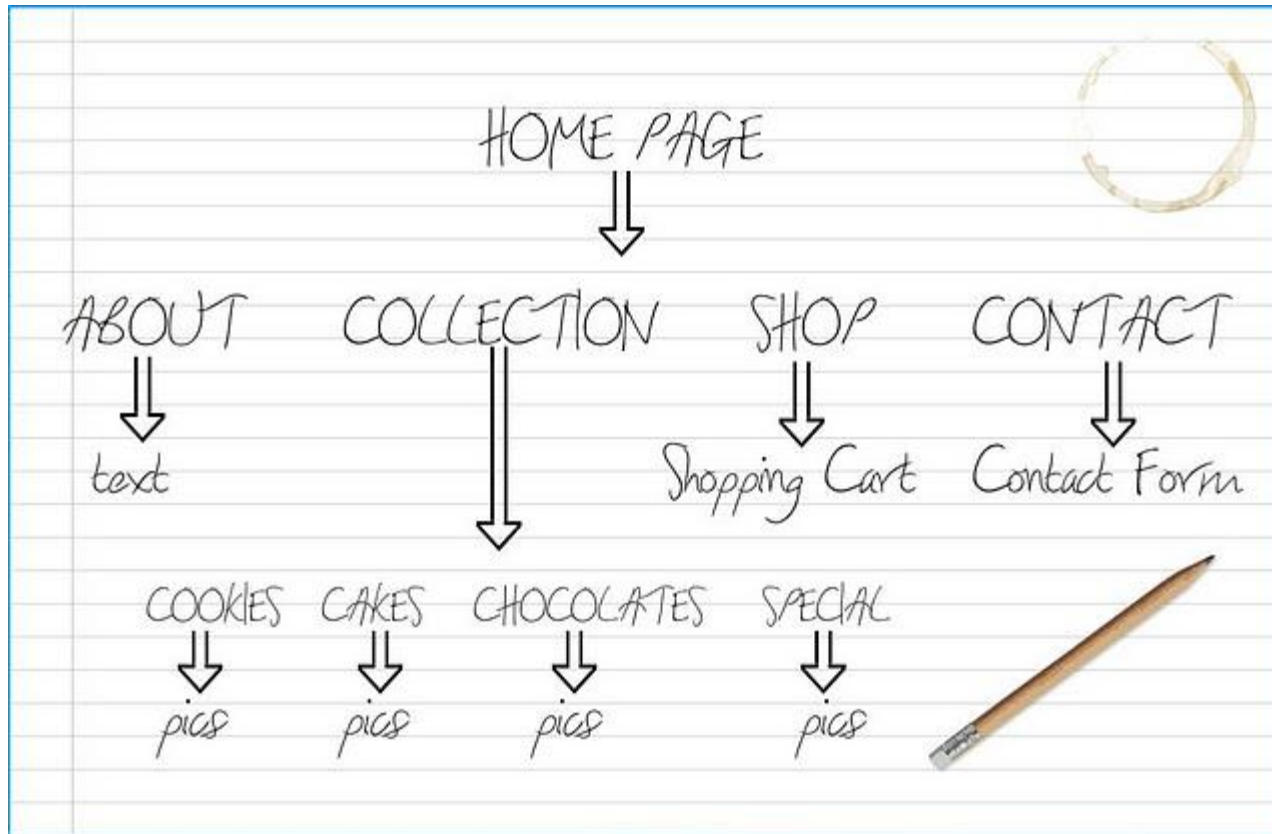
-  Template website
-  Template trong Laravel
-  Blade template

Phần II: Blade syntax

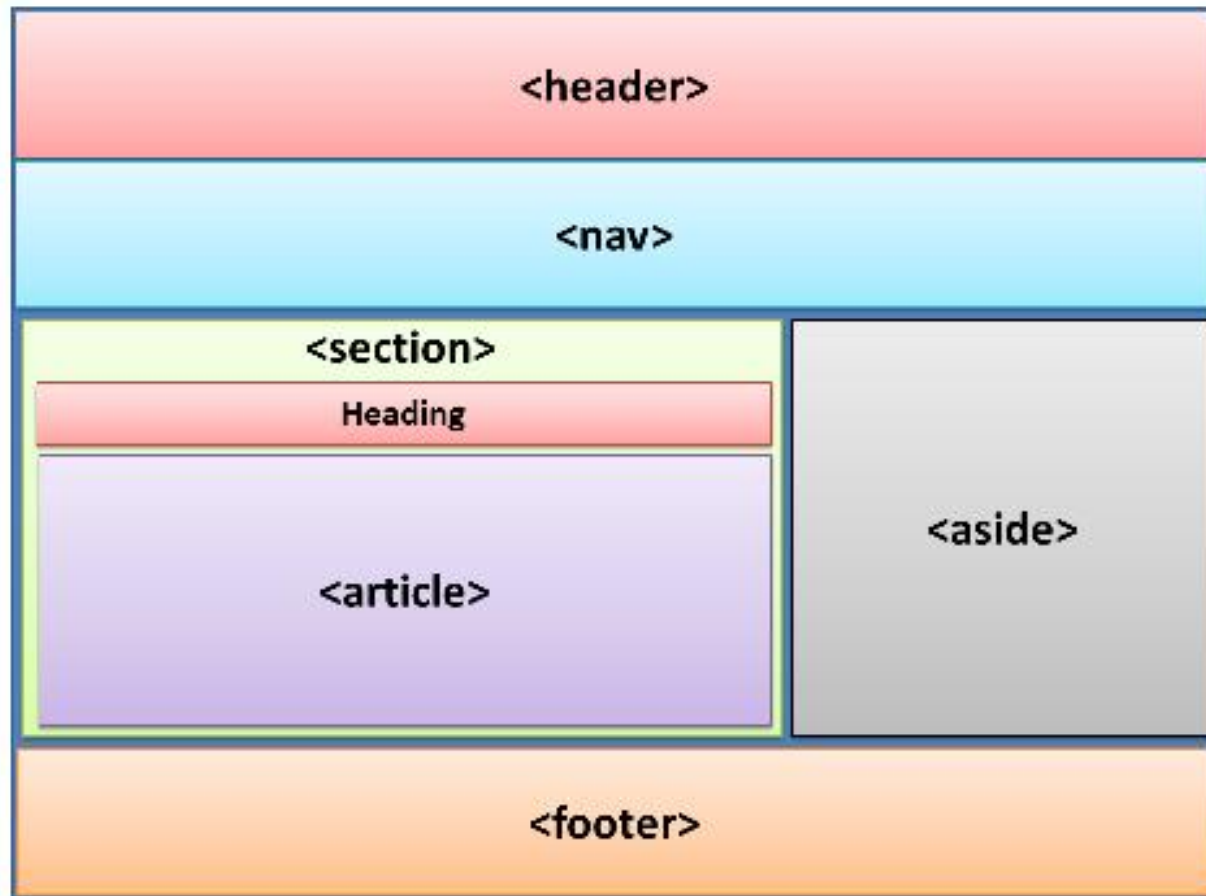
-  Control Structures
-  Executing PHP functions in blade



□ Sitemap



□ Webpage layout



- ❑ Trang web thông thường bao gồm các thành phần như header, footer, sidebar-left, sidebar-right,... các thành phần này thường cố định, không thay đổi. Chỉ có phần nội dung chính là sẽ thay đổi theo từng trang.
- ❑ Template chính là mẫu bố cục chung cho tất cả các trang có sử dụng lại những thành phần giống nhau mà không phải viết lại toàn bộ, từ đó trên mỗi trang, chỉ cần thay đổi ở một số nơi được chỉ định trên trang từ template.

- ❑ Blade là một templating engine đơn giản nhưng rất mạnh mẽ được tạo ra và đi cùng với Laravel.
- ❑ Blade Template là một templating engine giúp biên dịch những cú pháp nhất định thành code PHP tiện dụng hơn cho người dùng.
- ❑ Các file của Blade view sử dụng đuôi là `.blade.php` và cơ bản được lưu trong thư mục `resources/views`.

❑ Template inheritance

❖ Hai lợi ích chính của việc sử dụng Blade là *kế thừa template* và *sections*

❖ MASTER LAYOUT

- Tạo thư mục layout trong `/resources/views/`
- Tạo file `master.blade.php` in `/resources/views/layouts/master.blade.php`
- Thêm đoạn code Html vào `master.blade.php`

❑ Template inheritance

```
<html>
  <head>
    <title>App Name - @yield('title')</title>
  </head>
  <body>
    @section('sidebar')
      This is the master sidebar.
    @show

    <div class="container">
      @yield('content')
    </div>
  </body>
</html>
```

❖ MASTER LAYOUT

- @yield('title') hiển thị dữ liệu phần title
- @section('sidebar') tạo section với tên sidebar
- @show is hiển thị dữ liệu phần section
- @yield('content') hiển thị dữ liệu phần content

❑ Template inheritance

- ❖ Extending the master layout: tạo webpage
page.blade.php in /resources/views/page.blade.php
kế thừa master layout

```
1  @extends('layouts.master')
2
3  @section('title', 'Page Title')
4
5  @section('sidebar')
6
7
8      <p>This is appended to the master sidebar.</p>
9  @endsection
10
11 @section('content')
12     <p>This is my body content.</p>
13 @endsection
```

❑ Template inheritance

❖ Extending the master layout

`@extends('layouts.master')` extends the master layout

`@section('title', 'Page Title')` sets the value of the title section.

`@section('sidebar')` defines a sidebar section in the child page of master layout

`|` displays the contents of the sidebar section that is defined in the master layout

`<p>This is appended to the master sidebar.</p>` adds paragraph content to the sidebar section

`@endsection` ends the sidebar section

`@section('content')` defines the content section

`@section('content')` adds paragraph content to the content section

`@endsection` ends the content section

□ Template inheritance

❖ Extending the master layout

➤ Tạo route /app/Http/routes.php

```
Route::get('blade', function () {  
    return view('page');  
});
```

➤ Chạy <http://localhost/larashop/public/blade>



DEMO

- Demo sử dụng master layout





LARAVEL FRAMEWORK

BÀI 3 (PHẦN 2)

- ❑ Data truyền vào trong Blade views bằng cách đặt biến vào trong cặp **dấu ngoặc nhọn {...}**
- ❑ Tạo route có nội dung: *Route::get('greeting', function () { return view('welcome', ['name' => 'Fpoly']); });*
- ❑ Trong View, thông thường để hiển thị giá trị của biến thì sẽ sử dụng **echo** trong php
<?php echo \$name; ?>

- ❑ Khi sử dụng **Blade Template** có thể viết
{{ \$name }}//Xuất toàn bộ nội dung kể cả tag HTML,
Javascript
{!! \$name !!}// Biên dịch mã HTML, Javascript
- ❑ Cặp dấu {{ }} của Blade được tự động gửi tới hàm
htmlentities của PHP để ngăn chặn các hành vi
tấn công XSS.
- ❑ Sử dụng dấu @ để báo cho Blade biết là biểu
thức sẽ được giữ lại: Hello, @{{ name }}

Blade hỗ trợ kế thừa template, hiển thị dữ liệu, Blade cũng cung cấp các tiện ích cho các cấu trúc điều khiển cơ bản của PHP, như mệnh đề điều kiện hay vòng lặp.

Blade templating

- Provide own control structures.
 - Conditional statements
 - Loops
- Alternative way to echo variable values.
- Fast in rendering page.

Examples:

@if (....)	@for(....)
...	...
@elseif	@endfor
...	
@else	
...	
@foreach (....)	
...	
@endforeach	

My name is <?php echo \$user->name ?>
My name is {{ \$user->name }}

❑ BLADE CONDITIONAL

- ❖ Blade sử dụng if với các directive @if, @elseif, @else, và @endif, gần giống với Php syntax

```
@if (count($records) === 1)
    I have one record!
elseif (count($records) > 1)
    I have multiple records!
else
    I don't have any records!
@endif
```

❑ BLADE CONDITIONAL

- ❖ Blade cũng cung cấp kiểm tra @unless:

```
@unless (Auth::check())  
    You are not signed in.  
@endunless
```

- ❖ Kiểm tra @hasSection

```
@hasSection('navigation')  
    <div class="pull-right">  
        @yield('navigation')  
    </div>  
  
    <div class="clearfix"></div>  
@endif
```

❑ BLADE CONDITIONAL

- ❖ Switch Statements sử dụng @switch, @case, @break, @default và @endswitch directives:

```
@switch($i)
  @case(1)
    First case...
    @break

  @case(2)
    Second case...
    @break

  @default
    Default case...

@endswitch
```

❑ BLADE CONDITIONAL

```
1  @extends('layouts.master')
2
3  @section('title', 'Page Title')
4
5  @section('sidebar')
6
7      <p>This is appended to the master sidebar.</p>
8  @endsection
9
10
11 @section('content')
12 <h2>{{ $name }}</h2>
13 <p>This is my body content.</p>
14 <h2>If Statement</h2>
15 @if ($day == 'Friday')
16     <p>Time to party</p>
17 @else
18     <p>Time to make money</p>
19 @endif
20 @endsection
```

❑ BLADE LOOPS

- ❖ Blade cung cấp các phương thức hỗ trợ cho việc xử lý vòng lặp, và các phần này tương tự với PHP

```
@for ($i = 0; $i < 10; $i++)  
    The current value is {{ $i }}  
@endfor  
  
@foreach ($users as $user)  
    <p>This is user {{ $user->id }}</p>  
@endforeach  
  
@forelse ($users as $user)  
    <li>{{ $user->name }}</li>  
@empty  
    <p>No users</p>  
@endforelse  
  
@while (true)  
    <p>I'm looping forever.</p>  
@endwhile
```

❑ BLADE LOOPS

- ❖ Khi sử dụng vòng lặp, có thể cần kết thúc hay bỏ qua vòng lặp hiện tại

```
@foreach ($users as $user)
    @if($user->type == 1)
        @continue
    @endif

    <li>{{ $user->name }}</li>

    @if($user->number == 5)
        @break
    @endif
@endforeach
```

❑ BLADE LOOPS

- ❖ Thêm điều kiện trong vòng lặp

```
@foreach ($users as $user)
    @continue($user->type == 1)

    <li>{{ $user->name }}</li>

    @break($user->number == 5)
@endforeach
```


❑ BLADE LOOPS

❖ The Loop Variable

```
@foreach ($users as $user)

    @if ($loop->first)

        This is the first iteration.

    @endif

    @if ($loop->last)

        This is the last iteration.

    @endif

    <p>This is user {{ $user->id }}</p>

@endforeach
```

❑ EXECUTING PHP FUNCTIONS IN BLADE

```
@extends('layouts.master')

@section('title', 'Page Title')

@section('sidebar')

    <p>This is appended to the master sidebar.</p>
@endsection

@section('content')
<h2>{{ $name }}</h2>
<p>This is my body content.</p>
<h2>If Statement</h2>
@if ($day == 'Friday')
    <p>Time to party</p>
@else
    <p>Time to make money</p>
@endif

<h2>Foreach Loop</h2>
@foreach ($drinks as $drink)
    {{ $drink }} <br>
@endforeach

<h2>Execute PHP Function</h2>
<p>The date is {{ date(' D M, Y') }}</p>
@endsection
```

❑ Including Sub-Views

- ❖ Chèn thêm một Blade view vào trong một view có sẵn

```
<div>
    @include('shared.errors')

    <form>
        <!-- Form Contents -->
    </form>
</div>
```

- ❖ View được chèn thêm kế thừa tất cả dữ liệu từ view cha, cũng có thể truyền vào một mảng các dữ liệu bổ sung

```
@include('view.name', ['some' => 'data'])
```

❑ Stacks

- ❖ Blade sử dụng stack để có thể được render ở một vị trí nào trong view hay layout khác:

```
@push('scripts')  
    <script src="/example.js"></script>  
@endpush
```

- ❖ Sử dụng @stack để render

```
<head>  
    <!-- Head Contents -->  
  
    @stack('scripts')  
</head>
```



DEMO

- Tạo và sử dụng master page
- Xây dựng Blade dùng Control structure



Phần I: Giới thiệu về Template và Blade

 Template website

 Template trong Laravel

 Blade template

Phần II: Blade syntax

 Control Structures

 Executing PHP functions in blade





Cảm ơn