

Laravel Blade

- Laravel Blade is a templating engine that comes built-in with the Laravel PHP framework. It allows developers to write clean and structured HTML templates with dynamic content and reusable components.
- Blade templates use a combination of plain HTML and special Blade syntax, such as double curly braces `{{ }}` for variable output and at signs `@` for control structures like loops and conditionals.
- Blade templates can be extended to create a base template that can be reused across multiple pages with different content.
- Blade also provides several built-in directives that simplify common tasks like including subviews, injecting content into sections, and rendering JSON data.



PASS AND DISPLAYING DATA

- Display data that is passed to your Blade views by wrapping the variable in curly braces
- Blade's {{ }} echo statements are automatically sent through PHP's htmlspecialchars function to prevent XSS attacks.

```
● ● ● DemoController.php

7      function DemoAction(Request $request)
8      {
9          $sum=$request->num1+$request->num2;
10         return view('Home', ['sum' => $sum]);
11     }
```

```
● ● ● Home.blade.php

8      <body>
9          <h1>Sum is={{ $sum }}</h1>
10     </body>
```

BLADE IF STATEMENT

You may construct if statements using the @if, @elseif, @else, and @endif directives.

●●● Home.blade.php

```
8   <body>
9   @if ($sum === 1)
10      <h6>I have 1 record!</h6>
11   @elseif ($sum === 2)
12      <h6>I have 2 record!</h6>
13   @else
14      <h6>I have many record!</h6>
15   @endif
16 </body>
```

BLADE SWITCH CASE

Switch statements can be constructed using the @switch, @case, @break, @default and @endswitch directives

```
● ● ● Home.blade.php

8   <body>
9   @switch($sum)
10      @case(1)
11          <h3>First case...</h3>
12          @break
13      @case(2)
14          <h3>Second case...</h3>
15          @break
16      @default
17          Default case...
18  @endswitch
19  </body>
```


BLADE FOR LOOP

● ● ● Home.blade.php

```
8   <body>
9   @for ($i = 0; $i < 10; $i++)
10      The current value is {{ $i }}
11   @endfor
12 </body>
```

BLADE FOREACH LOOP

●●● DemoController.php

```
7     function DemoAction(Request $request)
8     {
9         $users=[
10             ["fname" => "Zoe", "lname" => "Upfield"],
11             ["fname" => "Bella", "lname" => "Cooke"],
12             ["fname" => "Liam", "lname" => "Ridley"]
13         ];
14         return view('Home', ['users' =>$users]);
15     }
```

●●● Home.blade.php

```
8     <body>
9     @foreach ($users as $user)
10         <p>This is user {{ $user['fname'] }}</p>
11     @endforeach
12 </body>
```

INCLUDING ASSET

●●● Home.blade.php

```
1  <!DOCTYPE html>
2  <html lang="en">
3      <head>
4          <meta charset="utf-8">
5          <meta name="viewport" content="width=device-width, initial-scale=1">
6          <title>Laravel</title>
7          <link href="{{asset('/css/bootstrap.min.css')}}" rel="stylesheet">
8      </head>
9      <body>
10     
11 </body>
12 </html>
```

INCLUDING SUBVIEWS

Blade's `@include` directive allows you to include a Blade view from within another view. All variables that are available to the parent view will be made available to the included view:

```
● ● ● Home.blade.php

10  <div>
11      @include('shared.errors')
12
13      <form>
14          <!-- Form Contents -->
15      </form>
16  </div>
```


MASTER LAYOUT CONCEPT

Layout.blade.php

```
5 <body>
6     @include('component.menu')
7     @yield('content')
8     @include('component.footer')
9 </body>
```

Home.blade.php

```
1 @extends('Layout')
2
3 @section('content')
4
5     @include('Component.HomeBanner')
6     @include('Component.HomeService')
7     @include('Component.HomeCourse')
8     @include('Component.HomeProjects')
9     @include('Component.HomeContact')
10    @include('Component.HomeReview')
11
12 @endsection
```