

TEMPLATE - BLADE IN LARAVEL

TEMPLATE

Organizziamo le view in modo da semplificare il codice e ridurre al minimo la duplicazione del codice.

Avremo:

- una master o layout page**
- pagine specifiche per ogni contenuto**

MASTER PAGE: resources/views/app.blade.php

```
<!doctype html>
<html lang="it">
<head>
  <meta charset="UTF-8">
  <title>@yield('title')</title>
</head>
<body><div class="container">@yield('content')</div></body>
</html>
```

IN OGNI VIEW:

@extends('app')

@section('title')

Pagina importante

@stop

@section('content')

<h1>Titolo importante</h1>

<p>Contenuto della pagina</p>

@stop

NELLE VIEW

E' possibile NON dichiarare una sezione che non viene usata nella pagina.

Nella master page devono essere indicate tutte le sezioni che prevediamo di avere.

Stiamo usando Blade di Laravel, non si possono usare le istruzioni che iniziano con @ nel semplice PHP.

CONDIZIONALI NELLE VIEW

Usando Blade:

```
@if ($name = 'Leonardo')
```

```
<p>Ciao docente!</p>
```

```
@else
```

```
<p>Ciao studente!</p>
```

```
@endif
```

CONDIZIONALI NELLE VIEW

Usando PHP semplice senza Blade:

```
<?php if ($name = 'Leonardo') { ?>
```

```
<p>Ciao docente!</p>
```

```
<?php } else { ?>
```

```
<p>Ciao studente!</p>
```

```
<?php } ?>
```

CICLI NELLE VIEW

<h3>Studenti del corso</h3>

@if (count(\$students))

@foreach (\$students as \$student)

{{ \$student }}

@endforeach

@endif

CICLI NELLE VIEW

Visto che usiamo la variabile `$students`, nel controller avremo:

```
$students = [  
    'A', 'B', 'C'  
];  
return view('pages.example', compact('students'));
```

GRAZIE
per l'attenzione!