

Blade Component in laravel 7

DRY Up Your HTML With Laravel 7 Blade Components

- ★ Follow the DRY (Don't Repeat Yourself) principle

ခန့်ခန့် ပြန်လည်ပြသပေးရမဲ့ view တွေအတွက် လိုအပ်သော data တွေအမြဲတမ်း view ဘက်ကို ထည့်ပေးနေတဲ့ MVC Design pattern တွေကို လိုက်နာလုပ်ဆောင်ရတာ မလွယ်ကူပါဘူး။ နောက်ပြီး တူညီတဲ့ code တွေကို ထပ်တစ်လဲလဲ ရေးပေးမယ်ဆို developer တွေအတွက် အတော်လေး မဟုတ်သေးပါဘူး။ များများရေးမယ်ဆို များများ error တက်စေမှာပါ။ laravel 7 ရဲ့ good feature တစ်ခု (blade component) က ဒီပြဿနာတွေကို အလွယ်ကူဆုံးဖြေရှင်းပေးမှာပါ။

To start using component in your project, please run the following artisan command

php artisan make:component TestingComponent

1. **app/View/Components/** directory အောက်မှာ component class တစ်ခုတည်ဆောက်ပြီး
 - a. Component Class (TestingComponent.php)
2. **resources/views/components/** directory အောက်မှာလဲ blade template file တစ်ခုတည်ဆောက်ပေးသွားပါတယ်
 - a. Component blade template (testing-component.blade.php)

- ★ အသုံးပြုပုံ

Component class နဲ့ blade template နှစ်ခုလုံးအသုံးပြုဖို့ အဆင်သင့်ဖြစ်နေပြီဆိုရင် ကိုယ်အသုံးပြုချင်တဲ့ blade file ထဲမှာ component blade template ရဲ့ နာမည်ကို x- ခံပြီး ပြန်ခေါ်လိုက်လို့ရပါပြီ။

[Passing Data To Components \(Parent Blade\)](#)

```
<x-testing-component :data="$data"></x-testing-component>
```

[app/View/Components](#) (TestingComponent Class)

```
<?php
namespace App\View\Components;
use Illuminate\View\Component;

class TestingComponent extends Component
{
    public $data;
    /**
     * Create a new component instance.
     *
     * @return void
     */
    public function __construct($data)
    {
        $this->data = $data;
    }

    /**
     * Get the view / contents that represent the component.
     *
     * @return \Illuminate\View\View|string
     */
    public function render()
    {
        return view('components.testing-component');
    }
}
```

[resources/views/components](#) (testing-component.blade.php)

```
<div>
    {{ $data }}
</div>
```