

Blade in laravel

developers love the Laravel [Blade](#) engine's robust ability to create modular and reusable views quickly

Laravel Blade is the default templating engine for the Laravel framework. It lets you use variables, loops, conditional statements, and other PHP features directly in your HTML code. To create Blade files, simply define blade

creating files with the `.blade.php` extension in the **resources/views** directory of your Laravel project, then structure your preferred dynamic pages in these files

Why Should You Use Blade?

One main benefit of Blade is its modular code organization. Blade helps organize your code into reusable modules that you can easily add, remove, or update without affecting the rest of your application.

Code encapsulation is another of Blade's advantages. Blade helps encapsulate functions, making testing, debugging, and code maintenance more manageable. This approach benefits larger applications since unorganized applications can quickly become challenging to manage.

Blade's templating engine is performant and the fastest PHP framework we tested. The engine compiles all your blade views into plain PHP code and then caches them until you modify them. This approach promotes faster rendering and overall better performance.

By default, Blade (and the Laravel `e` function) will double encode HTML entities. If you would like to disable double encoding, call

the `Blade::withoutDoubleEncoding` method from the `boot` method of your `AppServiceProvider`:

```
<?php

namespace App\Providers;

use Illuminate\Support\Facades\Blade;
use Illuminate\Support\ServiceProvider;

class AppServiceProvider extends ServiceProvider
{
    /**
     * Bootstrap any application services.
     */
    public function boot(): void
    {
        Blade::withoutDoubleEncoding();
    }
}
```