# user test

By default, Laravel includes the `App\Http\Middleware\TrimStrings` and `App\Http\Middleware\ConvertEmptyStringsToNull` middleware in your application's global middleware stack. These middleware are listed in the global middleware stack by the `App\Http\Kernel` class. These middleware will automatically trim all incoming string fields on the request, as well as convert any empty string fields to `null`. This allows you to not have to worry about these normalization concerns in your routes and controllers.

If you would like to disable this behavior, you may remove the two middleware from your application's middleware stack by removing them from the `$middleware` property of your `App\Http\Kernel` class.

By default, Laravel includes the `App\Http\Middleware\TrimStrings` and `App\Http\Middleware\ConvertEmptyStringsToNull` middleware in your application's global middleware stack. These middleware are listed in the global middleware stack by the `App\Http\Kernel` class. These middleware will automatically trim all incoming string fields on the request, as well as convert any empty string fields

to `null` . This allows you to not have to worry about these normalization concerns in your routes and controllers.

If you would like to disable this behavior, you may remove the two middleware from your application's middleware stack by removing them from the `$middleware` property of your `App\Http\Kernel` class.

By default, Laravel includes the `App\Http\Middleware\TrimStrings` and `App\Http\Middleware\ConvertEmptyStringsToNull` middleware in your application's global middleware stack. These middleware are listed in the global middleware stack by the `App\Http\Kernel` class. These middleware will automatically trim all incoming string fields on the request, as well as convert any empty string fields to `null` . This allows you to not have to worry about these normalization concerns in your routes and controllers.

If you would like to disable this behavior, you may remove the two middleware from your application's middleware stack by removing them from the `$middleware` property of your `App\Http\Kernel` class.