



# **Validation in Laravel**

**Create Validation in Laravel**

## • إيه هو ال Validation ؟

ال Validation ببساطة يعني "التأكد من صحة البيانات" اللي بيدخلها المستخدم، زي إنك تتأكد إن الإيميل مكتوب صح، أو إن كلمة السر مش أقل من ٨ حروف. Laravel بتقدم نظام جاهز وسهل للتعامل مع ده.

## • إزاي أعمل Validation ؟

عندي ٣ طرق :

### ١. استخدم () validate -> \$request مباشرة دي الطريقة السريعة والمباشرة :

```
public function store(Request $request) {
    $validatedData = $request->validate([
        'name' => 'required|string|max:255',
        'email' => 'required|email|unique:users',
    ]);

    // If the data is valid, proceed to create the user
    User::create($validatedData);
}
```

• required : يعني الحقل لازم يكتب.

• email : لازم يكون إيميل.

• unique:users : لازم ما يكونش مكرر في جدول users .

### ٢. استخدم Form Request Validation دي الطريقة الاحترافية لو عندك قواعد تحقق كتير :

• اعمل كلاس جديد:

```
php artisan make:request StoreUserRequest
```

• ضيف الرولز داخل الكلاس :

```
public function rules() {
    return [
        'name' => 'required|string|max:255',
        'email' => 'required|email|unique:users',
    ];
}
```

• استخدم الكلاس في الكونترولر :

- استخدم الكلاس في الكونترولر :

```
public function store(StoreUserRequest $request) {  
    User::create($request->validated());  
}
```

- ٣. استخدم Validator Facade  
لو محتاج مرونة أكثر :

```
use Validator;  
  
$validator = Validator::make($request->all(), [  
    'name' => 'required|string|max:255',  
    'email' => 'required|email|unique:users',  
]);  
  
if ($validator->fails()) {  
    return back()->withErrors($validator)->withInput();  
}
```

- إزاي أكتب رسائل خطأ مخصصة؟  
Laravel بتمكنك تغيير رسائل الخطأ بسهولة :

```
$request->validate([  
    'name' => 'required|max:255',  
], [  
    'name.required' => 'The name field is required!',  
    'name.max' => 'The name must not exceed 255 characters.',  
]);
```

- ليه ال Validation مهم؟

- يحمي تطبيقك من البيانات الخطأ.
- يعطي المستخدم رسائل واضحة عن الأخطاء.
- يمنع مشاكل في قاعدة البيانات.