

82. Usa eager loading en Laravel y carga correctamente imágenes de productos

Meter en log todas las queries

```
\DB::connection()>enableQueryLog();
```

Dumpearlo en welcome.blade.php

```
<div class="row">
    <!--Video 82-->
    @dump($products)

    @foreach($products as $product)
        <div class="col-3">
            @include('components.product-card')
        </div>
    @endforeach

    @dump($products)
    @dd(\DB::getQueryLog())
</div>
@endempty
@endsection
```

Eficiencia de consultas

Vemos que Laravel tiene que hacer una consulta por imagen, ya que, antes del foreach, estas no se han añadido a los productos. Este problema **n+1** se soluciona de manera muy sencilla cargando ya las imágenes con los productos desde el principio. ¿Cómo?

Así (MainController):

```
<?php

namespace App\Http\Controllers;
use App\Models\Product;

class MainController extends Controller {
    public function index() {
        // Momento aún más alfa (global scope)
        \DB::connection()>enableQueryLog();
        // images es el nombre de la relación
        // (o función en el Models\Product)
        $products = Product::with('images')>get();
        return view('welcome')>with(['products' => $products]);
    }
}

?>
```

O incluso mejor así (en Product):

```
class Product extends Model
{
    use HasFactory;

    protected $table = 'products';

    protected $with = [
        'images',
    ];

    ...
}
```

Pero ¿y si no necesito las imágenes?

Nos vamos al método donde no sea necesario ver las imágenes (index, en ProductController) y:

```
public function index() {
    return view('products.index')->with([
        // Para probar el condicional
        // 'products' => [],
        'products' => PanelProduct::without('images')->get(),
    ]);
}
```

Lo llamamos sin imágenes.