

## 98. Usa Laravel Soft Deleting para poder eliminar productos

El **Soft Deleting** es como enviar los productos a una papelera. Podemos recuperarlos.

Debemos añadir SoftDelete a las migraciones.

### En Product

```
...
use Illuminate\Database\Eloquent\SoftDeletes;

class Product extends Model
{
    use HasFactory, SoftDeletes;
    ...
}
```

### Database\migrations\create\_products\_table

```
...
public function up() {
    Schema::create('products', function (Blueprint $table) {
        $table->id();
        $table->string('title');
        // Máx 1000 caracteres (default 255)
        $table->string('description', 1000);
        $table->float('price')->unsigned();
        $table->integer('stock')->unsigned();
        $table->string('status')->default('unavailable');
        $table->softDeletes(); // Se añade
        $table->timestamps();
    });
}
...
```

### Migrar tras las modificaciones

```
php artisan migrate:fresh --seed
```

### Probar que se ha añadido

Cogemos un producto cualquiera y observamos que ahora tiene una nueva columna llamada `deleted_at`.

```
App\Models\Product {#4635
  id: "42",
  title: "Iste eveniet consectetur.",
  description: "Consequatur aut fuga quia sed distinctio totam laborum et.",
  price: "17.07",
  stock: "1",
  status: "available",
  deleted_at: null,
  created_at: "2023-02-11 18:34:23",
}
```

```

updated_at: "2023-02-11 18:34:23",
images: Illuminate\Database\Eloquent\Collection {#4609
  all: [
    App\Models\Image {#4782
      id: "148",
      path: "img/products/10.jpg",
      created_at: "2023-02-11 18:34:24",
      updated_at: "2023-02-11 18:34:24",
      imageable_type: "App\Models\Product",
      imageable_id: "42",
    },
    App\Models\Image {#4783
      id: "149",
      path: "img/products/9.jpg",
      created_at: "2023-02-11 18:34:24",
      updated_at: "2023-02-11 18:34:24",
      imageable_type: "App\Models\Product",
      imageable_id: "42",
    },
    App\Models\Image {#4784
      id: "150",
      path: "img/products/5.jpg",
      created_at: "2023-02-11 18:34:24",
      updated_at: "2023-02-11 18:34:24",
      imageable_type: "App\Models\Product",
      imageable_id: "42",
    },
  ],
},
},
},
},

```

## Probar que se ha borrado

1. Ir a Laravel y borrar algún producto con un usuario admin.
2. Abrir tinker y mirar los productos (no podemos ver el que hemos borrado).

### ⚠ Ignorar los global scopes

Ahora sí sale `deleted_at`.

```

>>>> App\Models\Product::withoutGlobalScopes()->get()
=> Illuminate\Database\Eloquent\Collection {#4589
  all: [
    App\Models\Product {#4739
      id: "1",
      title: "Sed ut labore molestiae.",
      description: "Aut eius modi consequatur ut.",
      price: "92.17",
      stock: "10",
      status: "available",
      deleted_at: "2023-02-11 19:00:05",
      created_at: "2023-02-11 18:34:23",
      updated_at: "2023-02-11 19:00:05",
      images: Illuminate\Database\Eloquent\Collection {#4964
        all: [
          App\Models\Image {#4968
            id: "21",
            path: "img/products/1.jpg",
            created_at: "2023-02-11 18:34:23",

```

```

        updated_at: "2023-02-11 18:34:23",
        imageable_type: "App\Models\Product",
        imageable_id: "1",
    },
    App\Models\Image {#4969
        id: "22",
        path: "img/products/9.jpg",
        created_at: "2023-02-11 18:34:23",
        updated_at: "2023-02-11 18:34:23",
        imageable_type: "App\Models\Product",
        imageable_id: "1",
    },
    App\Models\Image {#4970
        id: "23",
        path: "img/products/9.jpg",
        created_at: "2023-02-11 18:34:23",
        updated_at: "2023-02-11 18:34:23",
        imageable_type: "App\Models\Product",
        imageable_id: "1",
    },
],
},
},
},

```

## Recuperar solo los borrados

En este caso restauramos el primer borrado:

```

>>> App\Models\Product::withoutGlobalScopes()→onlyTrashed()→get()
⇒ Illuminate\Database\Eloquent\Collection {#4855
    all: [
        App\Models\Product {#4861
            id: "1",
            title: "Sed ut labore molestiae.",
            description: "Aut eius modi consequatur ut.",
            price: "92.17",
            stock: "10",
            status: "available",
            deleted_at: "2023-02-11 19:00:05",
            created_at: "2023-02-11 18:34:23",
            updated_at: "2023-02-11 19:00:05",
            images: Illuminate\Database\Eloquent\Collection {#4864
                all: [
                    App\Models\Image {#4865
                        id: "21",
                        path: "img/products/1.jpg",
                        created_at: "2023-02-11 18:34:23",
                        updated_at: "2023-02-11 18:34:23",
                        imageable_type: "App\Models\Product",
                        imageable_id: "1",
                    },
                    App\Models\Image {#4866
                        id: "22",
                        path: "img/products/9.jpg",
                        created_at: "2023-02-11 18:34:23",
                        updated_at: "2023-02-11 18:34:23",
                        imageable_type: "App\Models\Product",
                        imageable_id: "1",
                    },
                ],
            },
        },
    ],
}

```

```

App\Models\Image {#4867
  id: "23",
  path: "img/products/9.jpg",
  created_at: "2023-02-11 18:34:23",
  updated_at: "2023-02-11 18:34:23",
  imageable_type: "App\Models\Product",
  imageable_id: "1",
},
],
},
],
}
// Solo desechados
>>> App\Models\Product::withoutGlobalScopes()->onlyTrashed()->first()->restore()
=> true

```

## Borrar permanentemente

`forceDelete()`, pero no lo ha explicado mucho por los problemas con claves foráneas y ese rollo.

Que te suene por si en algún momento hace falta.