66. Usando recursos anidados de Laravel para agregar productos al carrito

Creamos el nuevo controlador

En cmd:

```
C:\Users\llemi\Desktop\Lunk\Laravel\ola>
php artisan make:controller ProductCartController -m Cart -p Product
Controller created successfully.
```

① El — sirve para indicarle el modelo para usar la inyección implícita del model; el — para la inyección implícita del padre.

En ProductCartController:

Eliminamos todas las funciones que no sean store() y destroy().

① Éstas se generan automáticamente, incluyendo incluso los parámetros que debemos pasar.

Agregamos una nueva ruta web.php:

```
vse Illuminate\Support\Facades\Route;

Route::get('/', 'MainController@index') \rightarrow name('welcome');

Route::resource('products', 'ProductController');

Auth::routes();

Route::get('/home', [App\Http\Controllers\HomeController::class, 'index']) \rightarrow name('home');

// Esta es la nueva ruta, seleccionando solo dos métodos
Route::resource('products.carts', 'ProductCartController') \rightarrow only(['store', 'destroy']);

?>
```

Comprobamos con php artisan route: list:

Y debería de salir algo así, pero con mejor formato (no cabe xd):

. . .

Agregamos un formulario que ejecute store()

En ProductCartController:

```
<?php
namespace App\Http\Controllers;
use App\Models\Cart;
use App\Models\Product;
use Illuminate\Http\Request;
class ProductCartController extends Controller
    // BORRAMOS TODAS LAS FUNCIONES MENOS STORE Y DESTROY
    // Las funciones se crean automáticamente
     * Store a newly created resource in storage.
     * @param \Illuminate\Http\Request $request
     * @param \App\Models\Product $product
     * @return \Illuminate\Http\Response
     */
    public function store(Request $request, Product $product)
        $cart = Cart::create();
        // Se accede a la cantidad de un determinado
        // producto que hay en el carrito
        $quantity = $cart→products()
             \rightarrowfind($product\rightarrowid)
             \rightarrowquantity ?? 0;
        // Se crea un carrito cada vez: no muy eficiente
        $cart→products()→attach([
            product \rightarrow id \Rightarrow ['quantity' \Rightarrow quantity + 1],
        ]);
             // Se redirige
        return redirect()→back();
    // Sigue destroy()
```

ి Mejora de eficiencia

No tener esto muy en cuenta, ya que en posteriores vídeos se mejorará la forma de agregar productos al carrito.