

Ejercicio 1: Catálogo de Productos Comerciales

Objetivo: Mostrar una lista de productos obtenidos desde el modelo y presentarlos en una vista utilizando un controlador.

Instrucciones:

1. Crea un nuevo proyecto Laravel llamado **catalogo-comercial**.
 2. Crea un modelo **Producto** sin base de datos, que devuelva una lista simulada (array estático) de productos con **nombre**, **precio** y **descripcion**.
 3. Crea un controlador **ProductoController** con un método **index** que:
 - Obtenga la lista de productos desde el modelo.
 - Pase los datos a una vista Blade.
 4. Crea una vista Blade que muestre los productos en formato de tarjetas o tabla.
 5. Define una ruta **/productos** que apunte al método **index** del controlador.
-

Estructura de Archivos

```
/routes
└── web.php

/app
├── Http
│   └── Controllers
│       └── ProductoController.php
└── Models
    └── Producto.php

/resources
├── views
│   └── productos
│       └── index.blade.php
```

Ejercicio 2: Listado de Facturas Emitidas

Objetivo: Practicar el uso de modelos con datos simulados, controladores que organizan la lógica, y vistas que presentan información comercial.

Instrucciones:

1. Crea un nuevo proyecto Laravel llamado **facturas-comerciales**.
2. Crea un modelo **Factura** con una función que devuelva un conjunto de facturas simuladas (array), incluyendo **numero**, **cliente**, **importe** y **fecha**.
3. Crea un controlador **FacturaController** con un método **index** para:
 - Obtener las facturas desde el modelo.
 - Enviarlas a una vista.
4. Crea una vista Blade que:
 - Muestre la lista de facturas en una tabla.
 - Destaque en otro color las que superen cierto importe.
5. Define una ruta **/facturas** que cargue esta vista a través del controlador.

Estructura de Archivos

```
/routes
└── web.php

/app
├── Http
│   └── Controllers
│       └── FacturaController.php

/app
├── Models
│   └── Factura.php

/resources
└── views
```

```
└─ facturas
   └─ index.blade.php
```

Ejercicio 3: Seguimiento de Pedidos Comerciales

Objetivo: Usar modelos para organizar datos, controladores para gestionar la lógica de negocio, y Blade para presentar el estado de cada pedido.

Instrucciones:

1. Crea un proyecto llamado **seguimiento-pedidos**.
2. Crea un modelo **Pedido** con una función que retorne un array con los siguientes datos por pedido: **id**, **cliente**, **producto**, **estado** (Ej: **Pendiente**, **Enviado**, **Entregado**).
3. Crea un controlador **PedidoController** con un método **index** que:
 - Obtenga todos los pedidos.
 - Los pase a una vista Blade.
4. En la vista:
 - Muestra los pedidos en una tabla.
 - Usa condicionales en Blade para mostrar un color diferente o icono según el estado del pedido.
5. Define la ruta **/pedidos** apuntando al controlador.

Estructura de Archivos

```
/routes
└─ web.php

/app
```

```
└─ Http
  └─ Controllers
    └─ PedidoController.php
```

```
/app
└─ Models
  └─ Pedido.php
```

```
/resources
└─ views
  └─ pedidos
    └─ index.blade.php
```