

Evidencia:

Product.js - Creación y validación.

```
> let p1 = new Product("Collar zafiro", 999, "Dama", "https://ejemplo.com/collar.jpg");
< undefined
> console.log(p1)
VM298:1
  ▶ Product { _title: 'Collar zafiro', _price: 999, _category: 'Dama', _imageUrl: 'https://ejempl
    o.com/collar.jpg', _id: null }
< undefined

> let p2 = new Product("Anillo zafiro", -10, "Dama", "https://ejemplo.com/anillo.jpg");
✖ ▶ Uncaught ProductException: El precio debe ser un número positivo. product.js:50
    at set price (http://127.0.0.1:5500/P2/product.js:50:13)
    at new Product (http://127.0.0.1:5500/P2/product.js:13:16)
    at <anonymous>:1:10
>
```

Data_handler.js

- Crea objeto

```
> let prod = {title: "Anillo Luna", price: 1550, category: "Dama",
  imageUrl: "https://ejemplo.com/anillo.jpg"};
< undefined
> data_handler.createProduct(prod).then(console.log);
< ▶ Promise {<pending>}
  {title: 'Anillo Luna', price: 1550, category: 'Dama', imageUr
  ▶ l: 'https://ejemplo.com/anillo.jpg', _id: '67e36333ddc32303e8d
    f1fbb'}
```

- Obtener todos los productos

```
> data_handler.getProducts().then(console.log);
< ▶ Promise {<pending>}
  ▶ (3) [Product, Product, Product] ⓘ
    ▶ 0: Product { _title: 'Collar estrella', _price: 1499, _category: 'Dama', _imageUrl: 'http:
    ▶ 1: Product { _title: 'Aretes sol', _price: 1200, _category: 'Dama', _imageUrl: 'https://in
    ▶ 2: Product { _title: 'Anillo Luna', _price: 1550, _category: 'Dama', _imageUrl: 'https://e
      length: 3
    ▶ [[Prototype]]: Array(0)
```

- Obtenerlo con ID

```
▶ Promise {<pending>}

▶ (3) [Product, Product, Product] ⓘ
  ▶ 0: Product {_title: 'Collar estrella', _price: 1499, _category: 'Dama', _imageUrl: 'https://'}
  ▶ 1: Product
    _category: "Dama"
    _id: "67e33dd4b3d59903e8d6af65"
    _imageUrl: "https://images.unsplash.com/photo-1680181724947-75f0956f1469?q=80&w=2080&auto"
    _price: 1200
    _title: "Aretes sol"
    category: (...)
    id: (...)
    imageUrl: (...)
    price: (...)
    title: (...)
  ▶ 2: Product {_title: 'Anillo Luna', _price: 1550, _category: 'Dama', _imageUrl: 'https://eje'}
    length: 3
  ▶ [[Prototype]]: Array(0)

data_handler.getProductById("67e33dd4b3d59903e8d6af65").then(console.log);

▶ Promise {<pending>}

Product {_title: 'Aretes sol', _price: 1200, _category: 'Dama', _imageUrl: 'https://images.u'}
▶ nsplash.com/photo-1680181724947-75...9?q=80&w=2080&auto=format&fit=crop&ixlib=rb-4.0.3', _id: '67e33dd4b3d59903e8d6af65'}
```

- Actualizar producto

```
> let actualizado = { title: "Aretes sol", price: 1300, category: "Dama",
  imageUrl:"https://ejemplo.com/aretes.jpg"};
< undefined

> data_handler.updateProduct("67e33dd4b3d59903e8d6af65",
  actualizado).then(console.log("Actualizado"));

Actualizado VM1821:1

< ▶ Promise {<pending>} ⓘ
  ▶ [[Prototype]]: Promise
    [[PromiseState]]: "fulfilled"
    [[PromiseResult]]: undefined
>
```

- Eliminar producto

```
> data_handler.deleteProduct ("67e33dd4b3d59903e8d6af65").then(() =>
  console.log("Eliminado"));

< ▶ Promise {<pending>} ⓘ
  ▶ [[Prototype]]: Promise
    [[PromiseState]]: "fulfilled"
    [[PromiseResult]]: undefined
  Eliminado VM1971:1

> data_handler.getProducts().then(console.log);

< ▶ Promise {<pending>}
  ▶ (2) [Product, Product] ⓘ
    ▶ 0: Product {_title: 'Collar estrella', _price: 1499, _category: 'Dama', _imageUrl: 'http:
    ▶ 1: Product {_title: 'Anillo Luna', _price: 1550, _category: 'Dama', _imageUrl: 'https://e
      length: 2
    ▶ [[Prototype]]: Array(0)
```

Index.js

```
> data_handler.getProducts().then(console.log);

< ▶ Promise {<pending>}
  ▶ (2) [Product, Product] ⓘ
    ▶ 0: Product {_title: 'Collar estrella', _price: 1499, _category: 'Dama', _imageUrl: 'http:
    ▶ 1: Product {_title: 'Anillo Luna', _price: 1550, _category: 'Dama', _imageUrl: 'https://e
      length: 2
    ▶ [[Prototype]]: Array(0)
```



Joyeria destacada

Collar estrella
\$ 1499
Agregar al carrito

Anillo Luna
\$ 1550
Agregar al carrito

« 1 2 3 »

Cart.js

- Agregar

```
0: Product { _title: 'Collar estrella', _price: 1499, _category: 'Dama', _imageUrl: 'https://ejemplo.com/anillo.jpg' }
1: Product
  _category: 'Dama'
  _id: '67e36333ddc32303e8df1fbb'
  _imageUrl: 'https://ejemplo.com/anillo.jpg'
  _price: 1550
  _title: 'Anillo Luna'
  category: (...)
  id: (...)
  imageUrl: (...)
  price: (...)
  title: (...)
  [[Prototype]]: Object
  length: 2
  [[Prototype]]: Array(0)

> carrito_handler.addItem("67e36333ddc32303e8df1fbb", 2);
< undefined

> console.log(carrito_handler.carrito);
VM2501:1
  [{...}]
  0: {uuid: '529babb4-467b-4e12-a01a-60ea9debb23c', productId: '67e36333ddc32303e8df1fbb', an
    length: 1
    [[Prototype]]: Array(0)
  }
  undefined
```

- Mostrar total:

```
> data_handler.getProducts().then(products => { let total =
  carrito_handler.calculateTotal(products); console.log("Total del carrito: $" + total); });

< Promise {<pending>}
Total del carrito: $3100
VM2817:1
```

Rubrica:

Sección	Puntuación	
Home muestra productos leídos del API o local storage.	30 puntos	✓
Uso de excepciones y validaciones	10 puntos	✓
CRUD del carrito de compras	35 puntos	✓
Pruebas y ejemplos	5 puntos	✓
Paginación de Pokédex	30 puntos	✓

Conclusiones:

A lo largo de esta práctica tuve muchas complicaciones, ya que al inicio no comprendía bien qué era un CRUD ni cómo debía estructurarse correctamente. Me tomó bastante

tiempo hacerlo funcionar, y con mucha ayuda de terceros logré que el sistema trabajara de manera adecuada, aunque no al 100% al principio.

Uno de los mayores desafíos fue al momento de conectar la API, ya que me aparecían diferentes errores (como los de CORS o rutas), y al no conocerlos todos, tuve que investigar y probar distintas soluciones por mi cuenta.

Sin embargo, esta práctica me ayudó muchísimo a recordar los fundamentos de la Programación Orientada a Objetos (POO) y ponerlos nuevamente en práctica. Esa parte me gustó mucho, ya que me permitió reforzar conocimientos que no había usado desde hace tiempo.

En conclusión, esta práctica me enfrentó a muchos retos técnicos y de lógica, pero me ayudó bastante a consolidar lo aprendido en clase. Fue un ejercicio completo y retador, que me deja mejor preparada para proyectos más complejos.