

■ Enunciado:

Una tienda online quiere almacenar información sobre sus productos. Crea la base de datos y realiza las

Parte A - Creación

1. Crea la base de datos `miTienda`.
2. Crea la colección `productos`.
3. Inserta los siguientes productos:
{ nombre: "Portátil", precio: 899.99, stock: 15, categoria: "Informática" }
{ nombre: "Ratón", precio: 19.99, stock: 50, categoria: "Accesorios" }
{ nombre: "Teclado", precio: 29.99, stock: 30, categoria: "Accesorios" }
{ nombre: "Monitor", precio: 199.99, stock: 20, categoria: "Informática" }

Parte B - Consultas (Read)

4. Muestra todos los productos con `precio` mayor a 50.
5. Muestra solo el `nombre` y `precio` de los productos de la categoría "Accesorios", sin mostrar `_id`.
6. Ordena los productos por `stock` de mayor a menor.

Parte C - Actualización (Update)

7. Añade un campo `oferta: true` a todos los productos con precio menor a 100.

Parte D - Eliminación (Delete)

8. Elimina los productos que tienen un `stock` menor a 20.

Extra (opcional)

9. Cuenta cuántos productos quedan después de las operaciones anteriores.

■ Soluciones esperadas:

```
use miTienda
```

```
// Parte A
db.createCollection("productos")
db.productos.insertMany([
  { nombre: "Portátil", precio: 899.99, stock: 15, categoria: "Informática" },
  { nombre: "Ratón", precio: 19.99, stock: 50, categoria: "Accesorios" },
  { nombre: "Teclado", precio: 29.99, stock: 30, categoria: "Accesorios" },
  { nombre: "Monitor", precio: 199.99, stock: 20, categoria: "Informática" }
])

// Parte B
db.productos.find({ precio: { $gt: 50 } })
db.productos.find({ categoria: "Accesorios" }, { nombre: 1, precio: 1, _id: 0 })
db.productos.find().sort({ stock: -1 })

// Parte C
db.productos.updateMany(
  { precio: { $lt: 100 } },
  { $set: { oferta: true } }
)

// Parte D
db.productos.deleteMany({ stock: { $lt: 20 } })

// Extra
db.productos.countDocuments()
```