

Actividad Práctica: Pensamiento Orientado a Objetos y ES6+ en JavaScript

Objetivo: Implementar una clase usando la sintaxis moderna de ES6+ para simular un sistema de inventario de productos. Aplicar métodos, propiedades, métodos estáticos y arrow functions.

Instrucciones:

1. Crea tu archivo de trabajo

- Abre Visual Studio Code.
 - Crea una nueva carpeta llamada InventarioProductos.
 - Dentro de la carpeta, crea un archivo con el nombre inventario.js.
-

2. Define la clase Producto

- Copia y pega el siguiente código en inventario.js:

javascript

```
class Producto {  
  constructor(nombre, precio, stock) {  
    this.nombre = nombre;  
    this.precio = precio;  
    this.stock = stock;  
  }  
  vender(cantidad) {  
    if (cantidad <= this.stock) {  
      this.stock -= cantidad;  
      return `Venta realizada: ${cantidad} unidad(es) de ${this.nombre}. Stock  
restante: ${this.stock}`;  
    } else {
```

```

        return `No hay suficiente stock de ${this.nombre} para realizar la venta.`;
    }
}

reponer(cantidad) {
    this.stock += cantidad;
    return `Stock de ${this.nombre} aumentado a ${this.stock}.`;
}

get informacion() {
    return `Producto: ${this.nombre} | Precio: $$${this.precio} | Stock:
    ${this.stock}`;
}

static compararPrecio(prodA, prodB) {
    return prodA.precio > prodB.precio ? prodA.nombre : prodB.nombre;
}
}

```

- Clase: Producto
- Propiedades: nombre, precio, stock

3. Crea instancias y simula acciones

- Agrega el siguiente código al final del archivo para probar tu clase:

javascript

```

const prod1 = new Producto('Teclado', 25000, 10);
const prod2 = new Producto('Mouse', 12000, 5);

```

```

console.log(prod1.informacion);    // Información inicial

```

```
console.log(prod1.vender(3));    // Simula venta
console.log(prod1.reponer(5));   // Repone stock
console.log(prod1.informacion);  // Nuevo stock
```

```
console.log(`El producto más caro es: ${Producto.compararPrecio(prod1,
prod2)}`);
```

4. Filtra productos con arrow function

- Copia y pega este bloque para filtrar productos con bajo stock:

javascript

```
const listaProductos = [prod1, prod2, new Producto('Monitor', 55000, 2)];
const productosBajoStock = listaProductos.filter(prod => prod.stock < 5);
console.log(productosBajoStock.map(prod => prod.informacion));
```

- Aquí estás usando una arrow function y el método filter para encontrar productos con stock menor a 5.
-


Detalles clave del ejercicio

- ¡No olvides comentar tu código! Escribe, sobre cada bloque, comentarios cortos que expliquen qué hace ese fragmento y por qué.
- Prueba tu código: Ejecuta el archivo en la terminal con el comando:

node inventario.js

- Si tienes dudas, puedes consultar la documentación oficial de JavaScript sobre clases y objetos para reforzar tu aprendizaje.
-

Tiempo estimado

 40 a 50 minutos (incluye tiempo para repaso, escritura y prueba del código).

Entrega

- Sube el proyecto (la carpeta completa o sólo el archivo inventario.js) a un repositorio público en GitHub.
- Copia el enlace de tu repositorio y pégalo en el espacio habilitado para entregas en Moodle.
- Fecha de entrega: Será indicada por el profesor.