



UEA

UNIVERSIDAD

ESTATAL AMAZÓNICA

UNIVERSIDAD ESTATAL AMAZÓNICA



INGENIERÍA EN TECNOLOGÍAS DE INFORMACIÓN

PROGRAMACIÓN ORIENTADA A OBJETOS

TEMA:

SISTEMA DE GESTIÓN DE INVENTARIOS MEJORADO

ESTUDIANTE:

NORIEGA BALDEON EDWIN FABIÁN

DOCENTE:

ING. NOGALES GUERRERO SANTIAGO ISRAEL

NIVEL:

SEGUNDO NIVEL_PARALELO “A”

2025-2025

PUYO-ECUADOR



UEA

UNIVERSIDAD

ESTATAL AMAZÓNICA

1. CÓDIGO DE GITHUB

https://github.com/EDFANOBA19/PROGRAMACION_ORIENTADA_A_OBJETOS.git

2. REPOSITORIO DE GITHUB

github.com/EDFANOBA19/PROGRAMACION_ORIENTADA_A_OBJETOS

PROGRAMACION_ORIENTADA_A_OBJETOS Public

main 1 Branch 0 Tags

Go to file Add file Code

EDFANOBA Implementación completa del Sistema de Gestión de Inventarios mejorado bc40f01 · 8 minutes ago 24 Commits

.idea	feat: Añadir clase Cliente con atributos y método para mostr...	2 months ago
CAPTURAS SEMANA 8	Implementación completa del sistema de gestión de inventa...	last week
EJERCICIO EN CLASE SEMANA 07	Implementación de ejercicios de POO en Python:	last month
SEMANA_10_POO_Sistema de Gestión de Inve...	Implementación completa del Sistema de Gestión de Inven...	8 minutes ago
SEMANA_2_POO_PILARES DE PROGRAMACIO...	Add files via upload	2 months ago
SEMANA_3_POO_Comparación de Programaci...	Add files via upload	2 months ago
SEMANA_4_POO_EJEMPLOMUNDOREAL	Merge remote-tracking branch 'origin/main'	last month
SEMANA_5_POO_Tipos de datos, Identificadores	Add files via upload	2 months ago
SEMANA_6_POO_Clases, objetos, herencia, enc...	Add files via upload	last month
SEMANA_7_POO_Constructores y Destruidores	Merge remote-tracking branch 'origin/main'	last month
SEMANA_9_POO_Sistema de Gestión de Inven...	Implementación completa del sistema de gestión de inventa...	last week
Dashboard.py	Implementación completa del sistema de gestión de inventa...	last week
README.md	Imolementación completa del sistema de gestión de inventa...	last week

About

No description, website, or topics provided.

Readme Activity 0 stars 0 watching 0 forks

Releases

No releases published
[Create a new release](#)

Packages

No packages published
[Publish your first package](#)

Contributors 2

EDFANOBA EDFANOBA19

github.com/EDFANOBA19/PROGRAMACION_ORIENTADA_A_OBJETOS/tree/main/SEMANA_10_POO_Sistema%20de%20Gestión%20de%20Inventarios%20Mejorado

EDFANOBA19 / PROGRAMACION_ORIENTADA_A_OBJETOS

Type to search

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

Files

main

Go to file

SEMANA_10_POO_Sistema de Ge...

PROGRAMACION_ORIENTADA_A_OBJETOS / SEMANA_10_POO_Sistema de Gestión de Inventarios Mejorado

Add file

EDFANOBA Implementación completa del Sistema de Gestión de Inventarios mejorado bc40f01 · 9 minutes ago History

Name	Last commit message	Last commit date
..		
Gestion de inventario.py	Implementación completa del Sistema de Gestión de Inventarios mejorado	9 minutes ago
inventario.csv	Implementación completa del Sistema de Gestión de Inventarios mejorado	9 minutes ago
inventario.py	Implementación completa del Sistema de Gestión de Inventarios mejorado	9 minutes ago
inventario.txt	Implementación completa del Sistema de Gestión de Inventarios mejorado	9 minutes ago
menu.py	Implementación completa del Sistema de Gestión de Inventarios mejorado	9 minutes ago
producto.py	Implementación completa del Sistema de Gestión de Inventarios mejorado	9 minutes ago

2.1 CÓDIGO

www.uea.edu.ec

Km. 2. 1/2 vía Puyo a Tena (Paso Lateral)

032892-118 / 032892-188 032892-098 / 032896-188 032896-476

#UEAesExcelencia



```
from inventario import Inventario
from producto import Producto

def mostrar_menu():
    print("\n--- Sistema de Gestión de Inventarios ---")
    print("1. Añadir nuevo producto")
    print("2. Eliminar producto por ID")
    print("3. Actualizar cantidad o precio de un producto")
    print("4. Buscar producto(s) por nombre")
    print("5. Mostrar todos los productos")
    print("6. Salir")

def pedir_entero(mensaje, permitir_cero=False):
    while True:
        try:
            valor = int(input(mensaje))
```

2.1.1 Almacenamiento de Inventarios en Archivos:

Modificar la clase *Inventario* para que, al añadir, actualizar, o eliminar productos, estas modificaciones se reflejen en un archivo de texto (por ejemplo, *inventario.txt*).

```
from producto import Producto
import os

class Inventario:
    """ 2 usages new """
    archivo = 'inventario.txt'

    def __init__(self):
        """ new """
        self.productos = []
        try:
            self.cargar_desde_archivo()
            print("Inventario cargado desde archivo.")
        except FileNotFoundError:
            print(f"Archivo '{self.archivo}' no encontrado. Creando uno nuevo.")
            with open(self.archivo, 'w', encoding='utf-8') as f:
                pass
        except PermissionError:
            print(f"No tienes permisos para acceder al archivo '{self.archivo}'.")
        except Exception as e:
            print(f"Error al cargar inventario: {e}")

    def agregar_producto(self, producto):
        """ 1 usage new """
        if self.buscar_producto_por_id(producto.get_id()) is not None:
            return False
        self.productos.append(producto)
        if self.guardar_en_archivo():
            print("Producto agregado y guardado exitosamente.")
            return True
        else:
            print("Error al guardar producto en archivo.")
            return False

    def eliminar_producto(self, id_producto):
        """ 1 usage new """
```



UEA

UNIVERSIDAD ESTATAL AMAZÓNICA

```
33     producto = self.buscar_producto_por_id(id_producto)
34     if producto:
35         self.productos.remove(producto)
36     if self.guardar_en_archivo():
37         print("Producto eliminado y guardado exitosamente.")
38         return True
39     else:
40         print("Error al guardar cambios en archivo.")
41         return False
42     return False
43
44     def actualizar_producto(self, id_producto, cantidad=None, precio=None): 1 usage new *
45     producto = self.buscar_producto_por_id(id_producto)
46     if producto:
47         if cantidad is not None:
48             producto.set_cantidad(cantidad)
49         if precio is not None:
50             producto.set_precio(precio)
51     if self.guardar_en_archivo():
52         print("Producto actualizado y guardado exitosamente.")
53         return True
54     else:
55         print("Error al guardar cambios en archivo.")
56         return False
57     return False
58
59     def buscar_producto_por_id(self, id_producto): 5 usages new *
60     for p in self.productos:
61         if p.get_id() == id_producto:
62             return p
63     return None
64
65     def buscar_productos_por_nombre(self, nombre): 1 usage new *
66     nombre = nombre.lower()
67     return [p for p in self.productos if nombre in p.get_nombre().lower()]
68
69     def mostrar_productos(self): 1 usage new *
70     return self.productos
71
72     def cargar_desde_archivo(self): 1 usage new *
73     self.productos.clear()
74     with open(self.archivo, 'r', encoding='utf-8') as f:
75         for linea in f:
76             linea = linea.strip()
77             if not linea:
78                 continue
79             partes = linea.split('|')
80             if len(partes) != 4:
81                 print(f"Línea ignorada por formato incorrecto en archivo: {linea}")
82                 continue
83             id_producto = partes[0].strip()
84             nombre = partes[1].strip()
85             try:
86                 cantidad = int(partes[2].strip())
87                 precio = float(partes[3].strip())
88             except ValueError:
89                 print(f"Línea ignorada por datos inválidos en cantidad o precio: {linea}")
90                 continue
91             self.productos.append(Producto(id_producto, nombre, cantidad, precio))
```



UEA

UNIVERSIDAD ESTATAL AMAZÓNICA

```
93 def guardar_en_archivo(self): 3 usages new *
94     try:
95         with open(self.archivo, 'w', encoding='utf-8') as f:
96             for p in self.productos:
97                 f.write(f"{p.get_id()} | {p.get_nombre()} | {p.get_cantidad()} | {p.get_precio()}\n")
98             return True
99     except PermissionError:
100         print(f"No tienes permisos para escribir en el archivo '{self.archivo}'.")
101         return False
102     except Exception as e:
103         print(f"Error al guardar en archivo: {e}")
104         return False
105
```

inventario.txt

1	P002 Naranjas 75 0.75
2	P003 Leche 30 1.25
3	P004 Pan 100 0.8
4	P005 Huevos 200 0.1
5	P001 Manzanas 250 0.25
6	P001 Manzanas 50 0.5
7	P007 CHUPETES 100 0.15
8	P008 GALLETAS 500 0.5
9	P009 KCHITOS 1000 1.0
10	P001 Manzanas 50 0.5
11	P001 Manzanas 50 0.5
12	

2.1.2. Recuperación de Inventarios desde Archivos:

Al iniciar el programa, cargar automáticamente los productos existentes en *inventario.txt* para reconstruir el inventario.

Project	producto.py	inventario.txt	Gestion de inventario.py	inventario.py	menu.py
PROGRAMACION_ORIENTADA_A_OBJETOS	1	P001 Manzanas 50 0.5			
> CAPTURAS SEMANA 8	2	P002 Naranjas 75 0.75			
> EJERCICIO EN CLASE SEMANA 07	3	P003 Leche 30 1.25			
> SEMANA_2_POO_PILARES DE PROGRAMACI	4	P004 Pan 100 0.8			
> SEMANA_3_POO_Comparación de Program	5	P005 Huevos 200 0.1			
> SEMANA_4_POO_EJEMPLUMUNDOREAL	6	P001 Manzanas 50 0.5			
> SEMANA_5_POO_Tipos de datos, Identificac	7				
> SEMANA_6_POO_Clases, objetos, herencia,					
> SEMANA_7_POO_Constructores y Destruct					
> SEMANA_9_POO_Sistema de Gestión de Inv					
> SEMANA_10_POO_Sistema de Gestión de In					
Gestion de inventario.py					
inventario.py					
inventario.txt					
menu.py					
producto.py					

2.1.3. Manejo de Excepciones:



UEA

UNIVERSIDAD ESTATAL AMAZÓNICA

Implementar manejo de excepciones para capturar y tratar adecuadamente posibles errores durante la manipulación de archivos, como **FileNotFoundError** y **PermissionError**.

```
producto.py  inventario.txt  Gestion de inventario.py  inventario.py  menu.py

1  class Inventario: 1usage new *
2      def __init__(self, archivo='inventario.txt'): new *
3          self.archivo = archivo
4          self.productos = []
5          self.cargar_desde_archivo()
6
7      def cargar_desde_archivo(self): 1usage new *
8          try:
9              with open(self.archivo, 'r', encoding='utf-8') as f:
10                 for linea in f:
11                     # Aquí deberías parsear cada línea para crear objetos producto, simplifico
12                     self.productos.append(linea.strip())
13                 print(f"Archivo '{self.archivo}' cargado exitosamente.")
14             except FileNotFoundError:
15                 print(f"Archivo '{self.archivo}' no encontrado. Creando archivo nuevo.")
16                 # Crear archivo vacío
17                 with open(self.archivo, 'w', encoding='utf-8') as f:
18                     pass
19             except PermissionError:
20                 print(f"Sin permiso para leer el archivo '{self.archivo}'.")
21
22      def agregar_producto(self, producto): 1usage new *
23          self.productos.append(producto)
24          try:
25              with open(self.archivo, 'a', encoding='utf-8') as f:
26                  f.write(producto + '\n')
27              print("Producto agregado y guardado correctamente.")
28          except PermissionError:
29              print(f"Sin permiso para escribir en el archivo '{self.archivo}'.")
30
31  # Uso
32  inv = Inventario()
```

Asegurar que el programa maneje casos en los que el archivo de inventario no exista, creándolo si es necesario.

```
Run  menu (1) x  Gestion de inventario x

"C:\Users\hp\PycharmProjects\SEMANA_2_POO_PILARES DE PROGRAMACION ORIENTADA A OBJETOS\.venv\Scripts\python.exe" "C:\Users\hp\PycharmP
Archivo 'inventario.csv' no encontrado. Creando archivo nuevo.
Producto agregado y guardado correctamente.

Process finished with exit code 0
```



```
1 | P001 | Manzanas | 50 | 0.5
2 | 
```

2.1.4. Modificaciones a la Interfaz de Usuario en la Consola:

Actualizar la interfaz de usuario para notificar al usuario sobre el éxito o fallo de operaciones de archivo (por ejemplo, notificar al usuario cuando un producto se añade exitosamente al archivo de inventario)

```
"C:\Users\hp\PycharmProjects\SEMANA_2_POO_PILARES DE PROGRAMACION ORIENT
Archivo 'inventario.txt' cargado exitosamente.
Producto agregado y guardado correctamente.
Process finished with exit code 0
```

```
6. Salir
Selecciona una opción: 1

--- Añadir nuevo producto ---
Ingresa ID único: P007
Nombre del producto: CHUPETES
Cantidad: 100
Precio: $0.15
Producto agregado y guardado exitosamente.
```



producto.py inventario.txt Gestion de i

1	P001		Manzanas		50		0.5
2	P002		Naranjas		75		0.75
3	P003		Leche		30		1.25
4	P004		Pan		100		0.8
5	P005		Huevos		200		0.1
6	P001		Manzanas		50		0.5
7	P001		Manzanas		50		0.5
8	P007		CHUPETES		100		0.15
9	P008		GALLETAS		500		0.5
10	P009		KCHITOS		1000		1.0
11							

```
Selecciona una opción: 2

--- Eliminar producto ---
Ingresa el ID del producto a eliminar: P001
Producto eliminado y guardado exitosamente.
```

Run menu (1) x inventario x

```
Selecciona una opción: 3

--- Actualizar producto ---
ID del producto a actualizar: P001
Producto actual: ID: P001 | Nombre: Manzanas | Cantidad: 50 | Precio: $0.50
Deja en blanco el campo que no deseas actualizar.
Nueva cantidad: 250
Nuevo precio: $0.25
Producto actualizado y guardado exitosamente.
```




```
Run  menu (1) x  inventario x

5. Mostrar todos los productos
6. Salir
Selecciona una opción: 4

--- Buscar productos por nombre ---
Ingresa parte o todo el nombre: MANZANAS
Se encontraron 2 producto(s):
ID: P001 | Nombre: Manzanas | Cantidad: 250 | Precio: $0.25
ID: P001 | Nombre: Manzanas | Cantidad: 50 | Precio: $0.50

Run  menu (1) x  inventario x

Selecciona una opción: 5

--- Mostrar todos los productos ---
ID: P002 | Nombre: Naranjas | Cantidad: 75 | Precio: $0.75
ID: P003 | Nombre: Leche | Cantidad: 30 | Precio: $1.25
ID: P004 | Nombre: Pan | Cantidad: 100 | Precio: $0.80
ID: P005 | Nombre: Huevos | Cantidad: 200 | Precio: $0.10
ID: P001 | Nombre: Manzanas | Cantidad: 250 | Precio: $0.25
ID: P001 | Nombre: Manzanas | Cantidad: 50 | Precio: $0.50
ID: P007 | Nombre: CHUPETES | Cantidad: 100 | Precio: $0.15
ID: P008 | Nombre: GALLETAS | Cantidad: 500 | Precio: $0.50
ID: P009 | Nombre: KCHITOS | Cantidad: 1000 | Precio: $1.00
```