

Participante: Mariel Gajardo

Mi Participación en la Actividad

Durante el desarrollo de este sistema de manejo de inventario, mi rol principal fue crear todo el código, implementando la lógica y comprendiendo las funcionalidades de lectura, escritura y manipulación de archivos CSV en Python. Trabajé específicamente en la creación de métodos para gestionar un inventario de productos con los componentes nombrados a continuación.

1. Módulo CSV de Python

Aprendí a utilizar el módulo csv que proporciona herramientas especializadas para trabajar con archivos de valores separados por comas.

2. Clase `csv.writer()`

Comprendí cómo usar `csv.writer()` para escribir datos de forma estructurada:

`writerow()`: Escribe una sola fila al archivo

`writerows()`: Escribe múltiples filas de una vez

El parámetro `newline=""` es importante para evitar líneas en blanco adicionales en Windows

3. Clase `csv.reader()`

Aprendí a leer archivos CSV de manera eficiente:

`csv.reader()` devuelve un iterador que lee línea por línea

`next()` permite saltar encabezados o leer la primera fila específicamente

4. Modos de Apertura de Archivos

Aprendí los diferentes modos para abrir archivos:

'r' (read): Solo lectura, el archivo debe existir

'w' (write): Escritura que sobrescribe todo el contenido existente

'a' (append): Añadir contenido al final del archivo sin borrar lo existente

'r+', 'w+', 'a': Modos combinados de lectura y escritura

5. Encoding UTF-8

Aprendí la importancia de especificar `encoding='utf-8'` para manejar correctamente caracteres especiales y acentos en español, evitando errores de codificación.

6. With open

Comprendí el uso de `with open()` que garantiza:

Cierre automático del archivo al terminar

7. Manejo de Excepciones en Archivos

FileNotFoundError: Cuando el archivo no existe

Exception general: Para capturar otros errores inesperados

Mensajes informativos para el usuario sobre qué salió mal

Además, el desarrollo de este código me permitió aplicar los principios de programación orientada a objetos de manera práctica, organizando las funcionalidades del sistema en una clase cohesiva que contiene tanto los datos (como los encabezados por defecto) como los métodos de manipulación de archivos. Esta estructura orientada a objetos facilita la integración fluida con las herramientas del módulo csv, como `csv.reader()` y `csv.writer()`, creando un sistema más organizado y mantenible. La experiencia me ha demostrado la importancia de esta metodología para proyectos de mayor complejidad, aunque reconozco que necesito continuar fortaleciendo estos conceptos en futuros desarrollos para dominar completamente este paradigma de programación.