
MANUAL TECNICO - PRACTICA NO. 1 -

201212891 – Edgar Rolando Ramírez López

Resumen

El programa en Python desarrollado es un sistema de gestión de inventario que permite cargar un inventario inicial desde un archivo de texto en formato .inv, realizar movimientos de stock mediante instrucciones en un archivo .mov, generar un informe de inventario y realizar acciones en un menú interactivo. El usuario puede elegir entre cargar inventario, cargar instrucciones de movimientos, generar informe de inventario y salir del programa. Las funciones cargarInv, elegirArchivo, elegirArchivoMovs, informeInv y actualizar_stock desempeñan roles clave en la funcionalidad del programa. Al ejecutarse, el programa interactúa con el usuario para ejecutar las opciones seleccionadas, gestionando productos y proporcionando un informe detallado de inventario en un archivo de salida.

Palabras clave

Python, funciones, clases, listas, librerías.

Abstract

The developed Python program is an inventory management system that allows loading an initial inventory from a .inv format text file, performing stock movements through instructions in a .mov file, generating an inventory report, and carrying out actions in an interactive menu. The user can choose to load inventory, load movement instructions, generate an inventory report, and exit the program. The functions loadInv, chooseFile, chooseMovsFile, generateReport, and update_stock play key roles in the program's functionality. When executed, the program interacts with the user to execute selected options, manage products, and provide a detailed inventory report in an output file.

Keywords

Python, functions, class, lists, libraries.

Introducción

El sistema de inventario es un programa en Python diseñado para gestionar un inventario de productos. Permite cargar un inventario inicial desde un archivo con extensión “. inv”, realizar movimientos de stock (agregar o vender productos) mediante instrucciones en un archivo con extensión “.mov”, y por último generar un informe de inventario y realizar acciones básicas del sistema.

Desarrollo del tema

En el desarrollo de la práctica se utilizó el lenguaje Python 3.11.1, el IDE utilizado fue Visual Studio Code y también se subió la practica a GitHub. El paradigma utilizado en la práctica fue la programación con orientación a objetos. Las librerías utilizadas fueron OS.

a. Python

Python es un lenguaje de programación interpretado multiparadigma, multiplataforma y de código abierto. Diseñado para facilitar la programación orientada a objetos, permite trabajar con un gran número de estructuras de datos como diccionarios, listas, tuplas y conjuntos. Es compatible con muchos sistemas operativos, como Windows, Mac OS, Linux, y también se puede usar para el desarrollo web, scripting y desarrollo de aplicaciones. Ofrece una extensa biblioteca estándar con muchas utilidades prácticas. Está aumentando su presencia en la industria de la tecnología y está ganando popularidad por su facilidad de uso y simplicidad.

b. Visual Code Studio (VSCODE)

Un IDE es un entorno de desarrollo integrado, es una aplicación de software que ayuda a los programadores a desarrollar código de software de manera eficiente. Visual Studio Code es un editor de código fuente de código abierto y gratuito desarrollado por Microsoft para MacOS, Linux y Windows. Incluye soporte para desarrollo web, lenguajes de edición de línea de comandos y lenguajes de scripting modernos, así como lenguajes de programación como HTML, CSS, CoffeeScript, PHP, Ruby y Python. Ofrece características como resaltado de sintaxis, completado inteligente, intenciones y muchos más. La eficiencia y la facilidad de uso de Visual Studio Code han hecho de él una herramienta de desarrollo muy utilizada en todo el mundo.

c. GitHub

GitHub es una plataforma de alojamiento web que ayuda a los desarrolladores a almacenar y mantener el código fuente para proyectos de software. Utiliza el sistema de control de versiones Git para permitir a los usuarios colaborar en el desarrollo de proyectos, realizar revisión de código, realizar tareas de seguimiento de errores y realizar pruebas. También ofrece una variedad de herramientas de integración de terceros, navegación web y soporte para proyectos públicos y privados.

d. Librería OS

La librería OS en Python es una colección de herramientas y funciones para interactuar de forma programática con el sistema operativo. Proporciona funcionalidad para acceder al sistema de archivos,

búsqueda de archivos, ejecutar procesos, manejar los permisos, administrar memorias, etc.

Estructura del Programa:

El programa está compuesto por varias funciones que se encargan de diferentes tareas. Aquí se presenta una descripción detallada de cada función:

- **cargarInv(nombreArchivo):** Lee un archivo de inventario (formato. inv) y crea una lista de objetos Producto con los datos del archivo.
- **elegirArchivo():** Solicita al usuario ingresar el nombre o ruta de un archivo, verifica que tenga extensión .inv y exista en el sistema.
- **elegirArchivoMovs():** Similar a elegirArchivo(), pero para archivos de instrucciones de movimientos (formato .mov).
- **informeInv(productos):** Genera un informe de inventario en formato de tabla y lo guarda en un archivo llamado "informe_inventario.txt". Solo incluye productos con cantidades mayores a 0.
- **actualizar_stock(productos, archivo_actualizar):** Lee un archivo de instrucciones de movimientos (.mov) y actualiza el stock de productos en base a las operaciones de agregar_stock y vender_producto.
- **menuPrincipal():** Muestra un menú con las opciones disponibles y permite al usuario seleccionar una.

Flujo del Programa:

1. Al ejecutar el programa (`__name__ == "__main__"`), se muestra el menú principal.
2. El usuario elige una opción (1, 2, 3 o 0 para salir) y se ejecuta el código correspondiente de acuerdo a la opción elegida.

```
=====
Practica 1 - Lenguajes formales de Programacion
=====
# Sistema de Inventario:

1.- Cargar Inventario inicial
2.- Cargar Instrucciones de movimiento
3.- Crear Informe de Inventario
0.- Salir

.....
Ingrese una opcion:
```

Figura 1. Menú Principal.

Fuente: terminal VScode.

3. Si la opción es cargar inventario (1), se solicita al usuario ingresar el nombre del archivo de inventario y se crea una lista de productos (nuevos_productos).

```
=====CARGA DE ARCHIVO=====

Escribe el nombre o ruta del archivo para cargar: inventario.inv

*.*.*.*-Archivo cargado exitosamente.*.*.*.*

Presione enter para continuar...
```

Figura 2. Carga de archivo inv.

Fuente: terminal VScode.

4. Si la opción es cargar instrucciones de movimientos (2), se verifica que haya productos cargados en nuevos_productos. Luego, se solicita al usuario el nombre del archivo de instrucciones y se actualiza el stock de productos. Si hay errores, por ejemplo, si el producto no existe o la cantidad que se quiere vender es mayor al stock, entonces marcara un mensaje de error.

```
=====CARGA DE ARCHIVO MOVIMIENTOS=====
Escribe el nombre o ruta del archivo para cargar: movimientos.mov
*_*_*_*_Instrucciones cargadas exitosamente.*_*_*_*_
```

Figura 3. Carga de archivo movimientos.

Fuente: terminal VSCode.

```
=====ERROR===== No se encontró el producto Tomates en BodegaB =====
=====ERROR===== No se encontró el producto Queso en BodegaG =====
=====ERROR===== No hay suficiente stock de Plátanos en BodegaC para vender 6 =====
```

Figura 3.a. Mensaje de errores.

Fuente: terminal VSCode.

5. Si la opción es generar el informe de inventario (3), se verifica que haya productos cargados en nuevos_productos. Luego, se genera y guarda el informe en un archivo.

Informe de Inventario:				
Producto	Cantidad	Precio Unitario	Valor Total	Ubicacion
Tomates	100	\$1.00	\$100.00	BodegaA
Papas	175	\$3.25	\$568.75	BodegaC
Plátanos	75	\$1.75	\$131.25	BodegaD
Queso	35	\$20.50	\$717.50	BodegaE
Helado	172	\$6.50	\$1118.00	BodegaF
Arándanos	735	\$0.50	\$367.50	BodegaG
Tomates	1	\$1.00	\$1.00	BodegaG
Manzanas	150	\$3.00	\$450.00	BodegaF
Papas	101	\$3.25	\$328.25	BodegaE
Queso	60	\$20.50	\$1230.00	BodegaD
Helado	80	\$6.50	\$520.00	BodegaB
Arándanos	155	\$0.50	\$77.50	BodegaA

Figura 4. Informe final.

Fuente: Notepad++.

6. El programa vuelve al menú principal después de cada opción ejecutada.
7. Si el usuario elige la opción 0, el programa finaliza.

Instrucciones de Uso:

- Ejecute el programa desde la línea de comandos o su entorno de desarrollo favorito.
- Siga las indicaciones del menú para cargar inventario, cargar instrucciones de movimientos, generar informe de inventario o salir del programa.
- Siga las indicaciones para ingresar nombres de archivos y opciones.
- Los resultados y mensajes de error se mostrarán en la consola.

<https://j2logo.com/python/tutorial/programacion-orientada-a-objetos/>

Conclusiones

El sistema de inventario es una herramienta útil para gestionar productos y movimientos de stock. Este manual técnico proporciona una visión general del programa, sus funciones y cómo utilizarlo. Al seguir las instrucciones y opciones del menú, los usuarios pueden gestionar su inventario de manera efectiva.

Referencias bibliográficas

Welcome to. (2023, 15 febrero). Python.org. <https://www.python.org/>

Documentation for Visual Studio Code. (2021, 3 noviembre). <https://code.visualstudio.com/docs>

Hello World. (s. f.). GitHub Docs. <https://docs.github.com/en/get-started/quickstart/hello-world>

J. (2022, 16 enero). *Funciones en Python: Definición de función y para qué se utilizan*. J2LOGO. <https://j2logo.com/python/tutorial/funciones-en-python/>

Bustamante, S. J. (2021, 21 febrero). *Guía de funciones de Python con ejemplos*. freeCodeCamp.org. <https://www.freecodecamp.org/espanol/news/guia-de-funciones-de-python-con-ejemplos/>

Programación orientada a objetos. (s. f.). <https://www.ibm.com/docs/es/spss-modeler/saas?topic=language-object-oriented-programming>

J. (2022b, enero 16). *Programación orientada a objetos (POO) en Python*. J2LOGO.