







Manual Tecnico

- Requerimientos

Programado en:

- Python 3.10.2
- Visual Studio Code
- Sistema operativo de 64 bits

| Nombre | Estado | 4% CPU | 25% Memoria | 0% Disco | 0% Red | 3% GPU | Motor de la GPU |
|------------------|---|-----------|----------------|-------------|-----------|-----------|-----------------|
| Aplicaciones (4) | | | | | | | |
| > |  Administrador de tareas | 0.1% | 26.6 MB | 0 MB/s | 0 Mbps | 0% | |
| > |  Bloc de notas | 0% | 1.3 MB | 0 MB/s | 0 Mbps | 0% | |
| > |  Brave Browser (23) | 0% | 1,153.9 MB | 0.1 MB/s | 0 Mbps | 0% | GPU 0 - 3D |
| > |  Visual Studio Code (26) | 0% | 712.7 MB | 0 MB/s | 0 Mbps | 0% | GPU 0 - 3D |

The screenshot shows a Windows desktop with two windows. The background window is a code editor (likely VS Code) showing a Python script named 'menu.py'. The script is a menu-driven program for managing inventory. It has three main options: 1. 'Cargar instrucciones de movimiento' (Load movement instructions), 2. 'Crear informe de inventario' (Create inventory report), and 3. 'Salir' (Exit). The code uses loops and conditional statements to handle these options. The foreground window is the Windows Task Manager, showing the 'Procesos' (Processes) tab. It lists various applications running, including 'Administrador de tareas' (Task Manager), 'Bloc de notas' (Notepad), 'Brave Browser', and 'Visual Studio Code'. The 'Visual Studio Code' process is highlighted, showing it is using 0.3% CPU and 743.2 MB of memory.

- Código Principal

```
EXPLORER    ...  menu.py M X  Producto_mock.py  Informe.py  Producto.py M
└─ LFP_S2_2023_PRACTICA_20...
  > __pycache__
  ▢ archivo.inv
  ◆ Informe.py
  ◆ menu.py M
  ◆ movimiento.py
  ◆ movimientos.mov
  ◆ Producto_mock.py
  ◆ Producto.py M
  ① README.md
  ▢ resultado_202200007...

menu.py > mostrar_menu
1  from Producto import cargar_inventario
2  from movimiento import cargar_movimientos
3  from Informe import archivo_texto
4
5  def mostrar_menu():
6      lugar = input("Ingrese la dirección del archivo de inventario: ")
7      inventario_inicial = cargar_inventario(lugar)
8      mov_archivo = input("Ingrese la dirección del archivo de movimientos: ")
9
10     while True:
11         print("\n-----")
12         print("Práctica 1 - Lenguajes formales y de programación")
13         print("-----")
14         print("SISTEMA DE INVENTARIO:")
15         print("1. Cargar inventario inicial")
16         print("2. Cargar instrucciones de movimiento")
17         print("3. Crear informe de inventario")
18         print("4. Salir")
19         opcion = input("Ingrese una opción: ")
20
21     if opcion == "1":
22         if len(inventario_inicial) == 0:
23             print("El inventario está vacío.")
24         else:
25             for producto in inventario_inicial:
26                 print("-----")
27                 print("Nombre:", producto.nombre, "Cantidad:", producto.cantidad, "Precio Unitario:", producto.precio_unitario, "Ubicación:", producto.ubicacion)
```

```

28
29 elif opcion == "2":
30     if mov_archivo:
31         print("Cargando instrucciones de movimiento")
32         cargar_movimientos(mov_archivo, inventario_inicial)
33         print("-----INVENTARIO ACTUALIZADO CON LOS MOVIMIENTOS-----")
34         for producto in inventario_inicial:
35             print("-----")
36             print("Nombre:", producto.nombre, "Cantidad:", producto.cantidad, "Precio Unitario:", producto.precio_unitario, "Ubicación:", pr
37         else:
38             print("ERROR - Cargue primero el inventario inicial antes de las instrucciones de movimiento.")
39
40     elif opcion == "3":
41         print("Creando informe de inventario")
42         print("INFORME DE INVENTARIO:")
43         print("INFORME DESCARGADO CORRECTAMENTE")
44         archivo_texto('resultado_202200007.txt', inventario_inicial)
45     elif opcion == "4":
46         print("Saliendo del sistema")
47         break
48     else:
49         print("ERROR - OPCION INVALIDA")
50
51 , mostrar_menu()

```

Descripción:

El archivo menu.py es el lugar en donde todas las clases y funciones se unen para poder realizar las acciones deseadas por el usuario en relación al inventario.

Al inicio del programa se le solicita al usuario la ubicación del archivo .inv y el archivo .mov con los que se va a estar trabajando posteriormente. Luego se despliega un menú en donde el usuario debe indicarle al programa que desea hacer a través de los números indicados correspondientes a la acción.

La primera opción le permite al usuario cargar y visualizar de una manera ordenada y sencilla el inventario actual. La segunda opción, es la encargada de registrar todas las ventas o ingresos de ciertos productos y posteriormente mostrar de manera ordenada las nuevas cantidades de productos disponibles. La tercera opción le permite al usuario descargar un reporte del inventario actual en existencia, este reporte es descargado bajo el nombre de "resultado_202200007.txt". La última opción es que le permite al usuario salir del sistema.

-Punto de vista del usuario

```

Ingrese la dirección del archivo de inventario: C:\Users\natalia\Documents\4sem\lfp\LFP_S
Ingrese la dirección del archivo de movimientos: C:\Users\natalia\Documents\4sem\lfp\LFP_

-----
Práctica 1 - Lenguajes formales y de programación
-----
SISTEMA DE INVENTARIO:
1. Cargar inventario inicial
2. Cargar instrucciones de movimiento
3. Crear informe de inventario
4. Salir
Ingrese una opción: 1
-----
Nombre: Manzanas Cantidad: 100 Precio Unitario: 2.5 Ubicación: BodegaA
-----
Nombre: Peras Cantidad: 50 Precio Unitario: 3.0 Ubicación: BodegaB
-----
Nombre: Platanos Cantidad: 75 Precio Unitario: 1.75 Ubicación: BodegaC

```

Práctica 1 - Lenguajes formales y de programación

SISTEMA DE INVENTARIO:

1. Cargar inventario inicial
2. Cargar instrucciones de movimiento
3. Crear informe de inventario
4. Salir

Ingrese una opción: 2

Cargando instrucciones de movimiento

No se encontró el producto Peras en la bodega: BodegaB

-----INVENTARIO ACTUALIZADO CON LOS MOVIMIENTOS-----

Nombre: Manzanas Cantidad: 150 Precio Unitario: 2.5 Ubicación: BodegaA

Nombre: Peras Cantidad: 30 Precio Unitario: 3.0 Ubicación: BodegaB

Nombre: Platanos Cantidad: 75 Precio Unitario: 1.75 Ubicación: BodegaC

Práctica 1 - Lenguajes formales y de programación

1. Cargar inventario inicial
2. Cargar instrucciones de movimiento
3. Crear informe de inventario
4. Salir

Ingrese una opción: 3

Creando informe de inventario

INFORME DE INVENTARIO:

INFORME DESCARGADO CORRECTAMENTE

Práctica 1 - Lenguajes formales y de programación

SISTEMA DE INVENTARIO:

1. Cargar inventario inicial
2. Cargar instrucciones de movimiento
3. Crear informe de inventario
4. Salir

Ingrese una opción: 4

Saliendo del sistema

Resultado:



resultado_202200007: Bloc de notas

Archivo Edición Formato Ver Ayuda

| PRODUCTO | CANTIDAD | PRECIO UNITARIO | TOTAL | UBICACION |
|----------|----------|-----------------|----------|-----------|
| Manzanas | 150 | \$2.5 | \$375.0 | BodegaA |
| Peras | 30 | \$3.0 | \$90.0 | BodegaB |
| Platanos | 75 | \$1.75 | \$131.25 | BodegaC |