

# INFORME TÉCNICO

---

PROYECTO 1 INTRODUCCIÓN A PYTHON

Amairani Guillen Chagala  
EMTECH | SANTANDER

## Índice

<b>Introducción.....</b>	<b>2</b>
<b>Definición del código .....</b>	<b>3</b>
<b>Solución al problema.....</b>	<b>6</b>
<b>Conclusión .....</b>	<b>7</b>

## Introducción

LifeStore una tienda virtual que maneja una amplia gama de artículos, recientemente, la Gerencia de ventas, se percató que la empresa tiene una importante acumulación de inventario. Asimismo, se ha identificado una reducción en las búsquedas de un grupo importante de productos, lo que ha redundado en una disminución sustancial de sus ventas del último trimestre.

Derivado de la situación, la Gerencia de Ventas solicita que se realice un análisis de la rotación de productos identificando los siguientes elementos

- Productos más vendidos y productos rezagados a partir del análisis de las categorías con menores ventas y categorías con menores búsquedas.
- Productos por reseña en el servicio a partir del análisis de categorías con mayores ventas y categorías con mayores búsquedas.
- Sugerir una estrategia de productos a retirar del mercado, así como sugerencia de cómo reducir la acumulación de inventario considerando los datos de ingresos y ventas mensuales.

## Definición del código

Se diseño una interfaz sencilla donde se debe ingresar el usuario y contraseña de las personas autorizadas para poder visualizar los diferentes.

Los administradores o usuarios registrados son:

Usuario: Admin, Contraseña: Admin

Usuario: Amairani, Contraseña: 12345

Este código demuestra el uso de dos variables o datos de entrada (usuario y contraseña), y variables banderas para conceder acceso y también detectar si los datos introducidos por el usuario satisfacen los permisos que permitan su acceso.

```
#Interfaz de acceso

admins = [['Admin','Admin'], ['Amairani','12345']]

usuario = input('Ingresa el usuario: ')
contraseña = input('Ingresa la contraseña: ')

es_admin = 0
contador = 1

while es_admin != 1 and contador <= 2:
    for admin in admins:
        if usuario == admin[0] and contraseña == admin[1]:
            print("¡Bienvenido!")
            es_admin = 1

    if es_admin == 0:
        print("Datos incorrectos")
        usuario = input("Ingresa el usuario: ")
        contraseña = input("Ingresa la contraseña: ")
        contador += 1

if es_admin == 1:
    print("Tienda Virtual LifeStore")
else:
    print("No existe usuario")
```

Se genero un menú para poder seleccionar el tipo de reporte al que se quiere acceder o visualizar. Se implementa el uso de variables bandera para poder permitir el acceso a alguno de los loops internos o para seguir pidiendo por una opción válida o bien, dar por terminada la sesión y la ejecución del programa.

```

#opciones de reportes disponibles
opcion_valida = 0

while opcion_valida != 1:
    print('Selección el reporte que desee visualizar: \n1.- Productos más vendidos y productos rezagados. \n2.- Productos por reseña en el servicio. \n3.- Ingresos ventas \n4.- Finalizar sesión.\n')

    opcion_seleccionada = input('Seleccione una opcion: ')

    #Mas vendidos y rezagados
    if opcion_seleccionada == '1':
        print('Seleccionó reporte de productos más vendidos y productos rezagados.\nHaga una selección del reporte que desea consultar: \n1.- 50 productos con mayores ventas.\n2.- 50 productos con mayores búsquedas.\n3.- 10 productos con menores ventas por categoría. \n4.- 10 productos con menores búsquedas por categoría. \n5.- Regresar al menú anterior')
        opcion_valida = 1

        subopcion1_seleccionada = input('Seleccione una opcion: ')
        subopcion1_valida = 0

        while subopcion1_valida != 1:
            #lista 50 productos con mayores ventas
            if subopcion1_seleccionada == '1':

```

Se implemento un conjunto de estrategias básicas que permitieran las iteraciones, para el desarrollo de la herramienta de análisis, mediante listas, operadores lógicos y condicionales para la clasificación de la información.

La premisa básica es el poder iterar sobre los datos proporcionados de la tienda virtual LifeStore existentes.

```

ventas_totales = 0
lista_ventas_ind = [] # [[id,ventas]]
for producto in lifestore_products:
    ventas_individual = 0
    for venta in lifestore_sales:
        if producto[0] == venta[1]:
            ventas_individual += 1
            ventas_totales += 1
    registro_individual =
[producto[0],producto[1],ventas_individual]
    lista_ventas_ind.append(registro_individual)

lista_ventas_max = list(lista_ventas_ind)
ventas_ordenadas_max = []

while lista_ventas_max:
    maximo = lista_ventas_max[0][2]
    registro_actual = lista_ventas_max [0]
    for registro in lista_ventas_max:
        if registro[2] > maximo:
            maximo = registro[2]
            registro_actual = registro
    ventas_ordenadas_max.append(registro_actual)
    lista_ventas_max.remove(registro_actual)

for i in range(50) :
    print(f'El producto número {i+1} en ventas: ',
ventas_ordenadas_max[i][1],'se ha vendido en total
',ventas_ordenadas_max[i][2], 'veces')
    opcion_valida = 0
    break
    subopcion1_valida = 1

```

## Solución al problema

Se realizó un análisis a partir de los datos existentes en el archivo `lifestore-file.py`, el cual contiene los registros de las compras, búsquedas y productos manejados por la tienda, considerando el desarrollo de un sistema de análisis, en el que mediante un login de usuario-administrador se muestre un reporte mensual que especifique los puntos señalados en la consigna, específicamente:

- Productos más vendidos y productos rezagados:
  - Generar un listado de los 50 productos con mayores ventas y uno con los 100 productos con mayores búsquedas.
  - Por categoría, generar un listado con los 50 productos con menores ventas y uno con los 100 productos con menores búsquedas.
- Productos por reseña en el servicio
  - Mostrar dos listados de 20 productos cada uno, un listado para productos con las mejores reseñas y otro para las peores, considerando los productos con devolución.
- Total, de ingresos y ventas promedio mensuales, total anual y meses con más ventas al año

Repositorio GitHub

<https://github.com/AmairaniGC/EMTECH--Proyecto1.git>

## Conclusión

Para el análisis y solución del problema planteado de la tienda virtual LifeStore fue necesario emplear y poner en práctica las bases de programación en Python para análisis y clasificación de datos mediante la creación de programas de entrada de usuario y validaciones, uso y definición de variables y listas, operadores lógicos y condicionales para la clasificación de información.