



---

# PROYECTO 1: LIFESTORE

---

Raúl Alvarado Franco



PROYECTO COMO PARTE DEL CURSO DE EMTECH

Índice

Introducción .....2

Definición de código.....2

Solución al problema.....5

Conclusión.....6

## Introducción

En el siguiente trabajo se realizará un análisis de los datos ficticios de una compañía de electrónicos utilizando las herramientas que nos ofrece Python para el manejo de listas. El objetivo de dicho análisis será presentar de manera clara y concisa las estadísticas de los productos que ofrece la compañía, haciendo especial énfasis en los datos de venta, búsqueda de productos y devoluciones de estos.

Para realizar el análisis se utilizaron preguntas booleanas, haciendo uso del método if en Python, dentro de un ciclo for para poder filtrar toda la información de las listas. Se ocuparon dos ciclos for, el primero para recorrer los datos de las ventas y el segundo para los datos de las búsquedas.

Los datos resumidos y ordenados que se obtienen del análisis permiten obtener nuevas perspectivas acerca de la importancia que tiene cada producto para que la compañía puede tener un aumento en sus ventas y evitar tener en almacén productos que no son solicitados por los clientes.

## Definición de código

Para poder empezar el análisis primero se importaron los datos y dos paqueterías de Python necesarias para diferentes métodos más adelante.

```
import lifestore_file as files #Importamos los datos
import sys
import matplotlib.pyplot as plt
```

Figura 1: Importación del programa donde se definieron los datos, de la paquetería sys y matplotlib.pyplot.

Se definieron las principales variables que se van a utilizar a lo largo del programa, los datos de las ventas se guardaron en la variable “sales”, las búsquedas se guardaron en la variable “searches” y los datos de los productos en la variable “products”. Al mismo tiempo se definió la variable “usuario”, necesaria para el método de log in y se define el número de intentos para poder introducir el usuario correcto.

```
"Se asignan los datos a variables"
searches=files.lifestore_searches
sales=files.lifestore_sales
products=files.lifestore_products
"Creación de las variables para el log in"
usuario='Administrador'
intentos=3
```

Figura 2: Asignación de las variables de datos y definición del usuario y número de intentos para el método de log in.

Inicio del método de log in, se empieza por preguntar el usuario y utilizando un ciclo while se hace una pregunta booleana para poder identificar si el usuario fue introducido correctamente, si así fue el ciclo no se ejecuta, pero en caso contrario las acciones del ciclo son:

- Restar un intento

- Imprimir una leyenda para indicar que existe un error en el usuario introducido e indicar el número de intentos restantes
- Revisar si ya no quedan intentos, en dicho caso se cierra el programa y se imprime la leyenda "Usuario incorrecto, ya no hay intentos"
- Volver a solicitar el usuario

```

entrada=input('Usuario:')
"Ciclo para verificar usuario correcto"
while usuario!=entrada:
    intentos-=1
    print(f"Error, le quedan {intentos} intentos")
    if intentos==0:
        sys.exit("Usuario incorrecto, ya no hay intentos")
    entrada=input('Usuario:')

```

Figura 3: Solicitud del usuario y ciclo while para verificar si es el correcto.

Se asignan tres variables que serán utilizadas para almacenar los diferentes cálculos necesarios para el análisis:

1. "total\_producto": Se define como una lista de 96 elementos, a su vez cada elemento es una lista de 5 elementos donde el primer elemento es el id del producto y los otros cuatro elementos se establecen como cero para poder realizar la suma de las ventas, búsquedas, calificaciones y devoluciones de cada producto.
2. "meses\_v": Se define como una lista de 12 ceros donde se realizará la suma de las ventas mensuales.
3. "total\_anual": Se define como el número entero 0 para poder ir sumando cada venta que se realizó y obtener el total anual.

```

total_producto=[[i+1,0,0,0,0] for i in range(96)] #[id_product,ventas,reembolso,score,busqueda]
meses_v=[0]*12
total_anual=0

```

Figura 4: Definición de variables necesarias para almacenar los datos requeridos.

Ciclo for para poder recorrer todos los datos de la lista sales, cada elemento de la lista se asigna a la variable "ventas". Dentro del ciclo se realiza una pregunta booleana para saber si en la venta existe devolución, en caso afirmativo se suma a la cantidad de reembolsos y se suma la reseña que le dieron al producto, en caso contrario se suma la venta del producto y la reseña obtenida.

```

"Ciclo que recorre todas las ventas"
for ventas in sales:
    if ventas[4]==1:
        total_producto[ventas[1]-1][2]+=1
        total_producto[ventas[1]-1][3]+=ventas[2]
        continue
    total_producto[ventas[1]-1][1]+=1
    total_producto[ventas[1]-1][3]+=ventas[2]
    meses_v[int(ventas[3][3:5])]+=products[ventas[1]-1][2]
    total_anual+=products[ventas[1]-1][2]

```

Figura 5: Ciclo for para poder sumar la cantidad de ventas (anuales y mensuales), reembolsos y calificación de todos los productos.

Se realiza un procedimiento similar al anterior para poder sumar todas las búsquedas de los productos y almacenarlas en la variable "total\_productos".

```
"Suma de las búsquedas de los productos"
for busq in searches:
    total_producto[busq[1]-1][4]+=1
```

Figura 6: Ciclo for para poder sumar todas las búsquedas de cada producto.

Una vez que se calcularon y almacenaron todos los totales solos nos queda presentar los resultados, usando la función print se muestran en pantalla el total de venta anual (en pesos) y el promedio de ventas mensuales. Así mismo utilizando la paquetería matplotlib.pyplot se muestra una gráfica de barras con el total de ventas de cada mes.

```
"Presentación de resultados"
print(f"El total de venta anual es de ${total_anual}")
print("El promedio de ventas mensuales es de:",sum(meses_v)/len(meses_v))
meses=['01','02','03','04','05','06','07','08','09','10','11','12']
plt.bar(meses,meses_v)
plt.xlabel('Meses en formato MM')
plt.ylabel('Cantidad de productos vendidos')
plt.title('Cantidad de productos vendidos mensualmente')
```

Figura 7: Presentación de total de venta anual, venta promedio mensual y gráfica de ventas mensuales.

En la presentación d resultados solamente se despliegan los datos más importantes, para poder mostrar listas específicas se utiliza un ciclo while, el cual contiene preguntas booleanas y la función input para poder saber que lista requiere el usuario que se muestre, las opciones son:

1. Los 50 primeros productos con más búsquedas.
2. Los 50 primeros productos con menos búsquedas.
3. Los 50 primeros productos con más ventas.
4. Los 50 primeros productos con menos ventas.
5. Los 20 primeros productos con mejores reseñas.
6. Los 20 primeros productos con peores reseñas.
7. Productos en los que se realizaron reembolsos.

```

"Ciclo para preguntar que datos imprimir"
var_con='Y'
while var_con=='Y':
    desple=input(f"Para ver los 50 productos con más búsquedas inserte '1', para los 50 productos con menos búsquedas inserte '2', para los 50 productos con más reseñas inserte '3', para los 50 productos con menos reseñas inserte '4', para los 20 productos con mejores reseñas inserte '5', para los 20 productos con peores reseñas inserte '6', para los 7 productos con más reembolsos inserte '7', para salir inserte 'N': ")
    if desple=='1':
        total_producto.sort(key=lambda cheq: cheq[4], reverse=True)
        print('Primeros 50 productos con mayores búsquedas\n')
        for i in range(50):
            print(products[total_producto[i][0]-1][1][:18], 'Total de búsquedas:', total_producto[i][4])
    elif desple=='2':
        total_producto.sort(key=lambda cheq: cheq[4])
        print('Primeros 50 productos con menores búsquedas\n')
        for i in range(50):
            print(products[total_producto[i][0]-1][1][:18], 'Total de búsquedas:', total_producto[i][4])
    elif desple=='3':
        total_producto.sort(key=lambda cheq: cheq[1], reverse=True)
        print('Primeros 50 productos con mayores ventas\n')
        for i in range(50):
            print(products[total_producto[i][0]-1][1][:18], 'Total de piezas vendidas:', total_producto[i][1], 'Categoría: ', total_producto[i][2])
    elif desple=='4':
        total_producto.sort(key=lambda cheq: cheq[1])
        print('Primeros 50 productos con menores ventas\n')
        for i in range(50):
            print(products[total_producto[i][0]-1][1][:18], 'Total de piezas vendidas:', total_producto[i][1], 'Categoría: ', total_producto[i][2])
    elif desple=='5':
        total_producto.sort(key=lambda cheq: cheq[3], reverse=True)
        print('Primeros 20 productos con mejores reseñas\n')
        for i in range(20):
            print(products[total_producto[i][0]-1][1][:18], 'Total de reseñas:', total_producto[i][3])
    elif desple=='6':
        total_producto.sort(key=lambda cheq: cheq[3])
        print('Primeros 20 productos con peores reseñas\n')
        for i in range(20):
            print(products[total_producto[i][0]-1][1][:18], 'Total de reseñas:', total_producto[i][3])
    elif desple=='7':
        total_producto.sort(key=lambda cheq: cheq[2], reverse=True)
        print('Productos con más reembolsos\n')
        for i in range(7):
            print(products[total_producto[i][0]-1][1][:18], 'Total de reembolsos:', total_producto[i][2])
    else:
        print('Error en entrada, elija una opción de la lista')
    var_con=input('¿Desea ver otra lista? si=Y, no=N: ')

```

Figura 8: Ciclo while para la presentación de diferentes listas dependiendo de la información que desee ver el usuario.

## Solución al problema

```

SSD Kingston A400, Total de piezas vendidas: 49 Categoría: discos duros
Procesador AMD Ryz Total de piezas vendidas: 42 Categoría: procesadores
Procesador Intel C Total de piezas vendidas: 20 Categoría: procesadores
Tarjeta Madre ASRo Total de piezas vendidas: 18 Categoría: tarjetas madre
SSD Adata Ultimate Total de piezas vendidas: 15 Categoría: discos duros
Procesador AMD Ryz Total de piezas vendidas: 13 Categoría: procesadores
Tarjeta Madre ASUS Total de piezas vendidas: 13 Categoría: tarjetas madre
Procesador AMD Ryz Total de piezas vendidas: 12 Categoría: procesadores
SSD XPG SX8200 Pro Total de piezas vendidas: 11 Categoría: discos duros
Tarjeta de Video A Total de piezas vendidas: 9 Categoría: tarjetas de video

```

Figura 9: 10 primeros productos con más ventas.

En la figura 9 se puede observar que las categorías con más ventas son: discos duros, procesadores, tarjetas madre y tarjetas de vídeo.

Kit Memoria RAM Co	Total de piezas vendidas: 0	Categoría: memorias usb
Makena Smart TV LE	Total de piezas vendidas: 0	Categoría: pantallas
Seiki TV LED SC-39	Total de piezas vendidas: 0	Categoría: pantallas
Samsung TV LED LH4	Total de piezas vendidas: 0	Categoría: pantallas
Samsung Smart TV L	Total de piezas vendidas: 0	Categoría: pantallas
Makena Smart TV LE	Total de piezas vendidas: 0	Categoría: pantallas
Hisense Smart TV L	Total de piezas vendidas: 0	Categoría: pantallas
Samsung Smart TV L	Total de piezas vendidas: 0	Categoría: pantallas
Samsung Smart TV L	Total de piezas vendidas: 0	Categoría: pantallas
Hisense Smart TV L	Total de piezas vendidas: 0	Categoría: pantallas
Samsung Smart TV L	Total de piezas vendidas: 0	Categoría: pantallas
Lenovo Barra de So	Total de piezas vendidas: 0	Categoría: bocinas
Acteck Bocina con	Total de piezas vendidas: 0	Categoría: bocinas

Figura 11: 14 primeros productos con cero ventas.

En la figura 11 se puede observar que las categorías con menores ventas son: memorias usb, pantallas y bocinas.

SSD Kingston A400,	Total de búsqueda: 263	Categoría: discos duros
SSD Adata Ultimate	Total de búsqueda: 107	Categoría: discos duros
Tarjeta Madre ASUS	Total de búsqueda: 60	Categoría: tarjetas madre
Procesador AMD Ryz	Total de búsqueda: 55	Categoría: procesadores
Procesador AMD Ryz	Total de búsqueda: 41	Categoría: procesadores
Logitech Audífonos	Total de búsqueda: 35	Categoría: audífonos
TV Monitor LED 24T	Total de búsqueda: 32	Categoría: pantallas
Procesador Intel C	Total de búsqueda: 31	Categoría: procesadores
Procesador Intel C	Total de búsqueda: 30	Categoría: procesadores
SSD XPG SX8200 Pro	Total de búsqueda: 30	Categoría: discos duros

Figura 12: 10 primeros productos con mayor búsqueda.

De la figura 12 se puede observar que las categorías más buscadas son: discos duros, tarjetas madre, procesadores, audífonos y pantallas.

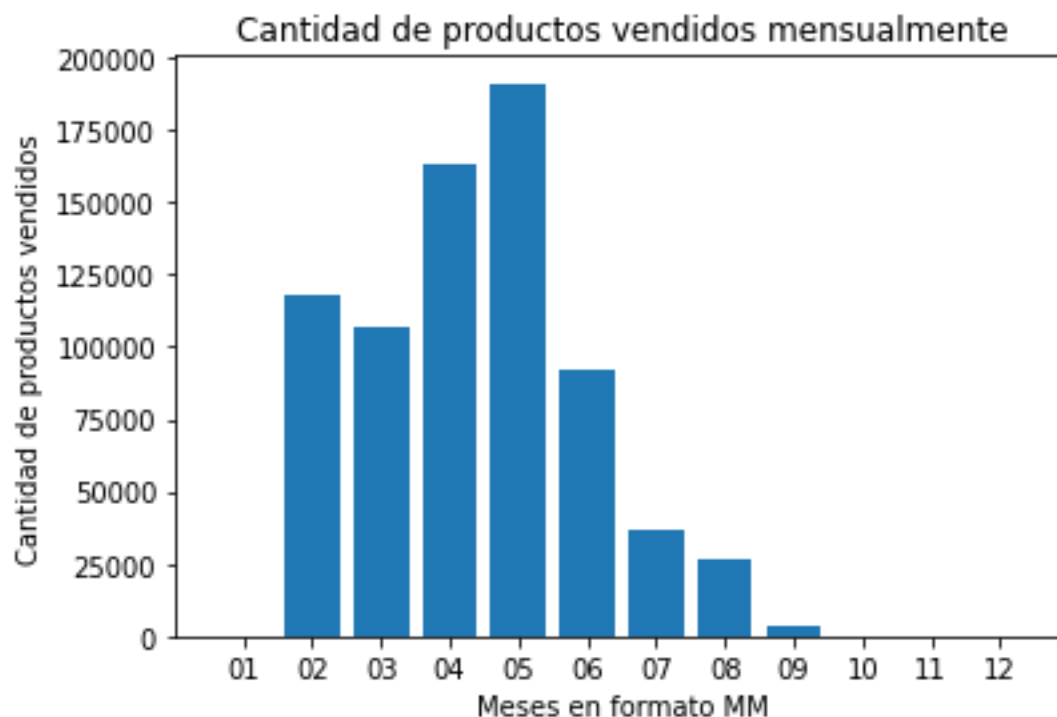
Kit Memoria RAM Co	Total de búsqueda: 0	Categoría: memorias usb
Kit Memoria RAM Co	Total de búsqueda: 0	Categoría: memorias usb
Makena Smart TV LE	Total de búsqueda: 0	Categoría: pantallas
Samsung TV LED LH4	Total de búsqueda: 0	Categoría: pantallas
Samsung Smart TV L	Total de búsqueda: 0	Categoría: pantallas
Makena Smart TV LE	Total de búsqueda: 0	Categoría: pantallas
Hisense Smart TV L	Total de búsqueda: 0	Categoría: pantallas
Samsung Smart TV L	Total de búsqueda: 0	Categoría: pantallas
Hisense Smart TV L	Total de búsqueda: 0	Categoría: pantallas
Lenovo Barra de So	Total de búsqueda: 0	Categoría: bocinas

Figura 13: 10 primeros productos con cero búsquedas.

En la figura 13 se puede observar que las categorías con menores búsquedas son: memorias usb y bocinas.

## Conclusión

Tenemos que las categorías con menores ventas y búsquedas son: memorias usb y bocinas, la única venta de memoria usb fue la de 'kit de memoria ram' y las dos únicas ventas de bocinas fueron las 'bocinas logitech', por lo que se recomienda sacar del mercado todos los demás productos de dichas categorías.



Gráfica 1: Gráfica de las ventas mensuales.

De la gráfica 1 se puede observar que en los meses de: enero, octubre, noviembre y diciembre no hay ninguna venta por lo que se recomienda no resurtir la tienda de ningún producto para poder evitar la acumulación de inventario ya que la venta podría ser nula en esos meses.

Finalmente recomienda que para la recolección de los datos de búsqueda se agregue la fecha en la que se realizó la búsqueda para así tener la información de cuales son los productos más buscados en los meses en los que existe menor venta y solo surtir la tienda de dichos productos para favorecer las ventas.