

Proyecto 1.

Introducción a Python

André Marx Puente Arévalo

Índice

Introducción.....	3
Análisis de Rotación.....	4
Productos rezagados y productos más vendidos.....	4
Productos por reseña en el servicio.....	6
Estrategia sugerida.....	7
Definición de Código.....	10
Inserción de datos.....	10
Productos más vendidos y productos rezagados.....	12
Productos por reseña.....	14
Total de ingresos y ventas pormedio mensuales, total anual y meses con más ventas al año.....	16
Extras.....	19
Inicio de sesión e interfaz de usuario.....	20
Conclusión.....	38

Introducción

En este reporte se pretende explicar el análisis obtenido de los datos que fueron brindados por la tienda **LifeStore**, con los que se busca averiguar el problema descubierto por su área de ventas, en el que se detectó una importante acumulación de inventario, reducción en las búsquedas de un grupo importante de productos provocando una disminución sustancial en las ventas.

Se proponen recomendaciones que puede implementar la tienda para dar solución a su problema. Finalmente se explica como está constituido el programa que fue creado como herramienta para la generación de reportes que se necesitaban para el análisis, explicando sus partes principales y mostrando textualmente el código utilizado para realizar lo dicho objetivo.

Análisis de rotación de Productos

Tras analizar los datos que la tienda **LifeStore** me proporcionó, he podido llegar a las siguientes conclusiones acerca de las diferentes consignias que me fueron propuestas.

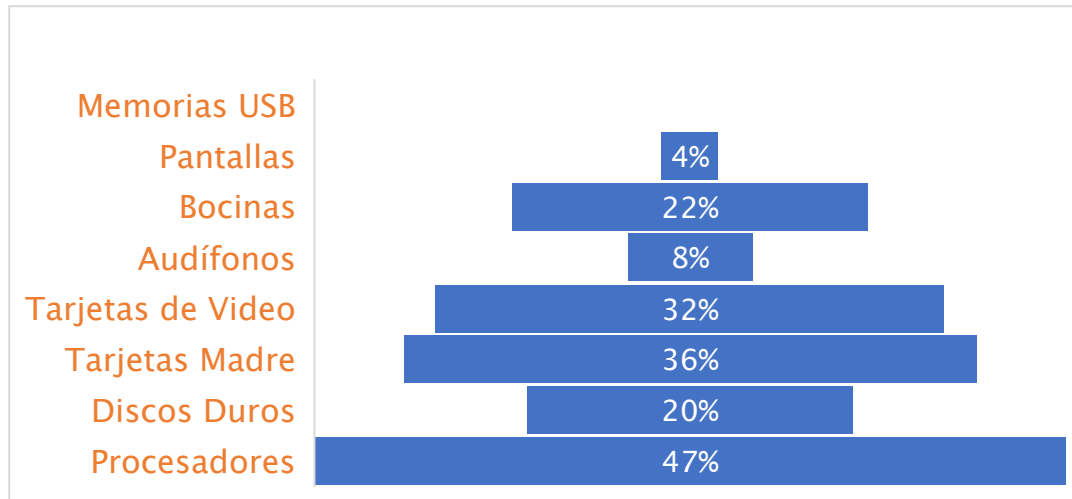
Productos rezagados y productos más vendidos

Categoría	Número de Ventas	Categoría	Número de Búsquedas
Memorias USB	1	Memorias USB	0
Pantallas	2	Bocinas	9
Bocinas	2	Pantallas	56
Audífonos	5	Audífonos	64
Tarjetas de Video	26	Tarjetas de video	82
Tarjetas Madre	49	Tarjetas Madre	137
Discos Duros	94	Procesadores	222
Procesadores	104	Discos Duros	463

Tabla 1: Top de las categorías menos vendidas y menos buscadas

En la tabla 1 se presenta la comparación entre las categorías menos vendidas y las menos buscadas, por lo que se puede apreciar que la tienda **LifeStore** tiene cuatro categorías de productos con ventas muy bajas. Si nos percatamos se ha vendido una sola memoria USB, de la cual, no se realizó ninguna búsqueda, es decir, se vendió por dos motivos o le salió al comprador al inicio de la página o fue una sugerencia de compra mientras veía otro de los productos, esto tiene sentido, ya que, por el aumento en la popularidad de los servicios de almacenamiento en la nube se están convirtiendo en productos obsoletos, por lo que los clientes ya no los consumen. Por otro lado es preocupante el caso de las pantallas, ya que, la proporción de ventas y búsquedas es de 4%, esto quiere decir que aproximadamente por cada 100 búsquedas de productos de esta categoría solamente se venden 4. Mientras que las “bocinas” tienen una proporción del 22% pero tienen muy bajas búsquedas, por lo que se esperaría que aumenten las búsquedas de estos productos. En cuanto a los audífonos, su proporción es de aproximadamente 8% por lo que es el otro producto preocupante, ya que, los datos nos indican que no se venderán tan fácilmente, pues los clientes los buscan pero no los compran.

Por lo tanto, las categorías que más productos rezagados tienen son las pantallas y los audífonos, ya que se están vendiendo demasiado mal y la tienda cuenta con mucho inventario de estos.



Gráfica 1: Proporción de ventas y búsquedas

Como se puede apreciar en la Gráfica 1 y en la tabla 1, la categoría estrella de **LifeStore** son los “procesadores”, ya que tanto es la más vendida como también es la categoría con mejor proporción de ventas y búsquedas con 47 ventas por cada 100 búsquedas. Nótese que el producto a destacar en la categoría de procesadores es el que tiene id “3” y cuenta con 42 ventas de un total de 104, es decir, equivale al 40% de las ventas de esta categoría.

Ahora bien, por otro lado tenemos la categoría de “discos duros”, que es la segunda de las categorías con más ventas en la tienda y a su vez la categoría con más búsquedas, lo que provoca un hecho peculiar, ya que, su proporción de ventas y búsquedas es de las más bajas, esto quiere decir, que son productos muy buscados pero al final hay algo que al cliente no lo convence para finalizar con la compra. El producto con mayores ventas en su categoría y en la tienda en genral es el de id “54” con nombre “SSD Kingston A400, 120GB, SATA III, 2.5”, 7mm”, el cual cuenta con un total de 50 ventas, lo equivalente al 53% de las ventas de esta categoría.

Finalmente se tienen las categorías que no destacan por sus altas o bajas ventas, más bien, son las que tienen una proporción bastante aceptable en cuanto a ventas y búsquedas. Estas son las “tarjetas madres” con una proporción del 36% y las “tarjetas de video” con una proporción del 32%. Son categorías no son completamente rezagadas, pero tampoco se venden a una velocidad moderada.

Productos por reseña en el servicio

Categoría	Stock	Calificación
Memorias USB	15	5
Bocinas	146	4
Tarjetas de Video	349	4
Discos Duros	433	5
Audífonos	573	4
Tarjetas Madre	814	4
Pantallas	1190	5
Procesadores	1821	5

Tabla 2: Stock y calificación promedio por categoría.

En este apartado se analizan más a profundidad las diferentes categorías de productos que vende la tienda, pero considerando las reseñas de sus clientes, datos que nos pueden brindar información suficiente para saber el por qué los productos no se están vendiendo como se espera.

Analizando la tabla dos nos damos cuenta que la única unidad de “memorias USB” vendidas, fue bien recibida, dado que cuenta con una calificación excelente, por lo que esto confirma la conclusión a la que se llegó anteriormente, es decir, los productos no se venden porque sean de mala calidad o defectuosos, si no, porque ya empiezan a ser prescindibles para los clientes. Por otro lado, en el caso de las bocinas, se tiene poco stock, pero y un bajo índice de ventas, por lo que se cree que puede ser por la calificación promedio de la categoría.

Id	Nombre del producto	Calificación	Búsquedas	Ventas
75	Lenovo Barra de Sonido	4	0	0
76	Acteck Bocina con Subwoofer AXF-290	4	2	0
77	Verbatim Bocina Portátil Mini	4	0	0
78	Ghia Bocina Portátil BX300	4	0	0
79	Naceb Bocina Portátil NA-0301	4	0	0
80	Ghia Bocina Portátil BX800	4	1	0
81	Ghia Bocina Portátil BX900	4	0	0
82	Ghia Bocina Portátil BX400	4	0	0
83	Ghia Bocina Portátil BX500	4	0	0
74	Logitech Bocinas para Computadora con Subwoofer G560	4	6	2

Tabla 3: Se muestran los datos de los productos pertenecientes a la categoría de “bocinas”

En la tabla 3 se puede apreciar que se están rezagando demasiados productos, de hecho las únicas ventas se han realizado en bocinas de la marca “Logitech” y también es la marca de esta categoría con más búsquedas. Tras analizar estos últimos datos notamos que el resultado de las ventas no son tanto por la calificación de los productos, si no, porque la gente no conoce las marcas que vende la tienda.

Nótese que las categorías que no tienen un número significativo o poco significativo de ventas y búsquedas, son los que presentan menos calificación, siendo que los productos que maneja la tienda, tampoco son de mala calidad, esto se puede explicar por la naturaleza del mercado al que se enfoca **LifeStore**, el nicho es bastante exigente, dado de que tratan de componentes de computadora con un precio considerable para los clientes, por lo que estos buscarán lo mejor por lo que pagan.

Ahora bien, las categorías que más ventas y más búsquedas tienen, son las que mejores reseñas poseen, ya que, se venden productos de marcas reconocidas en el mercado al que se enfoca la tienda.

Estrategia sugerida

Mes	Año	Ventas	Ingresos
Abril	2020	75	193295
Enero	2020	53	120237
Marzo	2020	51	164729
Febrero	2020	41	110139
Mayo	2020	35	96135
Julio	2020	11	26949
Junio	2020	11	36949
Agosto	2020	3	3077
Noviembre	2019	1	4209
Septiembre	2020	1	4199
Mayo	2002	1	259

Tabla 4: Ventas e ingresos al mes

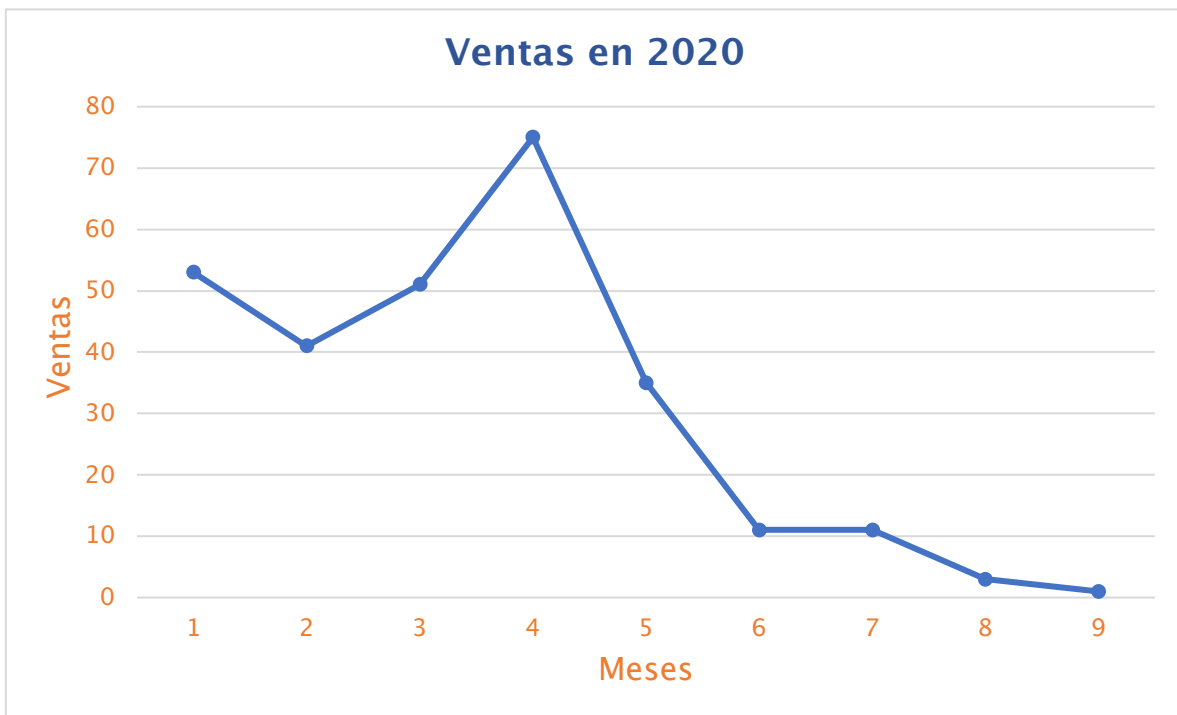
Dado el análisis obtenido de las categorías, se sugiere lo siguiente:

- Descontinuar la categoría de “memorias USB”, ya que, se espera que el mercado para estos productos siga disminuyendo, por lo que no es recomendable aumentar el inventario y lo mejor sería sólo tratar de vender las pocas unidades (15) con las que se cuentan actualmente.

- En cuanto a la categoría de “discos duros”, estos productos cuenta con una calificación promedio de 5 (lo más alto), por lo que considero que no los compra la gente porque sean productos de mala calidad o que salgan defectuosos, si no, por su precio, así que para poder incrementar la venta de estos productos se recomienda a lo largo del año, en particular en meses donde la venta es baja como en septiembre o en noviembre, poner varias ofertas de estos productos, pero que no sean muy altas, ya que, sí se venderán.
- Dado que las “bocinas” son de los productos que tienen una no tan baja proporción entre ventas y búsquedas, considero que existen dos factores que podrían aumentar significativamente el consumo de estas:
 - Dejar de adquirir productos de la marca Ghia, ya que, durante el año no se vendieron y no se buscaron ninguno de estos, por lo que se recomienda vender con grandes descuentos las unidades que se tienen y en los meses cuando más dinero tiene la gente como lo son en junio y diciembre.
 - Adquirir más variedad de bocinas de la marca “Logitech”, ya que, estos son los únicos productos de esta categoría que se han vendido y rellenar el inventario para el modelo que se está manejando actualmente. Finalmente se recomienda invertir en publicidad para las bocinas que vende la tienda, por la proporción que actualmente se tiene de ventas y búsquedas.
- Incentivar la venta de “pantallas” y “audífonos” porque **LifeStore** presenta un gran rezago de estos productos, será necesario poner grandes descuentos y promociones, como por ejemplo: “Tras la compra de un televisor de al menos 30 pulgadas, llévate unos audífonos con un 60% de descuento”. Esto lo más recomendable es hacerlo durante todo el año y poner los mayores descuentos en los meses con menos ventas como septiembre y agosto y en meses donde existe una mayor cantidad de ventas, como en enero, marzo y abril, mantener descuentos, pero no extremadamente altos. Esto no es todo, con forme aumente la muestra de ventas, búsquedas, etc. tomar nuevas decisiones al respecto del inventario.
- Por otro lado, enfocándome en particular en la categoría de “pantallas”, dada la naturaleza de los productos que en su mayoría vende la tienda, se recomienda, dejar de adquirir tantas pantallas y aumentar el catálogo de monitores, ya que el resto de categorías están más enfocadas al obtener componentes para armar una computadora.
- Dadas las tendencias de ventas durante el año 2020 que presenta la tienda, se tiene suficiente inventario de “tarjetas de video” por el momento, no se recomienda el aumento hasta obtener un número

considerablemente mayor de demandas de estos, es decir, mantenerse adquiriendo estas cantidades con forme se vayan agotando.

- Referente a la categoría de “tarjetas madre”, no adquirir más inventario de esta, ya que, la tienda cuenta con mucho rezago de producto, renovar inventario cuando se tengan menos de 100 unidades, dado que la demanda por el momento no ha sido muy elevada. De igual manera que en el caso de otras categorías, es recomendable incentivar la venta de estos productos con rebajas no muy elevadas, entre 20 y 30 por ciento, por lo que, son productos que según las tendencias, se venderán.
- En cuanto a la categoría de “procesadores”, se tiene suficiente inventario para hacer frente a los meses futuros, sin ningún problema, ya que se cuenta con un inventario de 1821 unidades, de las cuales el 54% de estas, pertenecen al producto más vendido es la categoría, por lo que se espera que no se adquiera más inventario hasta reducir significativamente el que se tiene actualmente, para no propiciar a tener productos rezagados.



Gráfica 2: Ventas por meses en el 2020

Definición del Código

El programa se realizó en Python y el código se divide en cinco grandes partes:

- Inserción de los datos
- Productos más vendidos y productos rezagados
 - Generar un listado de los 50 productos con mayores ventas y uno con los 100 productos con mayor 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 una, 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
- Extra
 - Contiene listas que generé para los reportes que se le muestran al administrador.
- Inicio de sesión e interfaz de usuario

Inserción de los datos

"""

Autor: Andre Puente

Fecha: 6 / 09 / 2020

Datos de inicio de sesion:

1- Cuenta del administrador

Usuario: administrador

Contrasenía: admin

2- Cuenta de usuario

Usuario: usuario

Contrasenía: asdf

This is the LifeStore-SalesList data:

lifestore_searches = [id_search, id product]
lifestore_sales = [id_sale, id_product, score (from 1 to 5), date, refund (1 for true or 0 to false)]
lifestore_products = [id_product, name, price, category, stock]

Listas creadas:

total_productos_vendidos = [id_producto, nombre_producto, numero_productos_vendidos, categoria]
total_productos_buscados = [id_producto, nombre_producto, numero_busquedas_productos, categoria]
ids_productos_contados = [id_producto]
top_21_mas_vendidos = [id_producto, nombre_producto, numero_productos_vendidos, categoria]
top_21_mas_buscados = [id_producto, nombre_producto, numero_busquedas_productos, categoria]
categorias = [categoria]
ventas_por_categorias = [categoria, numero_productos_vendidos]
busquedas_por_categorias = [categoria, numero_busquedas_productos]
top_4_categorias_menos_buscadadas = [categoria, numero_busquedas_productos]
top_4_categorias_menos_vendidos = [categoria, numero_productos_vendidos]
"categoria"_busquedas = [id_producto, nombre_producto, numero_busquedas_productos]
"categoria"_ventas = [id_producto, nombre_producto, numero_productos_vendidos]
lifestore_ventas = [id_venta, id_producto, calificacion (1 a 5), fecha, Devolucion/No Devuelto, nombre, precio]
productos_resenias = [id_producto, nombre_producto, precio_producto, numero_ventas, calificacion_promedio (1 a 5)]
top_20_peores_resenias = [id_producto, nombre_producto, precio_producto, numero_ventas, calificacion_promedio (1 a 5)]
top_20_mejores_resenias = [id_producto, nombre_producto, precio_producto, numero_ventas, calificacion_promedio (1 a 5)]
ventas_por_mes = [mes, anio, numero_ventas]
ingresos_por_mes = [mes, anio, numero_ingresos]
meses_anios = [mes/anio]
meses = [mes]
anios = [anio]
ventas_por_anio = [anio, numero_ventas]
ingresos_por_anio = [anio, numero_ingresos]
promedios_ingresos_por_mes = [mes, anio, ingreso_promedio]
promedios_ventas_anio = [anio, ventas_promedio]
lifestore_productos = [id_producto, nombre, precio, categoria, stock, numero_ventas, numero_busquedas, calificacion_promedio]
stock_categorias = [categoria, stock, calificacion]

```
lifestore_products = [
    [1, 'Procesador AMD Ryzen 3 3300X S-AM4, 3.80GHz, Quad-Core, 16MB
    L2 Cache', 3019, 'procesadores', 16],
    [2, 'Procesador AMD Ryzen 5 3600, S-AM4, 3.60GHz, 32MB L3 Cache, con
    Disipador Wraith Stealth', 4209, 'procesadores', 182],
    [3, 'Procesador AMD Ryzen 5 2600, S-AM4, 3.40GHz, Six-Core, 16MB L3
    Cache, con Disipador Wraith Stealth', 3089, 'procesadores', 987],
    [4, 'Procesador AMD Ryzen 3 3200G con Gráficos Radeon Vega 8, S-AM4,
    3.60GHz, Quad-Core, 4MB L3, con Disipador Wraith Spire', 2209,
    'procesadores', 295], ... , [95, 'logear Audífonos Gamer GHG601, Alámbrico,
    1.2 Metros, 3.5mm, Negro', 999, 'audifonos', 2],
    [96, 'Klip Xtreme Audífonos Blast, Bluetooth, Inalámbrico, Negro/Verde',
    769, 'audifonos', 2]
]
lifestore_sales = [
    [1, 1, 5, '24/07/2020', 0],
    [2, 1, 5, '27/07/2020', 0],
    [3, 2, 5, '24/02/2020', 0],
    .
    .
    .
    [282, 89, 3, '06/01/2020', 0],
    [283, 94, 4, '10/04/2020', 0]
]
lifestore_searches = [
    [1, 1],
    [2, 1],
    .
    .
    .
    [1032, 95],
    [1033, 95]
]
```

Productos más vendidos y productos rezagados

Obtengo una lista con los 21 productos mas vendidos y los 21 más buscados

Primero creare una lista la cual contenga el numero de unidades vendidas por cada producto

```
# Creo una lista vacía que guaradara los datos
ids_productos_contados = []
total_productos_vendidos = []
total_productos_buscados = []

# Los siguientes cilos son para contar los productos vendidos y los buscados
for producto in lifestore_products:
    # Obtengo uno de los productos vendidos
    id_producto = producto[0]
    nombre_producto = producto[1]
    categoria = producto[3]
    numero_productos_vendidos = 0
    numero_búsquedas_productos = 0
    # Verifico que no haya contado ya ese producto
    if id_producto in ids_productos_contados:
        continue
    else:
        # Cuento el numero de ventas
        for venta in lifestore_sales:
            if id_producto == venta[1]:
                numero_productos_vendidos += 1
        # Cuento el numero de búsquedas
        for busqueda in lifestore_searches:
            if id_producto == busqueda[1]:
                numero_búsquedas_productos += 1
        # Agrego a las listas el id del producto contado y el numero de ventas
        y búsquedas
        ids_productos_contados.append(id_producto)
        total_productos_vendidos.append([id_producto, nombre_producto,
        numero_productos_vendidos, categoria])
        total_productos_buscados.append([id_producto, nombre_producto,
        numero_búsquedas_productos, categoria])

# Ordeno de mayor a menor las ventas y obtengo el top 21
total_productos_vendidos.sort(reverse = True, key = lambda ventas:
ventas[2])
top_21_mas_vendidos = total_productos_vendidos[0:21]

total_productos_buscados.sort(reverse = True, key = lambda busquedas:
busquedas[2])
top_21_mas_buscados = total_productos_buscados[0:21]

## Obtengo una lista con las categorias menos vendidas y con las menos
buscadas

# Inicializo las listas que guardaran los datos
```

```
categorias = []
ventas_por_categorias = []
busquedas_por_categorias = []
```

Los siguientes ciclos son para contar las ventas y busquedas por categoria

```
for producto in lifestore_products:
    categoria = producto[3]
    numero_categorias_vendidas = 0
    numero_categorias_buscadadas = 0
    # Verifico que no haya contado esa categoria
    if categoria in categorias:
        continue
    else:
        # Cuento el numero de ventas
        for producto_2 in total_productos_vendidos:
            if categoria == producto_2[3]:
                numero_categorias_vendidas += producto_2[2]
        # Cuento el numero de busquedas
        for producto_2 in total_productos_buscados:
            if categoria == producto_2[3]:
                numero_categorias_buscadadas += producto_2[2]
        # Guardo los datos en las listas
        categorias.append(categoria)
        ventas_por_categorias.append([categoria,
numero_categorias_vendidas])
        busquedas_por_categorias.append([categoria,
numero_categorias_buscadadas])
```

```
ventas_por_categorias.sort(key = lambda x: x[1])
busquedas_por_categorias.sort(key = lambda x: x[1])
```

Productos por reseña en el servicio

```
lifestore_ventas = lifestore_sales
```

Le agrego el nombre a los productos

```
for indice in range(0, len(lifestore_ventas)):
    id_producto = lifestore_ventas[indice][1]
    for indice_2 in range(0, len(lifestore_products)):
        if id_producto == lifestore_products[indice_2][0]:
            lifestore_ventas[indice].append(lifestore_products[indice_2][1])
            lifestore_ventas[indice].append(lifestore_products[indice_2][2])
        else:
            continue
```

```
# Hago un cilo para que sea explicito si hubo devolucion o no
for indice in range(0, len(lifestore_ventas)):
    if lifestore_ventas[indice][4] == 1:
        lifestore_ventas[indice][4] = "Devolucion"
    else:
        lifestore_ventas[indice][4] = "No Devuelto"

# Creo una nueva lista que guardara el promedio de las resenias
productos_resenias = []
ids_productos_contados = []
for indice in range(0, len(lifestore_ventas)):
    id_producto = lifestore_ventas[indice][1]
    nombre_producto = lifestore_ventas[indice][5]
    precio_producto = lifestore_ventas[indice][6]
    numero_resenias = 0
    suma_resenias = 0
    promedio_resenias = 0
    if id_producto in ids_productos_contados:
        continue
    else:
        # Sumo las calificaciones que le dieron a cada producto
        for indice_2 in range(0, len(lifestore_ventas)):
            if id_producto == lifestore_ventas[indice_2][1]:
                suma_resenias += lifestore_ventas[indice_2][2]
            else: continue
        # Obtengo el numero de resnias que le dejaron
        for indice_3 in range(0, len(total_productos_vendidos)):
            if id_producto == total_productos_vendidos[indice_3][0]:
                numero_resenias = total_productos_vendidos[indice_3][2]
            else: continue
        promedio_resenias = round(suma_resenias / numero_resenias)
        ids_productos_contados.append(id_producto)
        productos_resenias.append([id_producto, nombre_producto,
precio_producto, numero_resenias, promedio_resenias])

# Obtengo los top 20 de los productos con mejores y peores resnias
productos_resenias.sort(reverse = True, key = lambda calificacion:
calificacion[4])
top_20_mejores_resenias = productos_resenias[0:20]
productos_resenias.sort(key = lambda calificacion: calificacion[4])
top_20_peores_resenias = productos_resenias[0:20]
```

Total de ingresos y ventas promedio mensuales, total anual y meses con más ventas al año

```
# Inicializo las listas que guardaran los datos
ventas_por_mes = []
ingresos_por_mes = []
meses_anios = []
anios = []
ventas_por_anio = []
ingresos_por_anio = []

# Obtengo el numero de ventas por mes
for indice in range(0, len(lifestore_ventas)):
    # Extraigo el mes y año de cada venta
    mes_venta = lifestore_ventas[indice][3][3:5]
    anio_venta = lifestore_ventas[indice][3][6:]
    # Inicializo una variable que guardara el numero de ventas
    numero_ventas_mes = 0
    ingreso_mes = 0
    # Verifico que no se haya contado ya ese mes
    if lifestore_ventas[indice][3][3:] in meses_anios:
        continue
    else:
        # Cuento las ventas e ingresos que se tuvieron por mes
        for indice_2 in range(0, len(lifestore_ventas)):
            if lifestore_ventas[indice][3][3:] == lifestore_ventas[indice_2][3][3:]:
                numero_ventas_mes += 1
                ingreso_mes = ingreso_mes + lifestore_ventas[indice_2][6]
            else: continue
        # Guardo el mes, anio y los datos obtenidos
        meses_anios.append(lifestore_ventas[indice][3][3:])
        ventas_por_mes.append([mes_venta, anio_venta,
                                numero_ventas_mes])
        ingresos_por_mes.append([mes_venta, anio_venta, ingreso_mes])

# Obtengo las ventas e ingresos por anio
for indice in range(0, len(ventas_por_mes)):
    anio = ventas_por_mes[indice][1]
    venta_por_anio = 0
    ingreso_por_anio = 0
    # Verifico que no haya contado ese anio
    if anio in anios:
        continue
    else:
        # Sumo las ventas por cada anio
        for indice_2 in range(0, len(ventas_por_mes)):
```



```
if anio == ventas_por_mes[indice_2][1]:
    venta_por_anio = venta_por_anio + ventas_por_mes[indice_2][2]
else: continue
# Sumo los ingresos por cada anio
for indice_3 in range(0, len(ingresos_por_mes)):
    if anio == ingresos_por_mes[indice_3][1]:
        ingreso_por_anio = ingreso_por_anio +
ingresos_por_mes[indice_3][2]
    else: continue
# Guardo los datos
anios.append(anio)
ventas_por_anio.append([anio, venta_por_anio])
ingresos_por_anio.append([anio, ingreso_por_anio])

# Obtengo el total de ventas
total_ventas = 0
for indice in range(0, len(ventas_por_mes)):
    total_ventas += ventas_por_mes[indice][2]

# Obtengo el promedio de las ventas generales por mes
promedio_ventas_general_mes = round(total_ventas / len(ventas_por_mes))

# Obtengo el total de ingresos
total_ingresos = 0
for indice in range(0, len(ingresos_por_mes)):
    total_ingresos += ingresos_por_mes[indice][2]

# Obtengo el promedio de ingresos general por meses
promedio_ingresos_general_mes = total_ingresos / len(ingresos_por_mes)

# Obtengo el promedio de ingresos por cada mes
promedios_ingresos_por_mes = []
meses_anios = []
for indice in range(0, len(ingresos_por_mes)):
    mes = ingresos_por_mes[indice][0]
    anio = ingresos_por_mes[indice][1]
    ingreso_mes = ingresos_por_mes[indice][2]
    promedio_ingreso_mes = 0
    # Verifico que no haya promediado ya el mes
    if mes + "/" + anio in meses_anios:
        continue
    else:
        for indice_2 in range(0, len(ventas_por_mes)):
            if mes == ventas_por_mes[indice_2][0] and anio ==
ventas_por_mes[indice_2][1]:
```

```
        promedio_ingreso_mes = round(ingreso_mes /
ventas_por_mes[indice_2][2], 3)
    else: continue
    # Guardo los datos en las listas
    meses_anios.append(mes + "/" + anio)
    promedios_ingresos_por_mes.append([mes, anio,
promedio_ingreso_mes])

# Ahora, obtengo el promedio general de ventas
ingresos_totales_anio = 0
for indice in range(0, len(ventas_por_anio)):
    ingresos_totales_anio += ventas_por_anio[indice][1]
promedio_ventas_general_anio = round(ingresos_totales_anio /
len(ventas_por_anio))

# Obtengo el promedio de ventas por cada anio
anios = []
promedios_ventas_anio = []
for indice in range(0, len(ventas_por_anio)):
    anio = ventas_por_anio[indice][0]
    ventas_anio = ventas_por_anio[indice][1]
    promedio_ventas_anio = 0
    numero_meses = 0
    if anio in anios: continue
    else:
        for indice_2 in range(0, len(ventas_por_mes)):
            if anio == ventas_por_mes[indice_2][1]:
                numero_meses += 1
        promedio_ventas_anio = round(ventas_anio / numero_meses)
        anios.append(anio)
        promedios_ventas_anio.append([anio, promedio_ventas_anio])

# Obtengo el promedio general por anio
total_ingreso=0
for ingreso in ingresos_por_anio[1]:
    total_ingreso += int(ingreso)
promedio_ingresos_general_anio =
round(total_ingreso/len(ingresos_por_anio), 3)

# Obtengo los meses con más ventas al anio
ventas_por_mes.sort(reverse = True, key = lambda ventas: ventas[2])
```

Extras

Hago un cruce de tablas

```
lifestore_productos = lifestore_products
```

```
for producto in lifestore_productos:
```

```
    id_producto = producto[0]
```

Obtengo el numero de ventas

```
for producto_2 in total_productos_vendidos:
```

```
    if id_producto == producto_2[0]:
```

```
        ventas = producto_2[2]
```

Obtengo el numero de busquedas

```
for producto_2 in total_productos_buscados:
```

```
    if id_producto == producto_2[0]:
```

```
        busquedas = producto_2[2]
```

Obtengo el promedio de su calificacion

```
for producto_2 in productos_resenias:
```

```
    if id_producto == producto_2[0]:
```

```
        resenia = producto_2[4]
```

Agrego los datos

```
producto.append(ventas)
```

```
producto.append(busquedas)
```

```
producto.append(resenia)
```

Obtengo el stock por cada categoria

Creo una lista por donde almacenare los datos

```
stock_categorias = []
```

```
categorias_contadas = []
```

```
for producto in lifestore_productos:
```

```
    categoria = producto[3]
```

```
    stock = 0
```

```
    calificacion = 0
```

```
    contador = 0
```

Verifico que no haya contado ya la categoria

```
if categoria in categorias_contadas:
```

```
    continue
```

```
else:
```

Sumo el stock y la calificacion por categoria

```
for producto_2 in lifestore_productos:
```

```
    if categoria == producto_2[3]:
```

```
        stock += producto_2[4]
```

```
        contador += 1
```

```
        calificacion += producto_2[7]
```

Obtengo la calificacion promedio

```
if contador == 0:
```

```
    calificacion_promedio = 0
```

```
else:
    calificacion_promedio = round(calificacion/contador)
    categorias_contadas.append(categoria)
    stock_categorias.append([categoria, stock, calificacion_promedio])
stock_categorias.sort(key = lambda stock: stock[1])
```

Inicio de sesión e interfaz de usuario

```
usuario = ""
contador = 0
usuarios = ["administrador", "usuario", "0"]
contrasenas = ["admin", "asdf", "0"]
contrasenia = ""
salir = True
```

Este ciclo es para mostrar las interfaces al usuario
while salir:

```
print(' -----')
print('|                                     |')
print('|             LifeStore                 |')
print('|                                     |')
print(' -----')
```

Pido que ingresen el usuario

```
while usuario not in usuarios:
    if contador == 0 and usuario == "":
        print('Introduzca "0" para salir')
        usuario = input("Ingrese el nombre de usuario: ")
        contador += 1
    elif usuario == "0":
        break
    elif contador < 5:
        contador += 1
        print("\nEl usuario que ingresaste NO existe \n")
        print('Introduzca "0" para salir')
        usuario = input("Ingrese el nombre de usuario: ")
    else:
        break
```

Termino el programa

```
if usuario == "0":
    print("\nHasta luego.")
    break
elif usuario not in usuarios:
```

```
print("\nSe le acabaron sus intentos, hasta luego ")
break
```

```
# Pido que ingresen la contrasenia
```

```
contador = 0
while (usuario == "administrador" and contrasenia != "admin") \
or (usuario == "usuario" and contrasenia != "asdf"):
    if contador == 0 and contrasenia == "":
        print('\nIntroduzca "0" para salir')
        contrasenia = input("Ingrese la contrasenia: ")
        contador += 1
    elif contrasenia == "0":
        break
    elif contador < 5:
        contador += 1
        print("\nLa contrasenia es INCORRECTA \n")
        print('\nIntroduzca "0" para salir')
        contrasenia = input("Ingrese la contrasenia: ")
    else:
        break
```

```
# Termino el programa
```

```
if contrasenia == "0":
    print("\nHasta luego.")
    break
elif contrasenia not in contrasenias:
    print("\nSe le acabaron sus intentos, hasta luego.")
    break
```

```
print("\nBienvenido, seleccione una de las siguientes opciones (solo
poner el numero)\n")
```

```
salir_2 = True
```

```
while salir_2:
```

```
# Propongo una interfaz para cada usuario
```

```
if usuario == "usuario":
    opcion = ""
    opcion_2 = ""
    opcion_3 = ""
    print("\nSeleccione el numero de la opcion que desea: \n")
    print(" 1. Top 21 de los productos mas vendidos")
    print(" 2. Top 21 de los productos mas buscados ")
    print(" 3. Top de las categorias menos vendidas")
    print(" 4. Top de las categorias menos buscadas")
    print(" 5. Top 20 productos con mejores resenias")
```

```
print(" 6. Top 20 productos con peores resenias")
print(" 7. Ingresos promedios por mes")
print(" 8. Ingresos promedios por anio")
print(" 9. Ventas promedio por anio")
print("10. Meses con mas ventas por mes")
print(" 0. Para cambiar de sesion (salir)\n")
opcion = input("Ingrese la opcion: ")
```

Dependiendo la opcion seleccionada brindo las distintas interfaces

```
if opcion == "0":
    salir_2 = False
    break
elif opcion == "1":
    salir_3 = True
    while salir_3:
        # Muestro el top
        print('\n-----Top 21 de los productos mas
vendidos-----\n')
        print("Indice | Id del producto | Nombre | Categoria | Numero
de ventas")
        print("-----")
        i = 1
        for producto in top_21_mas_vedidos:
            print(i, " | ", producto[0], " | ", producto[1], " | ",
producto[3], " | ", producto[2] )
            i += 1
        opcion_2 = input('Para regresar oprima "0": ')
        if opcion_2 == "0":
            salir_3 = False
        else:
            print('Caracter incorrecto')
            continue
        # Resgreso al menu principal
        if opcion_2 == "0":
            continue
        elif opcion == "2":
            salir_3 = True
            while salir_3:
                # Muestro el top
                print('\n-----Top 21 de los productos mas
buscados-----\n')
                print("Indice | Id del producto | Nombre | Categoria | Numero
de busquedas")
```

```

print("-----")
-----")
    i = 1
    for producto in top_21_mas_buscados:
        print(i, " | ", producto[0], " | ", producto[1], " | ",
producto[3], " | ", producto[2])
        i += 1
    opcion_2 = input('Para regresar oprima "0": ')
    if opcion_2 == "0":
        salir_3 = False
    else:
        print('\nCaracter incorrecto\n')
        continue
    # Resgreso al menu de opciones
    if opcion_2 == "0":
        continue
    elif opcion == "3":
        salir_3 = True
    while salir_3:
        # Muestro el siguiente sub-menu
        print("\nSeleccione una categoria:\n")
        indice = 1
        opcion_2 = ""
        opcion_3 = ""
        # Hago el print para enseñar las categorias
        for categoria in categorias:
            print(indice, ". ", categoria)
            indice += 1
        print('\nADVERTENCIA: si agurega algo que no sea un
numero, el programa explota')
        print("\n0. Regresar al menu principal")
        opcion_2 = int(input('Ingrese el numero de la categoria:'))
        if opcion_2 == 0:
            salir_3 = False
        elif opcion_2 in range(1, len(lifestore_productos)):
            salir_4 = True
            while salir_4:
                categoria = categorias[opcion_2 - 1]
                indice = 1
                lifestore_productos.sort(key=lambda ventas: ventas[5])
                print(
                    '\n----- Top de los
productos menos vendidos ----- \n')
                print(
                    'Indice | Id | Nombre | Categoria | Precio | Calificacion
| Stock | Numero Busquedas | Numero Ventas')

```

```

print(
    '-----'
    -----')
    for producto in lifestore_productos:
        if categoria == producto[3]:
            print(indice, ' | ', producto[0], ' | ', producto[1], ' | ',
producto[3], ' | ',
                producto[2], ' | ', producto[7], ' | ', producto[4], '
| ', producto[6],
                    ' | ', producto[5])
            indice += 1
        opcion_3 = input('\nRegresar "0": ')
        if opcion_3 == "0":
            salir_4 = False
        else:
            print('\nCaracter INCORRECTO')
    else:
        print('\nSolo puede ingresar los numeros de las opciones
que se le muestran')
    elif opcion == "4":
        salir_3 = True
        while salir_3:
            # Muestro el siguiente sub-menu
            print("\nSeleccione una categoria:\n")
            indice = 1
            opcion_2 = ""
            opcion_3 = ""
            # Hago el print para enseñar las categorias
            for categoria in categorias:
                print(indice, ". ", categoria)
                indice += 1
            print('\nADVERTENCIA: si agurega algo que no sea un
numero, el programa explota')
            print("\n0. Regresar al menu principal")
            opcion_2 = int(input('Ingresa el numero de la categoria:'))
            if opcion_2 == 0:
                salir_3 = False
            elif opcion_2 in range(1, len(lifestore_productos)):
                salir_4 = True
                while salir_4:
                    categoria = categorias[opcion_2 - 1]
                    indice = 1
                    lifestore_productos.sort(key=lambda busquedas:
busquedas[6])
                    print(

```



```

        '\n----- Top de los
productos menos buscados ----- \n')
    print(
        'Indice | Id | Nombre | Categoria | Precio | Calificacion
| Stock | Numero Ventas | Numero Búsquedas')
    print(
        '-----'
        -----')
    for producto in lifestore_productos:
        if categoria == producto[3]:
            print(indice, ' | ', producto[0], ' | ', producto[1], ' | ',
producto[3], ' | ',
                    producto[2], ' | ', producto[7], ' | ', producto[4], '
| ', producto[5],
                    ' | ', producto[6])
            indice += 1
        opcion_3 = input('\nRegresar "0": ')
        if opcion_3 == "0":
            salir_4 = False
        else:
            print('\nCaracter INCORRECTO')
    else:
        print('\nSolo puede ingresar los numeros de las opciones
que se le muestran')
        elif opcion == "5":
            salir_3 = True
            while salir_3:
                # Muestro el top
                print('\n----- Top 20 productos con mejores
resenas ----- \n')
                print("Indice | Id del producto | Nombre | Precio | Numero de
ventas | Carificacion")
                print("-----")
                -----")
                i = 1
                for producto in top_20_mejores_resenas:
                    print(i, " | ", producto[0], " | ", producto[1], " | ",
producto[3], " | ", producto[2])
                    i += 1
                opcion_2 = input('Para regresar oprima "0": ')
                if opcion_2 == "0":
                    salir_3 = False
                else:
                    print('Caracter incorrecto')
                    continue
                # Resgreso al menu principal

```

```

    if opcion_2 == "0":
        continue
    elif opcion == "6":
        salir_3 = True
        while salir_3:
            # Muestro el top
            print("\n----- Top 20 productos con peores
resenias -----")
            print("Indice | Id del producto | Nombre | Precio | Numero de
ventas | Carificacion")
            print("-----")
            i = 1
            for producto in top_20_peores_resenias:
                print(i, " | ", producto[0], " | ", producto[1], " | ",
producto[3], " | ", producto[2])
                i += 1
            opcion_2 = input('Para regresar oprima "0": ')
            if opcion_2 == "0":
                salir_3 = False
            else:
                print('Caracter incorrecto')
                continue
            # Resgreso al menu principal
            if opcion_2 == "0":
                continue
            elif opcion == "7":
                salir_3 = True
                while salir_3:
                    print("\nLos ingresos promedios por mes son de ",
promedio_ingresos_general_mes)
                    opcion_2 = input('Regresar al menu principal "0": ')
                    # Regresar al menu principal
                    if opcion_2 == "0":
                        salir_3 = False
                    else:
                        print("\nCaracter INCORRECTO")
            elif opcion == "8":
                salir_3 = True
                while salir_3:
                    print("\nLos ingresos promedios por anio son de ",
promedio_ingresos_general_anio)
                    opcion_2 = input('Regresar al menu principal "0": ')
                    # Regresar al menu principal
                    if opcion_2 == "0":
                        salir_3 = False

```

```

else:
    print("\nCaracter INCORRECTO")
elif opcion == "9":
    salir_3 = True
    while salir_3:
        print("\n----- Ventas promedio por anio -----")
        print("\nAnio | Numero de ventas promedio")
        print("-----")
        for anio in promedios_ventas_anio:
            print(anio[0], " | ", anio[1])
        opcion_2 = input('Regresar al menu principal "0": ')
        # Regresar al menu principal
        if opcion_2 == "0":
            salir_3 = False
        else:
            print("\nCaracter INCORRECTO")
elif opcion == "10":
    salir_3 = True
    while salir_3:
        print("\n----- Meses con mas ventas al anio ----")
        print("\nMes | Anio | Numero de ventas")
        print("-----")
        for mes in ventas_por_mes:
            print(mes[0], " | ", mes[1], " | ", mes[2])
        opcion_2 = input('Regresar al menu principal "0": ')
        # Regresar al menu principal
        if opcion_2 == "0":
            salir_3 = False
        else:
            print("\nCaracter INCORRECTO")

# Interfaz del administrador
elif usuario == "administrador":
    opcion = ""
    opcion_2 = ""
    opcion_3 = ""
    print("Seleccione el numero de la opcion que desea: \n")
    print(" 1. Top n de los productos mas vendidos")
    print(" 2. Top n de los productos mas buscados ")
    print(" 3. Top de las categorias menos vendidas")
    print(" 4. Top de las categorias menos buscadas")
    print(" 5. Top n productos con mejores resenias")
    print(" 6. Top n productos con peores resenias")
    print(" 7. Ingresos promedios por mes")

```

```

print(" 8. Ingresos promedios por anio")
print(" 9. Ventas promedio por anio")
print("10. Meses con mas ventas")
print("11. Numero de ventas por categoria")
print("12. Numero de busquedas por categoria")
print("13. Analisis de productos por meses")
print("14. Stock y calificacion por categoria")
print("15. Ingresos por mes")
print(" 0. Para cambiar de sesion (salir)\n")
opcion = input("Ingrese la opcion: ")

if opcion == "1":
    salir_3 = True
    while salir_3:
        print("\nADVERTENCIA: Si ingresa algo distinto a un numero,
el programa explota ")
        print('\nPresione "0" para volver al menu principal')
        n = int(input("Ingrese el numero de productos que desea ver
el top: "))
        if n > len(total_productos_vendidos) or n < 0:
            print("\n¡ERROR! n tiene que estar entre (1,",
len(total_productos_vendidos), ")")
            continue
        elif n == 0:
            salir_3 = False
        else:
            salir_4 = True
            while salir_4:
                # Muestro el top
                print('\n----- Top ',n,' de los productos
mas vendidos ----- \n')
                print("Indice | Id del producto | Nombre | Categoria |
Precio | Calificacion | Stock | Numero Busquedas | Numero de ventas")
                print("-----")
                -----")
                lifestore_productos.sort(key = lambda ventas: ventas[5],
reverse = True)
                i = 1
                for producto in lifestore_productos[:n]:
                    print(i, " | ", producto[0], " | ", producto[1], " | ",
producto[3], " | ", producto[2], " | ", producto[7], " | ", producto[4], " | ",
producto[6], " | ", producto[5])
                    i += 1
                opcion_2 = input('Para regresar oprima "0": ')
                if opcion_2 == "0":

```

```

        salir_4 = False
    else:
        print('Caracter incorrecto')
        continue
elif opcion == "2":
    salir_3 = True
    while salir_3:
        print("\nADVERTENCIA: Si ingresa algo distinto a un numero,
el programa explota ")
        print('\nPresione "0" para volver al menu principal')
        n = int(input("Ingrese el numero de productos que desea ver
el top: "))
        if n > len(total_productos_vendidos) or n < 0:
            print("\n¡ERROR! n tiene que estar entre (1,",
len(total_productos_vendidos), ")")
            continue
        elif n == 0:
            salir_3 = False
        else:
            salir_4 = True
            while salir_4:
                # Muestro el top
                print('\n----- Top ', n, ' de los productos
mas buscados ----- \n')
                print(
                    "Indice | Id del producto | Nombre | Categoria | Precio
| Calificacion | Stock | Numero de Ventas | Numero de Busquedas")
                print(
                    "-----
-----")
                lifestore_productos.sort(key=lambda ventas: ventas[6],
reverse = True)
                i = 1
                for producto in lifestore_productos[:n]:
                    print(i, " | ", producto[0], " | ", producto[1], " | ",
producto[3], " | ", producto[2],
                        " | ", producto[7], " | ", producto[4], " | ",
producto[5], " | ", producto[6])
                    i += 1
                opcion_2 = input('Para regresar oprima "0": ')
                if opcion_2 == "0":
                    salir_4 = False
                else:
                    print('Caracter incorrecto')
                    continue

```

```

elif opcion == "3":
    salir_3 = True
    while salir_3:
        # Muestro el siguiente sub-menu
        print("\nSeleccione una categoria:\n")
        indice = 1
        opcion_2 = ""
        opcion_3 = ""
        # Hago el print para enseñar las categorias
        for categoria in categorias:
            print(indice, ". ", categoria)
            indice += 1
        print('\nADVERTENCIA: si agurega algo que no sea un
numero, el programa explota')
        print("\n0. Regresar al menu principal")
        opcion_2 = int(input('Ingrese el numero de la categoria:'))
        if opcion_2 == 0:
            salir_3 = False
        elif opcion_2 in range(1, len(lifestore_productos)):
            salir_4 = True
            while salir_4:
                categoria = categorias[opcion_2-1]
                indice = 1
                lifestore_productos.sort(key = lambda ventas: ventas[5])
                print('\n----- Top de los
productos menos vendidos ----- \n')
                print('Indice | Id | Nombre | Categoria | Precio |
Calificacion | Stock | Numero Busquedas | Numero Ventas')
                print('-----')
                for producto in lifestore_productos:
                    if categoria == producto[3]:
                        print(indice, ' | ', producto[0], ' | ', producto[1], ' | ',
producto[3], ' | ', producto[2], ' | ', producto[7], ' | ', producto[4], ' | ',
producto[6], ' | ', producto[5])
                        indice += 1
                    opcion_3 = input('\nRegresar "0": ')
                    if opcion_3 == "0":
                        salir_4 = False
                    else: print('\nCaracter INCORRECTO')
            else:
                print('\nSolo puede ingresar los numeros de las opciones
que se le muestran')
        elif opcion == "4":
            salir_3 = True
            while salir_3:

```

```

# Muestro el siguiente sub-menu
print("\nSeleccione una categoria:\n")
indice = 1
opcion_2 = ""
opcion_3 = ""
# Hago el print para enseñar las categorias
for categoria in categorias:
    print(indice, ". ", categoria)
    indice += 1
print('\nADVERTENCIA: si agurega algo que no sea un
numero, el programa explota')
print("\n0. Regresar al menu principal")
opcion_2 = int(input('Ingrese el numero de la categoria:'))
if opcion_2 == 0:
    salir_3 = False
elif opcion_2 in range(1, len(lifestore_productos)):
    salir_4 = True
    while salir_4:
        categoria = categorias[opcion_2 - 1]
        indice = 1
        lifestore_productos.sort(key=lambda busquedas:
busquedas[6])
        print(
            '\n----- Top de los
productos menos buscados ----- \n')
        print(
            'Indice | Id | Nombre | Categoria | Precio | Calificacion
| Stock | Numero Ventas | Numero Busquedas')
        print(
            '-----')
        for producto in lifestore_productos:
            if categoria == producto[3]:
                print(indice, ' | ', producto[0], ' | ', producto[1], ' | ',
producto[3], ' | ',
                    producto[2], ' | ', producto[7], ' | ', producto[4], '
| ', producto[5],
                        ' | ', producto[6])
                indice += 1
        opcion_3 = input('\nRegresar "0": ')
        if opcion_3 == "0":
            salir_4 = False
        else:
            print('\nCaracter INCORRECTO')
    else:

```

```

        print('\nSolo puede ingresar los numeros de las opciones
que se le muestran')
        elif opcion == "5":
            salir_3 = True
            while salir_3:
                productos_resenias.sort(key=lambda calificacion:
calificacion[4], reverse = True)
                print("\nADVERTENCIA: Si ingresa algo distinto a un numero,
el programa explota ")
                print('\nPresione "0" para volver al menu principal')
                n = int(input("Ingrese el numero de productos que desea ver
el top: "))
                if n > len(productos_resenias) or n < 0:
                    print("\n¡ERROR! n tiene que estar entre (1,",
len(productos_resenias), ")")
                    continue
                elif n == 0:
                    salir_3 = False
                else:
                    salir_4 = True
                    while salir_4:
                        # Muestro el top
                        print('\n----- Top ', n,' productos con
mejores resenias ----- \n')
                        print("Indice | Id del producto | Nombre | Precio |
Numero de ventas | Carificacion")
                        print("-----")
                        -----")
                        i = 1
                        for producto in productos_resenias[:n]:
                            print(i, " | ", producto[0], " | ", producto[1], " | ",
producto[3], " | ", producto[2])
                            i += 1
                        opcion_2 = input('Para regresar oprima "0": ')
                        if opcion_2 == "0":
                            salir_4 = False
                        else:
                            print('Caracter incorrecto')
                            continue
                    elif opcion == "6":
                        salir_3 = True
                        while salir_3:
                            productos_resenias.sort(key=lambda calificacion:
calificacion[4])
                            print("\nADVERTENCIA: Si ingresa algo distinto a un numero,
el programa explota ")

```



```

print('\nPresione "0" para volver al menu principal')
n = int(input("Ingrese el numero de productos que desea ver
el top: "))
if n > len(productos_resenias) or n < 0:
    print("\n¡ERROR! n tiene que estar entre (1,",
len(productos_resenias), ")")
    continue
elif n == 0:
    salir_3 = False
else:
    salir_4 = True
    while salir_4:
        # Muestro el top
        print('\n----- Top ', n, ' productos con
peores resenias ----- \n')
        print("Indice | Id del producto | Nombre | Precio |
Numero de ventas | Carificacion")
        print("-----")
        i = 1
        for producto in productos_resenias[:n]:
            print(i, " | ", producto[0], " | ", producto[1], " | ",
producto[3], " | ", producto[2])
            i += 1
        opcion_2 = input('Para regresar oprima "0": ')
        if opcion_2 == "0":
            salir_4 = False
        else:
            print('Caracter incorrecto')
            continue
    elif opcion == "7":
        salir_3 = True
        while salir_3:
            print("\nLos ingresos promedios por mes son de ",
promedio_ingresos_general_mes)
            opcion_2 = input('Regresar al menu principal "0": ')
            # Regresar al menu principal
            if opcion_2 == "0":
                salir_3 = False
            else:
                print("\nCaracter INCORRECTO")
    elif opcion == "8":
        salir_3 = True
        while salir_3:
            print("\nLos ingresos promedios por anio son de ",
promedio_ingresos_general_anio)

```

```

opcion_2 = input('Regresar al menu principal "0": ')
# Regresar al menu principal
if opcion_2 == "0":
    salir_3 = False
else:
    print("\nCaracter INCORRECTO")
elif opcion == "9":
    salir_3 = True
while salir_3:
    print("\n----- Ventas promedio por anio -----")
    print("\nAnio | Numero de ventas promedio")
    print("-----")
    for anio in promedios_ventas_anio:
        print(anio[0], " | ", anio[1])
    opcion_2 = input('Regresar al menu principal "0": ')
    # Regresar al menu principal
    if opcion_2 == "0":
        salir_3 = False
    else:
        print("\nCaracter INCORRECTO")
elif opcion == "10":
    salir_3 = True
while salir_3:
    print("\n----- Meses con mas ventas al anio ---")
    print("\nMes | Anio | Numero de ventas")
    print("-----")
    for mes in ventas_por_mes:
        print(mes[0], " | ", mes[1], " | ", mes[2])
    opcion_2 = input('Regresar al menu principal "0": ')
    # Regresar al menu principal
    if opcion_2 == "0":
        salir_3 = False
    else:
        print("\nCaracter INCORRECTO")
elif opcion == "11":
    salir_3 = True
while salir_3:
    print("\n----- Numero de ventas por categorias ---")
    print("\n")
    print("Categoria | Numero de ventas")
    print("-----")
    for categoria in ventas_por_categorias:
        print(categoria[0], " | ", categoria[1])
    opcion_2 = input('\nRegresar al menu principal "0": ')

```

```

        if opcion_2 == "0":
            salir_3 = False
        else:
            print("\nCaracter INCORRECTO")
    elif opcion == "12":
        salir_3 = True
        while salir_3:
            print("\n----- Numero de busquedas por
categorias ----- \n")
            print("Categoria | Numero de busquedas")
            print("-----")
            for categoria in busquedas_por_categorias:
                print(categoria[0], " | ", categoria[1])
            opcion_2 = input('\nRegresar al menu principal "0": ')
            if opcion_2 == "0":
                salir_3 = False
            else:
                print("\nCaracter INCORRECTO")
    elif opcion == "13":
        salir_3 = True
        while salir_3:
            print("\nSeleccione el mes que desea analizar\n")
            print("Enero 2020 (01/2020)")
            print("Febrero 2020 (02/2020)")
            print("Marzo 2020 (03/2020)")
            print("Abril 2020 (04/2020)")
            print("Mayo 2020 (05/2020)")
            print("Junio 2020 (06/2020)")
            print("Julio 2020 (07/2020)")
            print("Agosto 2020 (08/2020)")
            print("Septiembre 2020 (09/2020)")
            print("Noviembre 2019 (11/2019)")
            print("Mayo 2002 (05/2002)")
            print('\nRegresar al menu principal "0"')
            opcion_2 = input('\nIngrese el mes seleccionado como
"mm/aaaa": ')
            if opcion_2 == "0":
                salir_3 = False
            elif opcion_2 in meses_anios:
                salir_4 = True
                while salir_4:
                    # Obtengo los productos vendidos en un determinado
                    productos_mes = []
                    for venta in lifestore_ventas:
                        mes_anio = venta[3][3:]

```

```

        if opcion_2 == mes_anio:
            productos_mes.append(venta)
        # Cuento las ventas de cada producto
        productos_mes_anio = []
        ids_productos_contados = []
        for producto in productos_mes:
            id_producto = producto[1]
            nombre = producto[5]
            precio = producto[6]
            ventas = 0
            ingresos = 0
            calificacion_total = 0
            calificacion_promedio = 0
            numero_devoluciones = 0
            # Verifico que no lo haya contado ya
            if id_producto in ids_productos_contados:
                continue
            else:
                # Cuentos las ventas
                for producto_2 in productos_mes:
                    if id_producto == producto_2[1]:
                        ventas += 1
                        ingresos += producto_2[6]
                        calificacion_total += producto_2[2]
                        if producto_2[4] == "Devolucion":
                            numero_devoluciones += 1
                # Obtengo la categoria
                for producto_2 in lifestore_products:
                    if id_producto == producto_2[0]:
                        categoria = producto_2[3]
                        stock_actual = producto_2[4]
                # Obtengo el promedio de las calificaciones
                if ventas == 0 or calificacion_total == 0:
                    calificacion_promedio = 0
                else:
                    calificacion_promedio = round(calificacion_total /
ventas)

                # Agregamos los datos a una lista
                ids_productos_contados.append(id_producto)
                productos_mes_anio.append([id_producto, nombre,
categoria, precio, calificacion_promedio, ventas, numero_devoluciones,
ingresos, stock_actual])
            productos_mes_anio.sort(reverse = False, key = lambda
ventas: ventas[5])
        # Muestro la tabla de los productos

```

```

        print('\n----- Productos vendidos en ',
opcion_2, '-----\n')
        print('Indice | Id | Nombre | Categoria | Precio |
Calificacion | Numero de Ventas | Numero de Devoluciones | Ingresos |
Stock Actual')
        print('-----
-----')
        indice = 0
        for producto in productos_mes_anio:
            indice += 1
            print(indice, ' | ', producto[0], ' | ', producto[1], ' | ',
producto[2], ' | ', producto[3], ' | ', producto[4], ' | ', producto[5], ' | ',
producto[6], ' | ', producto[7], ' | ', producto[8])
            opcion_3 = input('\nIngrese "0" para regresar: ')
            if opcion_3 == "0":
                salir_4 = False
            else:
                print('\nCaracter INCORRECTO')
        else:
            print('\nCaracter ICNORRECTO')
    elif opcion == "14":
        salir_3 = True
        while salir_3:
            print('\n----- Stock por categoria -----
-----\n')
            print(' Categoria | Stock | Calificacion')
            print("-----")
            for categoria in stock_categorias:
                print(categoria[0], ' | ', categoria[1], ' | ', categoria[2])
            opcion_2 = input('\nRegresar al menu principal "0"')
            if opcion_2 == "0":
                salir_3 = False
            else:
                print('\nCaracter INCORRECTO')
    elif opcion == "15":
        salir_3 = True
        while salir_3:
            print('\n----- Ingresos por mes -----
-----\n')
            print('Mes | Anio | Ingresos')
            print('-----')
            for mes in ingresos_por_mes:
                print(mes[0], ' | ', mes[1], ' | ', mes[2])
            opcion_2 = input('\nResegrar al menu principal "0": ')
            if opcion_2 == "0":

```

```
        salir_3 = False
    else:
        print('\nCaracter INCORRECTO')
    elif opcion == "0":
        salir_2 = False
# Regreso al inicio de sesion
if opcion == "0":
    usuario = ""
    contrasenia = ""
    contador = 0
    continue
```

Conclusión

En conclusión, se logró cumplir el objetivo del análisis, es decir, brindar muy buenas recomendaciones que la tienda puede llevar a cabo para aumentar sus ventas y tener una mejor rotación de los productos que se tienen en el inventario.

Se propusieron mecanismos para prevención del rezago de mercancía, se dieron meses clave en los que se debe aplicar o no un descuento y a cuáles productos hay que aplicárselo. Al igual que a su vez, se brindan recomendaciones para la eliminación de productos en el catálogo y sugerencias para expansión del catálogo.

Finalmente, se explicó de manera general y de manera muy detallada el funcionamiento de la herramienta utilizada para obtener información para el análisis.