



Proyecto 1: INTRODUCCIÓN A PYTHON

Caso práctico LIFESTORE.



16 DE SEPTIEMBRE DE 2021

Marisol Vazquez Herrera

Contenido

Introducción:	2
Definición del código.....	3
Solución del problema (resultados de código):	18
Solución parte 1	18
Solución parte 2	25
Solución parte 3	27
Conclusiones:	28

Introducción:

En el presente reporte se describirá el procedimiento y código creado mediante el lenguaje de Python, con la finalidad de realizar un análisis de las ventas, búsquedas, reseñas e ingresos de los diferentes artículos vendidos por la tienda virtual Lifestore, debido a que se ha detectado una acumulación de inventarios, así como una reducción en las búsquedas de un grupo importante de productos, lo que conlleva a una disminución sustancial de ventas y a su vez un menor ingreso.

El análisis se divide en tres partes:

- En la primera parte se determina el análisis para los productos más vendidos, más buscados y de acuerdo con la categoría del producto se enlistan los productos de menor a mayores ventas y búsquedas.
- En la segunda parte se analizan los productos de acuerdo con las mejores y peores reseñas de los clientes.
- En la ultima parte se analiza el total de ingresos y ventas mensuales y anuales, así como los meses con mayores ventas.

Parte 1: productos más vendidos y rezagados

Parte importante del análisis consiste en un listado de los 20 productos más vendidos, se enlistan solo 20 productos porque la lista de ventas contiene 41 productos diferentes, por lo tanto, se consideran 20 como la mitad de los productos que más se venden. Además de describir los productos más vendidos, se enlista el número de ventas realizadas por dicho producto, ya que se considera importante la diferencia existente entre la cantidad vendida entre dichos productos, no solo el nombre. Por lo que la lista esta formada por: id_producto, nombre de producto y número de ventas.

Adicional a los 20 productos más vendidos, se enlistan los 25 productos con mayor número de búsquedas debido a que solo se buscaron 56 productos totales. De igual forma se considero importante colocar el id_del producto, el nombre de dicho producto y el número de búsquedas para dicho producto para considerar la diferencia de búsquedas entre un producto y otro.

Para el análisis por categoría, se enlistan los productos que tuvieron menores a mayores ventas de dicha categoría. En estas listas adicional al id y nombre del producto se incluyen el número de ventas y el nombre de la categoría.

Parte 2: Productos por reseña en el servicio.

Se enlistan los 20 productos que tienen las mejores reseñas, para obtener las mejores reseñas, se sumaron el número asignado en cada reseña para cada producto y se ordenaron los que cuentan con mayores y menores puntos. En las listas se agrego el id del producto, el nombre, y el número de reseñas por cada categoría.

Parte 3: Total de ingresos, anuales, mensuales y meses con mejores ventas.

En este apartado se analizan los ingresos mensuales obtenidos por determinado número de ventas y devoluciones. Cabe resaltar que si el cliente devolvió algún producto se considero que se regreso el dinero pagado es decir disminuyen los ingresos. Se enlistan los ingresos totales por mes desde enero hasta septiembre debido que son los meses que cuentan con datos de ventas, así como el número de ventas y devoluciones para poder analizar dichos comportamientos.

Adicional a los ingresos mensuales, se muestra el ingreso anual total, donde se analizo solamente el año 2020 ya que se cuenta con un solo dato del 2019.

Finalmente se enlistan los tres meses con mayores ventas aunado del número de ventas de dichos meses.

Definición del código

Se copiará el código y se ira comentando cada parte de este.

```
#importar base de datos
fromm lifestore_file import lifestore_products
from lifestore_file import lifestore_sales
from lifestore_file import lifestore_searches
```

```
#Iniciar sesión (solo se da acceso al usuario que aparezca en la lista de
usuarios e ingrese la contraseña correcta). Solo te permite 3 intentos para
introducir el usuario y una para la contraseña, si no se sale del programa.
#La lista de usuarios-contraseña está formada por [el usuario, seguido de su
contraseña] y es la siguiente:
lista_usuarios=[["administrador1","ingresar"],["administrador2","12345"],
["administrador3","contraseña"]]
#El programa te da dos opciones (1) ingresar si ya cuentas con un usuario y
(2) crear un usuario, debes seleccionar la que deseas
```

```

print("Selecciona un opción: ")
print("(1) Iniciar sesión\n(2) Crear usuario")
registrar_usuario=input('Opción: ')
#De acuerdo con tu selección te mandara a diferentes procesos. Si seleccionas 1 te pedirá que
ingreses usuario y contraseña.
if registrar_usuario== "1":
    usuario=input("Ingresa nombre de usuario: ")
    contraseña=input("Ingresa contraseña: ")
    # Si seleccionas 2 te pedirá que ingreses nombre y contraseña de nuevo usuario, posteriormente
    te pide que ingreses sesión.
elif registrar_usuario=="2":
    print("Primero crea el usuario y contraseña, después inicia sesión")
    nuevo_usuario=input("Ingresa nombre de usuario: ")
    nueva_contraseña=input("Ingresa contraseña: ")
    lista_usuarios.append([nuevo_usuario,nueva_contraseña])
    print("Ahora inicia sesión")
    usuario=input("Ingresa nombre de usuario: ")
    contraseña=input("Ingresa contraseña: ")

else: #si no escribes las opciones correctas se saldrá del proyecto
    print('Vuelve a intentarlo Escribe (1) si ya cuentas con un usuario o(2)
    si deseas registrar nuevo usuario')
    exit()

es_admi=0 #Es =0 Si son incorrectos los datos de usuario y contraseña. Es=1,
si son correctos
intentos=0 # Número de intentos
#Tienes 3 intentos como máximo para escribir tu usuario y contraseña
correctos
while es_admi != 1 and intentos < 2:
    for admi in lista_usuarios:
        if admi[0]== usuario and admi[1]==contraseña:
            es_admi=1
    if es_admi==0:
        print("Datos incorrectos")
        usuario=input("Ingresa nombre de usuario: ")
        contraseña=input("Ingresa contraseña: ")
        intentos+=1

es_admi=1
if es_admi == 1:
    #Primer consigna: Productos
    intentos=0
    opcion=0

```

#ya que ingresaste te pide que coloques la opción que desees (parte1, parte2 o parte3) descritas en la introducción, cuentas con 3 oportunidades para ingresar la opción correcta.

```
while opcion != 1 and opcion !=2 and opcion !=3 and intentos<3:
```

```
    print("Selecciona un opción: ")
```

```
    print("(1) 20 Productos más vendidos, 25 productos con más búsquedas, productos ordenados con menores ventas y búsquedas por categoria.\n(2) Lista de productos con mejores reseñas y lista con peores reseñas.\n(3) total de ingresos y ventas promedio mensuales, total anual y meses con más ventas al año")
```

```
    opcion=int(input("Opción: "))
```

```
    intentos+=1
```

```
    if intentos==3 and opcion !=1 and opcion !=2 and opcion !=3:
```

```
        print("Lo sentimos, vuelva a intentarlo y elija una opcion correcta")
```

#para la primera parte (opción 1)

```
    if opcion==1:
```

#se cuentan los productos vendidos, por cada venta, si se devolvió el producto se cuenta como -1 ventas

```
    contador=0
```

```
    ventas_producto=[] #[id_producto,num_ventas]
```

```
    for producto in lifestore_products:
```

```
        for venta in lifestore_sales:
```

```
            if producto[0]==venta[1] and venta[4]==0:
```

```
                contador+=1 #Suma 1 venta si se vendio y no se devolvio
```

```
            elif producto[0]==venta[1] and venta[4]==1:
```

```
                contador-=1 #Resta 1 venta si se vendio y se devolvio
```

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

```
        ventas_producto.append([producto[0],producto[1],contador])
```

```
    contador=0
```

#Ordenar la lista de productos de mayor a menor ventas

```
    productos_ordenados_mayor=[]
```

```
    while ventas_producto:
```

```
        maximo=ventas_producto[0][2]
```

```
        lista_max=ventas_producto[0]
```

```
        for totalventa in ventas_producto:
```

```
            if totalventa[2]>maximo:
```

```
                maximo=totalventa[2]
```

```
                lista_max=totalventa
```

```
    productos_ordenados_mayor.append(lista_max)
```

```
    ventas_producto.remove(lista_max)
```

#se cuentan cuantos elementos tienen la lista para ordenarlos de menor a mayor.

```
productos_ordenados_menor=[]
tamaño=len(productos_ordenados_mayor)
#se ordenan los productos de menor a mayor ventas
for i in range(1,tamaño+1):
    productos_ordenados_menor.append(productos_ordenados_mayor[tamaño-i])
```

```
# solo se enlistan los 20 productos más vendidos
veint_productos_mas_vendidos=[]
for j in range(0,20):
    veint_productos_mas_vendidos.append(productos_ordenados_mayor[j])
print('Los 20 productos más vendidos son:["id_producto",
"nombre_producto","#ventas" ')
print(veint_productos_mas_vendidos) #se imprimen solo los 20 productos
con mayores ventas
```

#se hace el proceso de los productos con mayores búsquedas, en este caso todos los elementos suman búsquedas

```
contador=0
búsquedas_producto=[] #[id_producto,num_búsquedas]
for producto in lifestore_products:
    for busqueda in lifestore_searches:
        if producto[0]==busqueda[1]:
            contador+=1 #Suma 1 búsqueda si se buscó el producto
if contador !=0:
    búsquedas_producto.append([producto[0],producto[1],contador])
    contador=0
```

Se ordena la lista de búsquedas de mayor a menor

```
productos_busq_mayor=[]
while búsquedas_producto:
    maximo=búsquedas_producto[0][2]
    lista_max=búsquedas_producto[0]
    for totalbusqueda in búsquedas_producto:
        if totalbusqueda[2]>maximo:
            maximo=totalbusqueda[2]
            lista_max=totalbusqueda
    productos_busq_mayor.append(lista_max)
    búsquedas_producto.remove(lista_max)
#print(len(productos_busq_mayor))=56
```

#productos con menores búsquedas ordenados de menor a mayor

```
productos_búsquedas_ordenados_menor=[]
```

```

    tamaño=len(productos_busq_mayor) #se cuentan los productos buscados
    #se ordenan las busquedas de menor a mayor
    for p in range(1,tamaño+1):

productos_busquedas_ordenados_menor.append(productos_busq_mayor[tamaño-p])

    # se muestran solo los 25 productos con más Búsquedas
    veinticinco_productos_mas_buscados=[]
    for l in range(0,25):
        veinticinco_productos_mas_buscados.append(productos_busq_mayor[l])
    print('Los 25 productos más buscados son: ["id_producto",
"nombre_producto", "# busquedas"] ')
    print(veinticinco_productos_mas_buscados)

    #se enlista las categorías para poder clasificar las ventas y búsquedas
    por categoría
    categorias=[]
    categoria1=''
    for producto in lifestore_products: # en cada producto se ve la
categoría
        if categoria1==producto[3]: #si la categoría del producto es la misma
se continua
            continue
        else:
            categorias.append(producto[3]) #si es diferente se agrega a la lista
            categoria1=producto[3]
    #print(categorias)# ['procesadores','tarjetas de video','tarjetas
madre','discos duros','memorias usb','pantallas','bocinas','audifonos']

    #Se asigno a una lista ordenada de menor a mayores ventas igual a la
lista de ventas de menor a mayor para agregar la categoría
    productos_ordenados_menor2=productos_ordenados_menor
    #se agrega la categoría a la lista ordenada de productos con menores ventas
a mayores
    for elemento in lifestore_products:
        for producto in productos_ordenados_menor2:
            if producto[0]==elemento[0]:
                producto.append(elemento[3])
    lista_categorias_menores_ventas=[]

    #se muestran los productos de cada categoría ordenados de menor a mayor
ventas
    for categoria in categorias:

```



```

    for ventas in productos_ordenados_menor2:
        if ventas[3]==categoria:
            lista_categorias_menores_ventas.append(ventas)
            print("La lista esta conformada por ['id_producto','nombre_del
producto','# ventas, 'categoria']\nLos productos con menores ventas de la
categoria", categoria, " son: ")
            print(lista_categorias_menores_ventas)
            lista_categorias_menores_ventas
            lista_categorias_menores_ventas=[] #se elimina la lista para entrar a
la siguiente categoría en blanco

    #Se asigno a una lista ordenada de menor a mayores búsquedas igual a la
lista de búsquedas de menor a mayor para agregar la categoría
    productos_busq_ordenados_menor2=productos_busquedas_ordenados_menor
    #print(productos_busq_ordenados_menor2)
    #se agrega la categoría a la lista de búsquedas ordenadas de menor a mayor
    for elemento in lifestore_products:
        for producto in productos_busq_ordenados_menor2:
            if producto[0]==elemento[0]:
                producto.append(elemento[3])
            lista_categorias_menores_busq=[]
            #print(productos_busq_ordenados_menor2)
    #se muestran los productos con menores búsquedas por categoria
    for categoria in categorias:
        for busqueda in productos_busq_ordenados_menor2:
            if busqueda[3]==categoria:
                lista_categorias_menores_busq.append(busqueda)
    #se muestra la lista de productos de menor a mayor #busqueda por categoria
    print("La lista esta conformada por ['id_producto','nombre_del
producto','# busquedas, 'categoria']\nLos productos con menores busquedas de
la categoria", categoria, " son: ")

    print(lista_categorias_menores_busq)
    lista_categorias_menores_busq=[]

#si se elige la opción 2
    if opcion==2:
        contador=0
    #se cuenta mediante un ciclo for la suma de puntos de cada reseña
    reseñas_producto=[] #[id_producto,nombre del producto,reseña]
    for producto in lifestore_products:
        for venta in lifestore_sales:
            if producto[0]==venta[1]:
                contador+=int(venta[2]) #Suma el valor de la reseña escrito en la
venta o devolucion.

```

```

        if contador !=0:
            reseñas_producto.append([producto[0],producto[1],contador])
            contador=0
    #print(reseñas_producto)

#ordenar productos con mejores reseñas de mayor a menor
productos_mayor_reseña=[]
while reseñas_producto:
    maximo=reseñas_producto[0][2]
    lista_max=reseñas_producto[0]
    for totalreseñas in reseñas_producto:
        if totalreseñas[2]>maximo:
            maximo=totalreseñas[2]
            lista_max=totalreseñas
    productos_mayor_reseña.append(lista_max)
    reseñas_producto.remove(lista_max)
#print(len(productos_mayor_reseña))#tiene 42 articulos vendidos
#print(productos_mayor_reseña)

#productos con menores reseñas ordenados de menor a mayor
productos_peor_reseña=[]
tamaño=len(productos_mayor_reseña)
for t in range(1,tamaño+1):
    productos_peor_reseña.append(productos_mayor_reseña[tamaño-t])
#print(productos_peor_reseña)

#20 productos con mejor reseñas
veinte_productos_mejor_reseña=[]
for s in range(0,20):
    veinte_productos_mejor_reseña.append(productos_mayor_reseña[s])
    print('Los 20 productos con mejor reseña son: ["id_producto",
"nombre_producto","reseñas"]')
    print(veinte_productos_mejor_reseña)

#20 productos con peores reseñas
veinte_productos_peor_reseña=[]
for o in range(0,20):
    veinte_productos_peor_reseña.append(productos_peor_reseña[o])
    print('Los 20 productos con peor reseña son: ["id_producto",
"nombre_producto","reseñas"]')
    print(veinte_productos_peor_reseña)

#opción 3 total de ingresos y ventas promedio mensuales, total anual y
meses con más ventas al año

```

```

if opcion==3:

    #ingresos mensuales
    #enero
    contador=0
    sales1=0
    devoluciones1=0
    total_ingresos_mes1=0
    #asignas una lista para las ventas del mes de enero
    ventas_ingreso_producto_mes1=[] #[id_producto,nombre del
    producto,fecha_venta,ingresos]
    #cuenta las ventas del mes de enero para el año 2020 y le resta las
    devoluciones
    for producto in lifestore_products:
        for venta in lifestore_sales:
            if producto[0]==venta[1] and venta[4]==0 and venta[3][3:5]=="01" and
            venta[3][6:10]=="2020":
                contador+=1 #Suma 1 venta si se vendio y no se devolvio
                sales1+=1
            elif producto[0]==venta[1] and venta[4]==1 and venta[3][3:5]=="01"
            and venta[3][6:10]=="2020":
                contador-=1 #Resta 1 venta si se vendio y se devolvio
                devoluciones1+=1

        if contador !=0:
            #cuenta el ingreso de ventas
            ventas_ingreso_producto_mes1.append([producto[0],producto[1],contador,int(pr
            oducto[2])*contador])
            total_ingresos_mes1+=int(producto[2])*contador
            contador=0
    #muestra los ingresos con el numero de ventas y devoluciones
    print("El total de ingresos para el mes 01 de 2020 es:
    $",total_ingresos_mes1, "de ", sales1," ventas",
    "y " , devoluciones1, " devoluciones")
    #calcula los ingresos mes febrero
    #febrero
    contador=0
    sales2=0
    devoluciones2=0
    total_ingresos_mes2=0
    ventas_ingreso_producto_mes2=[] #[id_producto,nombre del
    producto,fecha_venta,ingresos]

    for producto in lifestore_products:

```

```

        for venta in lifestore_sales:
            if producto[0]==venta[1] and venta[4]==0 and venta[3][3:5]=="02" and
venta[3][6:10]=="2020":
                contador+=1 #Suma 1 venta si se vendio y no se devolvio
                sales2+=1
            elif producto[0]==venta[1] and venta[4]==1 and venta[3][3:5]=="02"
and venta[3][6:10]=="2020":
                contador-=1 #Resta 1 venta si se vendio y se devolvio
                devoluciones2+=1

        if contador !=0:

ventas_ingreso_producto_mes2.append([producto[0],producto[1],contador,int(pr
oducto[2])*contador])
        total_ingresos_mes2+=int(producto[2])*contador
        contador=0
        print("El total de ingresos para el mes 02 de 2020 es:
$",total_ingresos_mes2, "de ", sales2, " ventas",
        "y " , devoluciones2, " devoluciones")

#Marzo
        contador=0
        sales3=0
        devoluciones3=0
        total_ingresos_mes3=0
        ventas_ingreso_producto_mes3=[] #[id_producto,nombre del
producto,fecha_venta,ingresos]

        for producto in lifestore_products:
            for venta in lifestore_sales:
                if producto[0]==venta[1] and venta[4]==0 and venta[3][3:5]=="03" and
venta[3][6:10]=="2020":
                    contador+=1 #Suma 1 venta si se vendio y no se devolvio
                    sales3+=1
                elif producto[0]==venta[1] and venta[4]==1 and venta[3][3:5]=="03"
and venta[3][6:10]=="2020":
                    contador-=1 #Resta 1 venta si se vendio y se devolvio
                    devoluciones3+=1

        if contador !=0:

ventas_ingreso_producto_mes3.append([producto[0],producto[1],contador,int(pr
oducto[2])*contador])
        total_ingresos_mes3+=int(producto[2])*contador
        contador=0

```

```

    print("El total de ingresos para el mes 03 de 2020 es:
$",total_ingresos_mes3,"de ", sales3," ventas",
    "y " , devoluciones3, " devoluciones")

#Abril
contador=0
sales4=0
devoluciones4=0
total_ingresos_mes4=0
ventas_ingreso_producto_mes4=[] #[id_producto,nombre del
producto,fecha_venta,ingresos]

for producto in lifestore_products:
    for venta in lifestore_sales:
        if producto[0]==venta[1] and venta[4]==0 and venta[3][3:5]=="04" and
venta[3][6:10]=="2020":
            contador+=1 #Suma 1 venta si se vendio y no se devolvio
            sales4+=1
            elif producto[0]==venta[1] and venta[4]==1 and venta[3][3:5]=="04"
and venta[3][6:10]=="2020":
                contador-=1 #Resta 1 venta si se vendio y se devolvio
                devoluciones4+=1

    if contador !=0:

ventas_ingreso_producto_mes4.append([producto[0],producto[1],contador,int(pr
oducto[2])*contador])
        total_ingresos_mes4+=int(producto[2])*contador
        contador=0
    print("El total de ingresos para el mes 04 de 2020 es:
$",total_ingresos_mes4,"de ", sales4," ventas",
    "y " , devoluciones4, " devoluciones")

#Mayo
contador=0
sales5=0
devoluciones5=0
total_ingresos_mes5=0
ventas_ingreso_producto_mes5=[] #[id_producto,nombre del
producto,fecha_venta,ingresos]

for producto in lifestore_products:
    for venta in lifestore_sales:
        if producto[0]==venta[1] and venta[4]==0 and venta[3][3:5]=="05" and
venta[3][6:10]=="2020":

```

```

        contador+=1 #Suma 1 venta si se vendio y no se devolvio
        sales5+=1
        elif producto[0]==venta[1] and venta[4]==1 and venta[3][3:5]=="05"
and venta[3][6:10]=="2020":
            contador-=1 #Resta 1 venta si se vendio y se devolvio
            devoluciones5+=1

    if contador !=0:

ventas_ingreso_producto_mes5.append([producto[0],producto[1],contador,int(pr
oducto[2])*contador])
        total_ingresos_mes5+=int(producto[2])*contador
        contador=0
        print("El total de ingresos para el mes 05 de 2020 es:
$",total_ingresos_mes5,"de ", sales5," ventas",
        "y " , devoluciones5, " devoluciones")

#Junio
contador=0
sales6=0
devoluciones6=0
total_ingresos_mes6=0
ventas_ingreso_producto_mes6=[] #[id_producto,nombre del
producto,fecha_venta,ingresos]

for producto in lifestore_products:
    for venta in lifestore_sales:
        if producto[0]==venta[1] and venta[4]==0 and venta[3][3:5]=="06" and
venta[3][6:10]=="2020":
            contador+=1 #Suma 1 venta si se vendio y no se devolvio
            sales6+=1
            elif producto[0]==venta[1] and venta[4]==1 and venta[3][3:5]=="06"
and venta[3][6:10]=="2020":
                contador-=1 #Resta 1 venta si se vendio y se devolvio
                devoluciones6+=1

    if contador !=0:

ventas_ingreso_producto_mes6.append([producto[0],producto[1],contador,int(pr
oducto[2])*contador])
        total_ingresos_mes6+=int(producto[2])*contador
        contador=0
        print("El total de ingresos para el mes 06 de 2020 es:
$",total_ingresos_mes6,"de ", sales6," ventas",
        "y " , devoluciones6, " devoluciones")

```

```

#Julio
contador=0
sales7=0
devoluciones7=0
total_ingresos_mes7=0
ventas_ingreso_producto_mes7=[] #[id_producto,nombre del
producto,fecha_venta,ingresos]

for producto in lifestore_products:
    for venta in lifestore_sales:
        if producto[0]==venta[1] and venta[4]==0 and venta[3][3:5]=="07" and
venta[3][6:10]=="2020":
            contador+=1 #Suma 1 venta si se vendio y no se devolvio
            sales7+=1
        elif producto[0]==venta[1] and venta[4]==1 and venta[3][3:5]=="07"
and venta[3][6:10]=="2020":
            contador-=1 #Resta 1 venta si se vendio y se devolvio
            devoluciones7+=1

    if contador !=0:

ventas_ingreso_producto_mes7.append([producto[0],producto[1],contador,int(pr
oducto[2])*contador])
        total_ingresos_mes7+=int(producto[2])*contador
        contador=0
    print("El total de ingresos para el mes 07 de 2020 es:
$",total_ingresos_mes7, "de ", sales7," ventas",
      "y " , devoluciones7, " devoluciones")

#Agosto
contador=0
sales8=0
devoluciones8=0
total_ingresos_mes8=0
ventas_ingreso_producto_mes8=[] #[id_producto,nombre del
producto,fecha_venta,ingresos]

for producto in lifestore_products:
    for venta in lifestore_sales:
        if producto[0]==venta[1] and venta[4]==0 and venta[3][3:5]=="08" and
venta[3][6:10]=="2020":
            contador+=1 #Suma 1 venta si se vendio y no se devolvio
            sales8+=1

```

```

        elif producto[0]==venta[1] and venta[4]==1 and venta[3][3:5]=="08"
and venta[3][6:10]=="2020":
            contador-=1 #Resta 1 venta si se vendio y se devolvio
            devoluciones8+=1

    if contador !=0:

ventas_ingreso_producto_mes8.append([producto[0],producto[1],contador,int(pr
oducto[2])*contador])
        total_ingresos_mes8+=int(producto[2])*contador
        contador=0
        print("El total de ingresos para el mes 08 de 2020 es:
$",total_ingresos_mes8, "de ", sales8," ventas",
        "y " , devoluciones8, " devoluciones")

#Septiembre
    contador=0
    sales9=0
    devoluciones9=0
    total_ingresos_mes9=0
    ventas_ingreso_producto_mes9=[] #[id_producto,nombre del
producto,fecha_venta,ingresos]

    for producto in lifestore_products:
        for venta in lifestore_sales:
            if producto[0]==venta[1] and venta[4]==0 and venta[3][3:5]=="09" and
venta[3][6:10]=="2020":
                contador+=1 #Suma 1 venta si se vendio y no se devolvio
                sales9+=1
            elif producto[0]==venta[1] and venta[4]==1 and venta[3][3:5]=="09"
and venta[3][6:10]=="2020":
                contador-=1 #Resta 1 venta si se vendio y se devolvio
                devoluciones9+=1

    if contador !=0:

ventas_ingreso_producto_mes9.append([producto[0],producto[1],contador,int(pr
oducto[2])*contador])
        total_ingresos_mes9+=int(producto[2])*contador
        contador=0
        print("El total de ingresos para el mes 09 de 2020 es:
$",total_ingresos_mes9, "de ", sales9," ventas",
        "y " , devoluciones9, " devolución")

#Total de ingresos anuales 2020

```



```

    contador=0
    total_ingresos=0
#calcula el ingreso para el año 2020, multiplicando las ventas*precio
    ventas_ingreso_producto=[] #[id_producto,nombre del
    producto,fecha_venta,ingresos]
    for producto in lifestore_products:
        for venta in lifestore_sales:
            if producto[0]==venta[1] and venta[4]==0 and venta[3][6:10]=="2020":
                contador+=1 #Suma 1 venta si se vendio y no se devolvio
            elif producto[0]==venta[1] and venta[4]==1 and
venta[3][6:10]=="2020":
                contador-=1 #Resta 1 venta si se vendio y se devolvio

        if contador !=0:

ventas_ingreso_producto.append([producto[0],producto[1],venta[3],int(product
o[2])*contador])
            total_ingresos+=int(producto[2])*contador
            contador=0

        #total_ingresos+=ventas_ingreso_producto[3]
#muestra el total de ingresos
        #print(ventas_ingreso_producto)
        print("El total de ingresos para el año 2020 es: $",total_ingresos)

        #3 meses con mayores ventas
#se asigna la lista de ventas con los meses para posteriormente incluir
ventas e ingresos

lista_ventas_ingreso=[["Enero"],["Febrero"],["Marzo"],["Abril"],["Mayo"],["J
unio"],["Julio"],["Agosto"],["Septiembre"]]
    #se agrega a cada mes, las ventas e ingresos
    #La lista lista_ventas_ingreso contiene
[[enero,ventas,ingresos],[febrero,ventas,ingresos],[marzo,ventas,ingresos],[
abril,ventas,ingresos],[mayo,ventas,ingresos],[junio,ventas,ingresos],[julio
,ventas,ingresos],[agosto,ventas,ingresos],[septiembre,ventas,ingresos]]
    #mes1
    lista_ventas_ingreso[0].append(sales1-devoluciones1)
    lista_ventas_ingreso[0].append(total_ingresos_mes1)
    #mes2
    lista_ventas_ingreso[1].append(sales2-devoluciones2)
    lista_ventas_ingreso[1].append(total_ingresos_mes2)
    #mes3
    lista_ventas_ingreso[2].append(sales3-devoluciones3)
    lista_ventas_ingreso[2].append(total_ingresos_mes3)

```

```

#mes4
lista_ventas_ingreso[3].append(sales4-devoluciones4)
lista_ventas_ingreso[3].append(total_ingresos_mes4)
#mes2
lista_ventas_ingreso[4].append(sales5-devoluciones5)
lista_ventas_ingreso[4].append(total_ingresos_mes5)
#mes6
lista_ventas_ingreso[5].append(sales6-devoluciones6)
lista_ventas_ingreso[5].append(total_ingresos_mes6)
#mes7
lista_ventas_ingreso[6].append(sales7-devoluciones7)
lista_ventas_ingreso[6].append(total_ingresos_mes7)
#mes8
lista_ventas_ingreso[7].append(sales8-devoluciones8)
lista_ventas_ingreso[7].append(total_ingresos_mes8)
#mes9
lista_ventas_ingreso[8].append(sales9-devoluciones9)
lista_ventas_ingreso[8].append(total_ingresos_mes9)

#ordenar lista de meses por mayor a menor venta
#print(lista_ventas_ingreso)
mayor_ventas_mes=[]
while lista_ventas_ingreso:
    maximo=lista_ventas_ingreso[0][1]
    lista_max=lista_ventas_ingreso[0]
    for mes in lista_ventas_ingreso:
        if mes[1]>maximo:
            maximo=mes[1]
            lista_max=mes
    mayor_ventas_mes.append(lista_max)
    lista_ventas_ingreso.remove(lista_max)

#lista 3 meses mejores ventas
lista_tres_meses_mas_ventas=[]
for e in range(0,3):
    lista_tres_meses_mas_ventas.append(mayor_ventas_mes[e][0:2])

print("Los 3 meses con mayores ventas fueron [Mes,ventas]: ")
print(lista_tres_meses_mas_ventas)

```

Si considera más conveniente checar el código en replit lo puede observar en el siguiente enlace:

<https://replit.com/@losiram13/PROYECTO-01-VAZQUEZ-HERRERA-MARISOL#main.py>

Solución del problema (resultados de código):

Solución parte 1

Los 20 productos más vendidos son: ["id_producto", "nombre_producto", "#ventas"]

"id_producto",	"nombre_producto"	"#ventas"
54	"SSD Kingston A400, 120GB, SATA III, 2.5", 7mm"	48
3	'Procesador AMD Ryzen 5 2600, S-AM4, 3.40GHz, Six-Core, 16MB L3 Cache, con Disipador Wraith Stealth'	42
5	, 'Procesador Intel Core i3-9100F, S-1151, 3.60GHz, Quad-Core, 6MB Cache (9na. Generación - Coffee Lake)'	20
42	'Tarjeta Madre ASRock Micro ATX B450M Steel Legend, S-AM4, AMD B450, HDMI, 64GB DDR4 para AMD'	18
57	"SSD Adata Ultimate SU800, 256GB, SATA III, 2.5", 7mm"	15
4	'Procesador AMD Ryzen 3 3200G con Gráficos Radeon Vega 8, S-AM4, 3.60GHz, Quad-Core, 4MB L3, con Disipador Wraith Spire'	13
29	'Tarjeta Madre ASUS micro ATX TUF B450M-PLUS GAMING, S-AM4, AMD B450, HDMI, 64GB DDR4 para AMD'	12
2	'Procesador AMD Ryzen 5 3600, S-AM4, 3.60GHz, 32MB L3 Cache, con Disipador Wraith Stealth'	11
47	'SSD XPG SX8200 Pro, 256GB, PCI Express, M.2'	11
12	'Tarjeta de Video ASUS NVIDIA GeForce GTX 1660 SUPER EVO OC, 6GB 192-bit GDDR6, PCI Express x16 3.0'	9
48	'SSD Kingston A2000 NVMe, 1TB, PCI Express 3.0, M2'	9
7	'Procesador Intel Core i7-9700K, S-1151, 3.60GHz, 8-Core, 12MB Smart Cache (9na. Generación Coffee Lake)'	7
44	'Tarjeta Madre MSI ATX B450 TOMAHAWK MAX, S-AM4, AMD B450, 64GB DDR4 para AMD'	6
18	'Tarjeta de Video Gigabyte NVIDIA GeForce GT 1030, 2GB 64-bit GDDR5, PCI Express x16 3.0'	5
8	'Procesador Intel Core i5-9600K, S-1151, 3.70GHz, Six-Core, 9MB Smart Cache (9na. Generación - Coffee Lake)'	4
6	'Procesador Intel Core i9-9900K, S-1151, 3.60GHz, 8-Core, 16MB Smart Cache (9na. Generación Coffee Lake)'	3
11	'Tarjeta de Video ASUS AMD Radeon RX 570, 4GB 256-bit GDDR5, PCI Express 3.0'	3
49	'Kit SSD Kingston KC600, 1TB, SATA III, 2.5, 7mm'	3
51	'SSD Kingston UV500, 480GB, SATA III, mSATA'	3
1	'Procesador AMD Ryzen 3 3300X S-AM4, 3.80GHz, Quad-Core, 16MB L2 Cache'	2

Los 25 productos más buscados son: ["id_producto", "nombre_producto", "# busquedas"]

"id_producto",	"nombre_producto"	"# búsquedas"
54	"SSD Kingston A400, 120GB, SATA III, 2.5", 7mm"	263
57	"SSD Adata Ultimate SU800, 256GB, SATA III, 2.5" 7mm"	107
29	'Tarjeta Madre ASUS micro ATX TUF B450M-PLUS GAMING, S-AM4, AMD B450, HDMI, 64GB DDR4 para AMD'	60
3	'Procesador AMD Ryzen 5 2600, S-AM4, 3.40GHz, Six-Core, 16MB L3 Cache, con Disipador Wraith Stealth'	55
4	'Procesador AMD Ryzen 3 3200G con Gráficos Radeon Vega 8, S-AM4, 3.60GHz, Quad-Core, 4MB L3, con Disipador Wraith Spire'	41
85	'Logitech Audífonos Gamer G635 7.1, Alámbrico, 1.5 Metros, 3.5mm, Negro/Azul'	35
67	'TV Monitor LED 24TL520S-PU 24, HD, Widescreen, HDMI, Negro'	32
7	'Procesador Intel Core i7-9700K, S-1151, 3.60GHz, 8-Core, 12MB Smart Cache (9na. Generación Coffee Lake)'	31
5	'Procesador Intel Core i3-9100F, S-1151, 3.60GHz, Quad-Core, 6MB Cache (9na. Generación - Coffee Lake)'	30
47	'SSD XPG SX8200 Pro, 256GB, PCI Express, M.2'	30
48	'SSD Kingston A2000 NVMe, 1TB, PCI Express 3.0, M2'	27
44	'Tarjeta Madre MSI ATX B450 TOMAHAWK MAX, S-AM4, AMD B450, 64GB DDR4 para AMD'	25
2	'Procesador AMD Ryzen 5 3600, S-AM4, 3.60GHz, 32MB L3 Cache, con Disipador Wraith Stealth'	24
42	'Tarjeta Madre ASRock Micro ATX B450M Steel Legend, S-AM4, AMD B450, HDMI, 64GB DDR4 para AMD'	23
8	'Procesador Intel Core i5-9600K, S-1151, 3.70GHz, Six-Core, 9MB Smart Cache (9na. Generación - Coffee Lake)'	20
12	'Tarjeta de Video ASUS NVIDIA GeForce GTX 1660 SUPER EVO OC, 6GB 192-bit GDDR6, PCI Express x16 3.0'	15
21	'Tarjeta de Video MSI AMD Mech Radeon RX 5500 XT MECH Gaming OC, 8GB 128-bit GDDR6, PCI Express 4.0'	15
66	'TCL Smart TV LED 55S425 54.6, 4K Ultra HD, Widescreen, Negro'	15
18	'Tarjeta de Video Gigabyte NVIDIA GeForce GT 1030, 2GB 64-bit GDDR5, PCI Express x16 3.0'	11
51	'SSD Kingston UV500, 480GB, SATA III, mSATA'	11
1	'Procesador AMD Ryzen 3 3300X S-AM4, 3.80GHz, Quad-Core, 16MB L2 Cache'	10
6	'Procesador Intel Core i9-9900K, S-1151, 3.60GHz, 8-Core, 16MB Smart Cache (9na. Generación Coffee Lake)'	10
25	'Tarjeta de Video Sapphire AMD Pulse Radeon RX 5500 XT Gaming, 8GB 128-bit GDDR6, PCI Express 4.0'	10
31	, 'Tarjeta Madre AORUS micro ATX B450 AORUS M (rev. 1.0), S-AM4, AMD B450, HDMI, 64GB DDR4 para AMD'	10

40	'Tarjeta Madre Gigabyte XL-ATX TRX40 Designare, S-sTRX4, AMD TRX40, 256GB DDR4 para AMD',	10
----	---	----

La lista está conformada por ['id_producto','nombre_del producto','# ventas, 'categoría']

Los productos con menores ventas de la categoría procesadores son:

'id_producto'	'nombre_del producto'	'# ventas'
1	'Procesador AMD Ryzen 3 3300X S-AM4, 3.80GHz, Quad-Core, 16MB L2 Cache'	2
6	'Procesador Intel Core i9-9900K, S-1151, 3.60GHz, 8-Core, 16MB Smart Cache (9na. Generación Coffee Lake)'	3
8	'Procesador Intel Core i5-9600K, S-1151, 3.70GHz, Six-Core, 9MB Smart Cache (9na. Generación - Coffee Lake)'	4
7	'Procesador Intel Core i7-9700K, S-1151, 3.60GHz, 8-Core, 12MB Smart Cache (9na. Generación Coffee Lake)'	7
2	'Procesador AMD Ryzen 5 3600, S-AM4, 3.60GHz, 32MB L3 Cache, con Disipador Wraith Stealth'	11
4	'Procesador AMD Ryzen 3 3200G con Gráficos Radeon Vega 8, S-AM4, 3.60GHz, Quad-Core, 4MB L3, con Disipador Wraith Spire'	13
5	'Procesador Intel Core i3-9100F, S-1151, 3.60GHz, Quad-Core, 6MB Cache (9na. Generación - Coffee Lake)'	20
3	'Procesador AMD Ryzen 5 2600, S-AM4, 3.40GHz, Six-Core, 16MB L3 Cache, con Disipador Wraith Stealth'	42

La lista está conformada por ['id_producto','nombre_del producto','# ventas, 'categoría']

Los productos con menores ventas de la categoría tarjetas de video son:

'id_producto'	'nombre_del producto'	'# ventas'
17	'Tarjeta de Video Gigabyte AMD Radeon R7 370 OC, 2GB 256-bit GDDR5, PCI Express 3.0'	-1
28	'Tarjeta de Video Zotac NVIDIA GeForce GTX 1660 Ti, 6GB 192-bit GDDR6, PCI Express x16 3.0'	1
22	'Tarjeta de Video MSI NVIDIA GeForce GTX 1050 Ti OC, 4GB 128-bit GDDR5, PCI Express x16 3.0'	1
13	'Tarjeta de Video Asus NVIDIA GeForce GTX 1050 Ti Phoenix, 4GB 128-bit GDDR5, PCI Express 3.0'	1
10	'MSI GeForce 210, 1GB GDDR3, DVI, VGA, HDCP, PCI Express 2.0'	1
25	'Tarjeta de Video Sapphire AMD Pulse Radeon RX 5500 XT Gaming, 8GB 128-bit GDDR6, PCI Express 4.0'	2
21'	'Tarjeta de Video MSI AMD Mech Radeon RX 5500 XT MECH Gaming OC, 8GB 128-bit GDDR6, PCI Express 4.0'	2

11	'Tarjeta de Video ASUS AMD Radeon RX 570, 4GB 256-bit GDDR5, PCI Express 3.0'	3
18	'Tarjeta de Video Gigabyte NVIDIA GeForce GT 1030, 2GB 64-bit GDDR5, PCI Express x16 3.0'	5
12	'Tarjeta de Video ASUS NVIDIA GeForce GTX 1660 SUPER EVO OC, 6GB 192-bit GDDR6, PCI Express x16 3.0'	9

La lista está conformada por ['id_producto','nombre_del producto','# ventas, 'categoría']

Los productos con menores ventas de la categoría tarjetas madre son:

'id_producto'	'nombre_del producto'	'# ventas'
46	'Tarjeta Madre Gigabyte micro ATX GA-H110M-DS2, S-1151, Intel H110, 32GB DDR4 para Intel'	-1
45	'Tarjeta Madre ASRock ATX H110 Pro BTC+, S-1151, Intel H110, 32GB DDR4, para Intel'	-1
40	'Tarjeta Madre Gigabyte XL-ATX TRX40 Designare, S-sTRX4, AMD TRX40, 256GB DDR4 para AMD'	1
33	'Tarjeta Madre ASUS ATX PRIME Z390-A, S-1151, Intel Z390, HDMI, 64GB DDR4 para Intel\xa0'	2
44	, 'Tarjeta Madre MSI ATX B450 TOMAHAWK MAX, S-AM4, AMD B450, 64GB DDR4 para AMD'	6
29	'Tarjeta Madre ASUS micro ATX TUF B450M-PLUS GAMING, S-AM4, AMD B450, HDMI, 64GB DDR4 para AMD'	12
42	'Tarjeta Madre ASRock Micro ATX B450M Steel Legend, S-AM4, AMD B450, HDMI, 64GB DDR4 para AMD'	18

La lista está conformada por ['id_producto','nombre_del producto','# ventas, 'categoría']

Los productos con menores ventas de la categoría discos duros son:

'id_producto'	'nombre_del producto'	'# ventas'
50	'SSD Crucial MX500, 1TB, SATA III, M.2'	1
52	'SSD Western Digital WD Blue 3D NAND, 2TB, M.2'	2
51'	'SSD Kingston UV500, 480GB, SATA III, mSATA'	3
49	'Kit SSD Kingston KC600, 1TB, SATA III, 2.5, 7mm'	
48	'SSD Kingston A2000 NVMe, 1TB, PCI Express 3.0, M2'	9
47	'SSD XPG SX8200 Pro, 256GB, PCI Express, M.2'	
57	"SSD Adata Ultimate SU800, 256GB, SATA III, 2.5"	15
54	"SSD Kingston A400, 120GB, SATA III, 2.5", 7mm"	48

La lista está conformada por ['id_producto','nombre_del producto','# ventas, 'categoría']

Los productos con menores ventas de la categoría memorias usb son:

'id_producto'	'nombre_del producto'	'# ventas'
60	'Kit Memoria RAM Corsair Dominator Platinum DDR4, 3200MHz, 16GB (2x 8GB), Non-ECC, CL16, XMP'	1

La lista está conformada por ['id_producto','nombre_del producto','# ventas, 'categoría']

Los productos con menores ventas de la categoría pantallas son:

'id_producto'	'nombre_del producto'	'# ventas'
67	'TV Monitor LED 24TL520S-PU 24, HD, Widescreen, HDMI, Negro'	1
66	'TCL Smart TV LED 55S425 54.6, 4K Ultra HD, Widescreen, Negro'	1

La lista está conformada por ['id_producto','nombre_del producto','# ventas, 'categoría']

Los productos con menores ventas de la categoría bocinas son:

'id_producto'	'nombre_del producto'	'# ventas'
74	'Logitech Bocinas para Computadora con Subwoofer G560, Bluetooth, Inalámbrico, 2.1, 120W RMS, USB, negro	2

La lista está conformada por ['id_producto','nombre_del producto','# ventas, 'categoría']

Los productos con menores ventas de la categoría audífonos son:

'id_producto'	'nombre_del producto'	'# ventas'
94	'HyperX Audífonos Gamer Cloud Flight para PC/PS4/PS4 Pro, Inalámbrico, USB, 3.5mm, Negro'	1
89	'Cougar Audífonos Gamer Phontum Essential, Alámbrico, 1.9 Metros, 3.5mm, Negro.'	1
84	'Logitech Audífonos Gamer G332, Alámbrico, 2 Metros, 3.5mm, Negro/Rojo'	1
85	, 'Logitech Audífonos Gamer G635 7.1, Alámbrico, 1.5 Metros, 3.5mm, Negro/Azul'	2

La lista está conformada por ['id_producto','nombre_del producto','# búsqueda, 'categoría']

Los productos con menores búsqueda de la categoría procesadores son:

'id_producto'	'nombre_del producto'	'# búsqueda'
9	'Procesador Intel Core i3-8100, S-1151, 3.60GHz, Quad-Core, 6MB Smart Cache (8va. Generación - Coffee Lake)'	1
6	, 'Procesador Intel Core i9-9900K, S-1151, 3.60GHz, 8-Core, 16MB Smart Cache (9na. Generación Coffee Lake)'	10
1	'Procesador AMD Ryzen 3 3300X S-AM4, 3.80GHz, Quad-Core, 16MB L2 Cache'	10
8	'Procesador Intel Core i5-9600K, S-1151, 3.70GHz, Six-Core, 9MB Smart Cache (9na. Generación - Coffee Lake)	20
2	'Procesador AMD Ryzen 5 3600, S-AM4, 3.60GHz, 32MB L3 Cache, con Disipador Wraith Stealth'	24

5	'Procesador Intel Core i3-9100F, S-1151, 3.60GHz, Quad-Core, 6MB Cache (9na. Generación - Coffee Lake)'	30
7	'Procesador Intel Core i7-9700K, S-1151, 3.60GHz, 8-Core, 12MB Smart Cache (9na. Generación Coffee Lake)'	31
4	'Procesador AMD Ryzen 3 3200G con Gráficos Radeon Vega 8, S-AM4, 3.60GHz, Quad-Core, 4MB L3, con Disipador Wraith Spire'	41
3	'Procesador AMD Ryzen 5 2600, S-AM4, 3.40GHz, Six-Core, 16MB L3 Cache, con Disipador Wraith Stealth'	55

La lista está conformada por ['id_producto','nombre_del producto','# búsqueda, 'categoría']
 Los productos con menores búsqueda de la categoría tarjetas de video son:

'id_producto'	'nombre_del producto'	'# busquedas'
27	'Tarjeta de Video VisionTek AMD Radeon HD5450, 2GB GDDR3, PCI Express x16'	1
10	'MSI GeForce 210, 1GB GDDR3, DVI, VGA, HDCP, PCI Express 2.0'	1
13	'Tarjeta de Video Asus NVIDIA GeForce GTX 1050 Ti Phoenix, 4GB 128-bit GDDR5, PCI Express 3.0'	2
17	'Tarjeta de Video Gigabyte AMD Radeon R7 370 OC, 2GB 256-bit GDDR5, PCI Express 3.0'	3
15'	, 'Tarjeta de Video EVGA NVIDIA GeForce GTX 1660 Ti SC Ultra Gaming, 6GB 192-bit GDDR6, PCI 3.0'	4
28	'Tarjeta de Video Zotac NVIDIA GeForce GTX 1660 Ti, 6GB 192-bit GDDR6, PCI Express x16 3.0'	5
26	'Tarjeta de Video VisionTek AMD Radeon HD 5450, 1GB DDR3, PCI Express x16 2.1'	5
22	'Tarjeta de Video MSI NVIDIA GeForce GTX 1050 Ti OC, 4GB 128-bit GDDR5, PCI Express x16 3.0'	5
11	'Tarjeta de Video ASUS AMD Radeon RX 570, 4GB 256-bit GDDR5, PCI Express 3.0	5
25	'Tarjeta de Video Sapphire AMD Pulse Radeon RX 5500 XT Gaming, 8GB 128-bit GDDR6, PCI Express 4.0'	10
18	'Tarjeta de Video Gigabyte NVIDIA GeForce GT 1030, 2GB 64-bit GDDR5, PCI Express x16 3.0'	11
21	'Tarjeta de Video MSI AMD Mech Radeon RX 5500 XT MECH Gaming OC, 8GB 128-bit GDDR6, PCI Express 4.0'	15
12	'Tarjeta de Video ASUS NVIDIA GeForce GTX 1660 SUPER EVO OC, 6GB 192-bit GDDR6, PCI Express x16 3.0	15

La lista está conformada por ['id_producto','nombre_del producto','# búsquedas, 'categoría']
 Los productos con menores búsquedas de la categoría tarjetas madre son:

'id_producto'	'nombre_del producto'	'# búsqueda'
45	'Tarjeta Madre ASRock ATX H110 Pro BTC+, S-1151, Intel H110, 32GB DDR4, para Intel'	1

35	'Tarjeta Madre Gigabyte micro ATX Z390 M GAMING, S-1151, Intel Z390, HDMI, 64GB DDR4 para Intel\	1
39	'ASUS T. Madre uATX M4A88T-M, S-AM3, DDR3 para Phenom II/Athlon II/Sempron 100'	3
46	'Tarjeta Madre Gigabyte micro ATX GA-H110M-DS2, S-1151, Intel H110, 32GB DDR4 para Intel'	4
40	'Tarjeta Madre Gigabyte XL-ATX TRX40 Designare, S-sTRX4, AMD TRX40, 256GB DDR4 para AMD'	10
31	'Tarjeta Madre AORUS micro ATX B450 AORUS M (rev. 1.0), S-AM4, AMD B450, HDMI, 64GB DDR4 para AMD'	10
42	'Tarjeta Madre ASRock Micro ATX B450M Steel Legend, S-AM4, AMD B450, HDMI, 64GB DDR4 para AMD'	23
44	'Tarjeta Madre MSI ATX B450 TOMAHAWK MAX, S-AM4, AMD B450, 64GB DDR4 para AMD'	25
29	'Tarjeta Madre ASUS micro ATX TUF B450M-PLUS GAMING, S-AM4, AMD B450, HDMI, 64GB DDR4 para AMD'	60

La lista está conformada por ['id_producto','nombre_del producto','# búsqueda, 'categoría']
 Los productos con menores búsqueda de la categoría discos duros son:

'id_producto'	'nombre_del producto'	'# búsqueda'
59	'SSD Samsung 860 EVO, 1TB, SATA III, M.2'	2
56	"SSD para Servidor Lenovo Thinksystem S4500, 480GB, SATA III, 3.5", 7mm"	2
52	'SSD Western Digital WD Blue 3D NAND, 2TB, M.2'	5
50	'SSD Crucial MX500, 1TB, SATA III, M.2'	7
49	'Kit SSD Kingston KC600, 1TB, SATA III, 2.5, 7mm'	10
51	'SSD Kingston UV500, 480GB, SATA III, mSATA'	11
48	'SSD Kingston A2000 NVMe, 1TB, PCI Express 3.0, M2'	27
47	'SSD XPG SX8200 Pro, 256GB, PCI Express, M.2'	30
57	"SSD Adata Ultimate SU800, 256GB, SATA III, 2.5", 7mm"	107
54	"SSD Kingston A400, 120GB, SATA III, 2.5", 7mm"	263

La lista está conformada por ['id_producto','nombre_del producto','# búsqueda, 'categoría']
 Los productos con menores búsqueda de la categoría memorias USB son:

Las memorias usb no tienen busquedas.

La lista está conformada por ['id_producto','nombre_del producto','# búsqueda, 'categoría']
 Los productos con menores búsqueda de la categoría pantallas son:

'id_producto'	'nombre_del producto'	'# búsquedas'
70	'Samsung Smart TV LED 43, Full HD, Widescreen, Negro'	1
73	'Samsung Smart TV LED UN55TU7000FXZX 55, 4K Ultra HD, Widescreen, Negro/Gris'	4
63	'Seiki TV LED SC-39HS950N 38.5, HD, Widescreen, Negro'	4

66	'TCL Smart TV LED 55S425 54.6, 4K Ultra HD, Widescreen, Negro'	15
67	'TV Monitor LED 24TL520S-PU 24, HD, Widescreen, HDMI, Negro'	32

La lista está conformada por ['id_producto', 'nombre_del producto', '# búsqueda', 'categoría']
 Los productos con menores búsqueda de la categoría bocinas son:

'id_producto'	'nombre_del producto'	'# búsqueda'
80	'Ghia Bocina Portátil BX800, Bluetooth, Inalámbrico, 2.1 Canales, 31W, USB, Negro'	1
76	'Acteck Bocina con Subwoofer AXF-290, Bluetooth, Inalámbrico, 2.1, 18W RMS, 180W PMPO, USB, Negro'	2
74	'Logitech Bocinas para Computadora con Subwoofer G560, Bluetooth, Inalámbrico, 2.1, 120W RMS, USB, negro'	6

La lista está conformada por ['id_producto', 'nombre_del producto', '# búsqueda', 'categoría']
 Los productos con menores búsqueda de la categoría audífonos son:

'id_producto'	'nombre_del producto'	'# búsquedas'
93	'Ginga Audífonos con Micrófono GI18ADJ01BT-RO, Bluetooth, Alámbrico/Inalámbrico, 3.5mm, Rojo'	1
91	'Genius GHP-400S Audífonos, Alámbrico, 1.5 Metros, Rosa', 2, 'audífonos']	2
95	'logear Audífonos Gamer GHG601, Alámbrico, 1.2 Metros, 3.5mm, Negro', 3, 'audifonos'	3
94	'HyperX Audífonos Gamer Cloud Flight para PC/PS4/PS4 Pro, Inalámbrico, USB, 3.5mm, Negro', 6, 'audifonos'	6
89	'Cougar Audífonos Gamer Phontum Essential, Alámbrico, 1.9 Metros, 3.5mm, Negro.',	7
84	'Logitech Audífonos Gamer G332, Alámbrico, 2 Metros, 3.5mm, Negro/Rojo'	10
85	'Logitech Audífonos Gamer G635 7.1, Alámbrico, 1.5 Metros, 3.5mm, Negro/Azul'	35

Solución parte 2

Los 20 productos con mejor reseña son: ["id_producto", "nombre_producto", "reseñas"]

id_producto	Nombre_producto	reseña
54	"SSD Kingston A400, 120GB, SATA III, 2.5", 7mm"	236
3	'Procesador AMD Ryzen 5 2600, S-AM4, 3.40GHz, Six-Core, 16MB L3 Cache, con Disipador Wraith Stealth'	202
5	'Procesador Intel Core i3-9100F, S-1151, 3.60GHz, Quad-Core, 6MB Cache (9na. Generación - Coffee Lake)'	94
42	'Tarjeta Madre ASRock Micro ATX B450M Steel Legend, S-AM4, AMD B450, HDMI, 64GB DDR4 para AMD'	82
57	"SSD Adata Ultimate SU800, 256GB, SATA III, 2.5", 7mm"	73

4	'Procesador AMD Ryzen 3 3200G con Gráficos Radeon Vega 8, S-AM4, 3.60GHz, Quad-Core, 4MB L3, con Disipador Wraith Spire'	58
29	'Tarjeta Madre ASUS micro ATX TUF B450M-PLUS GAMING, S-AM4, AMD B450, HDMI, 64GB DDR4 para AMD'	58
2	'Procesador AMD Ryzen 5 3600, S-AM4, 3.60GHz, 32MB L3 Cache, con Disipador Wraith Stealth'	55
47	'SSD XPG SX8200 Pro, 256GB, PCI Express, M.2'	50
12	'Tarjeta de Video ASUS NVIDIA GeForce GTX 1660 SUPER EVO OC, 6GB 192-bit GDDR6, PCI Express x16 3.0'	43
48	'SSD Kingston A2000 NVMe, 1TB, PCI Express 3.0, M2'	48
7	'Procesador Intel Core i7-9700K, S-1151, 3.60GHz, 8-Core, 12MB Smart Cache (9na. Generación Coffee Lake)'	35
44	'Tarjeta Madre MSI ATX B450 TOMAHAWK MAX, S-AM4, AMD B450, 64GB DDR4 para AMD'	28
18	'Tarjeta de Video Gigabyte NVIDIA GeForce GT 1030, 2GB 64-bit GDDR5, PCI Express x16 3.0'	22
8	'Procesador Intel Core i5-9600K, S-1151, 3.70GHz, Six-Core, 9MB Smart Cache (9na. Generación - Coffee Lake)'	20
6	'Procesador Intel Core i9-9900K, S-1151, 3.60GHz, 8-Core, 16MB Smart Cache (9na. Generación Coffee Lake)'	15
11	'Tarjeta de Video ASUS AMD Radeon RX 570, 4GB 256-bit GDDR5, PCI Express 3.0'	15
49	'Kit SSD Kingston KC600, 1TB, SATA III, 2.5, 7mm'	15
51	'SSD Kingston UV500, 480GB, SATA III, mSATA'	14
31	'Tarjeta Madre AORUS micro ATX B450 AORUS M (rev. 1.0), S-AM4, AMD B450, HDMI, 64GB DDR4 para AMD'	11

Los 20 productos con peor reseña son: ["id_producto", "nombre_producto", "reseñas"]

id_producto	Nombre_producto	reseña
45	'Tarjeta Madre ASRock ATX H110 Pro BTC+, S-1151, Intel H110, 32GB DDR4, para Intel'	1
17	'Tarjeta de Video Gigabyte AMD Radeon R7 370 OC, 2GB 256-bit GDDR5, PCI Express 3.0'	1
46	'Tarjeta Madre Gigabyte micro ATX GA-H110M-DS2, S-1151, Intel H110, 32GB DDR4 para Intel'	2
89	'Cougar Audífonos Gamer Phontum Essential, Alámbrico, 1.9 Metros, 3.5mm, Negro.'	3
94	'HyperX Audífonos Gamer Cloud Flight para PC/PS4/PS4 Pro, Inalámbrico, USB, 3.5mm, Negro'	4
13	'Tarjeta de Video Asus NVIDIA GeForce GTX 1050 Ti Phoenix, 4GB 128-bit GDDR5, PCI Express 3.0'	4
10	'MSI GeForce 210, 1GB GDDR3, DVI, VGA, HDCP, PCI Express 2.0'	4
84	'Logitech Audífonos Gamer G332, Alámbrico, 2 Metros, 3.5mm, Negro/Rojo'	5
67	'TV Monitor LED 24TL520S-PU 24, HD, Widescreen, HDMI, Negro'	5
66	'TCL Smart TV LED 55S425 54.6, 4K Ultra HD, Widescreen, Negro'	5
60	'Kit Memoria RAM Corsair Dominator Platinum DDR4, 3200MHz, 16GB (2x 8GB), Non-ECC, CL16, XMP'	5
50	'SSD Crucial MX500, 1TB, SATA III, M.2'	5
40	'Tarjeta Madre Gigabyte XL-ATX TRX40 Designare, S-sTRX4, AMD TRX40, 256GB DDR4 para AMD'	5
28	'Tarjeta de Video Zotac NVIDIA GeForce GTX 1660 Ti, 6GB 192-bit GDDR6, PCI Express x16 3.0'	5
22	'Tarjeta de Video MSI NVIDIA GeForce GTX 1050 Ti OC, 4GB 128-bit GDDR5, PCI Express x16 3.0'	5
74	'Logitech Bocinas para Computadora con Subwoofer G560, Bluetooth, Inalámbrico, 2.1, 120W RMS, USB, negro'	9
33	'Tarjeta Madre ASUS ATX PRIME Z390-A, S-1151, Intel Z390, HDMI, 64GB DDR4 para Intel'	9
85	'Logitech Audífonos Gamer G635 7.1, Alámbrico, 1.5 Metros, 3.5mm, Negro/Azul'	10
52	'SSD Western Digital WD Blue 3D NAND, 2TB, M.2'	
25	'Tarjeta de Video Sapphire AMD Pulse Radeon RX 5500 XT Gaming, 8GB 128-bit GDDR6, PCI Express 4.0'	10

Solución parte 3

Los ingresos mensuales son:

El total de ingresos para el mes 01 de 2020 es: \$ 115,239 de 52 ventas y 1 devoluciones

El total de ingresos para el mes 02 de 2020 es: \$ 104,401 de 40 ventas y 1 devoluciones

El total de ingresos para el mes 03 de 2020 es: \$ 161,133 de 49 ventas y 2 devoluciones

El total de ingresos para el mes 04 de 2020 es: \$ 188,837 de 74 ventas y 1 devoluciones

El total de ingresos para el mes 05 de 2020 es: \$ 87,219 de 33 ventas y 2 devoluciones

El total de ingresos para el mes 06 de 2020 es: \$ 36,949 de 11 ventas y 0 devoluciones

El total de ingresos para el mes 07 de 2020 es: \$ 26,949 de 11 ventas y 0 devoluciones

El total de ingresos para el mes 08 de 2020 es: \$ 3,077 de 3 ventas y 0 devoluciones

El total de ingresos para el mes 09 de 2020 es: \$ -4,199 de 0 ventas y 1 devolución

El total de ingresos para el año 2020 es: \$ 719605

Los 3 meses con mayores ventas fueron [Mes, ventas]:

['Abril', 73], ['Enero', 51], ['Marzo', 47]]

Conclusiones:

Se considera que se debe tratar de vaciar inventarios de memorias usb, debido a que ya no son un producto de interés para el mercado, no se busco ninguna y solo se vendió 1, de igual forma los audífonos solo se vendieron 5, adicional a esto los productos que menos ventas tuvieron y considero que se debe hacer para vaciar el inventario son:

'Tarjeta Madre Gigabyte micro ATX GA-H110M-DS2, S-1151, Intel H110, 32GB DDR4 para Intel'
'Tarjeta Madre ASRock ATX H110 Pro BTC+, S-1151, Intel H110, 32GB DDR4, para Intel'
'Tarjeta de Video Gigabyte AMD Radeon R7 370 OC, 2GB 256-bit GDDR5, PCI Express 3.0'

Ya que tuvieron devoluciones cada una.

Se debe considerar una campaña publicitaria donde se dé a conocer la tienda, adicional a ello se oferten los productos devueltos, exista promociones para televisiones, bocinas, audífonos, debido a que son de las categorías, con menor número de búsquedas y ventas.

El reporte y archivos se encuentran en el repositorio de github en el siguiente enlace:

<https://github.com/losiram13/https-github.com-losiram13-REPORTE-01-VAZQUEZ-HERRERA-MARISOL->