

Reporte mensual de ventas para la tienda virtual LifeStore.

Lizbeth Cirilo Peña

16 de septiembre de 2021

Índice

1. Introducción	2
2. Código	2
2.1. Login	2
2.2. Menú de opciones.	3
2.3. Selección de la primera parte del menú (el proceso del analisis de las ventas)	3
2.4. Selección de la segunda parte del menú (Proceso de selección de los productos por las mejores y peores reseñas)	5
2.5. Selección de la tercera parte del menú (Proceso de selección de los productos para clasificar los ingresos y el numero de ventas por mes, anual y en promedio las ganancias.)	6
3. Conclusión	7

1. Introducción

En el siguiente reporte mensual de ventas, se contempla diversos aspectos a evaluar. Primeramente los productos se realizará un listado con los 15 productos más vendidos y una lista de los 20 productos con las mayores búsquedas, así como una clasificación de las categorías de los productos, en base a esta clasificación se hará un listado con los 5 productos con las menores ventas y 20 con las menores búsquedas productos.

Para evaluar el servicio que hay para cada producto, se clasificará en categorías a los productos, y se realizarán los enlistados que nos muestren cada uno 10 artículos con las mejores y las peores reseñas.

Y por último se aterrizará una estrategia para retirar productos del mercado según el criterio a partir de la acumulación de inventario (ingresos y ventas mensuales), esto mediante el análisis de un listado que nos muestre tanto los datos de los ingresos mensuales y ventas mensuales promedio, totales anualmente y meses con más ventas al año.

2. Código

A continuación se presenta el código con el cual se realizó dicho informe.

NOTA: Los usuarios y sus contraseñas son las siguientes:

- Usuario: admin / Contraseña: admin
- Usuario: liz / Contraseña: 1234
- Usuario: jimmy / Contraseña: 0000

2.1. Login

```
1 #AQUI INICIA EL LOGIN
2
3 reg_user = ["admin","liz","jimmy"] #aqui se encuentran los usuarios y sus contraseñas por
4 las que pueden acceder al sistema
5 reg_pass = ["admin","1234","0000"] #sus respectivas contraseñas
6
7 intentos=len(reg_user)
8
9 usuario = input("Ingrese un usuario: ")
10 password = input("ingrese una contraseña: ")
11
12 for i in range(intentos):
13     if usuario == reg_user[i] and password == reg_pass[i] : #aqui se condiciona que el
14     usuario y la contraseña pertenezca a la misma
15         print('\n\n\t=== Bienvenido al sistema -LifeStore ===\n') #si la condicion se
16         cumple, dara acceso al sistema
17         break
18     elif i==intentos-1: #si alguno de las sentencias ya sea el usuario o la contraseña sean
19     diferente se procedera a terminar el proceso
20         print("Error en el usuario y/o contraseña")
21         exit(1) #este exit(1) es para que salga del programa cuando una condicion no se
22         cumpla
```

2.2. Menú de opciones.

```

1 # M E N U       D E       O P C C I O N E S
2
3 while True:
4     #lis_opc= ''
5     # indice = 1
6     # indicacion = '\t'
7     print(f"Menu de opciones\n 1.-Productos mas vendidos\n 2.-Productos por reseña\n 3.-
8     Sugerir una estrategia \n 4.-Salir\n")
9     print("
10
11     ")
12     seleccion = input('Ingrese opcion -> ')
13     if seleccion == '4' :
14         exit(1)
15     seleccion = int(seleccion)

```

2.3. Selección de la primera parte del menú (el proceso del analisis de las ventas)

```

1 #AQUI SE ENCUENTRAN LOS PRECESOS DE LA PRIMERA SELECCION DENTRO DEL MENU
2     if seleccion == 1:
3         print("\n\t\tListado de los productos con mayores ventas \n")
4
5 #ORDEAMIENTO DE LOS PRODCUTOS _MAS VENDIDOS(15)_
6
7     '''LOS 15 PRODCUTOS M S VENDIDOS'''
8     ven = [] #una lista vacia nos ayudara a almacenar los datos para poder ordenarlos
9     posteriormente
10
11     for prodc in lifestore_products:
12         lis = [prodc[0], 0]
13         ven.append(lis)
14
15     # for ventas in ven: ESTE FOR NOS MUESTRA MEDIANTE EL print
16     () QUE LA LISTA YA NOS TRAJO LOS INDICES DE
17     #lifestore_products de la forma ven[id_product, 0]
18     # print(ventas)
19
20     for venta in lifestore_sales:
21         prod_ven = venta[1]
22         devoluciones = venta[4]
23         # print(prod_ven)
24         # print(devoluciones)
25         #estos print's nos muestran la cantidad de veces que han sido vendidos, y la
26         cantidad que han sido devueltos,
27         #solo como repeticiones,
28         # como aparecen en la lista
29         #Devoluciones solo toma el valor 0 y 1, donde 0 = NO devuelto, 1 = Devuelto. Por
30         tanto tendremos que usar condiciones logicas
31         ***IMPORTANTE LA IDENTACION**
32         if devoluciones == 0:
33             valido = True
34         else:
35             valido = False
36
37         #la condicion if, no ayudar para recorrer la lista y validar que cuando sea True= la
38         validacion de la venta, incrementando
39         #el conteo y descartando los que
40         # no cumplan la condicion
41         if valido == True:
42             ven[prod_ven - 1][1] += 1
43         #print(ven)

```

```

38     #Vamos a ordenar, por el segundo elemento de la lista ven[id_product, 0], pues el
    primero es indice el cual son las ventas y es lo que nos interesa ordenar.
39
40
41     def elem_2(l_list):
42         return l_list[1]
43
44     #Aqui vamos ya, trayendo el dato que nos interesea ordenar, mediante elem_2, lo
    ordenaremos con la herramienta .sort()
45     ven.sort(reverse = True, key = elem_2) # esta sintaxis usara, como llave para
    localizar el elemnto a "elem_2", y como condicional reverse == True
46
47     #AHORA QUE YA SE TIENE LOS ELEMENTOS QUE CONFORMAN LA LISTA YA ORDENADA, SE VA A
    DECLARAR UNA VARIABLE PARA SOLO OBTENER LOS 15 M S
    VENDIDOS
48     n=15
49     mas_vendidos = ven[:n]
50
51     print('\n\n\t                                     ESTOS SON LOS 15 PRODUCTOS
    QUE TIENEN LAS MAYORES VENTAS                                \n\n\t')
52
53     for mv in mas_vendidos:
54         print(mv)
55
56
57 #PRODUCTOS CON MAYORES BUSQUEDAS
58
59     print('\n\n\t                                     ESTOS SON LOS PRODUCTOS
    AGRUPADOS POR SUS BUSQUEDAS                                \n\n\t')
60
61     '''aquí me atore para poder ordenarlo con las menores visitas'''
62     m_vis = []
63     for visit in lifestore_searches:
64         if visit[1] not in m_vis: # en esta linea se esta indicando que todo lo qu no
    este en mi lista de busquedas,
65             m_vis.append(visit[1]) #se almacenara "bus", que seran las busquedas que se
    repiten
66     #print(m_vis) # Aquí me trae los datos que esta repetidos dentro de la lista de las
    1033 busquedas dentro de la pagina.
67
68     # for rezago in m_vis:
69     #     print(rezago)
70
71     rezagos = []
72     for busquedas in m_vis: #recordar que m_vis contiene los elemtos que se repiten en
    la lista de vistas que son los id's de los productos
73         m_busq = [busquedas, []] #este será el cuerpo de la lista donde se busca agrupar
    a los artuculos donde
74         rezagos.append(m_busq) #en este apartado se anexa al cuerpo los elemento que se
    filtraron anteriormente
75         #print(rezagos)
76
77     for enlistado in rezagos:
78         list_re = enlistado[0] #apartir de aqui, se extraen los id de la list de los
    rezagos, que son los productos buscados [id_product]
79
80     for search in lifestore_searches: #en este for mandamos a llamar ambos elementos
    de la lista lifestore_searches por separado
81         id_bus = search[0]
82         busquedas = search[1]
83
84         if busquedas == list_re: #se condiciona que sí busquedas(id_products) se

```

```

85     repiten, se almacenaran en el lugar 1 de la lista, para agruparlos
86         enlistado[1].append(id_bus)
87     for b in rezagos:
88         print(b) #aqui se imprimen los productos agrupados por la cantidad de busquedas
89         #que se realizaron para cada uno
90
91 #AGRUPACION DE LOS PRODUCTOS POR CATEGORIAS
92
93     print('\n\n\t                                     AQUÍ SE MUESTRA LA AGRUPACION DE
94         LOS PRODUCTOS POR LAS BUSQUEDAS                                     \n\n\n')
95 #PRODUCTOS CON LAS MAYORES VENTAS AGRUPADOS POR LAS BUSQUEDAS
96
97     #Se me dificulta mucho moverme atraves de las listas, pues el traer determinado
98     #valor no puedo encontrarlo o mandarlo a llamar
99     # a = []
100    # for vendidos in mas_vendidos:
101    #     ve = mas_vendidos
102    #     mas_vend_bus = [ve, []]
103    #     a.append(mas_vend_bus)
104    #     print(a)
105    # for ven in b:
106    #     v = ven[0]
107    #     for buscar in b:
108    #         r = buscar[1]
109    #         a.append()
110    #         # print(a)
111
112    print("\n\n\n\n\nSELECCIONE LA SIGUENTE SECCION\n")

```

2.4. Selección de la segunda parte del menú (Proceso de selección de los productos por las mejores y peores reseñas)

```

1 #PRODUCTOS AGRUPADOS POR SU RESEÑA EN EL SERVICIO
2 if seleccion == 2:
3     print("\n\tProductos por reseña en el servicio\n")
4
5
6     print('\n\n\t                                     AQUÍ SE MUESTRA LA AGRUPACION DE
7         LOS PRODCUTOS POR CALIFICACION EN EL SERVICIO                                     \n\n\n')
8     calf = []
9     for calificacion in lifestore_sales:
10         ca = calificacion[2]
11         #print(ca)
12
13         if ca not in calf:
14             calf.append(ca)
15             #print(calf)
16
17     score = []
18
19     for cali in calf:
20         calf_prod = [cali, []]
21         score.append(calf_prod)
22         #print(score)
23
24     for listas in score:
25         calif_list = listas[1]

```

```

25     for ca_pr in lifestore_sales:
26         prodt = ca_pr[1]
27         calis = ca_pr[2]
28         if score == calif_list:
29             listas[1].append(calis)
30     for califica_prd in score:
31         print(califica_prd)
32     #El proceso no me lamacena el siguiente dato que son los productos que contienen la
    calificación
33
34     print("\n\n\n\n\nSELECCIONE LA SIGUENTE SECCION\n")

```

2.5. Selección de la tercera parte del menú (Proceso de selección de los productos para clasificar los ingresos y el numero de ventas por mes, anual y en promedio las ganancias.)

```

1  #PRODUCTOS QUE MUESTRAN EL TOTAL DE LOS INGRESOS Y VENTAS MENSUALES Y ANUALES , ASI COMO LOS
    MESES CON MAS VENTAS AL A O
2      if seleccion == 3:
3          print("\nVentas promedio mensuales y anuales \n")
4
5  #LOS PRODCUTOS AGRUPADOS CON LAS FECHAS DE VENTA
6      print('\n\t                                     AQUÍ SE MUESTRA LA AGRUPACION DE
    LAS VENTAS DE LOS PORDUCTOS POR FECHA                                     \n\n\n')
7
8      ing_fech = []
9
10     for fecha in lifestore_sales: #QUISE EXTRAER EN DOS LISTAS DIFERENTES LAS FECHAS Y
        SUS ID DE LOS PRODUCTOS
11         id_p1 = fecha[1]
12         fechs = fecha[3]
13
14         fechas = (id_p1, fechs)
15         #print(fechas)#_-----
16
17     for ingreso in lifestore_products: #AQUI VA ESTA OTRA LISTA QUE CONTIENE LOS PRECIOS
        DE CADA PRODUCTO Y SUS ID
18
19         id_pp = ingreso[0]
20         ing = ingreso[2]
21
22         ing_prd = (id_pp, ing) #aqui esta la lista que tiene la forma de (id_product,
        precios)
23         #print(ing_prd)#_-----
24
25     # esto me trae los id de los prodcutos que se han vendido dentro el 2020
26     #de la forma prod_fe [id_product, []]
27
28     ingresos_fechas = []
29
30     for fe_prod in fechas:
31         fe= fechas[0]
32         date_prd = [fe, []]
33         ingresos_fechas.append(date_prd) #Aqui quise extraer los id's y contenerlos en
        esta lista
34
35     for id_prod in ing_prd:
36         d_prod = id_prod[0]
37
38     for idp in fechas:
39         id_p_f = idp[0]
40         fechas = [1]

```

```
41         if id_p_f == d_prod:
42             id_prod[0].append(id_p_f) #Aqui, lo que queria era contener comparando
43             cada id de la lita y agrupar las fechas para despues extraer los precios y clasificarlas
44
45         for producto_fechas in ing_prd:
46             print(producto_fechas) #me imprime las listas vacias a pesar de que hay un .
47             append(), no itera aun asi con ningun elemento
48 #lifestore_sales = [id_sale, id_product, score (from 1 to 5), date, refund (1 for true or 0
49                     to false)]
50 #lifestore_products = [id_product, name, price, category, stock]
51
52 print("\n\n\n\nSELECCIONE LA SIGUENTE SECCION\n")
```

3. Conclusión

En base a este reporte, no podría aportar alguna conclusión en base al analisis de los datos, pero por mi parte didactica, y por mi nula experiencia en dentro del lenguaje Python, puedo atreverme a decir que avance y comprendi cierta logica del lenguaje y más que nada su sintaxis (más que nada su indentación).

Su breve biografía.