

REPORTE FINAL

EMPRESA : LIFESTORE
ELABORA: JUDITH M. ARIAS C.

13 DE FEBRERO DE 2022

ÍNDICE

1. <u>Introducción</u>	3
2. <u>Explicación del código</u>	4
2.1 <u>Login</u>	5
2.2 <u>contenido</u>	5
2.3 <u>Lista productos más vendidos</u>	6
2.4 <u>Lista productos más buscados</u>	6
2.5 <u>Lista productos menos vendidos</u>	7
2.6 <u>Lista productos menos buscados</u>	7
2.7 <u>20 productos con mejores reseñas</u>	8
2.8 <u>20 productos con peores reseñas</u>	8
2.10 <u>Ventas por anuales y mensuales</u>	9
3. <u>Productos mejor vendidos</u>	10
4. <u>Productos más buscados</u>	10
5. <u>Productos mejor calificados</u>	11
6. <u>Productos peor calificados</u>	11
7. <u>Evidencias</u>	12
8. <u>Código completo</u>	16
8. <u>Solución</u>	17

INTRODUCCIÓN

La tecnología continúa produciendo avances sin precedentes a un ritmo veloz, la transformación digital sólo puede brindarnos su máximo potencial si canalizamos el poder de los datos que ésta brinda.

Brindando una capacidad para detectar problemas comerciales complejos con ayuda del

aprendizaje automático, como dificultades de investigación de operaciones, este proyecto tienen la clave para descubrir mejores soluciones para la empresa Lifestore. Desarrollando un conocimiento desde los resultados y antecedentes de las misma.





EXPLICACIÓN DEL CÓDIGO

LOGIN

Se dio inicio con la creación de un **Login** del usuario, por medio del cual tendrá acceso a los datos principales siendo de suma importancia mantener la privacidad de los datos usados. Donde por medio de la función de relaciones se configuró la manera en que se puede acceder al sistema.

```
12
13 # INICIO DEL LOGIN DE USUARIO #
14 users = {'Gerencia': '358554'}
15 status = ""
16 opcion = ""
17
18 while status != "q":
19     print(" -----")
20     print("|   LIFESTORE®   |")
21     print(" -----")
22     status = input("¿Estas Registrado? si/no (Ingresa q para salir)")
23     status = status.lower()
24     if status == "si":
25         login = input("Ingresa tu usuario: ")
26         passw = input("Ingresa tu contraseña: ")
27         #print(users)
28         if login in users and users[login] == passw:
29             print("\n====> Has iniciado sesión como ", login,"@")
30             print("-----")
```

CONTENIDO

La empresa tiene que tener conocimiento de la estructura del contenido; por ello, se creo un diccionario donde puede encontrar el contenido del conjunto de códigos desarrollados, donde predomina la función "print" según el numero del contenido a mostrar.

```
31 while opcion != "0":
32     print()
33     print("-----CONTENIDO-----")
34     print("| 1.- Lista de los 50 Productos con Mayores Ventas")
35     print("| 2.- Lista de los 100 Productos con Mayores Busquedas")
36     print("| 3.- Lista de los 50 Productos con Menores Ventas (Agrupados por categoría)")
37     print("| 4.- Lista de los 100 Productos con Menores Busquedas (Agrupados por categoría)")
38     print("| 5.- Lista de los Productos con Mejor Reseña (Unicamente con 5 de score)")
39     print("| 6.- Lista de los Productos con Mejor Reseña (5 o 4 de Score)")
40     print("| 7.- Lista de los Productos con Peor Reseña (1 o 2 de Score)")
41     print("| 8.- Lista de los Productos con Peor Reseña (Unicamente con 1 de score)")
42     print("| 9.- TOTAL DE INGRESOS, DINERO EN DEVOLUCIONES")
43     print("| 10.- VENTAS POR MES Y PROMEDIO MENSUAL")
44     print("| 11.- VENTAS QUE NO ESTAN DENTRO DEL AÑO 2020")
45
46     print("| 9.- TOTAL DE INGRESOS, DINERO EN DEVOLUCIONES")
47     print("| 10.- VENTAS POR MES Y PROMEDIO MENSUAL")
48     print("| 11.- VENTAS QUE NO ESTAN DENTRO DEL AÑO 2020")
49     print("| 12.- PRODUCTOS DEVUELTOS")
50     print("| 13.- LISTA DE PRODUCTOS CON MENORES VENTAS Y MAYOR STOCK")
51     print("| 14.- VALOR DEL INVENTARIO POR PRODUCTO")
52     print("| 15.- VENTAS POR CATEGORIA")
53     print("| 16.- PRODUCTOS CON MENOR BUSQUEDA Y MAYOR STOCK")
54     print("| 0- Salir")
55
56     print("-----")
57     opcion=input("Selecciona una opcion...")
```

LISTADO DE 5 PRODUCTOS MÁS VENDIDOS

Se itero sobre los productos, por cada producto itero sobre las ventas y mediante un if identifico cuales son las coincidencias. En esta lista irán los productos y sus respectivas ventas, reseteo su valor para que cada producto tenga su sublista con su nombre y sus respectivas ventas. Se utilizo el método bubble sort y le agrego el parámetro reverse para ordenar la lista de mayor a menor, utilizando la segunda posición de la sublista.

```
53 opcion=input("Selecciona una opcion...")
54 if opcion == '1':
55     #INICIO LISTA 5 PRODUCTOS MAS VENDIDOS#
56     productos_venta = []
57     #Itero sobre los productos, por cada producto itero sobre
    las ventas y mediante un if identifico cuales son las
    coincidencias
58     for n in range(len(lifystore_products)):
59         y = [] #En esta lista iran los productos y sus
            respectivas ventas, reseteo su valor para que cada
            producto tenga su sublista con su nombre y sus
            respectivas ventas
60         x=0
61         id_p = lifystore_products[n][0]
62         name = lifystore_products[n][1]
63         stock = lifystore_products[n][4]
64         for j in range(len(lifystore_sales)):
65             if (lifystore_products[n][0] == lifystore_sales[j]
                [1]) and (lifystore_sales[j][4] == 0): #Al añadir
                esta condicion no tomara en cuenta a los productos
                con devolucion
66                 x+=1 #Al encontrar una coincidencia en el
                    Id_Producto de ambas listas le suma uno a la
                    variable x, al final tendra el valor de las
                    veces que se vendio dicho producto
67             y.append(id_p)
```

LISTADO CON LOS 10 PRODUCTOS MÁS BUSCADOS

Mismo Procedimiento que las listas anteriores, solo que aquí añadiremos el valor categoría la lista y, para después poder agruparla por categoría. En esta lista irán los productos y sus respectivas ventas, reseteo su valor para que cada producto tenga su sublista con su nombre y sus respectivas ventas.

```
93 #INICIO LISTA 10 PRODUCTOS MAS BUSCADOS#
94 productos_busqueda = []
95
96 for n in range(len(lifystore_products)):
97     y = []
98     x=0
99     name = lifystore_products[n][1]
100     for j in range(len(lifystore_searches)):
101         if lifystore_products[n][0] == lifystore_searches[j][1]:
102             x+=1
103     y.append(lifystore_products[n][0])
104     y.append(name)
105     y.append(x)
106     productos_busqueda.append(y)
107
108 #productos_busqueda es la lista de listas que contiene el
109 #producto y cuantas busquedas tiene [Producto, Busquedas]
110 #print(productos_busqueda)
111 print("---ID---||-----PRODUCTOS-----||
112 -----BUSQUEDAS-----")
113 #Utilizo el metodo sort y le agrego el parametro reverse para
114 #ordenar la lista de mayor a menor, utilizando la segunda
    posicion de la sublista.
115 #productos_busqueda.sort(key=lambda x: x[2], reverse=True)
116 #Utilizamos el metodo sort para odrenar
117 l = len(productos_busqueda)
118 for i in range(0, l):
```


LISTADO DE 5 PRODUCTOS MENOS VENDIDOS

Mismo Procedimiento que las listas anteriores, solo que aquí añadiremos el valor categoría la lista y, para después poder agruparla por categoría, después de ordenar los productos mostrando primero los que tienen menores ventas, ordenamos por categoría y de esa forma agrupamos los datos por categoría.

```
187 #LISTA 5 PRODUCTOS CON MENOS VENTAS, AGRUPADOS POR CATEGORIA#
188 productos_con_menorventa = []
189
190 #Mismo Procedimiento que las listas anteriores, solo que aqui
    añadiremos el valor categoria la lista y, para despues poder
    agruparla por categoria
191 for n in range(len(lifestore_products)):
192     y = []
193     x = 0
194     name = lifestore_products[n][
195         1] #Tomamos el nombre de la categoria del producto
196     category = lifestore_products[n][3]
197     for j in range(len(lifestore_sales)):
198         if (lifestore_products[n][0]
199             == lifestore_sales[j][1] and (
200                 lifestore_sales[j][4] == 0):
201             x += 1
202     y.append(name)
203     y.append(
204         category
205     ) #Insertamos el nombre de la categoria, en la lista
206     y.append(x)
207     productos_con_menorventa.append(
208         y
209     ) #La lista final que se añadira sera y = ["Producto",
    "Categoria", Ventas]
210
```

LISTADO CON LOS 10 PRODUCTOS MENOS BUSCADOS

Mismo Procedimiento que las listas anteriores, solo que aquí añadiremos el valor categoría la lista y, para después poder agruparla por categoría, después de ordenar los productos mostrando primero los que tienen menores ventas, ordenamos por categoría y de esa forma agrupamos los datos por categoría.

```
241 elif opcion == '4':
242     #LISTA 10 PRODUCTOS CON MENOS BUSQUEDAS, AGRUPADOS POR
    CATEGORIA#
243     productos_con_menorbusqueda = []
244     #Mismo Procedimiento que las listas anteriores, solo que aqui
        añadiremos el valor categoria la lista y, para despues poder
        agruparla por categoria
245     for n in range(len(lifestore_products)):
246         y = []
247         x = 0
248         name = lifestore_products[n][
249             1] #Tomamos el nombre de la categoria del producto
250         category = lifestore_products[n][3]
251         for j in range(len(lifestore_searches)):
252             if lifestore_products[n][0] == lifestore_searches[
253                 j][1]:
254                 x += 1
255         y.append(name)
256         y.append(
257             category
258         ) #Insertamos el nombre de la categoria, en la lista
259         y.append(x)
260         productos_con_menorbusqueda.append(
261             y
262         ) #La lista final que se añadira sera y = ["Producto",
            "Categoria", Busqueda]
263     #productos_con_menorventa es la lista de listas que contiene
```

20 PRODUCTOS CON MEJORES RESEÑAS

En este caso, se utilizó el método sort y le agrego el parámetro reverse para ordenar la lista de mayor a menor, utilizando la segunda posición de la sublista. Se procede a extraer solo los 20 primeros productos.

```
294 elif opcion == '5':
295     #PRODUCTOS CON MEJOR EVALUACION (SCORE = 5)#
296     productos_con_mejorevaluacion = []
297     for n in range(len(lifestore_products)):
298         y = []
299         x = 0
300         name = lifestore_products[n][1]
301         for j in range(len(lifestore_sales)):
302             if (lifestore_products[n][0]
303                 == lifestore_sales[j][1]) and (
304                 lifestore_sales[j][2] == 5):
305                 x += 1
306         if x > 0:
307             y.append(lifestore_products[n][0])
308             y.append(name)
309             y.append(x)
310             productos_con_mejorevaluacion.append(y)
311
312     print(
313         """---ID---||-----PRODUCTOS-----||
314         -----VECES QUE A SIDO EVALUADO CON
315         5-----"""
316     )
317     #Utilizo el metodo sort y le agrego el parametro reverse para
318     #ordenar la lista de mayor a menor, utilizando la segunda
319     #posicion de la sublista.
320     l = len(productos_con_mejorevaluacion)
```

20 PRODUCTOS CON PEORES RESEÑAS

En este caso, se utilizó el método sort y le agrego el parámetro reverse para ordenar la lista de mayor a menor, considerando un menor valor. En esta lista irán los productos y sus respectivas ventas, reseteo su valor para que cada producto tenga su sublista con su nombre y sus respectivas ventas.

```
372 #PRODUCTOS CON MENOR EVALUACION (SCORE = 1 ó Score = 2, se
373 #toma en cuenta 1 y 2 de calificacion)#
374 productos_con_menorevaluacion = []
375 for n in range(len(lifestore_products)):
376     y = []
377     x = 0
378     name = lifestore_products[n][1]
379     for j in range(len(lifestore_sales)):
380         if (lifestore_products[n][0]
381             == lifestore_sales[j][1]) and (
382             lifestore_sales[j][2] == 1
383             or lifestore_sales[j][2] == 2):
384             x += 1
385     if x > 0:
386         y.append(lifestore_products[n][0])
387         y.append(name)
388         y.append(x)
389         productos_con_menorevaluacion.append(y)
390
391     print(
392         """---ID---||-----PRODUCTOS-----||
393         -----VECES QUE A SIDO EVALUADO CON 1, 2
394         -----"""
395     )
396     #Utilizo el metodo sort y le agrego el parametro reverse para
397     #ordenar la lista de mayor a menor, utilizando la segunda
398     #posicion de la sublista.
399     l = len(productos_con_menorevaluacion)
```


TOTAL DE INGRESOS Y VENTAS, PROMEDIO ANUALES Y MENSUALES.

En esta ocasión, para el desarrollo sobre las ventas, por cada venta se indica sobre los productos y mediante un if se identifica cuales son las coincidencias. En esta lista irán los productos y sus respectivas ventas, reseteo su valor para que cada producto tenga su sublista con su nombre y sus respectivas ventas. Solo que aquí añadiremos el valor categoría la lista y, para después poder agruparla por categoría en este caso meses.

```
558 #VENTAS POR MES#
559 meses = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
560 meses_ventas = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
561 #Itero sobre las ventas, por cada venta itero sobre los
    productos y mediante un if identifico cuales son las
    coincidencias
562 for n in range(len(lifestore_sales)):
563     #y = [] #En esta lista iran los productos y sus
        respectivas ventas, reseteo su valor para que cada
        producto tenga su sublista con su nombre y sus respectivas
        ventas
564     x = 0
565     date = lifestore_sales[n][3]
566     mes = date.split(sep='/')
567     for j in range(len(lifestore_products)):
568         if (lifestore_sales[n][1]
569             == lifestore_products[j][0]) and (
570                 lifestore_sales[n][4] == 0):
571             if mes[1] == '01':
572                 meses_ventas[0] += lifestore_products[j][2]
573                 meses[0] += 1
574             if mes[1] == '02':
575                 meses_ventas[1] += lifestore_products[j][2]
576                 meses[1] += 1
577             if mes[1] == '03':
578                 meses_ventas[2] += lifestore_products[j][2]
579                 meses[2] += 1
580             if mes[1] == '04':
```

5 PRODUCTOS MEJOR VENDIDOS

1. SSD Kingston A400
2. Procesador AMD Ryzen 5 2600
3. Procesador Intel Core i3-9100F
4. Tarjeta Madre ASRock Micro ATX B450M Steel
5. SSD Adata Ultimate SU800

10 PRODUCTOS MÁS BUSCADOS

1. Procesador AMD Ryzen 5 2600
2. Procesador Intel Core i3-9100F
3. Tarjeta Madre ASRock Micro ATX B450M Steel
4. SSD Adata Ultimate SU800
5. Procesador AMD Ryzen 3 3200G
6. Logitech Audífonos Gamer G635 7.1
7. TV Monitor LED 24TL520S-PU 24
8. Procesador Intel Core i7-9700K
9. Procesador Intel Core i3-9100F
10. SSD XPG SX8200 Pro, 256GB

PRODUCTOS MEJOR CALIFICADOS

1. Procesador Intel Core i7-9700K
2. Procesador Intel Core i5-9600K
3. Kit SSD Kingston KC600
4. ASUS AMD Radeon RX 570
5. Procesador Intel Core i9-9900K

PRODUCTOS PEOR CALIFICADOS

1. SSD KINGSTON A400, 120GB
2. PROCESADOR AMD RYZEN 5 2600
3. PROCESADOR INTEL CORE I3-9100F
4. SSD ADATA ULTIMATE SU800

EVIDENCIAS

LOGIN

```
-----
|  LIFESTORE®  |
-----
¿Estas Registrado? si/no (Ingresa q para salir)
si
Ingresa tu usuario: Gerencia
Ingresa tu contraseña: 358554

====> Has iniciado sesión como  Gerencia @
```

CONTENIDO

```
-----CONTENIDO-----
-----
|  1.- Lista de los 5 Productos con Mayores Ventas
|
|  2.- Lista de los 10 Productos con Mayores Busquedas
|
|  3.- Lista de los 5 Productos con Menores Ventas (Agrupados por categ
oría) |
|  4.- Lista de los 10 Productos con Menores Busquedas (Agrupados por c
ategoría) |
|  5.- Lista de los Productos con Mejor Reseña (Unicamente con 5 de sco
re) |
|  6.- Lista de los Productos con Mejor Reseña (5 o 4 de Score)
|
|  7.- Lista de los Productos con Peor Reseña (1 o 2 de Score)
|
|  8.- Lista de los Productos con Peor Reseña (Unicamente con 1 de
e) |
|  9.- TOTAL DE INGRESOS, DINERO EN DEVOLUCIONES
```


LISTA 1. PRODUCTOS MÁS VENDIDOS

Selecciona una opcion...1

=====>NO SE INCLUYEN PRODUCTOS CON DEVOLUCION

--ID--||-----PRODUCTOS-----|| -

-----VENTAS-----||-----STOCK----

1 .- 54 || SSD Kingston A400, 120GB, SATA III, 2.5'
' , 7mm || 49 || 300 ||

2 .- 3 || Procesador AMD Ryzen 5 2600, S-AM4, 3.40G
Hz, Six-Core, 16MB L3 Cache, con Disipador Wraith Ste
alth || 42 || 987 ||

3 .- 5 || Procesador Intel Core i3-9100F, S-1151, 3
.60GHz, Quad-Core, 6MB Cache (9na. Generación - Coffe
e Lake) || 20 || 130 ||

4 .- 42 || Tarjeta Madre ASRock Micro ATX B450M Ste
el Legend, S-AM4, AMD B450, HDMI, 64GB DDR4 para AMD
|| 18 || 0 ||

5 .- 57 || SSD Adata Ultimate SU800, 256GB, SATA II
I, 2.5'', 7mm || 15 || 15 ||

LISTA 2. PRODUCTOS MÁS BUSCADOS

Selecciona una opcion...2

---ID---||-----PRODUCTOS-----|| -----B

USQUEDAS-----

1 .- 54 || SSD Kingston A400, 120GB, SATA III, 2.5'', 7mm || 263
||

2 .- 57 || SSD Adata Ultimate SU800, 256GB, SATA III, 2.5'', 7mm
|| 107 ||

3 .- 29 || Tarjeta Madre ASUS micro ATX TUF B450M-PLUS GAMING, S-A
M4, AMD B450, HDMI, 64GB DDR4 para AMD || 60 ||

4 .- 3 || Procesador AMD Ryzen 5 2600, S-AM4, 3.40GHz, Six-Core, 1
6MB L3 Cache, con Disipador Wraith Stealth || 55 ||

5 .- 4 || Procesador AMD Ryzen 3 3200G con Gráficos Radeon Vega 8,
S-AM4, 3.60GHz, Quad-Core, 4MB L3, con Disipador Wraith Spire ||
41 ||

6 .- 85 || Logitech Audifonos Gamer G635 7.1, Alámbrico, 1.5 Metro
s, 3.5mm, Negro/Azul || 35 ||

7 .- 67 || TV Monitor LED 24TL520S-PU 24, HD, Widescreen, HDMI, Ne
gro || 32 ||

8 .- 7 || Procesador Intel Core i7-9700K, S-1151, 3.60GHz, 8-Core,
12MB Smart Cache (9na. Generación Coffee Lake) || 31 ||

9 .- 5 || Procesador Intel Core i3-9100F, S-1151, 3.60GHz, Quad-Co
re, 6MB Cache (9na. Generación - Coffee Lake) || 30 ||

10 .- 47 || SSD XPG SX8200 Pro, 256GB, PCI Express, M.2 || 30 |
|

PRODUCTOS MENORES VENTAS

Selecciona una opcion...3

-----PRODUCTOS MENOS VENTAS-----

- 1 .- ASUS Audífonos Gamer ROG Theta 7.1, Alámbrico, USB C, Negro || audifonos || 0 ||
- 2 .- Acer Audífonos Gamer Galea 300, Alámbrico, 3.5mm, Negro | | audifonos || 0 ||
- 3 .- Audífonos Gamer Balam Rush Orphix RGB 7.1, Alámbrico, USB, Negro || audifonos || 0 ||
- 4 .- Energy Sistem Audífonos con Micrófono Headphones 1, Bluetooth, Inalámbrico, Negro/Grafito || audifonos || 0 ||
- 5 .- Genius GHP-400S Audífonos, Alámbrico, 1.5 Metros, Rosa || audifonos || 0 ||

PRODUCTOS MEJORES RESEÑAS

Selecciona una opcion...5

---ID---||-----PRODUCTOS-----|| -----VE
CES QUE A SIDO EVALUADO CON 5-----

- 1 .- 54 || SSD Kingston A400, 120GB, SATA III, 2.5'', 7mm || 38 ||
- 2 .- 3 || Procesador AMD Ryzen 5 2600, S-AM4, 3.40GHz, Six-Core, 16MB L3 Cache, con Disipador Wraith Stealth || 35 ||
- 3 .- 5 || Procesador Intel Core i3-9100F, S-1151, 3.60GHz, Quad-Core, 6MB Cache (9na. Generación - Coffee Lake) || 14 ||
- 4 .- 57 || SSD Adata Ultimate SU800, 256GB, SATA III, 2.5'', 7mm || 13 ||
- 5 .- 42 || Tarjeta Madre ASRock Micro ATX B450M Steel Legend, S-AM4, AMD B450, HDMI, 64GB DDR4 para AMD || 10 ||

PRODUCTOS PEORES RESEÑAS

Selecciona una opcion...8

---ID---||-----PRODUCTOS-----|| -----VE
CES QUE A SIDO EVALUADO CON 1 -----

1 .- 31 || Tarjeta Madre AORUS micro ATX B450 AORUS M (rev. 1
.0), S-AM4, AMD B450, HDMI, 64GB DDR4 para AMD || 4 ||

2 .- 29 || Tarjeta Madre ASUS micro ATX TUF B450M-PLUS GAMING
, S-AM4, AMD B450, HDMI, 64GB DDR4 para AMD || 2 ||

3 .- 17 || Tarjeta de Video Gigabyte AMD Radeon R7 370 OC, 2G
B 256-bit GDDR5, PCI Express 3.0 || 1 ||

4 .- 45 || Tarjeta Madre ASRock ATX H110 Pro BTC+, S-1151, In
tel H110, 32GB DDR4, para Intel || 1 ||

TOTAL DE INGRESOS

B C, Negro || 0 || 0 ||

89 .- 87 || Acer Audifonos Gamer Galea 300, Alámbrico, 3.5mm,
Negro || 0 || 0 ||

90 .- 88 || Audifonos Gamer Balam Rush Orphix RGB 7.1, Alámbr
ico, USB, Negro || 0 || 0 ||

91 .- 90 || Energy Sistem Audifonos con Micrófono Headphones
1, Bluetooth, Inalámbrico, Negro/Grafito || 0 || 0 ||

92 .- 91 || Genius GHP-400S Audifonos, Alámbrico, 1.5 Metros,
Rosa || 0 || 0 ||

93 .- 92 || Getttech Audifonos con Micrófono Sonority, Alámbr
ico, 1.2 Metros, 3.5mm, Negro/Rosa || 0 || 0 ||

94 .- 93 || Ginga Audifonos con Micrófono GI18ADJ01BT-R0, Blu
etooth, Alámbrico/Inalámbrico, 3.5mm, Rojo || 0 || 0 ||

95 .- 95 || Iogear Audifonos Gamer GHG601, Alámbrico, 1.2 Met
ros, 3.5mm, Negro || 0 || 0 ||

96 .- 96 || Klip Xtreme Audifonos Blast, Bluetooth, Inalámbr
ico, Negro/Verde || 0 || 0 ||

-----TOTAL DE INGRESOS =====> 737916

-----DEVOLUCIONES =====> 22261

VENTAS POR MES

```
-----  
Selecciona una opcion...10  
--MES--||---NUMERO DE VENTAS---||--TOTAL INGRESOS POR MES--  
  
Enero: || 52 || $ 117738  
Febrero: || 40 || $ 107270  
Marzo: || 49 || $ 162931  
Abril: || 74 || $ 191066  
Mayo: || 34 || $ 91936  
Junio: || 11 || $ 36949  
Julio: || 11 || $ 26949  
Agosto: || 3 || $ 3077  
Septiembre: || 0 || $ 0  
Octubre: || 0 || $ 0  
Noviembre: || 0 || $ 0  
Diciembre: || 0 || $ 0  
  
-----  
---  
TOTALES: 274 || $ 737916 ||  
-----  
---  
PROMEDIO MENSUAL: $ 61493.0  
-----
```

CÓDIGO COMPLETO



SOLUCIÓN

Durante el desarrollo del proyecto se logró cumplir con lo solicitado y obtener los datos requeridos, partiendo de lo solicitado se pudo concretar que:

- De un total de 96 productos disponibles, solo se vendieron 41 (sin contar devoluciones), lo que significa que 55 productos del catalogo no registraron ninguna venta.
- De las 283 ventas reportadas, 9 fueron devoluciones, significa que las ventas reales fueron de 274.
- Existe una venta superior de 8 tarjetas madres, de las 18 que se encuentran en el inventario
- El valor de las devoluciones fue por \$22,261.
- Existen artículos que reportaron ventas y ya no tienen stock.

Existen varios productos que tienen un gran índice de poca preferencia en cuanto a las ventas superiores de otros modelos, productos que tienen una gran despectiva ante el cliente en cuanto a la poca calificación que existe de ellos.

Como alternativa a una solución eficaz y eficiente para la empresa, considerando el uso de los datos obtenidos seria replantear el stock de la mercancía que no es estacionalmente atractiva para el cliente y ofreciendo ciertas ofertas para poder crear una necesidad atractiva en el cliente. De la misma manera se podría usar como estrategia una investigación de mercadeo donde se envíe encuestas a los cliente que mostraron poca preferencia con ciertos productos para saber cuales son los principales factores de la calificación negativa obtenida.