

PROYECTO 1. INTRODUCCIÓN A PYTHON



Solución del Proyecto 1
Jorge Blaborne Velasco Ortega
México, Puebla.
06 de Septiembre 2020.

Índice

Introducción	3
Definición del código	4-11
Solución al problema	12-16
Conclusión	17
Contraseñas y Link	18

Introducción

El objetivo de este proyecto es poner en práctica las bases de programación en Python para análisis y clasificación de datos mediante la creación de programas de entrada de usuario y validaciones, uso y definición de variables y listas, operadores lógicos y condicionales para la clasificación de información visto durante el curso impartido por EMTECH, patrocinado por Santander.

Tenemos el caso de LifeStore es una tienda virtual que maneja una amplia gama de artículos, recientemente, la Gerencia de ventas, se percató que la empresa tiene una importante acumulación de inventario. Asimismo, se ha identificado una reducción en las búsquedas de un grupo importante de productos, lo que ha redundado en una disminución sustancial de sus ventas del último trimestre.

Derivado de la situación, la Gerencia de Ventas te solicita que realices un análisis de la rotación de productos identificando los siguientes elementos:

- 1) Productos más vendidos y productos rezagados a partir del análisis de las categorías con menores ventas y categorías con menores búsquedas.
- 2) Productos por reseña en el servicio a partir del análisis de categorías con mayores ventas y categorías con mayores búsquedas.
- 3) Sugerir una estrategia de productos a retirar del mercado así como sugerencia de cómo reducir la acumulación de inventario considerando los datos de ingresos y ventas mensuales.

El análisis deberá considerar el desarrollo de un sistema de análisis, en el que mediante un Login de usuario-administrador se muestre un reporte mensual que especifique los puntos señalados en la consigna, específicamente:

Productos más vendidos y productos rezagados.

Productos por reseña en el servicio.

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

Definición del código. Código explicado describiendo sus procesos y variables.

1.- Login de usuario

```
1  from lifestore_file import lifestore_products
2  #print (lifestore_products)
3
4  from lifestore_file import lifestore_sales
5  #print (lifestore_sales)
6
7  from lifestore_file import lifestore_searches
8  #print (lifestore_searches)
9
10 Productos = lifestore_products
11
12 #print(Productos)
13
14 Ventas = lifestore_sales
15 #print(Ventas)
16
17 Busquedas = lifestore_searches
18 #print(Busquedas)
```

Las primeras líneas de mi código lo que hacen es exportar los datos de cada una de las variables y de las listas de mi base de datos que están en el archivo lifestore_file, con el comando from y el comando import.

Al ya tener exportadas estas listas lo que hago es darle un nombre clave a cada una de esas listas para tener mejor conocimiento de mis variables, asignándoles el nombre de lo que contenía cada una de mis listas para así trabajar con ellas, como lo fueron las listas de Productos, Ventas y Busquedas.

```
21 listadecon = [["Javier","ELLC"],["Jorge","ELBLABO"]]
22
23 usuario = input("Ingresa tu Nombre ")
24 contraseña = input("Ingresa la contraseña ")
25
26 acc = 0
27 n_veces = 1
28
29 for u in listadecon :
30     if usuario == u[0] and contraseña == u[1]:
31         acc = acc + 1
32
33
34
35 while acc == 1 and n_veces < 2 :
36     print("Bienvenido al programa de Jorge Blaborne","\n\n" )
37
```

De la línea 21 a la 37 puedo decir que es la parte numero 1 de la consigna en la cual nos pide hacer un inicio de sesión.

En la línea 21 defino una lista con dos elementos que son los 2 únicos usuarios con sus respectivas contraseñas que serán capaz de poder acceder al programa, estos son Javier con la contraseña ELLC y Jorge con la contraseña ELBLABO.

En la línea 23 y 24 pedimos al usuario con la función input que escriba su nombre y su contraseña respectiva para poder acceder al programa.

En la línea 26 y 27 inicializamos unos contadores que nos servirán para poder ejecutar el programa sin que esté se quede ciclado.

De la línea 31-29 inicializamos un ciclo for en el que verifica si la contraseña y el usuario ingresados son los correctos, de ser así el contador acc incrementa 1 su valor inicial.

Ese contador acc sirve para indicarle al programa que los datos son correctos y pueda acceder al programa ,mediante un ciclo while en el que cumple esa condición y a parte de esa condición cumple también la del otro contador que es n_veces esto para que solo una vez se ejecute el ciclo y no se repita infinitamente como lo hacen os ciclos While, esto lo hace porque al final del código aparece el contador n_veces aumentando 1 su valor y haciendo que esa condición ya no sea real y así deteniendo el bucle.

Este bucle While esta indentado que cubre todo el programa, y si todo esta correcto ejecuta la leyenda de “Bienvenido al Programa de Jorge Blaborne”

2.- Variables, definiciones y Validaciones

```
39 ##### Busquedas #####
40 lista = []
41 for i in Productos:
42     conta=0
43     for k in Busquedas:
44         if k[1] == i[0] :
45             conta = conta + 1
46
47     lista.append(conta)
48 #print(lista)
49 lista2 = []
50 con = 0
51 for y in lista:
52     con = con + 1
53     #xd = ("El ID ", con , "Tiene ", y , "Busquedas")
54     xd = y , "Busquedas, tiene el articulo ", con
55
56     lista2.append(xd)
57
58 lp = sorted(lista2, reverse = True)
59 lp1 = sorted(lista2, reverse = False)
60
61 print("Los 20 productos más buscados son : ", "\n\n")
62 print(lp[0:20], "\n\n")
63 print("Los 40 productos que no fueron buscados son : ", "\n\n")
64 print(lp1[0:40], "\n\n")
65
```

En la línea 40 inicializo una lista vacía que luego en el for de la línea 41 a la 45 se irá llenando, este bucle for lo que hace es que va checando en toda la lista de productos sus ID que se encuentran en la posición [0] de su lista y los compara con su mismo ID en la lista de Busquedas que se localiza en la posición [1] es por eso en el if aparece que si son iguales el contador designados aumentara cada vez que los ID coincidieran, este proceso lo utilizo mucho en el programa por lo cual lo llamaré PROCESO ID, en este documento para hacer referencia en el futuro.

Lo que contiene esa lista ahora es cuantas veces aparecen los productos en la lista Busquedas, por lo cual en otra lista en blanco en la línea 49 iré llenando con otro for en la línea 51 el número del producto con el número de búsquedas para tenerlo ya concentrado en una nueva lista.

Y a esa nueva lista llamada lista2 con la función sorted acomodo de manera ascendente, para que me muestre desde el producto más buscado hasta el menos buscado, todo eso lo guardo en la variable lp, la cual imprimo en la línea 62 pero solo sus primeros 20 elementos. Caso contrario con la línea 59 en la que puse que los dato fueran del menor al mayor, para poder denotar cuales fueron los artículos que no fueron buscados, que en este caso fueron 40.

```

68 ##### Ventas #####
69 lista3 = []
70 for i in Productos:
71     conta=0
72     for k in Ventas:
73         if k[1] == i[0] :
74             conta = conta + 1
75
76     lista3.append(conta)
77
78 #print(lista3)
79 lista4 = []
80 con = 0
81 for y in lista3:
82     con = con + 1
83     #xd = ("El ID ", con , "Tiene ", y , "Busquedas")
84     xd = (y , "Ventas, tiene el articulo", con)
85
86     lista4.append(xd)
87
88 lp2 = sorted(lista4, reverse = True)
89 lp3 = sorted(lista4, reverse = False)
90
91 print("Los 20 productos más vendidos son : ", "\n\n")
92 print(lp2[0:20], "\n\n")
93 print("Los 54 productos que NO fueron vendidos son : ", "\n\n")
94 print(lp3[0:54], "\n\n")
95

```

Para encontrar el producto con más ventas hice el mismo procedimiento de PROCESO ID solo que en lugar de la lista de Busquedas utilicé la lista de Ventas, y con los print de las líneas salen los listados de los 20 productos más vendidos y los 54 artículos que no fueron comprados por los clientes.

```

97 ##### Por Categoría #####
98
99 ### Menores Ventas de categoría de Discos Duros ###
100 lista5 = []
101
102 for i in Productos:
103     conta=0
104     for k in Ventas:
105         if k[1] == i[0] and i[3] == "discos duros" :
106             conta = conta + 1
107
108     lista5.append(conta)
109 #print(lista5)
110
111 lista6 = []
112 con = 0
113 for y in lista5:
114     con = con + 1
115     #xd = ("El ID ", con , "Tiene ", y , "Busquedas")
116     xd = (y , "Ventas, tiene el articulo", con)
117
118     lista6.append(xd)
119
120 lp3 = sorted(lista6[46:59], reverse = False)
121
122 print("Las Ventas de la categoría Discos Duros son :""\n\n")
123 print(lp3, "\n\n")

```

Después en las línea 99 a 123 filtre las ventas de la categoría “Discos Duros” ya que fue la que tenía los productos más vendidos, por lo que realice un procesos similar a los anteriores solo que le agregue la condición AND en la condición del if con la posición 3 de la lista Productos, para filtrarlos, y para imprimirlos correctamente en los índices los agarre de sus posiciones por sus índices de la 46:59 en donde se localizan los productos de esa categoría.

```

129  ## Menores Ventas de la categoría Procesadores ###
130  lista5 = []
131
132  for i in Productos:
133      conta=0
134      for k in Ventas:
135          if k[1] == i[0] and i[3] == "procesadores" :
136              conta = conta + 1
137
138          lista5.append(conta)
139  #print(lista5)
140
141  lista6 = []
142  con = 0
143  for y in lista5:
144      con = con + 1
145      #xd = ("El ID ", con , "Tiene ", y , "Busquedas")
146      xd = (y , "Ventas, tiene el articulo", con)
147
148      lista6.append(xd)
149
150  lp3 = sorted(lista6[0:9], reverse = False)
151
152  print("Las Ventas de la categoría Procesadores son :""\n\n")
153  print(lp3,"\n\n")

```

Exactamente el mismo procedimiento que la imagen pasada solo que adaptada con la categoría de Procesadores.

Con lo anterior cumplimos con el paso 1 que era: Productos más vendidos y productos rezagados:

Generar un listado de los 50 productos con mayores ventas y uno con los 100 productos con mayores búsquedas. Por categoría, generar un listado con los 50 productos con menores ventas y uno con los 100 productos con menores búsquedas.

Ahora sigue el paso 2 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.

En este caso mostré 3 listados que a continuación mostraré como los saque


```

157 ##### Productos Reseña en el Servicio ###
158
159 lista7 = []
160
161 for i in Productos:
162     conta=0
163     for k in Ventas:
164         if k[1] == i[0] and k[2] == 5 :
165             conta = conta + 1
166
167     lista7.append(conta)
168
169 #print(lista7)
170 lista8 = []
171 con = 0
172 for y in lista7:
173     con = con + 1
174 #xd = ("El ID ", con , "Tiene ", y , "Busquedas")
175     xd = (y , "Calificaciones de 5 tiene el producto " , con)
176
177     lista8.append(xd)
178
179
180 lp3 = sorted(lista8, reverse = True)
181
182 print("Los 20 productos con mejores reseñas son : " , "\n\n")
183 print(lp3[0:20], "\n\n")
184

```

Esta subrutina que viene desde la línea 159 sigue el mismo razonamiento pasado de PROCESO ID pero agregando la condición AND para filtrar ahora los productos con mejores reseñas que tienen calificación de 5 es por eso por lo que en la lista de Ventas en la posición [2] muestra que sea estrictamente el valor de 5. Y se sigue la misma lógica para crear la lista con el for de la línea 172, imprimiendo la lista en la línea 183.

```

186 ##### Peor Calificación #####
187 lista5 = []
188
189 for i in Productos:
190     conta=0
191     for k in Ventas:
192         if k[1] == i[0] and (k[2] == 1 or k[2] == 2 or k[2] == 3) :
193             conta = conta + 1
194
195     lista5.append(conta)
196 #print(lista5)
197
198 lista6 = []
199 con = 0
200 for y in lista5:
201     con = con + 1
202 #xd = ("El ID ", con , "Tiene ", y , "Busquedas")
203     xd = (y , "Opiniones debajo de la media tiene el prodcto",
204           con)
205
206     lista6.append(xd)
207
208
209 lp3 = sorted(lista6, reverse = True)
210
211 print("Los 12 productos con PEORES reseñas son : " , "\n\n")
212 print(lp3[0:12], "\n\n")
213

```

Para encontrar los peores productos con las menores calificaciones se hizo exactamente que el proceso pasado, pero con la variante en la línea 192, en la que en lugar de obligar un 5 en la condición de AND, se utilizan condiciones de OR en los cuales cuentan las calificaciones más bajas ya sea 1,2 ó 3, por lo tanto ya consideran los peores productos, como se ve en el filtrado de la indexación del print de la línea 210 solo fueron 12 productos con muy bajas calificaciones.

```

213 ##### Productos devueltos #####
214 lista5 = []
215
216 for i in Productos:
217     conta=0
218     for k in Ventas:
219         if k[1] == i[0] and k[4] == 1 :
220             conta = conta + 1
221
222     lista5.append(conta)
223
224 #print(lista5)
225
226 lista6 = []
227 con = 0
228 for y in lista5:
229     con = con + 1
230     #xd = ("El ID ", con , "Tiene ", y , "Busquedas")
231     xd = (y , "Devoluciones tiene el producto", con)
232
233     lista6.append(xd)
234
235
236 lp3 = sorted(lista6, reverse = True)
237 print("Los 7 productos devueltos fueron :","\n\n")
238 print(lp3[0:7],"\n\n")

```

Para los productos devueltos se sigue la misma lógica que los pasados, solo que en la línea de la condición if de la línea 219 en la 5ta posición de la lista de Ventas se observa si se devolvió o no algún artículo con el número 1 como operados binario, por lo cual se denota que solo 7 productos fueron devueltos en la línea 238.

```

245 ##### Punto 3 Ingresos por mes y año ###
246 lista5 = []
247
248 for i in Productos:
249     conta=0
250     for k in Ventas:
251         if k[1] == i[0] :
252             conta = conta + 1
253
254     lista5.append(conta)
255
256 #print(lista5)
257
258 listadePrecios = []
259 for i in range(0,96):
260     Precios = (Productos [i][2] * lista5 [i] )
261     listadePrecios.append(Precios)
262
263
264 #print(listadePrecios)
265
266 d = sum(listadePrecios)
267
268 print ("Las ventas toales de la tienda fueron: ", d ,"\n\n")
269 print("Las ventas totales del año 2020 fueron 759,695 Pesos")
270 print("Las ventas totales del año 2002 fueron 300 Pesos")
271 print("Las ventas totales del año 2019 fueron 182 Pesos",
272       "\n\n")

```

Con lo pasado terminamos el punto 2, ahora solo faltaba el punto 3 que era el Total de ingresos y ventas promedio mensuales, total anual y meses con más ventas al año, por lo cual, en la línea 254 con ayuda de un for llene todos los precios de los productos y en la línea 260 los multiplique por el número de artículos que se vendió de cada producto, para tener el total de lo vendido por la tienda, ahora bien, a eso que me resulto le reste lo de 2 ventas que no fueron en ese año que fue la venta con el ID 219 que fue en el año 2002

(seguramente es un outlier) y la venta con ID 15 que fue en el mes de noviembre de 2019. Es por eso las otras especificaciones.

```
273 ##### Ventas por Mes ###
274 lista5 = []
275 fechas = []
276
277 for i in Productos:
278     conta=0
279     for k in Ventas:
280
281         if k[1] == i[0] :
282             conta = conta + 1
283             fechas.append(k[3])
284         lista5.append(conta)
285 #print(lista5)
286 #print(fechas)
287 lista_años = []
288 for año in fechas:
289     lista_años.append((año[3:5], int(año[6:10])))
290 #print(lista_años)
291 listaorda = sorted(lista_años, reverse = False)
292
293 #print(listaorda)
294 conta = 0
295 for k in listaorda:
296     if k [0] == "01" :
297         conta = conta + 1
298         #print(conta)
299
300 print("Hubo", conta, "Ventas en Enero","\n\n")
---
```

Por último en el programa y en la encomienda, saque las ventas por mes, realmente me quebré un poco la cabeza ahí, pero al muy final salió bien, lo que hice fue hacer el mismo procedimiento de ID PROCESO, pero ahora con las fechas que aparece en la lista ventas, y las guarde en la lista5 de la línea 284. Después de eso en la línea 287 cree una lista en vacía en donde con el for de la línea siguiente pude guardar solo los meses y año de cada fecha quitando los días, después ordenándolos de menor a mayor es decir de enero que es el mes 01 al ultimo mes que saliera en la lista de ventas esto en la línea 291.

Al tener ya en listaorda una lista con todas las fechas de las ventas ordenas y solo por mes y año hice un for en el que recorrí toda la listaorda y me contará todos los elementos del mes 01 que era de enero, sacando cuantas ventas se tuvieron en el mes de enero en la línea 300, este proceso lo repetí 10 veces, debido a que como el formato era de 01, 02 etc no pude hacer un for o range para recorrer los meses así, es por eso que este último código lo repetí 10 veces cambiando la variable de verificación de la línea 297 en lugar de 01 para enero, cambiar a 02 para Febrero y así sucesivamente como se muestra a continuación.

```

302     conta = 0
303     for k in listaorda:
304         if k [0] == "02" :
305             conta = conta + 1
306             #print(conta)
307
308     print("Hubo", conta, "Ventas en Febrero","\n\n")
309
310
311     conta = 0
312     for k in listaorda:
313         if k [0] == "03" :
314             conta = conta + 1
315             #print(conta)
316
317     print("Hubo", conta, "Ventas en Marzo","\n\n")
318
319     conta = 0
320     for k in listaorda:
321         if k [0] == "04" :
322             conta = conta + 1
323             #print(conta)
324
325     print("Hubo", conta, "Ventas en Abril","\n\n")
326
327     conta = 0
328     for k in listaorda:
329         if k [0] == "05" :
330             conta = conta + 1

```

```

366     print("Hubo", conta, "Venta en Septiembre","\n\n")
367
368     conta = 0
369     for k in listaorda:
370         if k [0] == "11" :
371             conta = conta + 1
372             #print(conta)
373
374     print("Hubo", conta, "Ventas en Noviembre de 2019","\n\n")
375
376     n_veces += 1
377
378     print("SI Llegaste hasta aquí omite el siguiente mensaje")
379
380     print("Contraseña Incorrecta, vuelve a correr el programa cuando
    te acuerdes de la contraseña por favor")
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000

```

Así con el mismo procedimiento anterior hasta la línea 378, así consiguiendo las ventas por mes de la tienda.

Por último en el código en la línea 376 aparece el contador/bandera con la cual aumentamos el valor de nuestro contador del while, para que se detenga en la primera iteración.

En la línea 378 un mensaje como manejo de errores para delimitar que si corre bien el código se haga caso omiso al mensaje que aparece fuera del ciclo while cuando un usuario no puede ingresar al programa.

Solución al problema.

```
[ (263, 'Busquedas, tiene el articulo ', 54)
, (107, 'Busquedas, tiene el articulo ', 57)
), (60, 'Busquedas, tiene el articulo ', 29)
), (55, 'Busquedas, tiene el articulo ', 3)
, (41, 'Busquedas, tiene el articulo ', 4),
(35, 'Busquedas, tiene el articulo ', 85),
(32, 'Busquedas, tiene el articulo ', 67),
(31, 'Busquedas, tiene el articulo ', 7),
(30, 'Busquedas, tiene el articulo ', 47),
(30, 'Busquedas, tiene el articulo ', 5), (
27, 'Busquedas, tiene el articulo ', 48), (
25, 'Busquedas, tiene el articulo ', 44), (
24, 'Busquedas, tiene el articulo ', 2), (2
3, 'Busquedas, tiene el articulo ', 42), (2
0, 'Busquedas, tiene el articulo ', 8), (15
, 'Busquedas, tiene el articulo ', 66), (15
, 'Busquedas, tiene el articulo ', 21), (15
, 'Busquedas, tiene el articulo ', 12), (11
, 'Busquedas, tiene el articulo ', 51), (11
, 'Busquedas, tiene el articulo ', 18)]

[(50, 'Ventas, tiene el articulo', 54)
, (42, 'Ventas, tiene el articulo', 3)
, (20, 'Ventas, tiene el articulo', 5)
, (18, 'Ventas, tiene el articulo', 42)
), (15, 'Ventas, tiene el articulo', 5
7), (14, 'Ventas, tiene el articulo',
29), (13, 'Ventas, tiene el articulo',
4), (13, 'Ventas, tiene el articulo',
2), (11, 'Ventas, tiene el articulo',
47), (9, 'Ventas, tiene el articulo',
48), (9, 'Ventas, tiene el articulo',
12), (7, 'Ventas, tiene el articulo',
7), (6, 'Ventas, tiene el articulo',
44), (6, 'Ventas, tiene el articulo',
31), (5, 'Ventas, tiene el articulo',
18), (4, 'Ventas, tiene el articulo',
8), (3, 'Ventas, tiene el articulo', 5
1), (3, 'Ventas, tiene el articulo', 4
9), (3, 'Ventas, tiene el articulo', 1
1), (3, 'Ventas, tiene el articulo', 6
```

Podemos denotar que el producto más vendido y con más ventas es el producto 54 que pertenece a la categoría de Discos Duros, posicionándose como el producto estrella de la tienda, aunque también es importante mencionar que el artículo 57 también parte de los discos duros fue el segundo más buscado y el 5to más vendido, colocando a los discos duros como una de las categorías líderes de la tienda.

Después denotamos que el segundo producto más vendido y el 4to mas buscado fue el artículo 3 que es parte de los procesadores, también es interesante denotar que el 3er y 7mo artículo más vendidos pertenece a la categoría de Procesadores, colocándola como una de mas categorías más importantes de la tienda.

Al encontrar las dos categorías más importantes de la tienda podemos hacer un análisis más detallado de estas dos categorías, y así filtrar cuales fueron las ventas de cada una de esas categorías.

```
[(0, 'Ventas, tiene el articulo', 9),
(0, 'Ventas, tiene el articulo', 14),
(0, 'Ventas, tiene el articulo', 15),
(0, 'Ventas, tiene el articulo', 16),
(0, 'Ventas, tiene el articulo', 19),
(0, 'Ventas, tiene el articulo', 20),
(0, 'Ventas, tiene el articulo', 23),
(0, 'Ventas, tiene el articulo', 24),
(0, 'Ventas, tiene el articulo', 26),
(0, 'Ventas, tiene el articulo', 27),
(0, 'Ventas, tiene el articulo', 30),
(0, 'Ventas, tiene el articulo', 32),
(0, 'Ventas, tiene el articulo', 34),
(0, 'Ventas, tiene el articulo', 35),
(0, 'Ventas, tiene el articulo', 36),
(0, 'Ventas, tiene el articulo', 37),
(0, 'Ventas, tiene el articulo', 38),
(0, 'Ventas, tiene el articulo', 39),
(0, 'Ventas, tiene el articulo', 41),
(0, 'Ventas, tiene el articulo', 43),
(0, 'Ventas, tiene el articulo', 53),
(0, 'Ventas, tiene el articulo', 55),
(0, 'Ventas, tiene el articulo', 56),
(0, 'Ventas, tiene el articulo', 58),
(0, 'Ventas, tiene el articulo', 59),
(0, 'Ventas, tiene el articulo', 61),
(0, 'Ventas, tiene el articulo', 62),
(0, 'Ventas, tiene el articulo', 63),
(0, 'Ventas, tiene el articulo', 64),
(0, 'Ventas, tiene el articulo', 65),
(0, 'Ventas, tiene el articulo', 68),
(0, 'Ventas, tiene el articulo', 69),
(0, 'Ventas, tiene el articulo', 70),

(0, 'Ventas, tiene el articulo', 71),
(0, 'Ventas, tiene el articulo', 72),
(0, 'Ventas, tiene el articulo', 73),
(0, 'Ventas, tiene el articulo', 75),
(0, 'Ventas, tiene el articulo', 76),
(0, 'Ventas, tiene el articulo', 77),
(0, 'Ventas, tiene el articulo', 78),
(0, 'Ventas, tiene el articulo', 79),
(0, 'Ventas, tiene el articulo', 80),
(0, 'Ventas, tiene el articulo', 81),
(0, 'Ventas, tiene el articulo', 82),
(0, 'Ventas, tiene el articulo', 83),
(0, 'Ventas, tiene el articulo', 86),
(0, 'Ventas, tiene el articulo', 87),
(0, 'Ventas, tiene el articulo', 88),
(0, 'Ventas, tiene el articulo', 90),
(0, 'Ventas, tiene el articulo', 91),
(0, 'Ventas, tiene el articulo', 92),
(0, 'Ventas, tiene el articulo', 93),
(0, 'Ventas, tiene el articulo', 95),
(0, 'Ventas, tiene el articulo', 96)]
```

```

Los 40 productos que no fueron buscados son
:

[(0, 'Busquedas, tiene el articulo ', 14),
(0, 'Busquedas, tiene el articulo ', 16),
(0, 'Busquedas, tiene el articulo ', 19),
(0, 'Busquedas, tiene el articulo ', 20),
(0, 'Busquedas, tiene el articulo ', 23),
(0, 'Busquedas, tiene el articulo ', 24),
(0, 'Busquedas, tiene el articulo ', 30),
(0, 'Busquedas, tiene el articulo ', 32),
(0, 'Busquedas, tiene el articulo ', 33),
(0, 'Busquedas, tiene el articulo ', 34),
(0, 'Busquedas, tiene el articulo ', 36),
(0, 'Busquedas, tiene el articulo ', 37),
(0, 'Busquedas, tiene el articulo ', 38),
(0, 'Busquedas, tiene el articulo ', 41),
(0, 'Busquedas, tiene el articulo ', 43),
(0, 'Busquedas, tiene el articulo ', 53),
(0, 'Busquedas, tiene el articulo ', 55),
(0, 'Busquedas, tiene el articulo ', 58),
(0, 'Busquedas, tiene el articulo ', 60),
(0, 'Busquedas, tiene el articulo ', 61),
(0, 'Busquedas, tiene el articulo ', 62),
(0, 'Busquedas, tiene el articulo ', 64),
(0, 'Busquedas, tiene el articulo ', 65),
(0, 'Busquedas, tiene el articulo ', 68),
(0, 'Busquedas, tiene el articulo ', 69),
(0, 'Busquedas, tiene el articulo ', 71),
(0, 'Busquedas, tiene el articulo ', 72),
(0, 'Busquedas, tiene el articulo ', 75),
(0, 'Busquedas, tiene el articulo ', 77),
(0, 'Busquedas, tiene el articulo ', 78),
(0, 'Busquedas, tiene el articulo ', 79),
(0, 'Busquedas, tiene el articulo ', 81),
(0, 'Busquedas, tiene el articulo ', 82),
(0, 'Busquedas, tiene el articulo ', 83),
(0, 'Busquedas, tiene el articulo ', 86),
(0, 'Busquedas, tiene el articulo ', 87),
(0, 'Busquedas, tiene el articulo ', 88),
(0, 'Busquedas, tiene el articulo ', 90),
(0, 'Busquedas, tiene el articulo ', 92),
(0, 'Busquedas, tiene el articulo ', 96)]

```

Ahora bien con estos listados de los 40 productos que no fueron buscados y los 54 artículos que no fueron comprados en la tienda podemos denotar el gran desperdicio de stock que hay en la tienda, podemos ver que la generalmente quitando las dos categorías de Procesadores y Tarjetas Madre, las demás categorías aunque tienen una variedad de 10 productos aproximadamente, solo se venden ó llaman la atención del público 3-4 artículos por cada categoría, entonces realmente existe un problema de sobre inventario de la empresa, por lo que es recomendable rematar esos artículos y no comprarlos más.

```

Las Ventas de la categoría Discos Duros
son :

[(0, 'Ventas, tiene el articulo', 53),
(0, 'Ventas, tiene el articulo', 55),
(0, 'Ventas, tiene el articulo', 56),
(0, 'Ventas, tiene el articulo', 58),
(0, 'Ventas, tiene el articulo', 59),
(1, 'Ventas, tiene el articulo', 50),
(2, 'Ventas, tiene el articulo', 52),
(3, 'Ventas, tiene el articulo', 49),
(3, 'Ventas, tiene el articulo', 51),
(9, 'Ventas, tiene el articulo', 48),
(11, 'Ventas, tiene el articulo', 47),
(15, 'Ventas, tiene el articulo', 57),
(50, 'Ventas, tiene el articulo', 54)]

Las Ventas de la categoría Procesadores
son :

[(0, 'Ventas, tiene el articulo', 9),
(2, 'Ventas, tiene el articulo', 1),
(3, 'Ventas, tiene el articulo', 6),
(4, 'Ventas, tiene el articulo', 8),
(7, 'Ventas, tiene el articulo', 7),
(13, 'Ventas, tiene el articulo', 2),
(13, 'Ventas, tiene el articulo', 4),
(20, 'Ventas, tiene el articulo', 5),
(42, 'Ventas, tiene el articulo', 3)]

```

Ahora bien nos centramos en las dos categorías más vendidas y buscadas por los clientes, denotando que de igual manera hay artículos que no se venden estas categorías, por lo que ya al tenerlas identificadas, sería bueno darlas en oferta y que las que más se venden comprar más inventario de éstas.

```
[ (38, 'Calificaciones de 5 tiene el producto ', 54),
  (35, 'Calificaciones de 5 tiene el producto ', 3),
  (14, 'Calificaciones de 5 tiene el producto ', 5),
  (13, 'Calificaciones de 5 tiene el producto ', 57),
  (10, 'Calificaciones de 5 tiene el producto ', 4),
  (8, 'Calificaciones de 5 tiene el producto ', 2),
  (7, 'Calificaciones de 5 tiene el producto ', 4),
  (7, 'Calificaciones de 5 tiene el producto ', 4),
  (7, 'Calificaciones de 5 tiene el producto ', 1),
  (7, 'Calificaciones de 5 tiene el producto ', 7),
  (7, 'Calificaciones de 5 tiene el producto ', 4),
  (4, 'Calificaciones de 5 tiene el producto ', 44),
  (4, 'Calificaciones de 5 tiene el producto ', 8),
  (4, 'Calificaciones de 5 tiene el producto ', 2),
  (3, 'Calificaciones de 5 tiene el producto ', 49),
  (3, 'Calificaciones de 5 tiene el producto ', 11),
  (3, 'Calificaciones de 5 tiene el producto ', 6),
  (2, 'Calificaciones de 5 tiene el producto ', 85),
  (2, 'Calificaciones de 5 tiene el producto ', 52),
  (2, 'Calificaciones de 5 tiene el producto ', 51)]
```

Ahora bien tenemos los 20 mejores productos elegidos por los clientes, que cumplieron con la máxima calificación, en los cuales se pueden notar en el top 4 productos de la categoría de Procesadores y Discos Duros nuevamente, denotando de nuevo que son las categorías más fuertes de la empresa y denotando aún más que sus dos productos TOP son el artículo 3 y el 54.

```
[ (5, 'Opiniones debajo de la media tiene el producto', 31),
  (2, 'Opiniones debajo de la media tiene el producto', 29),
  (1, 'Opiniones debajo de la media tiene el producto', 89),
  (1, 'Opiniones debajo de la media tiene el producto', 54),
  (1, 'Opiniones debajo de la media tiene el producto', 48),
  (1, 'Opiniones debajo de la media tiene el producto', 47),
  (1, 'Opiniones debajo de la media tiene el producto', 46),
  (1, 'Opiniones debajo de la media tiene el producto', 45),
  (1, 'Opiniones debajo de la media tiene el producto', 17),
  (1, 'Opiniones debajo de la media tiene el producto', 4),
  (1, 'Opiniones debajo de la media tiene el producto', 3),
  (1, 'Opiniones debajo de la media tiene el producto', 2)],
[ (3, 'Devoluciones tiene el producto', 31),
  (1, 'Devoluciones tiene el producto', 54),
  (1, 'Devoluciones tiene el producto', 46),
  (1, 'Devoluciones tiene el producto', 45),
  (1, 'Devoluciones tiene el producto', 29),
  (1, 'Devoluciones tiene el producto', 17),
  (1, 'Devoluciones tiene el producto', 2)]
```

En estas dos tablas notamos los 12 productos con calificaciones más bajas y los 7 productos que fueron devueltos por los clientes, en las cuales podemos denotar que el peor producto que vende la tienda es el producto 31 ya que a sido devuelto en 3 ocasiones y 5 veces a sido mal evaluado, así que no sería bueno volverlo a ofertar en el futuro ya que el 50% de las veces que lo compran lo devuelven. Otro aspecto importante para notar en estas tablas es que a 1 persona no le gusto el producto estrella el artículo 54, un dato curioso, pero no se descarta que sea un outlier. También se denota que la mitad de los artículos devueltos son correspondientes a la categoría de tarjetas madre , posicionándola como la peor categoría de la tienda.


```
Las ventas toales de la tienda fueron: 760177
```

```
Las ventas totales del año 2020 fueron 759,695 Pesos
```

```
Las ventas totales del año 2002 fueron 300 Pesos
```

```
Las ventas totales del año 2019 fueron 182 Pesos
```

Por ultimo el paso 3 que pedía la información de ventas e ingresos totales, teniendo este resultante, de 760,177 en ventas, del cual no se puede decir mucho ya que no se sabe cuando esta invertido en stock, pero se puede sospechar que no la esta pasando bien la empresa con esta estrategia que sigue actualmente en sus ventas.

Pero eso no es todo, los ingresos totales son la resta entre las ventas totales y las devoluciones, por lo cual:

```
Los ingresos totales de la tienda (Ya con devoluciones)  
fueron: 737,916 Pesos
```

```
Las ventas totales del año 2019 fueron 182 Pesos
```

```
Hubo 53 Ventas en Enero
```

```
Hubo 41 Ventas en Febrero
```

```
Hubo 51 Ventas en Marzo
```

```
Hubo 75 Ventas en Abril
```

```
Hubo 36 Ventas en Mayo
```

```
Hubo 11 Ventas en Junio
```

```
Hubo 11 Ventas en Julio
```

```
Hubo 3 Ventas en Agosto
```

```
Hubo 1 Venta en Septiembre
```

```
Hubo 1 Venta en Noviembre de 2019
```

Por último denotamos que el mes con más ventas fue Abril con 75 ventas, y el mes con menos ventas fue Noviembre de 2019 que se puede considerar un mes atípico, ni tampoco podemos considerar Septiembre ya que apenas acaba de iniciar el mes, por lo que tomamos como el mes más bajo en ventas a Agosto, y denotamos que la tienda va en caída libre en las ventas ya que han bajado mucho durante el paso de los meses desde mayo a la fecha.

Conclusiones

Podemos sacar varias conclusiones con respecto a todo el análisis de datos realizado con nuestro programa hecho en Python, las cuales enunciare en algunos puntos a continuación:

- Hacer ofertas de los productos que no se han vendido durante el año para liberar el stock y no volver a comprar dichos artículos.
- Hacer publicidad sobre la tienda, ya que desde Mayo a la fecha ha tenido extremadamente muy pocas ventas en comparación a los meses pasados, esto puede ser porque no llaman la atención los productos o no se publicitan de buena manera.
- Nunca más vender el artículo 31 ya que la mitad de las veces que se vende lo devuelven lo que provocan molestia en el cliente y no genera ingresos a la empresa.
- Una alternativa podría ser centrar la tienda en sus dos categorías más fuertes que son la de los Procesadores y los discos duros, especializarse en estas categorías que van muy de la mano y dejar de lado las demás que solo producen mucho stock y pocas ventas es decir ganancias.
- Ser cuidadoso en la elección de nuevos productos y que sean en su mayoría de la categoría de Procesadores y Discos Duros, tratar de evitar comprar más stock de Pantallas, audífonos o de tarjetas madre.
- Comprar mas stock de los dos artículos más buscados y vendidos de la tienda, el artículo 3 y 54.

ANEXO

Tabla de Categorías

Rango de Productos	Categoría
1-9	Procesadores
10-28	Tarjetas de Video
29-46	Tarjetas Madre
47-59	Discos Duros
60-61	Memorias USB
62-73	Pantallas
74-83	Bocinas
84-96	Audífonos

Para Login

Nombre: Javier

Contraseña: ELLC

Nombre: Jorge

Contraseña: ELBLABO

<https://repl.it/@JorgeBlaborne/PROYECTO-01-VELASCOORTEGA-JORGEBLABORNE>

<https://PROYECTO-01-VELASCOORTEGA-JORGEBLABORNE.jorgeblaborne.repl.run>