



PROYECTO 1

INTRODUCCION A PYTHON.

Fernanda Rocha
Mancera

SUPERVISOR

Supervisor's full name

5 de septiembre de 2020

Faculty of Engineering and Science
Department of Engineering Sciences

Índice general

1. Introducción	2
2. Solución al problema	3
2.1. Los productos más vendidos y buscados.	3
2.2. Para los productos por reseña en el servicio	14
2.3. Total de ingresos promedio mensuales, total anual y meses con más ventas al año	17
3. Conclusiones	18
A. Códigos	20

Capítulo 1

Introducción

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 solicita un análisis de la rotación de productos identificando los siguientes elementos:

1. Productos más vendidos y productos más buscados 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.

Capítulo 2

Solución al problema

Para realizar el análisis que se solicitó, lo que se realizó fue la creación de un programa, el cual contiene los registros de las compras, búsquedas y productos manejados por la tienda. Este archivo trabaja con un *login* de usuario-administrador donde se muestra un reporte que especifica los siguientes requerimientos:

El código puede ser consultado en la siguiente liga:

https://github.com/fernanda-rocha/Curso_Emtech

2.1. Los productos más vendidos y buscados.

Por categoría se generó un listado de los productos con mayores ventas y uno con los productos con mayor búsquedas.

Tabla 2.1: Procesadores

Más vendidos
Procesador AMD Ryzen 5 2600, S-AM4, 3.40GHz, Six-Core, 16MB L3 Cache, con Disipador Wraith Stealth
Procesador Intel Core i3-9100F, S-1151, 3.60GHz, Quad-Core, 6MB Cache (9na. Generación - Coffee Lake)
Procesador AMD Ryzen 5 3600, S-AM4, 3.60GHz, 32MB L3 Cache, con Disipador Wraith Stealth
Procesador AMD Ryzen 3 3200G con Gráficos Radeon Vega 8, S-AM4, 3.60GHz, Quad-Core, 4MB L3, con Disipador Wraith Spire
Procesador Intel Core i7-9700K, S-1151, 3.60GHz, 8-Core, 12MB Smart Cache (9na. Generación Coffee Lake)
Más buscados
Procesador AMD Ryzen 5 2600, S-AM4, 3.40GHz, Six-Core, 16MB L3 Cache, con Disipador Wraith Stealth
Procesador AMD Ryzen 3 3200G con Gráficos Radeon Vega 8, S-AM4, 3.60GHz, Quad-Core, 4MB L3, con Disipador Wraith Spire
Procesador Intel Core i7-9700K, S-1151, 3.60GHz, 8-Core, 12MB Smart Cache (9na. Generación Coffee Lake)
Procesador Intel Core i3-9100F, S-1151, 3.60GHz, Quad-Core, 6MB Cache (9na. Generación - Coffee Lake)
Procesador AMD Ryzen 5 3600, S-AM4, 3.60GHz, 32MB L3 Cache, con Disipador Wraith Stealth

Tabla 2.2: Tarjetas de video

Más vendidas
Tarjeta de Video ASUS NVIDIA GeForce GTX 1660 SUPER EVO OC, 6GB 192-bit GDDR6, PCI Express x16 3.0
Tarjeta de Video Gigabyte NVIDIA GeForce GT 1030, 2GB 64-bit GDDR5, PCI Express x16 3.0
Tarjeta de Video ASUS AMD Radeon RX 570, 4GB 256-bit GDDR5, PCI Express 3.0
Tarjeta de Video MSI AMD Mech Radeon RX 5500 XT MECH Gaming OC, 8GB 128-bit GDDR6, PCI Express 4.0
Tarjeta de Video Sapphire AMD Pulse Radeon RX 5500 XT Gaming, 8GB 128-bit GDDR6, PCI Express 4.0
Más buscadas
Tarjeta de Video ASUS NVIDIA GeForce GTX 1660 SUPER EVO OC, 6GB 192-bit GDDR6, PCI Express x16 3.0
Tarjeta de Video MSI AMD Mech Radeon RX 5500 XT MECH Gaming OC, 8GB 128-bit GDDR6, PCI Express 4.0
Tarjeta de Video Gigabyte NVIDIA GeForce GT 1030, 2GB 64-bit GDDR5, PCI Express x16 3.0
Tarjeta de Video Sapphire AMD Pulse Radeon RX 5500 XT Gaming, 8GB 128-bit GDDR6, PCI Express 4.0
Tarjeta de Video ASUS AMD Radeon RX 570, 4GB 256-bit GDDR5, PCI Express 3.0

Tabla 2.3: Tarjetas madre

Más vendidas
Tarjeta Madre ASRock Micro ATX B450M Steel Legend, S-AM4, AMD B450, HDMI, 64GB DDR4 para AMD
Tarjeta Madre ASUS micro ATX TUF B450M-PLUS GAMING, S-AM4, AMD B450, HDMI, 64GB DDR4 para AMD
Tarjeta Madre AORUS micro ATX B450 AORUS M (rev. 1.0), S-AM4, AMD B450, HDMI, 64GB DDR4 para AMD
Tarjeta Madre MSI ATX B450 TOMAHAWK MAX, S-AM4, AMD B450, 64GB DDR4 para AMD
Tarjeta Madre ASUS ATX PRIME Z390-A, S-1151, Intel Z390, HDMI, 64GB DDR4 para Intel/xa0
Más buscadas
Tarjeta Madre ASUS micro ATX TUF B450M-PLUS GAMING, S-AM4, AMD B450, HDMI, 64GB DDR4 para AMD
Tarjeta Madre MSI ATX B450 TOMAHAWK MAX, S-AM4, AMD B450, 64GB DDR4 para AMD
Tarjeta Madre ASRock Micro ATX B450M Steel Legend, S-AM4, AMD B450, HDMI, 64GB DDR4 para AMD
Tarjeta Madre AORUS micro ATX B450 AORUS M (rev. 1.0), S-AM4, AMD B450, HDMI, 64GB DDR4 para AMD
Tarjeta Madre Gigabyte XL-ATX TRX40 Designare, S-sTRX4, AMD TRX40, 256GB DDR4 para AMD

Tabla 2.4: Discos duros

Más vendidos
SSD Kingston A400, 120GB, SATA III
SSD Adata Ultimate SU800, 256GB, SATA III, 2.5
SSD XPG SX8200 Pro, 256GB, PCI Express, M.2
SSD Kingston A2000 NVMe, 1TB, PCI Express 3.0, M2
Kit SSD Kingston KC600, 1TB, SATA III, 2.5, 7mm
Más buscados
SSD Kingston A400, 120GB, SATA III, 2.5
SSD Adata Ultimate SU800, 256GB, SATA III, 2.5", 7mm
SSD XPG SX8200 Pro, 256GB, PCI Express, M.2
SSD Kingston A2000 NVMe, 1TB, PCI Express 3.0, M2
SSD Kingston UV500, 480GB, SATA III, mSATA

Tabla 2.5: Memorias USB

Más vendidas
Kit Memoria RAM Corsair Dominator Platinum DDR4, 3200MHz, 16GB (2x 8GB), Non-ECC, CL16, XMP
Kit Memoria RAM Corsair Vengeance LPX DDR4, 2400MHz, 32GB, Non-ECC, CL16
Más buscadas
Kit Memoria RAM Corsair Dominator Platinum DDR4, 3200MHz, 16GB (2x 8GB), Non-ECC, CL16, XMP
Kit Memoria RAM Corsair Vengeance LPX DDR4, 2400MHz, 32GB, Non-ECC, CL16

Tabla 2.6: Pantallas

Más vendidas
TCL Smart TV LED 55S425 54.6, 4K Ultra HD, Widescreen, Negro
TV Monitor LED 24TL520S-PU 24, HD, Widescreen, HDMI, Negro
Más buscadas
TV Monitor LED 24TL520S-PU 24, HD, Widescreen, HDMI, Negro
TCL Smart TV LED 55S425 54.6, 4K Ultra HD, Widescreen, Negro
Seiki TV LED SC-39HS950N 38.5, HD, Widescreen, Negro
Samsung Smart TV LED UN55TU7000FXZX 55, 4K Ultra HD, Widescreen, Negro/Gris
Samsung Smart TV LED 43, Full HD, Widescreen, Negro

Para el caso de las pantallas, no se puede hacer un top de las 5 pantallas más vendidas, ya que de todos los modelos, solo 2 tuvieron ventas.

Tabla 2.7: Bocinas

Más vendidas
Logitech Bocinas para Computadora con Subwoofer G560, Bluetooth, Inalámbrico, 2.1, 120W RMS, USB, negro
Más buscadas
Logitech Bocinas para Computadora con Subwoofer G560, Bluetooth, Inalámbrico, 2.1, 120W RMS, USB, negro
Acteck Bocina con Subwoofer AXF-290, Bluetooth, Inalámbrico, 2.1, 18W RMS, 180W PMPO, USB, Negro
Ghia Bocina Portátil BX800, Bluetooth, Inalámbrico, 2.1 Canales, 31W, USB, Negro

Al igual que en el caso de las pantallas, para el caso de las bocinas, no se puede hacer un top de las 5 pantallas más vendidas ya que de todos los modelos, solo 1 tuvo ventas. Caso similar en las búsquedas, donde solo 3 modelos fueron revisados.

Tabla 2.8: Audífonos

Más vendidos
Logitech Audífonos Gamer G635 7.1, Alámbrico, 1.5 Metros, 3.5mm, Negro/Azul
Logitech Audífonos Gamer G332, Alámbrico, 2 Metros, 3.5mm, Negro/Rojo
Cougar Audífonos Gamer Phontum Essential, Alámbrico, 1.9 Metros, 3.5mm, Negro
HyperX Audífonos Gamer Cloud Flight para PC/PS4/PS4 Pro, Inalámbrico, USB, 3.5mm, Negro
Más buscados
Logitech Audífonos Gamer G635 7.1, Alámbrico, 1.5 Metros, 3.5mm, Negro/Azul
Logitech Audífonos Gamer G332, Alámbrico, 2 Metros, 3.5mm, Negro/Rojo
Cougar Audífonos Gamer Phontum Essential, Alámbrico, 1.9 Metros, 3.5mm, Negro
HyperX Audífonos Gamer Cloud Flight para PC/PS4/PS4 Pro, Inalámbrico, USB, 3.5mm, Negro
logear Audífonos Gamer GHG601, Alámbrico, 1.2 Metros, 3.5mm, Negro

De nueva cuenta, no se pudo hacer un top 5 con los más vendidos, por que solo 4 modelos tuvieron ventas.

Por categoría, se creó un listado con los productos con menores ventas y uno con los productos con menores búsquedas.

Tabla 2.9: Procesadores	
Menos vendidos	
Procesador Intel Core i3-8100, S-1151, 3.60GHz, Quad-Core, 6MB Smart Cache (8va. Generación - Coffee Lake)	
Procesador AMD Ryzen 3 3300X S-AM4, 3.80GHz, Quad-Core, 16MB L2 Cache	
Procesador Intel Core i9-9900K, S-1151, 3.60GHz, 8-Core, 16MB Smart Cache (9na. Generación Coffee Lake)	
Procesador Intel Core i5-9600K, S-1151, 3.70GHz, Six-Core, 9MB Smart Cache (9na. Generación - Coffee Lake)	
Procesador Intel Core i7-9700K, S-1151, 3.60GHz, 8-Core, 12MB Smart Cache (9na. Generación Coffee Lake)	
Menos buscados	
Procesador Intel Core i3-8100, S-1151, 3.60GHz, Quad-Core, 6MB Smart Cache (8va. Generación - Coffee Lake)	
Procesador AMD Ryzen 3 3300X S-AM4, 3.80GHz, Quad-Core, 16MB L2 Cache	
Procesador Intel Core i9-9900K, S-1151, 3.60GHz, 8-Core, 16MB Smart Cache (9na. Generación Coffee Lake)	
Procesador Intel Core i5-9600K, S-1151, 3.70GHz, Six-Core, 9MB Smart Cache (9na. Generación - Coffee Lake)	
Procesador AMD Ryzen 5 3600, S-AM4, 3.60GHz, 32MB L3 Cache, con Disipador Wraith Stealth	

Tabla 2.10: Tarjetas de video

Menos vendidas
Tarjeta de Video EVGA NVIDIA GeForce GT 710, 2GB 64-bit GDDR3, PCI Express 2.0
Tarjeta de Video EVGA NVIDIA GeForce GTX 1660 Ti SC Ultra Gaming, 6GB 192-bit GDDR6, PCI 3.0
Tarjeta de Video EVGA NVIDIA GeForce RTX 2060 SC ULTRA Gaming, 6GB 192-bit GDDR6, PCI Express 3.0
Tarjeta de Video Gigabyte NVIDIA GeForce GTX 1650 OC Low Profile, 4GB 128-bit GDDR5, PCI Express 3.0 x16
Tarjeta de Video Gigabyte NVIDIA GeForce RTX 2060 SUPER WINDFORCE OC, 8 GB 256 bit GDDR6, PCI Express x16 3.0
Menos buscadas
Tarjeta de Video EVGA NVIDIA GeForce GT 710, 2GB 64-bit GDDR3, PCI Express 2.0
Tarjeta de Video EVGA NVIDIA GeForce RTX 2060 SC ULTRA Gaming, 6GB 192-bit GDDR6, PCI Express 3.0
Tarjeta de Video Gigabyte NVIDIA GeForce GTX 1650 OC Low Profile, 4GB 128-bit GDDR5, PCI Express 3.0 x16
Tarjeta de Video Gigabyte NVIDIA GeForce RTX 2060 SUPER WINDFORCE OC, 8 GB 256 bit GDDR6, PCI Express x16 3.0
Tarjeta de Video MSI Radeon X1550, 128MB 64 bit GDDR2, PCI Express x16

Tabla 2.11: Tarjetas madre

Menos vendidos
Tarjeta Madre AORUS ATX Z390 ELITE, S-1151, Intel Z390, HDMI, 64GB DDR4 para Intel
Tarjeta Madre ASRock Z390 Phantom Gaming 4, S-1151, Intel Z390, HDMI, 64GB DDR4 para Intel/xa0
Tarjeta Madre ASUS ATX ROG STRIX B550-F GAMING WI-FI, S-AM4, AMD B550, HDMI, max. 128GB DDR4 para AMD
Tarjeta Madre Gigabyte micro ATX Z390 M GAMING, S-1151, Intel Z390, HDMI, 64GB DDR4 para Intel/xa0
Tarjeta Madre Gigabyte micro ATX Z490M GAMING X (rev. 1.0), Intel Z490, HDMI, 128GB DDR4 para Intel
Menos buscados
Tarjeta Madre AORUS ATX Z390 ELITE, S-1151, Intel Z390, HDMI, 64GB DDR4 para Intel
Tarjeta Madre ASRock Z390 Phantom Gaming 4, S-1151, Intel Z390, HDMI, 64GB DDR4 para Intel/xa0
Tarjeta Madre ASUS ATX PRIME Z390-A, S-1151, Intel Z390, HDMI, 64GB DDR4 para Intel/xa0
Tarjeta Madre ASUS ATX ROG STRIX B550-F GAMING WI-FI, S-AM4, AMD B550, HDMI, max. 128GB DDR4 para AMD
Tarjeta Madre Gigabyte micro ATX Z490M GAMING X (rev. 1.0), Intel Z490, HDMI, 128GB DDR4 para Intel

Tabla 2.12: Discos duros

Menos vendidos
SSD Addlink Technology S70, 512GB, PCI Express 3.0, M.2
SSD para Servidor Supermicro SSD-DM128-SMCMVN1, 128GB, SATA III, mSATA, 6Gbit/s
SSD para Servidor Lenovo Thinksystem S4500, 480GB, SATA III, 3.5", 7mm
SSD para Servidor Lenovo Thinksystem S4510, 480GB, SATA III, 2.5", 7mm
SSD Samsung 860 EVO, 1TB, SATA III, M.2
Menos buscados
SSD Addlink Technology S70, 512GB, PCI Express 3.0, M.2
SSD para Servidor Supermicro SSD-DM128-SMCMVN1, 128GB, SATA III, mSATA, 6Gbit/s
SSD para Servidor Lenovo Thinksystem S4510, 480GB, SATA III, 2.5", 7mm
SSD Samsung 860 EVO, 1TB, SATA III, M.2
SSD para Servidor Lenovo Thinksystem S4500, 480GB, SATA III, 3.5", 7mm

Tabla 2.13: Memorias USB

Menos vendidas
Kit Memoria RAM Corsair Vengeance LPX DDR4, 2400MHz, 32GB, Non-ECC, CL16
Kit Memoria RAM Corsair Dominator Platinum DDR4, 3200MHz, 16GB (2x 8GB), Non-ECC, CL16, XMP
Menos buscadas
Kit Memoria RAM Corsair Dominator Platinum DDR4, 3200MHz, 16GB (2x 8GB), Non-ECC, CL16, XMP
Kit Memoria RAM Corsair Vengeance LPX DDR4, 2400MHz, 32GB, Non-ECC, CL16

Tabla 2.14: Pantallas

Menos vendidas
Makena Smart TV LED 32S2 32", HD, Widescreen, Gris
Seiki TV LED SC-39HS950N 38.5, HD, Widescreen, Negro
Samsung TV LED LH43QMREBGCXGO 43, 4K Ultra HD, Widescreen, Negro
Samsung Smart TV LED UN70RU7100FXZX 70, 4K Ultra HD, Widescreen, Negro
Makena Smart TV LED 40S2 40", Full HD, Widescreen, Negro
Menos buscadas
Makena Smart TV LED 32S2 32", HD, Widescreen, Gris
Samsung TV LED LH43QMREBGCXGO 43, 4K Ultra HD, Widescreen, Negro
Samsung Smart TV LED UN70RU7100FXZX 70, 4K Ultra HD, Widescreen, Negro
Makena Smart TV LED 40S2 40", Full HD, Widescreen, Negro
Hisense Smart TV LED 40H5500F 39.5, Full HD, Widescreen, Negro

Tabla 2.15: Bocinas

Menos vendidas
Lenovo Barra de Sonido, Alámbrico, 2.5W, USB, Negro
Acteck Bocina con Subwoofer AXF-290, Bluetooth, Inalámbrico, 2.1, 18W RMS, 180W PMPO, USB, Negro
Verbatim Bocina Portátil Mini, Bluetooth, Inalámbrico, 3W RMS, USB, Blanco
Ghia Bocina Portátil BX300, Bluetooth, Inalámbrico, 40W RMS, USB, Rojo - Resistente al Agua
Naceb Bocina Portátil NA-0301, Bluetooth, Inalámbrico, USB 2.0, Rojo
Menos buscadas
Lenovo Barra de Sonido, Alámbrico, 2.5W, USB, Negro
Verbatim Bocina Portátil Mini, Bluetooth, Inalámbrico, 3W RMS, USB, Blanco
Ghia Bocina Portátil BX300, Bluetooth, Inalámbrico, 40W RMS, USB, Rojo - Resistente al Agua
Naceb Bocina Portátil NA-0301, Bluetooth, Inalámbrico, USB 2.0, Rojo
Ghia Bocina Portátil BX900, Bluetooth, Inalámbrico, 2.1 Canales, 34W, USB, Negro - Resistente al Agua

Tabla 2.16: Audífonos

Menos vendidos
ASUS Audífonos Gamer ROG Theta 7.1, Alámbrico, USB C, Negro
Acer Audífonos Gamer Galea 300, Alámbrico, 3.5mm, Negro
Audífonos Gamer Balam Rush Orphix RGB 7.1, Alámbrico, USB, Negro
Energy Sistem Audífonos con Micrófono Headphones 1, Bluetooth, Inalámbrico, Negro/Grafito
Genius GHP-400S Audífonos, Alámbrico, 1.5 Metros, Rosa
Menos buscados
ASUS Audífonos Gamer ROG Theta 7.1, Alámbrico, USB C, Negro
Acer Audífonos Gamer Galea 300, Alámbrico, 3.5mm, Negro
Audífonos Gamer Balam Rush Orphix RGB 7.1, Alámbrico, USB, Negro
Energy Sistem Audífonos con Micrófono Headphones 1, Bluetooth, Inalámbrico, Negro/Grafito
Getttech Audífonos con Micrófono Sonority, Alámbrico, 1.2 Metros, 3.5mm, Negro/Rosa

2.2. Para los productos por reseña en el servicio

Se mostraron dos listados de 20 productos cada uno, un listado para productos con las mejores reseñas promedio y otro para las peores, considerando los productos con devolución.

Se puede observar que, por separado, son procesadores los artículos con la mejor reseña, pero en general son las pantallas las que tienen mejor reseña

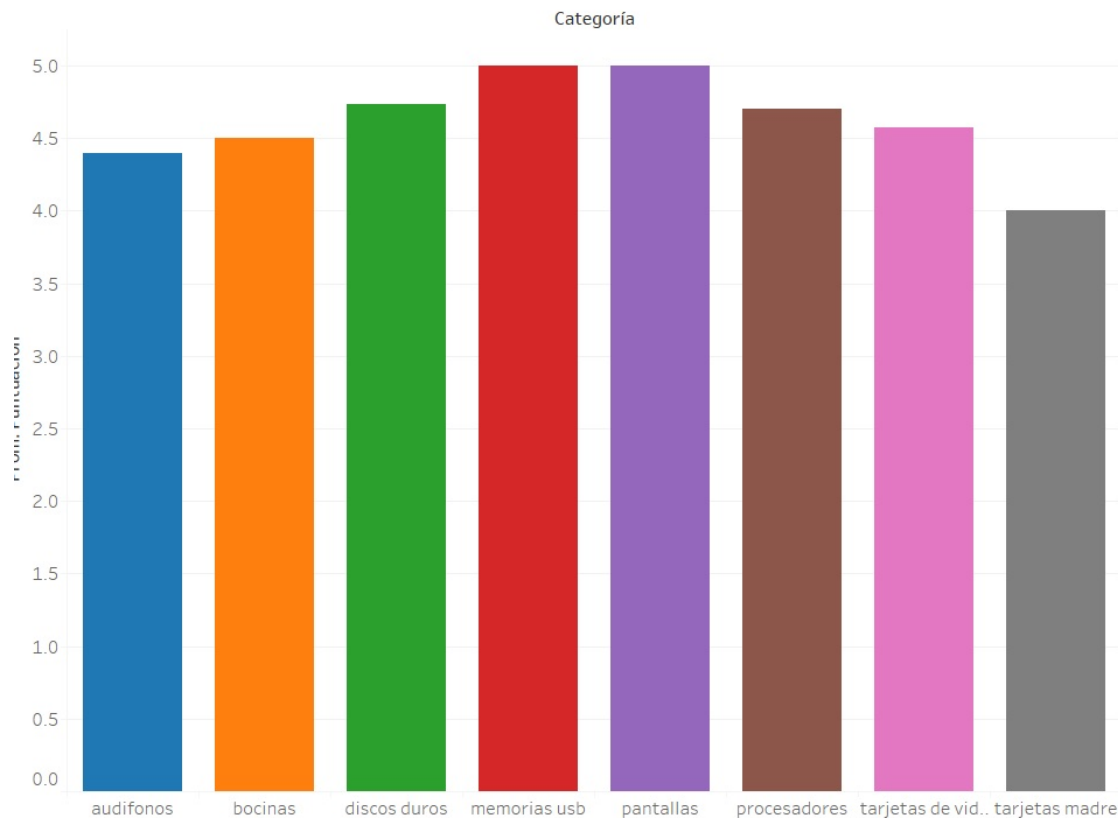


Figura 2.1: Categorías con las mejores reseñas en general.

Tabla 2.17: Top 20 productos con mejores reseñas

Producto	Reseña Promedio
Procesador AMD Ryzen 3 3300X S-AM4, 3.80GHz, Quad-Core, 16MB L2 Cache	5
Procesador Intel Core i9-9900K, S-1151, 3.60GHz, 8-Core, 16MB Smart Cache (9na. Generación Coffee Lake	5
Procesador Intel Core i7-9700K, S-1151, 3.60GHz, 8-Core, 12MB Smart Cache (9na. Generación Coffee Lake	5
Procesador Intel Core i5-9600K, S-1151, 3.70GHz, Six-Core, 9MB Smart Cache (9na. Generación - Coffee Lake	5
Tarjeta de Video ASUS AMD Radeon RX 570, 4GB 256-bit GDDR5, PCI Express 3.0	5
Tarjeta de Video MSI AMD Mech Radeon RX 5500 XT MECH Gaming OC, 8GB 128-bit GDDR6, PCI Express 4.0	5
Tarjeta de Video MSI NVIDIA GeForce GTX 1050 Ti OC, 4GB 128-bit GDDR5, PCI Express x16 3.0	5
Tarjeta de Video Sapphire AMD Pulse Radeon RX 5500 XT Gaming, 8GB 128-bit GDDR6, PCI Express 4.0	5
Tarjeta de Video Zotac NVIDIA GeForce GTX 1660 Ti, 6GB 192-bit GDDR6, PCI Express x16 3.0	5
Tarjeta Madre Gigabyte XL-ATX TRX40 Designare, S-sTRX4, AMD TRX40, 256GB DDR4 para AMD	5
Kit SSD Kingston KC600, 1TB, SATA III, 2.5, 7mm	5
SSD Crucial MX500, 1TB, SATA III, M.2	5
SSD Western Digital WD Blue 3D NAND, 2TB, M.2	5
Kit Memoria RAM Corsair Dominator Platinum DDR4, 3200MHz, 16GB (2x 8GB), Non-ECC, CL16, XMP	5
TCL Smart TV LED 55S425 54.6, 4K Ultra HD, Widescreen, Negro	5
TV Monitor LED 24TL520S-PU 24, HD, Widescreen, HDMI, Negro	5
Logitech Audífonos Gamer G332, Alámbrico, 2 Metros, 3.5mm, Negro/Rojo	5
Logitech Audífonos Gamer G635 7.1, Alámbrico, 1.5 Metros, 3.5mm, Negro/Azul	5
SSD Adata Ultimate SU800, 256GB, SATA III, 2.5", 7mm	4.86
Procesador AMD Ryzen 5 2600, S-AM4, 3.40GHz, Six-Core, 16MB L3 Cache, con Disipador Wraith Stealth	4.8

Tabla 2.18: Top 20 productos con peores reseñas

Producto	Reseña Promedio
Tarjeta de Video Gigabyte AMD Radeon R7 370 OC, 2GB 256-bit GDDR5, PCI Express 3.0	1
Tarjeta Madre ASRock ATX H110 Pro BTC+, S-1151, Intel H110, 32GB DDR4, para Intel	1
Tarjeta Madre AORUS micro ATX B450 AORUS M (rev. 1.0), S-AM4, AMD B450, HDMI, 64GB DDR4 para AMD	1.83
Tarjeta Madre Gigabyte micro ATX GA-H110M-DS2, S-1151, Intel H110, 32GB DDR4 para Intel	2
Cougar Audífonos Gamer Phontum Essential, Alámbrico, 1.9 Metros, 3.5mm, Negro.	3
MSI GeForce 210, 1GB GDDR3, DVI, VGA, HDCP, PCI Express 2.0	4
Tarjeta de Video Asus NVIDIA GeForce GTX 1050 Ti Phoenix, 4GB 128-bit GDDR5, PCI Express 3.0	4
HyperX Audífonos Gamer Cloud Flight para PC/PS4/PS4 Pro, Inalámbrico, USB, 3.5mm, Negro	4
Tarjeta Madre ASUS micro ATX TUF B450M-PLUS GAMING, S-AM4, AMD B450, HDMI, 64GB DDR4 para AMD	4.14
Procesador AMD Ryzen 5 3600, S-AM4, 3.60GHz, 32MB L3 Cache, con Disipador Wraith Stealth	4.23
Tarjeta de Video Gigabyte NVIDIA GeForce GT 1030, 2GB 64-bit GDDR5, PCI Express x16 3.0	4.4
Procesador AMD Ryzen 3 3200G con Gráficos Radeon Vega 8, S-AM4, 3.60GHz, Quad-Core, 4MB L3, con Disipador Wraith Spire	4.46
Tarjeta Madre ASUS ATX PRIME Z390-A, S-1151, Intel Z390, HDMI, 64GB DDR4 para Intel xa0	4.5
Logitech Bocinas para Computadora con Subwoofer G560, Bluetooth, Inalámbrico, 2.1, 120W RMS, USB, negro	4.5
SSD XPG SX8200 Pro, 256GB, PCI Express, M.2	4.54
Tarjeta Madre ASRock Micro ATX B450M Steel Legend, S-AM4, AMD B450, HDMI, 64GB DDR4 para AMD	4.55
Tarjeta Madre MSI ATX B450 TOMAHAWK MAX, S-AM4, AMD B450, 64GB DDR4 para AMD	4.66
SSD Kingston A2000 NVMe, 1TB, PCI Express 3.0, M2	4.66
SSD Kingston UV500, 480GB, SATA III, mSATA	4.66
Procesador Intel Core i3-9100F, S-1151, 3.60GHz, Quad-Core, 6MB Cache (9na. Generación - Coffee Lake	4.7

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

Tabla 2.19: Ingresos Totales	
Meses	Ganancia mensual
Enero	117738
Febrero	107270
Marzo	162931
Abril	191066
Mayo	91936
Junio	36949
Julio	26949
Agosto	3077
Septiembre	-
Octubre	-
Noviembre	-
Diciembre	-
Total	737916
Ganancia Promedio mensual	92339.5

Capítulo 3

Conclusiones

Analizando las ventas y búsquedas por categoría se puede ver que las memorias USB no son buscadas y apenas son compradas, por lo cual se recomienda sacarlas del catálogo, pues no representan ninguna ganancia.

Algunas pantallas son productos muy buscados, pero solo se vendieron dos, probablemente eso signifique que los clientes las buscan pero les parece demasiado elevado el precio y por eso no las compran. Por otro lado, hay siete modelos que nunca han sido buscadas, así que se debería reducir el stock, principalmente el modelo "Makena Smart TV LED 40S2 40", Full HD, Widescreen, Negro", porque nunca ha sido comprado y hay muchas pantallas en el stock.

Lo mismo pasa con los audífonos, pues hay modelos que a pesar de tener muchas búsquedas, han sido comprados solo una vez. Algunos modelos deberían ser retirados.

Considerando las reseñas, las tarjetas madre son el producto con menor promedio de reseñas, los modelos "Tarjeta Madre ASRock ATX H110 Pro BTC+, S-1151, Intel H110, 32GB DDR4, para Intel" "Tarjeta Madre Gigabyte micro ATX GA-H110M-DS2, S-1151, Intel H110, 32GB DDR4 para Intel" deberían ser retirados del catálogo porque se están acumulando en el inventario y la calidad no es buena, por eso las han devuelto

La conclusión final es que los productos menos vendidos también son poco buscados como si los clientes desconocieran la amplia gama de productos vendidos en la tienda. Debería invertirse más en publicidad para que la gente conozca todos los artículos. Además, dichos productos deberían estar un poco más a la vista en la tienda física.

Otra idea sería ofrecer algún paquete atractivo para los clientes, para sacar algunos de los productos que menos se venden. Algo como que en la compra de una pantalla se

te regalan unos audífonos o una bocina. Esto le daría buena publicidad a la tienda, se reduciría el stock estancado y los clientes conocerían más modelos disponibles en la tienda.

Apéndice A

Códigos

```
#!/usr/bin/env python
# coding: utf-8

# # Instrucciones
#
# ### El administrador podrá acceder a las siguientes funciones:
# – Ver los productos más vendidos y más buscados por categoría
# – Ver productos menos vendidos y menos buscados
# – Ver productos por reseña
# – Total de ingresos y ventas
# – Añadir nuevos usuarios
#
# El usuario del administrador es Admin y la contraseña es 0000
#
# ### Los usuarios normales solo podrán acceder a las siguientes funciones:
# – Ver los productos más vendidos y más buscados por categoría
# – Ver productos menos vendidos y menos buscados
#
# Los siguientes usuarios ya están registrados
# – Usuario: Fer, contraseña: 1234
# – Usuario: Miranda2, contraseña: asdfg
# – Usuario: Sal01, contraseña: chocobo
#
# In[43]:
```

```
##Se carga el archivo lifestore_file.py
```

```
import sys
```

```
sys.path.append(r"\Users\Fer\Desktop\Curso_Emtech\lifestore_file")
```

```
import lifestore_file
```

```
# In[44]:
```

```
##Se importan las listas lifestore_products , lifestore_sales , lifestore_searches
```

```
from lifestore_file import lifestore_products , lifestore_sales , lifestore_searches
```

```
# In[45]:
```

```
##Contador de las ventas totales
```

```
contador_ventas = 0
```

```
total_ventas =[]
```

```
for productoV in lifestore_products:
```

```
    for venta in lifestore_sales:
```

```
        if productoV[0] == venta[1]:
```

```
            contador_ventas += 1
```

```
tot_ventas = [productoV[1],contador_ventas , productoV[3]]
```

```
total_ventas.append(tot_ventas)
```

```
contador_ventas = 0
```

```
# In[84]:
```

```

'''
Contador de las ventas multiplicadas por los precios
para sacar las ganancias anuales y la ganancia promedio mensual
'''

contador_precios = 0
total_precios = []
tot_precios = 0
suma_precios = 0

for productoP in lifestore_products:
    for ventaP in lifestore_sales:
        if productoP[0] == ventaP[1] and ventaP[4] != 0:
            contador_precios += 1
        if contador_precios != 0:
            tot_precios = contador_precios * productoP[2]
            suma_precios = tot_precios + suma_precios
            total_precios.append([contador_precios, productoP[2], tot_precios, suma_precios])
        contador_precios = 0
        tot_precios = 0
suma_precios = 0
promedio_mensual = total_precios[-1][3] / 9

```

In[46]:

```

def mayorVenta(categoria_mV):
    '''
    Funci n que ordena las ventas de mayor a menor por categor a de producto
    '''
    totalMayor = []
    for categoria in total_ventas:
        if categoria[2] == categoria_mV:
            totalMayor.append(categoria)

```

```

ordenar_mV = []
while totalMayor:
    mayor_venta = totalMayor[0][1]
    lista_mayorV = totalMayor[0]
    for mayorV in totalMayor:
        if mayorV[1] > mayor_venta:
            mayor_venta = mayorV[1]
            lista_mayorV = mayorV
    ordenar_mV.append(lista_mayorV)
    totalMayor.remove(lista_mayorV)
print( "\nEstos son los", categoria_mV, "ordenados de mayor a menor venta\n")
for or_mV in ordenar_mV:
    print( '\n', or_mV, '\n')

```

In[47]:

```

def menorVenta(categoria_meV):
    '''
    Funci n que ordena las ventas de menor a mayor por categor a de producto
    '''
    totalMenor = []
    for categoriaMe in total_ventas:
        if categoriaMe[2] == categoria_meV:
            totalMenor.append(categoriaMe)

ordenar_meV = []
while totalMenor:
    menor_venta = totalMenor[0][1]
    lista_menorV = totalMenor[0]
    for menorV in totalMenor:
        if menorV[1] < menor_venta:
            menor_venta = menorV[1]
            lista_menorV = menorV
    ordenar_meV.append(lista_menorV)

```



```

        totalMenor.remove(lista_menorV)
print ("\nEstos son los", categoria_meV, "ordenados de menor a mayor venta\n")
for or_meV in ordenar_meV:
    print ('\\n', or_meV, '\\n')

# In[48]:

##Contador de las b s quedas totales
contador_busquedas = 0
total_busquedas = []

for productoS in lifestore_products:
    for busqueda in lifestore_searches:
        if productoS[0] == busqueda[1]:
            contador_busquedas += 1

    tot_busquedas = [productoS[1], contador_busquedas, productoS[3]]
    total_busquedas.append(tot_busquedas)
    contador_busquedas = 0

# In[49]:

def mayorBusqueda(categoria_mB):
    '''
    Funci n que ordena las b s quedas de mayor a menor por categor a de produ
    '''

    totalMayorB = []
    for categoriaB in total_busquedas:
        if categoriaB[2] == categoria_mB:
            totalMayorB.append(categoriaB)

    ordenar_mB = []

```

```

while totalMayorB:
    mayor_busqueda = totalMayorB[0][1]
    lista_mayorB = totalMayorB[0]
    for mayorB in totalMayorB:
        if mayorB[1] > mayor_busqueda:
            mayor_busqueda = mayorB[1]
            lista_mayorB = mayorB
    ordenar_mB.append(lista_mayorB)
    totalMayorB.remove(lista_mayorB)
print ("\nEstos son los", categoria_mB, "ordenados de mayor a menor b squeda\
for or_mB in ordenar_mB:
    print ('\\n', or_mB, '\\n')

```

In[50]:

```

def menorBusqueda(categoria_meB):
    '''
    Funci n que ordena las b sqedas de menor a mayor por categor a de produ
    '''
    totalMenorB = []
    for categoriaMeB in total_busquedas:
        if categoriaMeB[2] == categoria_meB:
            totalMenorB.append(categoriaMeB)

    ordenar_meB = []
    while totalMenorB:
        menor_busqueda = totalMenorB[0][1]
        lista_menorB = totalMenorB[0]
        for menorB in totalMenorB:
            if menorB[1] < menor_busqueda:
                menor_busqueda = menorB[1]
                lista_menorB = menorB
        ordenar_meB.append(lista_menorB)
        totalMenorB.remove(lista_menorB)

```

```

print ("\nEstos son los", categoria_meB, "ordenados de menor a mayor b s queda
for or_meB in ordenar_meB:
    print ( '\n', or_meB, '\n')

# In[51]:

'''
Contador de las reseñas de los productos vendidos
Adem s saca el promedio de reseña de cada producto
'''

cont_resenas =0
cont_promedio=0
resenas =[]
ceros =[]

for productoR in lifestore_products:
    for ventasR in lifestore_sales:
        if productoR[0] == ventasR[1]:
            cont_promedio += 1
            cont_resenas = ventasR[2] + cont_resenas
        if cont_promedio == 0 or cont_resenas == 0:
            ceros.append([productoR[1], cont_promedio, cont_resenas])
        else:
            promedios = cont_resenas/cont_promedio
            tot_resenas = [productoR[1], promedios]
            resenas.append(tot_resenas)

cont_promedio = 0
cont_resenas = 0

# In[52]:

```

```
##Copias de la lista de reseas
```

```
resenas_copia = resenas.copy()
```

```
resenas_copia2 = resenas.copy()
```

```
# In[53]:
```

```
'''
```

```
Ordenar el promedio de reseas de la m s alta a la m s baja
```

```
'''
```

```
ordenar_resenaAlta = []
```

```
while resenas_copia:
```

```
    mejor_resena = resenas_copia[0][1]
```

```
    lista_mejorR = resenas_copia[0]
```

```
    for mejorR in resenas_copia:
```

```
        if mejorR[1] > mejor_resena:
```

```
            mejor_resena = mejorR[1]
```

```
            lista_mejorR = mejorR
```

```
    ordenar_resenaAlta.append(lista_mejorR)
```

```
    resenas_copia.remove(lista_mejorR)
```

```
# In[54]:
```

```
'''
```

```
Ordenar el promedio de reseas de la m s baja a la m s alta
```

```
'''
```

```
ordenar_resenaBaja = []
```

```
while resenas_copia2:
```

```
    peor_resena = resenas_copia2[0][1]
```

```
    lista_peorR = resenas_copia2[0]
```

```
    for peorR in resenas_copia2:
```

```
        if peorR[1] < peor_resena:
```

```

        peor_resena = peorR[1]
        lista_peorR = peorR
    ordenar_resenaBaja.append(lista_peorR)
    resenas_copia2.remove(lista_peorR)

```

In[60]:

```

#Filtro de ventas por mes
enero = []
febrero = []
marzo = []
abril = []
mayo = []
junio = []
julio = []
agosto = []
septiembre = []
octubre = []
noviembre = []
diciembre = []
for mes in lifestore_sales:
    if mes[3][4] == '1' and mes[3][3] == '0':
        enero.append([mes[1], mes[3]])
    elif mes[3][4] == '2' and mes[3][3] == '0':
        febrero.append([mes[1], mes[3]])
    elif mes[3][4] == '3' and mes[3][3] == '0':
        marzo.append([mes[1], mes[3]])
    elif mes[3][4] == '4' and mes[3][3] == '0':
        abril.append([mes[1], mes[3]])
    elif mes[3][4] == '5' and mes[3][3] == '0':
        mayo.append([mes[1], mes[3]])
    elif mes[3][4] == '6' and mes[3][3] == '0':
        junio.append([mes[1], mes[3]])
    elif mes[3][4] == '7' and mes[3][3] == '0':

```

```

    julio.append([mes[1], mes[3]])
elif mes[3][4] == '8' and mes[3][3] == '0':
    agosto.append([mes[1], mes[3]])
elif mes[3][4] == '9' and mes[3][3] == '0':
    septiembre.append([mes[1], mes[3]])
elif mes[3][4] == '0' and mes[3][3] == '1':
    octubre.append([mes[1], mes[3]])
elif mes[3][4] == '1' and mes[3][3] == '1':
    noviembre.append([mes[1], mes[3]])
elif mes[3][4] == '2' and mes[3][3] == '1':
    diciembre.append([mes[1], mes[3]])

```

In[56]:

```

def totalMes(meses):
    '''
    Funci n que saca las ganancias por cada mes
    '''
    contador_ingresos = 0
    dinero = 0
    suma_dinero=0
    total_ingresos =[]
    for productoI in lifestore_products:
        for ventaI in meses:
            if productoI[0] == ventaI[0]:
                contador_ingresos += 1
                dinero = contador_ingresos*productoI[2]
            if contador_ingresos != 0:
                suma_dinero = dinero + suma_dinero
                total_ingresos.append([contador_ingresos , productoI[2] , dinero , suma_dinero])

    contador_ingresos = 0
    dinero=0
    suma_dinero=0

```

```
print ("El total de ganancias fue de", total_ingresos[-1][3], "pesos")
```

```
# In[57]:
```

```
##Cuentas de administrador y de usuarios
```

```
usuarios_normales = [ ["Fer", "1234"], ["Miranda2", "asdfg"], ["Sal01", "chocobo"] ]
```

```
admin = [ ["Admin", "0000"] ]
```

```
# In[85]:
```

```
##Bloque principal del programa
```

```
print ("Bienvenido al sistema de datos de LifeStore")
```

```
opciones = 1
```

```
opciones_admin = 1
```

```
opciones_usuario = 1
```

```
opciones_opciones = 1
```

```
while opciones:
```

```
    print ("\t1 Ingresar como administrador")
```

```
    print ("\t2 Ingresar como usuario")
```

```
    print ("\t3 Salir")
```

```
    opcion_menu = input ("\nSeleccione una opcion de login: ")
```

```
    if opcion_menu == '1':
```

```
        user_admin = input ("\nIngresa tu nombre de usuario: ")
```

```
        password_admin = input ("\nIngresa tu contraseña: ")
```

```
        for usuario_admin in admin:
```

```
            if usuario_admin[0] == user_admin and usuario_admin[1] == password_a
```

```
                print ("\nBienvenido administrador")
```

```
                opciones = 0
```

```

while opciones_admin:
    print("\t1_Ver los productos m s vendidos y m s buscados p
    print("\t2_Ver productos menos vendidos y menos buscados")
    print("\t3_Ver productos por rese a")
    print("\t4_Total de ingresos y ventas")
    print("\t5_Aadir nuevo usuario")
    print("\t6_Salir")
    menu_admin = input("\nSeleccione lo que desea hacer: ")

    if menu_admin == '1':
        while opciones_opciones:
            print("\t1_Procesadores")
            print("\t2_Tarjetas de video")
            print("\t3_Tarjetas madre")
            print("\t4_Discos duros")
            print("\t5_Memorias usb")
            print("\t6_Pantallas")
            print("\t7_Bocinas")
            print("\t8_Aud fonos")
            accion = input("\nElija la categoria que le gustar
            if accion == '1':
                mayorVenta('procesadores')
                mayorBusqueda('procesadores')
                opciones_opciones = 0
            elif accion == '2':
                mayorVenta('tarjetas de video')
                mayorBusqueda('tarjetas de video')
                opciones_opciones = 0
            elif accion == '3':
                mayorVenta('tarjetas madre')
                mayorBusqueda('tarjetas madre')
                opciones_opciones = 0
            elif accion == '4':
                mayorVenta('discos duros')
                mayorBusqueda('discos duros')
                opciones_opciones = 0

```



```

elif accion == '5':
    mayorVenta('memorias_usb')
    mayorBusqueda('memorias_usb')
    opciones_opciones = 0
elif accion == '6':
    mayorVenta('pantallas')
    mayorBusqueda('pantallas')
    opciones_opciones = 0
elif accion == '7':
    mayorVenta('bocinas')
    mayorBusqueda('bocinas')
    opciones_opciones = 0
elif accion == '8':
    mayorVenta('audifonos')
    mayorBusqueda('audifonos')
    opciones_opciones = 0
else:
    print("\\nEl_producto_est_mal_escrito\\n")
opciones_admin = 0

elif menu_admin == '2':
    while opciones_opciones:
        print("\\nElija_la_categoria_que_le_gustar_a_ver:")
        print("\\t1_Procesadores")
        print("\\t2_Tarjetas_de_video")
        print("\\t3_Tarjetas_madre")
        print("\\t4_Discos_duros")
        print("\\t5_Memorias_usb")
        print("\\t6_Pantallas")
        print("\\t7_Bocinas")
        print("\\t8_Aud_fonos")
        accion = input("\\nElija_la_categoria_que_le_gustar_a_ver:")
        if accion == '1':
            menorVenta('procesadores')
            menorBusqueda('procesadores')
            opciones_opciones = 0

```

```

elif accion == '2':
    menorVenta('tarjetas_de_video')
    menorBusqueda('tarjetas_de_video')
    opciones_opciones = 0
elif accion == '3':
    menorVenta('tarjetas_madre')
    menorBusqueda('tarjetas_madre')
    opciones_opciones = 0
elif accion == '4':
    menorVenta('discos_duros')
    menorBusqueda('discos_duros')
    opciones_opciones = 0
elif accion == '5':
    menorVenta('memorias_usb')
    menorBusqueda('memorias_usb')
    opciones_opciones = 0
elif accion == '6':
    menorVenta('pantallas')
    menorBusqueda('pantallas')
    opciones_opciones = 0
elif accion == '7':
    menorVenta('bocinas')
    menorBusqueda('bocinas')
    opciones_opciones = 0
elif accion == '8':
    menorVenta('audifonos')
    menorBusqueda('audifonos')
    opciones_opciones = 0
else:
    print("\\nEl_producto_esta_mal_escrito\\n")
    opciones_admin = 0

elif menu_admin == '3':

    print("\\nEste_es_el_top_20_de_productos_con_mejor_promedio")
    for i in range(21):

```

```

        print( ordenar_resenaAlta[i], '\n')
print( "\nEste es el top 20 de productos con peor promedio")
for index in range(21):
    print( ordenar_resenaBaja[index], '\n')
opciones_admin = 0

elif menu_admin == '4':
    while opciones_opciones:
        print( "\nSe pueden ver las ganancias de forma mensual")
        print( "\t1_Enero")
        print( "\t2_Febrero")
        print( "\t3_Marzo")
        print( "\t4_Abril")
        print( "\t5_Mayo")
        print( "\t6_Junio")
        print( "\t7_Julio")
        print( "\t8_Agosto")
        print( "\t9_Septiembre")
        print( "\t10_Octubre")
        print( "\t11_Noviembre")
        print( "\t12_Diciembre")
        print( "\t13_Anual")
        accion = input( "\nElija las ganancias que quiere ver")
        if accion == '1':
            totalMes(enero)
            opciones_opciones = 0
        elif accion == '2':
            totalMes(febrero)
            opciones_opciones = 0
        elif accion == '3':
            totalMes(marzo)
            opciones_opciones = 0
        elif accion == '4':
            totalMes(abril)
            opciones_opciones = 0
        elif accion == '5':

```

```

        totalMes(mayo)
        opciones_opciones = 0
    elif accion == '6':
        totalMes(junio)
        opciones_opciones = 0
    elif accion == '7':
        totalMes(julio)
        opciones_opciones = 0
    elif accion == '8':
        totalMes(agosto)
        opciones_opciones = 0
    elif accion == '9':
        totalMes(septiembre)
        opciones_opciones = 0
    elif accion == '10':
        print("\nA n no hay ganancias este mes\n")
        opciones_opciones = 0
    elif accion == '11':
        print("\nEn este mes solo hubo una venta y desp")
        opciones_opciones = 0
    elif accion == '12':
        print("\nA n no hay ganancias este mes\n")
        opciones_opciones = 0
    elif accion == '13':
        print("\nLa ganancia anual fue de", total_preci
        print("\nEl promedio mensual de ganancias es de
        opciones_opciones = 0
    else:
        print("\nEsta opci n no est  disponible\n")
    opciones_admin = 0

elif menu_admin == '5':
    nuevo_usuario = input("\nIngresa un nombre de usuario:")
    nuevo_password = input("\nIngresa una contrase a:")

    usuarios_normales.append([nuevo_usuario, nuevo_password])

```

```

        elif menu_admin == '6':
            opciones_admin = 0

        else:
            print("\nError. _Intente _de _nuevo")

    else:
        print("\nDatos _incorrectos")

elif opcion_menu == '2':
    user = input("\nIngresa _tu _nombre _de _usuario: _")
    password = input("\nIngresa _tu _contrase a: _")
    for usuario in usuarios_normales:
        if usuario[0] == user and usuario[1] == password:
            opciones = 0
            while opciones_usuario:
                print("\t1 _Ver _productos _m s _vendidos")
                print("\t2 _Ver _productos _menos _vendidos")
                print("\t3 _Salir")

            menu_usuario = input("\nSeleccione _lo _que _desea _hacer: _")

            if menu_usuario == '1':
                while opciones_opciones:
                    print("\t1 _Procesadores")
                    print("\t2 _Tarjetas _de _video")
                    print("\t3 _Tarjetas _madre")
                    print("\t4 _Discos _duros")
                    print("\t5 _Memorias _usb")
                    print("\t6 _Pantallas")
                    print("\t7 _Bocinas")
                    print("\t8 _Aud fonos")
                    accion = input("\nElija _la _categor a _que _le _gustar")
                    if accion == '1':

```

```

        mayorVenta('procesadores')
        mayorBusqueda('procesadores')
        opciones_opciones = 0
    elif accion == '2':
        mayorVenta('tarjetas_de_video')
        mayorBusqueda('tarjetas_de_video')
        opciones_opciones = 0
    elif accion == '3':
        mayorVenta('tarjetas_madre')
        mayorBusqueda('tarjetas_madre')
        opciones_opciones = 0
    elif accion == '4':
        mayorVenta('discos_duros')
        mayorBusqueda('discos_duros')
        opciones_opciones = 0
    elif accion == '5':
        mayorVenta('memorias_usb')
        mayorBusqueda('memorias_usb')
        opciones_opciones = 0
    elif accion == '6':
        mayorVenta('pantallas')
        mayorBusqueda('pantallas')
        opciones_opciones = 0
    elif accion == '7':
        mayorVenta('bocinas')
        mayorBusqueda('bocinas')
        opciones_opciones = 0
    elif accion == '8':
        mayorVenta('audifonos')
        mayorBusqueda('audifonos')
        opciones_opciones = 0
    else:
        print("\nEl producto no existe\n")
    opciones_usuario = 0
elif menu_usuario == '2':
    while opciones_opciones:

```

```

print("\nElija la categor a que le gustar a ver:")
print("\t1 Procesadores")
print("\t2 Tarjetas de video")
print("\t3 Tarjetas madre")
print("\t4 Discos duros")
print("\t5 Memorias usb")
print("\t6 Pantallas")
print("\t7 Bocinas")
print("\t8 Aud fonos")
accion = input("\nElija la categor a que le gustar
if accion == '1':
    menorVenta('procesadores')
    menorBusqueda('procesadores')
    opciones_opciones = 0
elif accion == '2':
    menorVenta('tarjetas de video')
    menorBusqueda('tarjetas de video')
    opciones_opciones = 0
elif accion == '3':
    menorVenta('tarjetas madre')
    menorBusqueda('tarjetas madre')
    opciones_opciones = 0
elif accion == '4':
    menorVenta('discos duros')
    menorBusqueda('discos duros')
    opciones_opciones = 0
elif accion == '5':
    menorVenta('memorias usb')
    menorBusqueda('memorias usb')
    opciones_opciones = 0
elif accion == '6':
    menorVenta('pantallas')
    menorBusqueda('pantallas')
    opciones_opciones = 0
elif accion == '7':
    menorVenta('bocinas')

```

```

        menorBusqueda( 'bocinas' )
        opciones_opciones = 0
    elif accion == '8':
        menorVenta( 'audifonos' )
        menorBusqueda( 'audifonos' )
        opciones_opciones = 0
    else:
        print( "\nEl producto no existe\n")
        opciones_usuario = 0

elif menu_usuario == '3':
    opciones_usuario = 0

elif opciones == 1:
    print( "\nDatos incorrectos. Intente otra vez o comun quese con_" )
    continue

elif opcion_menu == '3':
    opciones = 0

else:
    print( "\nError. Intente de nuevo")

# In[ ]:

```