

## Practica SC-115 - Práctica III

### Instrucciones generales.

La pulpería de su comunidad tiene a la venta múltiples productos que son parte de su oferta semanal, esta oferta puede ser cambiada ya sea porque se agrega un producto nuevo o porque se elimina uno de ellos. En estos productos se encuentran los siguiente grupos:

- Artículos enlatados.
- Productos de limpieza.
- Carnes.
- Granos. (Arroz, frijoles, entre otros)

Donde cada grupo indica los nombres de los productos. El administrador debe poder seleccionar alguno de los grupos realizar las siguientes acciones:

- Agregar productos.
- Eliminar uno de los productos.
- Actualizar uno de los productos.

El programa debe ser capaz de mostrar todos los elementos de la lista, de la siguiente manera:

- Producto 1: <producto>
- Producto 2: <producto>
- ...

- 
- Producto n: <producto>

Donde **n**, es el último producto del conjunto.

Los productos deben mostrarse en las siguientes circunstancias:

- Antes de eliminar un producto.
- Después de eliminar un producto.
- Después de agregar un producto.
- Después de actualizar un producto.

Ahora bien, si el producto que se desea agregar, ya existe en el conjunto, entonces no se agrega y se muestra un mensaje:

El producto ya existe.

Para realizar la eliminación de un producto, investigue cómo se elimina los elementos de las listas y de los arreglos en python.

## INTRODUCION:

Este programa se basa en poder gestionar los diferentes productos de una pulpería en general esto incluye la opción de agregar así como también darle la opción al usuario de eliminar los productos que no desea que estén en el inventario también se agregó el poder utilizar los productos

## **Entradas**

- **Agregar productos**
- **Eliminar productos**
- **Actualizar producto**

## **Salidas**

1. **Al agregar un producto:**
  - **Confirmación de que el producto ha sido agregado al grupo.**
  - **Lista actualizada de productos en el grupo correspondiente.**
2. **Al eliminar un producto:**
  - **Confirmación de que el producto ha sido eliminado del grupo.**
  - **Lista actualizada de productos en el grupo correspondiente.**
3. **Al actualizar un producto:**
  - **Confirmación de que el producto ha sido actualizado en el grupo.**
  - **Lista actualizada de productos en el grupo correspondiente.**
4. **Al intentar agregar un producto que ya existe:**
  - **Mensaje indicando que el producto ya existe en el grupo correspondiente.**
5. **Al mostrar los productos en un grupo:**
  - **Lista de productos en el grupo correspondiente, numerados secuencialmente.**

## **IMÁGENES DE LA ELABORACION DE MI CODIGO**

```

1  def mostrar_productos(grupo, productos):
2      print(f"Productos en el grupo {grupo}:")
3      for i, producto in enumerate(productos, 1):
4          print(f"Producto {i}: {producto}")
5      print()
6  def agregar_producto(grupo, productos):
7      producto = input(f"Ingrese el nombre del producto que desea agregar al grupo '{grupo}': ")
8      if producto in productos:
9          print("El producto ya existe.")
10     else:
11         productos.append(producto)
12         print(f"Producto '{producto}' agregado al grupo '{grupo}'.")
13         mostrar_productos(grupo, productos)
14 def eliminar_producto(grupo, productos):
15     producto = input(f"Ingrese el nombre del producto que desea eliminar del grupo '{grupo}': ")
16     if producto in productos:
17         productos.remove(producto)
18         print(f"Producto '{producto}' eliminado del grupo '{grupo}'.")
19         mostrar_productos(grupo, productos)
20     else:

```

```

20     else:
21         print("El producto no existe en este grupo.")
22 def actualizar_producto(grupo, productos):
23     producto_antiguo = input(f"Ingrese el nombre del producto que desea actualizar en el grupo '{grupo}': ")
24     if producto_antiguo in productos:
25         producto_nuevo = input(f"Ingrese el nuevo nombre para '{producto_antiguo}': ")
26         index = productos.index(producto_antiguo)
27         productos[index] = producto_nuevo
28         print(f"Producto '{producto_antiguo}' actualizado a '{producto_nuevo}' en el grupo '{grupo}'.")
29         mostrar_productos(grupo, productos)
30     else:
31         print("El producto no existe en este grupo.")
32
33 articulos_enlatados = ["Sardinas", "Atún", "Frijoles"]
34 productos_limpieza = ["Suavitel", "Jabón en polvo", "Limpiador de ventanas"] #USO DEL PROGRAMA
35 carnes = ["Pescado", "Carne molida", "Chorizo"]
36 granos = ["Arroz", "garbanzos", "Lentejas,"]
37
38 agregar_producto("Articulos enlatados", articulos_enlatados)
39 agregar_producto("Productos de limpieza", productos_limpieza)
40 agregar_producto("Carnes", carnes)
41 agregar_producto("Granos", granos)
42
43 eliminar_producto("Articulos enlatados", articulos_enlatados) #DE ESTA MANERA SE ELIMINAN LOS PRODUCTOS
44
45 actualizar_producto("Carnes", carnes) #LA OPCION DE PODER ACTULIZAR EL PRODUCTO
46
47 agregar_producto("Granos", granos) # AGREGAR UN PRODUCTO YA EXISTENETE
48
49 mostrar_productos("Articulos enlatados", articulos_enlatados)
50 mostrar_productos("Productos de limpieza", productos_limpieza) # CON ESTA BARRA DE CODIGO SE MOSTRARAN LOS PRODUCTOS FINALES
51 mostrar_productos("Carnes", carnes)
52 mostrar_productos("Granos", granos)

```

## **Referencias que utilice**

[Curso Python 3 desde cero #55 | Buscar elementos en una lista – Método index\(\) \(youtube.com\)](#)

[Listas, Tuples, Sets, Strings y Diccionarios en PYTHON \(youtube.com\)](#)

[Las FUNCIONES en PYTHON | ¿Para qué sirven y cómo se usan? \(youtube.com\) Metodo](#)

[index en listas python, rápido y sencillo. \(youtube.com\)](#)