

# AE6\_ABPRO-Ejercicio grupal

## GRUPO 1

### Integrantes:

Milena Sgombich

Raúl Jaque E.

Felipe Lobiano C

Tomás Pérez Rocha

Jonathan Matus V.

---

### Contexto

Un grupo de emprendedores ha comenzado un negocio de venta de ropa online denominado "Moda Xpress". La tienda vende ropa de temporada y cada artículo tiene información específica, como el nombre del producto, precio, cantidad disponible, y tamaño. Actualmente, gestionan el inventario a mano en hojas de cálculo, lo que a veces resulta confuso y propenso a errores. Debido a la expansión del negocio, decidieron implementar un sistema para administrar este inventario, donde se lean y escriban los datos directamente en archivos de texto.

Tomando como punto de partida las especificaciones del ejercicio, nos dividimos las funciones principales y comenzamos a desarrollar el programa:

### Funciones del Programa y Participación:

- Registrar nuevos productos en el inventario (Jonathan):

Fui el encargado de la estructura general del programa. Desarrollé la función **def main()** para inicializar y gestionar el flujo de la aplicación. También creé la función **def agregar\_producto()** que utiliza el modo de apertura **append** y el módulo **csv** para añadir nuevos productos al inventario de manera segura sin sobrescribir los datos existentes y **def mostrar\_menu()** para la interfaz de usuario con emojis.

- Consultar el inventario completo (Tomás):

Para poder acceder al inventario se creo la función **def consultar\_inventario()** que lee y muestra todo el contenido del inventario

- Leer los datos de un producto específico (Milena):

Se creó la función **def buscar\_producto\_especifico**, la cual busca en el inventario el producto requerido según su nombre. Para esto, se utilizó el método auxiliar **buscar\_por\_nombre**, que recorre la lista de productos y

compara los nombres hasta encontrar el producto con el nombre requerido. Se utilizó también un **try-except** para manejar cualquier error que pudiese ocurrir en la búsqueda.

- Modificar los datos de un producto ya existente (Raúl):

Con la función **def modificar\_producto()** modificamos los datos de un producto ya existente a través del INVENTARIO\_FILE

- Eliminar productos del inventario (Felipe):

Para eliminar productos del inventario se requiere filtrar el archivo y reescribir el inventario sin el producto deseado. Para esto se utilizó la función **def eliminar\_producto**, utilizando el método **.pop()** para eliminar el elemento por posición, luego filtrando los datos y reescribiendo el archivo sin el elemento. Se requiere una validación si el producto pertenece al inventario.

- Visualizar los detalles de un producto específico (Grupo completo):

Creamos la función **def obtener\_atributos** para visualizar los detalles de un producto específico y los atributos del archivo

- Crear backups de los datos de inventario (Grupo completo):

Con la función **crear\_backup()** hicimos las copias de seguridad del archivo de inventario para salvaguardar los datos ingresados