

Este proyecto consiste en un sistema de gestión de pedidos para una cafetería desarrollado en Python con tkinter para la interfaz gráfica. A continuación te cuento cómo funciona el sistema, qué necesidades cubre y cómo está organizado.

### ¿Qué necesidades cubre este sistema?

El sistema está pensado para resolver dos problemas importantes:

1. **Automatización de pedidos y gestión de inventario:** Los clientes pueden hacer sus pedidos de manera rápida y sencilla, y los empleados tienen una herramienta eficiente para gestionar los ingredientes disponibles.
2. **Interfaz gráfica amigable:** La interfaz es intuitiva y fácil de usar tanto para los clientes como para los empleados, lo que facilita su acceso y navegación.

### Clases y estructura del sistema

El sistema se organiza en varias clases, cada una con un propósito específico. Estas son las principales:

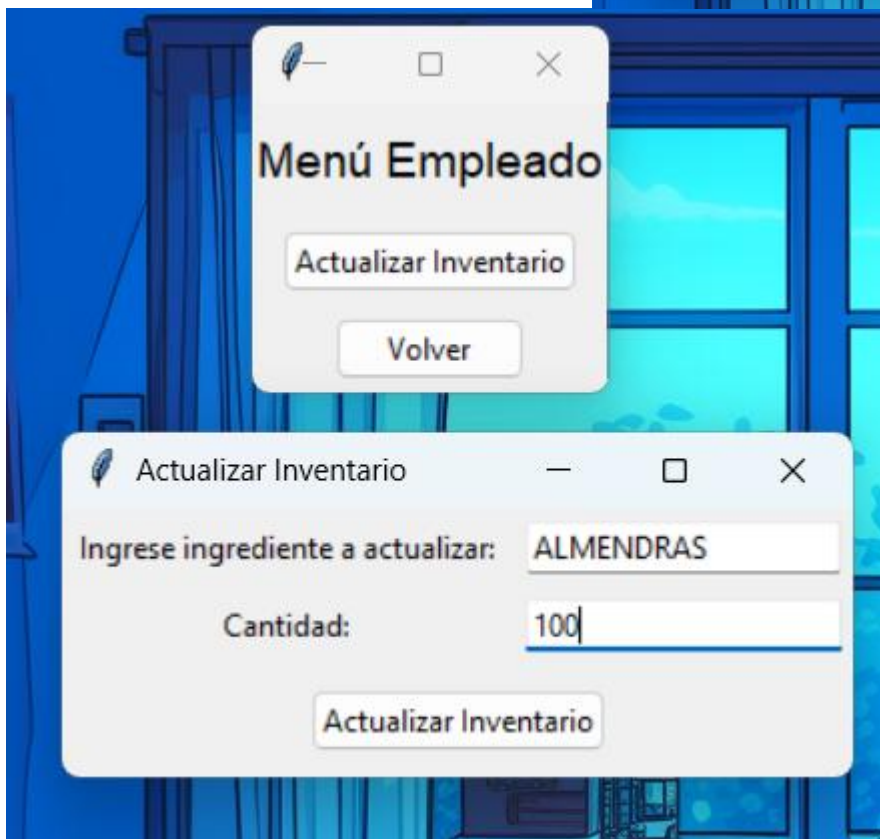
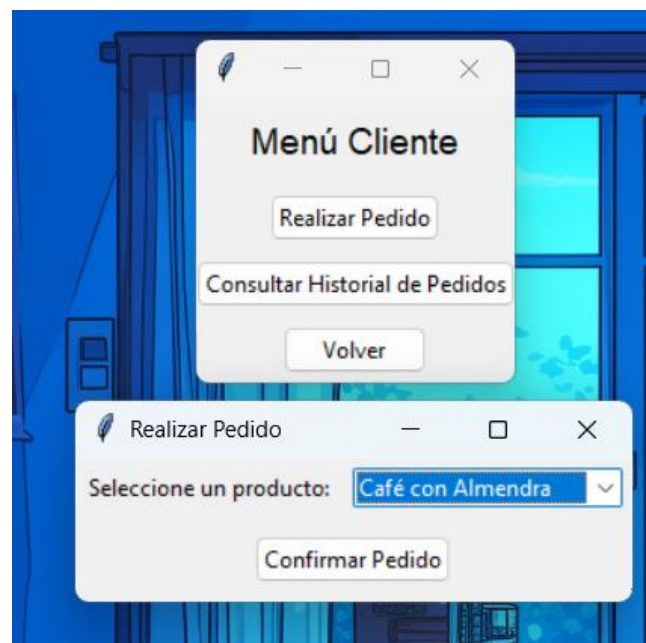
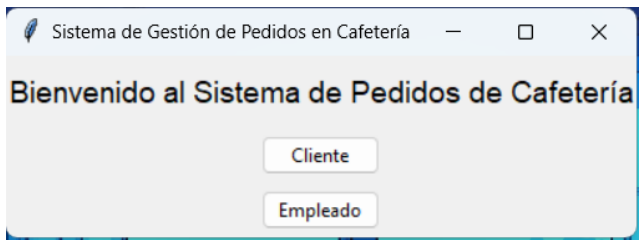
- **Persona:** Es la clase base para cualquier persona dentro del sistema, como un cliente o un empleado. Tiene los atributos básicos como nombre e identificación.
- **Cliente:** Hereda de Persona y se le agrega un historial de pedidos. Así, cada cliente tiene acceso a ver sus pedidos pasados.
- **Empleado:** También hereda de Persona, pero con un atributo extra: rol. Esto le permite al sistema saber qué funciones tiene el empleado dentro de la cafetería (por ejemplo, cajero, cocinero, etc.).
- **ProductoBase:** Representa los productos genéricos de la cafetería, como las bebidas o postres, con atributos como nombre, precio e ingredientes.
- **Bebida:** Hereda de ProductoBase y tiene atributos como tipo (caliente o fría), tamaño y personalizaciones (como agregar leche o jarabe).
- **Postre:** Hereda de ProductoBase y agrega características como vegano o sin\_gluten.
- **Inventario:** Esta clase maneja el stock de ingredientes disponibles, permitiendo a los empleados verificar la cantidad de cada ingrediente y actualizarlo conforme se usen o reciban más.
- **Pedido:** Representa un pedido realizado por un cliente, con una lista de productos, el estado del pedido (pendiente, en proceso, entregado) y el total a pagar.

- **Promocion:** Define las promociones especiales que pueden aplicarse a los pedidos, como descuentos. Esto permite que el sistema calcule el precio final con las promociones correspondientes.
- **CafeteriaGUI:** Es la clase que gestiona toda la interfaz gráfica del sistema. A través de esta clase, los clientes pueden hacer pedidos y consultar su historial, mientras que los empleados gestionan el inventario y los pedidos.

### Descripción de los atributos y métodos clave

- **Persona:**
  - Atributos: nombre, identificación.
  - Método: Inicializa los atributos mencionados.
- **Cliente:**
  - Atributos: Hereda de Persona y agrega historial\_pedidos para almacenar los pedidos anteriores.
  - Métodos: realizar\_pedido(), para realizar un nuevo pedido.
- **Empleado:**
  - Atributos: Hereda de Persona y añade el atributo rol.
  - Métodos: Inicializa los atributos y define las acciones que puede realizar el empleado.
- **ProductoBase:**
  - Atributos: nombre, precio, ingredientes.
  - Método: Inicializa los atributos mencionados.
- **Bebida:**
  - Atributos: Hereda de ProductoBase y añade tipo, tamaño, personalizaciones.
  - Método: Inicializa los atributos y define las características de la bebida.
- **Postre:**
  - Atributos: Hereda de ProductoBase y añade vegano, sin\_gluten.
  - Método: Inicializa los atributos y define las características del postre.
- **Inventario:**

- Atributos: ingredientes, un diccionario con los ingredientes y sus cantidades.
- Métodos: verificar\_stock() y actualizar\_stock(), para gestionar la disponibilidad de los ingredientes.
- **Pedido:**
  - Atributos: cliente, productos, estado, total.
  - Método: Inicializa los atributos y calcula el total del pedido.
- **Promocion:**
  - Atributos: descripcion, descuento.
  - Métodos: aplicar\_descuento(), para aplicar un descuento al pedido.
- **CafeteriaGUI:**
  - Atributos: root, productos, clientes, empleados, inventario.
  - Métodos: Varias funciones que permiten al cliente hacer pedidos, consultar su historial y al empleado gestionar los productos y el inventario.



<https://github.com/eduardo-antonio-rozete-flores/python>