

SEMANA 05

Ejercicio: Simulación de un Sistema Bancario en Python usando Programación Orientada a Objetos (POO)

Descripción:

Este programa simula un sistema bancario básico que permite gestionar clientes, cajeros y cuentas bancarias.

Se implementan las operaciones principales de un banco: **depósitos** y **retiros**, aplicadas a cuentas

de ahorro y cuentas corrientes, con un diseño orientado a objetos que facilita la extensión y mantenimiento del código.

Clases implementadas:

- 1. Persona (clase base):
 - Atributos: nombre (str), cédula (str), edad (int)
- 2. Cliente (hereda de Persona):
 - Atributos: es_vip (bool), cuentas (lista de cuentas)
 - Métodos: agregar_cuenta()
- 3. Cajero (hereda de Persona):
 - Métodos: procesar_deposito(), procesar_retiro()
- 4. Cuenta (clase base de tipo abstracto):





#UEAesExcelencia



- Atributos: número de cuenta (str), saldo (float), activa (bool), transacciones (int)
- Métodos: depositar(), retirar()
- 5. Cuenta Ahorro (hereda de Cuenta):
 - Método calcular_interes_anual()
- 6. Cuenta Corriente (hereda de Cuenta):
 - Atributo adicional: límite de sobregiro (float)
 - Método retirar() sobrescrito para permitir retiros con sobregiro

Características del ejercicio:

- Usa diferentes tipos de datos: `str`, `int`, `float`, `bool`
- Usa herencia para modelar relaciones entre clases
- Controla acceso y validación de operaciones con condiciones lógicas
- Incrementa automáticamente el contador de transacciones por cuenta
- Permite habilitar o deshabilitar cuentas usando el atributo `activa`

Objetivo para el estudiante:

Comprender y aplicar los conceptos fundamentales de la Programación Orientada a Objetos (POO),

incluyendo clases, herencia, encapsulamiento, y uso adecuado de tipos de datos en un contexto realista.







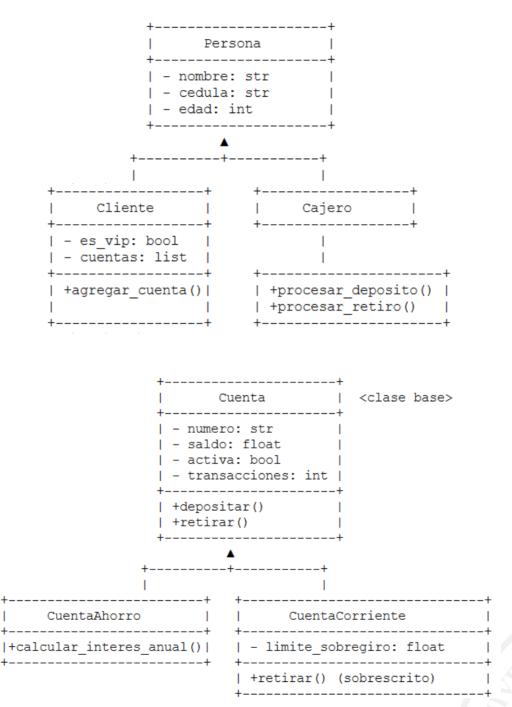
Mr. 2. 1/2 vía Puyo a Tena (Paso Lateral)



#UEAesExcelencia



Diagrama de Clases



- www.uea.edu.ec
- Mr. 2. 1/2 vía Puyo a Tena (Paso Lateral)