

INFORME EXPLICATIVO

Introducción

En este ejercicio se ha desarrollado una aplicación de gestión de cuentas bancarias, que permite realizar operaciones como: abrir cuentas, realizar ingresos y retirar cantidades y consultar saldos e información de las cuentas. Se ha utilizado la programación orientada a objetos (POO) e implementado el polimorfismo y la herencia.

Estructura del Proyecto

La aplicación está compuesta por varias clases:

1. **Clase Banco:** Se encarga de gestionar las acciones de las cuentas. Permite abrir nuevas cuentas, listar las cuentas existentes, obtener información de cuentas específicas, realizar ingresos y retirar cantidades, y verificar saldos. Utiliza una lista de tipo ArrayList para almacenar objetos de tipo CuentaBancaria.
2. **Clase CuentaBancaria:** Es una clase abstracta que representa una cuenta bancaria individual. Esta clase contiene un metodo abstracto para obtener información sobre la cuenta.
3. **Clase CuentaCorriente:** Es una subclase de CuentaBancaria que representa una cuenta corriente. Al igual que la clase padre, tiene metodos abstractos que son mplementados en las demas subclases.
4. **Controladores:** Se utilizan diferentes controladores para gestionar las diferentes operaciones de cuentas bancarias. En ellos, se valida distintos campos, como la verificación de que los valores ingresados sean numéricos y no contengan letras y viceversa, que el número de IBAN sea correcto, entre otros.

Implementación de POO

1. **Herencia:**
En este ejercicio se utiliza la herencia a través de una clase abstracta llamada CuentaBancaria. La clase CuentaCorriente hereda de CuentaBancaria, que actúa como clase padre para otras subclases. Al crear subclases como CuentaAhorro y CuentaCorriente, estas heredan las propiedades y metodos de CuentaBancaria. Cada subclase puede implementar métodos específicos, como devolverInfoString(), lo que permite crear objetos de tipo CuentaBancaria con las distintas subclases
2. **Polimorfismo:**
En este proyecto, el polimorfismo se implementa a través de método definido en la clase Cuenta bancaria llamado devolverInfoString (el cual se define en la interface y se implementa a la cuenta bancaria) y son sobrecargados a sus subclases. Asi, cada tipo de cuenta proporciona información de su propio tipo.