



1. ¿Cómo podría el patrón Singleton mejorar la gestión de configuraciones en un proyecto como JavaBank?

Previniendo inconsistencias en configuraciones en diferentes partes de la aplicación.



2. ¿Cómo podría el patrón Factory Method ayudar en la creación de diferentes tipos de cuentas?

Permitiendo instanciar diferentes cuentas sin modificar el código existente.



3. ¿Cómo puedes garantizar que el atributo saldo de una clase Cuenta solo se modifique mediante depósitos o retiros?

Definir el atributo saldo como privado y usar métodos públicos para acceder a él y para modificarlo.



4. ¿Cuál es la razón principal para usar la recolección de basura en Java?

Liberar memoria ocupada por objetos no utilizados.



5. Si necesitas que diferentes tipos de cuentas compartan un comportamiento común, ¿qué usarías?

Una clase abstracta



Continuar aprendiendo