

Ejercicios con “Class”

1. Implementar una clase que cargue los lados de un triángulo e implemente los siguientes métodos: imprimir el valor del lado mayor, otro método que muestre si es equilátero o no, otro para el área y la sobrecarga de toString.
2. A la anterior clase añadirle un constructor secundario sin parámetros que inicializa los lados a cero.
3. Crear una data clase **Producto** con atributos : nombre, precio, peso, largo . Una segunda clase **Almacen** con atributos: nombre, lista de productos con formato cola (primero entra , último sale) y dos métodos : **entrada(Producto)** y **salida(Producto)** donde se indica el producto y la cantidad

Crear un programa :

Desarrollar un programa que modele una cuenta bancaria que tiene los siguientes atributos, que deben ser de acceso protegido (elegir cual):

4. Saldo, de tipo float.
5. Número de ingresos con valor inicial cero, de tipo int.
6. Número de retiros con valor inicial cero, de tipo int.
7. Interés anual (porcentaje), de tipo float.
8. Comisión mensual con valor inicial cero, de tipo float.

La clase Cuenta tiene un constructor que inicializa los atributos saldo y comisión anual. (Valores por defecto 0)

La clase Cuenta tiene los siguientes métodos:

- **Ingresar** una cantidad de dinero en la cuenta actualizando su saldo.
- **Retirar** una cantidad de dinero en la cuenta actualizando su saldo. El valor a retirar no debe superar el saldo.

- **Calcular el interés mensual** de la cuenta y actualiza el saldo correspondiente. (saldo multiplicado por interés anual dividido por 12). Este método es privado.
- **Extracto mensual**: actualiza el saldo restándole la comisión mensual y calculando el interés mensual correspondiente (invoca el método anterior).
- **Imprimir**: muestra en pantalla los valores de los atributos.

La clase Cuenta tiene dos clases hijas:

Cuenta de ahorros: posee un atributo para determinar si la cuenta de ahorros está activa (tipo boolean). Si el saldo es menor a €1000, la cuenta está inactiva, en caso contrario se considera activa.

Los siguientes métodos se redefinen

- **Ingresar** : se puede ingresar dinero si la cuenta está activa. Debe invocar al método heredado.
- **Retirar**: es posible retirar dinero si la cuenta está activa. Debe invocar al método heredado.
- **Extracto mensual**: si el número de retiros es mayor que 4, por cada retiro adicional, se cobra €1,5 como comisión mensual. Al generar el extracto, se determina si la cuenta está activa o no con el saldo
- Un nuevo método **imprimir** que muestra en pantalla el saldo de la cuenta, la comisión mensual y el número de transacciones realizadas (suma de cantidad de consignaciones y retiros).

Cuenta corriente: posee un atributo de credito , el cual se inicializa en cero. Se redefinen los siguientes métodos:

- **Retirar**: se retira dinero de la cuenta actualizando su saldo. Se puede retirar dinero superior al saldo. El dinero que se debe queda como credito.
- **Ingresar**: invoca al método heredado. Si hay credito, la cantidad consignada reduce el sobregiro.
- **Extracto mensual**: invoca al método heredado.
- Un nuevo método **imprimir** que muestra en pantalla el saldo de la cuenta, la comisión mensual, el número de transacciones realizadas (suma de cantidad de consignaciones y retiros) y el valor de sobregiro.

A tener en cuenta: algún atributo debe ser privado/protegido para evitar que se modifique por otra clase.