TP 3: CLASSES, ENCAPSULATION

Exercise 01

- Create a ***BankAccount*** class, implement two methods, one depositing money,the other to withdraw money,

and of which you will ensure that the ***account balance*** attribute can never be negative,
add an other method that displays the account balance.

- Create the ManipulationSolde class for:
    1. Create an account1 and account2 object of type ***BankAccount***
    2. Add 5000 to account1 balance then withdraw 1000 from account1 balance
    3. Add 5000 to account balance2
    4. View the account balance of both accounts

Exercise 02

- Define a ***Book*** *class* with the following private attributes: Id, Title, Author (Full Name), Price.
- Define Setters/getters to the different attributes of the class.
- Define a constructor allowing the attributes of a book object to be initialized with values entered by the user. Knowing that Id must be auto-increment.

Exercise 03

- Consider the ***Employee class*** seen in the previous series whose private attributes are: Last name, First name, Function. The attributes Salary is public.
- add a public attribute **AccountB** of type ***Bankaccount***.
- In a separate ***Manipulation.class:***

    1. Create a ***TabEmp*** array of 3 elements of type Employee.
    2. Fill the table with values entered from the keyboard (Last name, first name, position, salary);
    3. Add to the balance attribute of each array element a value equivalent to the salary.
    4. Remove Amount from the different table elements.
    5. Display the employee's full name and their balance account for the five (03) table elements

Identifiant	Nom	Prénom	Fonction	salaire
01	Benaissa	Ahmed	Administrateur	32 000
02	Lakhal	Ali	Enseignant	45 000
03	Benali	Samia	Ingénieur	35 000