

Exercice 1. Classes et visibilité

Q. Parmi les appels suivant indiquez ceux qui sont corrects et ceux qui ne le sont pas

```
public class A {
    public int i;
    private int a;
    protected int b;
    public void afficher() { System.out.println(i+a+b); }
}
public class B extends A {
    public void afficher() { System.out.println(i+a+b); }
}
public class D {
    Public A a = new A();
    Public void afficher() { System.out.println(a.i + a.a + a.b); }
}
public class E {
    Public B b = new B();
    Public void afficher() { b.afficher();}
}
public class F {
    Public B b = new B();
    Public void afficher() { System.out.println(b.i + b.a + b.b) }
}
```

Exercice 2 - Classes abstraites

La classe *Vector* (package `java.util`) est couramment utilisée. Nous allons ici l'utiliser de deux façons pour créer une FIFO (First In / First Out) et une LIFO (Last In / First Out).

- Créer la classe abstraite *Stack* qui contient les classes abstraites:

```
Object get();
void put(Object);
```

- Implémenter cette classe de deux façons: par une classe *FIFOStack* puis par une classe *LIFOStack* selon le comportement de pile désiré. (Utiliser les méthodes *addElement()* et *removeElement()* de la classe *Vector*).

Exercice 3 - Interfaces

- Transformez la classe *Stack* pour qu'elle devienne une interface qui est implémentée par les classes *FIFOStack* et *LIFOStack*.
- De quels type(s) sont les instances de *FIFOStack* et *LIFOStack* ?