



Exercice 1 – Analyse de code 1

```
public class Parent {
    int x;
    Parent(int k) {x=k;}
    int ajoute(int a) {
        return x+a;
    }
    public void moi() {
        System.out.println(" x = "+ x);
    }
}
```

```
public class Enfant1 extends Parent {
    int y;
    Enfant1 (int k, int l) {
        super(k); y=l;
    }
    int ajoute(int a) {
        return x+2*a;
    }
}
```

```
public class Enfant2 extends Enfant1 {
    int z ;
    Enfant2 (int k, int l, int m) {
        super(k, l); z= m;
    }
    int ajoute(int a) {
        return x+3*a;
    }
    public void moi() {
        super.moi();
        System.out.println(" z = "+ z);
    }
}
```

```
public class Essai{
    public static void main (String args[]) {
        int a =2;
        Parent p = new Parent(3);
        p.moi();
        System.out.println(" ajoute("+ a +") = "+ p.ajoute(a) );
        Enfant1 e1 = new Enfant1(3, 4);
        e1.moi();
    }
}
```

```

        System.out.println(" ajoute("+ a +" ) = "+ e1.ajoute(a) );
        e1 = new Enfant2(3, 4, 5);
        e1.moi();
        System.out.println(" ajoute("+ a +" ) = "+ e1.ajoute(a) );
    }
}

```

1. Quels sont les attributs dont disposent les classes `Enfant1` et `Enfant2` ?
2. Ecrivez le résultat de l'exécution de la classe `Essai`.

Exercice 2. Analyse de code 2

Soit

```

public abstract class Animal {
    public abstract void afficher();
}
public interface peutNager {
    public abstract void nage() ;
}
public class Poisson extends Animal implements peutNager{
    public void afficher() { System.out.println("Je suis un poisson");
    public void nage() {System.out.println("Je nage je suis un poisson") ;}
}
public class Mamifere extends Animal{
    public void afficher() { System.out.println("Je suis un mamifère");
}
Public class Baleine extends Mamifere implements peutNager() {
    public void nage() {System.out.println("Je nage je suis une baleine") ;
}

```

Parmi les appels suivant indiquez ceux qui sont corrects et ceux qui ne le sont pas, donnez le résultat de ce qui est correct.

```

Animal A1 = new Animal() ;
Animal A2 = new Poisson() ;
Baleine B = new Mamifere() ;
Poisson B = new Baleine() ;
Mamifere M1 = new Baleine() ;
M1.afficher();
M1 = new Mamifere() ;
M1.afficher() ;
peutNager PN = new Poisson();
PN.afficher();
Poisson P = new peutNager();
peutNager Pp = new peutNager();
peutNager PN2[] = new peutNager[2] ;
PN2[0] = new Poisson() ;
PN2[1] = new Baleine() ;
PN2[0].afficher() ;
PN2[0].nage() ;
PN2[1].afficher() ;

```