

**Examen de la session de rattrapage
03 Mars 2016**

N.B.

- Toute réponse non justifiée sera considérée comme fausse.
- N'ajoutez pas d'autres méthodes si elles ne sont pas demandées.

Exercice 1 (3pts)

Donnez les résultats affichés par les instructions Q1, Q2, Q3, Q4, Q5 et Q6.

```
class A {  
    public String nom;  
    public static int k = 2;  
    public A(String nom) {  
        this.nom = nom;  
    }  
    public static void f() {  
        k += 10;  
    }  
}
```

```
class B extends A {  
  
    public static int k = 6;  
    public B(String nom) {  
        super(nom);  
    }  
    public static void f() {  
        k += 5;  
    }  
}
```

```
public class Test {  
    public static void main(String [] args) {  
        A a = new A("Ali");  
        B b = new B("Lina");  
        System.out.println("nom = " + a.nom + " k = " + A.k); // Q1  
        System.out.println("nom = " + b.nom + " k = " + B.k); // Q2  
        a = b;  
        System.out.println("nom = " + a.nom + " k = " + A.k); // Q3  
        System.out.println("nom = " + b.nom + " k = " + B.k); // Q4  
        A.f();  
        System.out.println("k = " + A.k); // Q5  
        System.out.println("k = " + B.k); // Q6  
    }  
}
```

Exercice 2 (3pts)

1. Écrivez une interface nommée **Salutation** qui contient la méthode **saluer()**.
2. Écrivez la classe **Personne** qui contient les deux attributs privés, nom et prénom.
3. Écrivez deux classes **Marocain** et **Etranger** qui héritent de **Personne** et implémentent la méthode **saluer()**. La méthode doit afficher un message relatif à chaque personne.
4. Utilisez les différentes classes dans la classe **TesterSaluer** pour afficher les différentes types de salutation.

Exercice 3 (2pts)

Quel est le résultat de l'exécution du programme suivant :

```

public class Test {
    public static void main(String args []) {
        ClasseA [] tab = new ClasseA [4];
        for (int i = 0; i < 4; i++) {
            tab[i].setK(i);
            System.out.println(tab[i]);
        }
    }
}

```

```

class ClasseA {
    private int k;
    public String toString() {
        return "k = " + k;
    }
    public void setK(int k) {
        this.k = k;
    }
}

```

Exercice 4 (2pts)

La méthode **Double.parseDouble()** nécessite un argument de type **String**. Elle génère une erreur si la chaîne ne peut être convertie vers un nombre. Écrivez une application qui essaie d'accepter un double et génère une exception de type **NumberFormatException** sinon. Le bloc **catch** affiche un message d'erreur.

Exercice 5 (10pts)

Le but est de développer une application basique pour la gestion des salles multi-media. Pour cela, vous aurez besoin des classes suivantes :

1. La classe **Personne** qui contient les attributs privés, nom et prénom ;
2. La classe **Enseignant** hérite de **Personne** et contient aussi l'attribut public département.
3. La classe **SeanceTP** qui contient les attributs publics, responsable (un enseignant), la matière enseigné et le groupe.
4. La classe **SalleTP** qui contient :
 - (a) la constante public et statique capacité (nombre maximale d'étudiants) ;
 - (b) un tableau à 2 dimensions de type **SeanceTP** qui contient l'emploi du temps de la salle. La première dimension désigne le jour (0 : lundi, 1 : Mardi ... 5 : samedi). La deuxième dimension désigne la séance (une séance dure 2h). Par exemple : **emploiTemps[1][2]** désigne la troisième séance (de 14h à 16h) du Mardi.
 - (c) les attributs privés, numéro et nombre de PCs (initialisé par défaut à 16) ;
 - (d) la méthode **afficheEmpoiTemps()** qui permet d'afficher l'emploi du temps.

Les différentes classes doivent contenir :

- un constructeur pour initialiser les différents attributs ;
 - la méthode **toString()** pour afficher les différentes informations concernant la classe.
5. Écrivez une classe **GestionSallesTPs** qui contient une méthode **main()**. Dans la méthode **main()** :
 - déclarez un tableau de 9 salles de TPs.
 - initialisez les données pour la salle 1 comme suit :

	8h-10h	10h-12h	14h-16h	16h-18h
Lundi	SMI-S5-A POO Oujdi Ali	SMI-S5-B POO Oujdi Ali		
Jeudi			SMA-S5-C C++ Berkani Lina	SMA-S5-D C++ Berkani Lina

- affichez l'emploi du temps de la salle 1.