 Ecole Supérieure Privée d'Ingénierie et de Technologies		Année Universitaire : 2018-2019 Devoir Surveillé	
Module : Conception orientée objet et programmation Java		Documents autorisés : Non	
Enseignants : Equipe JAVA		Nombre de pages : 4	
Date : Vendredi 23/11/2018	Heure : 11h	Durée : 1h	
Classes : 3A			

QCM / QUESTIONS DE COUR (4 points)

- QUESTION 1:(1 point)

Combien d'instances de la classe A crée le code suivant?

```
A x,u,v;
x=new A();
A y=x;
A z=new A();
```

- A. Aucune
- B. Cinq
- C. Trois
- D. Deux**
- E. erreur de Compilation

- QUESTION 2: (1 point)

```
public class ClasseA {
    public void affiche(){
        System.out.println("je suis un objet de ClasseA") ;
    }
}
class ClasseB extends ClasseA {
    public void affiche(){
        super.affiche() ;
        System.out.println("je suis un objet de ClasseB") ;}
}
```

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

```
public class ClasseTest {
    public void main(String [] args) {
        ClasseA b = new ClasseB() ;
        b.affiche() ;
    }
}
```



- **QUESTION 3: (1 point)**

Choisissez la bonne ou les bonnes réponse(s) :

- A. Une classe peut implémenter plusieurs interfaces mais doit étendre une seule classe**
- B. Une classe peut implémenter plusieurs classes mais doit étendre une seule interface
- C. Une classe peut implémenter plusieurs classes et peut étendre plusieurs interfaces
- D. Une classe doit, impérativement, implémenter une seule interface et étendre une seule classe

- **QUESTION 4: (1 point)**

package test;

public class Target { String name = "hello"; }

Qui peut accéder et modifier la variable name?

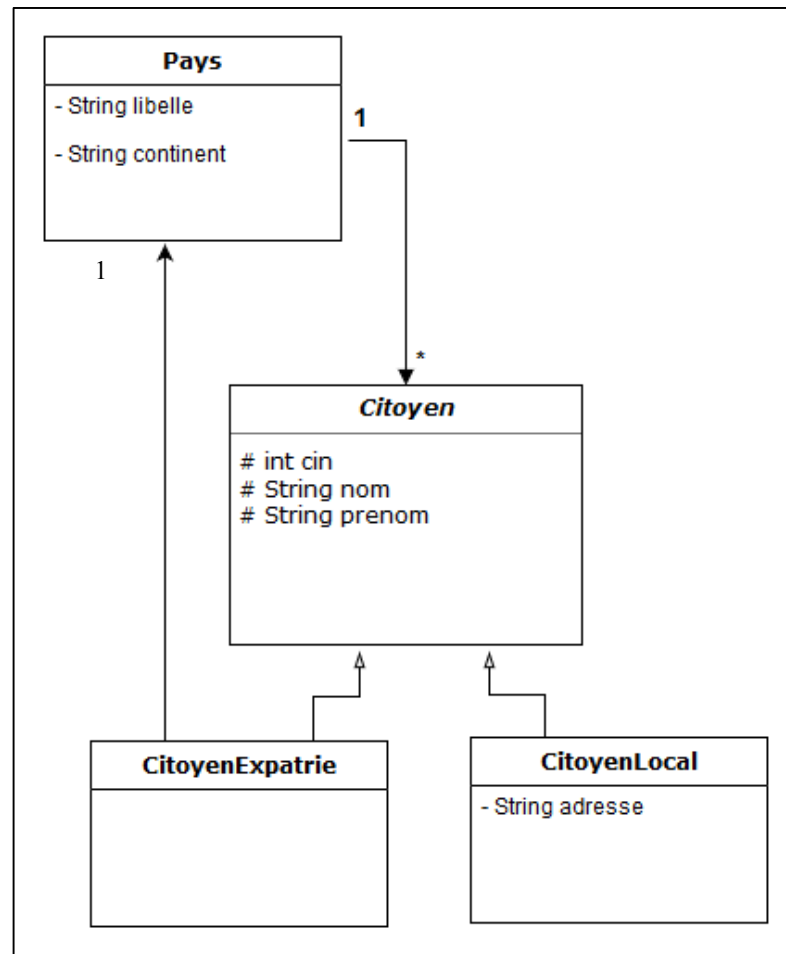
Choisissez la bonne ou les bonnes réponse(s) :

- A. n'importe quelles classes
- B. seulement la classe Target
- C. n'importe quelles classes dans le paquetage test**
- D. n'importe quelles classes qui hérite de Target en dehors du paquetage test

PROBLÈME (16 POINTS)

REMARQUE1 : L'utilisation des Collections (List, Set ou Map) est strictement interdite

On souhaite établir une étude, à travers un système informatisé, sur les populations de certains pays du monde, ainsi que les activités de la migration de leurs citoyens. Pour cela, vous avez le diagramme de classe suivant :



1^{ère} Question : (4 points)

- Développez la classe **Citoyen** qui est caractérisée par :
 - `int cin`
 - `String nom`
 - `String prenom`
- Les 2 constructeurs : par défaut et paramétré
- Les méthodes getter et setter de l'attribut **cin**.
- **REMARQUE2 : On suppose que les getters et setters des attributs de toutes les classes sont implémentés**
- La méthode **equals(Object obj)** qui permet de tester l'égalité entre deux citoyens, sachant que deux citoyens sont identiques s'ils ont les même nom et cin.
- La méthode **toString ()** qui permet d'afficher les caractéristiques d'un Citoyen

2^{ème} Question : (3 points)

Implémenter La classe un **CityoenExpatrie**

- ses attributs de chaque classe
- Le constructeur paramétré
- La méthode **toString()**

3^{ème} Question : (8 points)

On vous propose de développer la classe Pays :

- Ses attributs (0.5 point)
- Le constructeur paramétré(0.25 point)
- La méthode **void ajouterCitoyen(Citoyen c)** qui permet d'ajouter un citoyen au pays (Le nombre Max de citoyens dans un pays est 800) (1 point)
- La méthode **void afficherStat()** qui permet d'afficher le nombre de chaque type de citoyen sous le format suivant « **Nb Cityoens locaux= xxLoc // Nb Citoyens expatries= xxExpa** »(1.5 points)
- La méthode **void supprimer(Citoyen c)** qui permet de supprimer un citoyen (1.5 points)
- Une méthode **estTunisien(Citoyen c)** qui retourne true si le **CityoenExpatrie** passé en paramètre est tunisien, false sinon. (1.5 points)
- La méthode **afficherNbTunisiens()** qui retourne le nombre des citoyens tunisiens expatriés dans le pays courant. (1.5 points)

Cette méthode déclenchera une exception de type **NbMaxTunisiens** si le nombre des expatriés tunisiens dépasse le 150. Vous devez implémenter la classe d'exception. (1.25 points)