**TP: Eléments de bases du langage Java**

**Exercice 1 :**  
Ecrire une classe «Personne » qui a:  
Trois variables d’instances, nom et prénom, de types « String » ,et âge, de type byte ;  
Deux variables de classe, nbPersonne de type short, et nomDev de type string : nbPersonne désigne le nombre des objet, de type Personne, Crées. Et nomDev désigne le nom du développeur de la classe.  
Un constructeur pour initialiser le nom le prénom et l’âge.  
Comment peut-on initialiser les champs nomDev et nbPersonne. Donner tous les solutions possibles ?  
Les modificateurs du nom, prénom et âge.  
les accesseurs : getPersonne() qui retourne le nom et prénom de la personne. getNbPersonne et getNomDev.

**Exercice** **2 :** Compiler la classe suivante et analyser les erreurs   
Class Carre {double cote ;   
public static double surface () {return cote\*cote ;}  
}  
**Exercice 3 :** Tester sur machine les codes suivants, puis expliquer les erreurs et corriger-les !  
a) **int i = 60; short b =(short) (i + 5);  
b) int i = (int)1.99;  
c)byte b1 = (byte)200; byte b2 = (byte)315; byte b3 = (byte)(b1 + b2);   
Exercice 4:**   
Dans une classe de test contenant la méthode « main() » vérifier et critiquer le code suivant qui déclare:

final double MIN= 1 ;

final double MAX= ;

puis lire la valeur de Max du clavier.

MAX=MIN+10 ;

Comparer cette syntaxe avec celle de C/C++ .

**Exercice 5 :** Considérons la classe représentée par le diagramme de UML comme suis :

|  |
| --- |
| Personne |
| -nom : String  -prenom : String  -age : byte |
| +Personne (String , String , byte )  +setNomPrenom ( )  +setage( )  +toString() |

Dans une classe de teste, analyser et corriger le code suivant :

*final Personne f = new Personne («Tamis », « Karim », 30) ;*

*System.out.println(f)*

*f.setNomPrenom( « Najar », « Taha ») ;*

*f= new Personne(« Toumi », « Fatima », 25) ;*

**Exercice 6 :** Ecrire une classe Calendrier qui a les champs:  
Jour : type énuméré qui peut prendre comme valeur : Lundi, Mardi… Dimanche  
date : type byte   
Mois : type énuméré qui peut prendre comme valeurs :   Janvier, Février… Décembre  
année : type short  
Un constructeur pour initialiser les quatre champs.  
Une méthode toString pour afficher le jour, la date, le mois et l’année d’un Objet calendrier.  
Ecrire une classe teste pour tester les méthodes de la classe Calendrier.  
b) écrire les énumérations à l’intérieur de la classe Calendrier  
b) Refaire le même travail mais en écrivant les énumérations

**Exercice 7 :**

1) Déclarer un tableau d’entier et affecter-lui les éléments {5, 13,-7, 12 , 25}. Utiliser toutes les syntaxes possibles.  
2) Afficher la longueur du tableau  
2) Créer un autre tableau d’entier et comparer les éléments des deux tableaux.  
3) Appliquer les méthodes suivantes sur les tableau ci-dessous et afficher les résultats : toString, sort, binarySearch, fill , hashCode, copyOf, copyOfRange, asList.  
Refaire la même chose mais cette fois ci avec des tableaux de « Book ».  
 **Exercice 8:**

String [] str ={« youssef », «Ahmed », « ahmed »}  
Afficher la longueur du tableau ainsi que la longueur de chacune des chaines.  
Comparer str[0] et str[1] en utilisant equals , compareTo(), compareToIgnoreCase()   
Trier le tableau str par ordre alphabétique croissant  
Utiliser la classe StringBuilder pour créer la chaine s initialisé à « Prof ». Changer la chaine à « Prof ESTO »  
Créer une autre chaine t initialise « Prof ESTO » puis comparer str et t par la méthode equals() .  
Maintenant executer le code :  
String s= « Prof ESTO » , String t= « Prof ESTO » ;  
System.out.printls(s.equals(t)) ;  
Conclure !  
Insérer la sous chaine Université Med Premier a la chaine t.  
Supprimer ESTO de la chaine str.

**Exercice 9 :**

-écrire une application java qui permet de calculer le produit des entiers entrée au moment de l’exécution du programme.

-écrire une methode qui a un nombre variable d’arguments et qui retourne le produit de ses arguments effectifs .

-utiliser cette méthodes dans la question précédente.