

Devoir 3 :

1)

Une classe peut contenir des champs, des méthodes avec implémentation et peut être instanciée, tandis qu'une interface ne contient que des méthodes abstraites (sans implémentation) et des constantes, et elle est implémentée par les classes pour définir le comportement.

2)

Le but d'une interface fonctionnelle est de définir la signature d'une méthode qui pourra être utilisée pour passer en paramètre :

une référence vers une méthode statique.

une référence vers une méthode d'instance.

une référence vers un constructeur.

3)

Le type sert à déterminer la taille de l'espace mémoire et la façon dont est interprété le code binaire de la valeur qui y est stockée.

4)

L'interface Comparable modélise les objets qui possèdent un ordre total . Cette interface permet d'utiliser des collections d'objets ordonnés comme TreeSet.

5)

la méthode compareTo(autre) renvoie un entier positif si l'objet est supérieur à l'autre, 0 s'ils sont égaux, et négatif sinon.

6)

-Un supertype est un type d'une superclasse. Autrement dit, c'est un type de la classe dont on hérite, et au delà.

-Sous-type est un datatype lié à un autre type de données, ledit super type, par une relation de substituabilité, compris comme un programme écrite d'utiliser des éléments construits sur le modèle supertype, il peut aussi fonctionner avec des éléments construits sur le modèle de sous-type.

7)

Une classe abstraite ne peut pas être instanciée. Il est nécessaire de créer une classe concrète (c'est-à-dire non abstraite !) qui l'étende, et d'instancier cette classe concrète.

8)

L'interface est assez proche de la classe abstraite, mais aucune méthode n'est implémentée dedans. Il y a juste la déclaration (la signature) des méthodes. Ainsi, une classe qui implémente (on ne dit pas hériter) une interface, devra obligatoirement implémenter les méthodes. Ainsi, l'interface est une sorte de contrat : on présente les noms des méthodes (et ainsi souvent leur objectif) sans se soucier de la méthode dont elles seront implémentées.

9)

Une classe abstraite permet, entre-autres, de définir les caractéristiques communes de plusieurs classes d'objets.

10)

Les interfaces servent à créer des comportements génériques: si plusieurs classes doivent obéir à un comportement particulier, on crée une interface décrivant ce comportement, on est la fait implémenter par les classes qui en ont besoin. Ces classes devront ainsi obéir strictement aux méthodes de l'interface (nombre, type et ordre des paramètres, type des exceptions), sans quoi la compilation ne se fera pas.