Annexe 12 - Sous-Classes / héritage

Une sous-classe hérite des methodes et des variables d’instance de sa superclasse. Elle peut donc les utiliser, les redefinir et en créer d'autres, en autant que les modificateurs d'accès le permettent.

**Signature d'une sous-classe :**

public class *nom de la sous-classe* extends *nom de la classe*

**Mot-clé super :**

- permet d'accéder aux méthodes de la superclasse qui ont été modifiées à l'intérieur de la sous-classe.

Ex.:

Voir annex 12

On ajoute super devant une methode afin d’instister pour que la methode apppeler sois celle de la superclasse(en cas de confusion)

- super est très utilisé dans les constructeurs de sous-classes.

Ex.: Voir annex 12

NB. :super(paramètres) doit absolument être a la premiere ligne

**Exercice :**

A- Créer une classe Prisme servant à modéliser des prismes rectangulaires. Au-delà des méthodes habituelles, codez 2 constructeurs : un initialisant toutes les variables à 1 unité et l'autre permettant d'initialiser les variables à des valeurs quelconques.

B- De plus, coder une méthode permettant de calculer l'aire de la surface du Prisme modélisé et une permettant de calculer le volume du Prisme modélisé.

C- Créer dans le même package que la classe Prisme une nouvelle classe - Cube - modélisant des cubes !

1. Cette classe est une sous-classe de Prisme
2. Bâtir les deux constructeurs appropriés ( faire appel aux constructeurs de Prisme avec super )
3. Déterminer si de nouvelles versions d’aire et de volume sont nécessaires

D- Créer dans le même package que la classe Prisme une nouvelle classe - PyramideBaseCarree – modélisant des pyramides à base carrée

1. Cette classe est une sous-classe de Prisme
2. Bâtir les deux constructeurs appropriés
3. Redéfinir les méthodes aire() et volume() ( il vous faut connaître l'apothème )

E-Pour tester le tout, créer une classe TestPyramide contenant qu'une méthode, soit main ( String [] args ) dans laquelle vous…

* créez deux prismes:
  + p1 avec le constructeur par défaut
  + p2 ayant une longueur et une hauteur de 2 unités et une largeur de 4 unités
* créez deux pyramides à base carrée : py1 avec le constructeur par défaut, py2 avec une hauteur de 6 et une arête de base de longueur = 9.
* créez un Cube c1 avec une arête de longueur 35.
* affichez les données / résultats de méthodes suivantes, lorsque possible :

- p1.setLongueur(25)

- p1.aire()

- py2.volume ()

- py1.setHauteur (35)

- py1.volume()

- py2.aire()

- c1.aire()

Rep

102

162

11.6

2167350