



RPI

Année 2015 – 2016

Java – TP1

## Exercice 1 :

a) Créer un programme Java qui permet à l'utilisateur de saisir la longueur et le diamètre d'un crayon et d'afficher son volume.

Exemple de résultat à obtenir :

```
Saisir une longueur
8
Saisir un diametre
0.85
Le volume du crayon = 4.5373
```

Voici la classe Crayon (représentée selon UML) :



b) Créer la classe Crayon avec ses 2 propriétés « longueur » et « diamètre ». Pour le moment, les propriétés ne sont pas privées et la classe Crayon ne dispose pas de méthode.

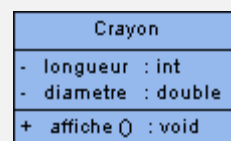
c) Créer un programme Java « UtiliseCrayon.java » qui permet à l'utilisateur de créer 2 crayons de longueurs différentes et d'afficher les propriétés de chacun.

d) Grâce au mot clé « private », indiquez maintenant que les propriétés de « Crayon » sont privées. Que fait votre programme « UtiliseCrayon » ?

e) Créer la méthode **publique** « affiche » de la classe « Crayon » et modifiez votre programme « UtiliseCrayon » pour afficher les propriétés de vos 2 crayons.

Voici la nouvelle classe Crayon

Que se passe t-il ?



f) Pour pouvoir créer des crayons avec des propriétés différentes, créer un constructeur « Crayon » qui reçoit 2 paramètres « lg » et « diam ». Grâce à votre programme « UtiliseCrayon », créez 2 crayons différents et afficher leurs propriétés.

Exemple de résultat à obtenir :

```
Proprietes du premier crayon:
diametre=0.5
longueur=5
```

```
Proprietes du second crayon:
diametre=0.8
longueur=7
```

Essayez de définir la méthode « affiche » privée. Que se passe-t-il quand vous essayez d'utiliser vos crayons ?

g) Créer la méthode publique « donneVolume » de la classe « Crayon » et modifiez votre programme « UtiliseCrayon » pour afficher aussi le volume de vos 2 crayons.

h) Votre crayon est usé ! Il faut donc le tailler ! Tailler un crayon consiste à diminuer sa longueur de quelques millimètres. Créer la méthode publique « taille » qui reçoit en argument le nombre de millimètres à tailler.

Avec votre programme « UtiliseCrayon », tailler vos crayons et afficher les nouvelles caractéristiques de vos crayons.

Exemple

```
.../...  
    leCrayon.taille(10); // Taille le crayon d'un centimètre  
    leCrayon.affiche(); // La longueur du crayon va être 1 cm plus courte  
.../...
```

Votre classe Crayon a maintenant cet aspect

Crayon		
-	longueur : int	
-	diametre : double	
+	<<Constructor>> Crayon (int lg, double diam)	
+	donneVolume ()	: double
+	affiche ()	: void
+	taille (int longu)	: void