

LES MODIFICATEURS D'ACCÈS EN JAVA

1. Introduction

Lorsque l'on crée une classe, il faut réfléchir à la façon dont sera utilisée la classe. On peut restreindre l'accès à certaine partie du code un peu comme on le fait avec des permissions dans un système de fichiers, ou avec des privilèges dans une base de données.

Ce résumé a pour but de montrer les usages les plus fréquents des modificateurs d'accès. La documentation complète se trouve à l'adresse suivante (en anglais) : <http://docs.oracle.com/javase/tutorial/java/javaOO/accesscontrol.html>

Le code source des exemples est dans *DemoEncapsulation.tar.gz*.

2. Cas des classes

Deux cas se présentent :

- la classe est déclarée publique (avec le mot-clé *public*) : elle est accessible par toutes les classes depuis n'importe quel paquet (c'est le cas le plus fréquent) ;
- la classe est déclarée sans modificateur d'accès : elle n'est pas utilisable en dehors du paquet où elle est déclarée.

Exemple :

```
package a;
```

```
public class Client {  
    public Client() {  
    }  
}
```

```
package a;
```

```
// Produit n'est pas déclaré public  
class Produit {  
    public Produit() {  
    }  
}
```

```
package b;
```

```
import a.Client;  
//import a.Produit; // Produit n'est pas visible hors du  
// paquet a : erreur de compilation
```

```
public class Entreprise {  
  
    public static void main(String args[]) {  
        Client c = new Client();  
        //Produit p = new Produit(); // impossible  
    }  
}
```

```
}
```

Une classe ne peut être déclarée privée (*private*) ou protégée (*protected*).

3. Cas des membres d'une classe

Définition : on appelle membres d'une classe les champs (qui peuvent être des constantes), les constructeurs, les méthodes de cette classe.

En Java, il existe quatre modificateurs d'accès (du plus restrictif au moins restrictif) : *private*, « rien », *protected*, *public*.

Nous avons déjà étudié et souvent utilisé les mots-clés *private* et *public*. Le cas où il n'y a pas de modificateurs d'accès ne sera pas étudié ici.

Un membre déclaré privé n'est pas visible, n'est pas accessible de l'extérieur de la classe.

Un membre déclaré publique est visible de partout.

Un membre déclaré protégé est un compromis entre privé et publique. Le membre protégé est visible, bien sûr, dans la classe où il est déclaré mais aussi dans les sous-classes de la classe où il est déclaré.

À titre d'exemple, voir le code dans le paquet *demo_protege*.

Les modificateurs d'accès s'appliquent uniquement aux classes, aux champs et aux méthodes. Ils ne s'appliquent donc pas aux paramètres de méthodes ni aux variables locales.

Les règles à suivre pour déterminer le bon modificateur d'accès :

1. toujours utiliser *private* pour les champs qui ne sont pas des constantes (les constantes pouvant être déclarées publiques selon les besoins) ;
2. Les constructeurs sont la plupart du temps publiques ;
3. les méthodes sont déclarées privées, protégées ou publiques selon les besoins (essayez d'être le plus restrictif possible).

4. L'encapsulation

L'encapsulation consiste à déclarer les champs d'une classe privés (ou pour être un peu moins restrictifs protégés) mais en aucun cas publiques. Pour accéder à ces champs de l'extérieur de la classe où ils sont définis, on utilise des méthodes appelés *getters* et *setters*.

À titre d'exemple, voir le code dans le paquet *demo_encapsulation*.

L'encapsulation permet de mieux contrôler les valeurs des attributs et permet de modifier l'implémentation de la classe sans modifier la partie publique d'une classe. L'encapsulation permet ainsi plus de souplesse.