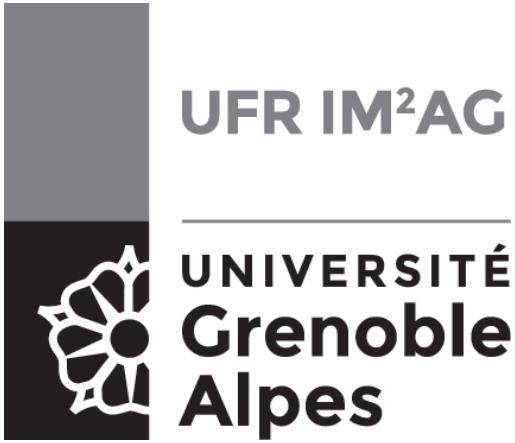

DU-ISN
Structuration de code en java

Présentation succinte et concise d'un petit bout du langage ayant pour objectif un passage rapide à la pratique

Paquetages

Un paquetage est un regroupement de classes et interfaces

- espace de nommage distinct
- seule une partie du contenu du paquetage est visible en dehors
- correspond à la notion de module logiciel

On définit un module `Toto` en

- plaçant tous les fichiers qui le constituent dans un répertoire Toto
- commençant tous ces fichiers par la ligne :
  `package Toto;`

Utilisation de paquetages

Deux possibilités pour utiliser un paquetage

- nommage complet (chemin vers le nom utilisé)

```
MesPiles.Pile f;

f = new MesPiles.PileListe();
f.empiler(42);
```

- import dans l'espace de nommage principal (attention à la pollution)

```
import MesPiles.Pile;       // classes et
import MesPiles.PileListe; // interfaces utilisées
// ou
import MesPiles.*;          // TOUTE la partie
...                         // publique du paquetage
Pile f;
f = new PileListe();
f.empiler(42);
```

Utilisation de paquetages de la bibliothèque standard

Deux possibilités pour utiliser un paquetage

- nommage complet (chemin vers le nom utilisé)

```
java.util.Random r;
r = new java.util.Random(graine);
System.out.println("Entier dans [0;9] : " +
                                   r.nextInt(10));
```

- import dans l'espace de nommage principal (attention à la pollution)

```
import java.util.Random; // classe Random
// ou
import java.util.*;      // TOUT java.util
...
Random r;
r = new Random(graine);
System.out.println("Entier dans [0;9] : " +
                                   r.nextInt(10));
```

Attention, organisation hiérarchique mais import à plat
`import java.util.*` n'inclut pas `import java.util.regex.*`

Visibilité

Qualificatif définissant où une classe/interface est accessible

- `public`, partout
- par défaut, dans le paquetage englobant

Qualificatif définissant où une méthode/attribut est accessible

- `public`, partout
- `protected`, dans le paquetage englobant et les classes descendantes
- par défaut, dans le paquetage englobant
- `private`, dans la classe

Avec les règles par défaut

- tout est caché dans le paquetage de définition (modularité)
- interface du paquetage avec l'extérieur explicitement `public`

Autres attributs

Qualificatifs associables aux méthodes

- `static` : la méthode est liée à une classe
  - appel via le nom de la classe
  - pas de `this`
- `final` : méthode non redéfinissable

Qualificatifs associables aux attributs

- `static` : l'attribut est lié à une classe
  - accès via le nom de la classe
  - instance unique, commune à tous les objets de la classe
- `final` : attribut constant