Java EE Spring, prêt à l'emploi

Réflexivité Java

Guillaume Dufrêne – Lionel Seinturier – Julien Wittouck

Université de Lille

Définition

Réflexivité (en anglais reflection)

La capacité d'un programme à se découvrir lui-même

M.C. Escher

Définition

Réflexivité (en anglais reflection)

La capacité d'un programme à se découvrir lui-même

- En Java
  - Découverte de la structure
  - Mécanisme présent dès Java 1.0/1.1 (1996-97)

- Existe aussi dans d'autres langages de programmation

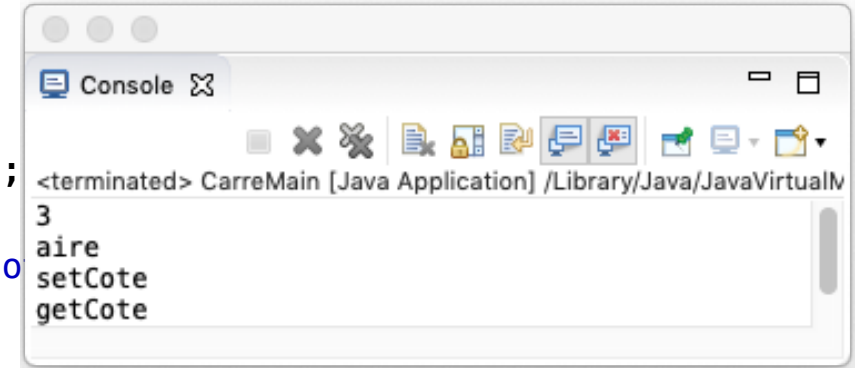
Exemple

```
public class Carre {
	private double cote;
	public Carre() { this.cote = 0.0; }
	public Carre( double c ) { this.cote = c; }
	public double getCote() { return cote; }
	public void setCote( double c ) { this.cote=c; }
	public double aire() { return cote*cote; }
}

Carre monCarre = new Carre(3.0);
```

Exemple

```
public class Carre {
        private double cote;
        public Carre() { this.cote = 0.0; }
        public Carre( double c ) { this.cote = c;
        public double getCote() { return cote; }
        public void setCote( double c ) { this.co
        public double aire() { return cote*cote;
```



Accès à la classe et à ses méthodes

```
Class cl = monCarre.getClass();
Method[] methodes = cl.getDeclaredMethods();
System.out.println(methodes.length);
for (Method method : methodes) {
        System.out.println(method.getName());
}
```

java.lang.reflect.Method

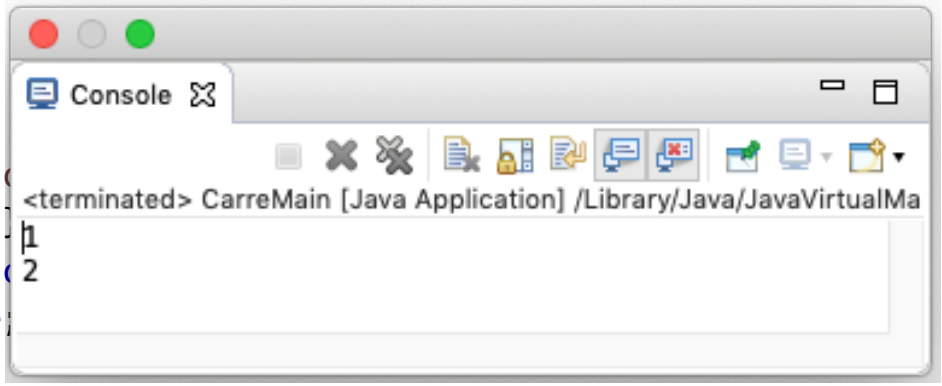
- Accès aux éléments d'une méthode
  - son nom
  - ses paramètres
  - son type de retour
  - les types d'exceptions déclarées (si il y en a)
  - sa visiblité (`public`, `private`, etc.)
  - les annotations
  - etc.

Limitation

- pas d'accès au code de la méthode

java.lang.reflect.Field et Constructor

```
public class Carre {
          private double cote;
          public Carre() { this.cote = 0.0; }
          public Carre( double c ) { this.cote = c
          public double getCote() { return cote; }
          public void setCote( double c ) { this.c
          public double aire() { return cote*cote;
```




```
Class cl = monCarre.getClass();
Field[] fields = cl.getDeclaredFields();
Constructor[] constructors = cl.getDeclaredConstructors();
System.out.println(fields.length);
System.out.println(constructors.length);
```

java.lang.Class

- getDeclared...
  - les éléments déclarés directement par la classe
  - quelle que soit leur visibilité
- getMethods(), getFields(), getConstructors()
  - les éléments, uniquement public, déclarés **et hérités**
- Une méthode particulière
  - `getMethod("setCote",double.class)`
- Les interfaces implémentées (si il y en a)
- La classe parente

Autres concepts

1. Invocation dynamique de méthode
2. Instantiation dynamique de classe

Invocation dynamique de méthode

Cas d'usage

- Invoquer une méthode sans que son nom soit connu a priori

Méthode `invoke` dans `java.lang.reflect.Method`

```
Object invoke( Object obj, Object... args )
```

Exemple

```
Class cl = monCarre.class;
Method m = cl.getMethod("setCote",double.class);
m.invoke(monCarre,4.0);
```

Limitation

- performance

Instantiation dynamique de classe

1. Accéder à un constructeur de la classe
2. Invoquer la méthode `newInstance`

```
Class cl = monCarre.class;
Constructor ctor = cl.getConstructor(double.class);
Object monCarre = ctor.newInstance(42.0);
```

Utilisation de la réflexivité

- Outils de génération de code
- Outils de visualisation, débuggage de code à l'exécution
- Framework de tests (par ex. JUnit)
- Serveurs d'application (par ex. Tomcat)
- Frameworks de développement (par ex. Spring)
- Environnements de développement (par ex. Eclipse, IntelliJ)

- Tout programme qui cherche à automatiser des traitements autour de programmes non connus a priori

En résumé

- Capacité d'un programme Java à découvrir sa structure
  - ses méthodes, ses constructeurs, ses variables, etc.
- Classe `java.lang.Class` donne accès au contenu d'une classe
- Package `java.lang.reflect`
  - `Method, Constructor, Field`
- Possibilité d'invoquer dynamiquement des méthodes
- Possibilité d'instantier dynamiquement des classes

- Les programmes deviennent des données qui peuvent être manipulées par d'autres programmes