

L'instruction try-with-resources

Temps de lecture : 1 minute

Syntaxe de try-with-resources

```
try (var scanner = new Scanner(System.in)) {  
    var saisieEntier = scanner.nextInt();  
    System.out.println(saisieEntier);  
} catch (InputMismatchException ime){  
    ime.printStackTrace();  
}
```

Copier

Les classes implémentant `Closeable` doivent être fermées explicitement par le développeur pour libérer les ressources sous-jacentes qu'elles utilisent. Si elles ne sont pas fermées manuellement, celles-ci sont fermées automatiquement quand le scope dans lequel elles sont définies est résolu (donc à la fin du bloc d'instructions qui les définit).

`Scanner` implémente `Closeable`. Nous verrons d'autres exemples de classes l'implémentant dans le chapitre sur les flux d'entrées / sorties. Pour résumer, ce sont des classes travaillant sur des fichiers, des flux, des connexions, ...

L'intérêt de l'instruction `try-with-resources` utilisée précédemment est de fermer automatiquement les flux définis dans les parenthèses du `try`. Cela permet d'éviter l'utilisation d'une instruction `finally` pour fermer les flux. Le code est ainsi plus simple à écrire et moins risqué.