

Rappels, la clause «**finally**» en Java

Fabrice.Kordon@lip6.fr



Vous avez dit «finally»?

2

Finaliser l'exécution du bloc

- Catch exécuté si l'exception correspondante est levée
- Finally toujours exécuté

Très utile pour «terminer proprement»

- En cas de déroutement
 - ▶ Traitement d'erreur
 - ▶ Autre raison

Ne pas oublier

- Utilisation intensive dans la gestion des «verrous»



Exemple type : manipulation de fichier

```
import java.io.*;

public class ExempleException3 {

    public static void main(String []args) {
        File f = null;
        BufferedReader bf = null;
        String buffer;

        try { // tenter l'ouverture
            f = new File("monFichier.txt");
            bf = new BufferedReader (new FileReader (f));
            do { // boucle de lecture
                buffer = bf.readLine();
                System.out.println("On a lu " + buffer);
            } while (buffer != null);
        }
        catch (FileNotFoundException exc) {
            System.err.println ("Fichier " + f + " non trouvé");
        }
        catch (IOException exc) {
            System.err.println ("Erreur d'E/S sur "+f+" (" + exc.getMessage() +)");
        }

        finally {
            System.err.println("dans finally");
            if (bf != null) {
                try {
                    bf.close();
                }
                catch (IOException exc) {
                    System.err.println ("Close impossible");
                }
            }
        }
    }
}
```

Bloc «finally»

Permet ici de toujours fermer le fichier

Exécution...

4

```
$ ls monFichier.txt
ls: monFichier.txt: No such file or directory
$ java ExempleException3
Fichier monFichier.txt non trouvé
dans finally ←
$ mv xxx.txt monFichier.txt
$ cat monFichier.txt
ligne 1
ligne 2
dernière ligne
$ java ExempleException3
On a lu ligne 1
On a lu ligne 2
On a lu dernière ligne
On a lu null
dans finally ←
```