

IFT1025 automne 2009

Miklós Csűrös

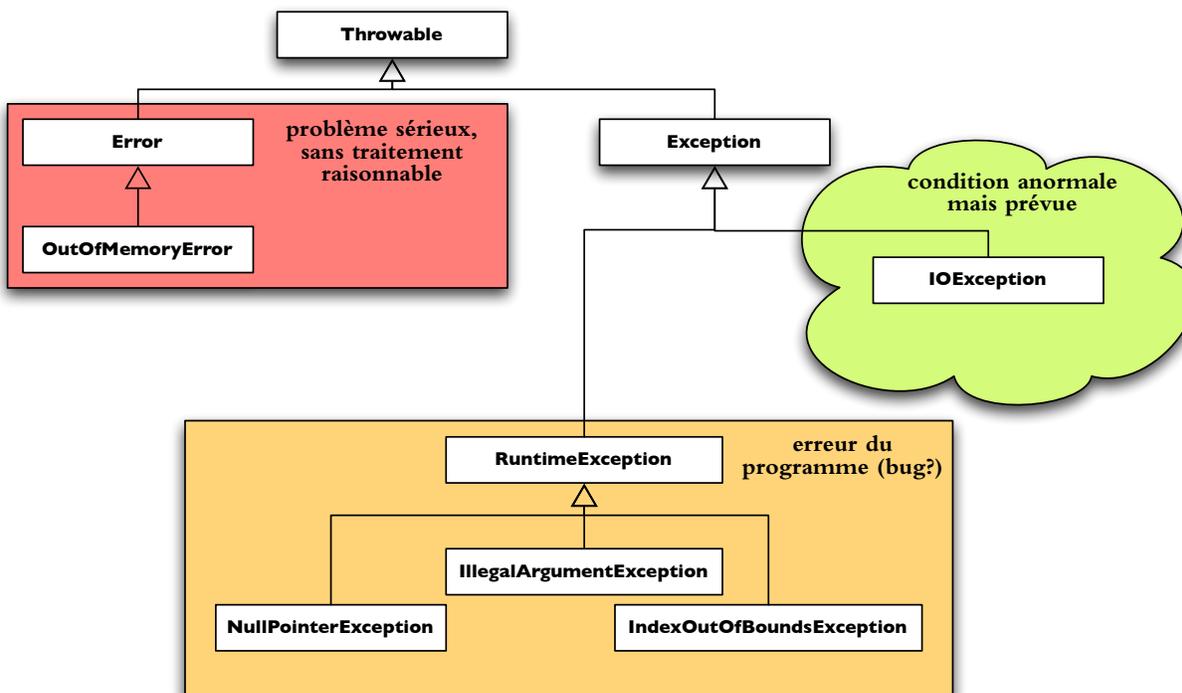
22 septembre 2009

Exceptions

Erreur vs. exceptions :

- ★ erreur logique impossible à réparer (p.e., «out of memory»)
- ★ événement imprévu (p.e., argument null)
- ★ événement anormal mais prévoyable (p.e., «file does not exist»)

En Java, il existe une classe pour modéliser de telles situations : Throwable.



- ★ pour signaler une situation anormale, faire `throw XXXX`
- ★ lorsqu'une exception ou erreur est jetée, l'exécution normale est interrompue
- ★ le signal est transmis vers la méthode appelant
- ★ une **Exception** jetable doit être déclarée dans la signature : `public void write() throws IOException` (sauf les **RuntimeExceptions**)

Traitement d'exceptions : `try{...} catch(Exception1 E){...} catch(Exception2 E){...} ...`

```
public void readFile(String file_name)
{
    try
    {
        BufferedReader BR = new BufferedReader(new FileReader(file_name));
        while(true) // condition d'arrêt dans la boucle
        {
            String line = BR.readLine();
            if (line == null) break;
            // parsing
        }
    } catch (FileNotFoundException E) // venant de new FileReader()
    {
        System.err.println("Pas de fichier "+file_name);
        // E.printStackTrace(System.err); // pour voir tous les détails
        System.exit(99);
    } catch (IOException E) // venant de BR.readLine()
    {
        System.err.println("Erreur de lecture ");
        System.exit(99);
    }
    ...
}
```

Conseils pratiques :

- ★ jeter une exception aussitôt que possible s'il y a un problème (au lieu d'essayer de le fixer ou retourner une «valeur d'erreur» dans la méthode)
- ★ capturer l'exception aussi tard que possible, seulement si on est capable de gérer la situation
- ★ jeter des exceptions aussi spécifiques que possible — définir ses propres sous-classes de `Exception`
- ★ ne pas utiliser `try ... catch` pour implémenter la logique normale d'exécution (p.e., attendre l'exception quand la fin du fichier est atteinte pour terminer une boucle lisant un fichier)