

## Exercice 4 : Exception

### Objectifs de l'exercice

- Créer ses propres exceptions contrôlées et non contrôlées
- Synthétiser les principes d'utilisation clés de ce concept langagier

### Question 1

Créer deux exceptions `My1Exception` (contrôlée) et `My2Exception` (non contrôlée). Il suffit pour cela de créer 2 classes, sous-classes directes respectivement de la classe `Exception` et `RuntimeException`.

Ces deux classes ne définissent qu'une seule méthode : un constructeur très simple qui ne fait appel qu'au constructeur de sa super-classe et qui se conforme au format exemple ci-dessous.

```
public MonException (String s) {  
  
    super(s) ;  
  
}
```

### Question 2

Créer une classe `TestException` qui va contenir un `main()` et uniquement des méthodes d'instances (pas de méthodes `static`).

Ces méthodes d'instances doivent pour certaines générer l'une ou l'autre des deux exceptions créées ci-dessus et pour d'autres les « catcher » ou les « passer » au niveau d'appel supérieur.

Le challenge de l'exercice est de faire en sorte que le code de cette classe mette en valeur toutes les situations possibles d'utilisation des exceptions contrôlées et non contrôlées. Les codes qui ne compilent pas seront placés en commentaire en indiquant la raison de cette non compilation.

En particulier ce code doit montrer :

- la levée d'une exception (`throw new MonException (« ... »)`), le catch d'une seule exception ou de plusieurs (`try{...}catch(...) {} catch (...) {...}`) et le fait qu'un bloc `finally` est toujours exécuté (qu'une exception ait été levée ou non).
- Le fait qu'une exception non contrôlée n'a pas d'obligation d'être catchée (compilation du code même sans catch) ni d'être « passée » (caractère optionnelle de la directive `throws MonException1, ...` en signature d'une méthode) mais qu'elle peut l'être malgré tout (catchée ou passée).
- Le fait qu'une Exception contrôlée doit toujours répondre à la règle du « catchée ou passée » sous peine de non compilation du bloc de code qui peut faire apparaître ce type d'exception.
- Donner des exemples d'exceptions passées ou catchées sur zéro, un ou deux niveaux d'appel : ainsi un exemple à deux niveaux d'appel est celui d'une méthode `m()` qui appelle `n()` qui appelle `p()` avec `p()` qui lève une exception, `n()` qui la passe et `m()` qui la catche.

**Soyez exhaustif et créatif !**