

## Strategy

### Exercice 1 (Écrire la date d'aujourd'hui)

Il existe à travers le monde différente manière d'écrire une date. Par exemple pour le 21 aout 2011, on écrira :

- En France, au Royaume-Uni : 21/08/2011
- Aux États-Unis : 8/21/2011
- En suivant la notation standard internationale ISO 8601 : 2011-08-21

On souhaite écrire un programme qui affiche la date d'aujourd'hui. Le but de cet exercice sera d'utiliser le pattern Strategy pour afficher la date à la manière UK, US ou internationale.

1. Faites sur papier le diagramme UML modélisant ce programme.
2. Implémentez votre modèle à l'aide des indications suivantes :

```
1 // Bibliothèques à importer pour les diverses fonctionnalités
2 // concernant les dates.
3 import java.text.*;
4 import java.util.*;
5
6 // Obtenir la date d'aujourd'hui
7 Date date = new Date();
8
9 // Format de la date UK
10 DateFormat formatUk =
11     DateFormat.getDateInstance( DateFormat.SHORT, Locale.UK );
12
13 // Format de la date US
14 DateFormat formatUs =
15     DateFormat.getDateInstance( DateFormat.SHORT, Locale.US );
16
17 // Format de la date internationale
18 DateFormat formatIntl = new SimpleDateFormat( "yyyy-MM-dd" );
19
20 // Obtenir la string d'une date selon le format DateFormat
21 String string = dateFormat.format( date );
```

### Exercice 2 (Cryptage de données)

En vous aidant du pattern Strategy, écrire les classes d'un programme cryptant des données (disons une String), avec la possibilité de choisir son système de cryptage (md5, sha256, sha512). Attention, on ne demande pas ici d'implémenter ces différents systèmes de cryptage.

Implémentez ceci en utilisant la classe MessageDigest (<http://docs.oracle.com/javase/6/docs/api/java/security/MessageDigest.html>).