

## Programmation Orientée Objet

### Trame de cours-Démo : e-uml2 et héritage

Durant cette séance, nous allons développer un exemple autour de l'héritage et apprendre à utiliser le plugin euml2. Nous nous baserons sur une implémentation existante de Coffres et Pierres disponible sur celene.

#### Exercice 1 : Étude du code fourni et génération de diagramme UML

- Jetons un coup d'oeil au code de départ
- Générons un diagramme UML lui correspondant
- Lançons le main, pour voir

#### Exercice 2 : Protégeons un peu notre coffre...

- en plus de **Coffres** classiques, on souhaite pouvoir utiliser des coffres avec code.
- Comment le faire ?
  - Faisons-le (en jouant un peu).

#### Exercice 3 : Et l'expertise dans tout ça ?

L'expertise d'une pierre n'est pour le moment pas implémentée. Pour simplifier, on peut se dire qu'il existera plusieurs type de pierres (diamant, émeraude, topaze) le prix dépendant de sa nature et de son volume.

- Comment pourrait-on le faire ?
- Si l'on impose de créer une classe pour chaque type de pierre, comment modifier le diagramme UML précédent ?
- Faisons-le ensemble.

#### Exercice 4 : Il n'y a pas de petit profit...

Et si nous vendions nos coffres et pierres ? Considérons un client capable d'acheter tout objet (du moment qu'il a une valeur) dont la valeur n'excède pas son budget. Comment modifier notre conception ?