

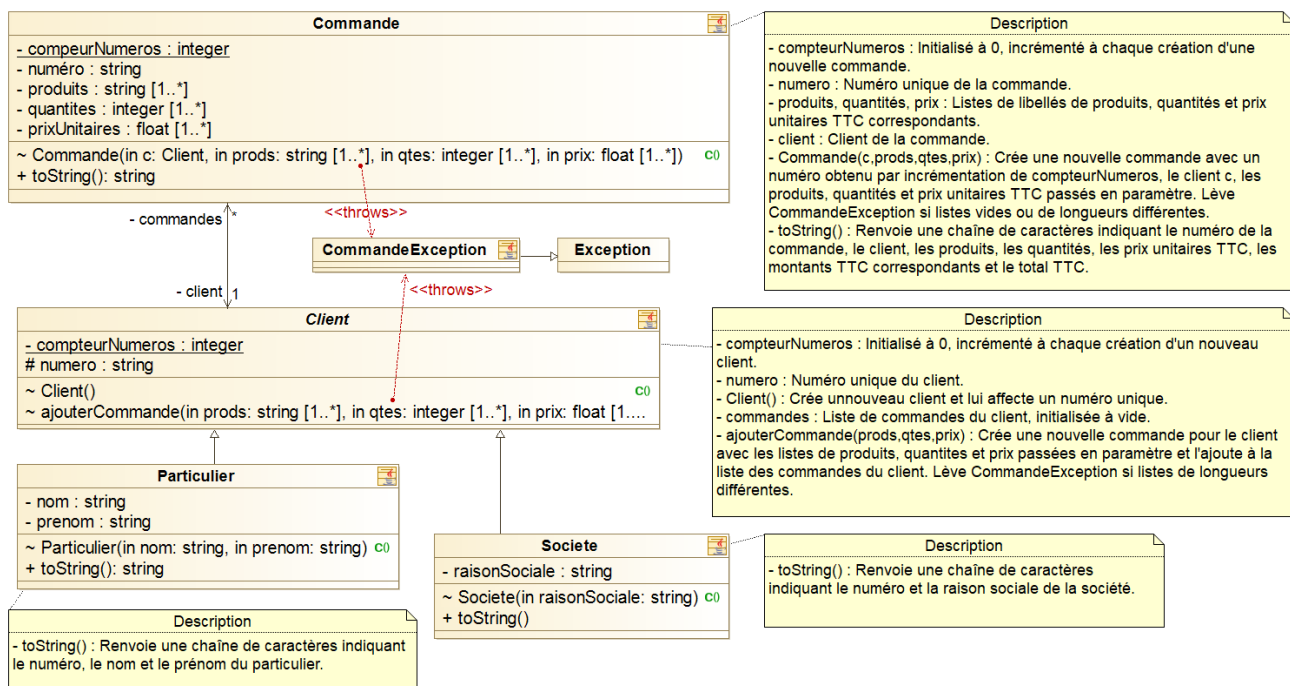
## Qualité de Développement – TP1

L'objectif de ce TP est la prise en main du logiciel de modélisation UML Modelio et de l'environnement de développement Java / Netbeans .

### 1. Prise en main du logiciel de modélisation UML Modelio

Le but ici est de créer sous Modelio le diagramme de classes ci-dessous.

- Ouvrir **Modelio** et créer un nouveau projet
  - o Choisir **File/Switch workspace...** puis sélectionner un dossier et valider.
  - o Choisir **File/Create projet...** puis nommer le projet, cocher la case « **Java projet** » et valider.
  - o Dans le volet navigateur de projets, déployer l'arborescence du projet jusqu'au dossier de même nom que le projet. Faire un clic-droit sur ce dossier et choisir **Create diagram...** Sélectionner **Class diagram** et valider.
  - o A l'aide de la palette qui s'affiche, définir chaque élément du diagramme.
- Générer le squelette de code Java à partir du diagramme
  - o Dans le volet navigateur de projets, faire un clic-droit sur le dossier du projet et choisir **Java Designer/Générer**.
  - o Ouvrir les fichiers créés dans le dossier <nom du projet>/src dans le workspace et examiner le code généré.



### 2. Prise en main de l'environnement de développement Java / Netbeans

Le but est de réaliser le codage Java du diagramme précédent en complétant le code généré sous Modelio.

- Ouvrir **NetBeans 8.2** et créer un projet de type Java Application :
  - o Choisir **File/New project ...** puis sélectionner la catégorie **Java/Java Application**
  - o Cliquer sur **Next** puis nommer le projet et définir son emplacement et valider.
- Copier les fichiers java générés sous Modelio dans le dossier **Source Packages**.
- Retoucher le code de manière à supprimer toutes les erreurs signalées par le compilateur (ajouter des "bouchons" pour les retours des méthodes).
- Compléter le code de toutes les classes.
- Coder une classe de test avec une méthode *main()* permettant de :
  - o Créer un client instance de la classe *Particulier* et un client instance de la classe *Societe*.
  - o Créer deux commandes pour chacun de ces clients.
  - o Afficher ces commandes sur la console de sortie.