

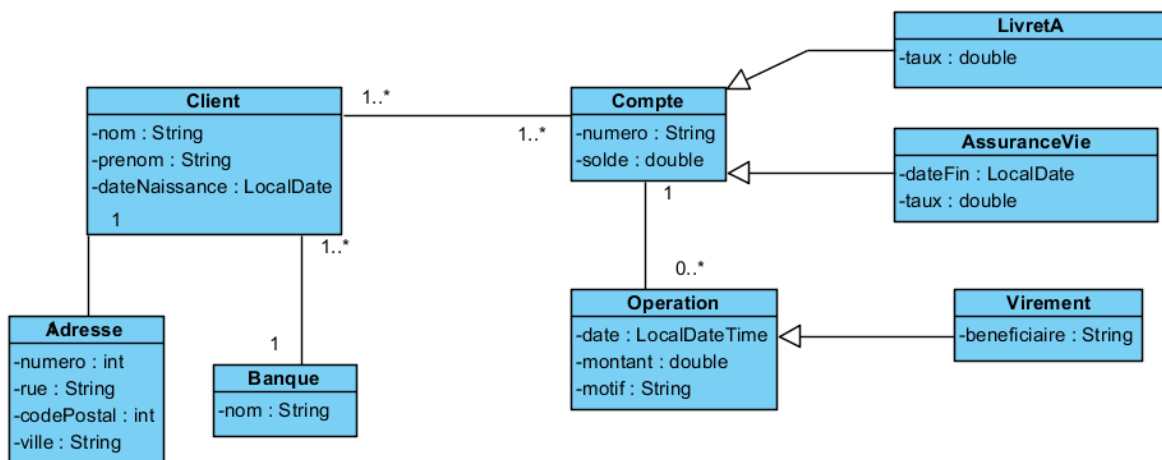
TP n°4 – Modélisation objet avec JPA

Objectifs

Se renforcer dans la modélisation JPA et notamment l'utilisation des annotations relationnelles.

Description

- Voici le modèle UML d'une application bancaire. Nous allons créer les entités JPA et essayer de générer les tables à partir des entités.



- Utilisez le projet existant mais créez un nouveau package
- Créez une nouvelle base de données appelée **banque**
- Créez une **nouvelle persistence-unit** dans **persistence.xml** pour y accéder
- Positionner la propriété suivante dans persistence.xml pour régénérer automatiquement le schema à chaque redémarrage de l'application:

```
<property name="javax.persistence.schema-generation.database.action" value="drop-and-create"/>
```

- Créer les entités JPA suivantes et mettez en place les annotations relationnelles :
 - o Banque
 - o Compte
 - o Operation
 - o Client
 - Adresse (Embeddable)
- Les tables seront générées à partir des relations définies dans votre modèle métier
- Insérer en base de données des instances des différents objets en utilisant l'EntityManager

Quelques propriétés utiles pour le persistence.xml :

- Cette propriété ne crée que les nouvelles tables après chaque redémarrage:

```
<property name="hibernate.hbm2ddl.auto" value="update"/>
```
- Cette propriété désactive l'auto-détection des entités dans les packages du projet:

```
<property name="hibernate.archive.autodetection" value="" />
```

Attention cette propriété s'utilise avec la balise class pour lister les entités manuellement (cf exemple ci-dessous).

```

<persistence-unit name="banque" transaction-type="RESOURCE_LOCAL">
    <provider>org.hibernate.jpa.HibernatePersistenceProvider</provider>
    <class>banque.entite.Banque</class>
    <class>banque.entite.Client</class>
    <class>banque.entite.Compte</class>
    <class>banque.entite.LivretA</class>
    <properties>
        <!-- desactiver l'autodetection des classes de type @Entity -->
        <property name="hibernate.archive.autodetection" value="" />

        <!-- Activation de la génération des tables manquantes uniquement -->
        <property name="hibernate.hbm2ddl.auto" value="update"/>

        <!-- Drop and create à chaque redémarrage -->
        <property name="javax.persistence.schema-generation.database.action"
            value="drop-and-create"/>

        <!-- Affichage des requêtes SQL -->
        <property name="hibernate.show_sql" value="true" />

        <!-- Paramètres d'accès à la BDD -->
        <property name="javax.persistence.jdbc.url"
            value="jdbc:mariadb://localhost:3306/banque" />
        <property name="javax.persistence.jdbc.user" value="root" />
        <property name="javax.persistence.jdbc.password" value="" />
        <property name="javax.persistence.jdbc.driver"
            value="org.mariadb.jdbc.Driver" />
    </properties>
</persistence-unit>

```