

## TP N°04 : Concepts UML vers Déclaration SQL (Suite)

### I. Diagramme de classe UML vers déclaration SQL :

#### I.1. Classe avec attribut :

<div style="border: 1px solid black; padding: 5px; margin-bottom: 5px;"> <b>relation_A</b> </div> <div style="border: 1px solid black; padding: 5px;">                 att1: String                  att2: Integer             </div>	Chaque classe devient une relation; s'il n'y a pas d'attribut, un numéro est créé en tant qu'identifiant primaire :  <pre>create table relation_A (     num_relation_A integer primary key,     att1 text,     att2 integer);</pre>
---	---

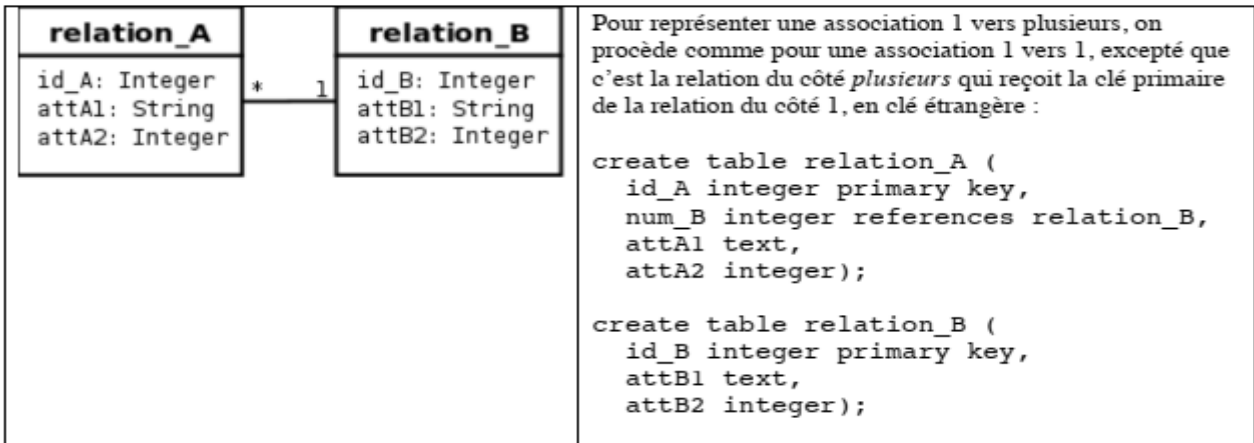
#### I.2. Association 1 vers 1 :

<div style="display: flex; align-items: center; gap: 20px;"> <div style="border: 1px solid black; padding: 5px; text-align: center;"> <b>relation_A</b>                  id_A: Integer                  attA1: String                  attA2: Integer             </div> <div style="text-align: center;">                 1 — 1             </div> <div style="border: 1px solid black; padding: 5px; text-align: center;"> <b>relation_B</b>                  id_B: Integer                  attB1: String                  attB2: Integer             </div> </div>	Pour représenter une association 1 vers 1 entre deux relations, la clé primaire de l'une des relations doit figurer comme clé étrangère dans l'autre relation :  <pre>create table relation_A (     id_A integer primary key,     attA1 text,     attA2 integer);  create table relation_B (     id_B integer primary key,     num_A integer references relation_A,     attB1 text,     attB2 integer);</pre>
--	---

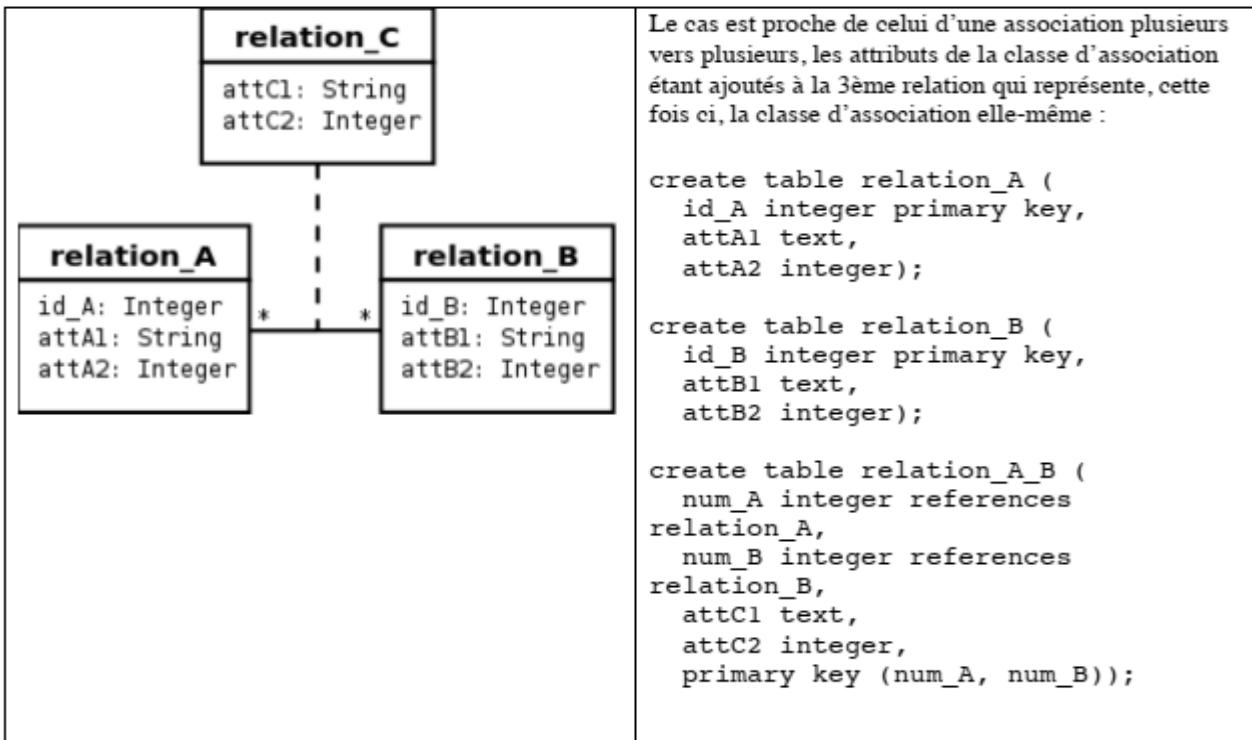
#### I.3. Association \* vers \* :

<div style="display: flex; align-items: center; gap: 20px;"> <div style="border: 1px solid black; padding: 5px; text-align: center;"> <b>relation_A</b>                  id_A: Integer                  attA1: String                  attA2: Integer             </div> <div style="text-align: center;">                 * — *             </div> <div style="border: 1px solid black; padding: 5px; text-align: center;"> <b>relation_B</b>                  id_B: Integer                  attB1: String                  attB2: Integer             </div> </div>	Pour représenter une association du type plusieurs vers plusieurs, il faut introduire une nouvelle relation dont les attributs sont les clés primaires des relations en association et dont la clé primaire est la concaténation de ces deux attributs :  <pre>create table relation_A (     id_A integer primary key,     attA1 text,     attA2 integer);  create table relation_B (     id_B integer primary key,     attB1 text,     attB2 integer);  create table relation_A_B (     num_A integer references relation_A,     num_B integer references relation_B,     primary key (num_A, num_B));</pre>
--	---

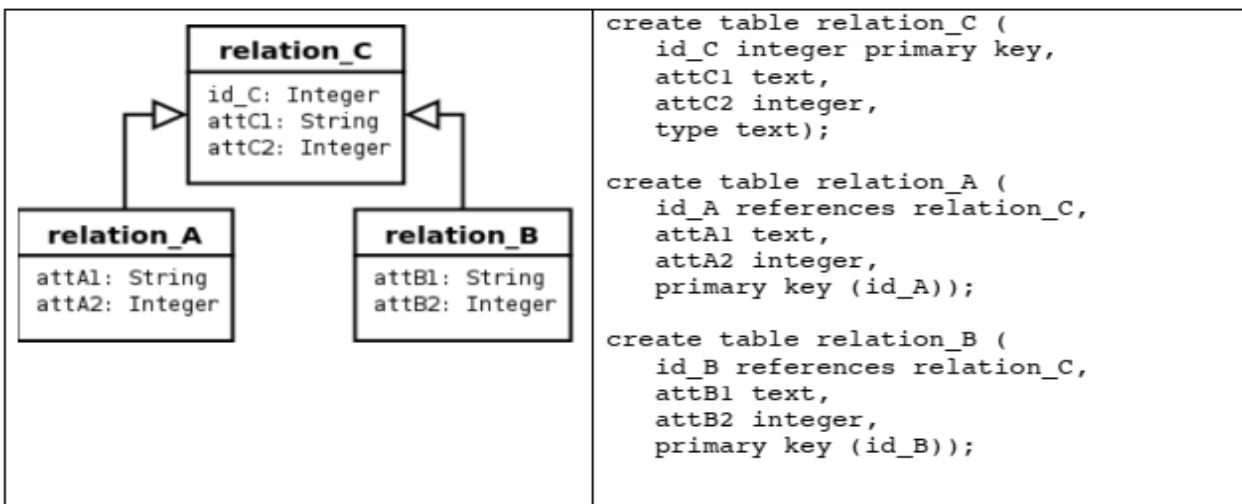
#### I.4. Association 1 vers \* :



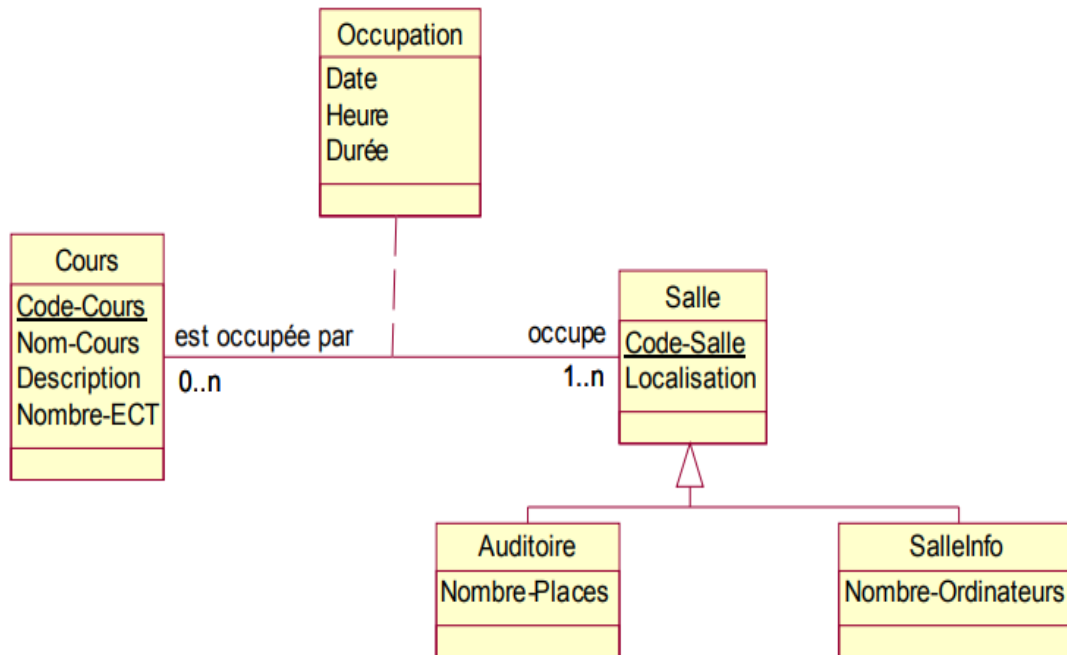
#### I.4. Classe association :



#### I.5. héritage :



## II. Exercice :



- 1- Construire le modèle relationnel correspondant au diagramme de classe ci-dessus
- 2- Créer la base de données MySQL correspondante au modèle relationnel