# RD – ECF – Conception et modélisation de base de données relationnelle + SQL

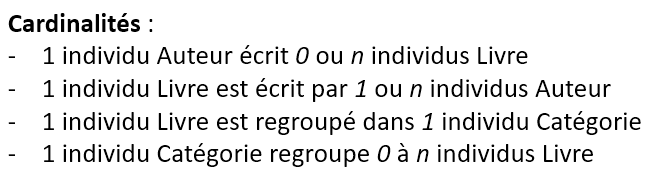
## Livrables attendus

* Le modèle conceptuel de données (image)
* Le modèle physique de données (image)
* Le DDL
* Le DML
* Un fichier avec les requêtes SQL
* Le tout dans un repository GitHub, soigner le nommage des fichiers
* Communiquer le lien du repository
* Note : les images peuvent être regroupées dans un PPT par exemple, choix libre pour le ou les outils de modélisation ainsi que pour le SGBDR (MySQL ou PostgreSQL)

## Consignes

Conseil : réaliser d’abord de bout en bout la partie 1 avant d’attaquer la partie 2.

Dans le MCD ajouter des notes contenant pour chaque relation la formulation dans les deux sens de la relation reprenant le nom des entités, le verbe et les cardinalités, exemple :



Prendre soin dans le MPD des types de colonnes, choisir le cas échéant des longueurs et précisions pertinentes, elles ne sont pas indiquées volontairement.

Dans le DML, veiller à ne pas insérer des clefs primaires « en dur ».

Utiliser les alias de tables et de colonnes dans les requêtes SQL.

### Partie 1

#### Conception et modélisation

Concevoir et modéliser une base de données pour stocker des informations sur les développeurs d’une entreprise et des technologies informatiques (langages, frameworks…).

Un développeur à une identité avec son nom et son prénom, une date de naissance, une date d’entrée en entreprise, un matricule (identifiant dans l’entreprise) et une technologie de prédilection. Toutes les informations sur un développeur sont obligatoires.

Nous souhaitons également stocker, en plus de la technologie de prédilection, les autres technologies que chaque développeur apprend. Sachant qu’un développeur peut apprendre aucune ou plusieurs autres technologies.

Une technologie a un code unique arbitraire composé d’un préfixe « CD\_ » suivi d’un incrément, chaque technologie a un nom en français et en anglais. Toutes les informations sur une technologie sont obligatoires.

Note : on part du principe que l’applicatif ne permettrait pas d’avoir un doublon entre la technologie de prédilection et les autres technologies apprises pour un même développeur.

#### DML

Le script doit insérer 3 développeurs et 4 technologies. Un développeur apprend 2 technologies, un autre en apprend 3 dont 1 seule en commun avec le premier développeur, et le dernier développeur n’en apprend aucune.

#### Requêtes SQL

Note : id en dur et usage du LIMIT non autorisés. Si on ajoutait des développeurs et des technologies, les requêtes retourneraient de bons résultats sans avoir à les modifier.

* Récupérer la liste des développeurs (matricule, nom, prénom et le nom français de la technologie de prédilection) qui n’apprennent aucune autre technologie que celle de prédilection
* Récupérer la liste des développeurs (matricule, nom, prénom, et le nombre de technologies apprises), donc qui en apprennent au moins une
* Récupérer la liste des développeurs (matricule, nom, prénom, et le nom anglais de la technologie apprise) qui apprennent au moins une technologie en commun

### Partie 2

#### Conception et modélisation

Ajouter le concept de « tech lead ». Un développeur peut être le tech lead d’aucun ou plusieurs autres développeurs. Un développeur peut avoir aucun ou un seul « tech lead ». Un « tech lead » peut avoir un « tech lead ».

#### DML

Modifier le DML existant pour introduire des « tech leads » le premier développeur (qui apprend 2 technologies) est le « tech lead » des 2 autres et le deuxième développeur (qui apprend 3 technologies) est le « tech lead » du troisième.

#### Requête SQL

* Récupérer la liste des développeurs « tech lead » (nom, prénom et le nombre de développeurs dont ils sont « tech lead »), les développeurs qui ne sont pas « tech lead » ne doivent pas être dans le résultat de la requête