

# TD11 : Révisions et Modèle Conceptuel de Données

---

## Rappels

### NULL

Un attribut est par défaut forcément non NULL. Si l'on rajoute NULL à sa définition, alors l'attribut peut être NULL. Le mot clef NULL veut dire que l'attribut n'a pas été spécifié. Par exemple on utilise NULL si on a un attribut Licence pour des Etudiants, qui est de type texte, pour pouvoir préciser qu'un étudiant n'a pas encore obtenu de Licence. On mettra Licence à 'Géographie' pour préciser que l'étudiant a une licence de Géographie. Pour tester si un champ est NULL on utilise IS NULL et pas = NULL.

La requête suivante ne marche pas :

```
SELECT * FROM Table  
WHERE Attribut = NULL;
```

Il faut utiliser :

```
SELECT * FROM Table  
WHERE Attribut IS NULL;
```

### UNIQUE

Un attribut peut être UNIQUE. Cela signifie que dans la table, il ne doit pas y avoir deux fois la même valeur. C'est comme une clef primaire mais on l'utilise quand on pas besoin de référencer les lignes par cet attribut.

### BOOL ou BOOLEAN

Le type BOOL ou BOOLEAN stocke une information valant TRUE ou FALSE. On teste cette valeur ainsi :

```
SELECT * FROM Table  
WHERE Attribute
```

```
SELECT * FROM Table  
WHERE Attribute = FALSE
```

## Exercice 1

1. Recréer en SQLite la base de données dont les schémas et les données des tables sont donnés en annexe ?

2. Quel requête faut il faire pour que le dresseur 1 soit aussi maître de Montpellier ? Que va t-il se passer ?

3. Mettre à jour la base de données :

- Bulbizare a une force de 11
- Jo a un deuxième pokemon qui est Florizare
- Il n'y a plus de maître de Montpellier
- Jo est maître de Bordeaux

4. Questions sur la structure de la base :

- Combien de Pokemons peut avoir un Dresseur ?
- Est ce qu'un Dresseur peut avoir plusieurs fois le même Pokemon ? Sinon que changer pour que cela puisse être le cas ?
- Est ce qu'un Dresseur peut être maître de plusieurs Arenes ? Sinon que changer pour que cela puisse être le cas ?
- Combien de Dresseurs peuvent être maîtres d'une même Arene ?

## Exercice 2

Créer des bases de données en SQLite pour les situations suivantes :

- des trains sont dans des gares
- des trains sont dans des gares ou nul part
- des trains sont dans des gares mais une gare ne peut contenir qu'un train
- des trains sont dans gares mais ou nul part mais une gare ne peut contenir qu'un train

## Exercice 3

Dans la base de données de l'exercice 1. Donner les requêtes SQL, la formule en algèbre relationnelle associée et le résultat de la requête pour les questions suivantes.

1. Quels sont les noms des Pokemons de Force  $\geq 20$  ?

2. Quels sont les noms des Pokemons du Dresseur numéro 0 ?

3. Quel est le nom de l'Arène dont le Dresseur numéro est maître ?

4. Quels sont les noms des pokemons qui ont la force maximale ? (Ne pas donner la formule en algèbre relationnelle pour cette question)

5. Combien de fois chaque pokemon est-il utilisé par des dresseurs ? On donnera l'Id et le nombre de fois. (Ne pas donner la formule en algèbre relationnelle pour cette question)

## Modèle conceptuel des données

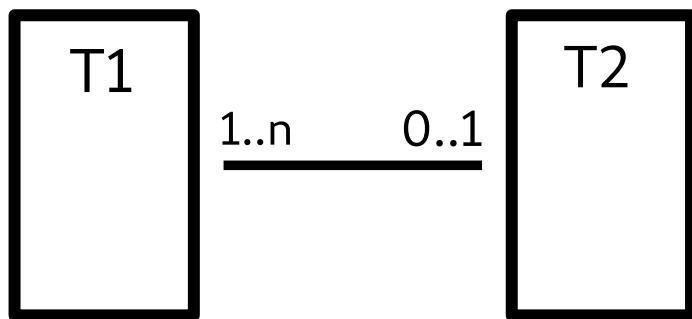
Le modèle conceptuel des données est constitué avec

- des entités (des boîtes avec des attributs)
- des associations entre les entités et ses cardinalités

Les entités vont devenir des tables et certaines associations vont devenir des tables lorsque l'on va créer une base de données à partir du modèle. Les cardinalités sont définies ainsi : étant donné une entité T1 et une entité T2, la cardinalité de T1 vers T2 est le nombre de lignes de T2 que peut référencer une ligne de T1 : plus exactement le nombre minimal et le nombre maximal. S'il n'y a pas de nombre maximal, on met un n (ou \* des fois) qui correspondrait mathématiquement à l'infini.

Par exemple dans l'association suivante, T1 peut référencer plusieurs lignes de T2 mais au moins 1. T2 peut référencer au plus 1 ligne de T1.

Des fois on peut mettre un titre sur l'association.



1. Dresser le MCD de l'exercice 1.

2. Dresser les MCD de l'exercice 2.