Données 25 – SQL JOIN ON

I – Principe des jointures

Le principe général est de créer des n-uplets en récupérant les attributs de deux ou plusieurs tables.
On parvient à faire les jonctions entre les informations en reliant un n-uplet ayant une valeur de clé étrangère et en allant chercher le n-uplet qui possède un clé primaire ayant cette même valeur..

II – Ecriture SQL d'une jointure

2.1 Exemple avec des attributs ayant des noms différents

Imaginons :

- une relation **livre** ayant une clé étrangère **fk_auteur** et un attribut **titre**. **livre (id, titre, fk_auteur)**
- une relation **auteur** ayant une clé primaire **id** et un attribut **nom**. **auteur (id, nom)**

La requête suivante permet de rajouter les n-uplets d'une autre table en cherchant clé primaire = clé étrangère
Il est en réalité inutile de noter les noms des tables mais c'est pratique pour être certain de ne pas se tromper.

```
SELECT titre, nom                              ← ici pas d'ambiguïté sur les attributs
FROM livre
JOIN auteur ON auteur.id = livre.fk_auteur     ← on place toujours la notation précise ici
```

2.2 Exemple avec des attributs ayant des noms similaires : notation point

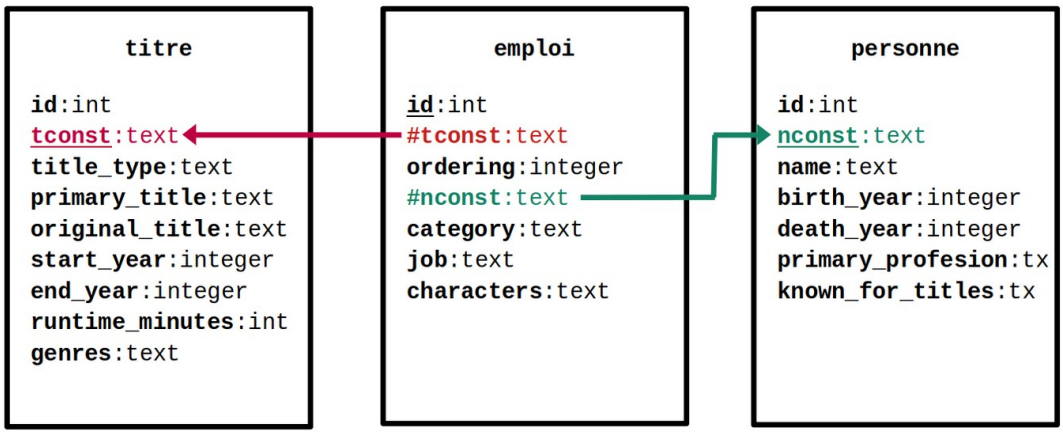
Imaginons que l'attribut **nom** soit un attribut dans les deux tables :

- une relation **livre** ayant une clé étrangère **fk_auteur** et un attribut **nom**. **livre (id, nom, fk_auteur)**
- une relation **auteur** ayant une clé primaire **id** et un attribut **nom**. **auteur (id, nom)**

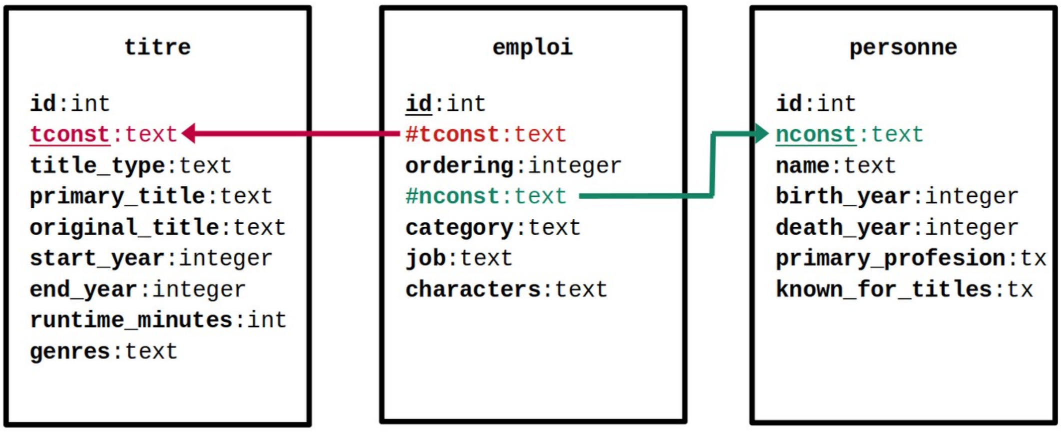
La requête permet de rajouter les n-uplets d'une autre table en cherchant clé primaire = clé étrangère

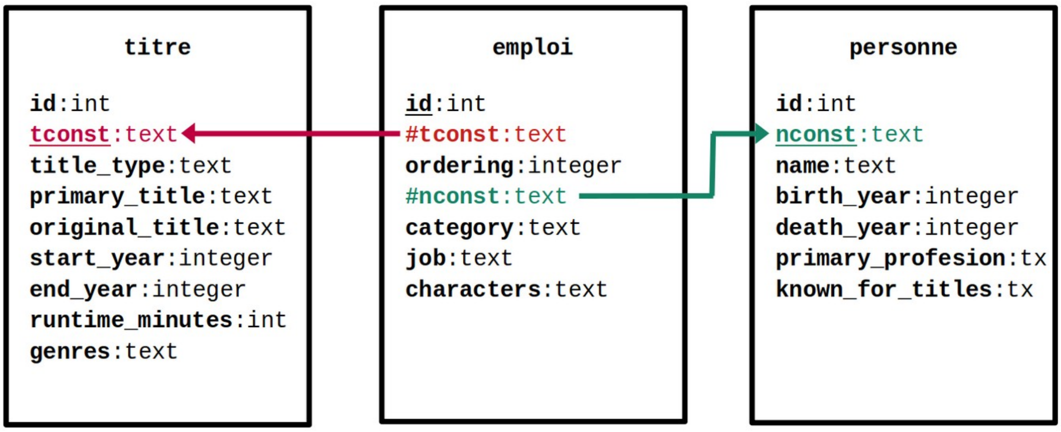
```
SELECT livre.nom, auteur.nom                   ← sans cette notation, difficile de savoir quel nom
FROM livre
JOIN auteur ON auteur.id = livre.fk_auteur
```

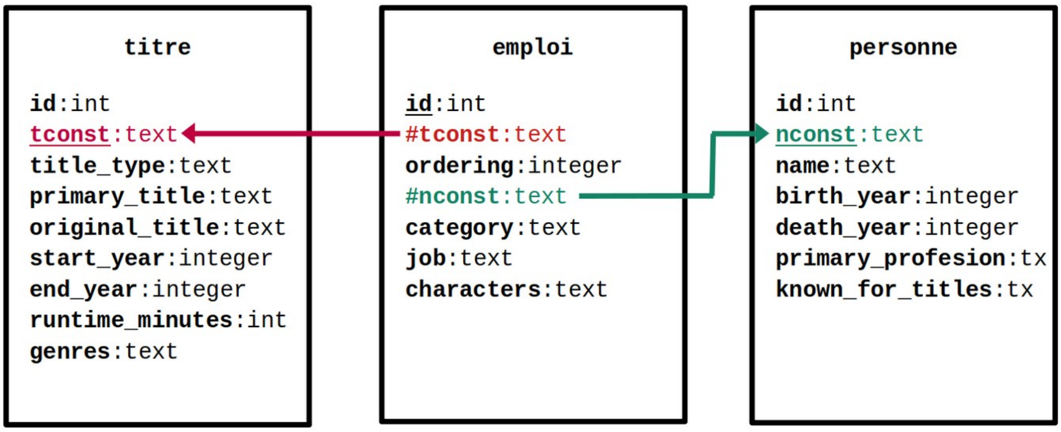
2.3 Double jonction (ou triple ou plus encore...)

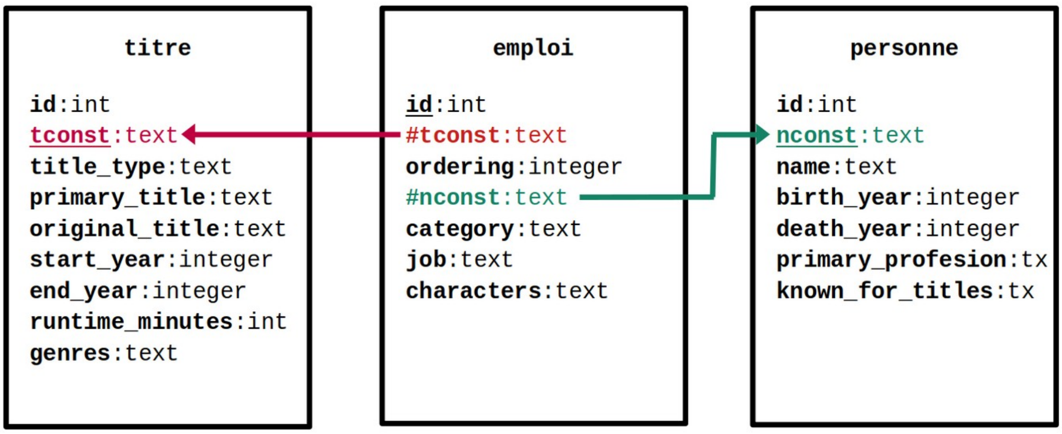



```
SELECT titre.primary_title, personne.name, emploi.category
FROM emploi
JOIN personne ON emploi.nconst = personne.nconst
JOIN titre ON emploi.tconst = titre.tconst
```


titre
id:int
tconst:text
title_type:text
primary_title:text
original_title:text
start_year:integer
end_year:integer
runtime_minutes:int
genres:text
emploi
id:int
#tconst:text
ordering:integer
#nconst:text
category:text
job:text
characters:text
personne
id:int
nconst:text
name:text
birth_year:integer
death_year:integer
primary_profesion:tx
known_for_titles:tx


titre
id:int
tconst:text
title_type:text
primary_title:text
original_title:text
start_year:integer
end_year:integer
runtime_minutes:int
genres:text
emploi
id:int
#tconst:text
ordering:integer
#nconst:text
category:text
job:text
characters:text
personne
id:int
nconst:text
name:text
birth_year:integer
death_year:integer
primary_profesion:tx
known_for_titles:tx


titre
id:int
tconst:text
title_type:text
primary_title:text
original_title:text
start_year:integer
end_year:integer
runtime_minutes:int
genres:text
emploi
id:int
#tconst:text
ordering:integer
#nconst:text
category:text
job:text
characters:text
personne
id:int
nconst:text
name:text
birth_year:integer
death_year:integer
primary_profesion:tx
known_for_titles:tx


titre
id:int
tconst:text
title_type:text
primary_title:text
original_title:text
start_year:integer
end_year:integer
runtime_minutes:int
genres:text
emploi
id:int
#tconst:text
ordering:integer
#nconst:text
category:text
job:text
characters:text
personne
id:int
nconst:text
name:text
birth_year:integer
death_year:integer
primary_profesion:tx
known_for_titles:tx