Normalisation des bases de données

Séparation des relations

i Il est important de ne pas multiplier les mêmes informations.
On doit pour cela créer des relations pour chaque groupe d'information et trouver un lien qui permet de relier entre-elles. C'est le rôle de ce que l'on appelle les `clés`.

• Formes normales

Pour s'assurer qu'une base de données est bien structurée, on peut s'appuyer sur des règles de Normalisation. Dans ce cours nous verrons la première forme normale.

Première forme normale

- Éliminer la répétition de groupes dans les différentes tables.
- Créer une table distincte pour chaque jeu de données connexes.
- Identifier chaque jeu de données connexes à l'aide d'une clé primaire.

• Clés primaires

Une `clé primaire` a deux qualités :

- Elle permet d'identifier de manière unique les différents enregistrements qui composent une table,
- Elle facilite la mise en relation d'une table avec une autre.

Définition

La **clé primaire** est un champ particulier qui permet d'établir des liaisons entre tables.
Elle peut :

- être liée à un champ naturel (un nom par exemple)
- être liée à un champ artificiel (un id par exemple)
- être cotermmposée (liée à plusieurs champs)

⚠ Elle ne peut pas prendre deux fois la même valeur dans deux enregistrements différents.

• Clés étrangères

Définition

Une **`clé étrangère`** est un attribut qui correspond à la clé primaire d'une autre table.
Elle permet de faire le lien entre deux enregistrements de la base de données répartis dans deux relations différentes.

Jointure de table

Si l'infotmation est séparée, il faudra trouver un moyen de la regrouper. C'est le rôle de l'instruction **`JOIN`**

SELECT ... FROM ... JOIN ... ON ...

Exemple :

```
SELECT * FROM table1 JOIN table2 ON table1.attribut1 = table2.attribut2
```

Exemple

On dispose de 2 relations : **`chansons`** et **`artistes`**.

chansons	
Titre	Id_chanteur
Redemption Song	02
That Day Will Come	03
Africain à Paris	01
Another One	05
So Strong	04
Y'En A Marre	01
Buffalo Soldier	02
Who Dem	03
Three Little Birds	02
Burn Dem Down	03
Soulstorm	05
Tour	03
Music	05
Violence verbale	06
Walking Alone	05
Up In My Room	05
Fè yomal	06
I Shot The Sheriff	02
Exodus	02

artistes			
Id	Nom	Pays	Date_naissance
01	Tiken Jah Fakoly	Côte d'Ivoire	1968
02	Bob Marley	Jamaïque	1945
03	Capelton	Jamaïque	1967
04	Admiral T	Guadeloupe	1981
05	Patrice	Allemagne	1979
06	Paille	Martinique	1982

1. Dans la relation **`chansons`** l'attribut **`id_artiste`**, peut-il être considérer comme une clé primaire.

   .....

2. Combien de chansons sont de `Bob Marley`.

. . . . . .

3. Ecrire une requête SQL permettant d'obtenir le résultat précédent.

. . . . . .

Schéma relationnel

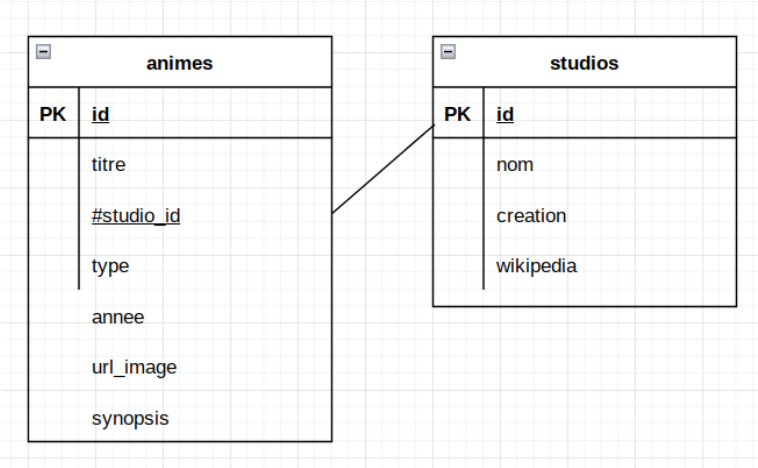
Définition

Dans un schéma relationnel :

- On souligne les clés (primaires et étrangères)
- Les clés étrangères sont précédées de **#**

Exemple

On dispose du schéma ci-dessous :



Dans la table `animes` :

- **id** est un clé primaire
- **studio_id** est un clé étrangère

Dans la table `studios` :

- **id** est un clé primaire
-

La clé **studio_id** de la table `animes` et la cle **id** de la relation sont liées.

Exercice 1

Dans cet exercice, on souhaite faire l'analyse d'une base de données dans laquelle vous voulez stocker :

- Le nom d'une série à regarder.
- Le réalisateur de cette série (Nom, prénom).
- La date de sortie de cette série.
- La date de naissance du réalisateur
- Le nombre de saisons.
- L'origine du réalisateur
- L'âge légal à partir duquel on peut visionner la série.
- La plateforme qui héberge cette série (nom).

- Le prix l'abonnement à la plateforme.
- Un descriptif de la série (texte bref).

Questions :

(a) Combien de tables mettre en place pour cette base de données ?

(b) Proposer un schéma relationnel de cette base de données.