

# HUMAN**booster**



VOTRE SOLUTION COMPÉTENCE

## **Bases de données relationnelles**

Conception ● Utilisation

Administration ● Optimisation

# Normalisation

# Les formes normales : objectifs ?


---

- **Lorsqu'une base de donnée est normalisée :**
  - Elle ne comporte pas redondances d'informations (absence de doublons)
  - Elle ne présente aussi peu de risques d'incohérence que possible (les informations contenues ne seront pas aberrantes)
  - Elle est plus facilement exploitable (les données sont faciles à lire, faciles à mettre à jour)
  - Sa maintenance et son évolutivité sont plus aisées
  - Ses performances et sa pérennité sont augmentées (la base « vieillira » bien)

# 1FN – Première forme normale

- Une base est en première forme normale si aucune table n'a de données « multiples » (les attributs doivent avoir des valeurs atomiques) :

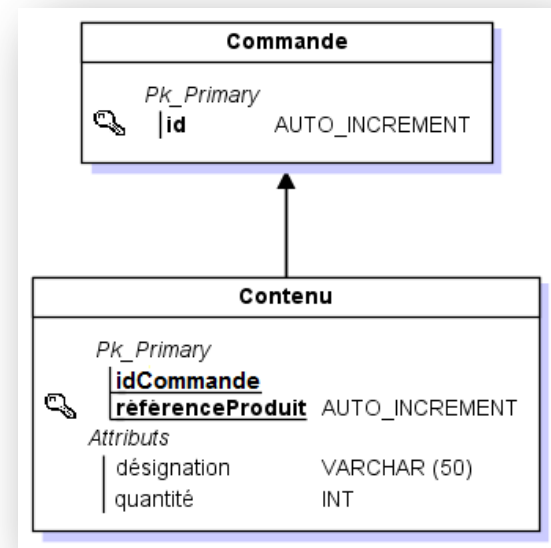
Cette table « Livre » n'est pas en 1FN car l'attribut « éditeurs » peut contenir plusieurs informations

LIVRE		
<i>Pk_Primary</i>		
	<b>id</b>	AUTO_INCREMENT
<i>Attributs</i>		
	titre	VARCHAR (50)
	éditeurs	VARCHAR (200)

# 2FN – Deuxième forme normale

- Une base est en deuxième forme normale si elle est déjà en première forme normale et que les attributs de chaque table dépendent bien de la totalité de la clé primaire :


Cette table « Contenu » n'est pas en 2FN car l'attribut « désignation » dépend de « référenceProduit », mais pas de « idCommande »



# 3FN – Troisième forme normale

- Une base est en troisième forme normale si elle est déjà en deuxième forme normale et que les attributs de chaque table dépendent bien directement de la clé primaire (et pas « indirectement » ; pas de manière « transitive ») :

Cette table « Individu » n'est pas en 3FN car les attributs « idCommuneNaissance » et « nomCommuneNaissance » ne dépendent pas directement de l'individu

Individu	
<i>Pk_Primary</i>	
 id	AUTO_INCREMENT
<i>Attributs</i>	
nom	VARCHAR (50)
prénom	VARCHAR (50)
idCommuneNaissance	INT
nomCommuneNaissance	VARCHAR (50)

# Il existe d'autres formes normales

---

- D'autres formes normales existent :
  - Forme normale de Boyce-Codd
  - Quatrième forme normale
  - Cinquième forme normale
  - Sixième forme normale
- Ces formes normales améliorent encore la cohérence des données. Elles sont plus pointues, et plus poussées.
- L'on considère généralement que la 3<sup>e</sup> forme normale constitue un bon équilibre entre normalisation et performance, mais...

# Dénormaliser consciemment

---

- **Respecter la 3<sup>e</sup> forme normale n'est pas une obligation absolue : selon les besoins, il peut être intéressant de « dénormaliser » un modèle relationnel (performance)**
- **Exemple : une table « Facture » est liée à une table « Ligne » : même s'il est possible de recalculer les montants totaux à partir des lignes de la facture, il peut être pertinent d'ajouter une colonne « montantTotal » dans la table « Facture ». Ceci permet :**
  - **De rechercher/trier plus simplement par montant total**
  - **De pouvoir purger la table « Ligne » en conservant tout de même le montant total**