

Introduction | STRUCTURES DE DONNÉES DE BASE | CONTRAINTES D'INTÉGRITÉ STRUCTURELLES

## SOLUTION

➤ Les clés étrangères:

- ✓ Etudiant(NumInscrEt, CinEt, NomEt, PrenomEt, AdrEt, TeleEt)
- ✓ Matiere(CodMat, Intitule)
- ✓ Note(#NumInscrEt, #CodMat, Note)

81

Introduction | STRUCTURES DE DONNÉES DE BASE | CONTRAINTES D'INTÉGRITÉ STRUCTURELLES

## CONTRAINTES D'UNICITÉ

Définitions:

- ✓ **Valeur nulle**: est une valeur conventionnelle introduite dans une relation pour représenter une information inconnue ou inapplicable.
- ✓ **Contraintes d'unicité**: est une contrainte d'intégrité imposant que toute relation possède une clé primaire et que tout attribut participant à cette clé primaire soit non nulle.

- La définition d'une clé primaire contient une **contrainte d'unicité**;
- Les valeurs de clés primaires doivent être :
  - ✓ Uniques (impossibles de trouvé deux tuples ou plus ayant les mêmes valeurs pour les attributs clés);
  - ✓ non nulles.

82

Introduction | STRUCTURES DE DONNÉES DE BASE | CONTRAINTES D'INTÉGRITÉ STRUCTURELLES

## CONTRAINTES DE RÉFÉRENCES

Définition:

**Contrainte de référence**: est une contrainte d'intégrité portant sur une relation consistant à imposer que la valeur d'un groupe d'attributs apparaisse comme valeur de clé dans une autre relation.

- Une contrainte d'intégrité référentielle est générée pour chaque clé étrangère qui exprime un lien (association) entre deux relations;
- Les valeurs des clés étrangères doivent correspondent aux valeurs des clé primaire auxquelles elles font référence ce qui signifie que l'ensemble des valeurs prises par l'attribut dans la table ou il est clé étrangère est inclus dans l'ensemble des valeurs de l'attribut dans la table ou il est clé primaire.

83

Introduction | STRUCTURES DE DONNÉES DE BASE | CONTRAINTES D'INTÉGRITÉ STRUCTURELLES

## CONTRAINTES DE DOMAINES

Définition:

**Contrainte de domaine**: est une contrainte d'intégrité imposant qu'un attribut d'une relation doit comporter des valeurs vérifiant une condition logique.

- Appliquer des conditions sur les données afin d'assurer leurs cohérences;
- Exemple:
  - ✓ CHECK (Filiere IN ('GI', 'TM', 'TIMQ', 'GIM'));
  - ✓ CHECK (annee\_naissance BETWEEN 1998 AND 2001);
  - ✓ telephone CHAR(14) UNIQUE; (impossibles de trouvé deux tuples ou plus ayant les mêmes valeurs pour l'attribut telephone);

84

Introduction | STRUCTURES DE DONNÉES DE BASE | CONTRAINTES D'INTÉGRITÉ STRUCTURELLES

## CONTRAINTES VALEURS PAR DÉFAUT

**Définition:**

Contrainte valeur par défaut: est une contrainte d'intégrité qui s'applique lorsqu'une valeur d'un attribut n'est pas renseignée lors d'une opération d'insertion.

- Cette contrainte consiste à attribuer automatiquement une valeur à un attribut ce qui permet de garantir qu'un attribut ne soit pas vide;
- Exemple:
  - ✓ Nationalite VARCHAR (30) DEFAULT 'Marocaine' .

85

Introduction | STRUCTURES DE DONNÉES DE BASE | CONTRAINTES D'INTÉGRITÉ STRUCTURELLES

## EXERCICE D'APPLICATION

➤ Soient les relations suivantes:

- ❑ Etudiant(NumInscrEt, CinEt, NomEt, PrenomEt, AdrEt, TeleEt)
- ❑ Matiere(CodMat, Intitule)
- ❑ Note(#NumInscrEt, #CodMat, Date\_obtention, Note)

1. Donner l'arité de chaque relation;
2. Proposer des valeurs (de votre choix) aux attributs des relations en respectant les contraintes d'intégrité structurelles;
3. Qu'elle est la cardinalité de chaque relation;
4. Est-ce qu'on peut insérer dans la relation **Note** un tuple portant une valeur de l'attribut **NumInscrEt** qui n'existe pas dans la relation **Etudiant**?
5. Que ce passe t'il lorsqu'on supprime un étudiant qui a obtenu des notes dans une matière?
6. Que ce passe t'il lorsqu'on modifie le **NumInscrEt** d'un étudiant?

86

Optimisation des Bases de données relationnelles

Chapitre 4:

### Introduction

#### La normalisation

- ✓ Pourquoi la normalisation?
- ✓ Théorie de la normalisation

#### Dépendance fonctionnelle

- ✓ Graphe des Dépendances Fonctionnelles: GDF
- ✓ La décomposition

#### Les formes normales

- ✓ 1<sup>ère</sup> Forme Normale: 1FN
- ✓ 2<sup>ème</sup> Forme Normale: 2FN
- ✓ 3<sup>ème</sup> Forme Normale: 3FN
- ✓ Forme Normale de Boyce-Codd: FNBC

### Conclusion

Introduction | NORMALISATION | DÉPENDANCE FONCTIONNELLE | FORMES NORMALES | CONCLUSION

## INTRODUCTION

La **conception** d'une base de données relationnelle consiste à déterminer l'ensemble des relations, des attributs et des contraintes qui constitueront le schéma de la BD;

l'objectif de l'**optimisation** c'est d'obtenir un schéma de base de données qui répond totalement aux exigences des clients et ne présente pas d'anomalies.

88

Introduction | **NORMALISATION** | DÉPENDANCE FONCTIONNELLE | FORMES NORMALES | CONCLUSION

## LA NORMALISATION

**Définition:**

La normalisation est le processus de restructuration de schéma de la BD afin de :

- ✓ Eliminer les redondances;
- ✓ Organiser les données efficacement;
- ✓ Eviter les anomalies qui peuvent se présenter pendant les opérations sur les données.

L'objectif de la normalisation c'est de veiller à la cohérence des données stockées dans la base de données.

89

Introduction | **NORMALISATION** | DÉPENDANCE FONCTIONNELLE | FORMES NORMALES | CONCLUSION

## POURQUOI LA NORMALISATION?

- ♦ La mise à jours des données d'une BD peut provoquer les anomalies suivantes:
  - ✓ Anomalie d'insertion (si on ajoute une note il faut que l'étudiant et la matière existent);
  - ✓ Anomalie de suppression (si on supprime un étudiant il faut supprimer toutes les notes de cet étudiant);
  - ✓ Anomalie de modification (si on modifie le numéro d'inscription d'un étudiant il faut modifier toutes les notes avec ce numéro d'inscription d'étudiant);
- ♦ La normalisation consiste à éviter ces anomalies sans perte de données ni de perte des associations entre les données.

90

Introduction | **NORMALISATION** | DÉPENDANCE FONCTIONNELLE | FORMES NORMALES | CONCLUSION

## THÉORIE DE LA NORMALISATION

La théorie de la normalisation: définit des critères appelées **formes normales** permettant d'avoir une base de données cohérente, elle définit aussi le processus de normalisation permettant de décomposer (diviser) une relation non normalisée en un ensemble équivalent de relations normalisées ;

La théorie de la normalisation repose sur l'analyse de **dépendances entre les attributs** qui sont la source de redondance et d'anomalie.

91

Introduction | **NORMALISATION** | **DÉPENDANCE FONCTIONNELLE** | FORMES NORMALES | CONCLUSION

## DÉPENDANCE FONCTIONNELLE

**Définition:**

On dit qu'un attribut B dépend fonctionnellement d'un attribut A si étant donné une valeur de A, il lui correspond une valeur unique de B (et ceci quel que soit l'instant considéré).

- ♦ On dit que A détermine fonctionnellement B et on note :
 
$$A \rightarrow B$$

A est le déterminant, B le déterminé;
- ♦ **Exemples:**
  - NumInscrEt  $\rightarrow$  CinEt, NomEt, PrenomEt, AdrEt, TeleEt
  - CodMat  $\rightarrow$  Intitule
  - NumInscrEt, CodMat, Date\_obtention  $\rightarrow$  Note

92

Introduction | NORMALISATION | DÉPENDANCE FONCTIONNELLE | FORMES NORMALES | CONCLUSION

## GRAPHE DES DÉPENDANCES FONCTIONNELLES: GDF

**Définition:**

Le graphe des dépendances fonctionnelles est une représentation graphique de toutes les dépendances fonctionnelles sous la forme d'un graphe orienté : chaque attribut est un nœud du graphe et chaque dépendance fonctionnelle est une arrête du graphe.

♦ **Exemple:** soient les dépendances fonctionnelles:

NumInscrEt → CinEt, NomEt, PrenomEt, AdrEt, TeleEt  
 CodMat → Intitulé  
 NumInscrEt, CodMat, Date\_obtention → Note

**GDF:**

93

Introduction | NORMALISATION | DÉPENDANCE FONCTIONNELLE | FORMES NORMALES | CONCLUSION

## LA DÉCOMPOSITION

♦ L'objectif de la décomposition est de:

- ✓ Décomposer (diviser) une relation en plusieurs relations sans perte de données;
- ✓ Obtenir des relations normalisées.

94

Introduction | NORMALISATION | DÉPENDANCE FONCTIONNELLE | FORMES NORMALES | CONCLUSION

## LES FORMES NORMALES

**Objectif:**

L'objectif des formes normales c'est d'étudier les dépendances fonctionnelles entre les attributs des relations afin d'éliminer les attributs redondant et de minimiser les anomalies.

♦ Un attribut dans une relation est redondant lorsque ses valeurs peuvent être éliminées de cette relation sans perte d'information;

♦ Les principales formes normales sont:

- ✓ 1<sup>ère</sup> forme normale (noté: 1FN);
- ✓ 2<sup>ème</sup> forme normale (noté: 2FN);
- ✓ 3<sup>ème</sup> forme normale (noté: 3FN);
- ✓ Forme normale de Boyce-Codd (noté: FNBC).

95

Introduction | NORMALISATION | DÉPENDANCE FONCTIONNELLE | FORMES NORMALES | CONCLUSION

## 1<sup>ère</sup> FORME NORMALE: 1FN

**Définition:**

Une relation est en première forme normale si elle possède une clé primaire et si la valeur de tout attribut est « atomique » (tout attribut contient une valeur non décomposable), c'est-à-dire n'est pas constituée par une liste de valeurs.

**Exemple:**

Soit la relation: Etudiant(CodE, PreE, Note)

CodE	PreE	Note
1	Ahmed	13, 17
2	Fadoua	14, 13

Soit créer un attribut par note si le nombre maximal de notes est connu;

CodE	PreE	Note1	Note2
1	Ahmed	13	17
2	Fadoua	14	13

Normalisation 1FN

Soit décomposer la relation en deux relation Etudiant(CodE, PreE) et Note(#CodE, Note).

CodE	PreE	#CodE	Note
1	Ahmed	1	13
1	Ahmed	1	17
2	Fadoua	2	14
2	Fadoua	2	13

La relation Etudiant n'est pas en 1FN car l'attribut Note est constitué de plusieurs valeurs, pour normaliser cette relation on peut:

96

Introduction | NORMALISATION | DÉPENDANCE FONCTIONNELLE | FORMES NORMALES | CONCLUSION

## 2ème FORME NORMALE: 2FN

**Définition:**

Une relation est en deuxième forme normale si:

- ✓ Elle est en première forme normale;
- ✓ Aucun attribut non clé ne dépend fonctionnellement d'une partie de cette clé (tout attribut dépend de toute la clé), c'est à dire qu'il n'y a pas de **DF Partielles**.

♦ La 2FN concerne les relations dont la clé est constituée de plusieurs attributs;

♦ **Exemple:** Soit la relation R(A1, A2, A3, A4) avec les Dépendances Fonctionnelles: (A1, A2 → A3), (A1, A2 → A4) et (A1 → A3).

Normalisation 2FN

R est en 1FN mais n'est pas en 2FN car il existe une DFP entre A3 et A1, pour normaliser cette relation on peut:

décomposer la relation R en deux relations R1(A1, A3) et R2(A1, A2, A4)

A1 A3 A1, A2 A4

97

Introduction | NORMALISATION | DÉPENDANCE FONCTIONNELLE | FORMES NORMALES | CONCLUSION

## 3ème FORME NORMALE: 3FN

**Définition:**

Une relation est en troisième forme normale si:

- ✓ Elle est en deuxième forme normale;
- ✓ Aucun attribut non clé ne dépend fonctionnellement d'un attribut non-clé (tout attribut ne dépend que de la clé), c'est à dire qu'il n'y a pas de **DF Transitives**.

♦ **Exemple:** Soit la relation R(A1, A2, A3, A4) avec les Dépendances Fonctionnelles: (A1, A2 → A3), (A1, A2 → A4) et (A3 → A4).

Normalisation 3FN

R est en 1FN, 2FN mais n'est pas en 3FN car il existe une DFT entre A4 et A3, pour normaliser cette relation on peut:

décomposer la relation R en deux relations R1(A3, A4) et R2(A1, A2, #A3)

A3 A4 A1, A2 A3

98

Introduction | NORMALISATION | DÉPENDANCE FONCTIONNELLE | FORMES NORMALES | CONCLUSION

## FORME NORMALE DE BOYCE-CODD: FNBC

**Définition:**

Une relation est en forme normale de BOYCE-CODD si:

- ✓ Elle est en troisième forme normale;
- ✓ Aucun attribut faisant partie de la clé ne dépend fonctionnellement d'un attribut ne faisant pas partie de la clé.

♦ **Exemple:** Soit la relation R(A1, A2, A3, A4) avec les Dépendances Fonctionnelles: (A1, A2 → A3), (A1, A2 → A4) et (A3 → A1).

Normalisation FNBC

R est en 1FN, 2FN mais n'est pas en 3FN car il existe une DFR(DF Réflexive) entre A1 et A3, pour normaliser cette relation on peut:

décomposer la relation R en deux relations R1(A3, A1) et R2(A2, A3, A4)

A3 A1 A2, A3 A4

99

Introduction | NORMALISATION | DÉPENDANCE FONCTIONNELLE | FORMES NORMALES | CONCLUSION

## CONCLUSION

1FN  
2FN  
3FN  
FNBC

Attribut clé ne DF d'attribut non clé.

Pas de DFT.

Pas de DFP.

Tous les valeurs des attributs sont atomiques.

Relations non normalisées.

♦ Un schéma de BDR en forme normale de Boyce-Codd est un schéma optimal (càd de qualité suffisante pour une implantation sous un SGBD).

100