

Exploitation d'une base de données

Ressource R2.06

Séance 1

30 janvier 2023

Objectifs de la ressource

- ▶ Poursuivre l'initiation aux bases de données.
- ▶ Ecrire des requêtes complexes en SQL.
- ▶ S'initier à l'administration et à la sécurité des données.
- ▶ Apprendre à présenter des données de manière synthétique.

Qualité des schémas relationnels

Qualité d'un schéma relationnel

Conception d'un schéma

Un schéma relationnel de base de données peut être conçu de plusieurs façons pour les besoins d'une application. Cependant, il y a des bons et des mauvais schémas relationnels.

Qualité d'un schéma relationnel

Conception d'un schéma

Un schéma relationnel de base de données peut être conçu de plusieurs façons pour les besoins d'une application. Cependant, il y a des bons et des mauvais schémas relationnels.

Notions abordées

- ▶ Les anomalies et incohérences dues à un schéma défectueux
- ▶ Les dépendances fonctionnelles
- ▶ Les clés primaires et étrangères
- ▶ La normalisation

Pourquoi pas une seule table ?

id	nomEtud	numLycée	ville	matière	enseignant	nomDépt	note
1	Bartoli M	2535	Montreuil	Programmation	Lovelace A	Informatique	14
2	Bartoli M	2535	Montreuil	Réseaux	Mayer M	Informatique	13
3	Aron C	2580	Rosny	Programmation	Lovelace A	Informatique	14
4	Bonaly S	2676	Créteil	Programmation	Lovelace A	Informatique	15
5	Bonaly S	2676	Créteil	Réseaux	Mayer M	Informatique	13
6	Bartoli M	2535	Montreuil	Econométrie	Lagarde C	Economie	14

Pourquoi pas une seule table ?

id	nomEtud	numLycée	ville	matière	enseignant	nomDépt	note
1	Bartoli M	2535	Montreuil	Programmation	Lovelace A	Informatique	14
2	Bartoli M	2535	Montreuil	Réseaux	Mayer M	Informatique	13
3	Aron C	2580	Rosny	Programmation	Lovelace A	Informatique	14
4	Bonaly S	2676	Créteil	Programmation	Lovelace A	Informatique	15
5	Bonaly S	2676	Créteil	Réseaux	Mayer M	Informatique	13
6	Bartoli M	2535	Montreuil	Econométrie	Lagarde C	Economie	14

Problèmes de qualité

Pourquoi pas une seule table ?

id	nomEtud	numLycée	ville	matière	enseignant	nomDépt	note
1	Bartoli M	2535	Montreuil	Programmation	Lovelace A	Informatique	14
2	Bartoli M	2535	Montreuil	Réseaux	Mayer M	Informatique	13
3	Aron C	2580	Rosny	Programmation	Lovelace A	Informatique	14
4	Bonaly S	2676	Créteil	Programmation	Lovelace A	Informatique	15
5	Bonaly S	2676	Créteil	Réseaux	Mayer M	Informatique	13
6	Bartoli M	2535	Montreuil	Econométrie	Lagarde C	Economie	14

Problèmes de qualité

► Redondance

Pourquoi pas une seule table ?

id	nomEtud	numLycée	ville	matière	enseignant	nomDépt	note
1	Bartoli M	2535	Montreuil	Programmation	Lovelace A	Informatique	14
2	Bartoli M	2535	Montreuil	Réseaux	Mayer M	Informatique	13
3	Aron C	2580	Rosny	Programmation	Lovelace A	Informatique	14
4	Bonaly S	2676	Créteil	Programmation	Lovelace A	Informatique	15
5	Bonaly S	2676	Créteil	Réseaux	Mayer M	Informatique	13
6	Bartoli M	2535	Montreuil	Econométrie	Lagarde C	Economie	14

Problèmes de qualité

- Redondance
- Anomalies de modification

Pourquoi pas une seule table ?

id	nomEtud	numLycée	ville	matière	enseignant	nomDépt	note
1	Bartoli M	2535	Montreuil	Programmation	Lovelace A	Informatique	14
2	Bartoli M	2535	Montreuil	Réseaux	Mayer M	Informatique	13
3	Aron C	2580	Rosny	Programmation	Lovelace A	Informatique	14
4	Bonaly S	2676	Créteil	Programmation	Lovelace A	Informatique	15
5	Bonaly S	2676	Créteil	Réseaux	Mayer M	Informatique	13
6	Bartoli M	2535	Montreuil	Econométrie	Lagarde C	Economie	14

Problèmes de qualité

- ▶ Redondance
- ▶ Anomalies de modification
- ▶ Anomalies de suppression

Pourquoi pas une seule table ?

id	nomEtud	numLycée	ville	matière	enseignant	nomDépt	note
1	Bartoli M	2535	Montreuil	Programmation	Lovelace A	Informatique	14
2	Bartoli M	2535	Montreuil	Réseaux	Mayer M	Informatique	13
3	Aron C	2580	Rosny	Programmation	Lovelace A	Informatique	14
4	Bonaly S	2676	Créteil	Programmation	Lovelace A	Informatique	15
5	Bonaly S	2676	Créteil	Réseaux	Mayer M	Informatique	13
6	Bartoli M	2535	Montreuil	Econométrie	Lagarde C	Economie	14

Problèmes de qualité

- ▶ Redondance
- ▶ Anomalies de modification
- ▶ Anomalies de suppression
- ▶ Anomalies d'insertion

Pourquoi pas une seule table ?

id	nomEtud	numLycée	ville	matière	enseignant	nomDépt	note
1	Bartoli M	2535	Montreuil	Programmation	Lovelace A	Informatique	14
2	Bartoli M	2535	Montreuil	Réseaux	Mayer M	Informatique	13
3	Aron C	2580	Rosny	Programmation	Lovelace A	Informatique	14
4	Bonaly S	2676	Créteil	Programmation	Lovelace A	Informatique	15
5	Bonaly S	2676	Créteil	Réseaux	Mayer M	Informatique	13
6	Bartoli M	2535	Montreuil	Econométrie	Lagarde C	Economie	14

Problèmes de qualité

- Redondance : répétition inutile d'informations dans la table.

Pourquoi pas une seule table ?

<u>id</u>	<u>nomEtud</u>	<u>numLycée</u>	<u>ville</u>	<u>matière</u>	<u>enseignant</u>	<u>nomDépt</u>	<u>note</u>
1	Bartoli M	2535	St Denis	Programmation	Lovelace A	Informatique	14
2	Bartoli M	2535	St Denis	Réseaux	Mayer M	Informatique	13
3	Aron C	2580	Rosny	Programmation	Lovelace A	Informatique	14
4	Bonaly S	2676	Créteil	Programmation	Lovelace A	Informatique	15
5	Bonaly S	2676	Créteil	Réseaux	Mayer M	Informatique	13
6	Bartoli M	2535	Montreuil	Econométrie	Lagarde C	Economie	14

Problèmes de qualité

- Redondance : répétition inutile d'informations dans la table.
- Anomalie de modification : mise à jour dans un tuple mais pas dans un autre.

Pourquoi pas une seule table ?

id	nomEtud	numLycée	ville	matière	enseignant	nomDépt	note
1	Bartoli M	2535	Montreuil	Programmation	Lovelace A	Informatique	14
2	Bartoli M	2535	Montreuil	Réseaux	Mayer M	Informatique	13
3	Aron C	2580	Rosny	Programmation	Lovelace A	Informatique	14
4	Bonaly S	2676	Créteil	Programmation	Lovelace A	Informatique	15
5	Bonaly S	2676	Créteil	Réseaux	Mayer M	Informatique	13
6	Bartoli M	2535	Montreuil	Econométrie	Lagarde C	Economie	14

Problèmes de qualité

- ▶ Redondance : répétition inutile d'informations dans la table.
- ▶ Anomalie de modification : mise à jour dans un tuple mais pas dans un autre.
- ▶ Anomalie de suppression : perte d'informations lorsque l'on supprime certaines valeurs.

Pourquoi pas une seule table ?

id	nomEtud	numLycée	ville	matière	enseignant	nomDépt	note
1	Bartoli M	2535	Montreuil	Programmation	Lovelace A	Informatique	14
2	Bartoli M	2535	Montreuil	Réseaux	Mayer M	Informatique	13
3	Aron C	2580	Rosny	Programmation	Lovelace A	Informatique	14
4	Bonaly S	2676	Créteil	Programmation	Lovelace A	Informatique	15
5	Bonaly S	2676	Créteil	Réseaux	Mayer M	Informatique	13
6	Bartoli M	2535	Montreuil	Econométrie	Lagarde C	Economie	14
NULL	NULL	NULL	NULL	Chimie	Curie M	Phys/Chimie	NULL
7	Pierce M	3475	Paris	Prog	Lovelace A	Info	16

Problèmes de qualité

- ▶ Redondance : répétition inutile d'informations dans la table.
- ▶ Anomalie de modification : mise à jour dans un tuple mais pas dans un autre.
- ▶ Anomalie de suppression : perte d'informations lorsque l'on supprime certaines valeurs.
- ▶ Anomalie d'insertion : ajout de nouvelles valeurs impossible en l'absence d'autres informations (clé primaire) ou insertion avec erreurs.

Qualité d'un schéma relationnel

Eviter la redondance

- ▶ Mise à jour des données faite en un minimum d'opérations, donc réduction des risques d'incohérences dans la base de données.
- ▶ Réduction de l'espace de stockage, donc réduction des coûts.

Qualité d'un schéma relationnel

Eviter la redondance

- ▶ Mise à jour des données faite en un minimum d'opérations, donc réduction des risques d'incohérences dans la base de données.
- ▶ Réduction de l'espace de stockage, donc réduction des coûts.

La décomposition

Les anomalies peuvent être évitées en décomposant une relation unique en plusieurs relations qui sont liées entre elles par des clés étrangères.

Qualité d'un schéma relationnel

Décomposition de la relation :

id	nomEtud	numLycée	ville	matière	enseignant	nomDépt	note
1	Bartoli M	2535	Montreuil	Programmation	Lovelace A	Informatique	14
2	Bartoli M	2535	Montreuil	Réseaux	Mayer M	Informatique	13
3	Aron C	2580	Rosny	Programmation	Lovelace A	Informatique	14
4	Bonaly S	2676	Créteil	Programmation	Lovelace A	Informatique	15
5	Bonaly S	2676	Créteil	Réseaux	Mayer M	Informatique	13
6	Bartoli M	2535	Montreuil	Econométrie	Lagarde C	Economie	14

en trois relations :

numEtud	nomEtud	numLycée	ville
1	Bartoli M	2535	Montreuil
2	Aron C	2580	Rosny
3	Bonaly S	2676	Créteil

numMat	matière	enseignant	nomDépt
1	Programmation	Lovelace A	Informatique
2	Réseaux	Mayer M	Informatique
3	Econométrie	Lagarde C	Economie

numEtud	numMat	note
1	1	14
1	2	13
1	3	14
2	1	14
3	1	15
3	2	13

Qualité d'un schéma relationnel

Décomposition de la relation :

id	nomEtud	numLycée	ville	matière	enseignant	nomDépt	note
1	Bartoli M	2535	Montreuil	Programmation	Lovelace A	Informatique	14
2	Bartoli M	2535	Montreuil	Réseaux	Mayer M	Informatique	13
3	Aron C	2580	Rosny	Programmation	Lovelace A	Informatique	14
4	Bonaly S	2676	Créteil	Programmation	Lovelace A	Informatique	15
5	Bonaly S	2676	Créteil	Réseaux	Mayer M	Informatique	13
6	Bartoli M	2535	Montreuil	Econométrie	Lagarde C	Economie	14

en trois relations :

numEtud	nomEtud	numLycée	ville
1	Bartoli M	2535	Montreuil
2	Aron C	2580	Rosny
3	Bonaly S	2676	Créteil

numMat	matière	enseignant	nomDépt
1	Programmation	Lovelace A	Informatique
2	Réseaux	Mayer M	Informatique
3	Econométrie	Lagarde C	Economie

numEtud	numMat	note
1	1	14
1	2	13
1	3	14
2	1	14
3	1	15
3	2	13

Normalisation

Normalisation

Définition

Technique de conception de base de données qui permet d'éliminer les incohérences et les anomalies, de réduire la redondance des données et de garantir qu'elles sont stockées de manière logique.

Normalisation

Définition

Technique de conception de base de données qui permet d'éliminer les incohérences et les anomalies, de réduire la redondance des données et de garantir qu'elles sont stockées de manière logique.

Caractéristiques

- ▶ Sémantique simple des attributs et des relations.
- ▶ Nombre minimum d'attributs : pas de répétition d'attributs sauf ceux formant la totalité ou une partie des clés étrangères.
- ▶ Réduction des valeurs nulles et redondantes dans les relations.
- ▶ Les **dépendances fonctionnelles** traduisent des contraintes sur les données et servent à analyser la qualité d'un schéma relationnel.
- ▶ Les **formes normales** permettent de définir des tables bien conçues.

Normalisation

Dépendance fonctionnelle

Soient A et B deux attributs d'une relation R . On dit qu'il y a une **dépendance fonctionnelle** entre A et B , ou **A détermine fonctionnellement B** , ou **B est fonctionnellement dépendant de A** , noté $A \rightarrow B$, si pour chaque valeur de A , il y a exactement une seule valeur de B .

Exemple :

[nom, prénom, noSS, dateNaissance, adresse, email]

Les dépendances fonctionnelles :

- ▶ $\text{email} \rightarrow \text{nom}, \text{prénom}, \text{noSS}, \text{dateNaissance}, \text{adresse}$
- ▶ $\text{noSS} \rightarrow \text{email}, \text{nom}, \text{prénom}, \text{dateNaissance}, \text{adresse}$

Normalisation

Appellation

Soient X et Y des attributs ou des ensembles d'attributs d'une relation R . Soit la dépendance fonctionnelle $X \rightarrow Y$.

- ▶ **Déterminant** : attribut(s) en partie gauche de la DF (X).
- ▶ **Dépendant** : attribut(s) en partie droite de la DF (Y).

Exemples

1) Commander [idCommande, idProduit, qtéCommandée]

DF : idCommande, idProduit \rightarrow qtéCommandée

déterminant = {idCommande, idProduit}

dépendant = qtéCommandée

2) Résultats [noEtudiant, noCours, année, note, titreCours]

DF1 : noEtudiant, noCours \rightarrow note

déterminant = {noEtudiant, noCours}

dépendant = note

DF2 : noCours \rightarrow titreCours

déterminant = noCours

dépendant = titreCours

Il y a encore d'autres DF pour cette relation.

Normalisation

DF complète (ou élémentaire)

Soit X un ensemble d'attributs. $X \rightarrow A$ est une DF complète si :

- ▶ A est un attribut unique
- ▶ $A \notin X$
- ▶ $\nexists X'$ t.q. $X' \subsetneq X$ (X' inclus strictement dans X) et $X' \rightarrow A$.

Exemple

$\text{idCommande}, \text{idProduit} \rightarrow \text{qtéCommandée}$ est une DF complète car qtéCommandée ne dépend pas uniquement de idCommande ou uniquement de idProduit . Elle dépend des deux attributs en même temps.

Normalisation

DF directe

Soit X un ensemble d'attributs. $X \rightarrow A$ est une DF directe si :
 $\nexists Y$ t.q. $X \rightarrow Y$ et $Y \rightarrow A$ (i.e., il n'existe pas d'autres chemins pour déterminer A).

DF transitive

Soient les attributs A, B et C tels que $A \rightarrow B$ et $B \rightarrow C$. Alors $A \rightarrow C$. On dit que C est **transitivement dépendant** de A (via B), à condition que A ne dépende pas fonctionnellement de B ou de C .

Exemple : $\text{noEmploye} \rightarrow \text{noFiliale}$ et $\text{noFiliale} \rightarrow \text{adresse}$, alors on a la DF transitive $\text{noEmploye} \rightarrow \text{adresse}$.

Normalisation

Rôle des DFs

Trouver ces types de DFs car risque potentiel d'anomalies lors des mises à jour des données.

Les DFs guident la normalisation

- ▶ Association 1 à 1 entre les déterminants et les attributs de droite de la DF.
- ▶ Elles sont vraies à tout moment.
- ▶ Le déterminant possède le minimum d'attributs nécessaires pour conserver la dépendance avec les attributs situés en partie droite, i.e., DF complètes.

Normalisation

Implication d'un ensemble de dfs

Soit F un ensemble de DFs, et $X \rightarrow Y$ une DF. $F \models X \rightarrow Y$ (F implique $X \rightarrow Y$) ssi $X \rightarrow Y$ est vraie dans tous les schémas de relations où F est vraie.

Remarque

L'ensemble F regroupe des DFs vraies sur une relation R .

Exemple

$R[A, B, C, D]$ et $F = \{A, B \rightarrow C ; B \rightarrow A ; C \rightarrow A\}$ Alors $F \models B \rightarrow C$.

Normalisation

Axiomes d'Armstrong

- **Réflexivité** : si B est un sous-ensemble d'attributs de A alors $A \rightarrow B$.

Exemple : $\text{idCommande}, \text{idProduit} \rightarrow \text{idProduit}$.

- **Augmentation** : si $A \rightarrow B$ alors $\forall C$ on a $A, C \rightarrow B, C$
- **Transitivité** : si $A \rightarrow B$ et $B \rightarrow C$ alors $A \rightarrow C$.

Normalisation

Règles dérivées des axiomes d'Armstrong

- ▶ **Autodétermination** : $A \rightarrow A$.
- ▶ **Décomposition** : si $A \rightarrow B, C$ alors $A \rightarrow B$ et $A \rightarrow C$.
- ▶ **Union** : si $A \rightarrow B$ et $A \rightarrow C$ alors $A \rightarrow B, C$.
- ▶ **Composition** : si $A \rightarrow B$ et $C \rightarrow D$ alors $A, C \rightarrow B, D$.