Requêtes sur plusieurs table

Lycée Thiers

Crédit

- Un cours de Quentin Fortier

Crédit

- Un cours de Quentin Fortier
- Le tuto SQL du W3C (indispensable)

Crédit

- Un cours de Quentin Fortier
- Le tuto SQL du W3C (indispensable)
- Un cours en français ici

Généralités

- Ces opérateurs ensemblistes concernent *uniquement* des relations ayant le même schéma.

P1		
Nom	Id	Prénom
Hoareau	2011	Patrice
Grondin	5256	Marie
Dupont	52	Patrice

Généralités

- Ces opérateurs ensemblistes concernent *uniquement* des relations ayant le même schéma.

P1		
Nom	Id	Prénom
Hoareau	2011	Patrice
Grondin	5256	Marie
Dupont	52	Patrice

-

P2		
Nom	Id	Prénom
Dupont	52	Patrice
Grondin	5256	Pierre

Généralités

- Ces opérateurs ensemblistes concernent *uniquement* des relations ayant le même schéma.

P1		
Nom	Id	Prénom
Hoareau	2011	Patrice
Grondin	5256	Marie
Dupont	52	Patrice

-

P2		
Nom	Id	Prénom
Dupont	52	Patrice
Grondin	5256	Pierre

- Il existe la notion de *schémas compatibles* quand deux relations ont le même nombre d'attributs et que les attributs ont le même domaine. PB : deux attributs de même domaine pouvant avoir des *sémantiques* différentes. Ex : `Age` et `Nombre de carottes`.

Union

Définition

L'*union* de deux relations $R_1(S)$ et $R_2(S)$ est l'ensemble ds n-uplets appartenant à $R_1(S)$ ou $R_2(S)$. On la note $R_1 \cup R_2(S)$ ou plus simplement $R_1 \cup R_2$.

-
-

P1∪P2		
Nom	Id	Prénom
Dupont	52	Patrice
Grondin	5256	Pierre
Hoareau	2011	Patrice
Grondin	5256	Marie

Intersection

Définition

L'*intersection* de deux relations $R_1(S)$ et $R_2(S)$ est l'ensemble ds n-uplets appartenant à $R_1(S)$ et $R_2(S)$. On la note $R_1 \cap R_2(S)$ ou plus simplement $R_1 \cap R_2$.

-
-

P1∩P2		
Nom	Id	Prénom
Dupont	52	Patrice

Différence

Définition

La *différence* de deux relations $R_1(S)$ et $R_2(S)$ est l'ensemble ds n-uplets appartenant à $R_1(S)$ mais pas à $R_2(S)$. On la note $R_1 - R_2(S)$ ou plus simplement $R_1 - R_2$.

- P1−P2

Nom	Id	Prénom
Grondin	5256	Pierre
Hoareau	2011	Patrice

Remarque

Pour tous ces opérateurs binaires, le schéma de la nouvelle table construite est le même que celui des deux tables entrées.

Opérations ensemblistes

UNION

- `UNION, INTERSECT` et `EXCEPT` sur des tables de même shéma.

Opérations ensemblistes

UNION

- UNION, INTERSECT et EXCEPT sur des tables de même shéma.
-

```
SELECT * FROM R1 UNION SELECT * FROM R2 ;
```

Opérations ensemblistes

UNION

- `UNION, INTERSECT` et `EXCEPT` sur des tables de même shéma.
-
```
SELECT * FROM R1 UNION SELECT * FROM R2 ;
```
- `UNION` donne la réunion sans doublon de lignes, `UNION ALL` donne la réunion avec potentiellement des doublons de lignes.

Opérations ensemblistes

UNION

- UNION, INTERSECT et EXCEPT sur des tables de même shéma.
-
```
SELECT * FROM R1 UNION SELECT * FROM R2 ;
```

- UNION donne la réunion sans doublon de lignes, UNION ALL donne la réunion avec potentiellement des doublons de lignes.
- La liste sans doublon des villes où il y a des clients ou des fournisseurs.

```
SELECT City FROM Customers
UNION
SELECT City FROM Suppliers
```

UNION ALL pour autoriser les doublons (pratique pour voir les effectifs par ville).

Opérations ensemblistes

Intersection

- Malheureusement, certains SGBD ne prennent pas en charge les commande `EXCEPT` et `INTERSECT` : il faut se débrouiller autrement : avec une jointure ou avec le mot clé `IN`.

Opérations ensemblistes

Intersection

- Malheureusement, certains SGBD ne prennent pas en charge les commande `EXCEPT` et `INTERSECT` : il faut se débrouiller autrement : avec une jointure ou avec le mot clé `IN`.
- Avec jointure (cf plus loin)

Opérations ensemblistes

Intersection

- Malheureusement, certains SGBD ne prennent pas en charge les commande EXCEPT et INTERSECT : il faut se débrouiller autrement : avec une jointure ou avec le mot clé IN.
- Avec jointure (cf plus loin)
- Avec IN on récupère les villes de clients qui sont aussi des villes d'opérateurs :

```sql
SELECT DISTINCT City FROM Customers
WHERE City IN(
SELECT City FROM Suppliers
ORDER BY City);
```

Noter le DISTINCT pour éviter les doublons

Opérations ensemblistes

Intersection

- Malheureusement, certains SGBD ne prennent pas en charge les commande EXCEPT et INTERSECT : il faut se débrouiller autrement : avec une jointure ou avec le mot clé IN.
- Avec jointure (cf plus loin)
- Avec IN on récupère les villes de clients qui sont aussi des villes d'opérateurs :

```
SELECT DISTINCT City FROM Customers
WHERE City IN(
SELECT City FROM Suppliers
ORDER BY City);
```

  Noter le DISTINCT pour éviter les doublons
- Remarque : En SQLite, la commande INTERSECT est bien reconnue.

Opérations ensemblistes

Soustraction ensembliste

- La commande `EXCEPT` (parfois `MINUS`) existante sur d'autres SGBD n'est pas implantée en MySQL.

Opérations ensemblistes

Soustraction ensembliste

- La commande EXCEPT (parfois MINUS) existante sur d'autres SGBD n'est pas implantée en MySQL.
- Mais on peut s'en passer.
  Donner sans doublon les villes de clients qui ne sont pas des villes de fournisseurs :

```
SELECT DISTINCT CITY FROM Customers
WHERE City NOT IN(
SELECT CITY FROM Suppliers);
```

Opérations ensemblistes

Soustraction ensembliste

- La commande EXCEPT (parfois MINUS) existante sur d'autres SGBD n'est pas implantée en MySQL.
- Mais on peut s'en passer.
  Donner sans doublon les villes de clients qui ne sont pas des villes de fournisseurs :

```
SELECT DISTINCT CITY FROM Customers
WHERE City NOT IN(
SELECT CITY FROM Suppliers);
```

- La commande MINUS est bien implantée en SQLite.

1 Opérateurs ensemblistes usuels

2 Produit cartésien, jointure
- Produit cartésien
- Division cartésienne

Produit cartésien

- La *concaténation* des listes S, S' est notée $S + S'$. Ex : $(1,2) + (2,4) = (1,2,2,4)$. Cet opérateur est parfois noté de façon plus précise $\uplus$.

Définition

Si $R(S)$ et $R'(S')$ sont deux relations de schémas disjoints, on appelle *produit cartésien* et on note $R \times R'$ la relation de schéma $S + S'$ définie par :

$$(R \times R')(S + S') = \{(u_1, \dots, u_n, v_1, \dots v_m) \mid (u_1, \dots, u_n) \in R(S) \wedge (v_1, \dots, v_m) \in R'(S')\}$$

-

Remarque

$\mathrm{card}(R \times R') = \mathrm{card}(R) \times \mathrm{card}(R')$ et $\mathrm{card}(S + S') = \mathrm{card}(S) + \mathrm{card}(S')$ (si S, S' ont des schémas disjoints).

Produit cartésien (2)

Elève		
Nom	Prénom	Classe
Dupont	Jean	MPSI 1
Michel	Jacques	MPSI 1

Prof	
Nom	Matière
Tartempion	Maths
Duchmol	Anglais
Schprountz	Allemand

- Faire le produit cartésien

Produit cartésien (2)

- Il faut d'abord renommer pour avoir des schémas disjoints :

$\texttt{Elève} \times \rho_{Nom \leftarrow Lehrer}(\texttt{Prof})$				
Nom	Prénom	Classe	Lehrer	Matière
Dupont	Jean	MPSI 1	Tartempion	Maths
Michel	Jeacques	MPSI 1	Duchmol	Anglais
Dupont	Jean	MPSI 1	Duchmol	Anglais
Michel	Jeacques	MPSI 1	Tartempion	Maths
Dupont	Jean	MPSI 1	Schprountz	Allemand
Michel	Jeacques	MPSI 1	Schprountz	Allemand

Produit cartésien : jointure croisée

- Il existe un opérateur CROSS JOIN (pas universellement reconnu)p mais la syntaxe suivante est universelle :

```
--produit cartésien de deux tables
SELECT * FROM table1 , table2
```

Produit cartésien : jointure croisée

- Il existe un opérateur CROSS JOIN (pas universellement reconnu)p mais la syntaxe suivante est universelle :

```sql
--produit cartésien de deux tables
SELECT * FROM table1 , table2
```

- Récupérer tous les tuples (fournisseur,expéditeur) :

```sql
SELECT * FROM Suppliers, Shippers;
--87 résultats
```

Remarque : le cardinal du produit cartésien est le produit des cardinaux :

```sql
SELECT COUNT(*) FROM Suppliers
UNION
SELECT COUNT(*) FROM Shippers
-- donne 3 et 29
```

Complément de notations

- $\pi_A(R)$ représente la projection de la relation R sur les colonnes de l'ensemble d'attributs A (on conserve toutes les lignes, on ne garde que certaine colonnes).

Complément de notations

- $\pi_A(R)$ représente la projection de la relation R sur les colonnes de l'ensemble d'attributs A (on conserve toutes les lignes, on ne garde que certaine colonnes).
- $\sigma_T(R)$ est la sélection de R selon le test T (on conserve toutes les colonnes, on ne garde que les lignes qui vérifient le test).

Exercice

On suppose que toutes les personnes (client ou fournisseur) ont renseigné l'attribut `City` dans la BDD `Northwind` du W3C.
Calculer le nombre de personnes (client ou fournisseur) qui vivent dans une ville hébergeant au moins deux personnes (client ou fournisseur).

Exercice

On suppose que toutes les personnes (client ou fournisseur) ont renseigné l'attribut `City` dans la BDD `Northwind` du W3C.
Calculer le nombre de personnes (client ou fournisseur) qui vivent dans une ville hébergeant au moins deux personnes (client ou fournisseur).

Indication

Il s'agit de faire la différence entre le cardinal de la réponse d'un `UNION ALL` avec celui de la réponse d'un `UNION`

Indication

Penser à un produit cartésien.

1. Opérateurs ensemblistes usuels

2. Produit cartésien, jointure
   - Produit cartésien
   - Division cartésienne

Division cartésienne

`Notion non explicitement au programme ITC et MPII.` (laissée pour info).
Soient deux tables R, S de schéma A et B telles que $B \subset A$. On pose $C = A - B$ (en notant la soustraction ensembliste comme la différence des entiers).

Définition

On appelle *division cartésienne* de R par S, la plus grande table T de shéma C telle que $S \times T \subset R$. On note $T = R \div S$

En notant π_C la projection sur C :

- On pose $T_1 = \pi_C(R)$. T_1 a C pour schéma.

Division cartésienne

`Notion non explicitement au programme ITC et MPII.` (laissée pour info).

Soient deux tables R, S de schéma A et B telles que $B \subset A$. On pose $C = A - B$ (en notant la soustraction ensembliste comme la différence des entiers).

Définition

On appelle *division cartésienne* de R par S, la plus grande table T de shéma C telle que $S \times T \subset R$. On note $T = R \div S$

En notant π_C la projection sur C :

- On pose $T_1 = \pi_C(R)$. T_1 a C pour schéma.
- Soit $T_2 = \pi_C((S \times T_1) - R)$.

Division cartésienne

`Notion non explicitement au programme ITC et MPII.` (laissée pour info).

Soient deux tables R, S de schéma A et B telles que $B \subset A$. On pose $C = A - B$ (en notant la soustraction ensembliste comme la différence des entiers).

Définition

On appelle *division cartésienne* de R par S, la plus grande table T de shéma C telle que $S \times T \subset R$. On note $T = R \div S$

En notant π_C la projection sur C :

- On pose $T_1 = \pi_C(R)$. T_1 a C pour schéma.
- Soit $T_2 = \pi_C((S \times T_1) - R)$.
- Alors $T = T_1 - T_2$

Division cartésienne

`Notion non explicitement au programme ITC et MPII.` (laissée pour info).
Soient deux tables R, S de schéma A et B telles que $B \subset A$. On pose $C = A - B$ (en notant la soustraction ensembliste comme la différence des entiers).

Définition

On appelle *division cartésienne* de R par S, la plus grande table T de shéma C telle que $S \times T \subset R$. On note $T = R \div S$

En notant π_C la projection sur C :

- On pose $T_1 = \pi_C(R)$. T_1 a C pour schéma.
- Soit $T_2 = \pi_C((S \times T_1) - R)$.
- Alors $T = T_1 - T_2$
- La division cartésienne est donc plus petite que la projection sur C

Division cartésienne

`Notion non explicitement au programme ITC et MPII.` (laissée pour info).

Soient deux tables R, S de schéma A et B telles que $B \subset A$. On pose $C = A - B$ (en notant la soustraction ensembliste comme la différence des entiers).

Définition

On appelle *division cartésienne* de R par S, la plus grande table T de shéma C telle que $S \times T \subset R$. On note $T = R \div S$

En notant π_C la projection sur C :

- On pose $T_1 = \pi_C(R)$. T_1 a C pour schéma.
- Soit $T_2 = \pi_C((S \times T_1) - R)$.
- Alors $T = T_1 - T_2$
- La division cartésienne est donc plus petite que la projection sur C
- La division est utilisée pour répondre à des requêtes de type : « quels sont les produits achetés par tous les clients ? »

Division cartésienne

- R’

Nom	Classe
Dupont	MPSI 1
Martin	MPSI 1
Bernard	PCSI 2

R

Nom	Classe	Nom-Prof
Dupont	MPSI 1	Tartempion
Martin	MPSI 1	Tartempion
Bernard	PCSI 2	Tartempion
Dupont	MPSI 1	Duchmol
Bernard	PCSI 2	Duchmol

- Chercher la division cartésienne `DC` de `R` par `R’`.

Division cartésienne

R'	
Nom	Classe
Dupont	MPSI 1
Martin	MPSI 1
Bernard	PCSI 2

R		
Nom	Classe	Nom-Prof
Dupont	MPSI 1	Tartempion
Martin	MPSI 1	Tartempion
Bernard	PCSI 2	Tartempion
Dupont	MPSI 1	Duchmol
Bernard	PCSI 2	Duchmol

- `R÷R'` a pour schéma le seul attribut qui est dans le schéma de `R` et pas dans celui de `R'`.

Division cartésienne

R'	
Nom	Classe
Dupont	MPSI 1
Martin	MPSI 1
Bernard	PCSI 2

R		
Nom	Classe	Nom-Prof
Dupont	MPSI 1	Tartempion
Martin	MPSI 1	Tartempion
Bernard	PCSI 2	Tartempion
Dupont	MPSI 1	Duchmol
Bernard	PCSI 2	Duchmol

- Posons `DC`$=$ {Tartempion, Duchmol}, {(Dupont,MPSI 1),(Martin, MPSI 2),(Bernard, PCSI 2)} $\times$ DC a 6 lignes, donc plus que nécessaire. Alors `DC` est trop gros.

Division cartésienne

R'	
Nom	Classe
Dupont	MPSI 1
Martin	MPSI 1
Bernard	PCSI 2

R		
Nom	Classe	Nom-Prof
Dupont	MPSI 1	Tartempion
Martin	MPSI 1	Tartempion
Bernard	PCSI 2	Tartempion
Dupont	MPSI 1	Duchmol
Bernard	PCSI 2	Duchmol

- Avec `DC`$=$ {Duchmol},
  {(Dupont,MPSI 1),(Martin, MPSI 2),(Bernard, PCSI 2)} $\times$ `DC`
  contient la ligne (Martin, MPSI 1, Duchmol) qui n'est pas dans `Eleve-Prof`.

Division cartésienne

R'	
Nom	**Classe**
Dupont	MPSI 1
Martin	MPSI 1
Bernard	PCSI 2

R		
Nom	**Classe**	**Nom-Prof**
Dupont	MPSI 1	Tartempion
Martin	MPSI 1	Tartempion
Bernard	PCSI 2	Tartempion
Dupont	MPSI 1	Duchmol
Bernard	PCSI 2	Duchmol

  Réponse : `DC` vaut

DC=R÷R'
Nom-Prof
Tartempion

R''		
Nom-Eleve	**Classe**	**Nom-Prof**
Dupont	MPSI 1	Duchmol
Bernard	PCSI 2	Duchmol

  On a $(\mathtt{R}\div\mathtt{R'}\times\mathtt{R'})\cup\mathtt{R''} = \mathtt{R}$.

Un exemple plus élaboré

FIGURE – Une table

La table Film

Titre	Directeur	Acteur
Mais qui a tué Harry ?	Hitchcock	Gwenn
Mais qui a tué Harry ?	Hitchcock	Forsythe
Mais qui a tué Harry ?	Hitchcock	MacLaine
Mais qui a tué Harry ?	Hitchcock	Hitchcock
...		
Cris et chuchotements	Bergman	Andersson
Cris et chuchotements	Bergman	Sylwan
Cris et chuchotements	Bergman	Thulin
Cris et chuchotements	Bergman	Ullman

Question : quels sont les acteurs qui ont tourné dans tous les films de Hitchcock ? (on ne dispose pas du quantificateur universel).

Un exemple plus élaboré (suite)

- Films tournés par Hitchcock :
  $T_H = \pi_{Titre,Directeur}(\sigma_{Directeur='Hitchcock'}(Film))$. La table T_H admet (Titre, Directeur) pour schéma et est constituée de couples (titre, Hitchcock).

Un exemple plus élaboré (suite)

- Films tournés par Hitchcock : $T_H = \pi_{Titre,Directeur}(\sigma_{Directeur='Hitchcock'}(Film))$. La table T_H admet (Titre, Directeur) pour schéma et est constituée de couples (titre, Hitchcock).
- Tous les acteurs : $A = \pi_{Acteur}(Film)$. On cherche A_2 : l'ensemble des acteurs qui ont tourné dans tous les films de Hitchcock. C'est le plus grand sous-ensemble de A tel que $T_H \times A_2 \subset$ Film, donc $A_2 =$ Film $\div$ T_H .

Un exemple plus élaboré (suite)

- Films tournés par Hitchcock :
  $T_H = \pi_{Titre,Directeur}(\sigma_{Directeur='Hitchcock'}(Film))$. La table T_H admet (Titre, Directeur) pour schéma et est constituée de couples (titre, Hitchcock).
- Tous les acteurs : $A = \pi_{Acteur}(Film)$. On cherche A_2 : l'ensemble des acteurs qui ont tourné dans tous les films de Hitchcock. C'est le plus grand sous-ensemble de A tel que $T_H \times A_2 \subset$ Film, donc
  $A_2 =$ Film $\div$ T_H .
- Remarque : Tous les acteurs dans A_2 ont tourné dans tous les films de Hitchcock et un acteur qui n'est pas dans A_2 a manqué au moins un film de Hitchcock.

Un exemple plus élaboré (suite)

- $T_H \times A$: toutes les associations (titre, Hitchcock, acteur) possibles de titres de Hitchcock avec un acteur, même celles qui n'existent pas.

Un exemple plus élaboré (suite)

- $T_H \times A$: toutes les associations (titre, Hitchcock, acteur) possibles de titres de Hitchcock avec un acteur, même celles qui n'existent pas.
- $(T_H \times A) -$ Film : seulement les associations (titre, Hitchcock, acteur) qui n'existent pas. Si un nom d'acteur et un film figurent dans une même ligne de cette table, cet acteur n'a pas tourné dans ce film d'Hitchcock.

Un exemple plus élaboré (suite)

- $T_H \times A$: toutes les associations (titre, Hitchcock, acteur) possibles de titres de Hitchcock avec un acteur, même celles qui n'existent pas.
- $(T_H \times A) - \text{Film}$: seulement les associations (titre, Hitchcock, acteur) qui n'existent pas. Si un nom d'acteur et un film figurent dans une même ligne de cette table, cet acteur n'a pas tourné dans ce film d'Hitchcock.
- $A' = \pi_{Acteur}((T_H \times A) - \text{Film})$: les acteurs qui n'ont pas tourné dans au moins un film d'Hitchcock.

Un exemple plus élaboré (suite)

- $T_H \times A$: toutes les associations (titre, Hitchcock, acteur) possibles de titres de Hitchcock avec un acteur, même celles qui n'existent pas.
- $(T_H \times A) - \text{Film}$: seulement les associations (titre, Hitchcock, acteur) qui n'existent pas. Si un nom d'acteur et un film figurent dans une même ligne de cette table, cet acteur n'a pas tourné dans ce film d'Hitchcock.
- $A' = \pi_{Acteur}((T_H \times A) - \text{Film})$: les acteurs qui n'ont pas tourné dans au moins un film d'Hitchcock.
- $A_2 = A - A'$: les acteurs qui ont tourné dans tous les films d'Hitchcok. $A_2 = \text{Film} \div \pi_{Titre,Directeur}(\sigma_{Directeur='Hitchcock'}(\text{Film}))$

Jointure symétrique

Recollement de deux relations

Définition

Soient $R(S)$ et $R'(S')$ deux relations de schémas disjoints et $A \in S$, $A' \in S'$ deux attributs de même domaine. On appelle *jointure symétrique de R et R' selon* (A, A') et on note $R \bowtie_{[A=A']} R'$ les éléments e de $R \times R'$ tels que $e.A = e.A'$

$$R \bowtie_{[A=A']} R' := \sigma_{A=A'} R \times R' = \left\{ e \in R \times R' \mid e.A = e.A' \right\}$$

- De la même façon qu' on peut se passer en logique classique de l'opérateur $\implies$ (défini avec $\vee$ et $\neg$), on n'a pas besoin en théorie de l'opérateur de jointure $\bowtie_{[\star=\square]}$ car il est défini comme composée d'autres opérateurs.

Jointure symétrique

Recollement de deux relations

Définition

Soient $R(S)$ et $R'(S')$ deux relations de schémas disjoints et $A \in S$, $A' \in S'$ deux attributs de même domaine. On appelle *jointure symétrique de R et R' selon* (A, A') et on note $R \bowtie_{[A=A']} R'$ les éléments e de $R \times R'$ tels que $e.A = e.A'$

$$R \bowtie_{[A=A']} R' := \sigma_{A=A'} R \times R' = \left\{e \in R \times R' \mid e.A = e.A'\right\}$$

- De la même façon qu' on peut se passer en logique classique de l'opérateur $\implies$ (défini avec $\vee$ et $\neg$), on n'a pas besoin en théorie de l'opérateur de jointure $\bowtie_{[\star=\square]}$ car il est défini comme composée d'autres opérateurs.
- Mais se serait une mauvaise idée en terme de complexité : coût en $O(|R| \times |R'|)$ tests alors qu'il existe des implantations utilisant des index avec un coût linéaire en nombre de tests.

Jointure symétrique

Recollement de deux relations

- Quel est le schéma de $R \bowtie_{[A=A']} R'$?

Jointure symétrique

Recollement de deux relations

- Quel est le schéma de $R \bowtie_{[A=A']} R'$?
- La jointure est une sélection du produit cartésien $R \times R'$ donc : $S + S'$.

Jointure symétrique (2)

Livre	
Titre	Nom-auteur
Madame Bovary	Flaubert
Le père Goriot	Balzac (de)

Auteur	
Nom	Prénom
Flaubert	Gustave
Balzac (de)	Honoré

- Réaliser la jointure symétrique selon (`Nom-auteur,Nom`).
-

Jointure symétrique (2)

Livre	
Titre	**Nom-auteur**
Madame Bovary	Flaubert
Le père Goriot	Balzac (de)

Auteur	
Nom	**Prénom**
Flaubert	Gustave
Balzac (de)	Honoré

- Réaliser la jointure symétrique selon (`Nom-auteur`,`Nom`).

$\texttt{Livre} \bowtie_{[\texttt{Nom-auteur}=\texttt{Nom}]} \texttt{Auteur}$			
Titre	**Nom-auteur**	**Nom**	**Prénom**
Madame Bovary	Flaubert	Flaubert	Gustave
Le père Goriot	Balzac (de)	Balzac (de)	Honoré

-

Jointure symétrique (2)

Livre	
Titre	**Nom-auteur**
Madame Bovary	Flaubert
Le père Goriot	Balzac (de)

Auteur	
Nom	**Prénom**
Flaubert	Gustave
Balzac (de)	Honoré

- Réaliser la jointure symétrique selon (`Nom-auteur`,`Nom`).
- Il y a des doublons de valeurs. Comment éviter cela ?

Jointure symétrique (2)

Livre	
Titre	Nom-auteur
Madame Bovary	Flaubert
Le père Goriot	Balzac (de)

Auteur	
Nom	Prénom
Flaubert	Gustave
Balzac (de)	Honoré

- Réaliser la jointure symétrique selon (`Nom-auteur`,`Nom`).
- Il y a des doublons de valeurs. Comment éviter cela ?

$\pi_{\texttt{Titre,Nom,Prénom}}(\texttt{Livre} \bowtie_{[\texttt{Nom-auteur}=\texttt{Nom}]} \texttt{Auteur})$		
Titre	Nom	Prénom
Madame Bovary	Flauber	Gustave
Le père Goriot	Balzac (de)	Honoré

Opérations ensemblistes

Jointure symétrique

Pour croiser des informations en provenance de plusieurs tables :

- Principe : récupérer les lignes de deux tables lorsque ces lignes ont une caractéristique commune.

Opérations ensemblistes

Jointure symétrique

Pour croiser des informations en provenance de plusieurs tables :

- Principe : récupérer les lignes de deux tables lorsque ces lignes ont une caractéristique commune.
- En théorie, ces tables ont des shémas disjoints.

Opérations ensemblistes

Jointure symétrique

Pour croiser des informations en provenance de plusieurs tables :

- Principe : récupérer les lignes de deux tables lorsque ces lignes ont une caractéristique commune.
- En théorie, ces tables ont des shémas disjoints.
- Syntaxe :

```
SELECT * FROM table1
INNER JOIN table2
ON table1.colt1 = table2.colt2
```

Opérations ensemblistes

Jointure symétrique

Pour croiser des informations en provenance de plusieurs tables :

- Principe : récupérer les lignes de deux tables lorsque ces lignes ont une caractéristique commune.
- En théorie, ces tables ont des shémas disjoints.
- Syntaxe :

```
SELECT * FROM table1
INNER JOIN table2
ON table1.colt1 = table2.colt2
```

- Noms des clients et leurs numéros d'achats :

```
SELECT Orders.OrderID, Customers.CustomerName
FROM Orders
INNER JOIN Customers
ON Orders.CustomerID = Customers.CustomerID;
```

Jointure et intersection

- On peut se servir d'une jointure pour obtenir l'intersection de deux tables de même schéma.

Jointure et intersection

- On peut se servir d'une jointure pour obtenir l'intersection de deux tables de même schéma.
- Donner les villes où il y a des clients et des fournisseurs :

```
SELECT C.City FROM Customers AS C
INNER JOIN Suppliers AS S
ON S.City = C.City
```

Jointures `LEFT JOIN`

Hors programme ITC

- La jointure que nous avons vue `R1 JOIN R2 ON A=B` est symétrique : les enregistrements de `R1` qui ont une valeur de `A` qui n'existe pas dans `D` n'apparaissent pas dans le résultat.

Jointures `LEFT JOIN`

Hors programme ITC

- La jointure que nous avons vue `R1 JOIN R2 ON A=B` est symétrique : les enregistrements de `R1` qui ont une valeur de `A` qui n'existe pas dans `D` n'apparaissent pas dans le résultat.
- Parfois on a besoin de faire apparaître, en plus du résultat de la jointure symétrique, les valeurs de `R1` : c'est la *jointure gauche* (`LEFT JOIN`)

Jointures LEFT JOIN

Hors programme ITC

- La jointure que nous avons vue R1 JOIN R2 ON A=B est symétrique : les enregistrements de R1 qui ont une valeur de A qui n'existe pas dans D n'apparaissent pas dans le résultat.
- Parfois on a besoin de faire apparaître, en plus du résultat de la jointure symétrique, les valeurs de R1 : c'est la *jointure gauche* (LEFT JOIN)
- Donner le nom et la ville de chaque client et compléter l'information par le nom des fournisseurs qui vivent dans la même ville que lui.

```sql
SELECT C.CustomerName, C.City, S.SupplierName
FROM Customers AS C LEFT JOIN Suppliers as S
ON C.City=S.City
```

Remarque : certaines des lignes obtenues sont complètes, d'autres pas.

Jointure RIGHT JOIN

Hors programme ITC

- Il existe de même une *jointure droite* de mot clé `RIGHT JOIN`

On constate que le pauvre Adam West n'a effectué aucune vente.

Jointure RIGHT JOIN

Hors programme ITC

- Il existe de même une *jointure droite* de mot clé RIGHT JOIN
- Donner toutes les informations sur les employés et les ventes qu'ils ont éventuellement assurées.

```
SELECT Orders.OrderID, Employees.LastName,
       Employees.FirstName
FROM Orders
RIGHT JOIN Employees
ON Orders.EmployeeID = Employees.EmployeeID
ORDER BY Orders.OrderID;
```

On constate que le pauvre Adam West n'a effectué aucune vente.

Auto-jointure

- Donner tous les couples de clients qui vivent dans la même ville. On n'accepte pas les « doublons » comme (Dupont, Durand) et (Durand, Dupont) ni les identifiants identiques comme (Dupont, Dupont) sauf si il s'agit de personnes différentes (le nom n'est pas une clé primaire).

Auto-jointure

- Donner tous les couples de clients qui vivent dans la même ville. On n'accepte pas les « doublons » comme (Dupont, Durand) et (Durand, Dupont) ni les identifiants identiques comme (Dupont, Dupont) sauf si il s'agit de personnes différentes (le nom n'est pas une clé primaire).
- Solution :

Jointure sur plus de deux tables

Pour chaque client, indiquer ses achats et l'entreprise qui a livré ces achats :

```
SELECT Orders.OrderID, Customers.CustomerName,
    Shippers.ShipperName
FROM ((Orders
INNER JOIN Customers
ON Orders.CustomerID = Customers.CustomerID)
INNER JOIN Shippers
ON Orders.ShipperID = Shippers.ShipperID);
```

Remarque

Les SGBD peuvent être plus ou moins permissifs sur le parenthésage. Mais l'opération de jointure interne est associative.