

Bases de données – Niveau 1

SQL - MySQL – MariaDB - Cours et TP 8

Jointures externes et SQL 2

join – on – using – left - right

Site officiel MariaDB : <https://mariadb.org/>

Site officiel MySQL : <http://www-fr.mysql.com/>

Documentations MySQL : <http://dev.mysql.com/doc/index.html>

Mémo SQL : http://www.volubis.fr/bonus/SQL_memo.htm

Documentations MySQL 5.0 en français : http://foad.univ-ubs.fr/pluginfile.php/1779/mod_resource/content/0/refman-5.0-fr.pdf

Bertrand LIAUDET

SOMMAIRE

SOMMAIRE	1
SQL : CONSULTATION DE LA BD - SUITE	3
1. Produits cartésiens et jointures en SQL 2	3
Les produits cartésiens - join	3
Les jointures naturelles – on, using	3
Exemples avec plusieurs tables	4
Remarque sur le natural join	4
2. Jointures externes – left, right	5
Présentation de la jointure externe	5
1 ^{er} usage : s'intéresser à tous les tuples de la table maître : left join	7
2 ^{ème} usage : s'intéresser aux tuples éliminés de la table jointe : right join	10
3 ^{ème} usage : soustraction ensembliste à partir d'une jointure artificielle : left join	12
TP N°8 : JOINTURES EXTERNES ET TOUT TYPES DE REQUETES	13
Travail à faire	13
Méthode de travail	13
Ordre de projection des attributs	13
Exercice 1 : usages de SQL 2	13
Exercice 2 : Jointures externes	14
Exercice 3 : BD ecoling	15
Présentation	15

Chargement et analyse de la BD

15

Interrogation de la BD

15

Dernière édition : juillet 2015

SQL : CONSULTATION DE LA BD - SUITE

PRINCIPALES NOTIONS

Left join
Outer join
Join
On

Right join
Inner join
Natural join
Using

1. Produits cartésiens et jointures en SQL 2

Les produits cartésiens - join

Syntaxe classique :

```
Select *  
from emp, dept ;
```

Le from peut être remplacé par :

```
Select *  
from emp join dept ;
```

ou

```
Select *  
from emp cross join dept ;
```

ou

```
Select *  
from emp inner join dept ;
```

Les jointures naturelles – on, using

Syntaxe classique :

```
Select e.ne, e.nom, e.nd, d.nomd  
from emp e, dept d  
where e.nd = d.nd;
```

Le from peut être remplacé par :

```
from emp e join dept d
```

ou

```
from emp e cross join dept d
```

ou

```
from emp e inner join dept d
```

Le where peut être remplacé par :

```
on e.nd = d.nd;
```

ou

```
using (nd);
```

Le from et le where peuvent être remplacés par :

```
from emp e natural join dept d;
```

Exemples avec plusieurs tables

Syntaxe classique :

```
Select e.datemp, e.nl, l.editeur, a.na, a.nom  
from emprunter e, adherents a, livres l  
where e.na = a.na  
and e.nl = l.nl  
and e?dateRet is null;
```

Avec des inner join

```
Select datemp, nl, editeur, na, nom  
from emprunter inner join adherents using (na)  
inner join livres using(nl)  
where e?dateRet is null;
```

Avec des natural join

```
Select datemp, nl, editeur, na, nom  
from emprunter natural join adherents  
natural join livres  
where e?dateRet is null;
```

Remarque sur le natural join

Dans l'exemple avec emp et dept, on a pris soin de différencier le code du nom de l'employé (nom) et le code du nom du département (nomd).

C'est obligatoire avec un « natural join ».

En effet, le natural join fait des restrictions de jointure avec tous les attributs de même nom des deux tables proposées. Donc avec nd, mais aussi avec nom le nom est le même dans les tables emp et dept. Dans ce cas, le résultat est vide ! **Le principe mathématique abstrait de la jointure naturelle n'est donc pas adaptée aux relations concrètes** qui peuvent avoir des attributs de même noms sans que ces attributs soient identiques.

On évitera donc toujours les « natural join » car l'ajout d'un nouvel attribut dans une table pourrait engendrer des erreurs sur des requêtes existant déjà.

2. Jointures externes – left, right

Toutes les possibilités syntaxiques qu'on vient d'aborder n'apportent rien de plus par rapport à la syntaxe standard du SQL 1.

L'intérêt de cette syntaxe est d'ouvrir à la notion de jointure externe : outer join (par opposition à inner join).

Présentation de la jointure externe

Définition

Une jointure externe est une jointure qui permet de rajouter au résultat de la jointure les tuples éliminés d'une des deux tables de la jointure.

Jointure à gauche et jointure à droite

Selon que l'on rajoute les tuples de la première table ou de la deuxième (première ou deuxième dans l'ordre de déclaration), on parlera de jointure externe à gauche (première table, à gauche du join) ou à droite (deuxième table, à droite du join).

Syntaxe de la jointure externe à gauche

Après la jointure, on rajoute les tuples de la première table (celle de gauche) :

```
Select *  
from t1 left outer join t2  
on t1.att = t2.att;
```

ou encore, le « outer » étant facultatif :

```
Select *  
from t1 left join t2  
on t1.att = t2.att;
```

Syntaxe de la jointure externe à droite

Après la jointure, on rajoute les tuples de la deuxième table (celle de droite) :

```
Select *  
from t1 right outer join t2  
on t1.att = t2.att;
```

ou encore, le « outer » étant facultatif :

```
Select *  
from t1 right join t2  
on t1.att = t2.att;
```

Non commutativité de la jointure externe

A la différence de la jointure interne et du produit cartésien, la jointure externe n'est pas commutative :

```
Select *  
from t1 right join t2  
on t1.att = t2.att;
```

n'est pas équivalent à :

```
Select *  
from t1 right join t2  
on t1.att = t2.att;
```

Semi-commutativité de la jointure externe

Le « right join » est équivalent au « left join » si on inverse l'ordre des tables.

```
Select *  
from t1 right join t2  
on t1.att = t2.att;
```

est équivalent à :

```
Select *  
from t2 left join t1  
on t1.att = t2.att;
```

Remarque syntaxique

```
Select *  
from t1 outer join t2 // BUG !!! il faut préciser gauche ou droite  
on t1.att = t2.att;
```

La requête précédente ne veut rien dire! Il faut nécessairement préciser right ou left pour que la requête soit correcte.

Ordre des opérations élémentaires dans une jointure externe

```
Select *  
from t1 right join t2  
on t1.att = t2.att  
where tx.attx = valeur ;
```

L'ordre de traitements des opérations élémentaires est le suivant :

- 1) produit cartésien : **from**
- 2) restriction de jointure : **on**
- 3) jointure externe (ajout des tuples éliminés) : **right** (ou left)
- 4) restriction spécifique : **where**

On comprend ici qu'on pourra mettre des restrictions spécifiques dans le ON pour permettre

Les 3 principaux usages de la jointure externe

La jointure externe a trois usages principaux :

- Le premier usage concerne des **jointures naturelles** avec une table maître et une table jointe. Il consiste à rajouter les tuples éliminés de la table maître et à **s'intéresser à tous les tuples de la table maître**.
- Le deuxième usage concerne des **jointures naturelles** avec une table maître et une table jointe. Il consiste à rajouter les tuples éliminés de la table jointe et à **s'intéresser uniquement aux tuples éliminés de la table jointe. C'est une soustraction ensembliste**.
- Le troisième concerne les **jointures artificielles**. Il reprend le second dans le sens où il s'agit d'une soustraction ensembliste, mais à partir d'une jointure artificielle et avec jointure externe à gauche.

1^{er} usage : s'intéresser à tous les tuples de la table maître : left join

Formalisme : jointure externe à gauche

La forme générale d'une jointure externe à gauche est :

```
Select TJ.NTJ, TJ.CS, TM.NTM, TM.CS
from table_maître TM left join table_jointe TJ
on TM.NTJ = TJ.NTJ;
```

Avec CS : clé significative.

En mettant la table maître en premier, on a une jointure à gauche.

Exemple

Question traitée : Afficher les employés avec leurs supérieurs hiérarchiques et les employés sans supérieurs hiérarchique :

```
Select e.ne, e.nom, echef.ne, echef.nom
from emp e left join emp echef
on e.nechef = echef.ne ;
```

Résultats :

```
+-----+-----+-----+-----+
| ne     | nom      | ne     | nom      |
+-----+-----+-----+-----+
| 7369   | SMITH    | 7902   | FORD     |
| 7499   | ALLEN    | 7698   | BLAKE    |
| 7521   | WARD     | 7698   | BLAKE    |
| 7566   | JONES    | 7839   | KING     |
| 7654   | MARTIN   | 7698   | BLAKE    |
| 7698   | BLAKE    | 7839   | KING     |
| 7782   | CLARK    | 7839   | KING     |
| 7788   | SCOTT    | 7566   | JONES    |
| 7839 | KING   | NULL  | NULL   |
| 7844   | TURNER   | 7698   | BLAKE    |
| 7876   | ADAMS    | 7788   | SCOTT    |
| 7900   | JAMES    | 7698   | BLAKE    |
| 7902   | FORD     | 7566   | JONES    |
| 7934   | MILLER   | 7782   | CLARK    |
+-----+-----+-----+-----+
14 rows in set (0.00 sec)
```

Requête équivalente : UNION

La jointure externe à gauche classique permet de remplacer une union.

```
Select e.ne, e.nom, echef.ne, echef.nom
from emp e join emp echef
on e.nechef = echef.ne
union
Select ne, nom, null, null // on ajoute deux attributs à à emp
from emp
where nechef is null
```

;

Clé primaire

Clé primaire = Clé Primaire (Table Maître)

On retombe forcément sur la clé primaire de la table maître.

Nombre de tuples

Nb Tuples = Nb Tuples (Table Maître)

On retombe forcément sur le nombre de tuples de la table maître.

Du bon usage

Optimisation d'exécution

La jointure est théoriquement préférable aux opérateurs ensemblistes classiques, donc à l'union.

Optimisation de maintenance

La jointure externe « à gauche » est préférable à l'union qui est compliquée à mettre en œuvre.

A noter que l'usage de la jointure externe « à gauche » peut être généralisé par défaut pour toutes les jointures naturelles afin de ne pas éliminer les tuples dont la clé étrangère vaut NULL.

2^{ème} usage : s'intéresser aux tuples éliminés de la table jointe : right join

Formalisme : jointure externe à droite

La forme générale d'une jointure externe à droite est :

```
Select TJ.*  
from table_maitresse TM right join table_jointe TJ  
on TM.NTJ = TJ.NTJ  
where TM.NTM is NULL;
```

En mettant la table maître en premier, on a une jointure à droite.

Exemple

Question traitée : quels sont les départements vides :

```
Select d.nd, d.nom, d.ville  
from emp e right join dept d  
on e.nd = d.nd  
where e.ne is NULL;
```

Résultats :

```
+-----+-----+-----+  
| nd | nomd          | ville |  
+-----+-----+-----+  
| 40 | OPERATIONS   | BOSTON |  
+-----+-----+-----+  
1 row in set (0.01 sec)
```

Requête équivalente : MINUS et INTERSECT, NOT IN, NOT EXISTS

La jointure externe à droite classique permet de remplacer les opérations ensemblistes MINUS et INTERSECT, et leurs équivalents en NOT IN et NOT EXISTS

➤ *Version MINUS (ne marche pas avec MySQL)*

```
Select *  
from dept  
minus  
select dept.*  
from emp e, dept d  
where e.ND = d.ND  
;
```

➤ *Version imbriquée NOT IN*

```
Select *  
from dept  
where ND not in (  
    select ND from emp  
);
```

➤ *Version imbriquée NOT EXISTS*

```
Select *
```

```
from dept
where not exists (
  select * from emp
  where emp.ND = dept.ND
);
```

Clé primaire

Pas de clé primaire.

Nombre de tuples

$Nb \text{ Tuples}(\text{Table jointe}) \leq Nb \text{ Tuples} \leq Nb \text{ Tuples}(\text{Table Maître}) + Nb \text{ Tuples}(\text{Table jointe}) - 1$

Si aucun tuple de la table maître ne pointe sur un tuple de la table jointe, la jointure naturelle donne un ensemble vide et la jointure externe rajoute tous les tuples de la table jointe.

Si tous les tuples de la table maître pointent sur le même tuple de la table jointe, la jointure naturelle donne tous les tuples de la table maître sur-informés. La jointure externe ajoute tous les tuples de la table jointe moins 1.

Du bon usage

Optimisation d'exécution

La jointure est théoriquement préférable à la requête imbriquée.

Optimisation de maintenance

La jointure externe étant un peu compliquée à comprendre, l'optimisation de la maintenance peut conduire à préférer le select imbriqué à la jointure externe.

Exemple

Question traitée : tous les employés ayant un job différent de ceux du département 20

```
Select e1.*  
from emp e1 left join emp e2  
on e1.job = e2.job and e2.nd=20  
where e2.ne is NULL;
```

Principe :

D'abord, on inverse la question : tous les employés moins ceux qui ont le même job que ceux du département 20. C'est une jointure artificielle.

On croise tous les employés e1 avec ceux du département 20 (e2) et on garde ceux qui ont un job identique à un employé du département 20

La restriction de jointure et la restriction spécifique sont faites dans le « on ».

Ensuite, on prend le complément du résultat précédent, ce qui répond à la question. Le « left » rajoute les employés de e1 qui ont été éliminés : ceux qui ont un job différent de ceux du département 20.

Enfin, on restreint à ce seul complément avec le where.

Il faut donc faire bien attention à mettre toutes les restrictions souhaités dans le « on » avant que le « left » s'applique et rajoute les tuples éliminés.

Requête équivalente : MINUS, NOT IN, NOT EXISTS

La jointure externe à droite classique permet de remplacer les opérations ensemblistes MINUS et INTERSECT, et leurs équivalents en NOT IN et NOT EXISTS

➤ *Version imbriquée NOT IN*

```
Select *  
from emp  
where job not in (  
    select job from emp  
    where nd=20  
);
```

Du bon usage

Optimisation d'exécution

La jointure est théoriquement préférable à la requête imbriquée.

Optimisation de maintenance

La jointure externe étant un peu compliquée à comprendre, l'optimisation de la maintenance peut conduire à préférer le select imbriqué à la jointure externe.

TP N°8 : JOINTURES EXTERNES ET TOUT TYPES DE REQUETES

Travail à faire

- Dans un fichier texte à votre nom+TPX, écrire les questions et les réponses les unes à la suite des autres.
- Après chaque requête, on met, en commentaire, le nom de l'attribut clé primaire de la table résultat
- Mettez le résultat obtenu dans le fichier (copier-coller).

Méthode de travail

A partir du fichier « rapport de TP », faire des copier-coller dans la calculette SQL ou dans un fichier de test et lancer le script de test.

Ordre de projection des attributs

Requête sans fonction de groupe

Attributs de tri, Cle Primaire, Clé Significative, Attributs demandés, Attributs de restriction

Requête avec fonctions de groupe

Attributs de tri, attributs du group by, fonctions de groupe demandées, fonctions de groupe de restriction (du having).

Exercice 1 : usages de SQL 2

En utilisant les BD des TP précédents, répondre aux questions en utilisant des jointures externes :

1. Faire une jointure entre « emp » et « dept » en projetant tous les attributs et en utilisant un « join » puis un « inner join ». Essayer avec un « using », puis un « on », puis un « natural join ».
2. Faire une jointure à gauche entre « emp » et « dept » en projetant tous les attributs.
3. Faire une jointure à droite entre « emp » et « dept » en projetant tous les attributs.
4. Faire une jointure entre « emp » et « emp » (pour avoir les chefs) en projetant tous les attributs et en utilisant un « join » puis un « inner join ». Essayer avec un « using », puis un « on », puis un « natural join ».
5. Faire une jointure à gauche entre « emp » et « emp » (pour avoir les chefs) en projetant tous les attributs.
6. Faire une jointure à droite entre « emp » et « emp » (pour avoir les chefs) en projetant tous les attributs.

Exercice 2 : Jointures externes

On utilise les BD déjà utilisées dans les TP précédents : empdept et biblio.

7. Quels sont les départements vides ? (empdept)
8. Quels sont les employés qui ne sont pas supérieurs hiérarchiques ? (empdept)
9. Quels sont les adhérents qui n'ont jamais emprunté de livres ? (biblio)
10. Quels sont les livres qui n'ont jamais été empruntés ? (biblio)
11. Tous les employés n'ayant pas le même chef que ALLEN (empdept)
12. Combien d'exemplaires du titre : « Narcisse et Goldmund » sont disponibles ? (biblio)

Exercice 3 : BD ecoling

Présentation

Modèle relationnel

Etudiants (**NET**, nom, prenom, email)

Examens (**NEX**, matiere, prof, session, annee, niveau, type, duree)

Groupe (**NGR**, (promo, annee, groupe))

Epreuves (**NEP**, (#NGR, dateheure), salle, #NEX)

Evaluer (**#NET, #NEP**, note)

Participer (**#NET, #NGR**)

Dans la table Epreuves, (#NGR, dateheure) est une clé secondaire.

Dans la table Groupes, (promo, annee, groupe) est une clé secondaire.

Les Epreuves sont des instances d'un Examen

Les étudiants sont évalués dans une épreuve (qui correspond à un examen): il y a une note. En cas d'absence, ils ne sont pas dans la table.

Chargement et analyse de la BD

1. Télécharger le script de création de la BD : Ecoling.txt
2. Faire le graphe des tables.
3. Lancer ce script de création de la BD.

Interrogation de la BD

4. Quelles sont les épreuves qui n'ont pas donné lieu à des évaluations ?
5. Quels sont les étudiants qui n'ont pas été évalués pour l'épreuve n°2 et qui auraient du l'être (autrement dit qui participent au groupe correspondant à l'épreuve.

On a intérêt à traiter d'abord les questions suivantes :

- Tous les étudiants ayant passé l'épreuve 2
- Tous les étudiants du groupe correspondant à l'épreuve n°2

6. Quels sont les étudiants du groupe B, promo 2011 n'ayant pas passé toutes les épreuves de leur groupe ?

On a intérêt à traiter d'abord les questions suivantes :

1-a : Liste des étudiants du groupe B, promo 2011

1-b : Liste des épreuves auxquels les étudiants du groupe B, promo 2011 ont participé

1-c : Nombre d'épreuves auxquels les étudiants du groupe B promo 2011 ont participé

Puis :

2-a : La liste des épreuves du groupe B, promo 2011

2-b : Liste des épreuves du groupe avec les étudiants, triées par étudiant

2-c : Nombre d'épreuves par étudiants

7. Quels sont tous les étudiants n'ayant pas passé les épreuves de leur groupe ?

Consultation de la BD

8. Pour chaque table verbes, afficher son contenu en sur-informant au maximum la table. Pour faciliter la lecture, on projettera uniquement les clés primaires et significatives en faisant les tris judicieux.

9. Quel est le parcours d'un étudiant donné (le 30): quelles années, quelles promos, quels groupes ?

10. Quelles sont toutes les notes d'un étudiant donné (le 30), classé par année, matière et date ? Ajouter éventuellement quelques tuples pour mieux exemplifier la requête.

Questions statistiques

11. Quels sont les étudiants ayant plus que la moyenne des étudiants pour l'épreuve n°1 ?

12. Quelles sont les moyennes générales des étudiants par année ?

13. Quels sont les étudiants ayant plus que la moyenne des étudiants pour l'épreuve n°1 ?

14. Quelle est la moyenne des groupes pour les examens 4, 5 et 6. On affichera le numéro d'examen, le numéro du groupe, le nombre de notes, la moyenne des notes *et la liste des notes*.

15. Quelle est l'épreuve avec sa date et sa matière dans laquelle on a la meilleure moyenne ?

16. Pour chaque épreuve avec le nom de l'examen, quels sont les élèves qui ont la meilleure note ?

17. Par promo, quels sont les élèves dont la moyenne par matière est toujours supérieure à 10 ?

On commencera par récupérer les moyennes <10 par matières.

Autres questions

13. Quelles sont les durées des épreuves orales ?

14. Quels sont les noms des groupes passant des épreuves dans la salle 301 ?

15. Quels sont les noms des groupes passant une épreuve de maths ?

16. Quels sont les noms et prénoms des élèves qui ne sont pas de niveau 1 ?

17. Quels sont les étudiants n'ayant pas passé l'épreuve numéro 2 ?

18. Quels sont les numéros des étudiants n'ayant pas passé les épreuves de leur groupe ?

19. Quelle est la moyenne générale de l'étudiant n°1624

20. Quelle est la salle dans laquelle se déroule la plus longue épreuve ?

21. Quelle est la moyenne de l'étudiant n°2 pour les examens 4, 5 et 6

22. Quels sont les étudiants ayant plus que la moyenne des étudiants pour l'épreuve n°1 ?

23. Quelle est l'épreuve dans laquelle on a la meilleure moyenne ?

24. Pour chaque épreuve, quel est l'élève qui a la meilleure note ?

25. Quels sont les élèves dont la moyenne est toujours supérieure à 10 ?

26. Quels sont les élèves qui n'ont pas de notes à l'épreuve n°2 ?