

Requêtes avec jointures

Une jointure est une opération relationnelle qui sert à chercher des lignes ou des enregistrements à partir de deux ou plusieurs tables disposant d'un ensemble de valeur communes, en général les clés primaires.

Produit cartésien

Le produit cartésien est une requête de sélection qui met en jeux plusieurs tables. Pour deux tables, la sélection consiste à afficher la première ligne de la première table avec toutes les lignes de la deuxième table, puis la deuxième ligne de la première table avec toutes les lignes de la deuxième table et ainsi de suite. Ce type de sélection implique beaucoup de redondances.

Exemple

```
SELECT NOM, PRENOM, NOMPROG  
FROM ETUDIANTS,PROGRAMME;
```

NOM	PRENOM	NOMPROG
PATOCHE	ALIAN	INFORMATIQUE
PATOCHE	ALIAN	ADMINISTRATION
PATOCHE	ALIAN	ELECTRONIQUE
FAFAR	CHANTALE	INFORMATIQUE
FAFAR	CHANTALE	ADMINISTRATION
FAFAR	CHANTALE	ELECTRONIQUE
MARTIN	RACHA	INFORMATIQUE
MARTIN	RACHA	ADMINISTRATION
MARTIN	RACHA	ELECTRONIQUE




.... (il y a une suite à la sortie de la requête)

Jointure simple:

Une jointure simple consiste est un produit cartésien avec un INNER JOIN faisant ainsi une restriction sur les lignes. La restriction est faite sur l'égalité de la valeur de deux champs (cas de deux tables) qui sont généralement les clés primaires.

Exemple

```
SELECT NOM, PRENOM, NOMPROG
FROM ETUDIANTS E INNER JOIN PROGRAMME P
ON E.CODEPRG =P.CODEPRG;
```

 NOM	 PRENOM	 NOMPROG
PATOCHE	ALIAN	INFORMATIQUE
FAFAR	CHANTALE	INFORMATIQUE
MARTIN	RACHA	ADMINISTRATION
PLOUFFE	STEVEN	ELECTRONIQUE
ALLARD	MATHIEU	ADMINISTRATION
VIENS	NATHALIE	ADMINISTRATION
DANIS	SAMUEL	ELECTRONIQUE
FAVRE	NICOLAS	INFORMATIQUE
JACOB	YANICK	INFORMATIQUE

Autres exemples :

```
select ename, job, sal, dname
from (syemp inner join sydept on syemp.deptno = sydept.deptno);
```

```
select ename, job, sal, loc
from (syemp inner join sydept on syemp.deptno = sydept.deptno)
where sydept.deptno = 10;
```

```
SELECT nom, prenom, description, note
FROM ((etudiant E INNER JOIN NOTE ON E.numad = R.numad)
INNER JOIN cours C ON C.code_cours = R.code_cours);
```

```
SELECT nom,prenom, description, note
FROM ((etudiant E INNER JOIN NOTE ON E.numad = R.numad)
INNER JOIN cours C ON C.code_cours = R.code_cours);
```

Jointure externe

Jointure externe droite: Dans la jointure externe droite, des enregistrements de table de la première table seront ramenés même si ceux-ci n'ont pas d'occurrences dans la deuxième table.

Jointure externe gauche: Dans la jointure externe gauche des enregistrements de table de deuxième table seront ramenés même si ceux-ci n'ont pas d'occurrences dans la première table.

Dans le cas d'une jointure externe, il faut faire suivre la colonne pour laquelle il n'est pas obligatoire d'avoir des lignes correspondant à l'égalité par l'opérateur LEFT OUTER JOIN ou RIGHT OUTER JOIN

Exemple

Cette requête ramène tous les étudiants y compris ceux qui ne sont pas inscrits dans un programme

```
SELECT NOM, PRENOM, NOMPROG
FROM ETUDIANTS E LEFT OUTER JOIN PROGRAMME P ON
E.CODEPRG=P.CODEPRG;
```

RESPONSABLE	EMPLOYE
alibaba	Martin
alibaba	Patoche
alibaba	Cristophe
Patoche	FAFAR
Cristophe	ROY
Patoche	SIMPSON

IMPORTANT:

Dans le cas d'une jointure externe, il faut faire suivre la colonne pour laquelle il n'est pas obligatoire d'avoir des lignes correspondant à l'égalité par l'opérateur LEFT OUTER JOIN ou RIGHT OUTER JOIN

Lorsqu'un champ sélectionné est présent dans plus d'une table alors il faut le précéder du nom de la table à partir de laquelle on désire l'extraire.

Exemple

Vous pouvez donner un alias aux noms de tables afin de faciliter la référence aux tables. Cependant si un alias est donné alors, il faudra utiliser l'alias à la place du nom de la table.

Quelques fonctions SQL:

Les fonctions agissant sur les groupes

Ces fonctions sont utilisées pour traiter des groupes de rangées et d'afficher un seul résultat. Même si ce sont des fonctions de groupement, elles ne s'utilisent pas tout le temps avec la clause GROUP BY.

Les fonctions MIN et MAX: ce sont des fonctions qui s'utilisent pour afficher la valeur MIN (ou MAX) parmi l'ensemble des valeurs de la colonne indiquée.

Exemple

```
SELECT MAX (NOTE) FROM RESULTATS WHERE CODE_COURS ='KED';
```

Les fonctions AVG et SUM

AVG s'utilise pour obtenir une valeur moyenne des valeurs de la colonne indiquée

SUM s'utilise pour obtenir une valeur totale des valeurs de la colonne indiquée

Exemple

```
SELECT AVG (NOTE) FROM RESULTATS WHERE CODE_COURS ='KED';
```

Les fonctions VARIANCE et STDDEV: Pour calculer la variance et l'écart type sur les valeurs d'une colonne

La fonction COUNT: cette fonction permet de compter le nombre de lignes (rangées) qui répondent à un critère. La clause GROUP BY peut être utilisée ou non

Si une colonne est présente dans la clause SELECT alors elle doit être présente dans la clause GROUP BY

La clause GROUP BY: cette clause permet d'indiquer au système de regrouper des enregistrements selon des valeurs distincts qui existent pour les colonnes spécifiées. La clause HAVING permet de mieux cibler les enregistrements spécifiés.

Exemples

```
SELECT CODEPRG, COUNT(CODEPRG)
FROM ETUDIANTS
GROUP BY CODEPRG;
```

CODEPRG	COUNT(CODEPRG)
430	2
420	4
410	3

```
SELECT CODEPRG, COUNT(CODEPRG)
FROM ETUDIANTS
GROUP BY CODEPRG
HAVING CODEPRG = '420';
```

Cette requête calcule le nombre total d'étudiants.

```
SELECT COUNT(*)
FROM ETUDIANTS;
```