

1] Principe

La notion de jointure consiste à combiner deux ou plusieurs tables afin d'en extraire les données, car il est rare de trouver tous les attributs d'une requête dans une seule table.

L'opération dite de jointure réalise une liaison entre deux tables, en se basant sur l'égalité d'un des attributs commun à chaque table. Ceci implique deux choses :

- le principe premier de la jointure est de réaliser le produit cartésien des tuples de la première table par les tuples de la deuxième table
- le premier principe s'oppose à la notion de requête imbriquée qui effectue de façon séquentielle une requête sur le résultat d'une autre

- *Précisions complémentaires*

D'abord un problème de définition : un attribut qui, dans une table fait référence à une autre table où il est clé primaire, est appelé clé étrangère. Conditions de validités d'une jointure :

1. l'égalité entre l'un des attributs de chaque table afin de réaliser la liaison
2. la condition d'égalité entre les attributs s'écrit en général par une clause WHERE
3. le principe de qualification des noms de colonnes par le préfixe de la table

```
SELECT table1.nom, table2.nom ....
```

La qualification des noms de colonnes est rendu nécessaire par l'emploi de noms identiques dans les tables.

2] Types de jointure

Nous utiliserons pour les exemples les simples tables suivantes :

Table t1

num int4	nom varchar
1	a
2	b
3	c

Table t2

num int4	valeur varchar
1	xxx
3	yyy
5	zzz

- ***La jointure de produit***

La jointure de produit (CROSS JOIN) effectue le produit cartésien de deux tables et n'utilise pas de clause WHERE. Très peu utilisée.

```
SELECT t1.*, t2.* FROM t1 CROSS JOIN t2;
SELECT t1.*, t2.* FROM t1, t2;      # celle qu'on utilise
```

Ligne	num (int4)	nom (varchar)	num (int4)	valeur (varchar)
1	1	a	1	xxx
2	1	a	3	yyy
3	1	a	5	zzz
4	2	b	1	xxx
5	2	b	3	yyy
6	2	b	5	zzz
7	3	c	1	xxx
8	3	c	3	yyy
9	3	c	5	zzz

En fait dans ce cas l'opération de jointure ne réalise... aucune jointure car il n'y a pas d'égalité entre l'un des attributs des deux tables avec la clause WHERE. Ici toutes les combinaisons sont rendus possibles.

• **Les jointures qualifiées**

Elles peuvent être interne (INNER) ou externe (OUTER). Ces deux mots sont optionnels dans la requête par contre en cas d'absence la jointure sera interne par défaut (cas des exemples) :

- La jointure d'égalité (jointure conditionnelle). La plus utilisée des jointures : elle joint deux tables à l'aide d'une colonne commune avec le signe = comme opérateur.

```
SELECT * FROM t1 INNER JOIN t2 ON t1.num = t2.num;
SELECT * FROM t1,t2 WHERE t1.num = t2.num;           # autre écriture
SELECT * FROM t1 INNER JOIN t2 USING (num);         # moins utilisé.
                                                    # L'attribut de liaison
# doit porter le même nom dans les deux tables : num
```

Ligne	num (int4)	nom (varchar)	valeur (varchar)
1	1	a	xxx
2	3	c	yyy

- La jointure naturelle (NATURAL JOIN). Elle choisit automatiquement les attributs possédant le même nom comme attribut de liaison. Suivant le même principe que la jointure d'égalité dans notre cas cela donne le même résultat puisque l'égalité se fait sur la colonne t1.num et t2.num.

```
SELECT * FROM t1 NATURAL INNER JOIN t2;
```

Ligne	num (int4)	nom (varchar)	valeur (varchar)
1	1	a	xxx
2	3	c	yyy

- La jointure de non égalité (NON-EQUIJOIN). Elle suit le principe inverse de la jointure d'égalité et utilise les opérateurs =, <>, <, >, <=, >=, BETWEEN, IN et LIKE.

```
SELECT * FROM t1,t2 WHERE t1.num<>t2.num and t1.num>2;
```

Ligne	num (int4)	nom (varchar)	num (int4)	valeur (varchar)
1	3	c	1	xxx
2	3	c	5	zzz

Note : voyez en plus dans l'exemple la condition multiple.

- L'auto-jointure. Consiste après avoir donné un alias, a effectuer une requête de la table sur elle-même. Le schéma de la relation résultante sera composé par une concaténation de

tous les attributs des relations dont elle est issue. On peut arriver au même résultat par une requête imbriquée.

```
select * from t1 t1bis, t1 t1ter where t1bis.num=t1ter.num;
```

num	nom	num	nom
3	c	3	c
2	b	2	b
1	a	1	a

La requête imbriquée équivalente serait :

• **Jointures externes (OUTER JOIN)**

Les jointures externes sont employées pour retourner toutes les lignes d'une table, même si celles-ci ne possèdent pas de correspondance dans la table jointe. On l'utilise avec les mots clés LEFT (ex : jointure externe gauche), RIGHT et FULL.

```
SELECT * FROM t1 LEFT JOIN t2 USING (num) ;
```

Ligne	num (int4)	nom (varchar)	valeur (varchar)
1	1	a	xxx
2	2	b	
3	3	c	yyy

Ceci s'explique par : en premier une jointure interne est faite, ensuite pour chaque ligne de t1 qui ne satisfait pas la condition de jointure avec aucune ligne de t2, une ligne est ajoutée avec une valeur nulle pour la colonne de t2. Ainsi, la table de jointure a au moins une rangée pour chaque rangée de t1.