

PARTIE 3 : Jointures, unions et sous-requêtes

Principe des jointures

Deux tables peuvent être jointes ligne par ligne, à condition que certaines valeurs spécifiées (colonnes) soient les mêmes dans les deux tables.

Le résultat d'une jointure contient toutes les colonnes des tables jointes.

Clause **ON**

- La clause ON peut contenir autant de conditions que nécessaire.
- Il s'agit généralement d'une égalité entre deux colonnes (pas plus), une de chaque table jointe.

Clause **LEFT JOIN**

- Jonction de la table de droite sur la table de gauche
- Les colonnes de la table jointe sont, dans cet ordre, les colonnes de gauche puis les colonnes de droite.
- Les lignes de la table de gauche (table principale) restent inchangées.
- Exception des doublons : si la clé jointe a des doublons dans la table de droite, la jointure gauche créera plus de lignes.
- Vous pouvez écrire autant de jointures gauches que vous le souhaitez (néanmoins, vérifiez vos résultats après chaque jointure).
- Pour chaque ligne où la clause ON est satisfaite (=vrai), la ligne jointe contient les valeurs de la table de gauche dans les colonnes de la table de gauche, et les valeurs de la table de droite dans les colonnes de la table de droite.
- Exception des doublons : si la clé_gauche a de nombreux équivalents dans la clé_droite, la table jointe contiendra une ligne pour chaque doublon/couple existant (clé_gauche, clé_droite).
- Pour chaque ligne où la clause ON n'est pas satisfaite (=false), la ligne jointe reste inchangée pour les colonnes de la table de gauche, et a des valeurs NULL dans les colonnes de la table de droite.

Clause **RIGHT JOIN**

- Jonction de la gauche vers la droite.
- Jonction de la table de gauche sur la table de droite
- La table de droite (table principale) les rangs restent inchangés.
- La jointure droite suit exactement les mêmes règles que la jointure gauche, elle est purement symétrique.

Clause **INNER JOIN**

- Jointure, quel que soit le côté, de la table de gauche et de la table de droite, uniquement lorsqu'une correspondance bilatérale est trouvée.
- Fonctionne pour des relations 1-1 ou 1-N.
- Pour les relations 1-N : lorsque plusieurs correspondances sont trouvées pour une clé primaire d'une table, les champs associés à cette clé primaire seront dupliqués dans le résultat de la table JOIN.

Union

L'opérateur `UNION` est utilisé pour combiner les résultats de deux ou plusieurs instructions `SELECT`.

- Chaque instruction `SELECT` dans `UNION` doit avoir le même nombre de colonnes.
- Les colonnes doivent également avoir des types de données similaires.
- Les colonnes de chaque instruction `SELECT` doivent également être dans le même ordre.

```
SELECT column_name(s) FROM table1
UNION
SELECT column_name(s) FROM table2
;
```

Union All

L'opérateur UNION ne sélectionne que des valeurs distinctes par défaut. Pour autoriser les valeurs en double, utilisez **UNION ALL** :

```
SELECT column_name(s) FROM table1
UNION ALL
SELECT column_name(s) FROM table2;
```

Remarque : les noms des colonnes dans le jeu de résultats sont généralement égaux aux noms des colonnes dans la première instruction SELECT.

EXCEPT et INTERSECT

Les opérateurs EXCEPT et INTERSECT renvoient des valeurs distinctes en comparant les résultats de deux requêtes.

- EXCEPT renvoie **toutes les valeurs distinctes** de la requête de gauche **qui ne se trouvent pas dans la requête de droite.**
- INTERSECT renvoie toutes les valeurs distinctes qui sont renvoyées à la fois par les requêtes de gauche et de droite de l'opérande INTERSECT.