

Jointures entre tables

Fiche de synthèse

① POURQUOI UNE JOINTURE

Définition 1 (Jointure) Une **jointure** associe les lignes de deux tables qui ont la même valeur pour un attribut commun, le plus souvent une clé étrangère et la clé primaire qu'elle désigne.

Le modèle relationnel répartit les données dans plusieurs tables pour éviter la redondance : pour afficher le nom d'un élève avec le nom de sa classe, il faut relier les tables **Eleves**(id_eleve, nom, prenom, #id_classe) et **Classes**(id_classe, nom_classe, salle).

nom	prenom	id_classe
Dupont	Marie	101
Martin	Lucas	102

id_classe	nom_classe	salle
101	Terminale A	B204
102	Terminale B	C112

Marie Dupont a la valeur 101 : on l'associe à la ligne 101 de Classes, Terminale A.

② SYNTAXE JOIN ... ON

```
SELECT Eleves.nom, Eleves.prenom,
       Classes.nom_classe
FROM Eleves
JOIN Classes
  ON Eleves.id_classe = Classes.id_classe
WHERE Classes.salle = 'B204';
```

- **JOIN** Classes combine la table Eleves avec la table Classes.
- **ON** donne la condition de jointure : on associe les lignes de même id_classe.
- On préfixe un attribut par le nom de sa table, Eleves .nom, pour lever toute ambiguïté.
- **WHERE** filtre ensuite le résultat, comme dans une requête sur une seule table.

Quand l'attribut commun porte le même nom dans les deux tables, **JOIN** Classes **USING** (id_classe) sous-entend l'égalité.

③ SYNTAXE AVEC WHERE

```
SELECT Eleves.nom, Classes.nom_classe
FROM Eleves, Classes
WHERE Eleves.id_classe = Classes.id_classe;
```

Le résultat est le même. Sans la condition, le SGBD calculerait le **produit cartésien** : toutes les combinaisons d'une ligne de chaque table, soit $4 \times 2 = 8$ lignes pour 4 élèves et 2 classes, sans signification. On préfère **JOIN ... ON**, qui sépare la condition de jointure des filtres.

④ TROIS TABLES ET ALIAS

Pour relier trois tables, on enchaîne les **JOIN**. Un **alias**, déclaré par **AS**, donne un nom court à une table dans toute la requête.

```
SELECT E.nom, M.nom_matiere, M.enseignant
FROM Eleves AS E
JOIN Classes AS C
  ON E.id_classe = C.id_classe
JOIN Matieres AS M
  ON C.id_classe = M.id_classe
WHERE M.enseignant = 'M. Dupuis';
```

⑤ AGRÉGATION ET JOINTURE

```
SELECT COUNT(*)
FROM Eleves
JOIN Classes
  ON Eleves.id_classe = Classes.id_classe
WHERE Classes.nom_classe = 'Terminale A';
```

Cette requête compte les élèves de Terminale A.

📖 Écrire une jointure.

1. Repérer les tables qui contiennent les informations demandées.
2. Repérer l'attribut qui les relie : une clé étrangère et la clé primaire qu'elle désigne.
3. Écrire **FROM** T1 **JOIN** T2 **ON** T1.cle = T2.cle, puis les colonnes, les filtres et le tri.

📌 À retenir.

- Une jointure relie les lignes de deux tables qui ont la même valeur d'un attribut commun.
- On écrit **FROM** T1 **JOIN** T2 **ON** T1.cle = T2.cle.
- Sans condition de jointure, on obtient le produit cartésien.