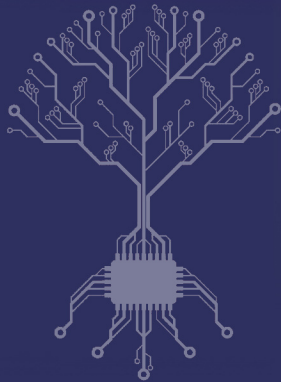




BASE DE DONNÉES (3)

INTERROGATION (SUITE)



JOINTURE

Joindre deux tables, qui sont en relation à l'aide d'une clé primaire et d'une clé étrangère, revient à créer une nouvelle table temporaire qui fusionne les lignes des deux tables.

Il s'agit d'identifier les correspondances entre les deux tables.

Requête SQL

```
SELECT * FROM Table1 JOIN Table2 ON Table1.colonne1=
Table2.colonne2 ;
```

Ou bien

```
SELECT * FROM Table1, Table2 WHERE Table1.colonne1=
Table2.colonne2 ;
```

Exemples

- Soit deux tables :
 - ▶ school (idschool, name, address)
 - ▶ student (idstudent, firstname, lastname, mail, #idschool)
- Afficher les mails des élèves dont l'école est 'Charif Alidrissi' :
 - ▶ `SELECT student.mail FROM student JOIN school ON student.idschool=school.idschool WHERE school.name='Charif Alidrissi' ;`
 - ▶ `SELECT student.mail FROM student, school WHERE student.idschool=school.idschool AND school.name='Charif Alidrissi' ;`

REQUÊTE IMBRIQUÉE

Une requête imbriquée est une sous-requête exécutée à l'intérieur d'une autre requête SQL. Elle remplace souvent une constante au sein d'une clause WHERE ou HAVING.

Les sous-requêtes sont souvent utilisées avec l'instruction SELECT. Toutefois, vous pouvez également les utiliser dans une instruction INSERT, UPDATE ou DELETE ou dans une autre sous-requête.

Requête SQL

```
SELECT ... FROM ... WHERE ... opérateur (SELECT ... FROM ... );
```

Opérateur peut être :

- Si la requête renvoie un seul résultat :

- ▶ Un opérateur simple : =, <, >, !=, <=, >=

- Si elle renvoie plusieurs résultats :

- ▶ IN
NOT IN

- ▶ EXISTS (renvoie vrai si la sous requête retourne au moins une ligne)
NOT EXISTS (renvoie vrai si la sous requête ne retourne aucune ligne)

Exemples

- Soit la table : student (idstudent, firstname, lastname, average, city)
- Afficher les élèves dont la note est supérieure ou égale à la moyenne des notes de tous les élèves :
 - ▶ `SELECT * FROM student WHERE average >= (SELECT AVG(average) FROM student);`
- Afficher l'élève qui a obtenu la meilleure note :
 - ▶ `SELECT * FROM student WHERE average = (SELECT MAX (average) FROM student);`

Exemples

■ Soit les tables :

- ▶ student (idstudent, firstname, lastname, average, city)
- ▶ book (idbook, title, author)
- ▶ borrow (#idstudent, #idbook)

■ Afficher les élèves qui ont emprunté un livre :

- ▶ `SELECT * FROM student WHERE EXISTS (SELECT * FROM borrow WHERE student.idstudent=borrow.idstudent) ;`
- ▶ *ou* `SELECT * FROM student WHERE idstudent IN (SELECT idstudent FROM borrow) ;`