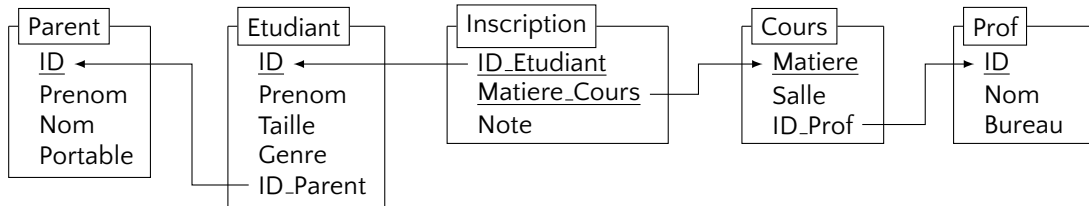


Bases de données et SQL (L1 SoQ et L3 MIASHS)

CM 3 : Requetes de jointure avances

La base de données `fac.db` est disponible sur la page du cours¹. Son schéma est présenté ci-dessous.



Requête 1. Afficher les noms des professeurs qui enseignent un (ou plusieurs) cours (sans répétition).

```
SELECT DISTINCT Prof.Nom FROM Cours JOIN Prof ON (Cours.ID_Prof=Prof.ID);
```

ou

```
SELECT Prof.Nom
FROM Prof
WHERE ID IN (SELECT ID_Prof FROM Cours);
```

ou encore

```
SELECT Prof.Nom
FROM Prof
WHERE EXISTS (SELECT * FROM Cours WHERE Cours.ID_Prof=Prof.ID);
```

Requête 2. Afficher les noms des professeurs qui n'enseignent aucun cours (sans répétition).

```
SELECT DISTINCT Prof.Nom
FROM Prof LEFT OUTER JOIN Cours ON (Prof.ID=Cours.ID_Prof)
WHERE Cours.Matiere IS NULL;
```

ou

```
SELECT Prof.Nom
FROM Prof
WHERE ID NOT IN (SELECT ID_Prof FROM Cours);
```

ou encore

```
SELECT Prof.Nom
FROM Prof
WHERE NOT EXISTS (SELECT * FROM Cours WHERE Cours.ID_Prof=Prof.ID);
```

1. <http://researchers.lille.inria.fr/~staworko/sql19.html>

Requête 3. Afficher les étudiants dont le parent n'est pas renseigné dans la base de données.

```
SELECT Etudiant.*
FROM Etudiant LEFT OUTER JOIN Parent ON (Etudiant.ID_Parent=Parent.ID)
WHERE Parent.ID IS NULL;
```

ou

```
SELECT *
FROM Etudiant
WHERE ID_Parent IS NULL OR ID_Parent NOT IN (SELECT ID FROM Parent);
```

ou encore

```
SELECT *
FROM Etudiant
WHERE NOT EXISTS (
    SELECT *
    FROM Parent
    WHERE Parent.ID = Etudiant.ID_Parent
);
```

Requête 4. Pour tout cours afficher sa matière avec le nom de prof qui l'assure.

```
SELECT Cours.Matiere, Prof.Nom
FROM Cours JOIN Prof ON (Cours.ID_Prof = Prof.ID);
```

ou

```
SELECT C.Matiere, P.Nom
FROM Cours AS C JOIN Prof AS P ON (C.ID_Prof = P.ID);
```

ou encore plus simplement

```
SELECT C.Matiere, P.Nom
FROM Cours C JOIN Prof P ON (C.ID_Prof = P.ID);
```

Requête 5. Afficher le nom complet de tout étudiant qui suit un cours assuré par un professeur dont le bureau est *D.42*

```
SELECT DISTINCT E.Prenom, P.Nom
FROM Etudiant E
LEFT OUTER JOIN Parent P ON (E.ID_Parent = P.ID)
JOIN Inscription I ON (E.ID = I.ID_Etudiant)
JOIN Cours C ON (I.Matiere_Cours = C.Matiere)
JOIN Prof Pr ON (C.ID_Prof = Pr.ID)
WHERE Pr.Bureau = 'D.42';
```

Requête 6. Pour tout étudiant afficher son prénom et le numéro de portable de son parent. La colonne avec le numéro de portable doit s'appeler *Portable de Parent*.

```
SELECT Etudiant.Prenom, Parent.Portable AS "Portable de Parent"
FROM Etudiant LEFT OUTER JOIN Parent ON (Etudiant.ID_Parent=Parent.ID);
```

Requête 7. Afficher les paires de nom de professeurs qui partagent le bureau.

```
SELECT P1.Nom, P2.Nom
FROM Prof AS P1 JOIN Prof AS P2
WHERE P1.Bureau = P2.Bureau
      AND P1.ID <> P2.ID;
```

Requête 8. Idem mais sans répéter les même deux professeurs

```
SELECT P1.Nom, P2.Nom
FROM Prof AS P1 JOIN Prof AS P2
WHERE P1.Bureau = P2.Bureau
      AND P1.ID < P2.ID;
```

Requête 9. Afficher les nom complètes des étudiants qui suivent au moins deux cours différents assurés par le même professeur (sans répétition).

```
SELECT DISTINCT E.Prenom, P.Nom
FROM Etudiant AS E
LEFT OUTER JOIN Parent AS P ON (E.ID_Parent = P.ID)
JOIN Inscription AS I1 ON (E.ID = I1.ID_Etudiant)
JOIN Cours AS C1 ON (I1.Matiere_Cours = C1.Matiere)
JOIN Inscription AS I2 ON (E.ID = I2.ID_Etudiant)
JOIN Cours AS C2 ON (I2.Matiere_Cours = C2.Matiere)
WHERE I1.Matiere_Cours <> I2.Matiere_Cours
      AND C1.ID_Prof = C2.ID_Prof;
```