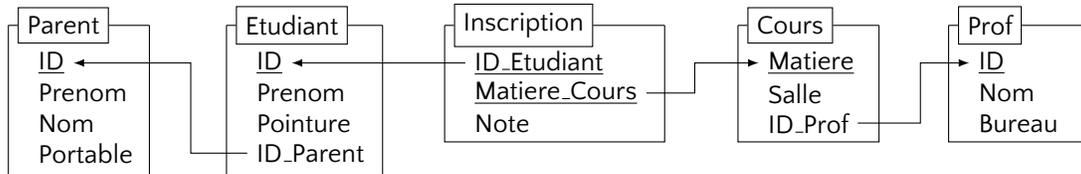


SQL 2018 (L1 SoQ)

CM 2 : Requêtes de jointure

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



Requête 1. Afficher l'information complète sur les cours avec l'information complète sur les professeurs qui les assurent

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

Requête 2. Idem mais en utilisant la jointure croisée.

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

Requête 3. Afficher l'information complète sur les cours avec les noms des professeurs qui les assurent.

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

Requête 4. 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);
```

Requête 5. 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;
```

Requête 6. Afficher les nom des professeurs qui enseigne des cours dans la salle 'B2.461'.

```
SELECT Prof.Nom FROM Cours JOIN Prof ON (Cours.ID_Prof=Prof.ID) WHERE Cours.Salle='B2.461';
```

Requête 7. Idem mais cette fois dans la requête utilise des aliases pour la rendre plus courte

```
SELECT P.Nom FROM Cours AS C JOIN Prof AS P ON (C.ID_Prof=P.ID) WHERE C.Salle='B2.461';
```

Requête 8. Afficher le prénom et le nom des étudiants

```
SELECT Etudiant.Prenom, Parent.Nom FROM Etudiant JOIN Parent ON (Etudiant.ID_Parent=Parent.ID);
```

Requête 9. Afficher le prénom et si renseigné le nom de famille de tout étudiant

```
SELECT Etudiant.Prenom, Parent.Nom
FROM Etudiant LEFT OUTER JOIN Parent ON (Etudiant.ID_Parent=Parent.ID);
```

1. <http://researchers.lille.inria.fr/~staworko/sql18.html>

Requête 10. Afficher la liste des notes de Marie

```
SELECT Inscription.Matiere_Cours, Inscription.Note
FROM Etudiant
JOIN Inscription ON (Etudiant.ID=Inscription.ID_Etudiant)
WHERE Etudiant.Prenom='Marie';
```

Requête 11. Calculer la moyenne des notes de Marie

```
SELECT AVG(Inscription.Note)
FROM Etudiant
JOIN Inscription ON (Etudiant.ID=Inscription.ID_Etudiant)
WHERE Etudiant.Prenom='Marie';
```

Requête 12. Afficher la liste des nom complets des étudiants qui suivent le cours de 'HTML' dans l'ordre décroissant de leur notes.

```
SELECT Etudiant.Prenom, Parent.Nom, Inscription.Note
FROM Etudiant
LEFT OUTER JOIN Parent ON (Etudiant.ID_Parent=Parent.ID)
JOIN Inscription ON (Etudiant.ID=Inscription.ID_Etudiant)
WHERE Matiere_Cours='HTML'
ORDER BY Inscription.Note DESC;
```

Requête 13. Afficher les cours et les nom les professeurs qui n'ont pas renseigne tous les notes

```
SELECT DISTINCT Cours.Matiere, Prof.Nom
FROM Inscription
JOIN Cours ON (Inscription.Matiere_Cours=Cours.Matiere)
JOIN Prof ON (Cours.ID_Prof=Prof.ID)
WHERE Inscription.Note IS NULL;
```