

Exercice

ETUDIANT (CodeEt, NomEt, DatnEt)

MATIERE (CodeMat, NomMat, CoefMat)

ENSEIGNANT (CodeEns, NomEns, GradeEns, #CodeMat)

NOTE (#CodeEt, #CodeMat, Note)

1. Afficher le nom étudiant, le nom matière et la note obtenue dans cette matière

```
SELECT NomEt, NomMat, Note
FROM Etudiant
NATURAL JOIN Note NATURAL JOIN Matiere
```

2. Afficher le nom enseignant, et les noms des matières qu'il enseigne

```
SELECT E.NomEns, M.NomMat,
FROM Enseignant E, Matiere M
WHERE M.CodeMat = E.CodeMat
```

3. Afficher la moyenne des notes par matière

```
SELECT AVG(Note) as 'Moyenne', CodeMat
FROM Note
GROUP BY CodeMat
```

4. Afficher le nom et la moyenne de chaque étudiant

```
SELECT NomEt, SUM(Note*CoefMat) / SUM(CoefMat)
FROM Etudiant NATURAL JOIN Note NATURAL JOIN Matiere
GROUP BY NomEt
```

5. Quel est l'étudiant qui a eu la plus grande moyenne ?

```
SELECT NomEt, SUM(Note*CoefMat) / SUM(CoefMat)
FROM Etudiant NATURAL JOIN Note NATURAL JOIN Matiere
GROUP BY NomEt
HAVING
    SUM(Note*CoefMat) / SUM(CoefMat) =
```

```
SELECT MAX(SUM(Note*CoefMat) / SUM(CoefMat))
FROM Note NATURAL JOIN Matiere NATURAL JOIN Etudiant
GROUP BY CodeEt)
```

6. Quels sont les étudiants qui ont obtenu une moyenne supérieure à la moyenne de la classe ?

```
SELECT NomEt, SUM (Note*CoefMat) / SUM(CoefMat)
FROM Etudiant NATURAL JOIN Note NATURAL JOIN Matiere
GROUP BY NomEt
HAVING SUM (Note*CoefMat) / SUM(CoefMat) >=
    SELECT AVG(MoyenEtud)
FROM (
    SELECT SUM (Note*CoefMat) / SUM(CoefMat) as
'MoyenEtud'
    FROM Etudiant NATURAL JOIN Note NATURAL JOIN
Matiere
    GROUP BY CodeEt)
...
SELECT AVG(Moyen_Etud)
FROM (
    SELECT SUM(Note*CoefMat) / SUM(CoefMat) as 'Moyen_Etud'
    FROM Etudiant NATURAL JOIN Note NATURAL JOIN Matiere
    GROUP BY CodeEt)
```

7. C'est dans quelle matière les étudiants ont plus travaillé ?

8. Faites un bonus de 2 points sur la matière UML aux étudiants ayant la note supérieure à 15 dans cette matière

9. Supprimer les matières dans lesquelles les étudiants n'ont pas composé (Pas de note)