ETUDIANT (Numetu, Nometu, Dtnaiss, Cdsexe)

SEXE (Cdsexe, Lbsexe)

ENSEIGNANT (Numens, Nomens, Grade, Ancien)

MATIERE (Numat, Nomat, Coeff, Numens)

NOTES (Numetu, Numat, Note)

À partir de la Base de données ETUDIANTS ci dessus, écrire les requêtes SQL permettant de répondre aux questions suivantes.

Exercice récapitulatif en Langage SQL

10. Afficher le nom, le numéro et le sexe des étudiants qui ont une note en informatique supérieure à leur note de Mathématiques.

11. Afficher le nom des couples, nom du garçon nom de la fille, dont la différence d’âge ne dépasse pas 6 mois (180 jours).

12. Pour les étudiants n’ayant pas de note dans une matière, afficher le numéro le nom de l’étudiant et le nom de la matière concernée.

13. Afficher, les matières pour lesquelles la moyenne des notes est inférieure à 10. Afficher le nom de l’enseignant correspondant.

14. Afficher, pour chaque matière, qu’elle est la meilleure note et quel est l’étudiant qui l’a obtenue.

15. Afficher, pour chaque sexe (Homme, Femme) la moyenne des notes dans la matière enseignée par Julie Boyer.

Exercice récapitulatif en Langage SQL

10. Afficher le nom, le numéro et le sexe des étudiants qui ont une note en informatique supérieure à leur note de Mathématiques.

SELECT Nometu, E.Numetu, Cdsexe

FROM ETUDIANT E, NOTES N, MATIERE M

WHERE E.Numetu=N.Numetu

AND N.Numat=M.Numat

AND Nomat='Informatique'

AND Note > ALL

(SELECT Note FROM NOTES N, MATIERE M

WHERE N.Numat=M.Numat

AND Nomat='Mathématiques'

AND N.Numetu=E.Numetu);

11. Afficher le nom des couples, nom du garçon nom de la fille, dont la différence d’âge ne dépasse pas 6 mois (180 jours).

SELECT E1.Nometu,E2.Nometu

FROM ETUDIANT E1, ETUDIANT E2

WHERE E1.CdSexe='H'

AND E2.CdSexe='F'

AND DATEDIFF(E1.Dtnaiss,E2.Dtnaiss) BETWEEN -180 AND 180;

12. Pour les étudiants n’ayant pas de note dans une matière, afficher le numéro le nom de l’étudiant et le nom de la matière concernée.

SELECT E.Numetu, Nometu, Nomat

FROM ETUDIANT E

LEFT OUTER JOIN NOTES N

ON E.Numetu=N.Numetu

LEFT OUTER JOIN Matiere M

ON N.Numat=M.Numat OR N.Numat IS NULL

WHERE Note IS NULL;

13. Afficher, les matières pour lesquelles la moyenne des notes est inférieure à 10. Afficher le nom de l’enseignant correspondant.

SELECT Nomens, Nomat, Avg(note)

FROM ENSEIGNANT E, MATIERE M, NOTES N

WHERE E.Numens=M.Numens

AND M.Numat=N.Numat

GROUP BY Nomens, Nomat

HAVING Avg(note)< 10;

14. Afficher, pour chaque matière, qu’elle est la meilleure note et quel est l’étudiant qui l’a obtenue.

SELECT Nomat, Nometu, Note

FROM MATIERE M, NOTES N, ETUDIANT E

WHERE N.Numat=M.Numat

AND E.Numetu=N.Numetu

AND Note=(SELECT Max(Note) FROM NOTES E1

WHERE E1.Numat=M.Numat);

15. Afficher, pour chaque sexe (Homme, Femme) la moyenne des notes dans la matière enseignée par Julie Boyer.

SELECT Nomat, Nometu, Note

FROM MATIERE M, NOTES N, ETUDIANT E

WHERE N.Numat=M.Numat

AND E.Numetu=N.Numetu

AND Note=(SELECT Max(Note) FROM NOTES E1

WHERE E1.Numat=M.Numat);

Requête 1 : Afficher les notes de l’étudiant numéro 1 qui sont égales aux notes de l’étudiant numéro 2.

SELECT Note

FROM NOTES

WHERE Numetu=1 AND Note IN

(SELECT Note FROM NOTES WHERE Numetu=2);

Requête 2 : Afficher les notes de l’étudiant numéro 1 qui sont supérieures aux notes de l’étudiant numéro 2.

SELECT Note

FROM NOTES

WHERE Numetu=1 AND Note > ANY

(SELECT Note FROM NOTES WHERE Numetu=2);

Requête 3 : Afficher les notes de l’étudiant numéro 1 qui sont inférieures à toutes les notes de l’étudiant numéro 9.

SELECT Note

FROM NOTES

WHERE Numetu=1 AND Note < ALL

(SELECT Note FROM NOTES WHERE Numetu=9);

Requête 4 : Afficher toutes les informations sur les étudiants qui n’ont aucune note.

SELECT \*

FROM ETUDIANT E

WHERE NOT EXISTS

(SELECT Numetu FROM NOTES WHERE Numetu=E.Numetu);