

TP N°2 : SQL

Dans ce TP, nous disposons d'une base de données sur les élèves d'une CPGE fictive, elle possède 5 tables :

- **ELEVE** : qui contient des informations sur les élèves de la CPGE
 - NOM
 - PRENOM
 - LYCEE_ORIGINE donnant le lycée d'origine de l'élève, le numéro correspond au numéro de la ligne de la table **LYCEE**
 - RANG_ADMISSION donnant le rang dans le classement parcoursup de l'élève
 - NOTE_MATH_BAC donnant la note de l'élève en mathématiques au bac
 - NOTE_PHY_BAC donnant la note de l'élève en physique au bac
 - MENTION donnant sa mention au bac
 - ANNEE donnant l'année à laquelle l'élève est rentré dans la CPGE
- **LYCEE** : qui contient des informations sur les lycées d'origine
 - NOM
 - VILLE donnant la ville du lycée
 - DISTANCE donnant la distance à vol d'oiseau entre Chartres et la ville.
- **MPSI** : qui contient les moyennes des élèves durant la MPSI
 - STATUT qui fait référence à la table **STATUT** qui permet de savoir si l'élève est passé en MP ou non
- La table **COEFF_MPSI** donne es coefficient de chaque matière en MPSI

Questions :

Pour chaque question, donner une requête SQL qui renvoie la demande :

1 Quels sont les élèves venant du lycée Branly ?

Correction : on peut s'aider du numéro de Branly (plutôt à éviter) :

```
SELECT NOM, PRENOM
FROM ELEVE
WHERE LYCEE_ORIGINE=6

sinon

SELECT NOM, PRENOM
FROM ELEVE
WHERE LYCEE_ORIGINE=(SELECT ROWID FROM LYCEE WHERE NOM="BRANLY")
```

2 Combien y en-a-t-il ? (on veut une requête)

Correction :

```
SELECT COUNT(NOM)
FROM ELEVE
WHERE LYCEE_ORIGINE=(SELECT ROWID FROM LYCEE WHERE NOM="BRANLY")
```

- 3 Quel est le numéro de l'élève ayant eu la meilleure note en mathématiques au semestre 1 ?

Correction : Si on est sûr qu'il y en a qu'un ou qu'on en veut qu'un :

```
SELECT ROWID, MAX(MATHS_S1)
FROM MPSI
WHERE 0<=MATHS_S1 AND MATHS_S1<=20
```

sinon :

```
SELECT ROWID
FROM MPSI
WHERE MATHS_S1=(SELECT MAX(MATHS_S1)
FROM MPSI
WHERE 0<=MATHS_S1 AND MATHS_S1<=20)
```

- 4 Faire une requête renvoyant les élèves (nom et prénom) ayant un rang d'admission inférieur à 30.

Correction :

```
SELECT NOM, PRENOM
FROM ELEVE
WHERE RANG_ADMISSION <=30
```

- 5 Donner une requête SQL qui renvoie le nombre d'élèves du lycée Branly par année.

Correction :

```
SELECT ANNEE, COUNT(NOM) AS NOMBRE_ELEVES
FROM ELEVE
WHERE LYCEE_ORIGINE=(SELECT ROWID FROM LYCEE WHERE NOM="BRANLY")
GROUP BY ANNEE
```

- 6 Donner une requête renvoyant le nom, le prénom et le nom (faire une jointure).

Correction :

```
SELECT ELEVE.NOM, PRENOM, LYCEE.NOM
FROM ELEVE
JOIN LYCEE ON LYCEE_ORIGINE=LYCEE.ROWID
```

- 7 Faire une requête SQL qui renvoie le nom des lycées et le nombre d'élèves de chaque lycée.

Correction : Petite remarque : il y a des lycées homonymes, il faudrait rajouter la ville.

```
SELECT COUNT(ELEVE.NOM), LYCEE.NOM
FROM ELEVE
JOIN LYCEE ON LYCEE_ORIGINE=LYCEE.ROWID
```

```
GROUP BY LYCEE_ORIGINE
```

8 Même question en rajoutant le taux de passage en MP.

Correction : Le plus "simple" est de créer une première requête donnant le nombre d'admis en MP et une autre donnant le nombre d'élèves par lycée :

```
SELECT COUNT(NOM) AS NB_ELV, LYCEE_ORIGINE
FROM ELEVE
GROUP BY LYCEE_ORIGINE
```

```
SELECT COUNT(STATUT) as NB_MP, LYCEE_ORIGINE
FROM(
SELECT *
FROM MPSI
JOIN ELEVE ON ELEVE.rowid=MPSI.rowid
WHERE STATUT="OK" or STATUT="R" or STATUT="OK2" or STATUT="R2")
GROUP BY LYCEE_ORIGINE
```

on combine les deux :

```
SELECT 100.0*COUNT(STATUT)/NB_ELV as TAUX_PASSAGE , LYCEE_ORIGINE
FROM(
SELECT *
FROM MPSI
JOIN ELEVE ON ELEVE.rowid=MPSI.rowid
WHERE STATUT="OK" or STATUT="R" or STATUT="OK2" or STATUT="R2")
JOIN (SELECT COUNT(NOM) AS NB_ELV, LYCEE_ORIGINE as LO
FROM ELEVE
GROUP BY LYCEE_ORIGINE) ON LO=LYCEE_ORIGINE
GROUP BY LYCEE_ORIGINE
```

9 Donner une requête qui renvoie le nombre d'élèves venant d'un lycée chartrain (ville de Chartres ou Luisant), Rambolitain (ville de Rambouillet) et les autres.

Correction :

```
SELECT CHARTRAINS, RAMBOLITAINS, AUTRES
FROM (
SELECT COUNT(ELEVE.NOM) as CHARTRAINS
FROM ELEVE
JOIN LYCEE ON ELEVE.LYCEE_ORIGINE=LYCEE.rowid
WHERE VILLE="CHARTRES" or VILLE="LUISANT"
)
JOIN (
SELECT COUNT(ELEVE.NOM) as RAMBOLITAINS
FROM ELEVE
JOIN LYCEE ON ELEVE.LYCEE_ORIGINE=LYCEE.rowid
WHERE VILLE="RAMBOUILLET"
)
JOIN (
SELECT COUNT(ELEVE.NOM) as AUTRES
FROM ELEVE
JOIN LYCEE ON ELEVE.LYCEE_ORIGINE=LYCEE.rowid
WHERE not( VILLE="RAMBOUILLET" or VILLE="CHARTRES" or VILLE="LUISANT")
)
```

10 Même question que la précédente, mais par année.

Correction :

```
SELECT ANNEE, CHARTRAINS, RAMBOLITAINS, AUTRES
FROM (
SELECT ANNEE, COUNT(ELEVE.NOM) as CHARTRAINS
FROM ELEVE
JOIN LYCEE ON ELEVE.LYCEE_ORIGINE=LYCEE.rowid
WHERE VILLE="CHARTRES" or VILLE="LUISANT"
GROUP BY ANNEE
)
JOIN (
SELECT ANNEE as a1, COUNT(ELEVE.NOM) as RAMBOLITAINS
FROM ELEVE
JOIN LYCEE ON ELEVE.LYCEE_ORIGINE=LYCEE.rowid
WHERE VILLE="RAMBOUILLET"
GROUP BY ANNEE
)
ON a1=ANNEE
JOIN (
SELECT ANNEE as a2, COUNT(ELEVE.NOM) as AUTRES
FROM ELEVE
JOIN LYCEE ON ELEVE.LYCEE_ORIGINE=LYCEE.rowid
WHERE not( VILLE="RAMBOUILLET" or VILLE="CHARTRES" or VILLE="LUISANT")
GROUP BY ANNEE
)
ON a2=ANNEE
```

- 11 Donner une requête renvoyant la distance moyenne des lycées d'origine par année.

Correction :

```
SELECT ANNEE, AVG(DISTANCE)
FROM(
SELECT ELEVE.NOM, ANNEE, DISTANCE
FROM ELEVE
JOIN LYCEE ON ELEVE.LYCEE_ORIGINE=LYCEE.rowid
WHERE not DISTANCE='X')
GROUP BY ANNEE
```

- 12 Donner une requête renvoyant le nom et prénom des élèves et leur moyenne générale.

Correction :

```
SELECT NOM, PRENOM,LYCEE_ORIGINE,ANNEE,STATUT, ROUND(MOYENNE,2) AS MOYENNE
FROM ELEVE
JOIN( SELECT ROWID AS R2,STATUT, (MPSI.MATHS_S1*(SELECT MATHS_S1 FROM COEFF_MPSI)
+ MPSI.MATHS_S2*(SELECT MATHS_S2 FROM COEFF_MPSI)
+ MPSI.PC_S1*(SELECT PC_S1 FROM COEFF_MPSI) + MPSI.PC_S2*(SELECT PC_S2 FROM COEFF_MPSI)
+ MPSI.SII_S1*(SELECT SII_S1 FROM COEFF_MPSI) + MPSI.SII_S2*(SELECT SII_S2 FROM COEFF_MPSI)
+ MPSI.FP_S1*(SELECT FP_S1 FROM COEFF_MPSI) + MPSI.FP_S2*(SELECT FP_S2 FROM COEFF_MPSI)
+ MPSI.LV1_S1*(SELECT LV1_S1 FROM COEFF_MPSI) + MPSI.LV1_S2*(SELECT LV1_S2 FROM COEFF_MPSI)
+ MPSI.TIPE*(SELECT TIPE FROM COEFF_MPSI))/(SELECT MATHS_S1+MATHS_S2+PC_S1+PC_S2+SII_S1
+SII_S2+INFO_S1+INFO_S2+FP_S1+FP_S2+LV1_S1+LV1_S2+TIPE FROM COEFF_MPSI) AS MOYENNE
FROM MPSI
)
ON ELEVE.rowid=R2
ORDER BY MOYENNE DESC
```

ou

```

SELECT NOM, PRENOM,LYCEE_ORIGINE,ANNEE,STATUT, ROUND(MOYENNE,2) AS MOYENNE
FROM ELEVE
JOIN( SELECT MPSI.ROWID AS R2,STATUT, (MPSI.MATHS_S1*COEFF_MPSI.MATH_S1
+ MPSI.MATHS_S2*COEFF_MPSI.MATH_S2
+ MPSI.PC_S1*COEFF_MPSI.PC_S1 + MPSI.PC_S2*COEFF_MPSI.PC_S2
+ MPSI.SII_S1*COEFF_MPSI.SII_S1 + MPSI.SII_S2*COEFF_MPSI.SII_S2
+ MPSI.FP_S1*COEFF_MPSI.FP_S1 + MPSI.FP_S2*COEFF_MPSI.FP_S2
+ MPSI.LV1_S1*COEFF_MPSI.LV1_S1 + MPSI.LV1_S2*COEFF_MPSI.LV1_S2
+ MPSI.TIPE*COEFF_MPSI.TIPE)/(SELECT MATHS_S1+MATHS_S2+PC_S1+PC_S2+SII_S1
+SII_S2+INFO_S1+INFO_S2+FP_S1+FP_S2+LV1_S1+LV1_S2+TIPE FROM COEFF_MPSI) AS MOYENNE
FROM MPSI
JOIN COEFF_MPSI
)
ON ELEVE.rowid=R2
ORDER BY MOYENNE DESC

```

- 13 Donner la moyenne des moyennes obtenus par les élèves de chaque lycée, on utilisera que les élèves qui sont passés en MP.

Correction :

```

SELECT LYCEE.NOM, ROUND(AVG(MOYENNE),2) AS MOYENNE, COUNT(MOYENNE) AS NOMBRE_ETUDIANTS
FROM ELEVE
JOIN( SELECT STATUT, ROWID AS R2, (MPSI.MATHS_S1*(SELECT MATHS_S1 FROM COEFF_MPSI)
+ MPSI.MATHS_S2*(SELECT MATHS_S2 FROM COEFF_MPSI)
+ MPSI.PC_S1*(SELECT PC_S1 FROM COEFF_MPSI) + MPSI.PC_S2*(SELECT PC_S2 FROM COEFF_MPSI)
+ MPSI.SII_S1*(SELECT SII_S1 FROM COEFF_MPSI) + MPSI.SII_S2*(SELECT SII_S2 FROM COEFF_MPSI)
+ MPSI.FP_S1*(SELECT FP_S1 FROM COEFF_MPSI) + MPSI.FP_S2*(SELECT FP_S2 FROM COEFF_MPSI)
+ MPSI.LV1_S1*(SELECT LV1_S1 FROM COEFF_MPSI) + MPSI.LV1_S2*(SELECT LV1_S2 FROM COEFF_MPSI)
+ MPSI.TIPE*(SELECT TIPE FROM COEFF_MPSI))/(SELECT MATHS_S1+MATHS_S2+PC_S1+PC_S2+SII_S1
+SII_S2+INFO_S1+INFO_S2+FP_S1+FP_S2+LV1_S1+LV1_S2+TIPE FROM COEFF_MPSI) AS MOYENNE
FROM MPSI
) ON ELEVE.rowid=R2
JOIN LYCEE ON ELEVE.LYCEE_ORIGINE=LYCEE.rowid
WHERE STATUT="OK"
GROUP BY LYCEE_ORIGINE
ORDER BY MOYENNE DESC

```

ou en effectuant le même principe que dans la question précédente.