

Corrigé Série TD N°02

Exercice 1

Requêtes SQL

1

```
SELECT Count( SELECT Num_C
              FROM Animal A
              GROUP BY Num_C
              HAVING Count(Num_C) = SELECT Capacite
                                   FROM Cage C
                                   WHERE A.Num_C = C.Num_C )
```

2

```
SELECT Espece
FROM Animal
GROUP BY Espece
HAVING Count(Espece)>=3
ORDER BY Espece DESC
```

3

```
SELECT Nom_A
FROM Entretien
GROUP BY Nom_A
HAVING Count(Code)>=1
```

4

```
SELECT Nom_A, Count(Nom_A) as Nbr
FROM Attraper
GROUP BY Nom_A
ORDER BY Nbr
LIMIT 5
```

5

```
SELECT Nom, Prenom
FROM Entretien E, Gardien G
WHERE (E.Code = G.code) and Nom_A IN SELECT Nom_A
                                       FROM Entretien E1, Gardien G1
                                       WHERE (E1.Code = G1.code) and (G1.prenom = « Mohamed »)
```

6

```
SELECT Sum(Nbr) /count(Num_C)
FROM SELECT Num_C, Count(Code) AS Nbr
      FROM Entretien
      GROUP BY Num_C
```

7

```
SELECT Espese, Count(Nom_A)
FROM Animal
GROUP BY Espese
```

Exercice 2

ETUDIANT (Num_E, Nom_E, Prenom_E, wilaya_E, Date_Naiss, Niveau, spécialité)

MODULE (Code, libellé, crédit, Coefficient)

ENSEIGNANT (Num_En, Nom_En, Prenom_En, wilaya_En, Salle, Grade)

ETUDIER (Num_E, Num_En, Code, Note_TD, Note_TP, Note_Examen)

A) Créer en SQL (LDD) les tables ETUDIANT, MODULE et ETUDIER

Create table Etudiant

```
(
  Num_E integer PRIMARY KEY,
  Nom_E chaine[30] NOT NULL,
  Prenom_E chaine[30] NOT NULL,
  wilaya_E chaine[30] NOT NULL,
  Date_Naiss date,
  Niveau integer NOT NULL CHECK IN [L1,L2,L3,M1,M2],
  Spécialité chaine[30] NOT NULL
)
```

Create table Module

```
(
  Code integer PRIMARY KEY,
  Libellé chaine[50] NOT NULL,
  Crédit integer NOT NULL,
  Coefficient integer NOT NULL
)
```

Create table ETUDIER

```
(
  Num_E integer REFERENCES Etudiant,
  Num_En integer REFERENCES Enseignant,
  Code integer REFERENCES module,
  Note_TD integer CHECK IN [0..20] DEFAULT 0,
  Note_TP integer CHECK IN [0..20],
  Note_Examen integer CHECK IN [0..20]
  PRIMARY KEY (Num_E, Num_En, Code)
)
```

B) Exprimer les requêtes suivantes en SQL

- Afficher la liste des Étudiants en deuxième année licence informatique par ordre alphabétique

```
Select *
From Etudiant E
Where (niveau =L2) and (spécialité= informatique)
Order by Nom_E, Prenom_E
```
- Quels sont les étudiants qui ont eu une note d'examen supérieure à 13 dans le module BDD

```
Select Nom_E, Prenom_E
From Etudiant E, Etudier ET
Where (E.Num_E=ET.Num_E) and (Note_Examen >= 13) and (Code=BDD)
```
- Afficher la liste des étudiants (nom, prénom) par niveau et spécialité

```
Select Nom_E, Prenom_E
From Etudiant
```

