

# BD : Gestion d'un zoo, TP1

Frances Tom, Gourdon Jérémie

January 25, 2022

# 1 Complétion du schéma

Le code de création des tables complété est donné en annexe.

Nous avons de plus rajouté 2 tuples dans les tables LesCages et LesAnimaux pour avoir 2 cages contenant des léopards :

```
01 | insert into LesCages values (13 , 'fauve' , 2);
02 | insert into LesAnimaux values ('Magalie', 'femelle' , 'leopard', 'fauve', 'Italie',
    2014, 13);
```

## 2 Requêtes SQL

### 2.1 Dans quelles cages se trouvent les lions et les léopards ?

Algèbre :  $(\text{LesAnimaux} : \text{type\_an} = \text{'lion'} \text{ OR } \text{type\_an} = \text{'leopard'}) [\text{noCage}]$

Code :

```
01 | SELECT DISTINCT noCage AS numeroCage_Q1
02 | FROM LesAnimaux
03 | WHERE type_an='lion' OR type_an='leopard';
```

Résultat :

	NUMEROCAGE_Q1
1	12
2	13
3	11

### 2.2 Quels sont les gardiens qui gardent des cages de léopards ?

Algèbre :  $((\text{LesGardiens} * \text{LesAnimaux}) : \text{type\_an} = \text{'leopard'}) [\text{nomE}]$

Code :

```
01 | SELECT DISTINCT nomE
02 | FROM LesGardiens NATURAL JOIN LesAnimaux
03 | WHERE type_an='leopard';
```

Résultat :

	NOME
1	Labbe
2	Dupont

### 2.3 Quels sont les gardiens qui gardent à la fois des cages de léopards et de lions ?

Algèbre :  $((\text{LesGardiens} * \text{LesAnimaux}) : \text{type\_an} = \text{'leopard'}) [\text{nomE}] \text{ INTERSECT } ((\text{LesGardiens} * \text{LesAnimaux}) : \text{type\_san} = \text{'lion'}) [\text{nomE}]$

Code :

```
01 | -- V1 --
02 | SELECT DISTINCT nomE
03 | FROM LesGardiens G JOIN LesAnimaux A1 USING (noCage)
```

```

04 | WHERE type_an='leopard'
05 | AND nomE IN ( SELECT nomE
06 |               FROM LesGardiens G JOIN LesAnimaux A1 USING (noCage)
07 |               WHERE type_an='lion' );
08 |
09 | -- V2 --
10 | SELECT nomE
11 | FROM LesGardiens JOIN LesAnimaux USING (noCage)
12 | WHERE type_an='leopard'
13 | INTERSECT
14 | SELECT nomE
15 | FROM LesGardiens JOIN LesAnimaux USING (noCage)
16 | WHERE type_an='lion';

```

Résultat :

	NOME
1	Labbe
2	Dupont

## 2.4 Quels sont les gardiens qui gardent des animaux qui sont ou ont été malades ?

Algèbre :  $((\text{LesMaladies} * \text{LesAnimaux}) * \text{LesGardiens}) [\text{nomE}]$

Code :

```

01 | SELECT DISTINCT nomE
02 | FROM LesMaladies JOIN LesAnimaux USING (nomA)
03 |                JOIN LesGardiens USING (noCage);

```

Résultat :

	NOME
1	Lachaize
2	Labbe
3	Dupont

## 2.5 Quels sont les responsables qui ont ou ont eu des animaux malades dans leur allée ?

Algèbre :  $((((\text{LesResponsables} * \text{LesCages}) * \text{LesAnimaux}) * \text{LesMaladies}) [\text{nomE}]$

Code :

```

01 | SELECT DISTINCT nomE
02 | FROM LesResponsables JOIN LesCages USING (noAllee)
03 |                JOIN LesAnimaux USING (noCage)
04 |                JOIN LesMaladies USING (nomA);

```

Résultat :

	NOME
1	Desmoulin
2	Verdier

## 2.6 Quels sont les gardiens qui ne gardent pas de cage de léopards ?

Algèbre :  $(\text{LesGardiens})[\text{nomE}] - ((\text{LesGardiens} * \text{LesAnimaux}) : \text{type\_an} = \text{'leopard'})[\text{nomE}]$

Code :

```
01 | SELECT DISTINCT nomE
02 | FROM LesGardiens
03 | WHERE nomE NOT IN ( SELECT nomE
04 |                     FROM LesGardiens NATURAL JOIN LesAnimaux
05 |                     WHERE type_an='leopard' );
```

Résultat :

	NOME
1	Lachaize

## 2.7 Combien d'animaux sont malades ?

Code :

```
01 | SELECT COUNT(DISTINCT nomA) AS NbAnimauxMalade
02 | FROM LesMaladies;
```

Résultat :

	NBANIMAUXMALADE
1	3

## 2.8 Combien y a-t-il d'animaux malades par type de maladie ?

Code :

```
01 | SELECT nomM, COUNT(nomA) AS NbAnimauxMaladeParMaladie
02 | FROM LesMaladies
03 | GROUP BY nomM;
```

Résultat :

	NOMM	NBANIMAUXMALADEPARMALADIE
1	grippe	2
2	angine	1
3	rage de dents	1

## 2.9 Quels sont les animaux avec au moins deux maladies ?

Code :

```
01 | SELECT nomA
02 | FROM LesMaladies
03 | GROUP BY nomA
04 | HAVING COUNT(nomM) >= 2;
```

Résultat :

	NOMA
1	CharLy

## 2.10 Quels gardiens gardent plus de deux cages avec la fonction "fauve" ?

Code :

```
01 | SELECT nomE
02 | FROM LesGardiens NATURAL JOIN LesCages
03 | WHERE fonction='fauve'
04 | GROUP BY nomE
05 | HAVING COUNT(noCage) >= 2;
```

Résultat :

	NOME
1	Labbe
2	Dupont

## 2.11 Quels responsables possèdent le plus de cages dans une de ses allées ?

Pour cette requête, nous avons supposé que toutes les allées devaient avoir un responsable. Code :

```
01 | SELECT nomE
02 | FROM LesResponsables NATURAL JOIN LesCages
03 | GROUP BY nomE, noAllée
04 | HAVING COUNT(noCage) = (SELECT MAX(COUNT(noCage))
05 |                        FROM LesCages
06 |                        GROUP BY noAllée);
```

Résultat :

	NOME
1	Jouanot

## 2.12 Quels sont les gardiens qui gardent toutes les cages de léopards ?

Pour cette requête, nous avons supposé que toutes les allées devaient avoir un responsable. Code :

```
01 | SELECT nomE
02 | FROM LesGardiens NATURAL JOIN LesAnimaux
03 | WHERE type_an='leopard'
04 | GROUP BY nomE
05 | HAVING COUNT(DISTINCT noCage) = ( SELECT COUNT(DISTINCT noCage)
06 |                                FROM LesAnimaux
07 |                                WHERE type_an='leopard');
```

Résultat :

	NOME
1	Dupont

## 2.13 Pour chaque gardien, donner le nombre de léopards gardés ? (on veut tous les gardiens)

Code :

```
01 | SELECT nomE, NVL(COUNT(nomA), 0) AS NbLeopardsGardés
02 | FROM LesGardiens G LEFT JOIN LESANIMAUX A ON (A.noCage=G.noCage AND type_an='leopard')
03 | GROUP BY nomE;
```

Résultat :

	NOME	NBLEOPARDSGARDES
1	Labbe	2
2	Lachaize	0
3	Dupont	3

2.14 Pour chaque gardien, donner le nombre de cage de léopards gardées ? (on veut tous les gardiens)

Code :

```
01 | SELECT nomE, nvl(COUNT(DISTINCT A.noCage), 0) AS NbCagesLeopardsGardees
02 | FROM LesGardiens G LEFT JOIN LesAnimaux A ON (A.noCage=G.noCage AND A.type_an='leopard')
03 | GROUP BY nomE;
```

Résultat :

	NOME	NBCAGESLEOPARDSGARDEES
1	Lachaize	0
2	Labbe	1
3	Dupont	2

### 3 Annexe

```
01 | drop table LesMaladies ;
02 | drop table LesAnimaux ;
03 | drop table LesResponsables ;
04 | drop table LesGardiens ;
05 | drop table LesEMployes ;
06 | drop table LesCages ;
07 |
08 | create table LesCages (
09 |     noCage number(3),
10 |     fonction varchar2(20),
11 |     noAllee number(3),
12 |     constraint pk_LesCages_noCage PRIMARY KEY (noCage),
13 |     constraint ck_LesCages_noCage CHECK(noCage > 0 AND noCage < 1000),
14 |     constraint ck_LesCages_noAllee CHECK(noAllee > 0 AND noAllee < 1000)
15 | );
16 |
17 | create table LesEmployes (
18 |     nomE varchar2(20),
19 |     adresse varchar2(20),
20 |     constraint pk_LesEmployes_nomE PRIMARY KEY (nomE)
21 | );
22 |
23 | create table LesGardiens (
24 |     noCage number(3),
25 |     nomE varchar2(20),
26 |     constraint pk_LesGardiens_noCage_nomE PRIMARY KEY (noCage, nomE),
27 |     constraint fk_LesGardiens_nomE FOREIGN KEY (nomE) REFERENCES LesEmployes (nomE)
28 | ),
29 |     constraint fk_LesGardiens_noCage FOREIGN KEY (noCage) REFERENCES LesCages (
30 |     noCage),
31 |     constraint ck_LesGardiens_noCage CHECK(noCage > 0 AND noCage < 1000)
32 | );
33 |
34 | create table LesResponsables (
35 |     noAllee number(3),
36 |     nomE varchar2(20),
37 |     constraint pk_LesResp_noAllee_nomE PRIMARY KEY (noAllee, nomE),
38 |     constraint fk_LesResp_nomE FOREIGN KEY (nomE) REFERENCES LesEmployes (nomE),
39 |     constraint ck_LesResponsables_noAllee CHECK(noAllee > 0 AND noAllee < 1000)
40 | );
41 |
42 | create table LesAnimaux (
43 |     nomA varchar2(20),
44 |     sexe varchar2(13),
45 |     type_an varchar2(15),
46 |     fonction_cage varchar2(20),
47 |     pays varchar2(20),
48 |     anNais number(4),
49 |     noCage number(3),
50 |     constraint pk_LesAnimaux_nomA PRIMARY KEY (nomA),
51 |     constraint fk_LesAnimaux_noCage FOREIGN KEY (noCage) REFERENCES LesCages (
52 |     noCage),
53 |     constraint ck_LesAnimaux_sexe CHECK (sexe IN ('femelle', 'male', '
54 |     hermaphrodite')),
55 |     constraint ck_LesAnimaux_noCage CHECK (noCage > 0 AND noCage < 1000),
56 |     constraint ck_LesAnimaux_annais CHECK (annais >= 2000)
57 | );
58 |
59 | create table LesMaladies (
60 |     nomA varchar2(20),
61 |     nomM varchar2(20),
62 |     constraint pk_LesMaladies_nomA_nomM PRIMARY KEY (nomA, nomM),
63 |     constraint fk_LesMaladies_nomA FOREIGN KEY (nomA) REFERENCES LesAnimaux (nomA)
64 | );
```