

## Contrôle Final

**EX1 (12 pts) :** Une agence immobilière se propose de gérer, par une base de données, la location des locaux qui peuvent être de différents types : maison, studio, appartement, magasin, garage, ...

Parmi les informations qui caractérisent un local, on peut en citer : son identifiant, sa superficie, ...

Pour chaque location, la société établit un contrat. Les informations qui caractérisent un contrat de location sont : le numéro du contrat, le montant de la location, la durée de la location et toutes les informations nécessaires à l'identification du locataire et du local loué.

Chaque locataire qui signe un contrat, est identifié essentiellement par le numéro de sa carte d'identité nationale (CIN), son nom et son prénom.

1. Identifier les ensembles d'entités avec leurs attributs.
2. Donner les règles de gestion montrant les associations qu'on peut établir entre les ensembles d'entités et précisant les cardinalités.
3. Dessiner le schéma E-A modélisant cette réalité.
4. Faire le passage du schéma E-A au schéma de la BD relationnelle.
5. Exprimer les requêtes **a)** et **b)** en langage algébrique et les requêtes **c)** et **d)** en SQL :
  - a) Donner le type et la superficie des locaux non loués.
  - b) Donner la liste des locataires ayant loué des studios à partir de janvier 2015.
  - c) Pour chaque type de local, donner la liste (nom et prénom) des locataires ainsi que la date, la durée et le montant du contrat de location.
  - d) Pour chaque type de local, calculer le nombre de contrats.

**EX1 (8 pts) :** On considère la gestion d'un parc d'animaux à travers la base de données suivante :

**ANIMAL** ( CodeA, GenreA, AnNaissA, EtatA, CodeF#, CodePays# )

**FAMILLE** ( CodeF, DesignatF )

**NOURRITURE** ( CodeN, DesignatN )

**ORIGINE** ( CodePays, LibellePays )

**REGIME** ( CodeA#, DateCh, CodeN#, QteN )

- La table **REGIME** suppose qu'un animal change de régime alimentaire au cours de son existence. Exemple : à la naissance, un lion se nourrit d'une certaine quantité **QteN** de lait, puis à l'âge adulte il se nourrit essentiellement de viande en quantité différente selon son âge. Donc à une date donnée **DateCh** une seule nourriture **CodeN** est affectée à un animal donné et en quantité **QteN** appropriée.
- La table **ANIMAL** ne contient que les informations concernant les animaux qui vivent dans le parc.
- La table **FAMILLE** contient toutes les espèces animales répertoriées sur terre.

1. Donner le schéma E-A correspondant en précisant les cardinalités des associations.
2. Exprimer les requêtes a), b), c) et d) en langage algébrique et les requêtes c), d), e) et f) en SQL.
  - a) La liste de toutes les familles animales (Code et Désignation)
  - b) L'historique du régime alimentaire (date, désignation de la nourriture et la quantité) de l'animal de code 'Lion32'
  - c) Les codes des animaux qui vivent dans le parc et qui sont, soit nés en 2006 soit originaires de 'Kenya'
  - d) Les codes et les dates de naissances de tous les animaux qui vivent dans le parc cet appartement à la famille 'BOVIN' (c-à-d les bœufs, les vaches et les taureaux)
  - e) Le nombre total d'animaux qui vivent dans le parc.
  - f) Les origines distinctes (code du pays et libellés du pays) de tous les animaux qui vivent dans le parc.

### EX1 : Conception d'une BD pour gérer une agence immobilière

Ensemble d'entités	Attributs
Local	IdLcl ; TypeLcl ; SupLcl
Locataire	CinLct ; NomLct ; PreLct ; DatNaiLct ; TelLct
Contrat	IdCtr ; DateCtr ; MontCtr

**RG2** : Un locataire peut louer un ou plusieurs locaux à des dates différentes (et durées séparées) des contrats

```

    erDiagram
        Local ||--o{ Contrat : "Signe"
        Local ||--o{ Locataire : "Signe"
        Contrat ||--o{ Locataire : "Signe"

```

The diagram illustrates the following entities and their attributes:

- Local** (Entity): Attributes include IdLcl (Primary Key), TypeLcl, and SupLcl.
- Contrat** (Entity): Attributes include IdCtr (Primary Key), DateCtr, and MontCtr.
- Locataire** (Entity): Attributes include CinLct (Primary Key), NomLct, PreLct, DateNaiLct, and TelLct.

The relationships are defined as follows:

- Local** (0.N) is connected to **Contrat** (1.1) via the relationship **Signe**.
- Local** (1.N) is connected to **Locataire** (1.N) via the relationship **Signe**.
- Contrat** (1.N) is connected to **Locataire** (1.N) via the relationship **Signe**.

4. Schéma relationnel (3 pts)

Local (IdLcl, TypeLcl, SupLcl)

Locataire (CinLct, NomLct, PreLct, DateNaiLct, TelLct)

Contrat (IdCtr, IdLct#, CinLct#, DateCtr, MontCtr, DurCtr)

- CI {
- Pour chaque contrat, la durée de contrat (DurCtr) commence à partir de la date du contrat (DatCtr).
  - La date de naissance du locataire (DateNaiLct) doit être < à la date du contrat de location (DatCtr).
  - Un local ne peut pas faire l'objet de 2 locations durant la même période.

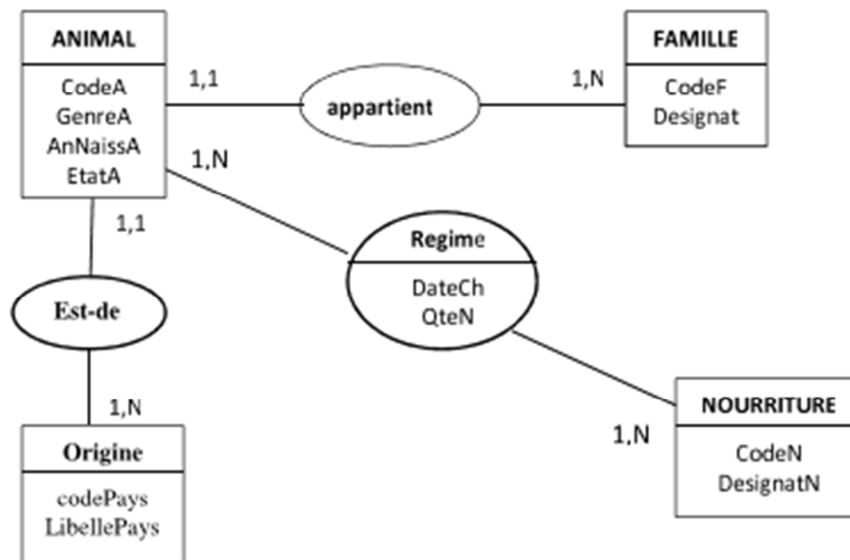
5. Expression des requêtes

Langage algébrique	<p>a) type et superficie des locaux non loués. (1 pt)</p> $Res = \Pi_{TypeLcl, SupLcl} (Local - (\Pi_{IdLcl, TypeLcl, SupLcl} (Local \mid > \mid Contrat)))$
Langage algébrique	<p>b) la liste des locataires ayant loué des studios à partir de janvier 2015. (1 pt)</p> $Res = \Pi_{NomLct, PreLct} (\sigma_{TypeLcl = 'Studio'}(Local) \mid > \mid \sigma_{DateCtr \geq 1 \text{ janvier } 2015}(Contrat))$
Langage SQL	<p>c) Pour chaque type de local, donner la liste des locataires ainsi que la date, la durée et le montant du contrat de location. (1 pt)</p> <pre> SELECT NomLct, PreLct, TypeLcl, DateCtr, DurCtr, MontCtr FROM Local, Contrat, Locataire WHERE Local.IdLcl = Contrat.IdLcl AND Contrat.CinLct = Locataire.IdLct GROUP BY TypeLcl </pre>

<b>Langage SQL</b>	<p><b>d)</b> nombre de contrats pour chaque type de local <b>(1 pt)</b></p> <pre> SELECT COUNT(IdCtr), TypeCtr FROM Local, Contrat WHERE Local.IdLcl =Contrat.IdLcl GROUP BY TypeLcl </pre>
--------------------	---

**EX2 : Une BD pour la gestion d'un parc d'animaux**

**1. Schéma Entité-Association (2 pts)**



**2. Expression des requêtes**

<b>Langage algébrique</b>	<p><b>g)</b> liste de toutes les familles animales (Code et Désignation) <b>(0,75 pt)</b></p> $Res = \Pi_{CodeF, DesignatF}(Famille)$
<b>Langage algébrique</b>	<p><b>h)</b> historique du régime alimentaire (date, désignation de la nourriture et la quantité) de l'animal de code 'Lion32'. <b>(0,75 pt)</b></p> $Res = \Pi_{DateCh, DesignatN, QteN}(Nourriture \mid \sigma_{CodeA = 'Lion32'}(Regime))$

<b>Langage algébrique</b>	<p>c) Codes des animaux qui vivent dans le parc et qui sont, soit nés en 2006 soit originaires de 'Kenya'. <b>(0,75 pt)</b></p> <p><math display="block">\text{Res} = \Pi_{\text{CodeA}} (\sigma_{\text{AnNaissA} = 2006} (\text{Animal}) \mid \vee \mid \sigma_{\text{LibellePays} = \text{'Kenya'}} (\text{Origine}))</math></p>
<b>Langage algébrique</b>	<p>d) Codes et dates de naissances de tous les animaux qui vivent dans le parc et appartement à la famille 'BOVIN'. <b>(0,75 pt)</b></p> <p><math display="block">\text{Res} = \Pi_{\text{CodeA}, \text{DateNaissA}} (\sigma_{\text{DesignatF} = \text{'BOVIN'}} (\text{Famille}) \mid \vee \mid \sigma_{\text{EtatA} = \text{'V'}} (\text{Animal}))</math></p>

<b>Langage SQL</b>	<p>c) Codes des animaux qui vivent dans le parc et qui sont, soit nés en 2006 soit originaires de 'Kenya'. <b>(0,75 pt)</b></p> <pre>SELECT CodeA FROM Animal, Origine WHERE Animal.CodePays = Origine.CodePays AND (Animal.AnNaissA = 2006) OR (Origine.LibellePays = 'Kenya')</pre>
<b>Langage algébrique</b>	<p>d) Codes et dates de naissances de tous les animaux qui vivent dans le parc et appartement à la famille 'BOVIN'. <b>(0,75 pt)</b></p> <pre>SELECT CodeA , DateNaissA FROM Animal, Famille WHERE Animal.CodeF = Famille.CodeF AND Animal.Etat = 'Vivant' AND DesignatF = 'BOVIN'</pre>

<b>Langage SQL</b>	<p>e) nombre total d'animaux qui vivent dans le parc. <b>(0,75 pt)</b></p> <pre>SELECT COUNT (*) FROM Animal WHERE EtatA = 'Vivant'</pre>
<b>Langage SQL</b>	<p>f) Les origines distinctes (code du pays et libellés du pays) de tous les animaux qui vivent dans le parc. <b>(0,75 pt)</b></p> <pre>SELECT DISTINCT CodePays, LibellePays FROM Origine, Animal WHERE Origine.CodePays = Animal.CodePays AND EtatA = 'Vivant'.</pre>