

Corrigé_Type du TD 1 : Introduction aux bases de données & Modèle EA

Partie I : Questions de cours

1. Expliquer en quoi l'approche par base de données diffère essentiellement de l'approche par traitement de fichiers classique.

| Dans l'approche « Traitement de Fichiers » | Dans l'approche « Bases de Données » |
|---|---|
| Absence de standardisation des méthodes de conception. Ex: l'utilisateur doit connaître l'organisation (séquentielle, indexée, ...) des fichiers qu'il utilise afin de pouvoir accéder aux informations dont il a besoin | Méthodes de conception standard avec plusieurs niveaux de modélisation. Ainsi, les mêmes méthodes de conception seront utilisées pour tout type d'application. Une personne pourra être chargée de la conception, et une autre pour la programmation. |
| Plusieurs formats de stockage et plusieurs langages de manipulation de ces fichiers : C++, Java, Python... | SQL : langage <u>universel</u> « de création, de manipulation et d'interrogation, de contrôle » des bases de données |
| Redondance des données (plusieurs copies de la même donnée) qui peut causer des problèmes lors de leur mise à jour (incohérence des données résultantes) | Éliminer la redondance des données. Ainsi, la Mise A Jour (MAJ) de la BD sera avec une cohérence globale assurée (les opérations de MAJ sont = Ajout, Suppression, Modification) |
| Difficulté de partage et de sécurisation des données stockées. Aussi, problèmes de maintenance des données, avec un coût élevé des pannes (arrêt brutal, panne de disque...) | Les SGBD assurent des mécanismes fiables : de centralisation et de partage des données, de maintenance et de sécurité des BD |
| L'utilisateur doit à chaque fois modifier ses codes sources pour pouvoir manipuler ses fichiers si ces derniers ont changé de format (pas d'isolation entre les données et les programmes). Il sera même obligé d'écrire de nouveaux programmes et créer peu être de nouveaux fichiers qui contiennent des données déjà présentes dans d'autres fichiers. | Les SGBD offrent les services nécessaires à la manipulation des BD tout en assurant : - l'indépendance logique (entre le schéma conceptuel et les vues (besoins) multiples de chaque utilisateur) - l'indépendance physique (entre le schéma conceptuel et les contraintes matériels et celles d'implémentation (de programmation)) |
| Mais, - on est libre de tout effort de conception - on n'a pas besoin d'un système (logiciel) de gestion - on n'aura pas besoin d'un administrateur de ces fichiers | Mais, pour bénéficier de ces perfections, on doit : - fournir des efforts de conception standardisée - installer des Systèmes de Gestion des BD (SGBD) - Avoir une personne chargée de l'administration des BD (un Administrateur de BD (DBA)) |

2. Quels sont les principaux objectifs d'un SGBD ?

Des objectifs ont été fixés aux **SGBD** pour résoudre les problèmes causés par **la démarche classique basée sur les fichiers de données**. Ces objectifs sont :

- **Indépendance physique** : La façon dont les données sont conçues doit être indépendante des structures de stockage utilisées (*donc, indépendance entre le niveau conceptuel (logique) et le niveau physique (interne)*).
- **Indépendance logique** : Un même ensemble de données peut être vu différemment par des utilisateurs différents. Toutes ces visions personnelles (appelées vues externes) des données doivent être intégrées dans une vision (schéma) globale (indépendance entre le niveau conceptuel (logique) et le niveau utilisateur (externe))
- **Accès aux données** : L'accès aux données se fait par l'intermédiaire d'un **Langage de Manipulation de Données (LMD)**. C'est un langage permette d'obtenir des réponses justes à des requêtes complexes en un temps raisonnable, cela via des opérations basées sur l'algèbre relationnel de façon totalement transparente pour l'utilisateur.
- **Cohérence des données** : Les données sont soumises à un certain nombre de **contraintes d'intégrité** qui définissent **un état cohérent de la base**. Elles doivent être exprimées initialement (lors de la création de la BD) et vérifiées automatiquement à chaque insertion, modification ou suppression des données. Les contraintes d'intégrité sont ainsi décrites par le **Langage de Description de Données (LDD)**. Exemple de cohérence : le domaine des valeurs présent par le champ num_groupe dans la table étudiant (clé étrangère) doit être cohérent avec (inclus dans) le domaine des valeurs du champ clé num_groupe dans la table groupe.

- **Partage des données** : Il s'agit de permettre à plusieurs utilisateurs d'accéder aux mêmes données au même moment de manière transparente. Si ce problème est simple à résoudre quand il s'agit uniquement de consultation, cela ne l'est plus quand il s'agit de modifications dans un contexte multi-utilisateurs, car il ne faut pas permettre à deux (ou plus) utilisateurs de modifier la même donnée « en même temps », cela pour assurer (via le mécanisme des transactions) un résultat d'interrogation cohérent pour un utilisateur consultant une table pendant qu'un autre la modifie. Le **Langage de Contrôle de Données (LCD)** assure cette tâche.
- **Administration centralisée des données (intégration)** : Toutes les données doivent être centralisées dans un réservoir unique commun à toutes les applications (c'est le serveur de BD). En effet, des visions différentes des données (entre autres) se résolvent plus facilement si les données sont administrées de façon centralisée (via un Administrateur de BD, appelé DBA).
- **Non redondance des données** : Afin d'éviter les *problèmes* lors *des mises à jour* (le risque d'incohérence, le travail répété, ...), chaque donnée ne doit être présente qu'une seule fois dans la base.
- **Sécurité des données** : Les données doivent pouvoir être protégées contre les accès non autorisés. Donc, on doit associer à chaque utilisateur des droits d'accès aux données (via des mots de passes à chaque utilisateur autorisé) ainsi que des droits de manipulations spécifiques (via la définition des *pouvoirs* de chaque utilisateur).
- **Résistance aux pannes** : Que se passe-t-il si une panne survient au milieu d'une modification de données, si certains fichiers contenant les données deviennent illisibles ... ? Il faut pouvoir récupérer une **base** dans un **état « sain »**. Ainsi, en cas d'une panne au milieu d'une modification deux solutions sont possibles : soit récupérer les données dans leur état initial avant la modification, soit terminer complètement l'opération interrompue.

3. Dans une BD, l'incohérence doit être évitée et la redondance minimisée. Expliquez cette phrase.

Exemple d'incohérence à éviter : donner à un étudiant un numéro de groupe inexistant dans la table groupe.

Exemple de redondance à minimiser : le numéro du groupe existe dans la table groupe (*clé primaire*) mais doit aussi exister (**se répéter**) dans la table étudiant (comme *clé étrangère* via les règles du passage relationnel). Cette redondance assure ainsi la liaison (un pont) entre ces deux tables, ce qui sert à naviguer entre les tables d'une BD.

4. Pourquoi l'abstraction de données ?

L'abstraction de données est une caractéristique qui permet l'indépendance entre les programmes et les données. Grâce à cette caractéristique les SGBD offrent aux utilisateurs une représentation conceptuelle des données sans les détails sur les modes de stockage de données, ni sur l'implémentation des opérations sur ces données.

Donc, la représentation conceptuelle des données n'est pas liée à la manière dont elles sont stockées ou manipulées.

5. Définir les termes suivants :

- **Donnée** : fait enregistrable (mesurable) ayant une signification commune. ex : nom, âge, poids, longueur ...
- **Information** : c'est une donnée mise dans un contexte clair (ex : l'âge de Mohamed est 23 ans) (le poids de Ridha est 75 kg). Donc, l'information est un élément de connaissance susceptible d'être conservé, traité ou transmis à l'aide d'un support et d'un mode de codification normalisé.
- **Base de données** : une BD est un ensemble de tables (données structurées) liées (via le passage des clés) et gérées par un Système de Gestion de Base de Données (SGBD).
- **SGBD** : c'est un pack de programmes qui permet à l'utilisateur la création, la gestion et le contrôle des BD.
- **Schéma de données** : appelé aussi « intention » d'une BD, c'est la description conceptuelle de cette BD
- **Etat de BD** : appelé aussi « extension » à l'instant t, c'est le contenu actuel (les enregistrements) de la BD.
- **Modèle de données** : c'est un ensemble de concepts décrivant la structure d'une BD (les types de données, les relations et les contraintes applicables sur la BD).
- **Schéma interne** (physique) : décrit la structure de stockage interne (physique) et les chemins d'accès à la BD.
- **Schéma conceptuel** (logique) : c'est une description de haut niveau (abstraite) de la BD (entités, relations, types de données et contraintes). Aussi, c'est une description générale de la BD pour la communauté des utilisateurs.
- **Schéma externe** (utilisateur) : décrit les vues des différents groupes d'utilisateurs. Chaque schéma externe (appelé aussi vue externe) décrit uniquement la partie de la BD qui concerne un groupe d'utilisateurs spécifique.
- **Indépendance des données** : permet de modifier un schéma quel que soit son niveau (interne, conceptuel, externe) par rapport à la BD sans influence sur les autres schémas.
- **LDD** : Langage de Définition des Données, utilisé pour définir le schéma conceptuel (structure et contraintes) de la BD.
- **LMD** : Langage de Manipulation des Données, utilisé pour extraire (sélectionner) ou actualiser (mettre à jour) les BD.
- **Langage de requêtes** : est un pseudo-langage informatique (de type requête (interrogation)) standardisé, destiné à interroger ou manipuler une BD, pour l'extraction ou d'actualisation (mise à jour) fiable des données.

- **Transaction** : c'est un programme (ou processus) qui implique un ou plusieurs accès à une BD (lecture ou écriture) et ayant la spécificité d'être *isolé* et *atomique*. Le SGBD doit assurer : l'**isolation** des transactions lorsque plusieurs transactions sont exécutées en même temps sur la BD, et l'**atomicité** qui veut dire que toute transaction (opération) sur la BD soit exécutée complètement ou annulée complètement (donc, soit « finir toute l'opération (*Commit*) » ou « annuler toute l'opération (*RollBack*) »)
- **Fiabilité** : c'est la « robustesse » et la « tolérance » aux problèmes matériels, logiciels ou humains.
- **Persistance de données** : les données persistantes sont les données enregistrées par le SGBD qui ne peuvent être enlevées que par une requête explicite envoyée au SGBD.
- **Modèle conceptuel** : décrit les entités du monde réel, en termes d'objets, de propriétés et de relations, indépendamment de toute technique d'implémentation des données. Ce modèle se concrétise par un *schéma entités-associations* représentant la structure du système d'information, du point de vue des données.
- **Modèle logique** : précise le modèle conceptuel par des choix organisationnels. Il s'agit d'une transcription (également appelée **dérivation**) du *schéma entités-associations* dans un formalisme sous forme de base de données relationnelle adaptée à une implémentation ultérieure. Les choix techniques d'implémentation (choix d'un SGBD) ne seront effectués qu'au niveau suivant (niveau physique).
- **Modèle physique** : permet d'établir la manière concrète dont la BD sera mise en place, ainsi que le choix du SGBD.
- **SQL** : (*Structured Query Language*, Langage de Requête (d'interrogation) Structuré) : c'est un langage normalisé (supporté par la majorité des SGBD) permettant la manipulation des données (le contenu d'une base de données) et des schémas relationnels (la structure d'une base de données).
- **MySQL** : c'est un SGBDR (SGBD Relationnel) très utilisé (surtout en programmation web à cause de sa légèreté), il est Open Source (ce qui signifie que son code source est librement disponible et que quiconque peut modifier MySQL pour l'améliorer ou l'adapter à ses besoins). Il existe une version gratuite ainsi qu'une version commerciale (payante) du SGBD MySQL.

6. Citez quelques exemples de SGBD commercialisés :

Par exemple : ORACLE, Microsoft SQL Server, Microsoft Access, MY SQL, INGRES, IBM DB2 ...

7. Quelle est la différence entre un état de BD et son schéma ?

Etat d'une BD : c'est le contenu de la BD, **se change avec le temps** (via les opérations de mise à jour des données)

Schéma d'une BD : la description (ex: la structure) de la BD qui doit être **stable** au max (ne change pas fréquemment)

8. Quelle est la relation entre un schéma de base de données et un modèle de base de données ?

Modèle (cadre de définition) : concepts utilisés pour structurer les données et définir le schéma de la BD

Schéma (plan) : c'est le schéma d'organisation de la BD, description de l'organisation des données et de leur type. Si on a fait une bonne modélisation des données, leur schéma ne varie pas (sera stable) au cours de l'utilisation de la BD

9. Expliquer la différence entre l'indépendance logique et l'indépendance physique des données

L'indépendance logique est la capacité de modifier le schéma conceptuel sans que cela affecte les schémas externes (les vues utilisateur (les programmes applicatifs)). Tandis que l'indépendance physique est la capacité de modifier le schéma interne (structures de stockage (fichiers, index, chemin d'accès, ...)) sans modifier le schéma conceptuel.

10. Expliquer la différence entre LDD et LMD ?

- **LDD** : Langage de définition de données (Data Definition Language DDL) permet **la création, la modification, et la suppression d'objets** (tables, vues, index ...).

C'est un langage de manipulation de la structure (colonnes (champs)) de la base de données. Il permet également de définir le domaine des valeurs des données (nombres, chaîne de caractères, dates, . . .) et d'ajouter des contraintes de valeur sur les données (ex : l'âge d'un employé doit être entre 18 et 60 ans). Les instructions de base du **LDD** sont : **CREATE, ALTER, DROP, AUDIT, NOAUDIT, ANALYZE, RENAME ...**

- **LMD** : Langage de manipulation de données (Data Manipulation Language DML) permet **l'interrogation, l'insertion, la modification et la suppression de tuples** (enregistrements, lignes).

Le LMD permet la Mise A Jour (MAJ) de la BD (**l'ajout**, la **suppression** et la **modification** de **lignes** (il ne peut pas modifier des colonnes qui sont les champs (structure) des tables)). Il permet aussi la visualisation du contenu des tables et leur verrouillage. Les instructions du **LMD** sont : **INSERT, UPDATE, DELETE, SELECT, EXPLAIN, PLAN ...**

11. L'utilisateur doit-il faire appel au LDD ou au LMD pour accomplir chacune des tâches suivantes ?

- Changer l'adresse d'un étudiant : **LMD**
- Définir une nouvelle **table** des cours : **LDD**
- Mettre à jour la moyenne d'un étudiant : **LMD**
- Ajouter une **colonne** à une table des étudiants : **LDD**
- Entrer les renseignements sur un nouvel client : **LMD**

12. Quelle est la différence entre les LMD procéduraux et non procéduraux ? Citer quelques exemples de ces langages.

SQL et PL/SQL (PL : Procedural Language) sont des langages de bases de données relationnelles. SQL est un langage de requête structuré qui ajoute, supprime, modifie ou consulte les données d'une BD. PL/SQL est un langage procédural qui est une extension de SQL et qui contient des instructions SQL dans sa syntaxe. La différence entre SQL et PL/SQL réside dans le fait qu'une seule requête est exécutée à la fois en SQL, alors qu'en PL/SQL, un bloc entier de code est exécuté à la fois, qui peut contenir (à la différence de SQL) des variables et des instructions de contrôle (boucles, tests).

La majorité des SGBD professionnels supportent les LMD non procéduraux (LMD classiques) ainsi que les LMD procéduraux. L'exemple de LMD procédural le plus utilisé est le PL/SQL qui est un langage propre au SGBD Oracle.

13. Que veut dire le terme modèle de données ? Expliquez la différence entre un modèle de données et son implémentation.

Modèle de données : un ensemble de concepts décrivant la structure (type de données, relations et contraintes) de la BD. Donc, c'est une définition logique abstraite (indépendante des contraintes physiques) des objets (données) et des opérations (traitements sur ces données).

L'implémentation consiste à réaliser physiquement le modèle conçu sur un système informatique (création et organisation physique des données sur un ordinateur).

14. Quels inconvénients résultent de l'utilisation d'un SGBD ?

En plus des efforts de modélisation à fournir pour concevoir une base de données, les inconvénients d'utilisation des SGBD pour gérer une base de données sont : coûts supplémentaires d'acquisition d'un SGBD en plus des coûts d'administration d'une BD (les administrateurs des BD (les DBA) sont des spécialistes généralement très bien payés !)

15. Citer trois mécanismes utilisés par les SGBD pour assurer la sécurité de donnée.

Voici quelques mécanismes de sécurité :

- **Authentification** : Contrôle d'accès (par mots de passe) et la gestion des autorisations (affecter à chaque utilisateur des droits spécifiques de manipulation des données).
- **Réplication** : Sauvegarde (faire des copies) et restauration automatique du contenu d'une BD.
- **Mécanismes transactionnels** et gestion des accès concurrentiels (accès simultané à une même donnée par plusieurs utilisateurs) cela via les mécanismes de « Verrouillage et Sérialisation ».

16. Citer trois problèmes liés à la présence de la redondance de données dans une BD ? Dans quels cas cette redondance est nécessaire ?

Les problèmes liés à la redondance dans une BD :

- Gaspillage de l'espace de stockage en enregistrant la même donnée plusieurs fois.
- Incohérence du contenu de la base de données. Par exemple, problème de la propagation des mises à jour des clés passées des tables pères aux tables fils lors du passage relationnel.
- Gaspillage de ressources humaines. Par exemple, une même donnée peut être saisie à plusieurs reprises par des employés différents, avec un risque toujours présent de commettre des erreurs lors de l'introduction de ces données au sein du système.

Mais, cette **redondance est nécessaire** dans le cas des **clés étrangères** pour assurer les **liens** entre les différentes tables de la base.

17. Qu'est ce qu'un Type d'Entité (TE) ? Qu'est ce qu'un Ensemble d'Entités ? Expliquer la différence entre Entité, un TE et Ensemble d'Entités.

Entité : une entité est un objet, une chose concrète ou abstraite qui peut être reconnue distinctement et qui est caractérisée par son unicité.

Type-Entité : Un type-entité désigne un **Ensemble d'Entités** qui possèdent une sémantique et des propriétés communes.

Exemple : Les personnes (entités « étudiant » et « enseignant »), les documents (entités « livre » et « support de cours »)

En effet, dans le cas d'une personne par exemple, des propriétés comme : le nom et le prénom, ... existent toujours.

- Une **Entité** est souvent appelée « occurrence » ou « instance » de son **Type-Entité**.

18. Qu'est qu'un type d'association récursive ? Donnez des exemples.

Une association récursive (appelée aussi association cyclique ou réflexive) décrit un lien entre les occurrences (les enregistrements) d'une même entité. Voici deux exemples :

L'association : « est fils_de » entre une personne est une autre personne (récursivité sur la même entité « personne »)

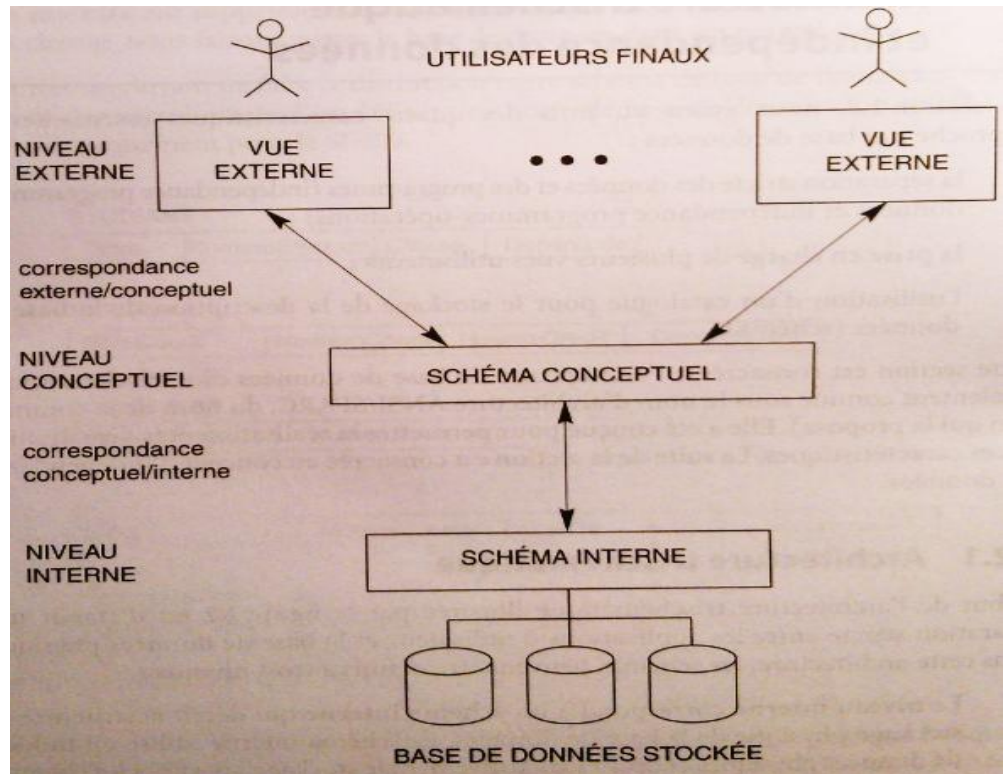
L'association : « est supérieur_de » entre un employé est un autre employé (récursivité sur la même entité « employé »)

Quelques connaissances supplémentaires utiles :

L'architecture ANSI/SPARC

L'architecture ANSI/SPARC, datant de 1975, définit les niveaux standards d'abstraction pour un SGBD, permettant une indépendance entre les données et les traitements :

- **Niveau interne (physique)** : il définit la façon selon laquelle sont stockées les données et les méthodes pour y accéder
- **Niveau conceptuel (logique)** : représenté par le MCD (Modèle Conceptuel des Données) et le MLD (Modèle Logique des Données). Il définit l'arrangement des données au sein de la future base de données
- **Niveau externe (utilisateur)** : il définit les « vues partielles » de la BD, suivant les points de vue de chaque utilisateur (selon les données dont il a besoin)



L'intégrité référentielle

L'intégrité référentielle est un mécanisme utilisé par les SGBD pour assurer la cohérence globale des données dans une BD. Exemple : pour assurer la cohérence des données entre la table facture vis-à-vis (en se référant à) sa table père client, les SGBD appliquent les trois règles suivantes (lors des MAJ) qui sont la base du mécanisme de l'intégrité référentielle :

- Si on supprime un client (supprimer son enregistrement (sa ligne)) de la table client, le SGBD nous signal que les factures de ce client doivent aussi être supprimées de la table facture (on ne doit pas avoir des factures sans un client)
- Le SGBD n'accepte pas d'ajouter (saisir) une facture (un fils) avec un num_client inexistant dans la table père client.
- Si on modifie le numéro d'un client dans la table client, ce numéro_client va être modifié par le SGBD dans toutes les factures de ce client (ce num_client sera donc modifié dans la table facture)

Partie II : QCM : Le numéro à gauche représente le numéro de la question, et le numéro à droite représente le numéro de choix (réponse) dans le QCM :

- | | |
|-----------------------|------------------|
| 1. Réponses : 2 | 6. Réponses : 3 |
| 2. Réponses : 1 | 7. Réponses : 3 |
| 3. Réponses : 1, 2 | 8. Réponses : 1 |
| 4. Réponses : 1, 3, 4 | 9. Réponses : 2 |
| 5. Réponses : 1 | 10. Réponses : 3 |

- 1) Une base de données est
- ☐ un ensemble de logiciels systèmes permettant de stocker et d'interroger un ensemble de fichiers inter-dépendants.
 - ☒ un ensemble structuré de données enregistrées avec le minimum de redondance pour satisfaire simultanément plusieurs utilisateurs de façon sélective en un temps opportun.
 - ☐ pas de réponse.
- 2) L'architecture ANSI/SPARC se compose des niveaux suivants :
- ☒ niveau interne, niveau utilisateur et niveau conceptuel.
 - ☐ niveau physique, niveau logique et niveau conceptuel.
 - ☐ niveau externe, niveau physique et niveau interne.
- 3) Le niveau externe est
- ☒ le niveau de l'utilisateur.
 - ☒ une représentation par plusieurs sous-schémas externes appelés aussi vues.
 - ☐ une représentation de bas niveau de l'ensemble de la base de données.
 - ☐ une représentation par un seul sous-schéma externe appelé aussi vue.
 - ☐ pas de réponse
- 4) Le niveau interne est
- ☒ une représentation de bas niveau de l'ensemble de la base de données.
- ☐ Pas de réponse.

4) Le niveau interne est

- ☒ une représentation de bas niveau de l'ensemble de la base de données.
- ☐ constitué de plusieurs occurrences des divers types d'enregistrements internes.
- ☒ décrit par un seul schéma interne.
- ☒ l'organisation et l'exploitation physique des données sur le support de stockage.
- ☐ pas de réponse.

5) L'indépendance physique permet

- ☒ de modifier le schéma interne sans avoir à modifier le schéma conceptuel.
- ☐ de modifier le schéma conceptuel sans avoir à modifier le schéma interne.
- ☐ de modifier le schéma conceptuel sans avoir à modifier les schémas externes.
- ☐ pas de réponse.

6) L'indépendance logique permet de

- ☐ modifier le schéma interne sans avoir à modifier le schéma conceptuel.
- ☐ modifier le schéma conceptuel sans avoir à modifier le schéma interne.
- ☒ modifier le schéma conceptuel sans avoir à modifier les schémas externes.
- ☐ modifier un schéma externe sans avoir à modifier le schéma conceptuel.

7) Une entité est

- ☐ un objet concret pourvu d'une existence propre et conforme aux choix de gestion de l'entreprise.
- ☐ un objet abstrait pourvu d'une existence propre et conforme aux choix de gestion de l'entreprise.

- ☒ un objet concret ou abstrait pourvu d'une existence propre et conforme aux choix de gestion de l'entreprise.

8) Une association entre entités est

- ☒ une relation perçue dans le monde réel entre deux ou plusieurs entités.
- ☐ un objet concret pourvu d'une existence propre.

9) La cardinalité d'une entité par rapport à une association s'exprime par

- ☐ la cardinalité moyenne indiquant le nombre moyen d'occurrences qu'une occurrence de l'entité participe aux occurrences de l'association.
- ☒ la cardinalité minimale et la cardinalité maximale qui sont respectivement le nombre de fois minimum et le nombre de fois maximum qu'une occurrence d'une entité participe aux occurrences de l'association.

10) L'identifiant d'une entité est

- ☐ un attribut particulier de l'entité tel qu'à chaque occurrence de l'entité corresponde une et une seule valeur de l'attribut.
- ☐ un attribut particulier de l'entité tel qu'à chaque valeur de l'entité corresponde une et une seule valeur des attributs.
- ☒ un attribut particulier de l'entité tel qu'à chaque valeur de l'attribut corresponde une et une seule occurrence de l'entité.

Partie II : Exercices

Exercice 01 :

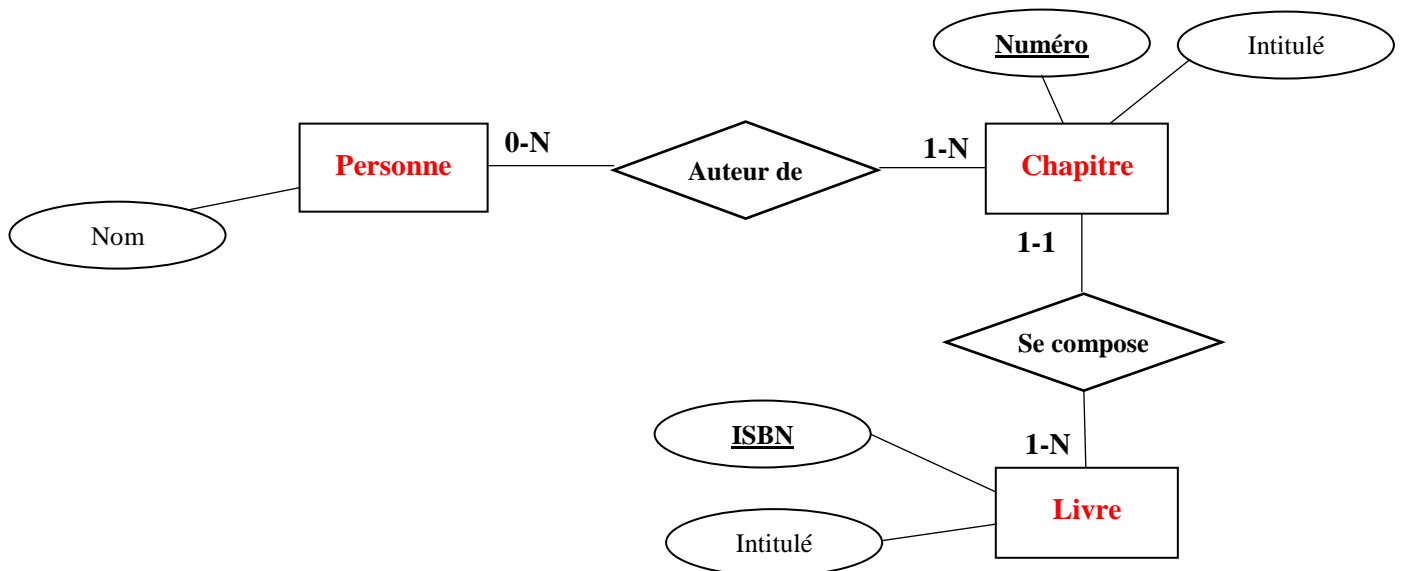
- **Est-ce que dans ce schéma un chapitre peut avoir plusieurs auteurs ?**

Réponse : OUI, suivant la cardinalité max N de 1-N dans l'association « Auteur de »

- **Est-ce qu'il peut y avoir des personnes qui ne sont pas des auteurs ?**

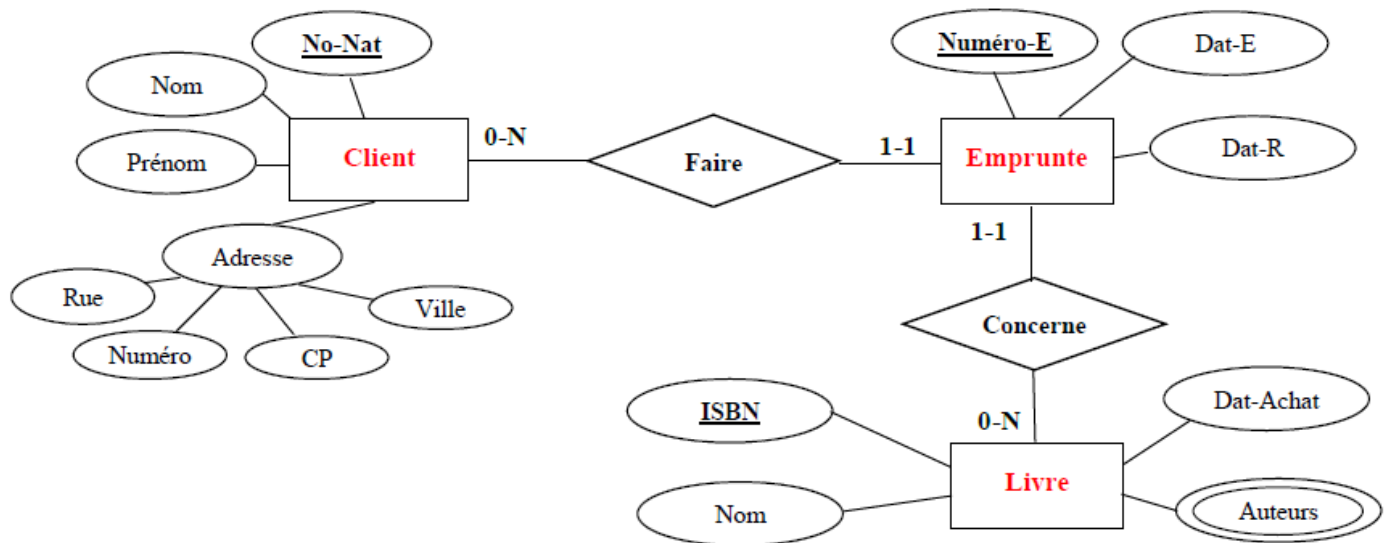
Réponse : OUI, suivant la cardinalité min 0 de 0-N de l'association « Auteur de »

- Dans le schéma ci-dessous, on a ajouté des informations (entités, associations et cardinalités manquantes) concernant les livres qui contiennent les chapitres, en respectant les contraintes suivantes :
 - Chaque livre a un titre et un numéro ISBN.
 - Un livre est composé d'au moins un chapitre.
 - Chaque chapitre fait partie d'un seul livre.



L'entité « Chapitre » est une entité faible, car un chapitre ne peut exister sans son livre (supprimer un livre de la BD implique la suppression de ces chapitres). En plus le numéro du chapitre ne peut pas le distinguer des chapitres des autres livres.

Exercice 2 : Bibliothèque



Modèle relationnel :

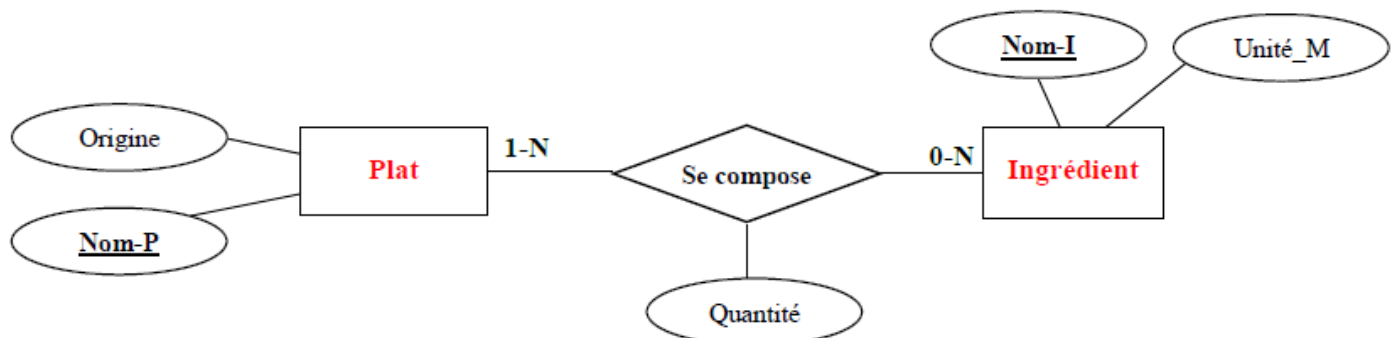
Client(No-Nat, Nom, Prénom, Rue, CP, Numéro, Ville).

Emprunte(Numéro-E, Dat-E, Dat-R, #No-Nat, #ISBN).

livre(#ISBN, Nom, Dat-Achat).

Auteurs(#ISBN, Auteur).

Exercice 3 : Plats



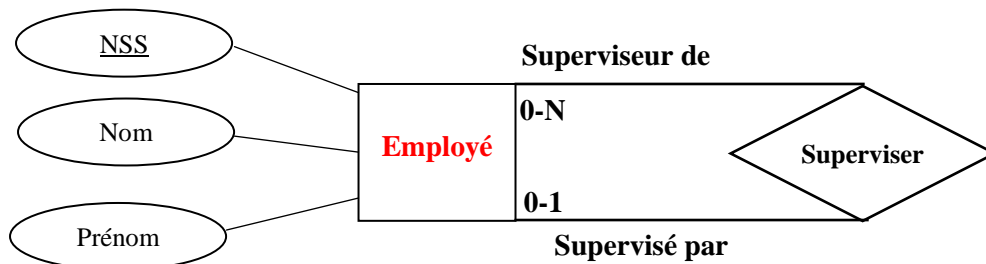
Modèle relationnel (logique) :

Plat(**Nom-P**, Origine)

Ingrédient(**Nom-I**, Unité_M)

Composition(**#Nom-P**, **#Nom-I**, Quantité)

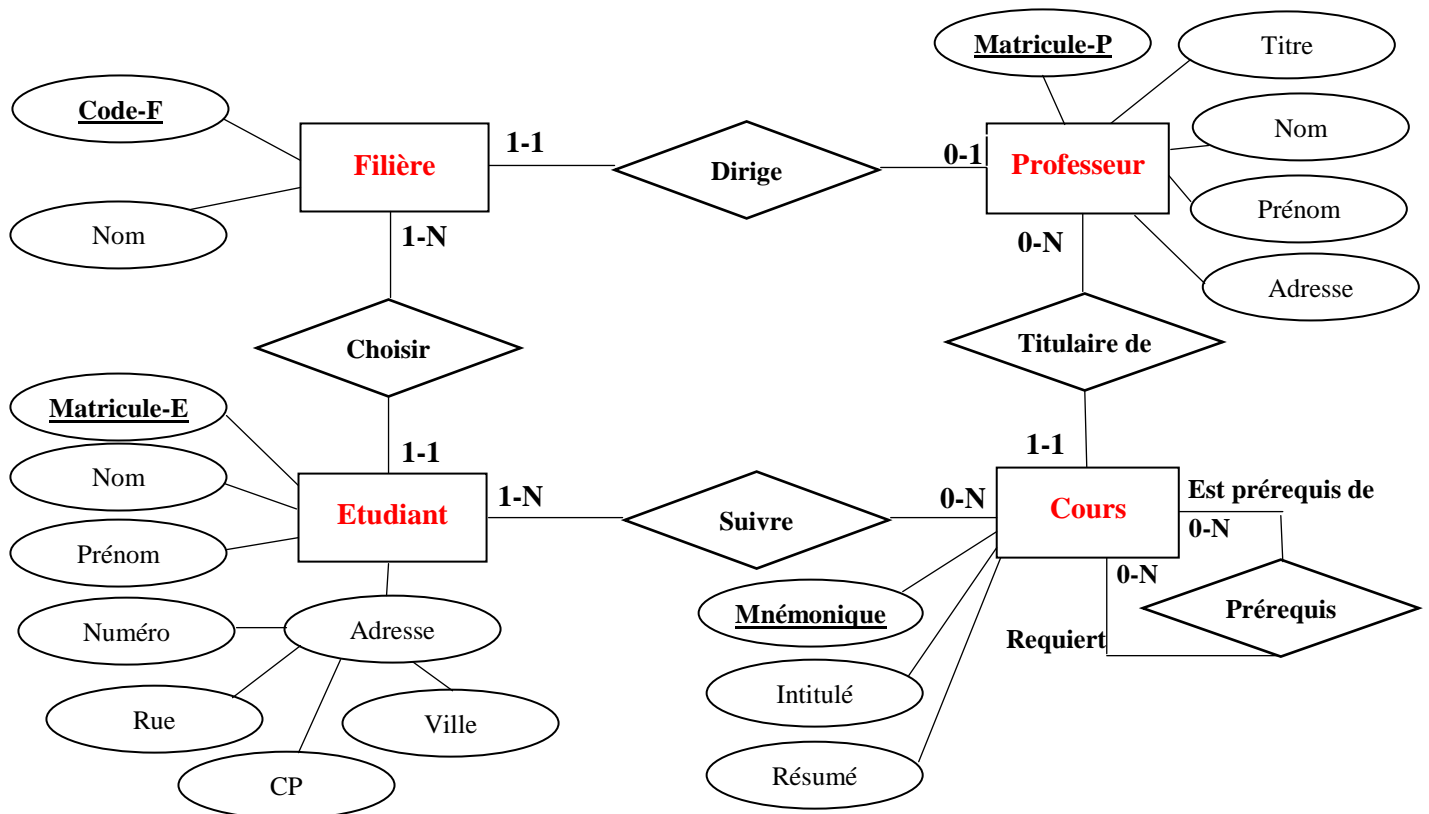
Exercice 4 : Employés



Modèle logique :

Employé (NSS, Nom, Prénom, # NSS_Superviseur).

Exercice 5 : Université



Modèle logique :

Etudiant (Matricule, Nom, Prénom, Numéro, Rue, CP, Ville, #Code-F).

Filière (Code-F, Nom, #Matricule-P).

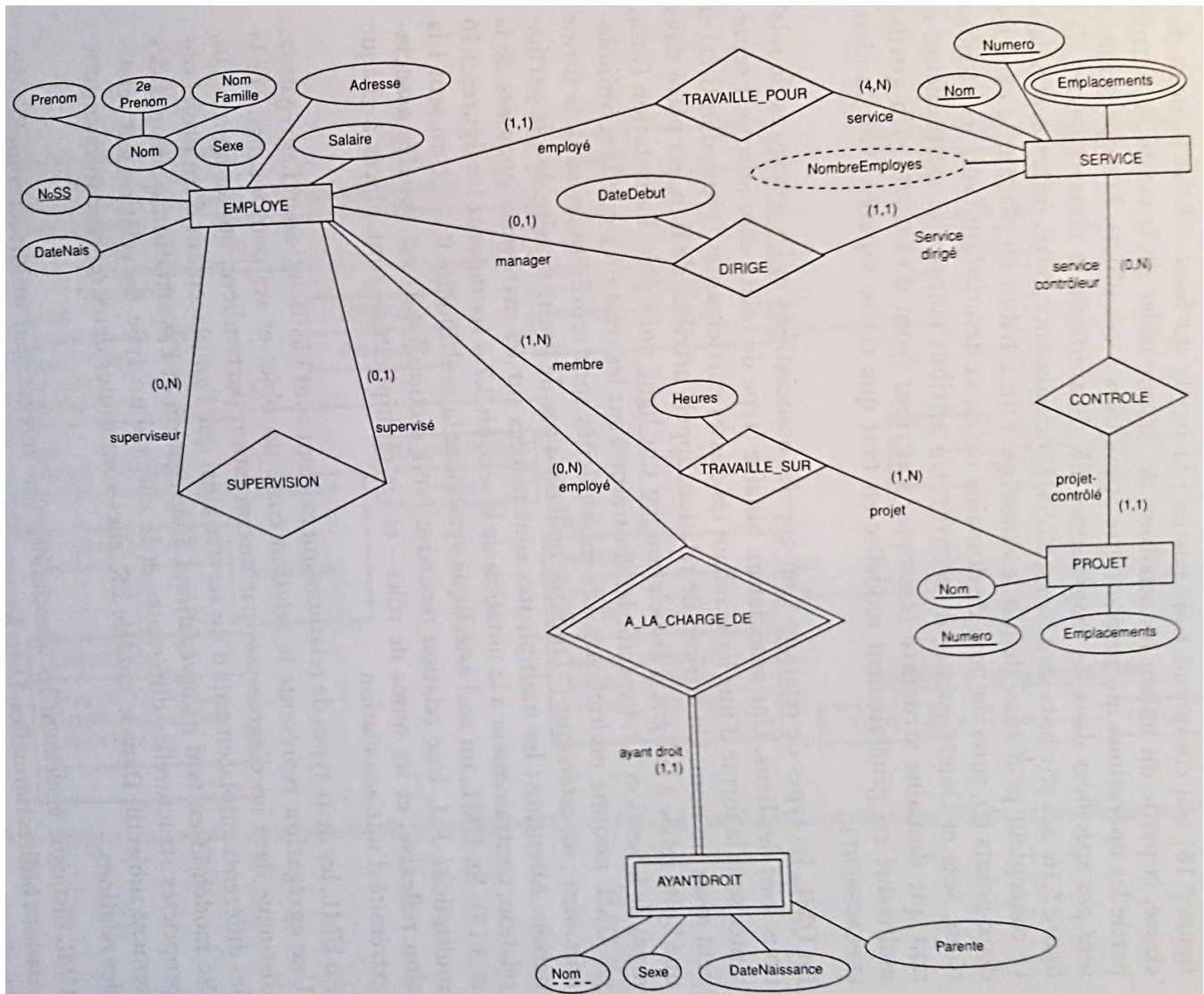
Professeur (Matricule-P, Titre, Nom, Prénom, Adresse).

Cours (Mnémonique, Intitulé, Résumé, #Matricule-P).

Suivre (**#Matricule-E**, **#Mnémonique**).

Prérequis (**#Mnémonique**, **#Mnémonique-Requis**).

Exercice 6 : Entreprise



Modèle relationnel :

Employe(NoSS, Nom, Prénom, Rue, Ville, Salaire, Sexe, DateNais, #NoSS_Superviseur, #Num_Serv)

Service(NumServ, Nom_Serv, #NoSS_Directeur, Date_Début)

Emplacements(#Num Serv, Emplacement)

Projet(Num Proj, Nom_Proj, Emplacement, #Num_Serv)

Ayant_Droit(NoSS, Prénom, Sexe, DateNaissance, Parenté)

Travail_Sur(#NoSS, #Num Proj, Heures)

Exercice 7 : le schéma relationnel correspondant au diagramme EA donné est :

A (atA1, atA2) // atA3 est un champ calculé, il ne doit pas figurer dans le schéma relationnel

B (atB1, atB2, #atA1, atR1)

C (#atB1, atC1) // l'entité C est une entité faible par rapport à l'entité B

D (atD2, atD1, #atB1, #atC1, #atE1)

E (atE1, atE2, #R6_atE1) // #R6_atE1 est la clé atE1 passée suivant l'association récursive R6

E3 (atE3, #atE1) // atE3 est un champ multi-valué qui deviendra une entité

R3 (#atB1, #atE1, atR3)