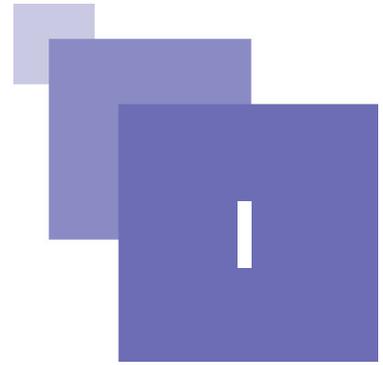


Final NF18/AI23 P2022



- Tous les documents papier sont autorisés.
- Aucun appareil connecté n'est autorisé (ordinateur, téléphone, montre connectée, etc.).
- Les réponses peuvent être données en anglais pour les étudiants étrangers (le préciser sur la copie).
- Le nom doit être écrit lisiblement sur chaque copie.
- La durée totale de l'examen est de 2 (2h40 pour les tiers-temps).
- **Répondez sur deux copies séparées : copie 1 pour les exercices I.1, I.2 et I.3, copie 2 pour les exercices II.1, II.2 et II.3 !** Vous pourrez être pénalisés si vous ne respectez pas cette consigne.

Barème :

- Ex I.1 - 2 pts
Q1 : 2pts
- Ex I.2 - 3 pts
Q1 : 3pts
- Ex I.3 - 5 pts
Q1 : 1pt ; Q2 : 1pt ; Q3 : 1pt ; Q4 : 1pts ; Q5 : 1pts
- Ex II.1 - 4.5 pts (environ 25mn)
Q1 : 2.5 pts ; Q2 : 1 pt ; Q3 : 1 pt
- Ex II.2 - 2.5 pts (environ 15mn)
Q1 : 1 pt ; Q2 : 1.5 pt
- Ex II.3 - 3 pts (environ 20mn)
Q1 : 1.5 pts ; Q2 : 1.5 pts

A. Exercice 1.1 : Banque

Vous êtes en charge de la conception d'une base de données pour le compte d'une banque, afin de gérer ses agences, ses comptes bancaires, ses clients et leurs emprunts. Une agence est identifiée par un numéro, elle a un nom, elle est située dans une ville et elle a un actif (qui est la somme des soldes de ses comptes). L'agence gère les comptes et les emprunts de ses clients.

Un client est caractérisé par un identifiant client unique, son nom, prénom et ville (qui n'est pas forcément la même ville que celle de l'agence), et il est titulaire d'un ou plusieurs comptes dans une unique agence (il ne peut pas créer de comptes dans différentes agences).

Un compte est domicilié dans une agence et identifié par un numéro de compte, avec un solde déterminé, et il peut avoir un ou plusieurs clients titulaires.

Un client peut aussi contracter des emprunts dans son agence. Un emprunt est identifié par l'identifiant client plus un numéro d'emprunt, et il est caractérisé par un montant.

La banque souhaite aussi gérer les virements entre comptes bancaires. Un virement est un transfert d'un certain montant d'argent entre deux comptes différents réalisé à une date donnée.

Question

Proposez un modèle UML qui répond à ce besoin. Vous utiliserez au moins une association de composition et des classes d'association. Vous y marquez toutes les contraintes nécessaires.

Donnez le modèle relationnel correspondant. Vous indiquerez toutes les contraintes nécessaires.

B. Exercice 1.2 : Googlelite Scholar

La société Googlelite Scholar gère une base de données d'articles scientifiques en ligne. La base de données est structurée selon le modèle relationnel suivant :

Auteur(#num_auteur: int, nom: text, prénom: text, affiliation : text), avec nom et prénom not nuls

Articles(#num_article: int, titre: text, abstract : text, url_contenu: url), avec titre, abstract et contenu non nuls

Mot_clé(#mot: text, #article => Article)

Ecrit(#auteur => Auteur, #article=>Article)

avec *Projection*(*Ecrit*,*auteur*) = *Projection*(*Auteur*,*num_auteur*) et *Projection*(*Ecrit*, *article*) = *Projection*(*Article*, *num_article*)

RéférencesArticles(#articleRéféréncant => Article, #articleRéféréncé => Article), avec articleRéféréncant <> article Référéncé

ArticleDeConférence(#num_article => Article, nomConf : text, année: int), avec nomConf, année non nul

ArticleDeJournal(#num_article => Article, nomJournal: text, numéro: int, volume : int, nb_pages : int, année: int), avec nomJournal, numéro, volume, nb_pages et année non nuls

EvalueJournal(#auteur => Auteur, #article => ArticleDeJournal, avis : enum{"trop spécifique", "populaire", "très populaire"}), avec avis, non nul, exprimant l'évaluation qu'un auteur attribut à un article de journal (on suppose qu'un auteur ne peut donner son avis sur un article dont il est l'auteur).

Proj(*Articles*, num_article) = Union (*Proj*(*ArticleDeJournal*, num_article) , *Proj*(*ArticleDeConférence*, num_article))

Intersection(*Proj*(*ArticleDeJournal*, num_article) , *Proj*(*ArticleDeConférence*, num_article)) = {}

Question

Retro-concevez l'UML qui correspond à ce modèle relationnel. Vous utiliserez au moins une association d'héritage et des classes d'associations. Rajoutez toutes les contraintes nécessaires aux modèles conceptuel.

C. Exercice 1.3 : Recherche bibliographique sur Googlelite scholar

En considérant le MLD donné dans l'énoncé de la question I.2 (de la base de Googlelite Scholar), exprimez les requêtes suivantes :

Question 1

Exprimez deux requêtes qui renvoient respectivement les articles de journaux (en AR seulement) et les articles de conférences (en SQL seulement), avec toutes leurs informations.

Question 2

Écrire en AR et en SQL une requête qui affiche les articles de conférence où Albert Einstein n'est pas auteur.

Question 3

Ecrire en SQL seulement une requête qui affiche la liste d'articles de conférence suivie de leur nombre de citation (nombre de fois qu'un article a été cité par d'autres articles).

Question 4

Ecrire en SQL seulement :

- Une VUE "journal_année_articles" qui donne pour chaque journal le nombre d'articles parus par année
- En utilisant la VUE "journal_année_articles" écrire une requête qui affiche pour chaque journal le nombre d'articles moyens parus par année

Question 5

En SQL seulement :

- Écrire la VUE "article_journal_avis" qui affiche les articles de journal avec leur évaluation (avis des auteurs).
- Écrire une requête, en utilisant la vue "article_journal_avis", qui affiche pour chaque auteur le nombre d'articles notés comme "plus populaires" si ce nombre est supérieur à 2.

D. Exercice 2.1 : Normalisation fonctionnelle d'une BDD de projets et plateformes

Un entreprise veut gérer ses projets et ses plateformes dans une base de données. Elle considère les attributs suivants : le nom du projet (projet), le chef de projet (chef), le service dans lequel est effectué le projet (service), le bâtiment dans lequel est effectué le projet (bâtiment), l'étage où est effectué le projet (etage), la date de début du projet (debut), une année où le projet a eu un financement (annee_f), le montant du financement pour cette année (montant_f), le nom d'une plateforme utilisée dans le projet (plateforme), le nom d'un équipement matériel de la plateforme utilisé dans le projet (équipement), le coût d'utilisation de l'équipement (coût_e), et le sponsor de la plateforme.

L'entreprise a identifié les dépendances fonctionnelles suivantes :

projet → chef, debut

chef, debut → projet

projet, annee_f → montant_f

chef → service

service → bâtiment, etage

etage, bâtiment → service

projet, service → bâtiment

équipement → plateforme
équipement → sponsor
équipement → cout_e
plateforme → sponsor

Question 1

Donnez la fermeture transitive, la ou les couvertures minimales et la ou les clés de la relation contenant l'ensemble des attributs identifiés. Justifiez vos réponses.

Question 2

Quelle est la forme normale de cette relation ? Justifiez.

Question 3

Proposez un découpage en 3NF sans perte d'informations.

E. Exercice 2.2 : Base de données NoSQL d'habitations intelligentes

On vous demande de mettre en place une base de données MongoDB d'habitations intelligentes. Une habitation intelligente a deux adresses IP : une adresse IP externe et une adresse IP interne. Notez que ces adresses doivent être rassemblées dans un même sous-document du document représentant l'habitation intelligente. L'habitation utilise aussi un logiciel d'administration qui a un nom, et un entier qui représente la version. Chaque habitation a plusieurs équipements, chacun possédant une adresse IP interne, un type, une marque, et une description textuelle. Enfin, une habitation gère des alertes, chacune ayant une condition de déclenchement écrite en texte, et un contact à prévenir (numéro de téléphone ou email).

Question 1

Insérez dans une collection 'SmartHome' l'habitation intelligente suivante:

Une habitation avec une adresse IP externe '196.34.57.28', une adresse IP interne '196.34.0.1' et le logiciel 'HouseGenie' version 2. L'habitation comporte deux équipements : un premier de type chauffage centralisé, d'adresse '196.34.0.2' et de marque 'HeatWell' ; un second de type lave linge, d'adresse '196.34.0.5' et de marque 'WashWool'. L'habitation a une alarme , avec la condition de déclenchement 'température >26' et comme contact '0645231298'.

Question 2

Proposez une réponse en MongoDB aux requêtes suivantes :

- retournez l'adresse IP externe des habitations utilisant le logiciel d'administration 'eJeeves' de version supérieure à 3.
- retournez à quels numéros et emails sont envoyés les alertes d'habitations intelligentes qui ont un équipement de type 'alarme incendie'.

Indice :

On utilisera l'objet `{>:value}` à la place de la valeur de l'attribut à tester (`>` pour greater than).

F. Exercice 2.3 : Optimisation d'une bases de données de bibliothèques

Soit la base de données d'ouvrage d'une bibliothèque suivante (tous les attributs sont NOT NULL par défaut):

Livres(#ISSBN : int, titre : str, annee : int, editeur : str, collection : str, pays : str, categorie : enum{mathématique, physique, histoire, roman, science-fiction, mystère, théâtre}) avec {collection optionnel}

Auteurs(#id : int, nom : str, prenom : str, date_naissance : date)

Auteur_de (#livre => Livres.ISSBN, #auteur => Auteurs.id)

Chaque question de cet exercice travaillera à partir de la BdD initiale donnée ci-dessus. Des champs d'interrogation sur un attribut dans une requête seront notés [attribut] (par exemple [nom] pour une recherche sur un nom d'auteur donné).

Question 1

On considère une requête renvoyant le titre et année d'édition des livres appartenant à une catégorie donnée ([catégorie]), triés par année d'édition. Proposez des optimisations de la base de données pour rendre cette requête aussi performante que possible. Donnez le nouveau modèle relationnel correspondant (pas besoin de réécrire les relations qui restent les mêmes). Ecrivez la requête dans la base de données que vous proposez.

Question 2

On considère une requête renvoyant le titre, le nom d'auteur et le prénom d'auteur de livres pour un nom d'auteur donné ([nom]), classés par prénom d'auteur. Proposez des optimisations de la base de données pour rendre cette requête aussi performante que possible. Donnez le nouveau modèle relationnel correspondant (pas besoin de réécrire les relations qui restent les mêmes). Ecrivez la requête dans la base de données que vous proposez.