

Ex01 (04 points)

1. C'est quoi une base de données ? (1 pt)

Une base de données est un ensemble structuré de données (1) enregistrées sur des supports accessibles par l'ordinateur (2) pour satisfaire simultanément plusieurs utilisateurs (3) de manière sélective (4) en un temps opportun (5).

- *(1) : Organisation et description de données*
- *(2) : Stockage sur disque*
- *(3) : Partage des données*
- *(4) : Confidentialité*
- *(5) : Performance*

2. Donner la définition de la décomposition. Dans quel cas elle sera sans perte d'information ? (1.5 points)

Si R est décomposée à R1 ; R2 ; Rn.

La décomposition sera sans perte SSI

- a. **La jointure naturelle des relations résultat de la décomposition (R1 ; R2 ; Rn.) a le même schéma que la relation de base (R).**
- b. **L'extension de la relation jointure naturelle des relations résultat de la décomposition (R1 ; R2 ; Rn.) a la même extension que la relation de base (R).**

3. Quelle l'objectif de l'algorithme de synthèse et quelles sont ces étapes ? (1.5 pts)

Objectif : Concevoir un schéma relationnel en 3FN, à partir de cet ensemble de DFs en appliquant l'algorithme de synthèse.

Étapes de l'algorithme de synthèse

- **Trouver l'ensemble IRR(F) contenant la couverture irréductible (minimale) de F;**
- **Partitionner l'ensemble IRR(F) en sous ensemble de Fi, tels que toutes les DFs de Fi ont la même partie gauche;**
- **Pour chaque ensemble Fi de DFs, construire une relation composée de tous les attributs formant les DFs de Fi, la clé de la relation sera la partie gauche commune à ces DFs;**
- **Traiter les attributs isolés**

Ex02 (04 points)

Soit la relation

Departement (Mle-Etud, Note, Classe, Cours, Module, No-Ens, Nom-Ens, Nom-Etud, Nb-h) avec les DFs :

Mle-Etud → Nom-Etud, Classe

No-Ens → Nom-Ens

Cours → Module

Cours,Module → Nb-h
Classe, Cours, Module → No-Ens, Nom-Ens
Mle-Etud,Cours,Module → Note

- **Eliminer les DFs redondantes.**
- **Normalier la relation en 3FN.**

Correction

1.

Dans cet exemple aussi on va examiner les DFs pour les réduire à un ensemble minimum. L'indication donnée suggère que

Si on a Cours → Module (3), on peut éliminer l'attribut Module de la partie gauche des DFs (4), (5) et (6). Pour avoir finalement

1. Mle-Etud → Nom-Etud, Classe
2. No-Ens → Nom-Ens
3. Cours → Module
4. Cours → Nb-h
5. Classe, Cours → No-Ens, Nom-Ens
6. Mle-Etud, Cours → Note

Par ailleurs, dans (5) la dépendance Classe, Cours → Nom-Ens est transitive à cause de (2), on l'élimine pour ne garder que

5. Classe, Cours → No-Ens

2.

on va créer un schéma de relation correspondant à chaque DF, vu que l'ensemble final obtenu est sans transitivité et toutes les DFs sont totales. On groupe aussi les DFs (3) et (4), car ayant même partie gauche. D'où

Elèves (Mle-Etud, Nom-Etud, Classe)
Profes (No-Ens, Nom-Ens)
Matières (Cours, Module, Nb-h)
Enseignement (Classe, Cours, No-Ens)
Evaluation (Mle-Etud, Cours, Note)

Ex03 (04 points)

Soit la base relationnelle suivante relative à la gestion simplifiée des étapes du Tour de cycliste:

EQUIPE(CodeEquipe, NomEquipe, DirecteurSportif)
COUREUR(NuméroCoureur, NomCoureur, CodeEquipe*, CodePays*)
PAYS(CodePays, NomPays)
TYPE_ETAPE(CodeType, LibelléType)

ETAPE(NuméroEtape, DateEtape, VilleDép, VilleArr, NbKm, CodeType*)
PARTICIPER(NuméroCoureur*, NuméroEtape*, TempsRéalisé)
ATTRIBUER_BONIFICATION(NuméroEtape*, km, Rang, NbSecondes, NuméroCoureur*)

Remarque : les clés primaires sont soulignées et les clés étrangères sont marquées par *

A. Répondre aux requêtes suivantes en utilisant SQL :

1. Donner la ville de départ et la ville d'arrivée de l'étape la plus longue.

```
SELECT Villedep, VilleArr FROM ETAPE  
WHERE Nbkm = (SELECT max(Nbkm) FROM ETAPE);
```

2. Liste des coureurs n'ayant eu aucune bonification.

```
SELECT NomCoureur FROM COUREUR  
WHERE NuméroCoureur NOT IN  
(SELECT NuméroCoureur FROM ATTRIBUER_BONIFICATION) ;
```

Ex04 (08 points)

On cherche à représenter le système d'information d'une agence d'intérim qui enregistre dans celui-ci les CV des intérimaires. Pour la modélisation de ce système, les éléments suivants ont été identifiés :

- Individu : Les intérimaires sur lesquels on enregistre les informations suivantes :
 - nom et prénom de l'individu
 - date de naissance de l'individu
 - langues pratiquées par l'individu
 - niveau dans les langues pratiquées
 - désignation des centres d'intérêt concernant un individu (sport, loisir divers...),
 - salaire actuel de l'individu
 - salaire recherché
- Diplômes, avec les informations suivantes :
 - Code du diplôme
 - Désignation du diplôme
 - Personnes ayant obtenu ce diplôme et date d'obtention du diplôme
- Employeurs, les employeurs des intérimaires avec les informations suivantes :
 - raison sociale de l'employeur
 - date d'entrée et de départ de l'intérimaire dans l'entreprise
 - fonction exercée par l'intérimaire chez l'employeur ainsi que les dates de début et de fin de fonction

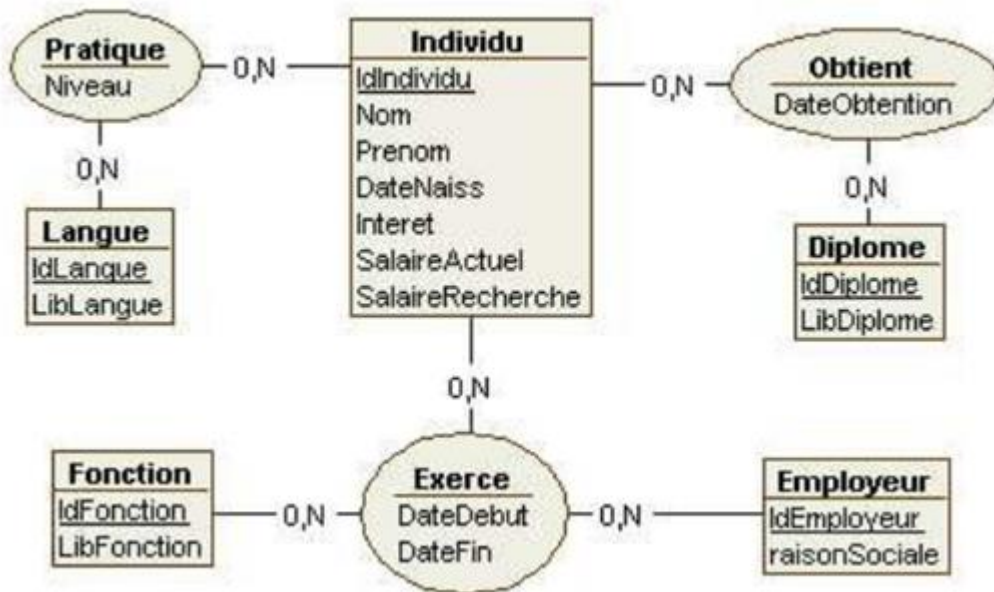
On précise également qu'un intérimaire peut avoir travaillé plusieurs fois chez le même employeur pour la même fonction ou non.

1. Donner une modélisation de ce système en utilisant le modèle E/A.
2. Donner le schéma conceptuel équivalent dans le modèle Relationnel.

3. Répondre aux requêtes suivantes en utilisant le SQL :
 - a. La liste des employés qui parle une langue donnée par niveau ;
 - b. Le nombre d'employés par diplôme ;
 - c. Le salaire maximum des employés par fonction ;

Correction Exercice N° 03

1. Le modèle entité-association correspondant



2. Le modèle relationnel correspondant :

Individu(IdIndividu, nom, prénom, dateNaiss, Inteêt, SalaireActuel, Salaire Recherché) ;
Langue (IdLangue,LibLangue) ;
Diplôme (Iddiplome, LibDiplome) ;
Fonction (IdFonction, LibFonction) ;
Employeur (IdEmployeur, RaisonSociale) ;
Obtient (IdIndividu, Iddiplome, dateObtention) ;
Pratique(IdIndividu, IdLangue, Niveau) ;
Exercice (Idindividu, IdFonction, IdEmployeur, DateDebut, DateFin).

3. Réponse aux requête SQL

- a. La liste des employés qui parle une langue donnée par niveau ;

```

Select nom, prénom
From Individu, pratique
Groupby niveau
Having Idlangue=Y.

```

- b. Le nombre d'employés par diplôme ;
Select Libdiplome, Count(IdIndividu)
From Obtient, Diplôme
Groupby Iddiplome
- c. Le salaire maximum des employés par fonction ;
Select LibFonction, Max(Salalaireactuel)
From Exerce, Individu
Groupby IdFonction

والله ولي التوفيق