



Concours Biologie et Géologie
Epreuve d'Informatique

Date : Vendredi 26 Mai 2017 Heure : 14 H Durée : 2 H Nombre de pages : 4

Barème : PROBLEME 1 : 10 points
PROBLEME 2 : 10 points

DOCUMENTS NON AUTORISES
L'USAGE DES CALCULATRICES EST INTERDIT

PROBLEME 1

Soit le schéma relationnel de la base de données d'un hôpital défini par les relations suivantes :

- **Patient** (IdPatient, Nom, Prenom, Age, Adresse, Date_Entree, Date_Sortie)

La table *Patient* contient tous les patients actuellement présents à l'hôpital.

- IdPatient : identifiant du patient (entier), clé primaire.
- Nom : nom du patient (chaîne de caractères).
- Prenom : prénom du patient (chaîne de caractères).
- Age : âge du patient (entier).
- Adresse : adresse du patient (chaîne de caractères).
- Date_Entree : date d'entrée du patient (chaîne de caractères).
- Date_Sortie : date de sortie du patient (chaîne de caractères égale NULL si la date de sortie est inconnue).

- **Plat** (IdPlat, Nom, Calories)

La table *Plat* contient tous les plats servis par la cuisine de l'hôpital.

- IdPlat : identifiant du plat (entier), clé primaire.
- Nom : nom du plat (chaîne de caractères).
- Calories : nombre total de calories (entier strictement positif).

- **Servi** (IdPlat, IdPatient, Date, Quand)

La table *Servi* énumère les plats identifiés par IdPlat servis, à une date donnée, aux patients identifiés par IdPatient.

- IdPlat: identifiant du plat (entier), clé primaire.
- IdPatient: identifiant du patient (entier), clé primaire.
- Date : date à laquelle le plat a été servi (chaîne de caractères).
- Quand : caractère indiquant quand le plat a été servi ; Quand = 'P' pour le petit déjeuner ; Quand = 'M' pour le repas de midi et Quand = 'S' pour le repas du soir.

• **Interdit (IdPlat , IdPatient)**

La table *Interdit* indique les plats, identifiés par IdPlat, interdits aux patients identifiés par IdPatient.

Travail demandé :

1. En utilisant le module sqlite3 donner le script PYTHON permettant de créer la table **Plat** dans la base "hopital.db" en exprimant toutes les contraintes d'intégrité mentionnées ci-dessus.

Dans la suite on suppose que les quatre tables sont créées et remplies. Donner les requêtes SQL permettant de :

2. Ajouter le patient ayant les informations suivantes : (1256 , 'BEN FOULEN' , 'Mohamed' , 61 , 'Tunis' , '26 / 05 / 2017' , 'NULL')
3. Modifier le nombre de calories du plat ayant l'identifiant 26 par la valeur 1200.
4. Supprimer tous les plats dont le nombre de calories est inférieur à 400.
5. Déterminer tous les patients de l'hôpital.
6. Déterminer les identifiants, les noms, les prénoms et les dates d'entrée des patients âgés entre 25 et 50 ans.
7. Déterminer le nombre total de calories consommé par la patiente 'Fattouma' 'FALTENA' durant son séjour.
8. Déterminer, pour chaque patient, le nombre de calories consommé le matin du '26/05/2017'. Le résultat doit être trié selon le nombre de calories par ordre décroissant.
9. Déterminer le(s) nom(s) du(des) plats le plus calorique.
10. Déterminer le nom et le prénom des patients qui ont pris le plat soupe le soir du '25/05/2017'.
11. Déterminer le nom, prénom et IdPatient de tous les patients dont la date de sortie est inconnue.
12. Déterminer le nom des plats qui n'ont jamais été servis.
13. Déterminer le nom, prénom et IdPatient de tous les patients interdits de prendre le plat 'salade tunisienne'.
14. Déterminer le nombre de patients qui n'ont aucun plat interdit.

PROBLEME 2

Dans ce problème on se propose d'informatiser un championnat quelconque de football comportant N équipes.

Les noms des différentes équipes sont stockés dans une liste LE .

Exemple :

$LE = ['TUNISIE', 'SENEGAL', 'ALGERIE', 'ZIMBABWE']$

Une rencontre entre deux équipes A et B est représentée par le tuple R défini comme suit :

$R = (< \text{indice de } A >, < \text{indice de } B >)$

où :

$< \text{indice de } A >$: Indice de l'équipe A dans la liste LE .

$< \text{indice de } B >$: Indice de l'équipe B dans la liste LE .

Toutes les rencontres entre les équipes de LE sont stockées dans une liste LR .

Exemple :

$R = (0, 2)$ représente la rencontre entre l'équipe d'indice 0 ('TUNISIE') et l'équipe d'indice 2 ('ALGERIE').

$LR = [(0, 1), (0, 2), (0, 3), (1, 2), (1, 3), (2, 3)]$

Le résultat d'une rencontre entre deux équipes A et B est représenté par le tuple Res défini comme suit :

$Res = (< \text{indice de } A >, < \text{indice de } B >, < \text{résultat} >, < \text{buts de } A >, < \text{buts de } B >)$

où :

$< \text{indice de } A >$: Indice de l'équipe A dans la liste LE .

$< \text{indice de } B >$: Indice de l'équipe B dans la liste LE .

$< \text{résultat} >$: Un entier qui représente le résultat de la rencontre entre les deux équipes A et B ; 0 : match nul, 1 : victoire de l'équipe A et 2 : victoire de l'équipe B .

$< \text{Buts de } A >$: Nombre de buts marqués par l'équipe A .

$< \text{Buts de } B >$: Nombre de buts marqués par l'équipe B .

Les résultats des différentes rencontres entre les équipes sont stockés dans une liste $LRES$ qui est une liste de tuples.

Exemple :

$Res = (0, 2, 1, 2, 1)$ représente le résultat de la rencontre entre l'équipe d'indice 0 ('TUNISIE') et l'équipe d'indice 2 ('ALGERIE') avec victoire de la 'TUNISIE' par 2 buts à 1.

$LRES = [(0, 1, 2, 0, 2), (0, 2, 1, 2, 1), (0, 3, 1, 4, 2), (1, 2, 1, 2, 1), (1, 3, 1, 3, 1), (2, 3, 0, 2, 2)]$

Les informations des matchs joués par toutes les équipes sont représentées par une liste de tuples LTR ayant le même ordre que LE . Chaque tuple contient, dans l'ordre, le nombre de matchs joués, le nombre de victoires, le nombre de matchs nuls, le nombre de défaites, le nombre des buts marqués et le nombre des buts encaissés.

Exemple :

$LE = ['TUNISIE', 'SENEGAL', 'ALGERIE', 'ZIMBABWE']$

Le tuple $(3, 2, 0, 1, 6, 5)$ représente les informations relatives aux matchs joués de la 'TUNISIE' (équipe d'indice 0 dans LE).

La liste *LTR*, associée à *LE*, est alors la suivante :

$LTR = [(3, 2, 0, 1, 6, 5), (3, 3, 0, 0, 7, 2), (3, 0, 1, 2, 4, 6), (3, 0, 1, 2, 5, 9)]$

Le total de points cumulés pour toutes les équipes est stocké dans une liste *LP* dans l'ordre des équipes de *LE*.

Le total des points est comptabilisé à partir de la liste *LTR* qui contient le résultat de chacune des équipes sachant qu'une victoire vaut 3 points, un match nul vaut 1 point et une défaite vaut 0 point.

Exemple :

$LE = ['TUNISIE', 'SENEGAL', 'ALGERIE', 'ZIMBABWE']$

$LTR = [(3, 2, 0, 1, 6, 5), (3, 3, 0, 0, 7, 2), (3, 0, 1, 2, 4, 6), (3, 0, 1, 2, 5, 9)]$

$LP = [6, 9, 1, 1]$

Les équipes sont classées selon l'ordre décroissant du total des points cumulés. En cas d'égalité des points entre deux ou plusieurs équipes, la première classée est la première rencontrée.

Exemple :

SENEGAL	9
TUNISIE	6
ALGERIE	1
ZIMBABWE	1

Travail demandé :

Nomenclature :

Nom	Type	Description
<i>N</i>	<int>	Nombre des équipes
<i>LE</i>	<list>	Liste des noms des équipes
<i>LR</i>	<list>	Liste de toutes les rencontres (liste de tuples)
<i>LRES</i>	<list>	Liste des résultats de toutes les rencontres (liste de tuples)
<i>LTR</i>	<list>	Liste des informations des matchs joués (liste de tuples)
<i>LP</i>	<list>	Liste des points cumulés pour toutes les équipes

1. Ecrire une fonction PYTHON, nommée **saisiEquipes**, qui saisit et retourne dans une liste les noms des *N* équipes.
2. Ecrire une fonction PYTHON, nommée **genRencontre**, qui, à partir de *LE* génère la liste de toutes les rencontres *LR*.
3. Ecrire une fonction PYTHON, nommée **saisiResultats**, qui, à partir de *LE*, génère la liste de toutes les rencontres *LR*, puis saisit les résultats des différentes rencontres en faisant les contrôles nécessaires et retourne la liste *LRES*.
4. Ecrire une fonction PYTHON, nommée **resultatsEquipes**, qui, à partir de la liste des résultats des rencontres *LRES* et la liste des équipes *LE*, détermine et retourne la liste des tuples *LTR*.
5. Ecrire une fonction PYTHON, nommée **resultats**, qui, à partir de la liste des rencontres *LTR*, retourne la liste *LP*.
6. Ecrire une fonction PYTHON, nommée **classement**, qui, à partir des listes *LP* et *LE*, affiche le classement des équipes selon l'ordre décroissant du total des points cumulés. En cas d'égalité des points entre deux ou plusieurs équipes, la première affichée est la première rencontrée.