

Base de Données : TD3

Langage SQL

Exercice 01 : Tri et regroupement

On considère les relations suivantes :

- **Personne** (NumeroP, Nom, Prénom, Telephone, NumeroE) ;
- **Hopital** (NumeroH, Nom, NumeroE)
- **Endroit** (NumeroE, NumeroVoie, NomVoie, CodePostal, Ville)
- **EstMedecinA** (NumeroM, NumeroH)
- **Soigne** (NumeroM, NumeroP, NumeroH)

La relation **Personne** permet de stocker l'ensemble des personnes dans la BD. Une personne peut être soit un patient ou soit un médecin. Chaque personne est identifiée par un numéro et possède un nom, un prénom, un numéro de téléphone et numéro d'adresse présent dans la relation Endroit.

La relation **Hopital** permet de stocker les hôpitaux dans lesquels exercent des médecins. Un hôpital est identifié par un numéro et possède un nom et un numéro d'adresse.

La relation **Endroit** permet de stocker les adresses. Une adresse est identifiée par un numéro, et possède un numéro de voie, un type de voie, un code postal et une ville.

La **relationEstMedecinA** permet de stocker les médecins pour chaque hôpital. Un médecin peut exercer dans un ou plusieurs hôpitaux simultanément.

La relation **Soigne** permet de recenser l'ensemble des personnes soignées par médecin dans un hôpital.

Avec le langage SQL, écrivez les requêtes permettant de connaître :

1. La liste des noms des médecins triée par ordre alphabétique ;
2. La liste des personnes triée par nom de ville de façon alphabétique ;
3. Pour chaque hôpital, la liste des médecins triée par ordre alphabétique ;
4. La liste des médecins (ordre alphabétique) et pour chacun d'eux, la liste de leurs patients (ordre alphabétique aussi) ;
5. La liste des hôpitaux (ordre alphabétique) et pour chacun d'eux, la liste des patients (ordre alphabétique aussi) qui y sont soignés ;

Exercice 02 : Fonctions de calcul

On rajoute au modèle relationnel précédent, la relation **Consultation** permettant de stocker les consultations exercées par un médecin dans un hôpital.

- **Consultation** (NumeroM, NumeroP, NumeroH, DateRV, PrixConsultation)

Avec le langage SQL, écrivez les requêtes permettant de connaître :

- 1) Montant total des consultations assurées par le docteur Chloé Langlois au CHU de Poitiers ;
- 2) Nombre de consultations assurées au CHU de Poitiers ;
- 3) Prix de la consultation la plus chère au CHU de Poitiers ;
- 4) Prix de la consultation la moins chère au CHU de Poitiers ;
- 5) Noms des médecins qui ont effectué au moins 3 consultations au CHU de Poitiers ;
- 6) Numéro, nom et montant dépensé du patient qui a dépensé le plus en soins médicaux au CHU de Poitiers.

Exercice 03 : Les Banques

On considère le modèle relationnel suivant :

- **Banque** (NumBanque, Nom)
- **Agence** (NumAgence, Nom, CodePostal, NumBanque)
- **Client** (NumClient, Nom, Prénom, DateNaissance, LieuNaissance, CodePostal, Ville, TypeVoie, NumeroVoie, NomVoie, NumAgence)
- **Possede** (NumClient, NumCompte)
- **Compte** (NumCompte, Solde, Libelle)

La relation **Banque** permet de stocker l'ensemble des banques. Chaque banque est identifiée par un numéro et possède un nom (par exemple, "Crédit fonctionnaire du Poitou").

La relation **Agence** permet de stocker l'ensemble des agences d'une banque. Chaque agence est identifiée par un numéro et possède un nom, un code postal et un numéro de banque permettant d'identifier la banque à laquelle l'agence est rattachée.

La relation **Client** permet de stocker l'ensemble des clients d'une agence. Un client est identifié par un numéro et possède un nom, un prénom, une date de naissance, un lieu de naissance, un code postal, une ville, un type de voie (exemple : "rue", "impasse", ...), un numéro de voie, un nom de voie et un numéro d'agence à laquelle est rattaché le client.

La relation **Possede** permet de stocker l'ensemble des numéros de compte pour chaque client (un client possède entre 1 ou plusieurs comptes).

La relation **Compte** permet de stocker les caractéristiques de chaque compte. Un compte est identifié par un numéro et possède un solde et un libellé (exemple : "Compte Courant", "Livret Jeune", ...).

Avec le langage SQL, écrivez les requêtes permettant de connaître :

- 1) La liste des clients (numéro, nom et prénom) triée par ordre alphabétique, habitant à Poitiers ;
- 2) Pour chaque client (numéro, nom et prénom), le détail de ses comptes (numéro de compte et solde) ;
- 3) La liste des clients (numéro, nom et prénom) qui ont un compte joint avec le(s) client(s) se nommant Pierre Durand (Pierre étant le prénom, Durand étant le nom) triée par NumClient de façon croissante ;
- 4) Pour chaque client (numéro, nom et prénom) habitant dans la Vienne, la somme des soldes de l'ensemble de leurs comptes ;
- 5) Le numéro du compte courant (le libellé du compte est « Compte Courant ») qui dispose du solde le plus important ;
- 6) La liste des clients (numéro, nom et prénom) qui ont au moins deux comptes.