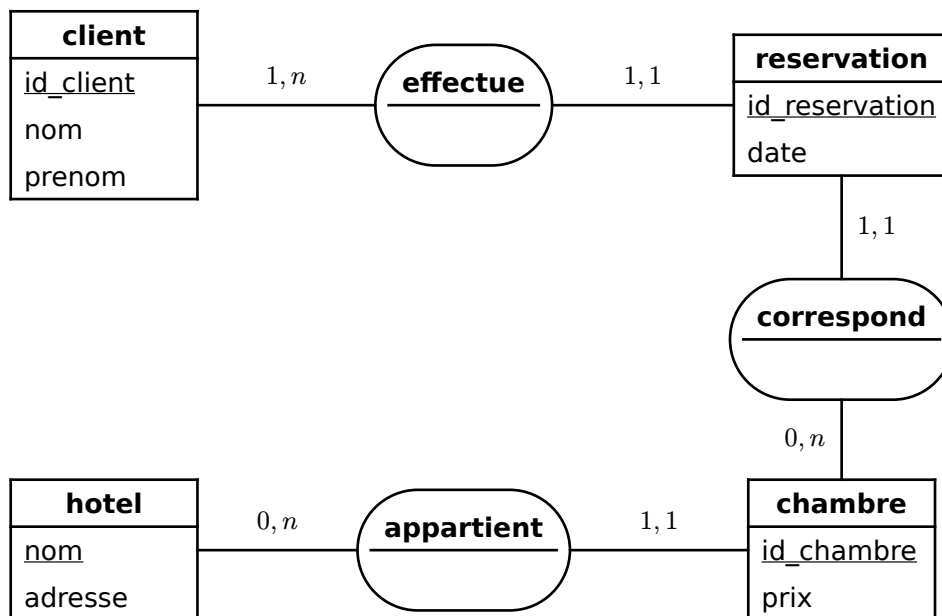


TNSI - Conception BDD - Exercices

Exercice 1

On souhaite gérer des réservations dans une compagnie d'hôtels. On considère donc le modèle Entités/Associations suivant :

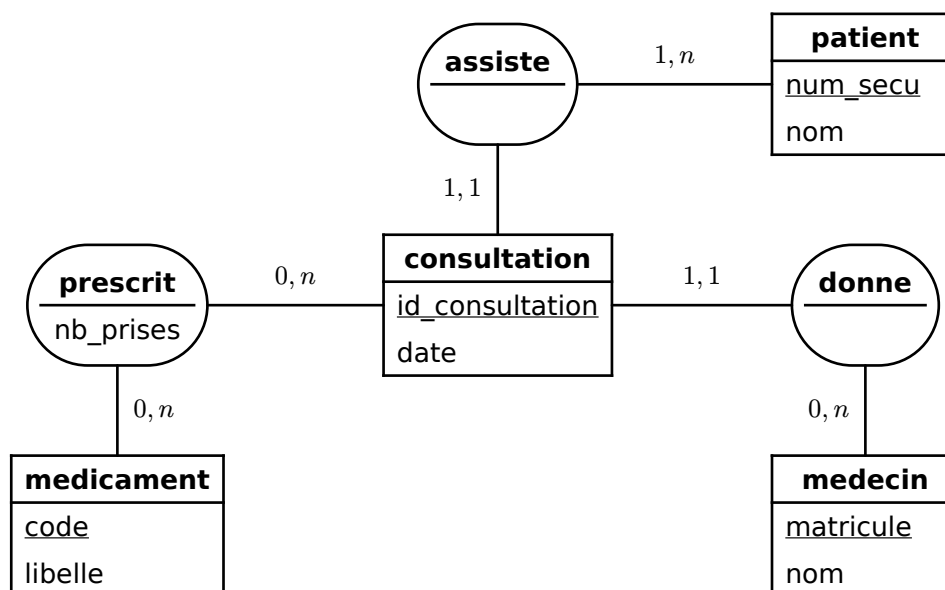


À l'aide de ce modèle, répondre aux questions suivantes :

1. Peut-on avoir des clients homonymes ?
2. Un client peut-il réserver plusieurs chambres à une date donnée ?
3. Est-il possible de réserver une chambre sur plusieurs jours ?
4. Peut-on savoir si une chambre est libre à une date donnée ?
5. Peut-on réserver plusieurs fois une chambre à une date donnée ?

Exercice 2

On donne le modèle Entités/associations représentant des visites dans un centre médical.



En utilisant ce modèle, répondre aux questions suivantes :

1. Un patient peut-il effectuer plusieurs visites ?
2. Un médecin peut-il recevoir plusieurs patients dans la même consultation ?
3. Peut-on prescrire plusieurs médicaments dans une même consultation ?
4. Deux médecins différents peuvent-ils prescrire le même médicament ?

Compétence : *Savoir passer du modèle Entités/Associations au modèle relationnel.*

Exercice 3

Donner le schéma relationnel de la base de donnée “compagnie d’Hôtels” décrite par le modèle Entités/Associations dans l’exercice 1.

Exercice 4

Donner le schéma relationnel de la base de données “visites médicales” décrite par le modèle Entités/Associations dans l’exercice 2.

Compétence : *Identifier le type des données ainsi que les contraintes d’intégrité dans un modèle relationnel.*

Exercice 5

À partir du modèle relationnel construit dans l’exercice 3, remplir le tableau ci-dessous :

Relation	Attribut	Type	Unicité	Valeur nulle permise	Clé
chambre	nom_hotel				
chambre	prix				
reservation	date				
client	id_client				

- Pour la colonne **Type**, on choisira parmi : *Entier, Réel, Texte, Date*.
- Pour la colonne **Unicité** et Valeur nulle permise, on répondra par *Oui* ou *Non*.
- Pour la colonne **Clé**, on mettra *PK* pour la clé primaire et *FK* pour la clé étrangère ou on laissera vide.

Compétence : *Identifier les anomalies parmi les occurrences d’une relation donnée.*

Exercice 6

On donne ci-dessous les occurrences de la relation Consultation issue du modèle relationnel construit dans l’exercice 4. Citer les anomalies constatées :

id_consultation	matricule	num_secu	date
1	123		21/11/2018
2	123	182086926825812	
2	526	'Aspirine'	13/09/2020

Compétence : Convertir une relation du modèle relationnel au modèle physique

Exercice 7

Supposons que la base de données de l'exercice 4 existe dans le SGBD SQLite.

On a déjà écrit les requêtes suivantes pour la création des relations Patient, Médecin et Médicament.

Donner l'écriture avec le langage SQL permettant la création des relations *consultation* et *prescrit*.

```
CREATE TABLE patient (
    num_secu INT,
    nom TEXT,
    PRIMARY KEY(num_secu)
)

CREATE TABLE medecin (
    matricule INT,
    nom TEXT,
    PRIMARY KEY (matricule)
)

CREATE TABLE medicament (
    code INT,
    libelle TEXT,
    PRIMARY KEY (code)
)
```

Exercice 8. Bilan

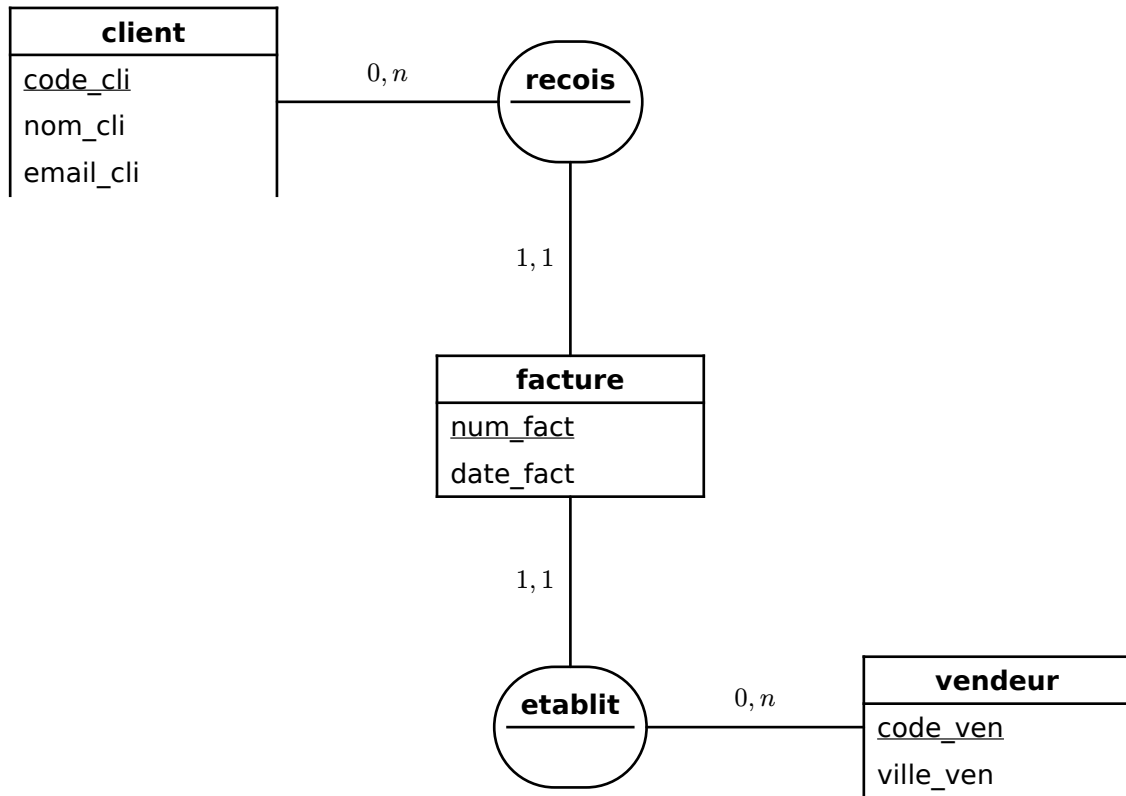
Pendant la phase de conception, toutes les données recueillies et spécifiées sont inscrites dans ce que l'on appelle un dictionnaire des données. On dispose du dictionnaire de données suivant :

Code de la donnée	Description	Type
code_ven	Identifiant du vendeur	Entier
nom_ven	Nom du vendeur	Texte
ville_ven	Ville où travaille le vendeur	Texte
code_cli	Identifiant du client	Entier
nom_cli	Nom du client	Entier
rue_cli	Rue où habite le client	Texte
cp_cli	Code postal du client	Entier
ville_cli	Ville où habite le client	Texte
dnaiss_cli	Date de naissance du client	Date
email_cli	Adresse mail du client	Texte
num_fact	Identifiant de la facture	Entier
date_fact	Date de facturation	Date
num_prod	Identifiant du produit	Entier
des_prod	Désignation du produit	Texte
prix_prod	Prix du produit	Réel
quantite	Quantité commandée	Entier

1. À partir de l'analyse du dictionnaire de données précédent :

- Identifier les différentes entités en jeu ainsi que leurs identifiants
- Les 3 associations mises en jeu par ces entités seront nommées « **établit** » « **recoit** » et « **ajoute** ». Pour chacune d'entre elle, spécifier leur cardinalité et éventuellement leurs attributs.

2. Compléter les données manquantes au modèle conceptuel Entités/Associations suivant :



3. En déduire le modèle relationnel
4. Donner le code pour créer la relation Facture dans le SQBD SQLite en considérant que la date de facturation ne peut pas être nulle.