

Démonstration 4

IFT6800

2 octobre 2015

1 Librairie

Donnez un diagramme d'entité relation pour une base de données gérant une librairie.

Votre système doit conserver certaines données sur les livres en vente, soit l'ISBN, le titre, l'auteur, l'éditeur, le genre (science-fiction, policier, cuisine, classique, etc.).

Votre système doit aussi inclure les comptes clients. Chaque client a un nom et une adresse. Il peut aussi avoir un code spécial qui lui donne un pourcentage de rabais sur les prochaines transactions.

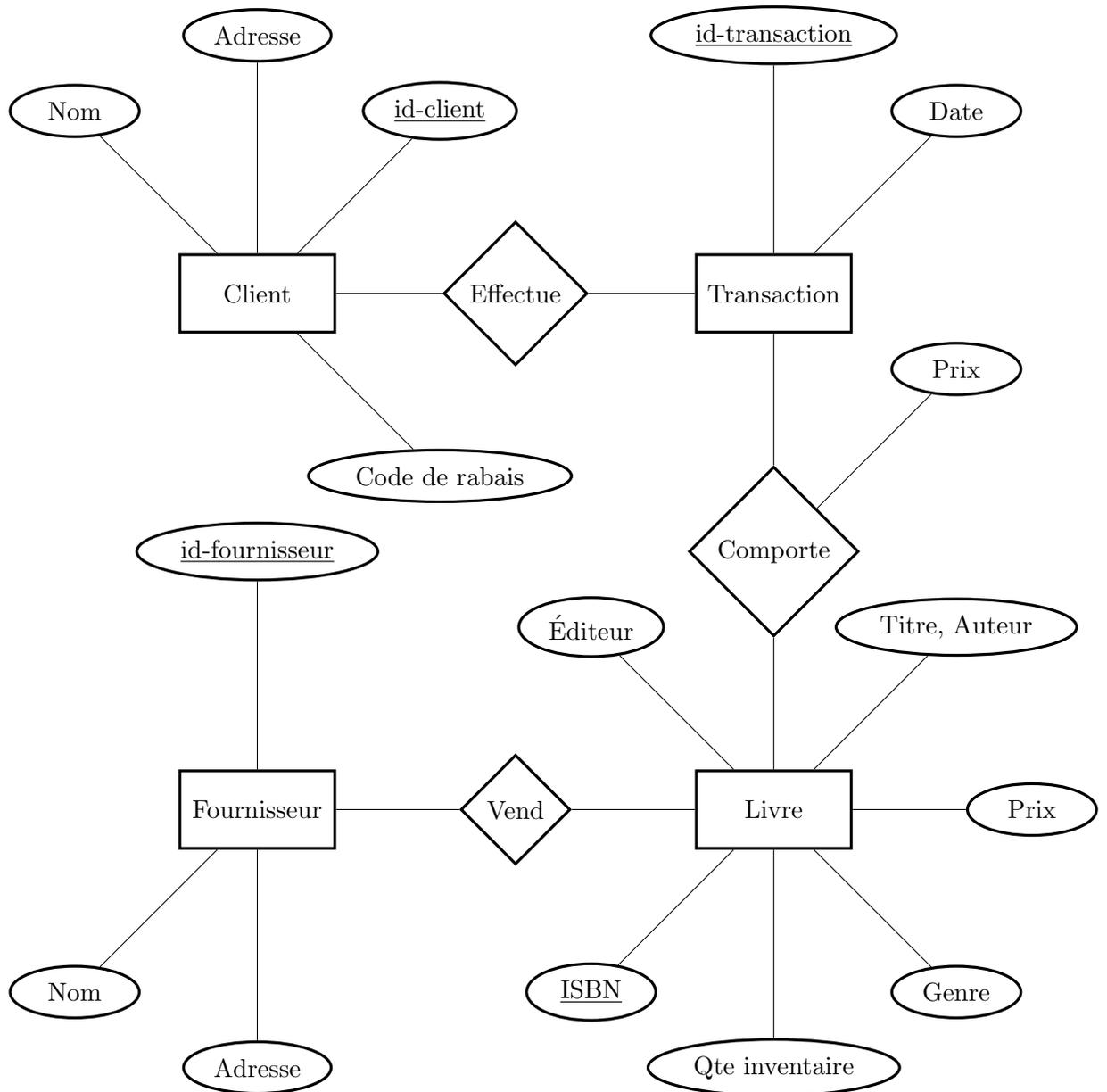
Les clients doivent pouvoir rechercher dans la base de donnée la liste des livres selon l'auteur, le titre, le genre ou l'ISBN. La base de donnée ne doit leur afficher que les livres dont des exemplaires sont dans l'inventaire. (On ne doit pas vendre un livre qu'on n'a plus!)

Le système doit permettre d'obtenir la date d'un achat, l'identité du client qui a effectué l'achat et la liste des livres achetés par celui-ci.

Votre système doit enregistrer pour chaque livre, la liste des fournisseurs qui peuvent le vendre à la librairie. Chaque fournisseur a un nom et une adresse et un catalogue de livres.

Identifiez les clés et les contraintes de domaines.

Exemple de diagramme répondant aux besoins :



Votre modèle permet-il les scénarios suivants :

Le client souhaite obtenir des suggestions selon les achats que d'autres clients ayant acheté un certain livre ont faits.

Oui, lors de la recherche d'un livre d'*ISBN* x , il suffit de faire une requête :

liste des *Livre* dont l'*ISBN* apparait dans un *Comporte* relié à une *Transaction* en relation *Effectue* avec un *Client* relié par *Effectue* avec une *Transaction* relié par *Comporte* avec le *Livre* d'*ISBN* x .

Le gérant souhaite obtenir la liste des 10 bestsellers de la dernière semaine.
Oui, il suffit qu'il fasse la requête : liste des *ISBN* des *Livre* en relation *Comporte* avec une *Transaction* ayant l'attribut *Date* comme il le désire. Ensuite, il doit compter le nombre d'occurrences de chaque *ISBN* dans cette liste et conserver les 10 plus grands.

Le gérant souhaite modifier le prix d'un livre sans perdre les données des transactions antérieures.

Oui. Le gérant peut modifier l'attribut *Prix* d'un *Livre* sans affecter le prix dans les transactions antérieures

Le gérant souhaite donner un 10% de rabais automatiquement à tous les clients qui ont acheté pour plus de 100\$ en 2015

Oui. Le gérant doit lister les *Client* et pour chacun de ceux-ci, effectuer la somme de *Prix* de la relation *Comporte* liée à une *Transaction* d'attribut *Date* en 2015 liée par *Effectue* à ce *Client*. Si la somme est supérieure à 100, alors le gérant modifie l'attribut *Code de rabais* de ce *Client*.

Un vendeur et un client souhaitent négocier un prix.

Oui, le vendeur peut modifier l'attribut *Prix* de la relation *Comporte* pour le *Livre* acheté dans la nouvelle *Transaction*. Ceci ne touche pas le prix des autres exemplaires de ce livre, qui lui est l'attribut *Prix* de *Livre*.

Lorsque tous les exemplaires d'un livre ont été vendus, le gérant souhaite recevoir un modèle de bon de commande pour chaque fournisseur proposant ce livre.

Oui, le gérant doit faire une requête qui liste les *Fournisseur* reliés par *Vend* avec un *Livre* ayant l'attribut *Qte inventaire* à 0.

Imprimer une étiquette postale et une facture pour une commande. (Une commande peut comporter plusieurs livres).

Oui, le vendeur peut aller chercher l'attribut *Adresse* du *Client* en relation *Effectue* avec la *Transaction* en cours.

Pour imprimer une facture, comportant le titre, le prix et le total de la commande, le vendeur fait une première requête de l'attribut *Titre* de tous les

Livre en relation *Comporte* avec la *Transaction* en cours. Il fait la somme du *Prix* de chacune des relations *Comporte*.

Un livre ayant plusieurs auteurs.

Non, il faudrait lier *Livre* avec une nouvelle relation *Écrit* vers une entité *Auteur*. Ceci permettrait d'atomiser les auteurs.

Le gérant souhaite obtenir une liste de suggestions de livres à acheter selon les recherches infructueuses de la clientèle.

Non, il faudrait ajouter dans *Livre*, par exemple, un attribut *Nb recherche* qu'on incrémente à chaque fois qu'un client recherche ce *Livre*. Ensuite, le gérant pourrait lister les *Livre* ayant cet attribut très gros.

Le libraire souhaite permettre à ses clients de chercher par étiquette ("tag") et d'ajouter des étiquettes sur des livres. Vous devez permettre une liste arbitraire de tags pour chaque livre. Un tag peut être "gagnant du prix Goncourt 1971", "personnage féminin fort", "pas de scène violente".

Non, il faudrait ajouter une relation *Décrit* à *Livre* vers l'entité *Tag* ayant comme attribut, en plus d'une clé, une *Description*.