



**ESTINATION FORMATION**

*Un aller simple vers le savoir-faire*

# **SUPPORT DE COURS**

## **INITIATION A L'ANALYSE ET A LA CONCEPTION DE BASE DE DONNEES**

## AVANT PROPOS

Ce support de cours est un outil personnel, il ne constitue pas un guide de référence.

C'est un outil pédagogique élaboré dans un souci de concision : il décrit les actions essentielles à connaître pour appréhender le sujet de la formation.

### Après réalisation d'un exercice :

- Notez votre démarche, vous aurez ainsi une trace des remarques formulées "à chaud".
- Renforcez vos connaissances en demandant des précisions au formateur.

### Charte graphique :

Certaines images sont utilisées comme repère de lecture, notamment :



Remarque importante

*Exemple*

*Contient des exemples*

# SOMMAIRE

## ANALYSE ET CONCEPTION DE BASE DE DONNEES

---

<b>I. INTRODUCTION</b>	<b>1</b>
<i>A. LES LIMITES DU TABLEUR EXCEL</i>	2
<i>B. LA METHODE MERISE</i>	3
<b>II. RECUEIL ET ORGANISATION DES INFORMATIONS</b>	<b>4</b>
<i>A. CREER LE DICTIONNAIRE DE DONNEES</i>	5
<i>B. PLACER LES INFORMATIONS DANS UNE MATRICE</i>	6
<b>III. LES SCHEMAS CONCEPTUELS</b>	<b>9</b>
<i>A. LE MODELE CONCEPTUEL DE DONNEES</i>	10
<i>B. LE MODELE LOGIQUE DE DONNEES</i>	13
<i>C. LES REQUETES</i>	14
<b>IV. ANNEXE : EXERCICE D'APPLICATION.</b>	<b>15</b>




# ***I. Introduction***

## A. LES LIMITES DU TABLEUR EXCEL

Excel est un tableur, qui est capable de gérer des bases de données simples. Le problème est que les limites d'Excel sont vite atteintes en ce qui concerne **la redondance de l'information**.

*Exemple*

La feuille de calcul ci-dessous représente une table de facture avec les clients.



	A	B	C	D	E	F
1	<u>Num Facture</u>	<u>Num client</u>	<u>Raison sociale</u>	<u>Adresse</u>	<u>Code postal</u>	<u>Ville</u>
2	1	CLI15	Promotrans	Boulevard Yves Farge	69007	Lyon
3	2	CLI26	Micro's pas cher	Quai Fulchiron	69005	Lyon
4	3	CLI67	Le Tanneur	Rue Saint-Just	69100	Villeurbanne
5	4	CLI15	Promotrans	Boulevard Yves Farge	69007	Lyon
6	5	CLI15	Promotrans	Boulevard Yves Farge	69007	Lyon

Il serait plus logique d'avoir deux tableaux avec un lien entre les deux tables.

<u>Num Facture</u>	<u>Num client</u>
1	CLI15
2	CLI26
3	CLI67
4	CLI15
5	CLI15

LIEN

<u>Num client</u>	<u>Raison sociale</u>	<u>Adresse</u>	<u>Code postal</u>	<u>Ville</u>
CLI15	Promotrans	Boulevard Yves Farge	69007	Lyon
CLI26	Micro's pas cher	Quai Fulchiron	69005	Lyon
CLI67	Le Tanneur	Rue Saint-Just	69100	Villeurbanne
CLI15	Promotrans	Boulevard Yves Farge	69007	Lyon
CLI15	Promotrans	Boulevard Yves Farge	69007	Lyon

Il reste à savoir où mettre les champs, c'est à dire quel champ doit être placé dans quelle table ?

Pour ce faire, il existe une méthode, nommée MERISE (Méthode d'Etude de Réalisation Informatique par Sous-Ensemble). Cette méthode permet de conceptualiser une base de données.



## **B. LA METHODE MERISE**

Pour MERISE, il est classique de mettre en évidence trois niveaux de réflexion :

\***Le niveau conceptuel** correspond à la définition des finalités de l'entreprise en expliquant sa raison d'être. Ce niveau traduit les objectifs et les contraintes qui pèsent sur l'entreprise. L'informatique doit les intégrer sans remise en cause. Ces finalités constituent généralement le niveau le plus stable. On y trouvera par exemple les règles de gestion du personnel, de tenue de la comptabilité ou de livraison des produits finis.

\***Le niveau organisationnel** permet de définir l'organisation qu'il est souhaitable de mettre en place dans l'entreprise pour atteindre les objectifs visés. On parle alors de choix d'organisation, pour lesquels la marge de manœuvre est plus importante. Ce niveau précise les postes de travail, la chronologie des opérations, les choix d'automatisation, tout en intégrant les contraintes éventuelles.

\***Le niveau technique** intègre les moyens techniques nécessaires au projet. Ils s'expriment en termes de matériels ou de logiciels et sont (par suite des progrès technologiques) les plus sujets à changement.

Si MERISE a la vocation de concevoir un système d'information aisé à maintenir et bien intégré à l'entreprise, il est aussi un gage de sécurité et un atout précieux pour un analyste, auquel il propose une organisation du travail. La séparation des données et des traitements, jointe à la définition des niveaux permet d'aborder successivement les problèmes et de se situer à tout moment dans l'avancement des travaux. Le parcours des différents niveaux est alors le suivant :

- 1- Le dictionnaire des données.
- 2- La matrice des dépendances fonctionnelles.
- 3- Le Modèle Conceptuel de Données (MCD).
- 4- Le Modèle Logique de Données (MLD).
- 5- Le passage sur un logiciel de SGBDR (Système de Gestion de Base de Données Relationnelles), par exemple Access.



## ***II. Recueil et organisation des informations***

## A. CREER LE DICTIONNAIRE DE DONNEES

Après un travail rigoureux de recueil de l'information sur le terrain auprès des utilisateurs du système d'informations (il s'agit du recueil de l'existant) et l'intégration des nouvelles demandes, il faut regarder l'ensemble des données utiles à sauvegarder. Cet ensemble forme le dictionnaire de données.

Ce dictionnaire est présenté sous forme de tableaux :

Champs retenus	Type de données	Commentaires
		...

Exemple : la notation du groupe BTS 1 chez Promotrans

Les données retenues peuvent être :

- le nom, le prénom, l'adresse et le téléphone des étudiants
- le libellé de la matière
- les notes de chaque contrôle
- les moyennes des étudiants par matière
- la moyenne générale
- la date des contrôles

Ce sont des attributs ou des rubriques


Tous ces attributs vont être placés dans un modèle de données. Il existe des champs de type calculé, tels les moyennes, qui ne figureront pas dans le modèle de données.

Les types de données sont :

- \*Alphanumérique
- \*Numérique
- \*Date/heure
- \*Booléen (oui/non)

Une fois le dictionnaire de données mis en place, il faut créer un dictionnaire simplifié, duquel les champs calculés sont ôtés. Il faut aussi enlever tous les champs redondants (un « code article » et une « référence article » par exemple, qui qualifient la même information). Enfin, il peut exister des homonymes. Or, chaque enregistrement doit pouvoir être identifier comme unique de façon certaine. Pour ce faire, il faut créer **un identifiant**.

Dans l'exemple des étudiants, on ajoute donc un numéro d'étudiant et un code matière. **Un identifiant permet de retrouver un et un seul enregistrement**. L'identifiant possède généralement une forme courte. Il peut être créé artificiellement dans le cas où il n'existerait pas dans le dictionnaire de données. L'exemple type d'identifiant reste le numéro de sécurité sociale.

 **Le dictionnaire de données simplifié regroupe donc tous les champs non redondants à conserver dans le système d'informations. Chaque champ possède un type de données. Ce tableau permet de créer ensuite la matrice des dépendances fonctionnelles.**

Pour s'entraîner, créer le dictionnaire de données de l'exercice placé en annexe.





## **B. PLACER LES INFORMATIONS DANS UNE MATRICE**

A partir de maintenant, tous les chapitres seront illustrés par l'exercice placé en annexe.

Avant de mettre en place la matrice, il faut déterminer **les règles de gestion**. Ces règles sont identifiées suite aux interviews menées auprès des utilisateurs. Elles sont en général données dans les énoncés de vos exercices. Dans le cas contraire, il faut les isoler.

Dans notre exercice, les règles de gestion sont les suivantes :

- A- « *La société ne veut avoir affaire qu'à un et un seul contact* ».
- B- « *Chaque catégorie référence logiquement plusieurs produits* ».
- C- « *Chaque commande peut contenir un ou plusieurs produits* ».
- D- « *Le prix de vente peut différer du prix unitaire* ».

La matrice des dépendances fonctionnelles est un simple tableau. La première colonne reprend tous les attributs du dictionnaire simplifié. Une deuxième colonne numérote chaque champ de un à l'infini.

<u>Num Client</u>	1					
Raison sociale Client	2					
Nom contact	3					
Prénom contact	4					
Fonction contact	5					
Adresse Client	6					
Code postal Client	7					
Ville Client	8					
Fax Client	9					
<u>Référence produit</u>	10					
Nom produit	11					
Prix unitaire du produit	12					
Prix de vente du produit	13					
<u>Code catégorie</u>	14					
Nom catégorie	15					
Description catégorie	16					
<u>Num Commande</u>	17					
Date commande	18					

Dans la liste des attributs, les identifiants sont soulignés. En face de chaque identifiant, il faut placer une étoile dans la colonne correspondant à l'identifiant.

A partir de chaque identifiant, il faut alors se poser la question « *si je prends un identifiant, est ce qu'un attribut correspond une et une seule fois à cet identifiant ?* ». Ainsi, si je prends le numéro de client, est-ce que j'ai une et une seule raison sociale qui correspond ?



Si la réponse est « oui », on place un 1 dans la colonne portant le numéro de l'identifiant. Cela nous donne la matrice suivante :

		1	10	14	17	10+17
<u>Num Client</u>	1	*			1	
Raison sociale Client	2	1				
Nom contact	3	1				
Prénom contact	4	1				
Fonction contact	5	1				
Adresse Client	6	1				
Code postal Client	7	1				
Ville Client	8	1				
Fax Client	9	1				
<u>Référence produit</u>	10		*			*
Nom produit	11		1			
Prix unitaire du produit	12		1			
Prix de vente du produit	13					1
<u>Code catégorie</u>	14		1	*		
Nom catégorie	15			1		
Description catégorie	16			1		
<u>Num Commande</u>	17				*	*
Date commande	18				1	

Lors de cette recherche des attributs correspondants à un identifiant, il faut faire attention de toujours repartir de la ligne 1. L'attention doit aussi porter sur les règles de gestion. Ici, ces règles donnent un seul contact par client. S'il y en avait eu plusieurs, il aurait été impossible de mettre des 1 dans la colonne 1 face aux nom, prénom et fonction du contact.

Il arrive que certains attributs ne répondent pas à la question posée. Ainsi, si je prends le prix de vente du produit, il n'est pas le même pour un même produit. Cet attribut possède donc plus d'un identifiant. Ici, l'identifiant du prix de vente est à la fois la « référence produit » et le « numéro de commande ». En effet, le prix de vente de l'article 1 peut être de 14 francs dans la commande 12, mais de 16 francs dans la commande 5. Il faut bien avoir le numéro de la commande pour retrouver le prix de vente.

La matrice des dépendances fonctionnelles de cet exercice est donc :

		1	10	14	17	10+17
<u>Num Client</u>	1	*			1	
Raison sociale Client	2	1				
Nom contact	3	1				
Prénom contact	4	1				
Fonction contact	5	1				
Adresse Client	6	1				
Code postal Client	7	1				
Ville Client	8	1				
Fax Client	9	1				
<u>Référence produit</u>	10		*			*
Nom produit	11		1			
Prix unitaire du produit	12		1			
Prix de vente du produit	13					1
<u>Code catégorie</u>	14		1	*		
Nom catégorie	15			1		
Description catégorie	16			1		
<u>Num Commande</u>	17				*	*
Date commande	18				1	



Il existe trois relations. Les relations 1 et 2 sont simples. La troisième est complexe, puisqu'elle forme une colonne à part entière de la matrice. Elles lient respectivement:

- 1- La table Client à la table Commande
- 2- La table Produit à la table Catégorie
- 3- La table Produit à la table Commande

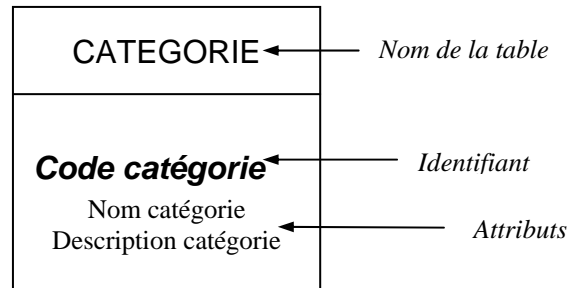
Une fois que la matrice est déterminée avec tous les attributs affectés à des identifiants, la partie la plus dure est réalisée. Le passage au MCD, puis au MLD reste simplement une question de formalisme. La question des relations se posera dans cette partie.



## ***III. Les schémas conceptuels***

### A. LE MODELE CONCEPTUEL DE DONNEES

Pour créer ce modèle, il faut transformer chaque colonne en table. Une table comporte un nom, un ou plusieurs identifiant(s) et des attributs. Elle se présente sous cette forme :

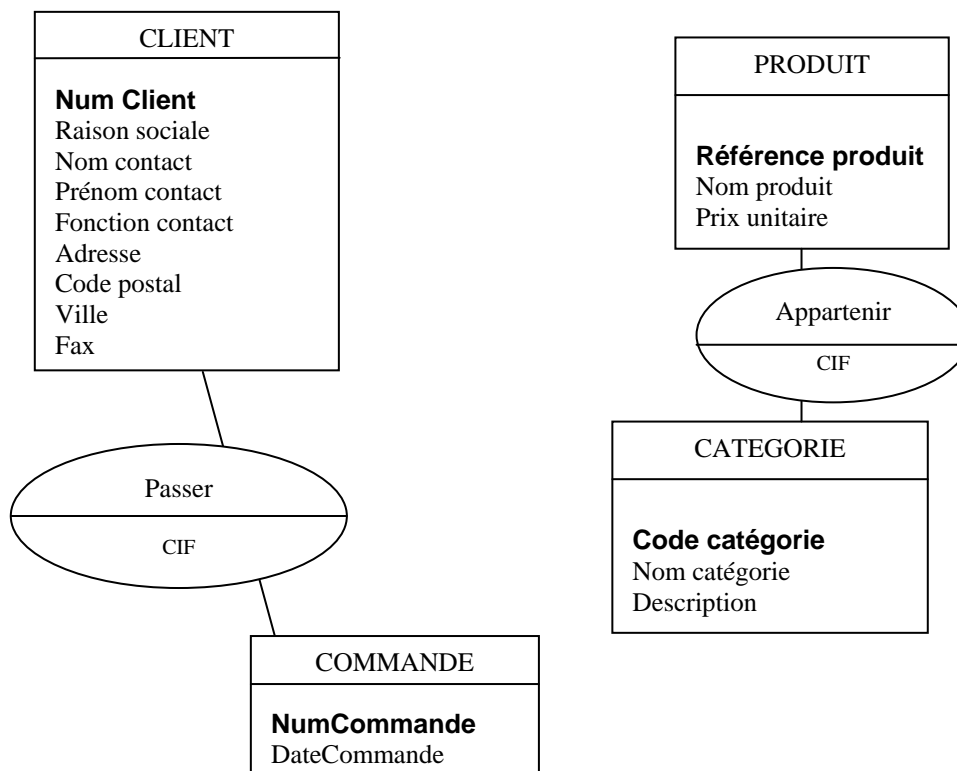


L'identifiant est maintenant nommé « clé primaire ». Une clé primaire est constituée d'un ou plusieurs attributs. Elle permet de repérer un et un seul enregistrement. La clé est toujours soulignée.

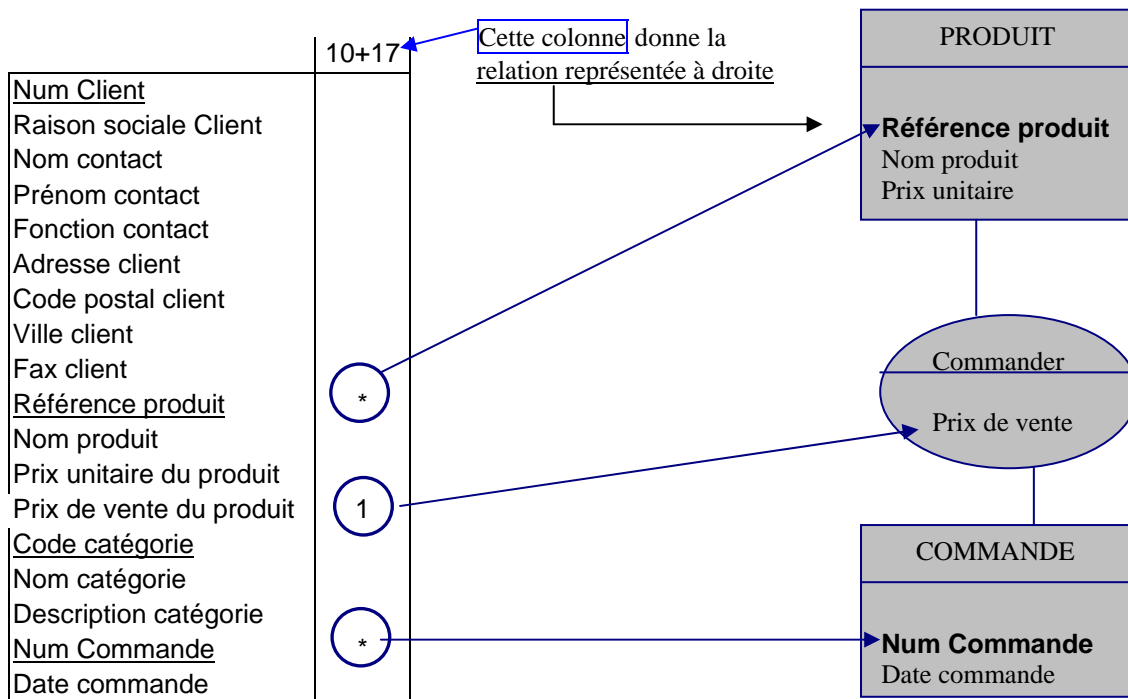
Pour déterminer les associations entre les différentes tables, il suffit de prendre les lignes qui possèdent à la fois \* et 1. Une association est dite binaire si elle relie deux tables, n-aire si elle lie plus de deux tables. En règle générale, il existe peu de cas où une association concerne plus de quatre tables à la fois.

Lorsque la ligne possède à la fois 1 et \*, il s'agit d'une CIF (Contrainte d'Intégrité Fonctionnelle). La CIF est une relation simple, à laquelle on donne un nom (en général, un verbe). Dans notre cas, il existe deux CIF entre :

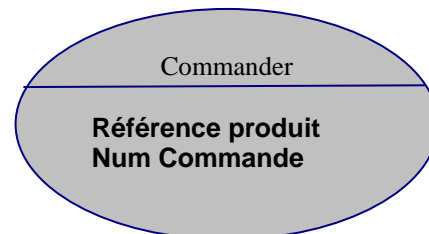
- La table « CATEGORIE » et la table « PRODUIT »
- La table « COMMANDE » et la table « CLIENT »



Lorsque la matrice des dépendances fonctionnelles possède des colonnes avec deux numéros, la relation est plus complexe, car elle contient des attributs.



N.B : la relation « commander » possède une clé primaire constituée de deux champs : la référence produit et le num commande. On peut faire apparaître ces champs dans la relation, ce qui donnerait ceci :

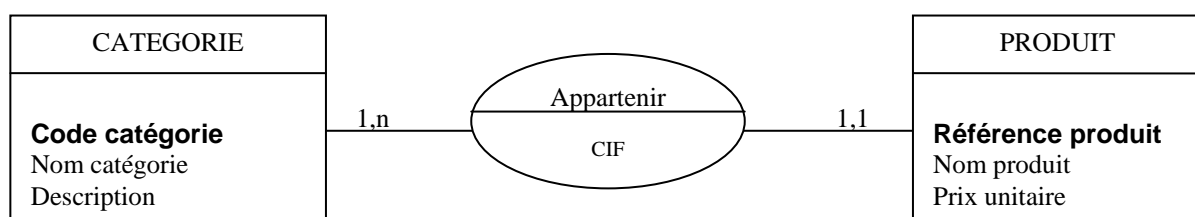


Quelles sont les possibilités des associations ? La nature des associations est déterminée par les cardinalités. Une cardinalité détermine le nombre minimum et maximum d'occurrences qu'il peut exister. Elle est déterminée par la logique ou par les règles de gestion.

Voici tout les types de cardinalités possibles :

0,1                      1,1                      0,n                      1,n

Exemple : « Chaque catégorie référence plusieurs produits ».



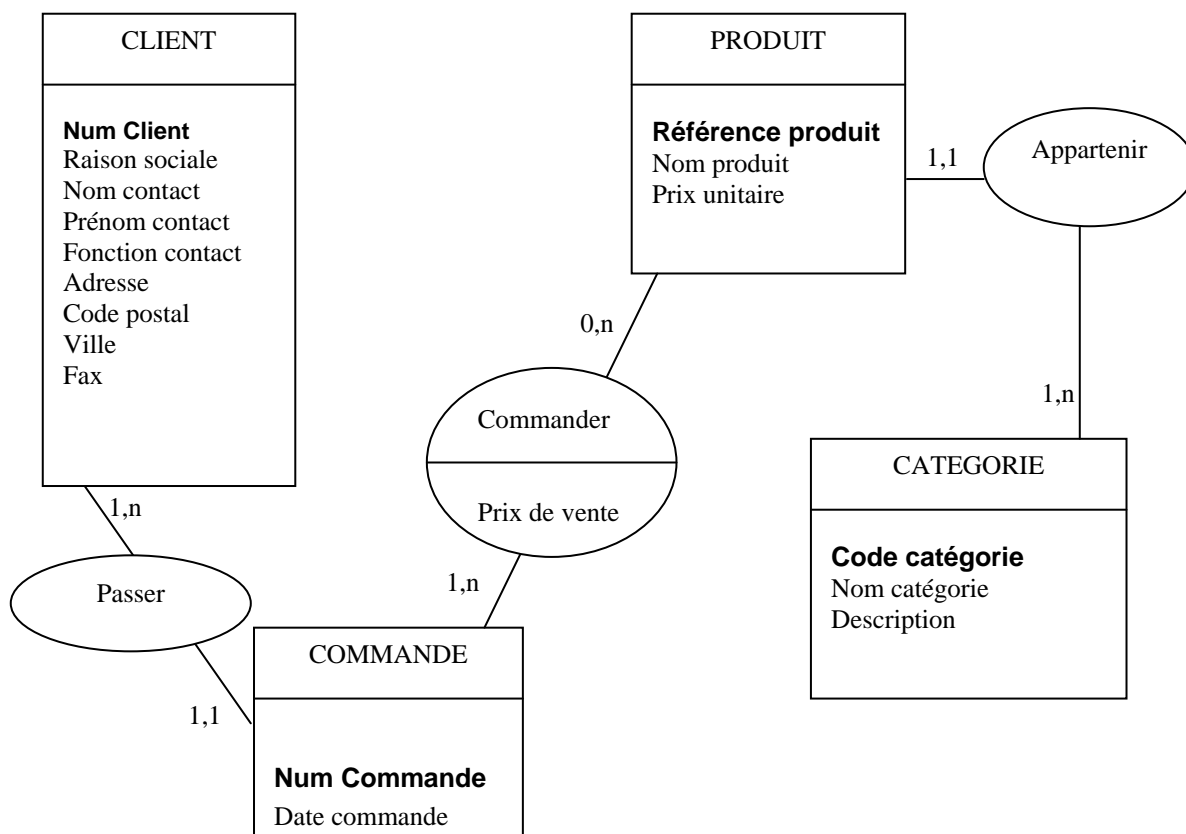


**La cardinalité 1,n signifie qu'une catégorie peut regrouper plusieurs produits, alors que la cardinalité 1,1 veut dire qu'un produit appartient à une seule catégorie.**

Les CIF ont toujours des cardinalités 1,1 et 1,n (on dit « *un à plusieurs* »), tandis que les associations compliquées ont des cardinalités 1,n à 1,n (« *plusieurs à plusieurs* »).

Le zéro existe lorsque des attributs de la table sont stockés mais pas forcément utilisés (par exemple, on aurait mis 0,1 si des produits existaient dans la table sans jamais avoir été catégorisés).

Le MCD complet de notre exercice donne ceci :



## B. LE MODELE LOGIQUE DE DONNEES

Le modèle logique constitue la dernière étape avant le passage sur Access. Il permet de déterminer les clés étrangères et les requêtes. Il consiste donc en deux étapes : supprimer les relations complexes pour les transformer en table à part entière d'une part et d'autre part trouver les clés étrangères.

La clé étrangère est formée d'un ou plusieurs attributs permettant de relier la clé primaire d'une table. Cette clé est toujours placée du côté 1,1 de l'association. Elle est précédée du signe #.

Dans notre exercice, le modèle logique est le suivant :

CLIENT (Num Client, Raison sociale, Nom contact, Prénom contact, Fonction contact, Adresse, Code postal, Ville, Fax)

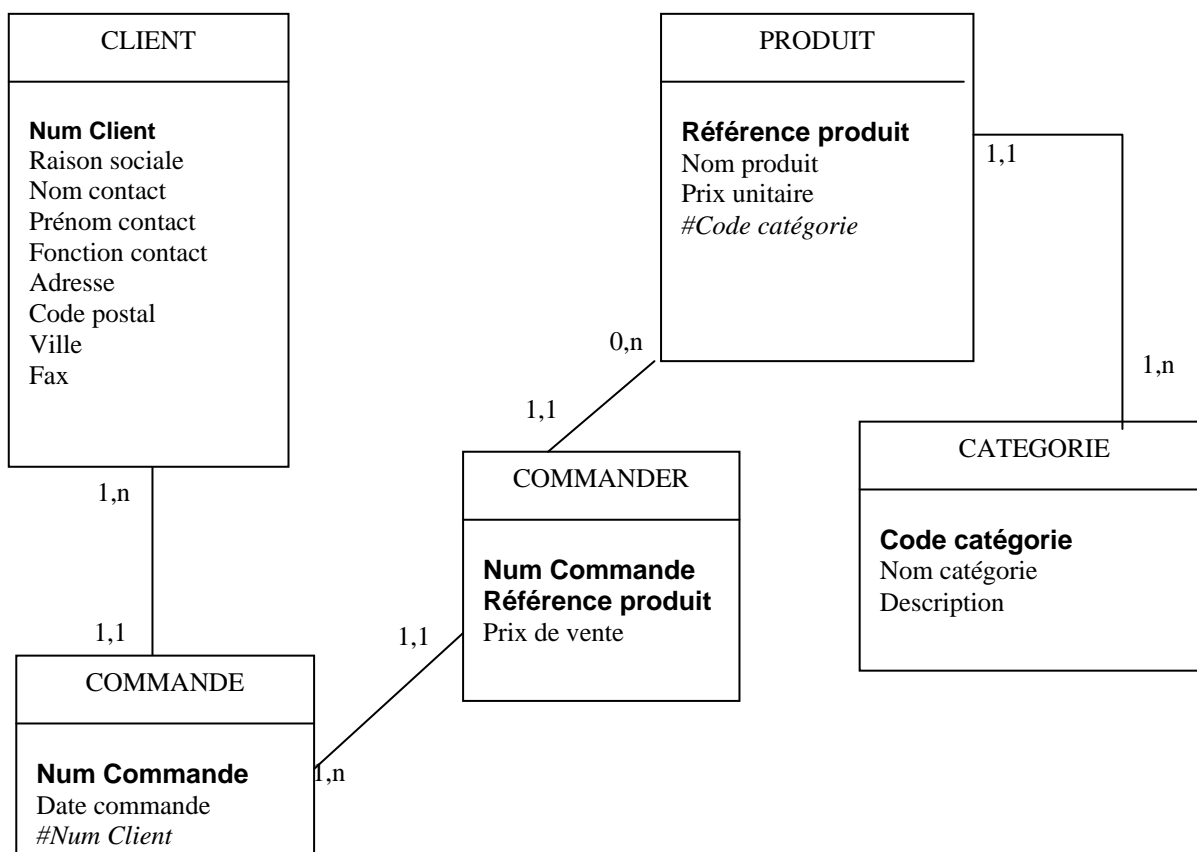
PRODUIT (Référence produit, Nom produit, Prix unitaire, #Code catégorie)

CATEGORIE (Code catégorie, Nom catégorie, Description)

COMMANDE (Num Commande, Date commande, #Num Client)

COMMANDER (Num Commande, Référence produit, Prix de vente)

Il est également possible de le représenter sous une forme proche du MCD :







### **C. LES REQUÊTES**

Avant de passer à la partie logicielle, il reste à déterminer les requêtes dont l'utilisateur a besoin pour extraire des informations des tables. Ces requêtes sont définies à l'aide des interviews et du recueil des documents manipulés au sein du système d'informations de l'entreprise.

Il existe trois types de requêtes :

- **Les requêtes sélection**, qui permettent de choisir des enregistrements selon certains critères (exemple des filtres Excel).
- **Les requêtes jointure**, qui relient deux tables, laissant ainsi la possibilité d'extraire des informations de plusieurs tables à la fois.
- **Les requêtes projection**, qui récupèrent certains champs sans passer de critères.

Pour utiliser les requêtes, il existe le langage SQL (Structured Query Language). Cependant, pour simplifier, nous utiliserons un langage « papier ». Les exemples suivants donnent un aperçu de l'écriture des requêtes :

---

R1 = PROJECTION CLIENT (Num Client, Raison sociale)

*Cette requête ressort tous les clients avec leur numéro.*

---

R2 = SELECTION CLIENT (Raison sociale = « Promotrans »)

R3 = PROJECTION R2 (Nom contact, Prénom contact, Fonction contact, Adresse, Code postal, Ville, Fax)

*Cette requête nous donne toutes les informations concernant Promotrans. On peut inverser R2 et R3.*

---

R4 = JOINTURE CLIENT, COMMANDE (Num Client.Client = Num Client.Commande)

R5 = JOINTURE COMMANDE, COMMANDER (NumCommande.Commande = NumCommande.Commander)

R6 = JOINTURE COMMANDER, PRODUIT (RefProduit.Produit = RefProduit.Commander)

R8 = SELECTION CLIENT (Raison sociale = « Promotrans »)

R9 = PROJECTION R8 (Raison sociale, Nom produit, Prix unitaire)

*Cette requête permet de savoir tout ce que Promotrans nous a commandé comme produit.*

---



## ***IV. ANNEXE : exercice d'application.***



Votre société veut créer une petite base de données de gestion des commandes pour contrôler ses stocks et ainsi pouvoir alimenter ses clients de manière ininterrompue, et ce qu'elle que soit la demande.

Chaque client est une société avec une raison sociale et un numéro de client interne à votre entreprise. Pour des raisons pratiques, vous possédez le nom d'un contact et sa fonction au sein de l'entreprise. Votre société, pour des raisons d'organisation, ne veut avoir à faire qu'à un et un seul contact. De plus, vous connaissez évidemment le numéro de téléphone, le fax et l'adresse de votre client.

Vous possédez une gamme de produits en catalogue, parmi lesquels les clients peuvent commander. Un produit possède une référence, un nom et un prix unitaire. Il entre dans une catégorie codée. Chaque catégorie a en plus d'un code, un nom et une description. Elle référence logiquement plusieurs produits.

Vous recevez ainsi des commandes d'un client, chacune d'entre elles pouvant contenir un ou plusieurs produits. Cette commande est numérotée et datée. Le ou les produits qui la composent sont vendus à un prix de vente qui peut différer du prix unitaire, puisque votre société décide de sa marge selon ses clients.

---

**A partir de ce texte, construire un dictionnaire de données, une table des dépendances fonctionnelles, un MCD, un MLD, puis construire les tables et les relations sous Access 97.**

---



Après utilisation de votre support, vous avez peut être des remarques ou des suggestions à formuler.

Nous vous remercions de bien vouloir nous en faire part en nous adressant ce coupon réponse, par courrier, à l'adresse suivante :

**DESTINATION FORMATION**  
*2, rue des Meilles*  
**38370 LES ROCHES DE CONDRIEU**  
rd.formation@wanadoo.fr

Vos propositions et critiques permettront d'améliorer ce support de cours.

Merci d'avance.



VOTRE SOCIETE : .....

Votre nom : ..... Prénom : .....

Intitulé Formation : .....

Date de la formation : .....

Vos remarques et suggestions :