

Algèbre relationnel

C

Dr C. THIAM

cthiam@univ-thies.sn

les limites à l'utilisation des fichiers

- L' utilisation de fichiers impose d'une part, à l'utilisateur de connaître l'organisation (séquentielle, indexée, ...) des fichiers qu'il utilise afin de pouvoir accéder aux informations dont il a besoin et, d'autre part, d'écrire des programmes pour pouvoir effectivement manipuler ces informations.
- Pour des applications nouvelles, l'utilisateur devra obligatoirement écrire de nouveaux programmes et il pourra être amené à créer de nouveaux fichiers qui contiendront peut-être des informations déjà présentes dans d'autres fichiers.
- De telles applications sont :
 - rigides, • contraignantes, • longues et coûteuses à mettre en œuvre.
- Les données associées sont :
 - mal définies et mal désignées, • redondantes, • peu accessibles de manière ponctuelle, • peu fiables. .

base de données.

- Définition (base de données) : Une base de données est un ensemble d'informations sur un sujet qui est :
 - exhaustif,
 - non redondant,
 - structuré,
 - persistant.
- Définition (système de gestion de base de données) : Un système de gestion de base de données est un logiciel qui permet de :
 - décrire,
 - modifier,
 - interroger,
 - administrer, les données d'une base de données

Le modèle relationnel

- Le modèle relationnel a été formalisé par CODD en 1970.
- Dans ce modèle, les données sont stockées dans des tables, sans préjuger de la façon dont les informations sont stockées dans la machine.
- Un ensemble de données sera donc modélisé par un ensemble de tables.
- Le succès du modèle relationnel auprès des chercheurs, concepteurs et utilisateurs est dû à la puissance et à la simplicité de ses concepts.
- En outre, contrairement à certains autres modèles, il repose sur des bases théoriques solides, notamment la théorie des ensembles et la logique mathématique(théorie des prédicats d'ordre 1).

Les objectifs du modèle relationnel

- proposer des schémas de données faciles à utiliser,
- améliorer l'indépendance logique et physique,
- mettre à la disposition des utilisateurs des langages de haut niveau pouvant éventuellement être utilisés par des non informaticiens,
- optimiser les accès à la base de données,
- améliorer l'intégrité et la confidentialité,
- fournir une approche méthodologique dans la construction des schémas

Définition formelle du modèle relationnel

- De façon informelle, on peut définir le modèle relationnel de la manière suivante :
 - Les données sont organisées sous forme de tables à deux dimensions, encore appelées relations et chaque ligne n-uplet ou tuple,
 - les données sont manipulées par des opérateurs de l'algèbre relationnelle,
 - l'état cohérent de la base est défini par un ensemble de contraintes d'intégrité.
 - Au modèle relationnel est associée la théorie de la normalisation des relations qui permet de se débarrasser des incohérences au moment de la conception d'une base de données.

Définitions

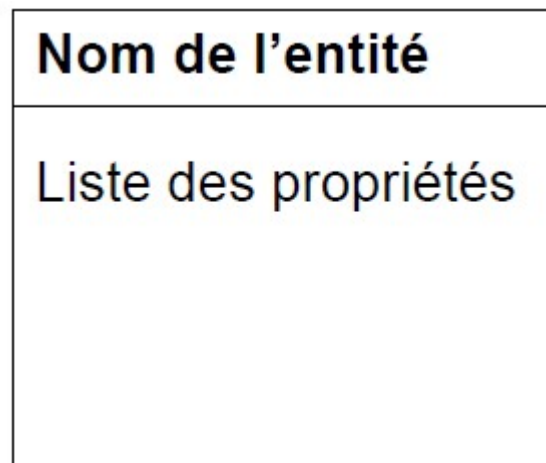
- Définition (Domaine) : Ensemble de valeurs.
- Définition (Relation) : Sous-ensemble du produit cartésien d'une liste de domaines caractérisé par un nom. En d'autres termes, une relation n'est ni plus ni moins qu'une table dans laquelle chaque colonne correspond à un domaine et porte un nom ce qui rend leur ordre sans aucune importance.
- Définition (Attribut) : Colonne d'une relation caractérisée par un nom.
- Définition (Schéma de relation) : Nom de la relation, suivi de la liste des attributs avec leurs domaines.
- Définition (Base de données relationnelles) : Base de données dont le schéma est un ensemble de schémas de relations et dont les occurrences sont les tuples de ces relations.
- Définition (Système de gestion de bases de données relationnel) : C'est un logiciel supportant le modèle relationnel, et qui peut manipuler les données avec des opérateurs relationnels.

Le modèle Entité-Association

- EA en français, ER en anglais (pour Entity Relationship)
- Formalisme retenu par l'ISO pour décrire l'aspect conceptuel des données à l'aide d'*entités* et d'*associations*

Le concept d'entité

- Représentation d'un objet matériel ou immatériel
- Par exemple un employé, un projet, un bulletin de paie



- Les entités peuvent être regroupées en **types d'entités**
- Par exemple, on peut considérer que tous les employés
- particuliers sont des **instances** du type d'entité générique EMPLOYE

Les propriétés

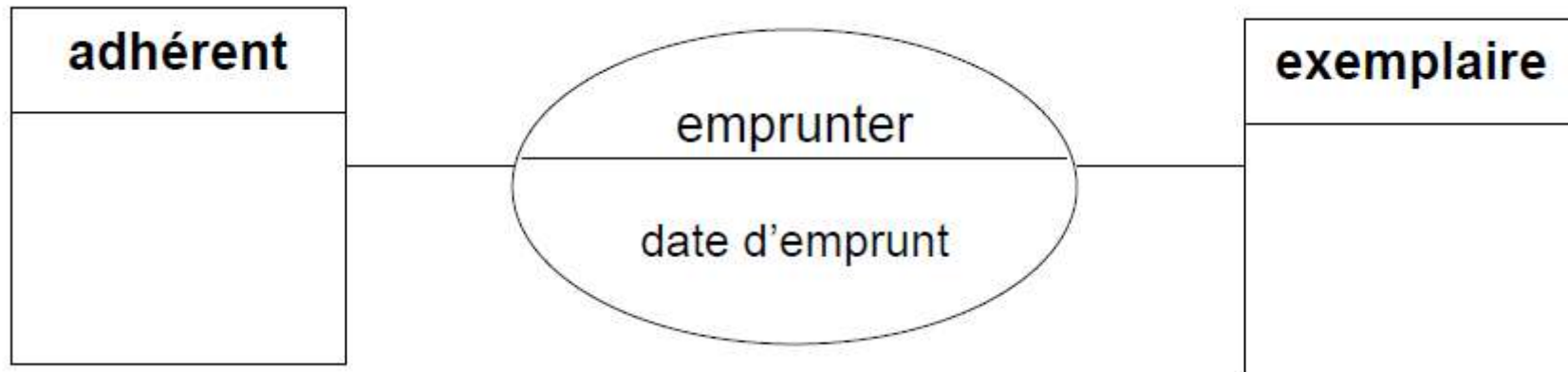
- données élémentaires relatives à une entité
- Par exemple, un numéro d'employé, une date de début de projet
 - on ne considère que les propriétés qui intéressent un contexte particulier
 - Les propriétés d'une entité sont également appelées des attributs, ou des caractéristiques de cette entité

L'identifiant

- propriété ou groupe de propriétés qui sert à identifier une entité
- L'identifiant d'une entité est choisi par l'analyste de façon à ce que deux occurrences de cette entité ne puissent pas avoir le même identifiant
- Par exemple, le numéro d'employé sera l'identifiant de l'entité EMPLOYE

Les associations

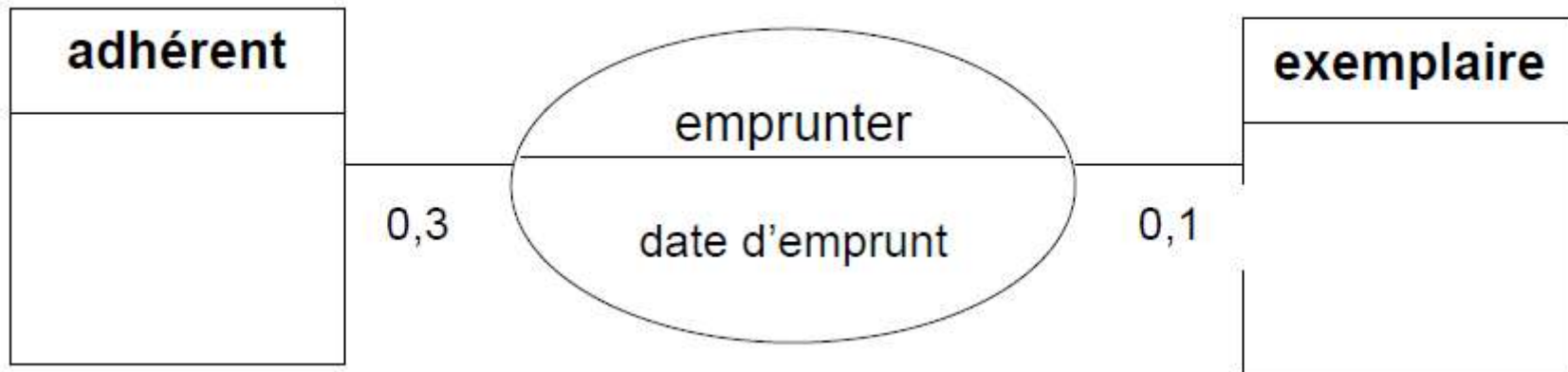
- Représentation d'un lien entre deux entités ou plus
 - une association peut avoir des propriétés particulières
 - Par exemple, la date d'emprunt d'un livre



Les cardinalités

- La cardinalité d'une association pour une entité constituante est constituée d'une borne minimale et d'une borne maximale :
 - Minimale : nombre minimum de fois qu'une occurrence de l'entité participe aux occurrences de l'association, généralement 0 ou 1
 - Maximale : nombre maximum de fois qu'une occurrence de l'entité participe aux occurrences de l'association, généralement 1 ou n

Les cardinalités



- La cardinalité 0,3 indique qu'un adhérent peut être associé à 0, 1, 2 ou 3 livres, c'est à dire qu'il peut emprunter au maximum 3 livres.
- A l'inverse un livre peut être emprunté par un seul adhérent, ou peut ne pas être emprunté.

Les cardinalités

- Les cardinalités maximum sont nécessaires pour concevoir le schéma de la base de données
- Les cardinalités minimums sont nécessaires pour exprimer les contraintes d'intégrité
- En notant uniquement les cardinalités maximum, on distingue 3 type de liens :
 - • **Lien fonctionnel 1:n**
 - • **Lien hiérarchique n:1**
 - • **Lien maillé n:m**

LE DOMAINE

- ensemble de valeurs atomiques d'un certain type
- sémantique
- Ex. `nom_etud=(thiam, diop)`

LA RELATION

- sous ensemble du produit cartésien de plusieurs domaines
- $R \subset D1 \times D2 \times \dots \times Dn$
- $D1, D2, \dots, Dn$ sont les domaines de R n est le degré ou l'arité de R

LES N-UPLETS

- un élément d'une relation est un n-uplet de valeurs (tuple en anglais)
- • un n-uplet représente un fait
- Ex.:
- « Demba est un élève né le 1 janvier 1992 »

LES ATTRIBUTS

- Chaque composante d'une relation est un attribut
 - Le nom donné à un attribut est porteur de sens
 - Il est en général différent du nom de domaine
 - Plusieurs attributs peuvent avoir le même domaine

LE SCHÉMA D'UNE RELATION

- Le schéma d'une relation est défini par :
 - - le nom de la relation
 - - la liste de ses attributs
- on note : $R (A_1, A_2, \dots, A_n)$
- Ex.:
- ELEVE (NOM, PRENOM, NAISS)
- INSCRIPT (NOM_ELIV, SPORT)
- TRAJET (VD, VA)

LE SCHÉMA D'UNE BDR

- Le schéma d'une base de données est défini par :
 - - l'ensemble des schémas des relations qui la composent
- Notez la différence entre :
 - • le schéma de la BDR qui dit comment les données sont organisées dans la base
 - • l'ensemble des n-uplets de chaque relation, qui représentent les données stockées dans la base

1 RELATION = 1 TABLE

U1	V1	W1	X1	Y1
U2	V2	W2	X2	Y2
U3	V3	W3	X3	Y3

1 ÉLÉMENT ou n-uplet = 1 LIGNE

LIGNE →

1 élément

U1	V1	W1	X1	Y1

* une relation est un ensemble \Rightarrow on ne peut pas avoir 2 lignes identiques

1 ATTRIBUT = 1 COLONNE

U1				
U2				
U3				



COLONNE

1 attribut ou propriété

Formes normales

Dépendance fonctionnelle

- Soit $R(A_1, A_2, \dots, A_n)$ un schéma de relation
- Soit X et Y des sous ensembles de $\{A_1, A_2, \dots, A_n\}$
- On dit que Y dépend fonctionnellement de X ($X \rightarrow Y$) si à chaque valeur de X correspond une valeur unique de Y
 - on écrit : $X \rightarrow Y$
 - on dit que : X détermine Y
- Ex.:
- PRODUIT ($no_prod, nom, prixUHT$)
- $no_prod \rightarrow (nom, prixUHT)$
- NOTE ($no_contrôle, no_élève, note$)
- $(no_contrôle, no_élève) \rightarrow note$

D.F. élémentaire

- D.F. $X \rightarrow A$ mais A est un attribut unique non inclus dans X et il n'existe pas de X' inclus dans X tel que $X' \rightarrow A$

Notion de clé

- Définition (clé de relation) : Soit $R(A_1, A_2, \dots, A_n)$ un schéma de relation, et X un sous-ensemble de (A_1, A_2, \dots, A_n) , X est une clé si, et seulement si, :
 - $X \rightarrow (A_1, A_2, \dots, A_n)$
 - X est minimal
- Ex.:
- PRODUIT (**no_prod**, nom, prixUHT)
- **no_prod** \rightarrow (nom, prixUHT)
- **no_prod** est une clé
- • Une clé détermine un n-uplet de façon unique
- • Pour trouver la clé d'une relation, il faut examiner attentivement les hypothèses sur le monde réel
- • Une relation peut posséder plusieurs clés, on les appelle clés candidates
- Ex.:
- dans la relation PRODUIT, nom est une clé candidate (à condition qu'il n'y ait jamais 2 produits de même nom)

clé

- **Clé primaire**

- choix d'une clé parmi les clés candidates

- **Clé étrangère ou clé secondaire**

- attribut (ou groupe d'attributs) qui fait référence à la clé primaire d'une autre relation

- Ex.:

- CATEG (**no_cat**, design, tva)

- PRODUIT(no_prod, nom, marque, no_cat, prixUHT)

- no_cat dans PRODUIT est une clé étrangère

- CLÉ ÉTRANGÈRE = CLÉ PRIMAIRE dans une autre relation

La théorie de la normalisation

- elle met en évidence les relations "indésirables"
- elle définit les critères des relations "désirables« appelées **formes normales**
- Propriétés indésirables des relations
 - - Redondances
 - - Valeurs NULL
- elle définit le processus de normalisation permettant de décomposer une relation non normalisée en un ensemble équivalent de relations normalisées

La décomposition

- **Objectif:**
 - - décomposer les relations du schéma relationnel sans perte d'informations
 - - obtenir des relations canoniques ou de base du monde réel
 - - aboutir au schéma relationnel normalisé
 - • Le schéma de départ est le schéma universel de la base
 - • Par raffinement successifs on obtient des sous relations sans perte d'informations et qui ne seront pas affectées lors des mises à jour (non redondance)

Les formes normales

- **5 FN**, les critères sont de plus en plus restrictifs
- $FN_j \Rightarrow FN_i \ (j > i)$
- **Notion intuitive de FN**
- une « *bonne relation* » peut être considérée comme une fonction de la clé primaire vers les attributs restants

1ère Forme Normale 1FN

- **Définition**

- Une relation est en 1FN si tout attribut est atomique (non décomposable)

- **Contre-exemple**

- ELEVE (**no_elv**, nom, prenom, liste_notes)
- Un attribut ne peut pas être un ensemble de valeurs

- **Décomposition**

- ELEVE (**no_elv**, nom, prenom)
- NOTE (no_elv, no_matiere, note)

2ème Forme Normale 2FN

- **Définition :**

- Une relation est en 2FN si
 - - elle est en 1FN
 - - si tout attribut n'appartenant pas à la clé ne dépend pas d'une partie de la clé
- C'est la phase d'identification des clés
- Cette étape évite certaines redondances
- Tout attribut doit dépendre fonctionnellement de la totalité de la clé

- **Contre-exemple**

- une relation en 1FN qui n'est pas en 2FN
- COMMANDE (**date, no_cli, no_pro**, qte, prixUHT)
- elle n'est pas en 2FN car la clé = (**date, no_cli,**
- **no_pro**), et le prixUHT ne dépend que de no_pro

- **Décomposition**

- COMMANDE (**date, no_cli, no_pro**, qte)
- PRODUIT (**no_pro**, prixUHT)

3ème Forme Normale 3FN

- **Définition :** Une relation est en 3FN si
 - - elle est en 2FN
 - - si tout attribut n'appartenant pas à la clé ne dépend pas d'un attribut non clé
- Ceci correspond à la non transitivité des D.F. ce qui évite les redondances.
- En 3FN une relation préserve les D.F. et est sans perte.
- **Contre-exemple**
 - une relation en 2FN qui n'est pas en 3FN
 - VOITURE (**matricule**, marque, modèle, puissance)
 - on vérifie qu'elle est en 2FN ; elle n'est pas en 3FN car la clé =
 - **matricule**, et la puissance dépend de (marque, modèle)
- **Décomposition**
 - VOITURE (**matricule**, marque, modèle)
 - MODELE (**marque, modèle**, puissance)

Forme Normale de BOYCE-CODD BCNFD

• Définition

- Une relation est en BCNF :
 - - elle est en 1FN et
 - - ssi les seules D.F. élémentaires sont celles dans lesquelles une clé détermine un attribut
- BCNF signifie que l'on ne peut pas avoir un attribut (ou groupe d'attributs) déterminant un autre attribut et distinct de la clé
- Ceci évite les redondances dans l'extension de la relation: mêmes valeurs pour certains attributs de nuplets différents
- BCNF est plus fin que FN3 : $BCNF \Rightarrow FN3$

• Contre-exemple

- une relation en 3FN qui n'est pas BCNF
- CODEPOSTAL (**ville**, **rue**, code)
- on vérifie qu'elle est FN3, elle n'est pas BCNF car la clé = (ville, rue) (ou (code, ville) ou (code, rue)), et code \rightarrow ville

Les opérations de l'Algèbre relationnelle

- L'Algèbre relationnelle est une collection d'opérations
- **OPÉRATIONS**
 - opérandes : 1 ou 2 relations
 - résultat : une relation
- **DEUX TYPES D'OPÉRATIONS**
 - **OPÉRATIONS SPÉCIFIQUES**
 - PROJECTION
 - RESTRICTION
 - JOINTURE
 - DIVISION
 - **OPÉRATIONS ENSEMBLISTES**
 - UNION
 - INTERSECTION
 - DIFFÉRENCE

OPÉRATIONS SPÉCIFIQUES
PROJECTION
RESTRICTION
JOINTURE
DIVISION

Opérateurs relationnels

Projection

- **Définition :**
 - Opération qui consiste à supprimer des attributs d'une relation et à éliminer les tuples en double apparaissant dans la nouvelle relation.
 - Cette opération est notée Π ;

Opérateurs relationnels


Projection

- La projection d'une relation R1 est la relation R2 obtenue en supprimant les attributs de R1 non mentionnés puis en éliminant éventuellement les nuplets identiques
- On notera :
 - **$R2 = \pi_{A_i, A_j, \dots, A_m}(R1)$**
 - la projection d'une relation R1 sur les attributs A_i, A_j, \dots, A_m

Opérateurs relationnels

Projection

- La projection permet d'éliminer des attributs d'une relation
 - Elle correspond à un découpage vertical :



A1	A2	A3	A4

Opérateurs relationnels

Projection

- **Requête 1 :**
- « Quels sont les références et les prix des produits ? »

PRODUIT (IdPro, Nom, Marque, Prix)



IdPro	Nom	Marque	Prix
P	PS1	IBM	1000
Q	Mac	Apple	2000
R	PS2	IBM	3000
S	Word	Microsoft	4000

- $\pi_{\text{IdPro, Prix}}(\text{Produit})$

IdPro	Prix
P	1000
Q	2000
R	3000
S	4000

Opérateurs relationnels

Projection

- **Requête 2 :**
- « Quelles sont les marques des produits ? »

Opérateurs relationnels

Restriction

- **Définition :**

- Opération qui consiste à supprimer les tuples d'une relation ne satisfaisant pas la condition précisée.
- Cette opération est notée σ

Opérateurs relationnels


Restriction

- La restriction d'une relation R1 est une relation R2 de même schéma n'ayant que les n-uplets de R1 répondant à la condition énoncée
- On notera :
 - **$R2 = \sigma_{\text{condition}}(R1)$**
- la restriction d'une relation R1 suivant le critère "condition« où "condition" est une relation d'égalité ou d'inégalité entre 2 attributs ou entre un attribut et une valeur

Opérateurs relationnels

Restriction

- La restriction permet d'extraire les n-uplets qui satisfont une condition
 - Elle correspond à un découpage horizontal :




A1	A2	A3	A4

Opérateurs relationnels

Restriction

- **Requête 3 :**
- « Quelles sont les produits de marque 'IBM' ? »
- **PRODUIT (IdPro, Nom, Marque, Prix)**



IdPro	Nom	Marque	Prix
P	PS1	IBM	1000
Q	Mac	Apple	2000
R	PS2	IBM	3000
S	Word	Microsoft	4000

Opérateurs relationnels

Restriction

- $\sigma_{\text{Marque}=\text{"IBM"}}(\text{Produit})$

IdPro	Nom	Marque	Prix
P	PS1	IBM	1000
R	PS2	IBM	3000

Opérateurs relationnels

Jointure

- Définition :
 - Opération qui consiste à faire le produit cartésien de deux relations, puis à supprimer les tuples ne satisfaisant pas une condition portant sur un attribut de la première relation et sur un attribut de la seconde.

Opérateurs relationnels

Jointure

- La jointure de deux relations R1 et R2 est une relation R3 dont les n-uplets sont obtenus en concaténant les nuplets de R1 avec ceux de R2 et en ne gardant que ceux qui vérifient la condition de liaison
- On notera :
- **$R3 = R1 \bowtie R2 \text{ (condition)}$ ou $R1 \ltimes R2 \text{ (condition)}$**
- la jointure de R1 avec R2 suivant le critère condition
- Le schéma de la relation résultat de la jointure est la concaténation des schémas des opérandes (s'il y a des attributs de même nom, il faut les renommer)
- Les n-uplets de $R1 \bowtie R2 \text{ (condition)}$ sont tous les couples (u1,u2) d'un n-uplet de R1 avec un n-uplet de R2 qui satisfont "condition"

Opérateurs relationnels

Jointure

- La jointure de deux relations R1 et R2 est le produit
- cartésien des deux relations suivi d'une restriction
- La condition de liaison doit être du type :
- $\langle \text{attribut1} \rangle :: \langle \text{attribut2} \rangle$
- où : $\text{attribut1} \in 1^{\text{ère}} \text{ relation}$ et $\text{attribut2} \in 2^{\text{ème}} \text{ relation}$
- $::$ est un opérateur de comparaison (égalité ou inégalité)

Opérateurs relationnels

Jointure

- La jointure permet de composer 2 relations à l'aide d'un critère de liaison

R1(A, B, C)

A	B	C
A1	B1	10
A2	B2	10
A3	B3	20
A4	B4	30



R2(U, V)

U	V
10	V1
20	V2
30	V3

R1 × R2 (R1.C = R2.U)



A	B	C	U	V
A1	B1	10	10	V1
A1	B2	10	10	V1
A3	B3	20	20	V2
A4	B4	30	30	V3

Jointure naturelle

- Jointure où l'opérateur de comparaison est l'égalité dans le résultat on fusionne les 2 colonnes dont les valeurs sont égales
- La jointure permet d'enrichir une relation
- **Requête 5 :**
- « Donnez pour chaque vente la référence du produit, sa désignation, son prix, le numéro de client, la date et la quantité vendue »

VENTE As V

IdCli	IdPro	Date	Qte
X	P	1/1/98	1
Y	Q	2/1/98	1
Z	P	3/1/98	1

×

PRODUIT As P

IdPro	Désignation	Prix
P	PS	100
Q	Mac	100

Jointure naturelle

- **VENTE** ⋈ **PRODUIT** (V.IdPro=P.IdPro)

Idcli	IdPro	Date	Qte	Désignation	Prix
X	P	1/1/98	1	PS	100
Y	Q	2/1/98	1	Mac	100
Z	P	3/1/98	1	PS	100

- La normalisation conduit à décomposer ; la jointure
- permet de recomposer