

MODÈLE RELATIONNEL

I. Définition du modèle relationnel.

L'objectif est de découvrir comment manipuler des données. Le **modèle relationnel** est une manière de **représenter** et de **manipuler des données structurées**. Ce modèle est à l'origine des bases de données relationnelles, qui sont massivement utilisées, et dont découle le célèbre langage SQL.

Ce type de représentation de la donnée est indépendant de la manière dont la donnée est physiquement stockée dans la mémoire de l'ordinateur. Ainsi, quel que soit le type de technologie (processeur, système d'exploitation, etc.) que vous utilisez, vous serez toujours capable de dialoguer avec d'autres personnes utilisant des technologies différentes de la vôtre.

Une relation, c'est un tableau dans lequel on met des données. Un tableau dans lequel une ligne représente un objet, et où chaque ligne représente des objets de même nature.

Si nous représentons des tomates, nous pouvons les caractériser par leur masse, leur diamètre et leur couleur. Nous pouvons aussi leur attribuer un identifiant, c'est-à-dire un nombre (ou un code) unique qui permet de les différencier.

Table **tomate**

identifiant	masse	diamètre	couleur
1	230	100	rouge
2	100	70	jaune
3	110	50	verte
4	230	100	rouge

Remarques :

Une **relation** s'appelle aussi une **table**.

Une **ligne** s'appelle aussi un **enregistrement** ou un **tuple**.

Une **colonne** s'appelle aussi un **attribut** ;

Dans une relation, l'ordre des lignes n'a pas d'importance et n'a pas de signification ;

Le **schéma** est **l'ensemble des attributs d'une relation**.

Une relation ne peut contenir deux tuples identiques : chaque ligne est donc unique !

Or deux tomates distinctes peuvent avoir les mêmes caractéristiques.

Comment donc les distinguer alors ? On utilise une **clé** qui permet l'identification des **tuples** d'une relation.

II. Définition d'une clé primaire.

Dans le modèle relationnel, une clé sert à identifier et accéder à un tuple unique.

Généralement, une clé est soit :

- un groupe d'attributs (colonnes). Il est courant d'indiquer les attributs de la **clé primaire** en indiquant leur nom suivi de [PK] (mis pour **primary key**, **clé primaire** en anglais)
- une **clé artificielle**. Une **clé artificielle** est un attribut que l'on ajoute à la relation. Cet attribut n'a pas de réelle signification dans le domaine que l'on modélise, mais sa seule fonction est d'identifier de manière unique les **tuples** de la relation. Dans l'exemple

précédent, l'**attribut** *identifiant* est une **clé artificielle**. Elle ne caractérise pas l'objet (tomate dans l'exemple précédent).

III. Définition d'une clé étrangère.

En général, une base de données contient plusieurs tables et elles sont dépendantes entre elles. On ajoute une seconde **relation** variété destinée à répertorier les différentes variétés de tomates.

Table **variété**

nom	prix_sachet	maturation	saveur
Noire de Crimée	4,90	précocce	sucrée
Rose de Berne	4,50	tardive	sucrée
Supersteak	5,90	mi-tardive	cacahuète
Reine Sainte-Marthe	4,50	mi-saison	acide

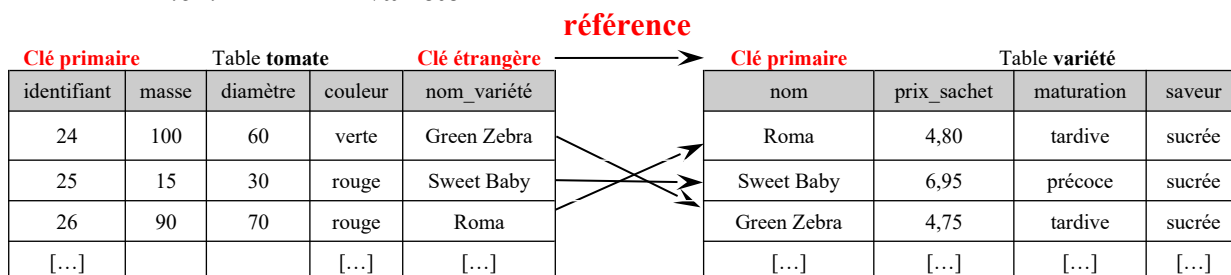
Chez un maraîcher, le prix d'une tomate dépend de sa variété. Le prix du sachet est donc une caractéristique propre à la variété. On aura donc dans la relation variété un **attribut** *prix_sachet*.

Si je prends une tomate, et que je veux connaître son prix au sachet, il faudra d'abord que je connaisse sa variété. Ensuite, à partir de sa variété, il faudra que je retrouve dans la table **variété** la ligne correspondante pour connaître le prix.

Dans notre relation **variété**, il y a une clé candidate qui peut servir de **clé primaire** pour cette table : l'attribut identifiant. Ainsi, si pour chaque ligne de la table **tomate** nous renseignons le nom de la variété dans une colonne, alors il sera possible, grâce à cette clé, de retrouver le prix au sachet. Voici à quoi pourrait ressembler notre nouvelle table **tomate** :

identifiant	masse	diamètre	couleur	nom_variété
24	100	60	verte	Green Zebra
25	15	30	rouge	Sweet Baby
26	90	70	rouge	Roma
[...]			[...]	[...]

Dans la table **tomate**, la colonne *nom_variété* fait référence à la **clé primaire** d'une autre table (la table **variété**). On dit donc que la colonne *nom_variété* est une **clé étrangère** référençant la colonne *nom* de la table **variété**.

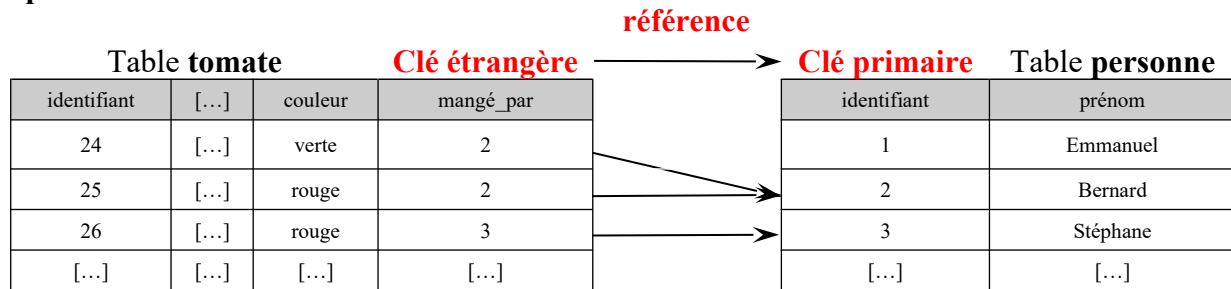


Pour indiquer qu'un **attribut** (ou groupe d'attributs) est une **clé étrangère**, on peut le spécifier en faisant suivre le nom de cet attribut (ou de chacun de ces attributs) par [FK], qui signifie **foreign key** (**clé étrangère** en anglais).

IV. Table d'association.

On ajoute la table **personne** qui contient des gens susceptible de manger des tomates. Une personne peut manger plusieurs tomates, mais plusieurs personnes peuvent aussi se partager une grosse tomate.

Dans la table **tomate**, mettons une colonne mangé_par qui est une **clé étrangère** vers la table **personne**.



Problème : comment faire si une tomate est mangée par un nombre indéterminé de personnes différentes ? Inverser le sens de la clé étrangère ne fait qu'inverser le problème.

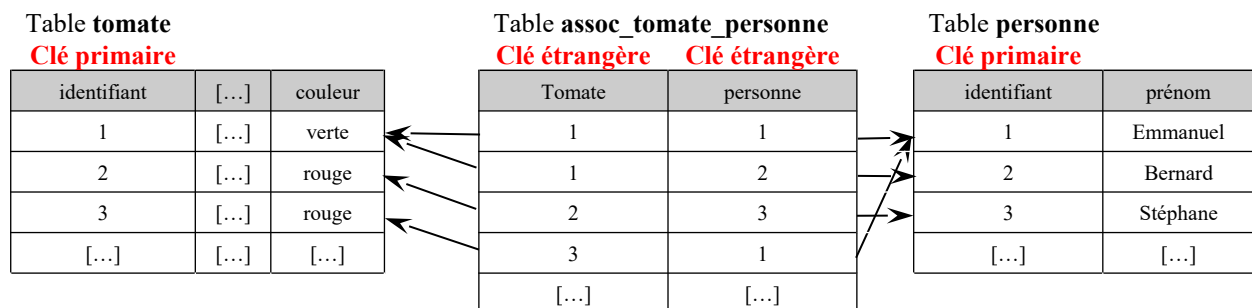
Se poser la question "Combien de tomates peut manger une personne, et par combien de personnes peut être mangée une tomate", c'est se poser la question de la cardinalité du lien entre tomate et personne.

Plusieurs cardinalités sont possibles :

- de 1 à 1 (ex : une tomate pour une personne) ;
- de 1 à plusieurs (ex : une tomate pour plusieurs personnes) ;
- de plusieurs à 1 (ex : plusieurs tomates pour une personne) ;
- de plusieurs à plusieurs (ex : plusieurs tomates pour une personne, plusieurs personnes pour une tomate)

La solution est de créer une troisième table, que l'on appelle parfois **table d'association**.

La table d'association est utilisée dans le cadre d'une cardinalité plusieurs-à-plusieurs entre deux objets. Elle est composée d'au moins 2 **clés étrangères**, référençant chacune l'un des 2 objets.



On remarque ici qu'Emmanuel mange à la fois la tomate 1 et la tomate 3, mais que la tomate 1 est également mangée par Bernard. **La clé primaire d'une table d'association est composée d'au moins des deux clés étrangères.** Mais il est parfois nécessaire d'ajouter des colonnes supplémentaires pour créer cette **clé primaire**.

V. Projection et restriction.

Ces deux opérations portent sur **une seule relation**.

- La **projection**, c'est simplement le fait de sélectionner les **attributs** d'une **relation** que l'on souhaite, en oubliant les autres. La table qui résultera d'une **projection** aura donc le même nombre de lignes mais avec moins de colonnes.
- La **restriction**, c'est simplement le fait de filtrer les lignes d'une **relation** que l'on souhaite. Pour faire cela, il faut une **condition**. La table qui résultera d'une **restriction** aura donc le même nombre de colonnes mais avec moins de lignes.

Remarques concernant la **restriction** :

- Comme les lignes à garder (ou enlever) sont dynamiques, il nous faut donc établir une condition qui nous permette de *restreindre* les lignes.
- Une **restriction** peut également porter sur plusieurs colonnes.

VI. Les opérateurs ensemblistes.

Les opérateurs ensemblistes sont des opérations qui portent sur 2 **relations** de même **schéma** (elles ont les mêmes **attributs**).

6.1 L'union.

L'union de deux relations R1 et R2 de même schéma produit une troisième relation, également de même schéma, qui contient l'ensemble des lignes (ou tuples) de R1 et de R2.

6.2 La différence.

La différence entre une relation R3 et R2 donne une relation R1 qui contient tous les lignes (ou tuples) de R3 qui n'appartiennent pas à R2.

6.3 L'intersection.

L'intersection entre deux relations R1 et R2 donne une troisième relation contenant les lignes (ou tuples) qui sont présents à la fois dans R1 et dans R2.

VII. Le produit cartésien.

En algèbre relationnelle, le **produit cartésien** entre deux relations R1 et R2 est justement composé de toutes les combinaisons possibles entre les tuples de R1 et les tuples de R2.

Exemple.

Table **personne**

identifiant	prénom
1	Bernard
2	Emmanuel
3	Stéphane
4	Toto

Table **variété**

nom	prix_sachet	maturation	goût
Noire de Crimée	4,90	précoce	sucrée
Rose de Berne	4,50	tardive	sucrée
Supersteak	5,90	mi-tardive	cacahuète

Le **produit cartésien** des 2 tables précédentes est la table dégustation suivante :

Table dégustation

identifiant	prénom	nom	prix_sachet	maturation	goût
1	Bernard	Noire de Crimée	4,90	précoce	sucrée
1	Bernard	Rose de Berne	4,50	tardive	sucrée
1	Bernard	Supersteak	5,90	mi-tardive	cacahuète
2	Emmanuel	Noire de Crimée	4,90	précoce	sucrée
2	Emmanuel	Rose de Berne	4,50	tardive	sucrée
2	Emmanuel	Supersteak	5,90	mi-tardive	cacahuète
3	Stéphane	Noire de Crimée	4,90	précoce	sucrée
3	Stéphane	Rose de Berne	4,50	tardive	sucrée
3	Stéphane	Supersteak	5,90	mi-tardive	cacahuète
4	Toto	Noire de Crimée	4,90	précoce	sucrée
4	Toto	Rose de Berne	4,50	tardive	sucrée
4	Toto	Supersteak	5,90	mi-tardive	cacahuète

Le nombre de lignes d'un **produit cartésien** entre une relation de 4 tuples par une relation de 3 tuples donne une relation de 12 tuples.

VIII. Les jointures.

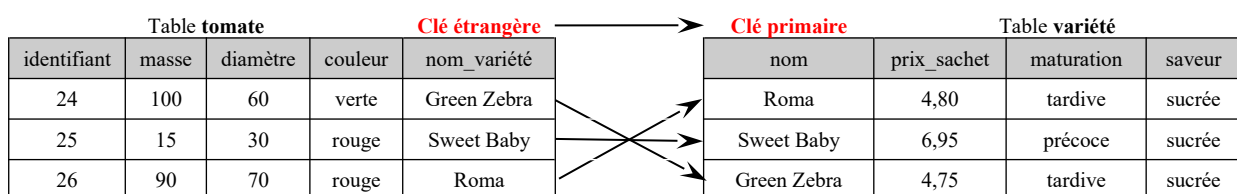
Une jointure permet d'exploiter les liaisons entre deux tables sachant qu'elles sont déjà liées à l'aide d'une clé étrangère.

La jointure aura pour but de coller deux tables. La jointure créera une grande table qui contiendra à la fois les informations de la première et de la seconde table.

8.1 La jointure interne.

On veut connaître le prix du sachet de graines de tomate prise au hasard. Il faut d'abord connaître sa variété puis à l'aide de cette information trouver, dans la table **variété** la ligne correspondante, pour connaître son prix.

Plus formellement, on dit que l'on effectue une jointure interne de la relation tomate et de la relation variété selon la condition $\text{tomate.nom_variété} = \text{variété.nom}$.



On jointe les deux tables, le résultat est le suivant :

tomate. identifiant	tomate. masse	tomate. diamètre	tomate. couleur	tomate. nom_variété	variété. nom	variété. prix_sachet	variété. maturation	variété. saveur
24	100	60	verte	Green Zebra	Green Zebra	4,75	tardive	sucrée
25	15	30	rouge	Sweet Baby	Sweet Baby	6,95	précoce	sucrée
26	90	70	rouge	Roma	Roma	4,80	tardive	sucrée

L'objectif est atteint : l'identifiant de la tomate permet d'accéder au prix du sachet qui est sur la même ligne.

En général, quand on emploie le terme jointure, on parle de jointure interne (**INNER JOIN**).

8.2 La jointure externe.

Prenons le cas où nous ne connaissons pas la variété de la tomate 27 ? Dans ce cas, la clé étrangère aura une valeur nulle, elle n'aura donc pas de correspondance dans la table variété.

Table **tomate**

identifiant	masse	diamètre	couleur	nom_variete
24	100	60	verte	Green Zebra
25	15	30	rouge	Sweet Baby
26	90	70	rouge	Roma
27	230	100	rouge	null

La condition de jointure que nous avons vu précédemment ne sera pas satisfaite :
tomate.nom_variete = variete.libelle car tomate.nom_variete est nul.

Si on souhaite quand même garder les tomates dont on ne connaît pas la variété, il faut faire une **jointure externe**. Si on considère que la table tomate est à gauche, et que la table variété est à droite, on effectue une **jointure externe à gauche**, car on garde toutes les lignes de la table de gauche.

Résultat d'une jointure externe à gauche.

tomate. identifiant	tomate. masse	tomate. diamètre	tomate. couleur	tomate. nom_variété	variete. nom	variete. prix_sachet	variete. maturation	variete. saveur
24	100	60	verte	Green Zebra	Roma	4,80	tardive	sucrée
25	15	30	rouge	Sweet Baby	Sweet Baby	6,95	précoce	sucrée
26	90	70	rouge	Roma	Green Zebra	4,75	tardive	sucrée
27	230	100	rouge	null	null	null	null	null

Si on connaît la variété d'une tomate mais on ne connaît pas ses caractéristiques, il faut faire une **jointure externe à droite**, car on garde toutes les lignes de la table de droite.

Résultat d'une jointure externe à droite.

tomate. identifiant	tomate. masse	tomate. diamètre	tomate. couleur	tomate. nom_variété	variete. nom	variete. prix_sachet	variete. maturation	variete. saveur
24	100	60	verte	Green Zebra	Roma	4,80	tardive	sucrée
25	15	30	rouge	Sweet Baby	Sweet Baby	6,95	précoce	sucrée
26	90	70	rouge	Roma	Green Zebra	4,75	tardive	sucrée
null	null	null	null	null	Ananas	5,2	tardive	fruitée

Si on souhaite garder à la fois les informations de la table tomate de celles de la table variété, il faut faire une **jointure externe totale**.

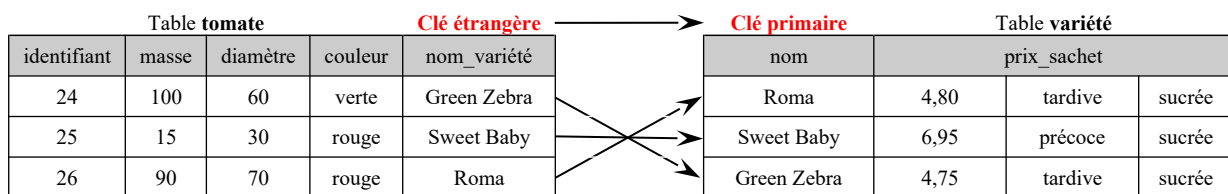
Résultat d'une jointure externe totale.

tomate. identifiant	tomate. masse	tomate. diamètre	tomate. couleur	tomate. nom_variété	variété. nom	variété. prix_sachet	variété. maturation	variété. saveur
24	100	60	verte	Green Zebra	Roma	4,80	tardive	sucrée
25	15	30	rouge	Sweet Baby	Sweet Baby	6,95	précoce	sucrée
26	90	70	rouge	Roma	Green Zebra	4,75	tardive	sucrée
27	230	100	rouge	null	null	null	null	null
null	null	null	null	null	Ananas	5,2	tardive	Fruitée

8.3 Différences entre un produit cartésien et une jointure naturelle.

Une jointure naturelle est en fait l'enchaînement d'un produit cartésien et d'une restriction.

Exemple : on reprend le premier exemple pour la jointure naturelle.



Le produit cartésien de tomate par variété est le suivant :

tomate. identifiant	tomate. masse	tomate. diamètre	tomate. couleur	tomate. nom_variété	variété. nom	variété. prix_sachet	variété. maturation	variété. saveur
24	100	60	verte	Green Zebra	Roma	4,80	tardive	sucrée
24	100	60	verte	Green Zebra	Sweet Baby	6,95	précoce	sucrée
24	100	60	verte	Green Zebra	Green Zebra	4,75	tardive	sucrée
25	15	30	rouge	Sweet Baby	Roma	4,80	tardive	sucrée
25	15	30	rouge	Sweet Baby	Sweet Baby	6,95	précoce	sucrée
25	15	30	rouge	Sweet Baby	Green Zebra	4,75	tardive	sucrée
26	90	70	rouge	Roma	Roma	4,80	tardive	sucrée
26	90	70	rouge	Roma	Sweet Baby	6,95	précoce	sucrée
26	90	70	rouge	Roma	Green Zebra	4,75	tardive	sucrée

Pour le moment, cela n'a pas vraiment de sens, car on a associé chacune des tomates avec chacune des variétés. Des tomates sont associées à des variétés auxquelles elles n'appartiennent pas. Il y a donc des lignes incohérentes donc inutiles.

tomate. identifiant	tomate. masse	tomate. diamètre	tomate. couleur	tomate. nom_variété	variété. nom	variété. prix_sachet	variété. maturation	variété. saveur
24	100	60	verte	Green Zebra	Roma	4,80	tardive	sucrée
24	100	60	verte	Green Zebra	Sweet Baby	6,95	précoce	sucrée
24	100	60	verte	Green Zebra	Green Zebra	4,75	tardive	sucrée
25	15	30	rouge	Sweet Baby	Roma	4,80	tardive	sucrée
25	15	30	rouge	Sweet Baby	Sweet Baby	6,95	précoce	sucrée
25	15	30	rouge	Sweet Baby	Green Zebra	4,75	tardive	sucrée
26	90	70	rouge	Roma	Roma	4,80	tardive	sucrée
26	90	70	rouge	Roma	Sweet Baby	6,95	précoce	sucrée
26	90	70	rouge	Roma	Green Zebra	4,75	tardive	sucrée

On retrouve le même résultat que lorsque l'on a joint les deux tables :

tomate. identifiant	tomate. masse	tomate. diamètre	tomate. couleur	tomate. nom_variété	variete. nom	variete. prix_sachet	variete. maturation	variete. saveur
24	100	60	verte	Green Zebra	Roma	4,80	tardive	sucrée
25	15	30	rouge	Sweet Baby	Sweet Baby	6,95	précoce	sucrée
26	90	70	rouge	Roma	Green Zebra	4,75	tardive	sucrée

8.4 La jointure naturelle.

C'est une jointure classique qui est effectuée sur deux relations dont les colonnes à lier ont exactement le même nom d'une table à l'autre.

Il suffit par exemple de renommer la colonne de la table variété en nom_variété.

Table tomate						Table variété			
identifiant	masse	diamètre	couleur	Clé étrangère nom_variété		Clé primaire nom_variété	prix_sachet		
24	100	60	verte	Green Zebra		Roma	4,80	tardive	sucrée
25	15	30	rouge	Sweet Baby		Sweet Baby	6,95	précoce	sucrée
26	90	70	rouge	Roma		Green Zebra	4,75	tardive	sucrée

Avec la jointure naturelle, plus besoin d'indiquer une condition, car elle est implicite, vu que les colonnes ont le même nom.

IX. L'agrégation.

L'**agrégation** est utilisée lorsque l'on veut calculer un résultat qui porte sur plusieurs lignes d'une table. On dit que l'on agrège ces lignes, c'est-à-dire que l'on forme des agrégats pour effectuer sur eux une opération.

Une agrégation se fait en deux étapes :

- **partitionnement avec un groupe d'attributs ;**
- **application d'une ou de plusieurs fonction(s) d'agrégation.**

Table tomate

identifiant	masse	diamètre	couleur	mûre
1	230	100	rouge	VRAI
2	100	70	jaune	FAUX
3	110	50	verte	FAUX
4	230	100	rouge	FAUX
5	200	90	jaune	VRAI
6	120	80	rouge	FAUX
7	80	50	verte	FAUX
8	90	60	rouge	VRAI

On désire savoir la masse moyenne de chaque tomate selon leur couleur et leur maturation. Il y aura autant de valeurs de masse moyenne calculées que de tomates de la même couleur mûres et de tomates de la même couleur non mûres.

On partitionne selon les **attributs** [couleur, mûre]

identifiant	masse	diamètre	couleur	mûre		identifiant	masse	diamètre	couleur	mûre
1	230	100	rouge	VRAI		1	230	100	rouge	VRAI
2	100	70	jaune	FAUX		8	90	60	rouge	VRAI
3	110	50	verte	FAUX		2	100	70	jaune	FAUX
4	230	100	rouge	FAUX		3	110	50	verte	FAUX
5	200	90	jaune	VRAI		7	80	50	verte	FAUX
6	120	80	rouge	FAUX		4	230	100	rouge	FAUX
7	80	50	verte	FAUX		6	120	80	rouge	FAUX
8	90	60	rouge	VRAI		5	200	90	jaune	VRAI

On obtient 5 agrégats sur lesquels on peut appliquer une ou plusieurs fonctions d'**agrégation**.
La table agrégée aura de lignes que d'agrégats.

couleur	mûre	average	COUNT
rouge	VRAI	80	2
jaune	FAUX	70	1
verte	FAUX	50	2
rouge	FAUX	90	2
jaune	VRAI	90	1

La masse moyenne des 2 tomates rouges mûres est de 80 g.

Remarques :

- En général, une fonction d'**agrégation** prend en entrée une liste et renvoie une unique valeur.
- La liste en entrée correspond à 1 **attribut** (ici, l'**attribut** masse). Mais il est également possible de trouver des fonctions d'**agrégation** qui prennent plusieurs **attributs** en entrée (donc plusieurs listes) ou parfois même l'ensemble des **attributs** d'une table.