

Introduction aux bases de données

C.Defrance

version v1.1

Objectifs du cours

À l'issue de ce cours, vous devez être capable de ... :

- définir ce qu'est une base de données
- citer les étapes de conception d'une base de données
- définir le vocabulaire associé aux bases de données
- interpréter les multiplicités et associations présents sur un modèle conceptuel de données
- évaluer l'impact d'un certain type d'association sur les tables d'une base de données
- citer les rôles d'un Système de Gestion de Bases de Données

Introduction



Définition :

Une base de données est un ensemble **structuré** de **données**.

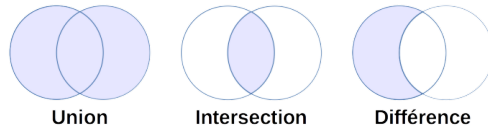
- donnée = information éventuellement codée pour être traitée, conservée, communiquée
- «structuré» indique que les données sont organisées



Base de données relationnelles

- Base de données dont les informations sont organisées sous forme de **tables**
- Supporte les opérations de :
 - l'algèbre relationnelle :

idPays	codeCIO	nom
1	AFG	Afghanistan
2	ALB	Albanie
3	ALG	Algérie
4	AND	Andorre

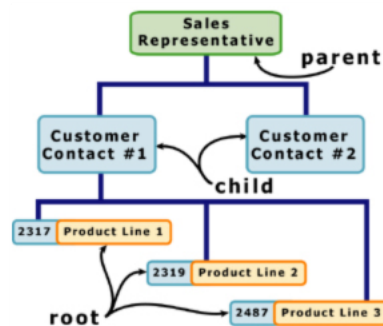


- la logique des proposition et des prédicats :
opérateurs booléens (NOT, OR, AND...) et de comparaison (<, >, =, !=...) pour sélectionner des valeurs sur un ou plusieurs critères
ex. : sélectionner tous les pays dont le nom débute par un **A** et comporte plus de 10 caractères

→ `(SUBSTR(nom, 1, 1) = "A") AND (LENGTH(nom) > 10)`

- Type de base de données le plus répandu mais il y en a d'autres...

Les différents modèles de base de données



BDD Hiérarchique



BDD Réseau

Object Type <i>person_type</i>	
Attributes	Methods
idno first_name last_name email phone	get_idno display_details

Object	
idno:	65
first_name:	Verna
last_name:	Mills
email:	vmills@oracle.com
phone:	1-800-555-4412

Object	
idno:	101
first_name:	John
last_name:	Smith
email:	jsmith@oracle.com
phone:	1-800-555-1212

BDD Objet

Database 1		
1	First Name	Last Name Social Security No.
2	John	Smith 010-22-9432
3	John	Smith 003-63-0037
4	John	Smith 000-45-0000
5	Sally	Smith
6	Steve	Smith
7		

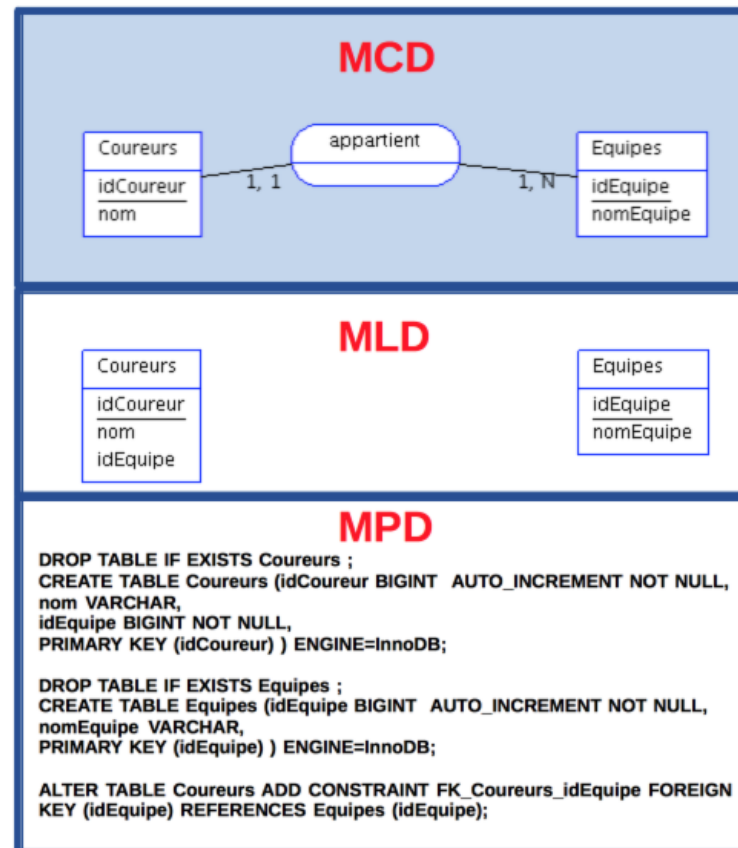
Database 2		
1	Date of Birth	Social Security No.
2	6/12/82	010-22-9432
3	5/9/40	003-63-0037
4	12/4/57	020-45-9326
5		
6		

Database 3		
1	Address	Social Security No.
2	321 Byberry Road	010-22-9432
3	268 Monroe Avenue	003-63-0037
4	8120 Venshire Drive	020-45-9326
5	207 Congress Drive	289-56-4321
6	1519 Ashbury Lane	070-54-2333

BDD Relationnelle

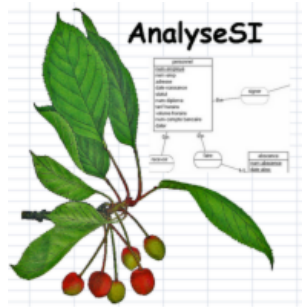
Conception d'une Base De Données (BDD) [1/2]

- 3 étapes :
 - Identifier à partir des règles métier les différents objets ainsi que leurs relations
⇒ **Modèle Conceptuel de Données (MCD)**
 - Transformer ce modèle sous formes de tables
⇒ **Modèle logique de données (MLD)**
 - Créer physiquement ces tables dans le logiciel qui gèrera la BDD à l'aide d'un langage spécialisé appelé SQL
⇒ **Modèle Physique de Données (MPD)**



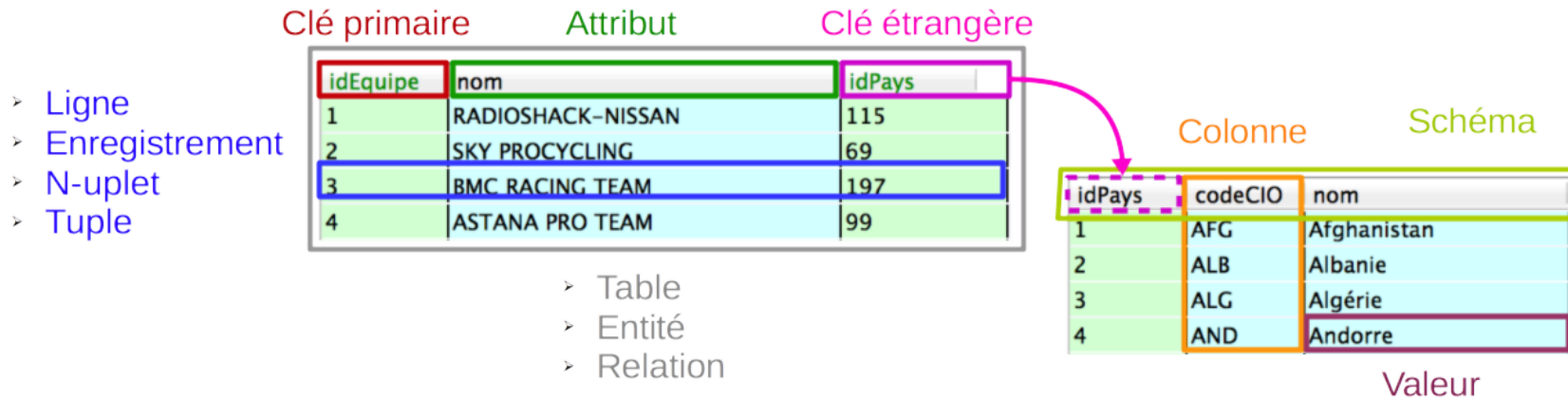
Conception d'une Base De Données (BDD) [2/2]

- Suggestion de logiciels pour réaliser des MCD :
 - AnalyseSI (→ formalisme entité-association) : gratuit
 - MagicDraw (→ diagramme de classes) : commercial
- Suggestions de logiciels pour réaliser des MLD :
 - MySQL Workbench : gratuit
 - PHPMyAdmin : gratuit



Vocabulaire associé aux BDDs

- Exemple : équipes du Tour de France et leur pays d'origine.



Plus de détails après...

Clés

- Servent d'identifiant pour définir ou se référer à des enregistrements des tables
- 2 types de clé :
 1. **clé primaire** (*primary key*)
 2. **clé étrangère** (*foreign key*)

Clés primaires

- Clé primaire identifie **de manière unique** un enregistrement.

Important

Parmi les attributs suivants d'une table qui renseigne sur des étudiants, quels sont ceux qui conviendraient pour une clé primaire ?

Nom, date de naissance, n° de sécurité sociale, Nom & prénom, n° INE (Identifiant National Étudiant).



La meilleure clé primaire :

Un nombre entier sans signification particulière :

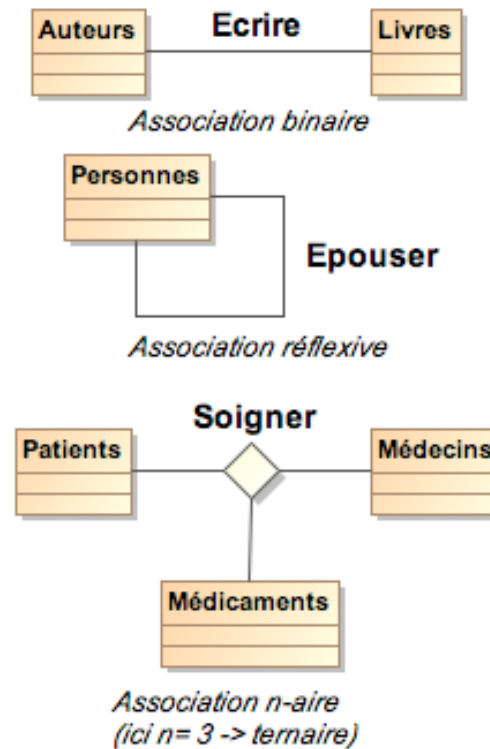
- Efficace lors des traitements
- Evite qu'on se pose des questions sur l'unicité ou non d'un attribut.

Clés étrangères

- **Clé étrangère** permet de se référer à un enregistrement d'une autre table.
- Objectif principal des clés étrangères : **éviter la duplication de l'information**
 - Gain de place
 - Participe au maintien de la cohérence des données
- Ex. : On renseigne les informations d'un département français dans une table (Nom, population, superficie...) et on se réfère ensuite à cette table à chaque fois qu'on a besoin d'une information liée aux départements plutôt que de la répéter à différents endroits.
 - Protège des fautes d'orthographe
Côte d'or ≠ Cote d'or
 - Facilite la mise à jour des données
Si le nom d'un département change (Côtes du Nord ⇒ Côtes d'Armor), il suffit de changer son nom dans la table des départements et non dans toutes les tables qui dupliqueraient l'information.

Associations

- Spécifient la nature du lien entre plusieurs tables de la BDD.
- Ex. :
 - Un auteur **écrit** des livres
 - Une personne **épouse** une autre personne.
 - Un médecin **soigne** des patients avec des médicaments
- Une table peut être reliée à :
 - 1 seule table ⇒ **association binaire**
 - elle-même ⇒ **association réflexive**
 - à plusieurs tables ⇒ **association n-aire**
- Concept d'association directement lié aux concepts des clés primaires et (surtout) clés étrangères.

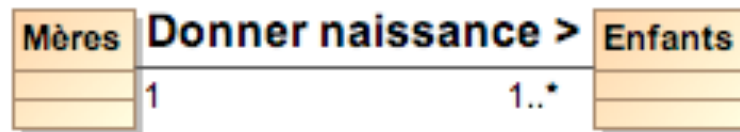


Multiplicités

- Précisent la nature de l'association entre les tables
- Représentées sous la forme «x...y» (comme en SysML).
 - «x» représente le **caractère facultatif (0) ou obligatoire (1...n)** de l'association
 - 1 femme **peut** avoir des enfants $\Rightarrow x = 0$
 - 1 article **doit** comporter une seule référence $\Rightarrow x = 1$
 - «y» représente le **caractère unique (1) ou multiple (n)** de l'association
 - 1 femme peut avoir **des** enfants $\Rightarrow y = n$
 - 1 article doit comporter **1 seule** référence $\Rightarrow y = 1$
- Multiplicités fréquentes :
 - 0...1
 - 1...1 $\Rightarrow$ noté 1
 - 0...* $\Rightarrow$ «*» signifie «plusieurs»
 - 1...*

Multiplicités

- Doivent figurer aux 2 extrémités d'une association
 - **1** mère donne naissance à **1 ou plusieurs** enfants
mais...
 - **1** enfant n'a qu'**1** seule mère.

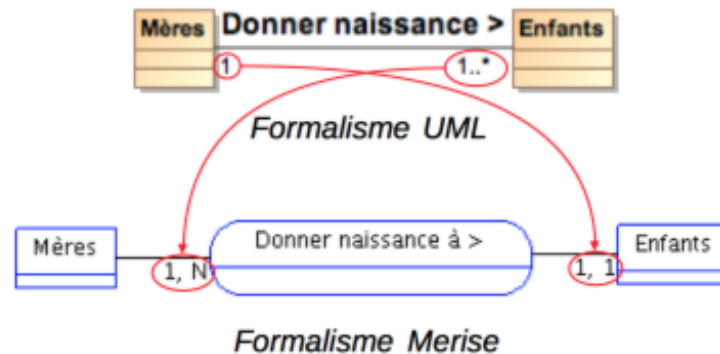


Multiplicité vs. Cardinalités

- On entend parfois parler de cardinalités : elles représentent le même concept mais sont souvent liées à 2 formalismes différents.
 - multiplicités ⇒ formalisme UML
 - cardinalités ⇒ formalisme entités-association (→ méthode Merise)

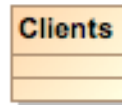
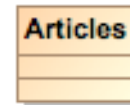
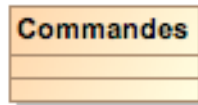


Au niveau de la représentation, les emplacements des cardinalités de Merise sur l'association sont inversés par rapport aux multiplicités UML.



Exercice

- Représenter les associations «Effectuer» et «Comporter» entre les entités «Clients», «Commandes», «Articles»
- Établir les multiplicités sachant que :
 - 1 client peut passer plusieurs commandes
 - 1 commande est associée exactement à un seul client
 - 1 commande est constituée d'au moins 1 article
 - 1 article ne fait pas forcément partie d'une commande



Exercice

- Interpréter le modèle suivant (tiré du livre «UML2 pour les bases de données» de C. Soutou) :



Noter qu'avec le formalisme UML, la clé primaire apparaît soulignée.

Associations binaires importantes

- 3 types d'association binaire importants :
 1. **1 à 1** (*one-to-one*)
 2. **1 à plusieurs** (*one-to-many*)
 3. **plusieurs à plusieurs** (*many-to-many*)
- Impact direct sur la structure des tables
- Découlent des **multiplicités maximales** des associations
- Ex. :
 - $0...1 \text{ ----- } 1 \Rightarrow$ association 1 à 1
 - $0...1 \text{ ----- } 0...n \Rightarrow$ association 1 à plusieurs
 - $0...n \text{ ----- } 0...n \Rightarrow$ association plusieurs à plusieurs



Ne pas confondre association et multiplicité.

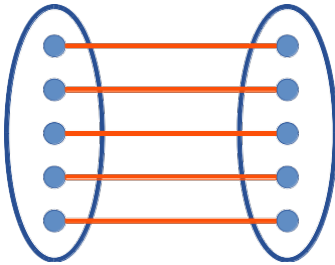
Une multiplicité concerne un seul côté de l'association et dans le cas d'une multiplicité 1 à plusieurs, elle se note d'une seule façon : $1...n$.

Une association est caractérisée par les 2 multiplicités présentes à ses 2 extrémités.

Une association 1 à plusieurs peut donc s'appliquer aux relations suivantes : $0...1 \text{ ----- } 0...n$, $1 \text{ ----- } 0...n$, $1 \text{ ----- } 1...n$.

L'association 1 à 1

- Pas très fréquente
- Chaque ligne d'une table est associée à exactement 1 seule ligne dans l'autre table.



- Permet :
 - de détailler un enregistrement
 - d'isoler certaines informations liées à un enregistrement dans le cas où celles-ci sont :
 - volumineuses
 - peu utilisées
- Association 1 à 1 entre 2 tables $\Rightarrow$ même nombre de lignes dans les 2 tables

Table « Auteurs »

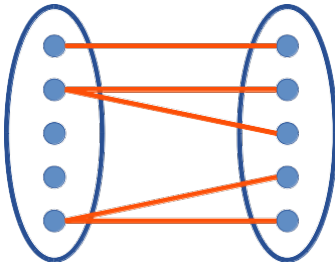
idAuteur	Nom	Prénom	idAuteurInfo
1	Steinbeck	John	1
2	Paasilinna	Arto	3
3	Zola	Emile	2

Table « Informations Auteurs »

idAuteurInfo	Nationalité	Date de Naissance	Photo
1	américaine	27/02/1902	
2	française	02/04/1840	
3	finlandaise	20/04/1942	

L'association 1 à plusieurs

- Très fréquente
- Chaque ligne de la 1^{ière} table est reliée à **0**, 1 ou plusieurs lignes de l'autre table



Chaque ligne de la 2^{ième} table est reliée à 1 ligne de la 1^{ière} table.

- Le nombre de lignes de la table «parent» ou *referenced* (ici «Auteurs») comporte souvent moins de lignes que la table «enfant» ou *referencing* (ici «Livres»)

Table « Auteurs »

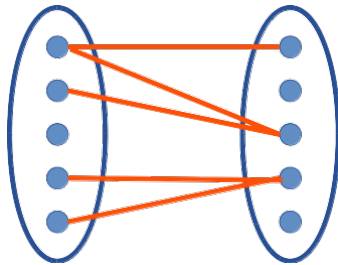
idAuteur	Nom	Prénom
1	Steinbeck	John
2	Paasilinna	Arto
3	Manvussa	Gérard

Table «Livres»

idLivre	Titre	idAuteur
1	Les raisins de la colère	1
2	Le lièvre de Vatanen	2
3	Le bestial serviteur du pasteur Huuskonen	2
4	Des souris et des hommes	1
5	La cavale du géomètre	2

L'association plusieurs à plusieurs [1/2]

- Très fréquente
- Chaque ligne de **chacune** des tables est reliée à **0**, 1 ou plusieurs lignes de l'autre table



- Ex : Un client peut acheter plusieurs livres ET un livre peut être acheté par plusieurs clients
- Nécessite la mise en place d'une 3^{ième} table («Clients_Livres»)
 - L'ensemble des 2 clés étrangères constitue la clé primaire de cette table

Table «Clients»

idClient	nom	prénom
1	Doe	John
2	Tarénase	Maguy

Table «Clients_Livres»

idClient	idLivre
1	1
2	2
2	1
1	5

Table «Livres»

idLivre	titre
1	Les raisins de la colère
2	Le lièvre de Vatanen
3	Le bestial serviteur du pasteur Huuskonen
4	Des souris et des hommes
5	La cavale du géomètre

L'association plusieurs à plusieurs [2/2]

- La table intermédiaire (appelée parfois «table associative ou de jointure») peut comporter des attributs
- Représentation sous forme d'une classe association en UML :

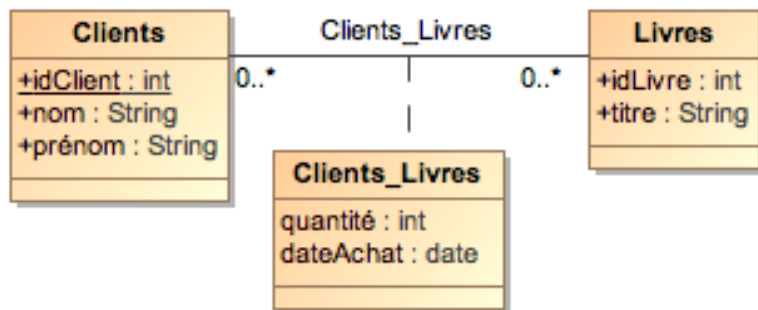


Table «Clients»

idClient	nom	prénom
1	Doe	John
2	Tarénase	Maguy

Table «Clients_Livres»

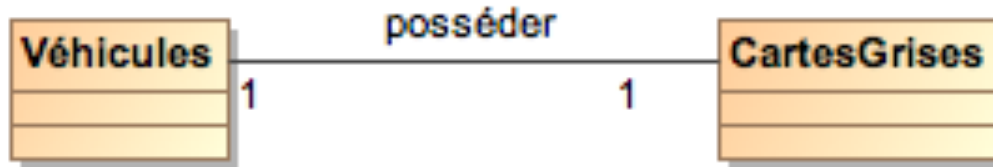
idClient	idLivre	quantité	dateAchat
1	1	2	19/01/2015
2	2	1	20/01/2015
2	1	1	20/01/2015
1	5	1	21/01/2015

Table «Livres»

idLivre	titre
1	Les raisins de la colère
2	Le lièvre de Vatanen
3	Le bestial serviteur du pasteur Huuskonen
4	Des souris et des hommes
5	La cavale du géomètre

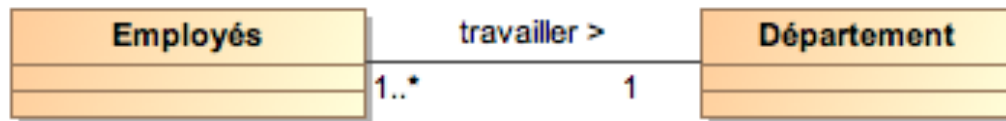
Exercice

- Soit l'association représentée ci-dessous :
 - quel est le type de l'association ?
 - identifier les tables «enfant» (*referencing*) et «parent» (*referenced*)
 - dans quelle table figurera la clé étrangère ?
 - en admettant que la table «parent» comporte 10 lignes, combien de lignes comportera la table «enfant» ? (moins, plus ou exactement le même nombre)



Exercice

- Soit l'association représentée ci-dessous :
 - quel est le type de l'association ?
 - identifier les tables «enfant» (*referencing*) et «parent» (*referenced*)
 - dans quelle table figurera la clé étrangère ?
 - quelle table a plus de chances de comporter moins de lignes ?



Exercice

- Modéliser en UML l'association suivante :
«Une ville est reliée à d'autres villes par des voies de circulation possédant un type (nationale, autoroute, départementale), un kilométrage.»

SGBD

- SGBD = **S**ystème de **G**estion de **B**ase de **D**onnées
(*DBMS : Data Base Managment System*)
- SGBD**R** = SGBD pour BDD **R**elationnelle
(*RDBMS : Relational Data Base Managment System*)
- Ensemble de programmes qui assurent la gestion et l'accès aux BDDs



Rôles du SGBD :

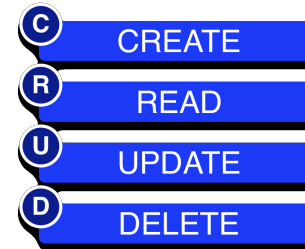
1. Centraliser l'accès aux données ;
2. Fournir une interface de haut niveau vers les données ;
3. Sécuriser l'accès aux données ;
4. Garantir la cohérence des données (intégrité, non-redondance).

Quelques SGBDRs



Généralités sur SQL

- SQL = **S**tructured **Q**uery **L**anguage
- Langage de requête «relativement» standardisé pour communiquer avec un SGBDR
 - Utilisé par tous les SGBDRs (MySQL, PostgreSQL, SQLite...)
- Fournit les 4 opérations essentielles de manipulation de données
 - Ajout, recherche, modification, destruction
- Syntaxe proche de l'anglais
 - Nom initial : *SEQUEL* → *Structured English as QUery Language*
 - On demande ce qu'on veut sans expliquer au système comment le faire
- Requêtes SQL :
 - s'appliquent à des tables
 - **retournent des tables**



Catégories d'instructions SQL [1/4]

4 catégories d'instructions :

1. LDD : Langage de Définition des Données

- *DDL : Data Definition Language*
- création / modification / suppression de tables
- instructions : **CREATE**, **ALTER**, **DROP**

Catégories d'instructions SQL [2/4]

4 catégories d'instructions :

1. LDD : Langage de Définition des Données
2. LMD : Langage de Manipulation des Données
 - *DML : Data Manipulation Language*
 - consultation / ajout / modification / suppression d'enregistrements
 - instructions : **SELECT, INSERT, UPDATE, DELETE**
 - partie du langage qu'on utilise le plus souvent

Catégories d'instructions SQL [3/4]

4 catégories d'instructions :

1. LDD : Langage de Définition des Données
2. LMD : Langage de Manipulation des Données
3. LCD : Langage de Contrôle des Données
 - *DCL : Data Control Language*
 - ajout / suppression de privilèges
 - instructions : **GRANT**, **REVOKE**

Catégories d'instructions SQL [4/4]

4 catégories d'instructions :

1. LDD : Langage de Définition des Données
2. LMD : Langage de Manipulation des Données
3. LCD : Langage de Contrôle des Données
4. LCT : Langage de Contrôle des Transactions
 - *TCL : Transaction Control Language*
 - gestion des transactions (séquences d'ordres SQL indissociables)
 - instructions : **COMMIT**, **ROLLBACK**, **SET TRANSACTION**

Fin

