

01 – La base de données

# SI3 – Exploitation des données

# Définition

- Abrégé BD ou DB
- Ensemble structuré et organisé permettant le stockage de grandes quantités d'infos
- But : faciliter l'exploitation
  - Ajout
  - Mise à jour
  - Recherche
  - Suppression
- Contient les données informatisées du SI

# Logiciels correspondants

- SGBDR:
  - Oracle, SQL Server, Access
  - Mysql
  - PostgreSQL...
- SGBDOO
  - ObjectStore
  - Db4o
  - O2...

# Objectifs d'un SGBDR

- Indépendances physique et logique vis-à-vis du programme
- Manipulation de données par des utilisateurs non informaticiens
- Administration simplifiée grâce aux outils distribués
- Cohérence des données (contraintes)
- Partage des données entre plusieurs applications
- Sécurisation des données (Droits d'accès)

# Les tables

- Tables reliées entre elles (BDR)
- Ligne = enregistrement = occurrence
- Intitulé des colonnes = champ = attribut
- Intersection ligne/colonne = valeur définit par:
  - Nom
  - Type
  - Caractère obligatoire ou facultatif
  - Plage de valeur
  - Valeur par défaut
  - ...

# Les types de données

- A préciser à la création d'une table
- Nécessaire au SGBD pour l'occupation mémoire
- Dépend du SGBD utilisé

# Les types MySQL : Entiers

- Tinyint, smallint, mediumint, int, bigint

Type	Octets	De	A
TINYINT	1	-128	127
SMALLINT	2	-32768	32767
MEDIUMINT	3	-8388608	8388607
INT	4	-2147483648	2147483647
BIGINT	8	-9223372036854775808	9223372036854775807

- Possibilité d'ajouter Unsigned

# Les types MySQL : Réels

- Float
- Double ( M,D )
- Decimal ( M,D )
- M : Taille totale d'affichage (ex: 56,3456 -> M=6)
- D : Nombre de chiffre après la virgule
- Arguments non obligatoires



# Les types MySQL : Du texte

- Stocke des chaînes de caractère (donc aussi des nombres ou des dates!)
- Taille fixe :
  - char : 1 seul caractère
  - char(n) : une chaîne de n caractères (n inférieur à 255, sinon TinyText)
- Taille variable:
  - varchar(m) : m caractères maxi (m inférieur à 255 sinon TEXT)

# Les types MySQL : De + longs textes

- Types particuliers pour de + longues zone de texte:
  - Tinytext : maxi de 255 caractères (256 B)
  - Text: maxi de 65535 caractères (65 KB)
  - Mediumtext: maxi de 16777215 car. (16 MB)
  - Longtext : maxi de 4294967295 car. (4 GB)

# Les types MySQL : les dates

- Pour permettre le stockage de données temporelles:
  - Date: une date de '1000-01-01' à '9999-12-31'
  - Datetime: une heure un jour donné de '1000-01-01 00:00:00' à '9999-12-31 23:59:59'
  - Time: une heure de 00:00:00 à 23:59:59
  - Year: L'année de 1901 à 2155
  - Timestamp: nbr de secondes depuis le 1/1/1970

# Les types MySQL : les objets binaires

- Permet de stocker tout type de fichier.
- Exemple : des sons, des images, des vidéos, des documents Word, programme...
- Stockage binaire dans le BLOb :
  - Binary Large Object
- TINYBLOb : 256 B max.
- BLOb : 65 KB max.
- MEDIUMBLOb : 16 MB max.
- LONGBLOb : 4 GB max.

# Les types MySQL : Données spatiales

- Depuis MySQL 4.1, pour SIG
- POINT : un point
- LINESTRING : une ligne
- POLYGON : Polygone avec anneau extérieur et intérieur
- MULTIPOINT : un groupe de point
- MULTILINESTRING : un groupe de ligne
- MULTIPOLYGON : un groupe de polygone
- GEOMETRYCOLLECTION : un groupe géométrique

# Capacités de stockage des colonnes numériques

Type de colonne	Espace requis
TINYINT	1 octet
SMALLINT	2 octets
MEDIUMINT	3 octets
INT, INTEGER	4 octets
BIGINT	8 octets
FLOAT (p)	4 if $X \leq 24$ or 8 if $25 \leq X \leq 53$
DOUBLE	4 octets
DOUBLE PRECISION, REAL	8 octets
DECIMAL (M, D)	$M+2$ octets si $D > 0$ , $M+1$ octets si $D = 0$ ( $D+2$ , si $M < D$ )

# Capacités de stockage des colonnes de temporelles

Type de colonne	Espace requis
DATE	3 octets
DATETIME	8 octets
TIMESTAMP	4 octets
TIME	3 octets
YEAR	1 octet

# Capacités de stockage des colonnes de texte et binaire

Type de colonne	Espace requis
<code>CHAR (M)</code>	M octets, $1 \leq M \leq 255$
<code>VARCHAR (M)</code>	L+1 octets, avec $L \leq M$ et $1 \leq M \leq 255$
<code>TINYBLOB, TINYTEXT</code>	L+1 octets, avec $L < 2^8$
<code>BLOB, TEXT</code>	L+2 octets, avec $L < 2^{16}$
<code>MEDIUMBLOB, MEDIUMTEXT</code>	L+3 octets, avec $L < 2^{24}$
<code>LONGBLOB, LONGTEXT</code>	L+4 octets, avec $L < 2^{32}$
<code>ENUM('valeur1', 'valeur2', ...)</code>	1 ou 2 octets, suivant le nombre d'éléments de l'énumération (65535 au maximum)
<code>SET('valeur1', 'valeur2', ...)</code>	1, 2, 3, 4 ou 8 octets, suivant le nombre de membres de l'ensemble (64 au maximum)

Enum -> 1 seule élément à la fois

Set -> Plrs éléments à la fois



# Les contraintes

- Contrainte de domaine:
  - NOT NULL
  - Plage de valeur (check pas implémentée dans mysql)
- Contrainte d'intégrité d'entité:
  - PRIMARY KEY
- **Contrainte d'intégrité référentielle** liée aux clés étrangères.

# La contrainte d'intégrité référentielle

## ■ Exemple :

### ■ Une table CLIENT contenant 3 occurrences:

- (1,Toto,1987)
- (2,Titi,1991)
- (3,Tata,1990)

### ■ Une table COMMANDE contenant 2 occurrences:

- (1,22 juin 2011,1)
- (2,24 juin 2011,4)

Problème : Le client identifié  
par la clé 4 n'existe pas

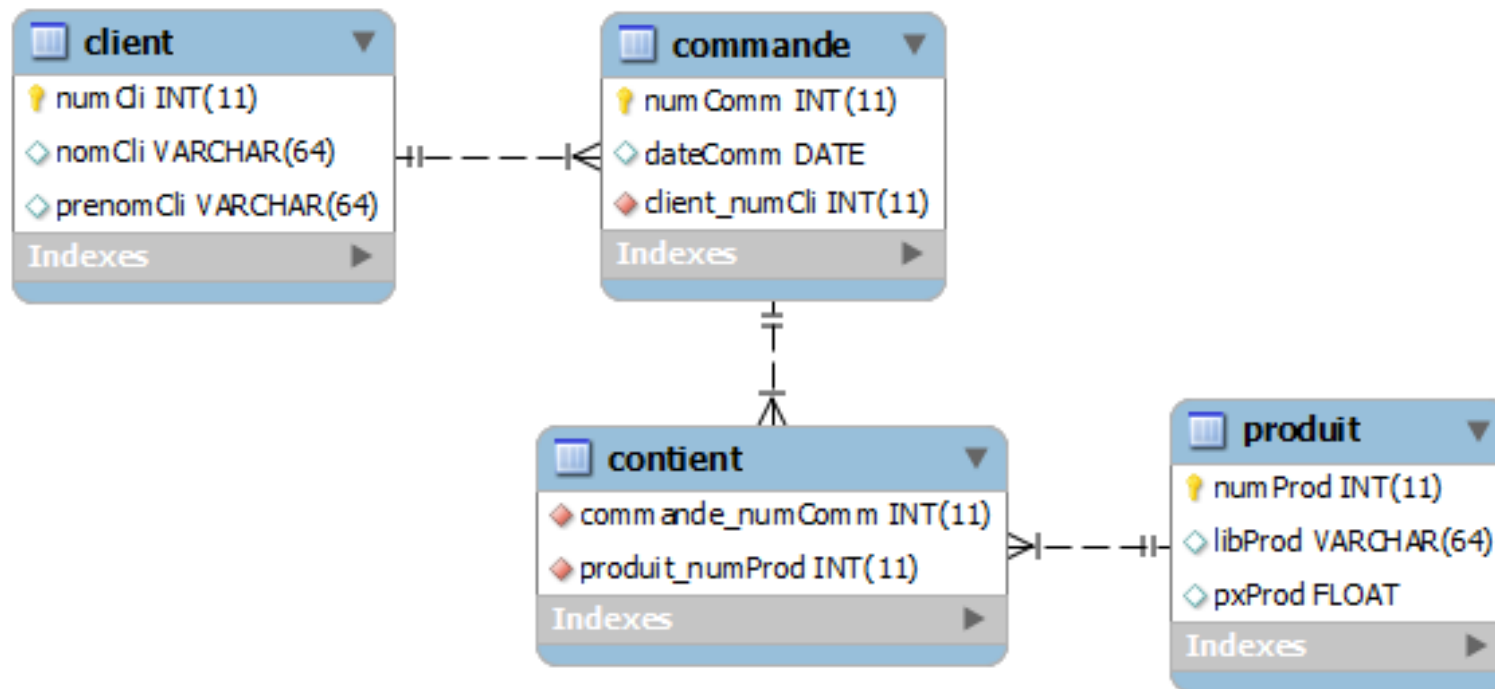
- La valeur d'une clé étrangère doit exister dans la table à laquelle elle fait référence

# La représentation du modèle relationnel

- Afin de représenter la structure de la base de données:
  - Représentation textuelle:
    - **CLIENT**(numCli, nomCli, prenomCli)
    - **COMMANDE**(numComm, dateComm, #numCli)
    - **PRODUIT**( numProd, libProd, pxProd)
    - **CONTIENT**(#numComm, #numProd)

# La représentation du modèle relationnel

- Afin de représenter la structure de la base de données:
  - Représentation graphique MySQL:



# FIN

- Questions

