# Cours 4 et 5: Le langage SQL

- Origine
- Définition des structures
- Définition des contraintes
- Modification des structures
- Création d'index
- Création de vues
- Définition d'un schéma



Parallèle avec le monde objet

Philippe LAHIRE - Cours Base de Données L2I

Cours 4

Janvier 2012

83

#### **Origine**

#### Structured Query Language

- caractéristiques des langages déclaratifs
- origine : IBM, System R, milieu des années 70
- -implémenté dans de nombreux SGBD

#### Plusieurs versions:

- SQL1 initial : ANSI\* —1986
- SQL1 avec intégrité référentielle, ANSI —1989
- SQL2 ANSI —1992
- SQL3 ANSI 1999 incorpore la notion d'objet

Philippe LAHIRE – Cours Base de Données L21

Cours 4

Janvier 2012

<sup>\*</sup> ANSI = American National Standard Institute

# Caractéristiques de SQL

- Fonctionnalités :
  - Définition des objets de la base de données (LDD)
  - Manipulation de données (LMD)
  - Contrôle des accès aux données
  - Gestion de transactions
- Utilisé par : DBA, développeurs, quelques utilisateurs



Parallèle avec mySQL

Philippe LAHIRE - Cours Base de Données L2I

Cours 4

Janvier 2012

85

# **Principales Instructions**

- Définitions (LDD)

  CREATE, DROP, ALTER
- Mises à jour (LMD)

  INSERT, UPDATE, DELETE
- Interrogations (LMD) SELECT



- Contrôle d'accés aux données GRANT, REVOKE
- Gestion de transactions COMMIT, ROLLBACK

Philippe LAHIRE – Cours Base de Données L21

Cours 4

Janvier 2012

#### Consultation des données

- Hypothèse:
  - un schéma de base de données est créé
  - Une base de données a été remplie
- La création a été faite par une interface QBE (mySQL)
- On expérimente la consultation avec SQL
- On expérimentera la création du schéma et le remplissage de la base avec SQL plus tard
  - Langage algébrique en SQL
  - Requêtes mono et multi table(s)

Philippe LAHIRE - Cours Base de Données L2I

Cours 4

Janvier 2012

87

#### Exemple pour les requêtes

#### CLIENT

numéro	nom	adresse	numéro_téléphone
101	Durand	NICE	0493456743
108	Fabre	PARIS	NULL
110	Prosper	PARIS	NULL
125	Antonin	MARSEILLE	NULL

#### PRODUIT

référence	marque	Prix HT
153	BMW	1000
589	PEUGEOT	1800
158	TOYOTA	1500

#### VENTE

numéro	référence_produit	numéro_client	date
00102	153	101	12/10/04
00809	589	108	20/01/05
11005	158	108	15/03/05
12005	589	125	30/03/05

Philippe LAHIRE - Cours Base de Données L21

Cours 4

Janvier 2012

# Opérateur Projection

SELECT xx yy: définition du format du résultat + spécifier la cible

• Afficher le nom et l'adresse des clients

**SELECT** nom, adresse

FROM Client;

• Afficher toutes les informations des clients

**SELECT** \*

FROM Client;

Philippe LAHIRE - Cours Base de Données L21

Cours 4

Janvier 2012

89

Elimination des doublons

• Afficher l'adresse de tous les clients

**SELECT** adresse

**SELECT ALL** adresse

FROM Client;

FROM Client;

Afficher toutes les adresses existantes (sans doublons)

SELECT DISTINCT adresse FROM Client;

Cherit,

Par défaut

· Projection

Relationnel: On doit pouvoir distinguer chaque tuple!

SQL : le résultat peut ne pas être une relation

Philippe LAHIRE – Cours Base de Données L2I

Cours 4

Janvier 2012

# Opérateur Sélection

```
+ spécifier la cible + where expression : définition des tuples du résultat
```

• Quels sont les clients dont l'adresse est Paris

```
SELECT *
FROM Client
WHERE adresse = \Paris';
```

• Quels sont les produits dont le prix TTC est supérieur à 1000

```
SELECT *
FROM Produit
WHERE prix_HT + prix_HT * 0.195 > 1000;
```

Philippe LAHIRE – Cours Base de Données L2I Cours 4 Janvier 2012 91

# Opérations possibles (mySQL) • Booléennes and, or, xor, =, !=, <, >, <=, >= • Arithmétiques

+, - opérateurs unaires
• Fonctions numériques

abs, log, cos, sin, mod, power ...

Arithmétiques sur date

• Fonctions sur chaînes length, concat, ...

Dans Select: attribut calculé (résultat)

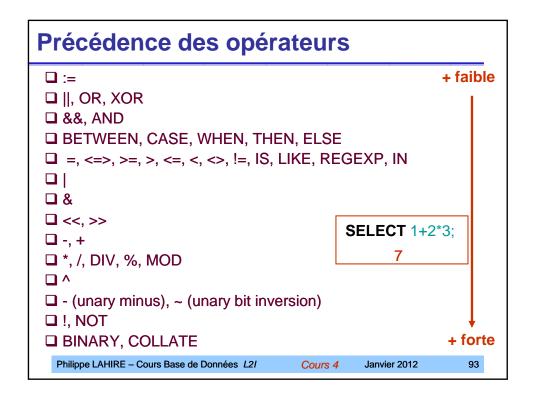
Dans Where: participer à la sélection

Philippe LAHIRE – Cours Base de Données L2I

Cours 4

Janvier 2012

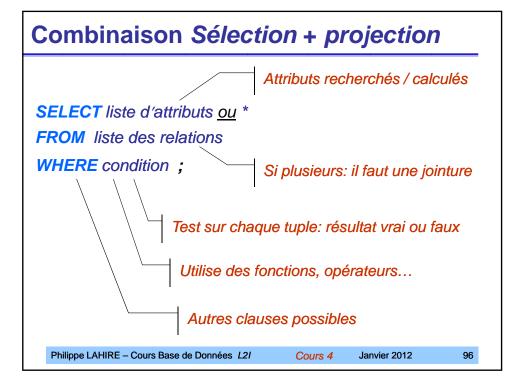
**Un grand nombre** 





# **Utilisation des opérateurs (Chaînes)**

```
• SELECT CONCAT ('My', 'S', 'QL');
                                       'MySQL'
• SELECT CHAR_LENGTH ('MySQL');
• SELECT LOCATE ('bar', 'foobarbar');
• SELECT LOCATE ('bar', 'foobarbar', 5); 7
                                               'QuWhattic'
• SELECT INSERT ('Quadratic', 3, 4, 'What');
• SELECT LOWER ('MySQL');
                                 'mysql'
• SELECT SUBSTRING ('Quadratically',5,6);
                                              'ratica'
• SELECT 'David!' LIKE 'David ';
                                                       et %
• SELECT 'David!' LIKE '%D%v%':
                                     1
                                                 0 (false), 1 (true)
• SELECT STRCMP (S1, S2);
                                    -1.0.1
                                                             95
  Philippe LAHIRE – Cours Base de Données L2I
                                     Cours 4
                                              Janvier 2012
```



# Y a-t-il des produits dont le nom est « XBOX » SELECT \* FROM Produit WHERE nom = 'XBOX';

- Quels sont les ventes réalisés il y a plus de 30 jours?
   SELECT \*
- FROM Vente

WHERE CURRENT\_DATE () > 30 + date;

• Quels sont les ventes faites après le 1er janvier 2007?

**SELECT** \*

FROM Vente

WHERE date > DATE ('2007-01-01');

Philippe LAHIRE - Cours Base de Données L2I

Cours 4

Janvier 2012

97

# Requêtes simples (2)

• Quels sont les ventes dont le montant HT est entre 1000 et 3000 euros et dont le client n'est pas le numéro 101?

**SELECT** \*

**FROM** Vente

WHERE (prix\_ht between 1000 and 3000) and (numero != 101);

 Quels sont les clients dont le nom est soit Prosper, soit Durand, soit Anthonin ?

**SELECT**\*

FROM Client

WHERE nom in ('Prosper', 'Durand, 'Anthonin');

Philippe LAHIRE – Cours Base de Données L21

Cours 4

Janvier 2012

#### Requêtes simples (3)

Quels sont les clients dont le nom commence par 'P'

**SELECT** \*

FROM Employe

WHERE nom LIKE 'P%';

 Quels sont les clients dont le nom commence par 'P' et a un 'S' comme 4ème lettre

**SELECT** \*

FROM Client

WHERE nom LIKE 'P\_S%';

Philippe LAHIRE - Cours Base de Données L21

Cours 4

Janvier 2012

99

Requêtes et valeurs nulles (1)

• Quels sont les ventes dont la date de réalisation est inconnue?

**SELECT**\*

FROM Vente

WHERE date is null;

Attention:

WHERE date = NULL

Whatever comparé avec NULL → Ni vrai ni faux

Ignore les valeurs NULL **COUNT, MIN, SUM** 

→ sauf COUNT (\*)

Toute opération appliquée à NULL donne pour résultat NULL

Philippe LAHIRE – Cours Base de Données L21

Cours 4

Janvier 2012

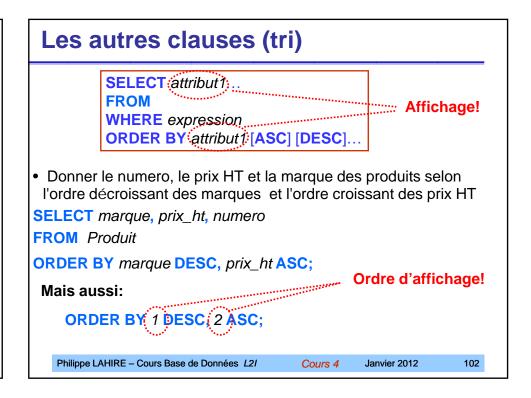
Ou: IS NOT NULL

#### Requêtes et valeurs nulles (2)

```
SELECT nom
FROM Produit
WHERE prix_ht > 1000;

SELECT nom
FROM Produit
WHERE prix_ht > 1000 or prix_ht <= 2500;

Que se passe-t-il si certains produits
ont un prix HT inconnu?
```



#### Requêtes multi- tables (Opérateur *Jointure*)

#### Deux points de vue:

- exécuter des boucles imbriquées (une table par boucle) appliquer la clause WHERE dans les boucles
- calculer le produit cartésien (une nouvelle table)
   Appliquer la clause WHERE sur chaque ligne

Donner le nom d'un produit et le montant de la vente

SELECT Produit.nom, Vente.prix\_HT
FROM Produit, Vente
WHERE Vente.reference\_produit = Produit.numero;
Critères de jointure

#### Il faut joindre les tables (jointure)

Philippe LAHIRE - Cours Base de Données L2I

Cours 4

Janvier 2012

103

#### Définition de la jointure

**SELECT** Client.nom, Vente.prix\_ht FROM Client, Vente;

- → tous les tuples (*nom*, *prix\_ht*)
- → nom est un nom de client et prix\_ht est un prix de vente

Pas de critère de jointure → Produit cartésien Intérêt?



Mots-clés pour exprimer le critère de sélection

Philippe LAHIRE – Cours Base de Données L21

Cours 4

Janvier 2012

# • Donner la marque des produits dont le prix HT est supérieur à celui d'une BMW SELECT Produit.marque FROM Produit Produit\_reference, Produit WHERE Produit\_reference.marque = 'BMW' AND Produit.prix\_HT > Produit\_reference.prix\_HT; Ou de même prix HT: SELECT Produit.marque FROM Produit Produit\_reference, Produit WHERE Produit\_reference.marque = 'BMW' AND Produit.prix\_ht = Produit\_reference.prix\_ht AND Produit.marque != Produit\_reference.marque;

Cours 4

Janvier 2012

105

Philippe LAHIRE - Cours Base de Données L21

#### **SQL** et les différentes jointures SELECT ... FROM [INNER] JOIN Jointure interne ON <condition de jointure> SELECT ... FROM LEFT | RIGHT | FULL OUTER JOIN ON condition de Jointure externe jointure dans mysql pas reconnu SELECT ... FROM NATURAL JOIN Jointure naturelle [USING <noms de colonnes>] SELECT ... FROM CROSS JOIN Jointure croisée dans mysql = INNER JOIN SELECT ... FROM UNION JOIN Jointure d'union Philippe LAHIRE - Cours Base de Données L21 Janvier 2012 Cours 4 106

#### **Quelques jointures**

• Donner le nom des client et le prix HT

SELECT Client.nom, Vente.prix\_ht

FROM Vente, Client

Jointure naturelle

WHERE Vente.numero\_client = Client.numero

 Donner la marque des produits dont le prix HT est supérieur à celui d'une BMW

**SELECT** Produit.marque

**Theta Jointure** 

FROM Produit Produit\_reference, Produit

WHERE Produit\_reference.marque = 'BMW' AND

Produit.prix\_ht > Produit\_reference.prix\_ht ;

Philippe LAHIRE - Cours Base de Données L21

Cours 4

Janvier 2012

107

#### **Quelques jointures (compléments)**

#### Que deviennent les tuples non sélectionnés de R1 ou R2?

- dans la jointure (interne **INNER JOIN**) les tuples qui ne peuvent pas être joints sont éliminés du résultat
- dans la jointure externe (OUTER JOIN) les tuples qui ne peuvent pas être joints sont conservés dans le résultat

Pour les tuples de la relation de gauche (R1) et / ou de droite (R2)

Défaut

R1 FULL OUTER JOIN R2 : Remplit R1.\* et R2.\*

R1 LEFT OUTER JOIN R2 : Remplit R1.\*

R1 RIGHT OUTER JOIN R2: Remplit R2.\*

avec **NULL** si nécessaire.

Philippe LAHIRE – Cours Base de Données L21

Cours 4

Janvier 2012

#### Exercice (1)

#### **Relations**:

- Journal (<u>code-i</u>, titre, prix, type, périodicité)
- Dépôt (<u>no-dépôt</u>, nom-dépôt, adresse)
- Livraison (<u>no-dépôt</u>, <u>code-i</u>, <u>date-liv</u>, quantité-livrée)

#### Requêtes : donner...

- le prix des journaux livrés le 15/01/07 ?
- tous le nom des hebdomadaires reçus par le dépôt de Paris..
- les titre des journaux livrés à Nice.
- le nom des dépôts qui reçoivent des hebdomadaires dont la quantité livrée excède 100.

Philippe LAHIRE - Cours Base de Données L2I

Cours 4

Janvier 2012

109

# Exercice (2)

#### **SELECT\***

FROM LIVRAISON RIGHT JOIN (DEPOT, JOURNAL)

ON (LIVRAISON.no-depot = DEPOT.no-depot AND

LIVRAISON.code-j = JOURNAL.code-j)

Toutes les lignes de DEPOT et JOURNAL seront présentes Avec éventuellement rien pour la partie livraison

#### **SELECT**\*

FROM LIVRAISON, DEPOT, JOURNAL

WHERE LIVRAISON.no-depot = DEPOT.no-depot AND

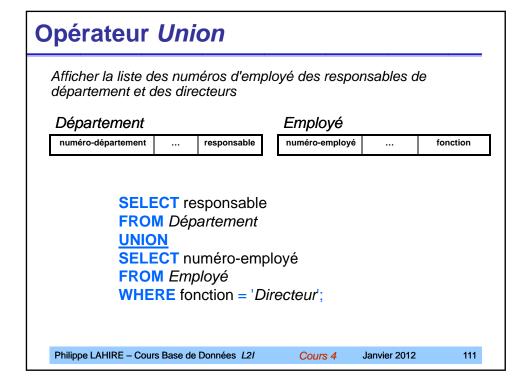
LIVRAISON.code-j = JOURNAL.code-j)

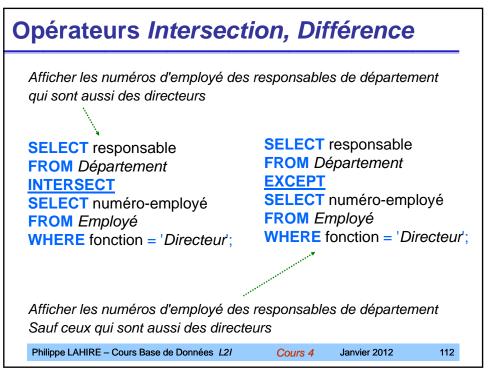
Seulement les lignes de DEPOT et JOURNAL qui correspondent à une livraison

Philippe LAHIRE – Cours Base de Données L21

Cours 4

Janvier 2012





# Requêtes (Fonctions statistiques)

- SUM (nom d'attribut)
- COUNT(\*)
- COUNT(DISTINCT nom d'attribut)
- MAX (nom d'attribut)
- MIN (nom d'attribut)
- AVG (nom d'attribut)
- AVG (DISTINCT nom d'attribut)

#### Dans les clauses :



- SELECT
- HAVING TO

Philippe LAHIRE - Cours Base de Données L21

Cours 4

Janvier 2012

113

# **Clause Group by**

**SELECT** attributs recherchés

**FROM** liste des relations

WHERE condition

**GROUP BY** attributs de regroupement

[HAVING condition sur le groupe];

**SELECT** COUNT (\*)

**FROM** Produit

**GROUP BY** marque

SELECT AVG (prix\_ht), nom

FROM Produit

**GROUP BY** marque

Philippe LAHIRE – Cours Base de Données L21

SELECT COUNT (\*)

FROM Produit

WHERE marque <> 'BMW'

**GROUP BY** marque

HAVING AVG (prix\_ht) > 10.5

Cours 4 Janvier 2012

#### **Utilisation des fonctions statistiques**

- Donner la moyenne des prix HT et les prix min. et max.
   SELECT AVG (prix\_ht), MIN (prix\_ht), MAX (prix\_ht)
   FROM Produit
- Même chose mais par marque
   SELECT AVG (prix\_ht), MIN (prix\_ht), MAX (prix\_ht)
   FROM Produit

GROUP BY marque

Même chose mais par marque si la moyenne > 10,5
 SELECT AVG (prix ht), MIN (prix ht), MAX (prix ht)

**FROM** Produit

**GROUP BY** margue **HAVING** AVG (prix ht) > 10.5

Philippe LAHIRE - Cours Base de Données L2I

Cours 4

Janvier 2012

115

#### Requêtes imbriquées (1)

SELECT liste d'attributs recherchés

FROM liste des relations

WHERE bloc SFW dans la condition

<u>Intérêt</u>: indiquer qu'un attribut doit prendre ses valeurs dans une liste de valeurs définies par un autre bloc *SWF* 

- Itérations imbriquées
- autre façon de faire certaines formes de jointure
- Sous-requêtes indépendantes ou pas

Philippe LAHIRE – Cours Base de Données L21

Cours 5

Janvier 2012

# Requêtes imbriquées (2)

Principaux connecteurs: = != > < IN EXISTS ANY ALL

**EXISTS** *R* : retourne <u>TRUE</u> si *R* n'est pas vide, <u>FALSE</u> sinon

t IN R: retourne TRUE si t appartient à R, FALSE sinon

valeur  $\underline{comp}$  **ANY** R: retourne  $\underline{TRUE}$  si la comparaison avec  $\underline{au}$  **moins un** des tuples de  $\underline{R}$  renvoie  $\underline{TRUE}$ 

valeur <u>comp</u> ALL R: retourne <u>TRUE</u> si la comparaison avec **tous** les tuples de **R** renvoie <u>TRUE</u>

Philippe LAHIRE – Cours Base de Données L2I Cours 5 Janvier 2012

# Sous-requêtes indépendantes

Quelles sont les références produit dont le prix HT est supérieur au prix HT moyen des produits ?

**SELECT** référence

**FROM** Produit

WHERE Produit.prix\_ht > (SELECT AVG(P.prix\_ht) FROM Produit P);



117

Le bloc SFW imbriqué peut être évalué séparément du bloc principal

Philippe LAHIRE – Cours Base de Données L2I

Cours 5 Ja

Janvier 2012

# Sous-requêtes indépendantes (1)

Quels sont les marques de produits vendus le 1/1/2007 ?

**SELECT** marque

FROM Produit, Vente

WHERE Vente.ref-produit = Produit.reférence

AND date = 1/1/2007;

<u>ou</u>

**SELECT** Produit.marque

Jointure ou requête imbriquée

FROM Produit

WHERE Produit.référence IN

(SELECT Vente.ref-produit

**FROM** Vente

**WHERE** date = 1/1/2007);

Philippe LAHIRE - Cours Base de Données L2I

Cours 5

Janvier 2012

119

# Sous-requêtes indépendantes (2)

Quelles sont les marques de produits qui n'ont pas le prix HT le plus élevé ?

**SELECT** marque

FROM Produit < SELECT max (P.prix\_ht) ...

WHERE Produit.prix-ht < ANY

(SELECT P.prix-ht FROM Produit P;)

Quelles sont les marques de produits qui ont le prix HT le plus élevé ?

**SELECT** marque

> SELECT max (P.prix ht) ...

**FROM** Produit

WHERE Produit.prix\_ht >= ALL

(SELECT P.prix\_ht FROM Produit P;)

Philippe LAHIRE – Cours Base de Données L2I

Cours 5

Janvier 2012

#### Sous-requêtes corrélées (1)

Quels produits dont le prix HT est supérieur au prix HT moyen des produits de la même marque ?

**SELECT** *Produit*.référence

FROM Produit

WHERE Produit.prix-ht > (SELECT AVG(P.prix-ht)

**FROM** *Produit P* 

**WHERE** *Produit*.marque = *P*.marque);)

Le bloc *SFW* principal et le bloc *SFW* imbriqué doivent être évalués simultanément

Philippe LAHIRE - Cours Base de Données L2I

Cours 5

Janvier 2012

121

#### Sous-requêtes corrélées (2)

Quels sont les marques dont aucun produit n'a été vendu le 1/01/2007 ?

**SELECT** *Produit*.marque

**FROM** *Produit* 

WHERE NOT EXISTS

(SELECT Vente.ref-produit

**FROM** Vente, Produit P

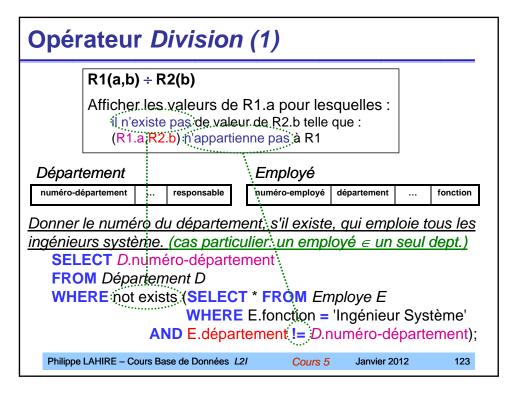
WHERE Vente.ref-produit = P.référence and P.marque = Produit.marque and

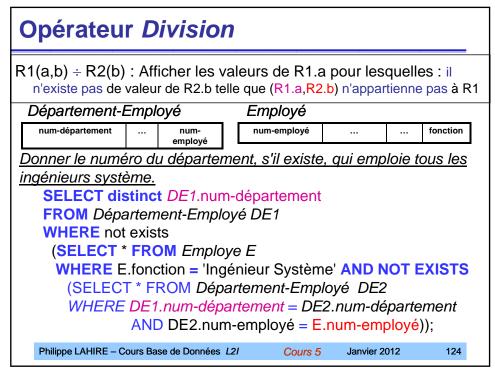
date = 1/01/2007;)

Philippe LAHIRE – Cours Base de Données L2I

Cours 5

Janvier 2012





# Opérateur *Division*

```
SELECT distinct département
FROM EMPLOYE
WHERE fonction = 'Ingénieur Système';
GROUP BY département
HAVING count(numéro-employé) =
    (SELECT count(numéro-employé) FROM EMPLOYE E)
    WHERE E.fonction = 'Ingénieur Système');
SELECT *
FROM EtudiantUE
GROUP BY etudiant
HAVING count(uniteValeur) =
    (SELECT count(code) FROM UE);

Philippe LAHIRE - Cours Base de Données 121

Janvier 2012

125
```



#### Manipulation des structures de données

- Schéma logique / Base de données
  - → CREATE/DROP SCHEMA/DATABASE ...
- Schéma de tables (relations) et de leur contenu
  - → CREATE/ALTER/DROP TABLE ...
- Définition des contraintes qui assurent des contrôles sur l'intégrité des données
- Mais encore:
  - les index
  - les utilsateurs et les privilèges

Philippe LAHIRE - Cours Base de Données L21

Cours 5

Janvier 2012

127

#### Création/Suppression d'un schéma

CREATE (DATABASE | SCHEMA) [IF NOT EXISTS] db name DROP DATABASE [IF EXISTS] db name

> Pas de message d'erreur si la BD

- Dans la norme SQL: création de schéma • MySQL: schema = database
- Il faut avoir les droits de le faire (administrateur?)
- Création d'un compte s'il n'existe pas
- Association des droits

**CREATE DATABASE** *lahire06-07* 

**GRANT ALL PRIVILEGES** 

ON lahire06-07.\*

TO ' lahire06-07'@localhost

IDENTIFIED BY ' lahire06-07';

Philippe LAHIRE - Cours Base de Données L21

Cours 5

Janvier 2012

**DROP DATABASE** *lahire06-07* 

Attention!

existe!

128

# Types de la norme ANSI

Philippe LAHIRE – Cours Base de Données L2I

INTEGER	Entiers relatifs	4 octets
SMALLINT	Entiers relatifs	2 octets
BIGINT	Entiers relatifs	8 octets
FLOAT	Flottants	4 octets
DOUBLE	Flottants	8 octets
REAL	Flottants	4 ou 8 octets
NUMERIC(N,D)/ DECIMAL(N,D)	Décimaux à précision fixe	N octets

CHAR(M)	Chaînes de longueur fixe	M octets
VARCHAR(M)	Chaînes de longueur variable	au plus M octets
BIT VARYING	Chaînes d'octets	longueur de la chaîne
DATE	date(hour, mois, an)	4 octets
TIME	heure(h, mn, sh)	4 octets
DATETIME	date et heure	8 octets
YEAR	année	2 octets

Cours 5

Janvier 2012

# Création et gestion des relations

• Structures de données (relations, attributs, tuples)

relation  $\longrightarrow$  table attribut  $\longrightarrow$  column tuple  $\longrightarrow$  row

• Instructions sur les relations

CREATE TABLE créer une relation

**DROP TABLE** supprimer une relation

**ALTER TABLE** modifier la structure d'une relation

Philippe LAHIRE – Cours Base de Données L2I

129

Cours 5

Janvier 2012

# Création de table/relation (1) CREATE TABLE Produit (

```
référence CHAR(4),
nom VARCHAR(20),
marque VARCHAR(15),
constructeur CHAR(20)
);
```

```
CREATE TABLE Client (
numéro INT(10),
nom VARCHAR(20),
adresse VARCHAR(30),
téléphone CHAR(10)
);
```

CREATE TABLE Vente (
numéro INT(10),
ref-produit INT(10),
ref-client INT(10),
date DATE
);

- Compléter la description
- Voir avec les types MySQL

Philippe LAHIRE - Cours Base de Données L21

Cours 5

Janvier 2012

131

**Création de table (valeurs nulles, défaut)** 

- NOT NULL: l'attribut correspondant doit <u>toujours</u> avoir une valeur nom VARCHAR(20) <u>NOT NULL</u>
- DEFAULT : définir la valeur d' un attribut par défaut adresse VARCHAR (30) DEFAULT 'Inconnue'

```
CREATE TABLE Produit (
référence CHAR(4) NOT NULL,
nom VARCHAR(20) NOT NULL,
marque VARCHAR(15),
constructeur CHAR(20) DEFAULT 'Renault'
);
```

Philippe LAHIRE – Cours Base de Données L21

Cours 5

Janvier 2012

#### **Création de table (clé primaire)**

#### Identification des tuples

```
PRIMARY KEY (att)
PRIMARY KEY (att1, ..., attn)
```

- → mono attribut ou multi attributs
- → Attributs figurant dans une clé : déclaration NOT NULL

```
CREATE TABLE Produit (
référence CHAR(4) NOT NULL,
nom VARCHAR(20) NOT NULL,
marque VARCHAR(15),
constructeur CHAR(20) DEFAULT 'Renault',
PRIMARY KEY (référence)
);
```

Philippe LAHIRE - Cours Base de Données L21

Cours 5

Janvier 2012

133

# Création de table (unicité)

#### Unicité des valeurs dans une colonne

**UNIQUE** (nom, prenom)

→ Principalement pour les clés secondaires

```
CREATE TABLE Produit (
référence CHAR(4) NOT NULL,
nom VARCHAR(20) NOT NULL,
marque VARCHAR(15),
constructeur CHAR(20),
PRIMARY KEY (référence),
UNIQUE (nom)
);
```

NOT NULL non spécifié UNIQUE ne s'applique pas aux valeurs nulles

Philippe LAHIRE – Cours Base de Données L21

Cours 5

Janvier 2012

#### **Création de table (clé étrangère)**

```
FOREIGN KEY (att) REFERENCES ...
FOREIGN KEY (att1, ..., attn) REFERENCES ...
```

#### FOREIGN KEY(ref-produit) REFERENCES Produit

→ ref-produit référence la clé primaire de la table Produit

```
CREATE TABLE Vente (
numéro INT(10) NOT NULL,
ref-produit INT(10) NOT NULL,
ref-client INT(10) NOT NULL,
date DATE,
PRIMARY KEY (numéro),
FOREIGN KEY (ref-produit) REFERENCES Produit,
FOREIGN KEY (ref-client) REFERENCES Client
);
```

Philippe LAHIRE - Cours Base de Données L21

Cours 5

Janvier 2012

135

#### Gestion de l'intégrité référentielle

Vérification, si insertion, suppression, mise à jour ...

Les valeurs prises par la clé étrangère correspondent-elles à la clé?

Action par défaut : rejet de l'opération

Autres actions si celle par défaut ne convient pas

- ON UPDATE (en cas de mise à jour)
- ON DELETE (en cas de suppression)

Objectif: faire respecter la contrainte

- SET NULL clé étrangère mise à NULL
- CASCADE application de la même opération
- SET DEFAULT valeur par défaut pour la clé étrangère

Philippe LAHIRE – Cours Base de Données L21

Cours 5

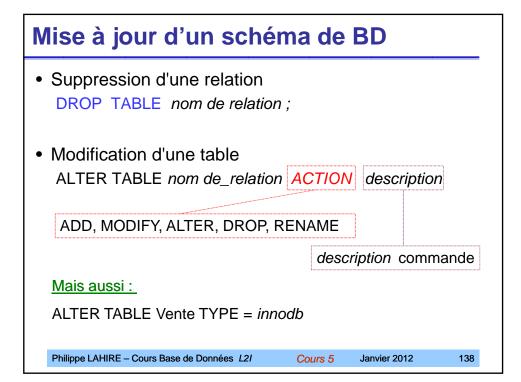
Janvier 2012

#### Intégrité référentielle (exemple)

```
CREATE TABLE Vente (
numéro INT(10) NOT NULL,
ref-produit INT(10) NOT NULL,
ref-client INT(10) NOT NULL,
date DATE,
PRIMARY KEY (numéro),
FOREIGN KEY (ref-produit) REFERENCES Produit
ON DELETE SET NULL ON UPDATE CASCADE,
FOREIGN KEY (ref-client) REFERENCES Client
ON DELETE SET NULL ON UPDATE CASCADE
);
```

- Si suppression dans Produit : ref-produit = NULL
- Si mise à jour dans Produit : ref-produit = mise à jour

Philippe LAHIRE – Cours Base de Données *L21* Cours 5 Janvier 2012 137



#### **Modification d'une table (structure)**

ALTER TABLE *Produit* MODIFY nom VARCHAR(25);
ALTER TABLE *Produit* ADD quantité-stock VARCHAR(20);

ALTER TABLE Client ALTER adresse SET DEFAULT 'NICE';

ALTER TABLE Client DROP adresse;

ALTER TABLE Client
ADD CONSTRAINT CT\_UN UNIQUE (nom);

ALTER TABLE Client
ADD CONSTRAINT CT\_PR PRIMARY KEY (numéro);

ALTER TABLE Vente ADD CONSTRAINT CT\_ET FOREIGN KEY (ref-client) REFERENCES Client;

Philippe LAHIRE – Cours Base de Données L2I

Cours 5

Janvier 2012

139

#### Mises à jour des données

INSERT INTO

pour insérer des tuple

DELETE FROM

pour supprimer des tuples

UPDATE

pour mettre à jour des tuples

Philippe LAHIRE – Cours Base de Données L21

Cours 5

Janvier 2012

# **Insertion de tuples (1)**

Insertion avec désignation explicite des colonnes

Insertion dans l'ordre des colonnes

Philippe LAHIRE – Cours Base de Données L2I

Cours 5

Janvier 2012

141

# Insertion de tuples (2)

Insertion à partir d'une autre table

**INSERT INTO** *Produit* (nom, marque, constructeur)

**SELECT** P.nom, P.marque, p.constructeur

FROM OldProduit P

WHERE P.marque = 'Peugeot'

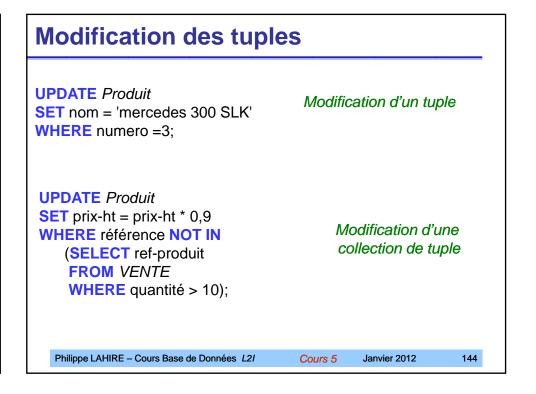
**Evolution du schéma relationnel** 

Philippe LAHIRE – Cours Base de Données L21

Cours 5

Janvier 2012

#### **Suppression de tuples** Suppression de tous les tuples **DELETE FROM Produit** ou **DELETE FROM Produit WHERE 1 > 0** (retourne le nombre) Suppression conditionnelle **DELETE FROM Produit WHERE** marque = 'peugeot' **DELETE FROM VENTE WHERE ref-produit IN** (SELECT référence **FROM** Produit **WHERE** *Produit.*marque = 'peugeot'); Philippe LAHIRE – Cours Base de Données L21 Cours 5 Janvier 2012 143



#### Compléments: Définition de vues

Une <u>vue</u> est une <u>table virtuelle</u> dérivée de tables de base :

- On stocke seulement la définition de vue
- Son contenu est généré dynamiquement

# CREATE VIEW ProduitPhare AS

**SELECT\*** 

FROM Vente V, Produit P WHERE

*V*.ref-produit = *P*.référence AND *V*.quantité > 100 ;

CREATE VIEW VenteInfo (référence, date, marque)

٩S

SELECT *P*.référence, *V*.date, *P*.marque

FROM Vente V, Produit P

**WHERE** *P*.référence = *V*.ref-produit;



DROP VIEW VenteInfo

Philippe LAHIRE - Cours Base de Données L2I

Cours 5

Janvier 2012

145

#### Compléments: Création de contraintes

#### **CREATE ASSERTION**

CHECK pour décrire des contraintes générales

**CREATE ASSERTION** QuantiteVendue

**CHECK NOT EXISTS (** 

**SELECT** *V.*quantité

FROM Vente V, Produit P

**WHERE** 

*V.*ref-produit = *P.*référence AND

P.quantité-vendue < V.quantité);

On suppose que la table Produit contient la colonne quantité-vendue

Philippe LAHIRE – Cours Base de Données L21

Cours 5

Janvier 2012

#### Compléments: Création d'index

#### **CREATE INDEX**

Un index offre un chemin d'accès aux lignes d'une table

#### **CREATE INDEX index1 ON Client(nom,prenom)**;

un index est systématiquement défini sur la clé primaire de chaque table

pour chaque clause UNIQUE utilisée dans la création de la table, un index permet de vérifier rapidement, au moment d'une insertion, que la clé n'existe pas déjà

Philippe LAHIRE - Cours Base de Données L21

Cours 5

Janvier 2012

147

#### Compléments: Transactions

Enlever la validation implicite:

SET AUTOCOMMIT=0;

**Déclencher une transaction et terminer par valider/annuler** START TRANSACTION;

SELECT @A:=SUM(salary) FROM table1 WHERE type=1; UPDATE table2 SET summary=@A WHERE type=1; COMMIT; ou ROLLBACK;

Un mécanisme complexe et sophistiqué

Philippe LAHIRE – Cours Base de Données L2I

Cours 5

Janvier 2012

# Compléments: Gestion des privilèges

SHOW DATABASES; SHOW PRIVILEGES;

GRANT privilèges ON ... TO ...

REVOKE privilèges ON ... FROM ...

ALL [PRIVILEGES]	Tous les droits sauf WITH GRANT OPTION
ALTER	Autorise l'utilisation de ALTER TABLE
CREATE	Autorise l'utilisation de CREATE TABLE
DELETE	Autorise l'utilisation de DELETE
DROP	Autorise l'utilisation de DROP TABLE

Philippe LAHIRE – Cours Base de Données L2I Cours 5 Janvier 2012 149