

# Internet et Outils

## L1/IO2 2007 - 2008

S2-IO2

### Bases de données: Introduction à MySQL, Tables simples

**François Armand**

armand@informatique.univ-paris-diderot.fr

François Armand

1

# Plan Général Prévisionnel (1/2)

(non contractuel)

- Cours Internet et Outils:
  - [1/12] Intro, Internet, Web, XHTML (2H)
  - [2/12] XHTML(2H)
  - [3/12] CSS (2H)
  - [4/12] PHP (2H)
  - [5/12] PHP suite (2H)
  - [6/12] Introduction MySQL, Table simple (2H)
  - [7/12] ► **Partiel (2H)** ◀

François Armand

2

# Plan Général Prévisionnel (2/2)

(non contractuel)

- Bases de données
  - [8/12] Tables multiples, Schémas (2H)
  - [9/12] Concepts un peu plus avancés (2H)
  - [10/12] Introduction aux réseaux(2H)
  - [11/12] Réseaux: compléments (2H)
  - [12/12] Révisions (2H)

François Armand

3

# BD: Plan

- Ce que vous allez découvrir
- Systèmes de Gestion de Bases De Données
- MySQL, Connexion, Création d'une base
- Table : Création et Insertion de données
- Table : Interrogation, modification, suppression de données
- Accès depuis PHP
- Types SQL, Commandes SQL,
- Fonctions (My)SQL
- Ce qu'il faut retenir

François Armand

4

## Ce que vous allez découvrir

- Pourquoi des bases de données
- Le langage SQL (très petit aperçu)
- Manipulation d'une table
- Puissance du langage
- Utilisation depuis PHP
- Survol des types de données, espace requis
- <http://dev.mysql.com/doc/refman/5.0/fr/index.html>

## BD: Plan

- Ce que vous allez découvrir
- **Systèmes de Gestion de Bases De Données**
- MySQL, Connexion, Création d'une base
- Table : Création et Insertion de données
- Table : Interrogation, modification, suppression de données
- Accès depuis PHP
- Types SQL, Commandes SQL,
- Fonctions (My)SQL
- Ce qu'il faut retenir

## Une liste (de films)

- Pour gérer une liste (de films), on peut utiliser un fichier manipulé avec un éditeur de texte:

```
--> cat films.txt
```

```
Alien 1979 Scott Ridley 1943
```

```
Vertigo 1958 Hitchcock Alfred 1899
```

```
Psychose 1960 Hitchcock Alfred 1899
```

```
Kagemusha 1980 Kurosawa Akira 1910
```

```
Volte-face 1997 Woo John 1946
```

## Problèmes

- Accès difficile aux données
  - Ouvrir le fichier,
  - Parcourir les "lignes",
  - Effectuer les comparaisons,
  - Format complexe (tenir compte des espaces...)
- Sécurité
  - Comment empêcher un programme erroné ou malveillant d'écrire des données autres (d'écraser le contenu du fichier)?

## Problèmes

- Concurrency:
  - Que se passe-t-il si plusieurs utilisateurs
    - ajoutent,
    - modifient
    - suppriment
  - des lignes simultanément?

## Bases de Données : SGBD

- Les Systèmes de Gestion de Bases de Données ont été conçus pour apporter des (bonnes) réponses à ces problèmes
  - Manipuler facilement des informations complexes,
  - Vérifier ce qui peut l'être,
  - Assurer un résultat cohérent quand plusieurs utilisateurs accèdent simultanément aux données.
- MySQL, Oracle, PostgreSQL, SQL Server, DB2
  - Systèmes de Gestion de Bases de Données

## SGBD Relationnels

- Représentation de l'information sous forme de tables
- La manière dont l'information est réellement stockée sur disque est inconnue de l'application
  - Peut-être plusieurs fichiers
- L'application ne voit que la "table" présentée par le SGBD
- Langage "standard": **SQL** (Norme ANSI de 1992)
  - Structured Query Language

## Films: Sous forme de Table

Film Nom de la table

Titre	Annee	Nom	Prenom	Naissance
Alien	1979	Scott	Ridley	1943
Vertigo	1958	Hitchcock	Alfred	1899
Psychose	1960	Hitchcock	Alfred	1899
Kagemusha	1980	Kurosawa	Akira	1910
Volte-face	1997	Woo	John	1946
Pulp Fiction	1995	Tarantino	Quentin	1964
Titanic	1997	Cameron	James	1954
Sacrifice	1986	Tarkovski	Andrei	1932

- Informations conformes à une description précise,
- Possibilité de **flexibilité** mais aussi de contraintes
- Attributs définis: Titre, Année, ...

## BD: Plan

- Ce que vous allez découvrir
- Systèmes de Gestion de Bases De Données
- **MySQL, Connexion, Création d'une base**
- Table : Création et Insertion de données
- Table : Interrogation, modification, suppression de données
- Accès depuis PHP
- Types SQL, Commandes SQL,
- Fonctions (My)SQL
- Ce qu'il faut retenir

François Armand

13

## SQL

- Permet de définir la structure d'une table
  - (ou de plusieurs)
- Permet aussi d'interroger la base de manière simple:
  - SELECT** Titre // L'attribut recherché
  - FROM** Film // La table interrogée
  - WHERE** Année > 1980 // Le critère de sélection
- SQL: langage déclaratif

François Armand

14

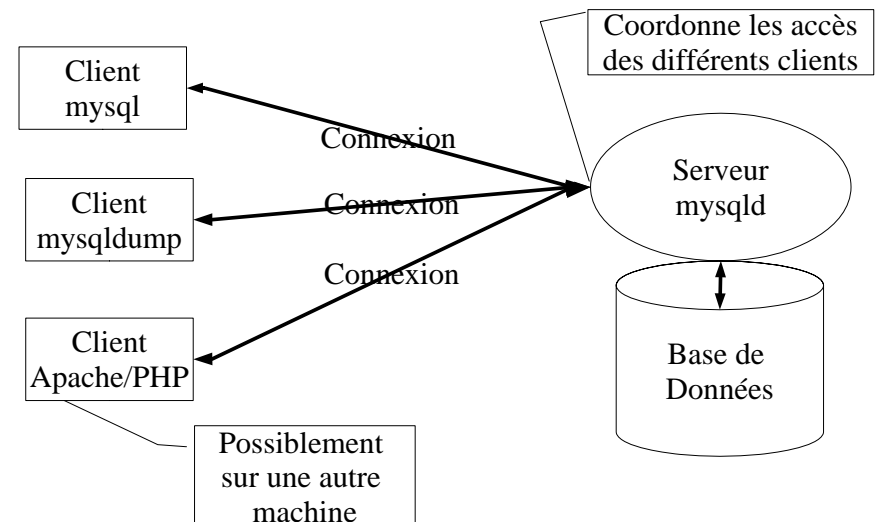
## MySQL

- Un serveur : `mysqld`
  - Seul capable d'accéder aux informations
- Des utilitaires:
  - `mysql` : client permettant de manipuler et interroger les données
  - `mysqldump`: client permettant de faire des sauvegardes,
  - Apache/PHP: client intégré avec un serveur "web"
  - `phpMyAdmin`: Interface graphique d'administration

François Armand

15

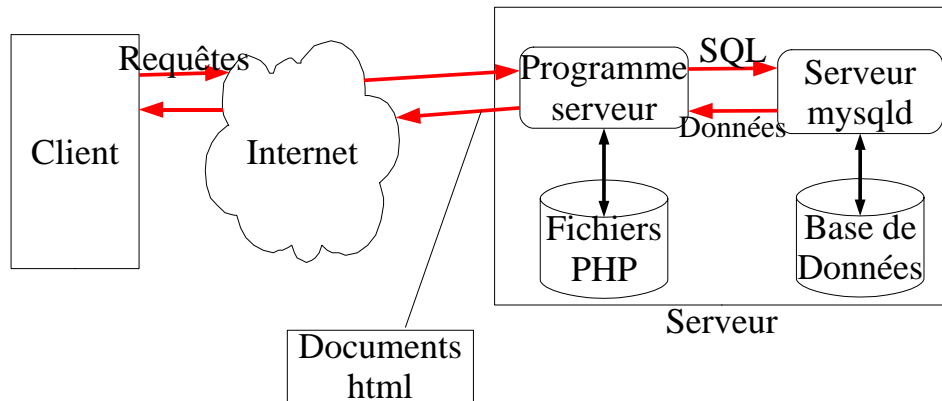
## Serveurs Clients MySQL



François Armand

16

# Site Web avec MySQL / PHP



François Armand

17

# Connexion à MySQL

```
-> mysql -u root -p
```

Enter password: \*\*\*\*\*

Welcome to the MySQL monitor.  
Commands end with ; or \g.

Your MySQL connection id is 13 to  
server version: 4.1.9-max

Type 'help;' or '\h' for help. Type  
'\c' to clear the buffer.

```
mysql> help
```

François Armand

18

# Création d'une base

```
mysql> CREATE DATABASE Films;
```

```
mysql> # Commande SQL
```

```
mysql> GRANT ALL PRIVILEGES ON  
Films.* TO admFilm@localhost  
IDENTIFIED BY 'motdepasse';
```

```
Query OK, 0 rows affected (0.00 sec)
```

```
mysql> exit # ordre mysql
```

François Armand

19

# Bases et Tables

- Au sein d'un serveur on peut créer plusieurs bases différentes
- Chaque base à ses propres tables
  - Une table appartient à une seule base
- On peut associer des droits d'accès aux utilisateurs des bases, et des tables

François Armand

20

# Connexion à une base MySQL

- > **mysql -u admin -p Films**
- idem que précédemment, mais utilisateur admin, et connexion à la base Films immédiatement.
- On peut aussi utiliser le fichier de configuration
  - \$HOME/.my.cnf
  - [client]
  - user= admin
  - password= motdepasse
- > **mysql Films**

François Armand

21

# BD: Plan

- Ce que vous allez découvrir
- Systèmes de Gestion de Bases De Données
- MySQL, Connexion, Création d'une base
- **Table : Création et Insertion de données**
- Table : Interrogation, modification, suppression de données
- Accès depuis PHP
- Types SQL, Commandes SQL,
- Fonctions (My)SQL
- Ce qu'il faut retenir

François Armand

22

# Création d'une table

- S'assurer d'abord que l'on travaille sur la bonne base!
- > **mysql**
- mysql> USE Films;**
- Database changed
- mysql>**

François Armand

23

# Création d'une table

```
mysql> CREATE TABLE FilmSimple
-> (Titre      VARCHAR(30),
->  Annee      INTEGER,
->  Nom        VARCHAR(30),
->  Prenom     VARCHAR(30),
->  Naissance  INTEGER
-> );
```

Query OK, 0 rows affected (0.05 sec)

François Armand

24

## Description d'une table

```
mysql> DESC FilmSimple;
```

Field	Type	Null	Key	Def	Xtr
Titre	varchar(30)	YES		NULL	
Annee	int(11)	YES		NULL	
NOM	varchar(30)	YES		NULL	
Prenom	varchar(30)	YES		NULL	
Naiss.	int(11)	YES		NULL	

5 rows in set (0.00 sec)

François Armand

25

## Champs

- Les champs de la table (**Fields**) sont souvent appelés:
  - Attributs
  - Colonnes
- A sa création, une table est "vide":
  - Elle ne contient aucune donnée
  - Aucune "ligne"

François Armand

26

## Détruire une table

```
mysql> DROP TABLE FilmSimple;
```

```
Query OK, 0 rows affected (0.03 sec)
```

François Armand

27

## SCRIPTS SQL

- On peut entrer les commandes SQL depuis un fichier préalablement édité.
- On utilise conventionnellement le suffixe '.sql'
 

```
-> mysql -u admin -p < script.sql
```

```
->
```
- Ou depuis mysql:
 

```
mysql> source script.sql
```

François Armand

28

## Insérer des données

```
mysql> INSERT INTO FilmSimple
-> (Titre, Annee, Prenom, Nom)
-> VALUES ('Pulp Fiction', 1995,
->         'Quentin', 'Tarantino'
-> );
```

Query OK, 1 row affected (0.00 sec)

- Note: on peut ne saisir que quelques attributs, et dans un ordre différent de celui défini pour la table.

François Armand

29

## Insérer des données

- Note: on peut ne saisir que quelques attributs, et dans un ordre différent de celui défini pour la table.

```
mysql> INSERT INTO FilmSimple
-> (Nom, Titre)
-> VALUES ('Allen',
->         'Match Point'
-> );
```

Query OK, 1 row affected (0.00 sec)

François Armand

30

## Films

Film

Titre	Annee	Nom	Prenom	Naissance
Alien	1979	Scott	Ridley	1943
Vertigo	1958	Hitchcock	Alfred	1899
Psychose	1960	Hitchcock	Alfred	1899
Kagemusha	1980	Kurosawa	Akira	1910
Volte-face	1997	Woo	John	1946
Pulp Fiction	1995	Tarantino	Quentin	
Titanic	1997	Cameron	James	1954
Sacrifice	1986	Tarkovksi	Andrei	1932
Match Point		Allen		

François Armand

31

## Insérer des données

- On peut aussi insérer des données depuis

- une interface graphique spécifique,
- un fichier (insertion dite en masse)

```
mysql> LOAD DATA LOCAL INFILE
-> 'films.txt' INTO TABLE FilmSimple
-> FIELDS TERMINATED BY ' ';
```

Query OK, 8 rows affected, 4 warnings (0.00 sec)

Records: 8 Deleted: 0 Skipped: 0 Warnings: 0

François Armand

32



## BD: Plan

- Ce que vous allez découvrir
- Systèmes de Gestion de Bases De Données
- MySQL, Connexion, Création d'une base
- Table : Création et Insertion de données
- **Table : Interrogation, modification, suppression de données**
- Accès depuis PHP
- Types SQL, Commandes SQL,
- Fonctions (My)SQL
- Ce qu'il faut retenir

François Armand

33

## Interroger la table

```
mysql> SELECT * FROM FilmSimple;
+-----+-----+-----+-----+-----+
|Titre   |Annee|  NOM  |Prenom|Naiss  |
+-----+-----+-----+-----+-----+
|Alien   |1979 | Scott|Ridley| 1943  |
+-----+-----+-----+-----+-----+
9 rows in set (0.00 sec)
```

- Résultat tronqué pour la présentation

François Armand

34

## Interroger la table

```
mysql> SELECT Titre, Annee FROM
FilmSimple;
+-----+-----+
|Titre   |Annee|
+-----+-----+
|Alien   |1979 |
+-----+-----+
9 rows in set (0.00 sec)
```

- On peut sélectionner les colonnes de son choix

François Armand

35

## Interroger la table

```
mysql> SELECT Titre, Annee
FROM FilmSimple
WHERE Titre='Vertigo'
OR Prenom='Alfred'
OR ((Annee >= 1980
AND Annee < 2000));
```

- La clause "WHERE" permet de définir un ensemble de conditions qui doivent être vérifiées par les lignes retenues.

François Armand

36

## Conditions Clause WHERE

- = égalité
- <> différence
- < inférieur, <= inférieur ou égal
- > supérieur, >= supérieur ou égal
- BETWEEN a AND b
- LIKE modèle
- On peut spécifier le contraire avec NOT
- On peut combiner les clauses avec AND / OR

François Armand

37

## Interroger la table: BETWEEN

```
mysql> SELECT Titre, Annee
FROM FilmSimple WHERE Titre
BETWEEN 'Psychose' AND 'Titanic';
```

Titre	Annee
Pulp Fiction	1995
Psychose	1960
Titanic	1997
Sacrifice	1986

François Armand

38

## Interroger la table: LIKE

```
mysql> SELECT Titre, Annee
FROM FilmSimple WHERE Titre
LIKE '%ti%';
```

Titre
Pulp Fiction
Vertigo
Titanic

François Armand

39

## Interroger la table: DISTINCT

```
mysql> SELECT DISTINCT Nom
FROM FilmSimple
WHERE Naissance < 1942;
```

Nom
Hitchcock
Kurosawa
Tarkovski

François Armand

40

## Interroger la table: ORDER BY

```
mysql> SELECT Titre, Annee FROM FilmSimple
WHERE Titre BETWEEN 'Psychose' AND
'Titanic' ORDER BY Titre;
```

Titre	Annee
Pulp Fiction	1995
Psychose	1960
Sacrifice	1986
Titanic	1997

François Armand

41

## Interroger la table: ORDER BY

- Note : Le tri n'est pas lié à "BETWEEN"!
- On peut faire un tri sur plus d'une colonne
- On peut trier dans l'ordre croissant (ASC) ou décroissant (DESC)
- Ex:
  - SELECT Annee, Titre FROM FilmSimple ORDER BY Annee ASC, Titre DESC;
  - Liste les films par année
  - Dans une Année: ordre alphabétique inverse.

François Armand

42

## ORDER BY: résultat

Annee	Titre
1958	Vertigo
1960	Psychose
1979	Alien
1980	Kagemusha
1986	Sacrifice
1995	Pulp Fiction
1997	Volte-face
1997	Titanic

François Armand

43

## Supprimer des données

```
mysql> DELETE FROM FilmSimple WHERE
Annee <= 1960;
```

- Détruit tous les films dont l'année est inférieure ou égale à 1960.
- Fonctionnement par défaut de MySQL:
  - Données perdues
- Si mode transactionnel (optionnel) supporté alors possibilité de "rollback" (voir cours suivants)

François Armand

44

## Modifier des données

```
mysql> UPDATE FilmSimple
      SET Nom='Wu', Prenom='Yusen'
      WHERE Nom='Woo' ;
```

- Met à jour les champs 'Nom' et 'Prenom' de toutes les lignes sélectionnées par la clause 'WHERE'.
- Pour vérifier les lignes affectées par un ordre 'UPDATE' ou 'DELETE', faire préalablement un ordre 'SELECT' avec la même clause 'WHERE'.

## Quelques Commandes

- SELECT DATABASE();
  - Affiche le nom de la base courante;
- SELECT USER();
  - Affiche le nom de l'utilisateur courant;
- SHOW DATABASES;
  - Liste les bases de données;
- SHOW TABLES;
  - Liste les tables de la base courante;

## BD: Plan

- Ce que vous allez découvrir
- Systèmes de Gestion de Bases De Données
- MySQL, Connexion, Création d'une base
- Table : Création et Insertion de données
- Table : Interrogation, modification, suppression de données
- **Accès depuis PHP**
- Types SQL, Commandes SQL,
- Fonctions (My)SQL
- Ce qu'il faut retenir

## Accès à MySQL depuis PHP

- L'interface MySQL/PHP permet de
  - récupérer
  - modifier,
  - créer,
  - Toute information d'une base de données MySQL depuis un script PHP (y compris un script PHP inclus dans un document XHTML).
- Accès à MySQL via des fonctions PHP prédéfinies.

## Principales fonctions MySQL/PHP

- `mysql_connect()`
- `mysql_pconnect()`
  - Connexion avec un serveur pour un utilisateur,
  - Equivalent de "mysql"
- `mysql_select_db()`
  - Equivalent à la commande "USE base" de mysql.
- `mysql_query()`
  - Execution d'une requête SQL (SELECT, CREATE, UPDATE, INSERT....)

François Armand

49

## Principales fonctions MySQL/PHP

- Les résultats renvoyés par une requête SQL (SELECT) sont composés de plusieurs "lignes", ce sont des ensembles ("**result set**")
- On veut travailler sur les "lignes" individuelles du résultat:
- `mysql_fetch_array()`
  - Ligne courante représentée sous forme de tableau
    - Entrées du tableau indexés par 'attribut':
    - Ex: `$film['Nom']` donne le nom du réalisateur
  - Le prochain appel rendra la ligne suivante du résultat

François Armand

50

## Principales fonctions MySQL/PHP

- `mysql_fetch_row()`
  - Similaire à `mysql_fetch_array`
  - mais une ligne est un tableau indexée par 0, 1...
- `mysql_fetch_object()`
  - Similaire à `mysql_fetch_array`
  - Ligne représentée sous forme "d'objet"
    - Ex: `$film->Nom` pour accéder au nom du réalisateur du film
- `mysql_error()`
  - Message correspondant à la dernière erreur.

François Armand

51

## Exemple: Définitions

```
--> cat connect-def.php
<?php
define ('NOM', "admFilm");
define ('MDP', "motdepasse");
define ('SRV', "localhost");
define ('BASE', "Films");

?>
```

- Le suffixe 'php' assure que ce fichier ne sera pas transmis en clair à un navigateur (Ouf!)

François Armand

52

## Exemple

```
<?php
require ("connect-def.php");
$connexion = mysql_pconnect(SRV, NOM, MDP);
if (!$connexion) {
    echo "Pas connecté\n"; exit;
}
if (!mysql_select_db(BASE, $connexion)) {
    echo "Pas d'accès à la base\n"; exit;
}
```

François Armand

53

## Exemple suite

```
$resultat = mysql_query( "SELECT * FROM
    FilmSimple", $connexion);
if ($resultat) {
    while ($film = mysql_fetch_array($resultat)) {
        echo $film['Titre'].", de ".$film['Prenom']." ".$film['Nom'];
        echo "<br />\n";
    }
}
else { echo "Erreur<br />\n";}
?>
```

François Armand

54

## BD: Plan

- Ce que vous allez découvrir
- Systèmes de Gestion de Bases De Données
- MySQL, Connexion, Création d'une base
- Table : Création et Insertion de données
- Table : Interrogation, modification, suppression de données
- Accès depuis PHP
- Types SQL, Commandes SQL,
- Fonctions (My)SQL
- Ce qu'il faut retenir

François Armand

55

## Types de Données (My)SQL

- Principaux types de données SQL ANSI supportés par MySQL:
- SMALLINT (2 octets)
- INTEGER (4 octets)
- BIGINT (8 octets)
- FLOAT (4 octets)
- REAL (8 octets)
- DOUBLE PRECISION (8 octets)

François Armand

56

## Types de Données (My)SQL

- VARCHAR (*M*) *M* < 255 taille variable
- CHAR (*M*) *M* < 255 taille fixe
- MySQL: BLOB / TEXT taille < 2<sup>16</sup>
- MySQL: MEDIUMBLOB / MEDIUMTEXT taille < 2<sup>24</sup>
- MySQL: LONGBLOB / LONGTEXT taille < 2<sup>32</sup>
- DATE (3 octets)
- TIME (3 octets)

## Types de Données (My)SQL

- DATETIME (8 octets)
- TIMESTAMP (4 octets)
- YEAR (1 octets)
- MySQL: ENUM (<= 2 octets)
- MySQL: SET (<= 8 octets)

## Types de Données

- INTEGER[(*M*)] [UNSIGNED] [ZEROFILL]
  - Entier taille max: *M* caractères
  - Non signé si UNSIGNED
  - ZEROFILL: 3 -> 0000000003
- VARCHAR(*M*) [BINARY]
  - Par défaut, pas de différence entre minuscules et majuscules => 'aaa' identique à 'AAA'
  - Sauf si BINARY est précisé

## Principales Commandes SQL

- CREATE DATABASE *nombase*
  - Créer une base
- DROP DATABASE [IF EXISTS] *nombase*
  - Détruire une base
- CREATE TABLE *nomTable* (*descr*)
  - Créer une table
- ALTER TABLE *nomTable* *commande*
  - Modifier la description d'une table

## Modification d'une table

- ALTER TABLE *maTable*
- ADD (*nomAttr Type*) [FIRST | AFTER *nomAttr2*]
  - Ajout d'un nouvel attribut dans une table existante
- DROP *nomAttr*
  - Suppression d'une colonne
- CHANGE *nomAttr (nouvelAttr Type)*
  - Modifier un attribut existant
- RENAME *NouveauNomDeLaTable*

## Principales Commandes SQL

- DELETE FROM *Table* WHERE *clause*
  - Détruit les lignes vérifiant "*clause*"
- SELECT *listeAttr* FROM *Table* WHERE *clause*
  - Extraction des lignes appropriées d'une table
- INSERT *Table* (*listeAttribut*) VALUES (...)
- UPDATE *Table* SET *Attr=value* [WHERE *clause*]
  - Mise à jour

## BD: Plan

- Ce que vous allez découvrir
- Systèmes de Gestion de Bases De Données
- MySQL, Connexion, Création d'une base
- Table : Création et Insertion de données
- Table : Interrogation, modification, suppression de données
- Accès depuis PHP
- Types SQL, Commandes SQL,
- Fonctions (My)SQL
- Ce qu'il faut retenir

## Fonctions MySQL

- Avec MySQL on peut utiliser des fonctions dans les requêtes:
  - Exemple: clauses WHERE ou expression pour affecter des valeurs à des champs
- Pour la plupart Ajout de MySQL à la norme SQL
- Exemples:
  - ABS(nombre) : valeur absolue
  - CONCAT (chaîne1, [chaîne2,...]) concaténation des chaînes



## Quelques Fonctions MySQL

- CEILING(*nombre*)
  - Renvoie l'entier immédiatement  $\geq$  *nombre*
- CURDATE()
  - Renvoie la date AAAAMMJJ ou AAAA-MM-JJ
- CURTIME
  - Renvoie l'heure HHMMSS ou HH:MM:SS
- DATE\_FORMAT(*date*, *format*)
- FLOOR(*nombre*)
  - Renvoie l'entier immédiatement  $\leq$  *nombre*

François Armand

65

## Quelques Fonctions MySQL

- IF (*test*, *val1*, *val2*)
  - Renvoie *val1* si *test* est vrai, *val2* sinon
- INSTR (*chaine*, *sous-chaine*)
  - Position de *sous-chaine* dans *chaine*
- LENGTH (*chaine*)
  - Renvoie la longueur de *chaine*
- STRCMP (*chaine1*, *chaine2*)
  - 0 si égalité, -1 si *chaine1* < *chaine2*, +1 sinon

François Armand

66

## BD: Plan

- Ce que vous allez découvrir
- Systèmes de Gestion de Bases De Données
- MySQL, Connexion, Création d'une base
- Table : Création et Insertion de données
- Table : Interrogation, modification, suppression de données
- Accès depuis PHP
- Types SQL, Commandes SQL,
- Fonctions (My)SQL
- **Ce qu'il faut retenir**

François Armand

67

## Ce qu'il faut retenir

- Bases de Données relationnelles:
  - Moyen simple et puissant pour manipuler des tables
- SQL:
  - Langage puissant pour exprimer des requêtes sur des tables
  - Principales commandes SQL
- PHP:
  - Connexion simple de PHP à un serveur MySQL

François Armand

68