

# Bases de données

## Premiers usages du SGBD

### Avec PhpMyADMIN et la Calculette

Site officiel MySQL : <https://www.mysql.com>

Site officiel MariaDB : <https://mariadb.org/>

Documentations MySQL : <http://dev.mysql.com/doc/index.html>

Mémo SQL : [http://www.volubis.fr/bonus/SQL\\_memo.htm](http://www.volubis.fr/bonus/SQL_memo.htm)

Documentations MySQL 5.0 en français :

<http://bliaudet.free.fr/IMG/pdf/MySQL-refman-5.0-fr.pdf>

Bertrand LIAUDET

## SOMMAIRE

<b>SOMMAIRE</b>	<b>1</b>
<b>INTRODUCTION A L'USAGE D'UNE BD MYSQL</b>	<b>3</b>
<b>Présentation</b>	<b>3</b>
<b>0. Composition d'un SGBD MySQL</b>	<b>3</b>
Principes	3
<b>1. Manipulation des bases de données</b>	<b>3</b>
Consultation des BD du SGBD dans la calculette	3
Consultation des tables d'une BD	4
Création d'une BD	4
Suppression d'une BD	5
Synthèse	5
Usages PhpMyAdmin : bouton droit, phpMyAdmin	6
TP : création de la table EmployesTP01.sql	8
<b>2. Manipulation des tables</b>	<b>9</b>
Consultation des attributs d'une table dans la calculette	9
Préfixe : nomBD.nomTable	9
Création d'une table	9
Consultation du code de création d'une table enregistré par le SGBD	9
Suppression d'une table	10
Synthèse	10
TP : consultation de la table EmployesTP01.sql	10
<b>3. Manipulation des tuples</b>	<b>13</b>

Consultation de tous les tuples d'une table dans la calculette SQL	13
Consultation de certains tuples d'une table	13
Modification des tuples d'une table	13
TP : consultation et modification des tuples de la table EmployesTP01.sql	14
<b>4. DATADIR et commandes de la calculette</b>	<b>17</b>
Datadir	17
Contenu du répertoire des données	17
Quelques commandes de la calculette	17
TP 17	
<b>5. Précisions sur la création des tables : CREATE TABLE</b>	<b>18</b>
Syntaxe SQL	18
Exemple	18
Les types des données	18
« Moteurs » MyIsam et InnoDB	18
TP 19	
<b>6. Création des tuples : INSERT INTO</b>	<b>20</b>
Syntaxe SQL	20
Exemples	20
Auto-increment	20
Afficher la situation de l'auto-increment	20
TP 20	
<b>7. Les utilisateurs</b>	<b>21</b>
Utilisateur courant	21
root@localhost	21
Tous les utilisateurs	21
Que faire si on a perdu le mot de passe de root ?	21
<b>8. Messages du serveur</b>	<b>22</b>
<b>9. Sauvegarde</b>	<b>23</b>
Sauvegarder en utilisant PhpMyAdmin	23
Sauvegarder en utilisant Mysqldump	24
<b>10. Import - Export</b>	<b>25</b>
Principes	25
Exporter avec PhpMyAdmin	25
Importer avec PhpMyAdmin	25
TP 26	
Exporter les données avec la calculette : SELECT ... INTO OUTFILE	26
Importer les données avec la calculette : LOAD DATA INFILE	27

Dernière édition : Novembre 2018

# INTRODUCTION A L'USAGE D'UNE BD MYSQL

## PRINCIPALES NOTIONS

Show databases	Use
Create database	Desc
Drop database	CREATE TABLE
Show tables from	INSERT INTO

## Présentation

Dans ce cours, il faut faire tous les TP qui apparaissent en jaune. Le cours qui précède permet de préparer le TP.

## 0. Composition d'un SGBD MySQL

### Principes

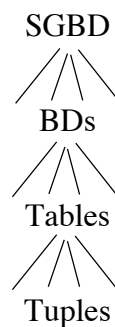
Un SGBD contient des BDs.

Une BD contient des Tables.

Une Tables contient des Tuples.

La relation « contenir » est transitive : un SGBD contient des Tuples.

La relation forme une arborescence (un arbre) :



## 1. Manipulation des bases de données

### Consultation des BD du SGBD dans la calculette

La BD contient des tables. C'est une sorte de dossier contenant des tables (et d'autres choses).

Les BD peuvent aussi s'appeler « schema ».

#### Show databases

La commande « show tables » permet de lister toutes les BD du SGBD :

```
mysql> show databases;
+-----+
| Database           |
+-----+
| information_schema |
| mysql              |
| performance_schema |
| sys                |
+-----+
4 rows in set (0.00 sec)
```

**information\_schema** et **mysql** sont deux bases qui **gèrent le dictionnaire des données**. Le dictionnaire des données contient des métadonnées : des données sur les données enregistrées dans la BD.

**performance\_schema** et **sys** sont deux bases qui **gèrent les performances**. La BD sys est arrivée dans les versions les plus récentes.

#### ➤ *Documentation*

- mysql : <https://dev.mysql.com/doc/refman/5.5/en/system-database.html>
- information\_schema : <http://dev.mysql.com/doc/refman/5.7/en/information-schema.html>
- performance\_schema : <http://dev.mysql.com/doc/refman/5.7/en/performance-schema.html>
- sys : <http://dev.mysql.com/doc/refman/5.7/en/sys-schema.html>

### Use database

La commande « use database » permet de sélectionner une BD :

```
mysql> use database mysql;
```

### Select database()

La commande « select databases » permet de savoir qu'elle est la BD actuellement sélectionnée :

```
Select database() ;
```

### Consultation des tables d'une BD

#### Show tables

La commande « show tables » permet d'afficher toutes les tables d'une BD sélectionnée :

```
mysql> show tables;
```

On peut aussi préciser la BD souhaitée :

```
mysql> show tables from mysql;
```

### Création d'une BD

#### Create database

La commande « create database » permet de créer une nouvelle BD. C'est une sorte de dossier vide qui contiendra les futures tables de la BD :

```
Create database NomBD ;
```

On peut aussi vérifier la non-existence de la BD avant de la créer :

```
Create database if not exists NomBD ;
```

Ca évite un signal d'erreur si la BD existe.

### **Show create database**

La commande « show create database » permet montrer la commande qui a permis la création de la BD à l'identique.

```
Show create database mysql ;
```

### **Suppression d'une BD**

#### **Drop database**

La commande « drop database » permet de supprimer une BD. Attention, ça supprime aussi toutes les données !!! C'est à manier avec prudence !!!

```
Drop database NomBD ;
```

On peut aussi vérifier l'existence de la BD avant de la supprimer:

```
Drop database if exists NomBD ;
```

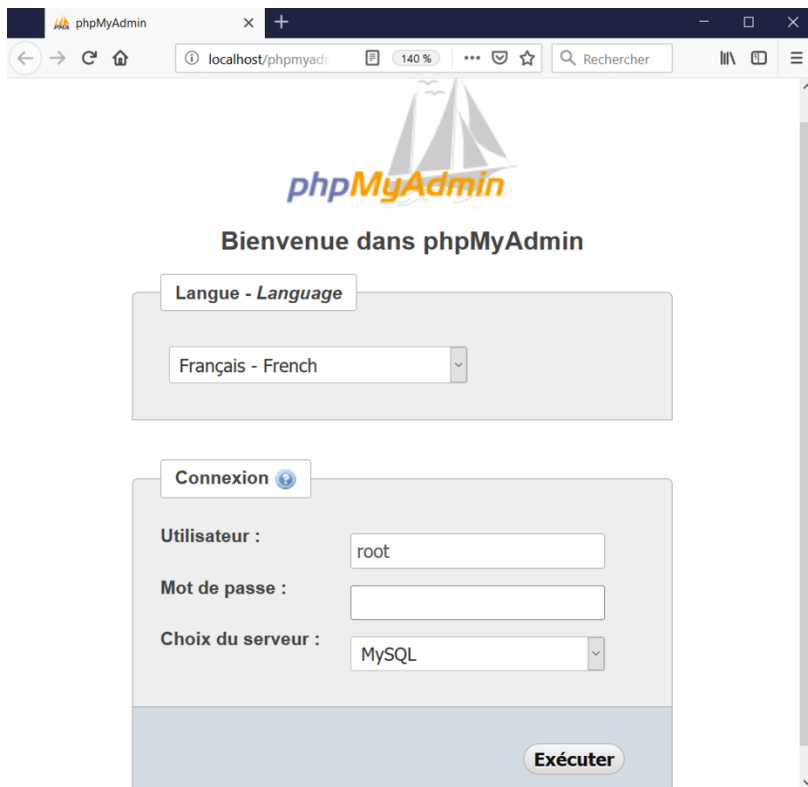
Ca évite un signal d'erreur si la BD n'existe pas.

### **Synthèse**

```
mysql> show databases;
mysql> use database mysql;
mysql> show tables
mysql> show tables from mysql
Select database() ;
Create database NomBD ;
Create database if not exists NomBD ;
Show create database mysql ;
Drop database NomBD ;
Drop database if exists NomBD ;
```

## Usages PhpMyAdmin : bouton droit, phpMyAdmin

### Connexion



### Fenêtre d'accueil



## Sélection d'une BD

On clique sur une des BD de la colonne de gauche (use nomBD). On fait apparaître ses tables (show tables).

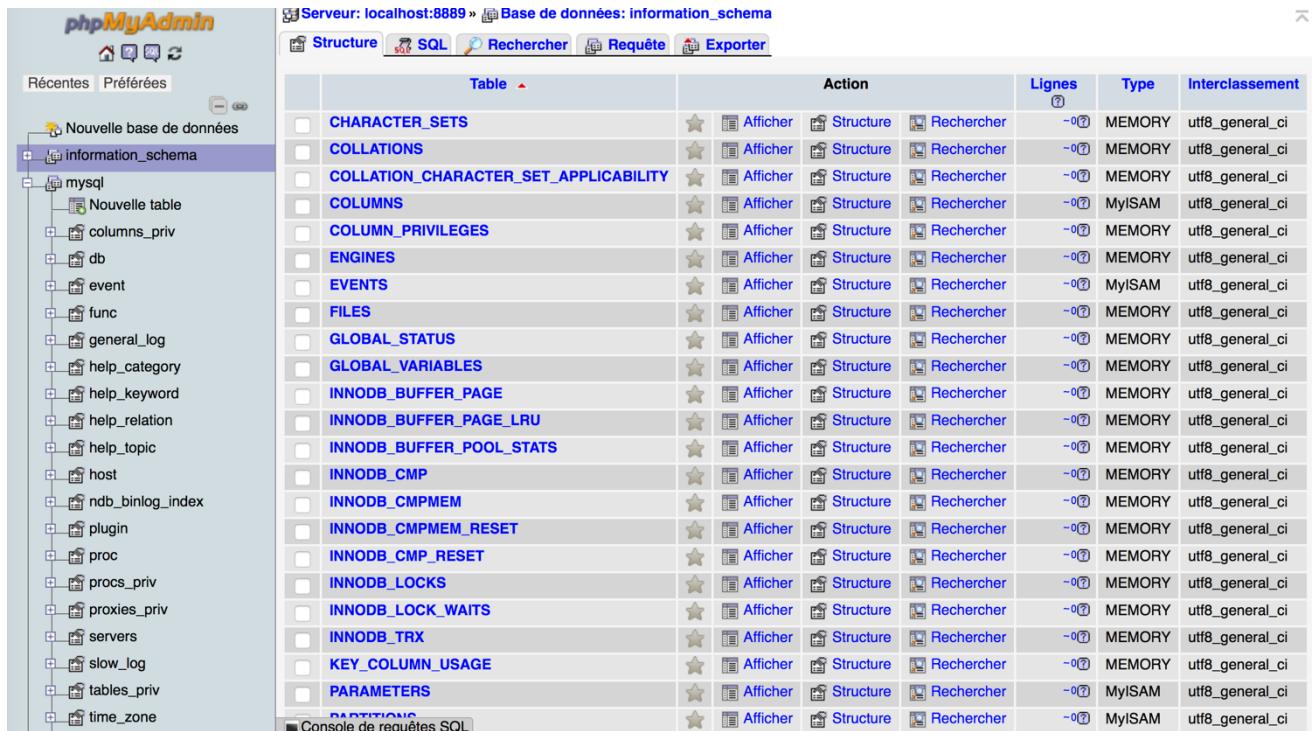
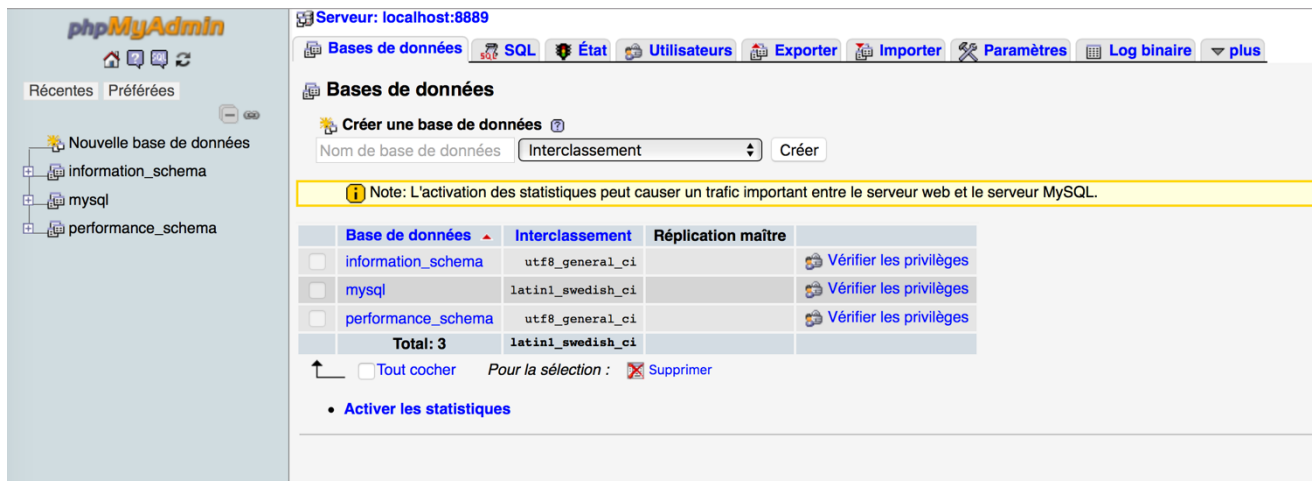


Table	Action	Lignes	Type	Interclassement
CHARACTER_SETS	Afficher Structure Rechercher	-0?	MEMORY	utf8_general_ci
COLLATIONS	Afficher Structure Rechercher	-0?	MEMORY	utf8_general_ci
COLLATION_CHARACTER_SET_APPLICABILITY	Afficher Structure Rechercher	-0?	MEMORY	utf8_general_ci
COLUMNS	Afficher Structure Rechercher	-0?	MyISAM	utf8_general_ci
COLUMN_PRIVILEGES	Afficher Structure Rechercher	-0?	MEMORY	utf8_general_ci
ENGINES	Afficher Structure Rechercher	-0?	MEMORY	utf8_general_ci
EVENTS	Afficher Structure Rechercher	-0?	MyISAM	utf8_general_ci
FILES	Afficher Structure Rechercher	-0?	MEMORY	utf8_general_ci
GLOBAL_STATUS	Afficher Structure Rechercher	-0?	MEMORY	utf8_general_ci
GLOBAL_VARIABLES	Afficher Structure Rechercher	-0?	MEMORY	utf8_general_ci
INNODB_BUFFER_PAGE	Afficher Structure Rechercher	-0?	MEMORY	utf8_general_ci
INNODB_BUFFER_PAGE_LRU	Afficher Structure Rechercher	-0?	MEMORY	utf8_general_ci
INNODB_BUFFER_POOL_STATS	Afficher Structure Rechercher	-0?	MEMORY	utf8_general_ci
INNODB_CMP	Afficher Structure Rechercher	-0?	MEMORY	utf8_general_ci
INNODB_CMPMEM	Afficher Structure Rechercher	-0?	MEMORY	utf8_general_ci
INNODB_CMPMEM_RESET	Afficher Structure Rechercher	-0?	MEMORY	utf8_general_ci
INNODB_CMP_RESET	Afficher Structure Rechercher	-0?	MEMORY	utf8_general_ci
INNODB_LOCKS	Afficher Structure Rechercher	-0?	MEMORY	utf8_general_ci
INNODB_LOCK_WAITS	Afficher Structure Rechercher	-0?	MEMORY	utf8_general_ci
INNODB_TRX	Afficher Structure Rechercher	-0?	MEMORY	utf8_general_ci
KEY_COLUMN_USAGE	Afficher Structure Rechercher	-0?	MEMORY	utf8_general_ci
PARAMETERS	Afficher Structure Rechercher	-0?	MyISAM	utf8_general_ci
PARTITIONS	Afficher Structure Rechercher	-0?	MyISAM	utf8_general_ci

## Nouvelle base de données – suppression d'une BD

On clique sur « nouvelle base de données » dans la colonne de gauche. On peut alors créer une BD (create database). On peut aussi supprimer une BD existante (drop database).



Base de données	Interclassement	Réplication maître
<input type="checkbox"/> information_schema	utf8_general_ci	Vérifier les privilèges
<input type="checkbox"/> mysql	latin1_swedish_ci	Vérifier les privilèges
<input type="checkbox"/> performance_schema	utf8_general_ci	Vérifier les privilèges
<b>Total: 3</b>	latin1_swedish_ci	

## TP : création de la table EmployesTP01.sql

Chargez le fichier EmployesTP01.sql qui se trouve sur le site.

Ce fichier crée une BD nommée employesTP01.

Faites le travail dans une calculette et dans phpMyAdmin.

### Dans une calculette SQL

Coller le code du fichier dans une calculette SQL.

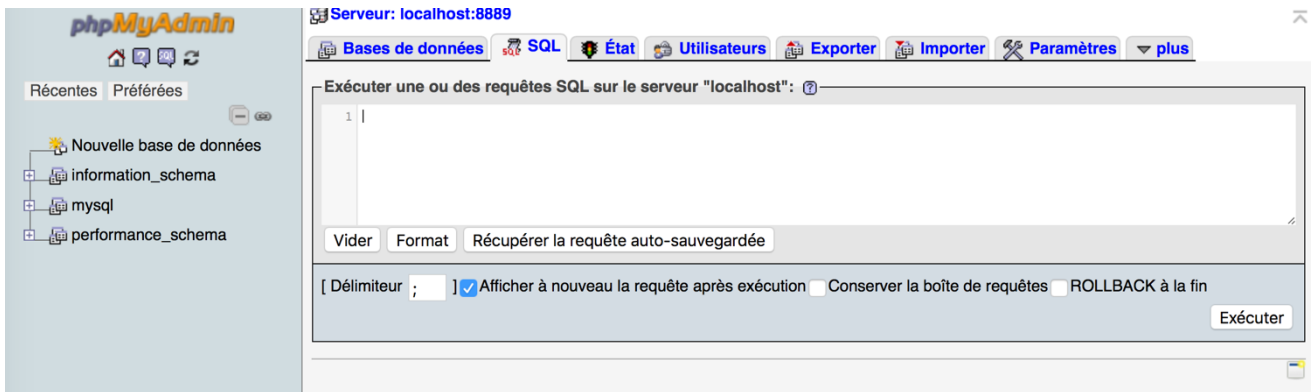
Consulter les tables créées.

Consulter le code de création de la BD.

### Dans WAMP


Sélectionner « Nouvelle base de données »

Aller sur l'onglet SQL.

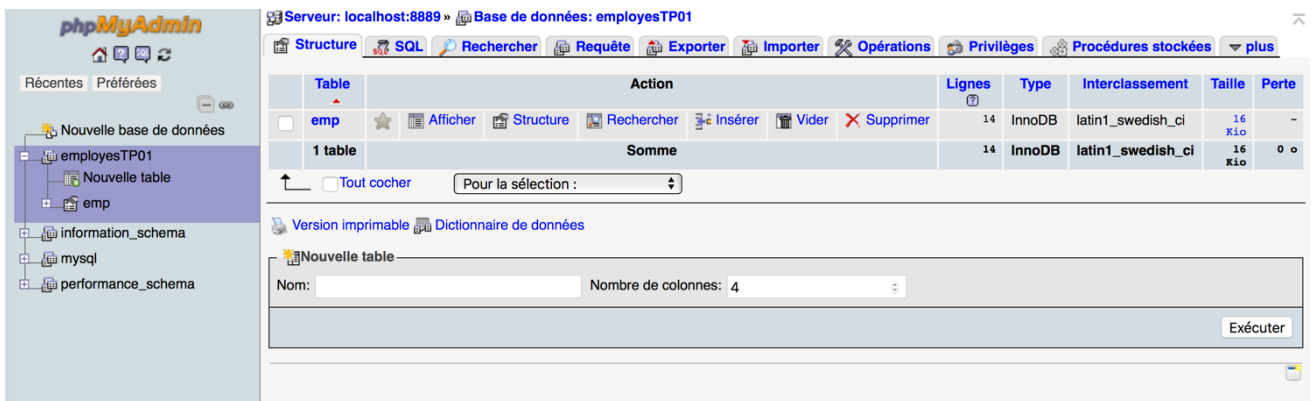


Coller le code du fichier dans la zone blanche.

Cliquer sur exécuter.

Puis cliquer sur  sous le logo phpMyAdmin : ça recharge les BDs et ça fait apparaître la BD employesTP01.

Sélectionner la BD employesTP01 : ça fait apparaître sa table.





## 2. Manipulation des tables

### Consultation des attributs d'une table dans la calculette

#### Use

Pour pouvoir manipuler les tables d'une BD, il faut d'abord signaler qu'on utilise la BD

```
mysql> Use nomBD ;
```

#### Show tables

La commande « show tables » permet d'afficher la liste des tables d'une bd:

```
mysql> show tables ;
```

#### Desc nomDeLaTable

La commande « desc » permet d'afficher la liste des attributs d'une table de façon propre :

```
mysql> desc nomTable ;
```

#### Préfixe : nomBD.nomTable

On peut aussi accéder à toutes les tables en préfixant le nom de leur BD : nomBD.nomTable

```
mysql> desc nomBD.nomTable ;
```

### Création d'une table

#### Create table

La commande « create table » permet de créer une nouvelle table.

Une table est un ensemble de tuples.

Elle peut par exemple être schématiquement définie ainsi : DEPT(ND, nom, ville)

Chaque attribut du tuple a des caractéristiques, au minimum son type.

Le CREATE TABLE minimum sera donc :

```
mysql> CREATE TABLE DEPT(ND integer, nom varchar(30), ville varchar(30));
```

En général, on met en page la commande avec une ligne par attribut :

```
mysql> CREATE TABLE DEPT(  
    ND integer,  
    nom varchar(30),  
    ville varchar(30)  
);
```

### Consultation du code de création d'une table enregistré par le SGBD

#### Show create table

La commande « show create table » permet de consulter le code effectivement enregistré dans le SGBD. Ce code prend en compte les paramètres par défaut du SGBD.

```
mysql> show create table nomTable ;
```

La commande permet donc de voir les attributs

## Suppression d'une table

### Drop table

La commande « drop table » permet de supprimer une table. Attention, ça supprime aussi toutes les données !!! C'est à manier avec prudence !!!

```
mysql> drop table nomTable ;
```

On peut aussi vérifier l'existence de la table avant de la supprimer:

```
mysql> drop table if exists nomTable ;
```

Ca évite un signal d'erreur si la table n'existe pas.

### Synthèse

```
mysql> Use nomBD ;
mysql> show tables ;
mysql> desc nomTable ;
mysql> desc nomBD.nomTable ;
mysql> CREATE TABLE DEPT(
    ND integer,
    nom varchar(30),
    ville varchar(30)
);
mysql> show create table nomTable ;
mysql> drop table nomTable ;
mysql> drop table if exists nomTable ;
```

### **TP : consultation de la table EmployesTP01.sql**

Avec le fichier EmployesTP01.sql chargé précédemment, on a créé une BD nommée employesTP01 avec une table : emp.

Faites le travail dans une calculette et dans phpMyAdmin.

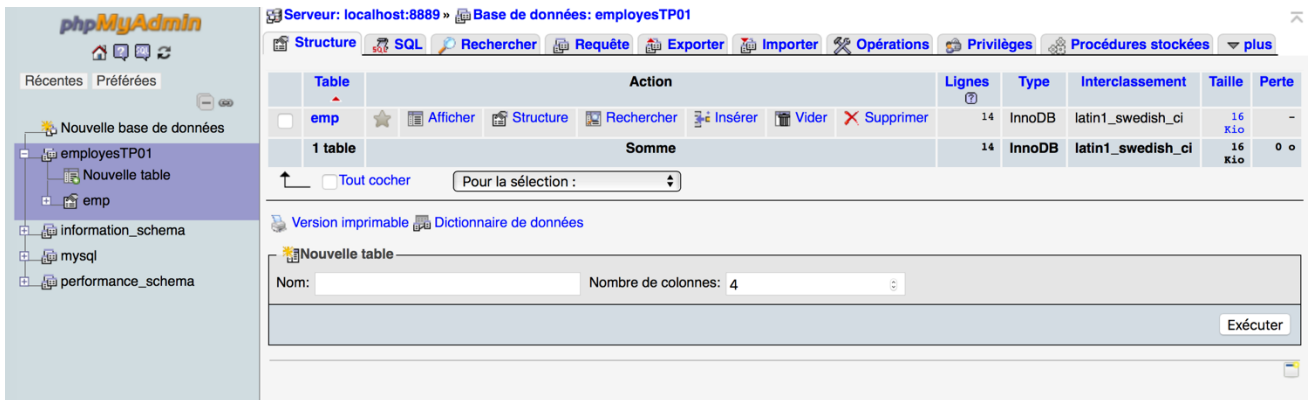
### Dans une calculette SQL

- *Afficher la table de la BD EmployesTP01.sql.*
- *Afficher la structure de la table emp.*
- *Afficher le code de création de la table emp.*

## Dans WAMP

### ➤ Afficher la table de la BD *EmployesTP01.sql*.

On sélectionne la BD dans la colonne de gauche.



Server: localhost:8889 » Base de données: employesTP01

Structure SQL Rechercher Requête Exporter Importer Opérations Privilèges Procédures stockées plus

Table	Action	Lignes	Type	Interclassement	Taille	Perte
emp	Afficher Structure Rechercher Insérer Vider Supprimer	14	InnoDB	latin1_swedish_ci	16 Kio	-
1 table		Somme			16 Kio	0 o

Tout cocher Pour la sélection :

Version imprimable Dictionnaire de données

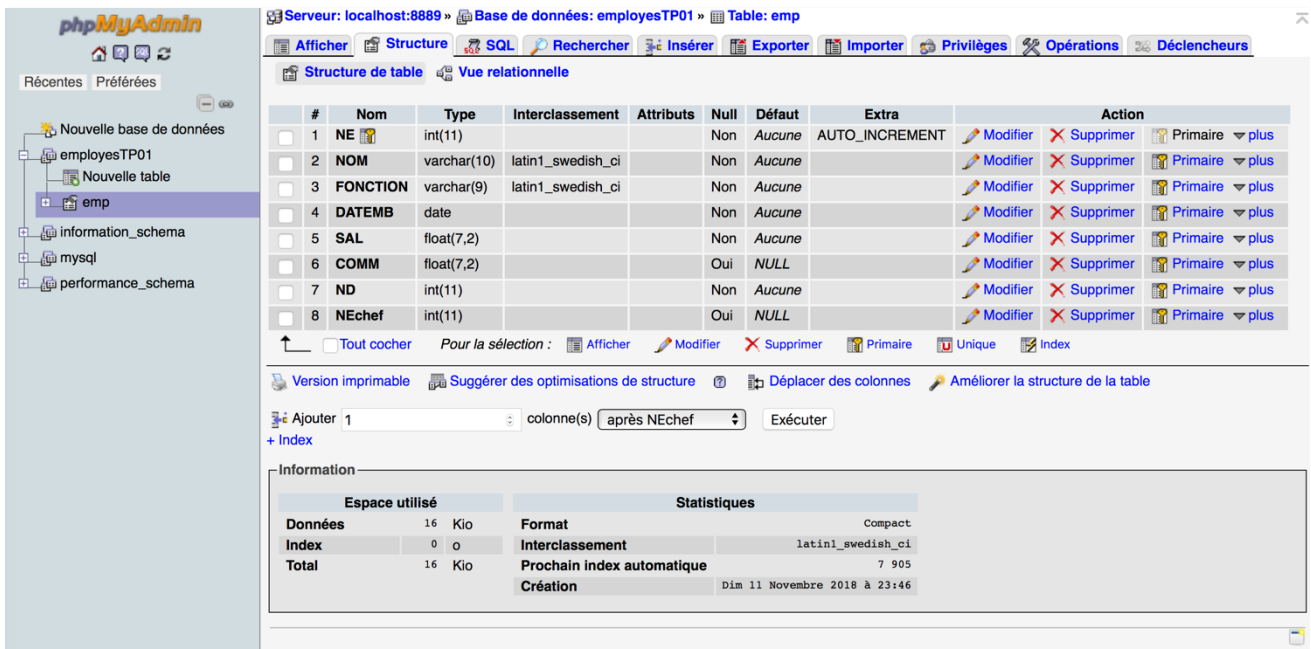
Nouvelle table

Nom: Nombre de colonnes: 4

Exécuter

### ➤ Afficher la structure de la table *emp*.

On sélectionne la BD dans la colonne de gauche puis la table concernée. L'onglet « Structure » est sélectionné par défaut.



Server: localhost:8889 » Base de données: employesTP01 » Table: emp

Afficher Structure SQL Rechercher Insérer Exporter Importer Privilèges Opérations Déclencheurs

Structure de table Vue relationnelle

#	Nom	Type	Interclassement	Attributs	Null	Défaut	Extra	Action
1	NE	int(11)			Non	Aucune	AUTO_INCREMENT	Modifier Supprimer Primaire plus
2	NOM	varchar(10)	latin1_swedish_ci		Non	Aucune		Modifier Supprimer Primaire plus
3	FONCTION	varchar(9)	latin1_swedish_ci		Non	Aucune		Modifier Supprimer Primaire plus
4	DATEMB	date			Non	Aucune		Modifier Supprimer Primaire plus
5	SAL	float(7,2)			Non	Aucune		Modifier Supprimer Primaire plus
6	COMM	float(7,2)			Oui	NULL		Modifier Supprimer Primaire plus
7	ND	int(11)			Non	Aucune		Modifier Supprimer Primaire plus
8	NEchef	int(11)			Oui	NULL		Modifier Supprimer Primaire plus

Tout cocher Pour la sélection : Afficher Modifier Supprimer Primaire Unique Index

Version imprimable Suggérer des optimisations de structure Déplacer des colonnes Améliorer la structure de la table

Ajouter 1 colonne(s) après NEchef Exécuter

+ Index

Information

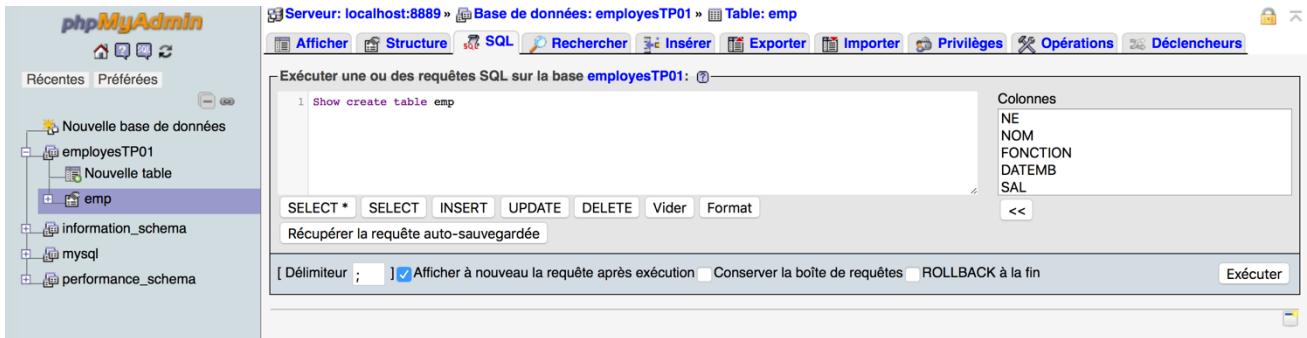
Espace utilisé		Statistiques	
Données	16 Kio	Format	Compact
Index	0 o	Interclassement	latin1_swedish_ci
Total	16 Kio	Prochain index automatique	7 905
		Création	Dim 11 Novembre 2018 à 23:46

## Afficher le code de création de la table emp.

On sélectionne la BD dans la colonne de gauche puis la table concernée.

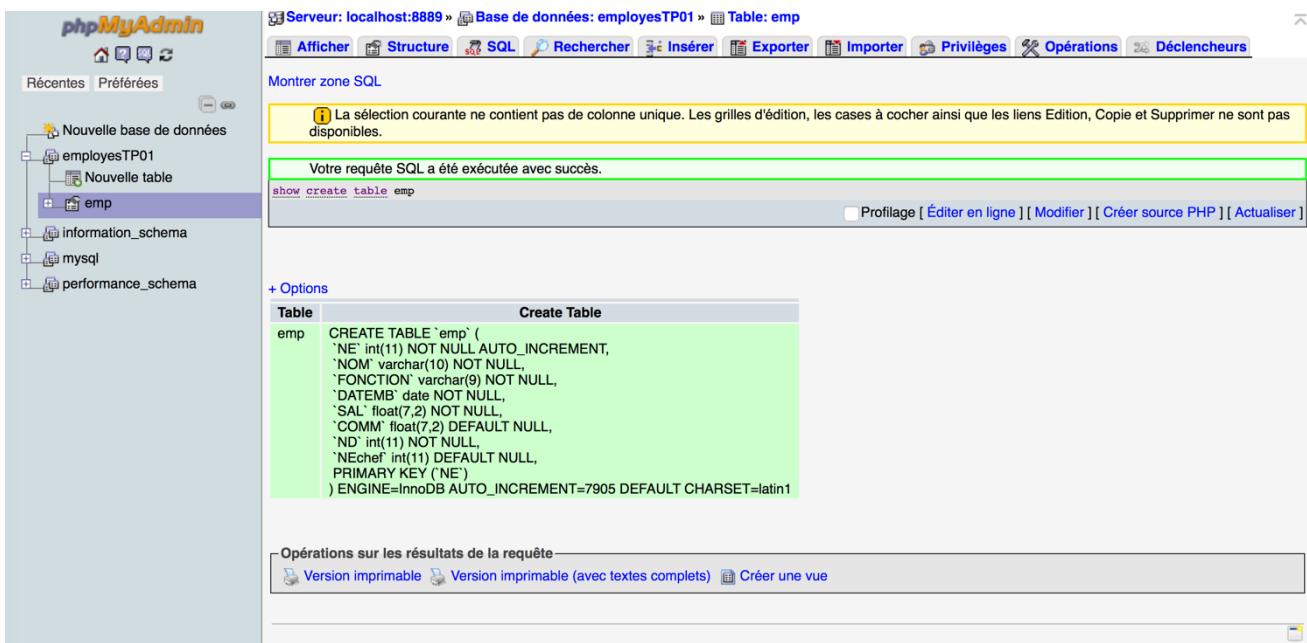
On sélectionne l'onglet SQL.

On tape la requête « show create table emp » dans la zone blanche d'édition :



On clique sur Exécuter.

Pour voir tout le résultat, il faut aller dans « Options » et choisir « texte complet »



On voit que le moteur (ENGINE) est InnoDB.

Le prochain employé aura le numéro 7905 si on utilise l'auto-incrément.

### 3. Manipulation des tuples

#### Consultation de tous les tuples d'une table dans la calculatrice SQL

##### Syntaxe SQL

```
SELECT liste d'attributs FROM NomTable
```

##### Exemple : tous les attributs

Requête : Tous les employés.

Définition ensembliste :  $RES = \{x / x \in EMP\}$

```
SELECT * FROM EMP;
```

##### Exemple : tous les attributs triés par nom

```
SELECT * FROM EMP ORDER BY NAME
```

##### Exemple : certains les attributs

Requête : Le numéro et le nom de tous les employés.

```
SELECT NE, nom FROM EMP ;
```

#### Consultation de certains tuples d'une table

##### Syntaxe SQL

```
SELECT liste d'attributs FROM NomTable  
WHERE restriction ;
```

##### Exemple

Requête : Tous les employés MANAGER.

Définition ensembliste :  $RES = \{x / x \in EMP \text{ et } x.FONCTION = \text{« MANAGER»}\}$

```
SELECT * FROM EMP  
WHERE FONCTION = « MANAGER» ;
```

#### Modification des tuples d'une table

##### Principes

On sélectionne les tuples qu'on veut modifier comme avec un SELECT

On applique la modification

##### Structure de la requête

On choisit la table de départ (l'ensemble de tuples de départ) : UPDATE nomTable

On dit quelle modification on veut faire avec un SET.

On Restreint la table de départ aux seuls tuples qu'on veut modifier avec un WHERE

## Exemple

Requête : Augmenter le salaire de l'employé 7566 de 50.

Définition ensembliste des tuples qu'on veut modifier :  $RES = \{x / x \in EMP \text{ et } x.NE = 7566 \}$

```
UPDATE EMP
SET SAL = SAL + 50
WHERE NE=10;
```

## TP : consultation et modification des tuples de la table EmployesTP01.sql

Avec le fichier EmployesTP01.sql chargé précédemment, on a créé une BD nommée employesTP01 avec une table : emp.

Faites le travail dans une calculette et dans phpMyAdmin.

### Dans une calculette SQL

- *Requête : Tous les employés et tous les attributs triés par nom, puis par FONCTION, puis par FONCTION et par NOM.*
- *Requête : Le numéro et le nom de tous les employés.*
- *Requête : Tous les employés MANAGER.*
- *Requête : Augmenter le salaire de l'employé 7566 de 50.*

### Dans WAMP

- *Requête : Tous les employés.*

On sélectionne la BD, puis la table concernée dans la BD, puis l'onglet « afficher ».

On peut trier les données en cliquant sur le nom d'un attribut. On peut trier avec plusieurs attributs en appuyant sur « maj » pour ajouter un attribut de tri.

	NE	NOM	FONCTION	DATEMB	SAL	COMM	ND	NEchef
<input type="checkbox"/>	7369	SMITH	CLERK	1980-12-17	800.00	NULL	20	7902
<input type="checkbox"/>	7499	ALLEN	SALESMAN	1981-02-20	1600.00	300.00	30	7698
<input type="checkbox"/>	7521	WARD	SALESMAN	1981-02-22	1250.00	500.00	30	7698
<input type="checkbox"/>	7566	JONES	MANAGER	1981-04-02	2975.00	NULL	20	7839
<input type="checkbox"/>	7654	MARTIN	SALESMAN	1981-09-28	1250.00	1400.00	30	7698
<input type="checkbox"/>	7698	BLAKE	MANAGER	1981-05-01	2850.00	NULL	30	7839
<input type="checkbox"/>	7782	CLARK	MANAGER	1981-06-09	2450.00	NULL	10	7839
<input type="checkbox"/>	7788	SCOTT	ANALYST	1982-12-09	3000.00	NULL	20	7566
<input type="checkbox"/>	7839	KING	PRESIDENT	1981-11-17	5000.00	NULL	10	NULL
<input type="checkbox"/>	7844	TURNER	SALESMAN	1981-09-08	1500.00	0.00	30	7698
<input type="checkbox"/>	7900	JAMES	CLERK	1981-12-03	950.00	NULL	30	7698
<input type="checkbox"/>	7902	FORD	ANALYST	1981-12-03	3000.00	NULL	20	7566
<input type="checkbox"/>	7903	ADAMS	CLERK	1983-01-12	1100.00	NULL	20	7788
<input type="checkbox"/>	7904	MILLER	CLERK	1982-01-23	1300.00	NULL	10	7782

➤ **Requête : Le numéro et le nom de tous les employés.**

On sélectionne la BD, puis la table concernée dans la BD, puis l'onglet « afficher ».

On saisie la requête et on clique sur Exécuter.

Server: localhost:8889 » Base de données: employesTP01 » Table: emp

Afficher Structure SQL Rechercher Insérer Exporter Importer Privilèges Opérations Déclencheurs

Montrer zone SQL

✓ Affichage des lignes 0 - 13 (total de 14, Traitement en 0.0016 secondes.)

```
Select NE, nom from emp
```

Profilage [ Éditer en ligne ] [ Modifier ] [ Expliquer SQL ] [ Créer source PHP ] [ A

Tout afficher | Nombre de lignes : 25 | Filtrer les lignes: Chercher dans cette table

Trier sur l'index: Aucune

+ Options

				NE	nom
<input type="checkbox"/>	Modifier	Copier	Effacer	7369	SMITH
<input type="checkbox"/>	Modifier	Copier	Effacer	7499	ALLEN
<input type="checkbox"/>	Modifier	Copier	Effacer	7521	WARD
<input checked="" type="checkbox"/>	Modifier	Copier	Effacer	7566	JONES
<input type="checkbox"/>	Modifier	Copier	Effacer	7654	MARTIN
<input type="checkbox"/>	Modifier	Copier	Effacer	7698	BLAKE
<input type="checkbox"/>	Modifier	Copier	Effacer	7782	CLARK
<input type="checkbox"/>	Modifier	Copier	Effacer	7788	SCOTT
<input type="checkbox"/>	Modifier	Copier	Effacer	7839	KING
<input type="checkbox"/>	Modifier	Copier	Effacer	7844	TURNER
<input type="checkbox"/>	Modifier	Copier	Effacer	7900	JAMES
<input type="checkbox"/>	Modifier	Copier	Effacer	7902	FORD
<input type="checkbox"/>	Modifier	Copier	Effacer	7903	ADAMS
<input type="checkbox"/>	Modifier	Copier	Effacer	7904	MILLER

Tout cocher Pour la sélection : Modifier Effacer Exporter

Tout afficher | Nombre de lignes : 25 | Filtrer les lignes: Chercher dans cette table

➤ **Requête : Tous les employés MANAGER.**

Server: localhost:8889 » Base de données: employesTP01 » Table: emp

Afficher Structure SQL Rechercher Insérer Exporter Importer Privilèges Opérations Déclencheurs

Montrer zone SQL

✓ Affichage des lignes 0 - 2 (total de 3, Traitement en 0.0014 secondes.)

```
Select NE, nom, fonction from emp where fonction='MANAGER'
```

Profilage [ Éditer en ligne ] [ Modifier ] [ Expliquer SQL ] [ Créer source PHP ] [ Actu

Tout afficher | Nombre de lignes : 25 | Filtrer les lignes: Chercher dans cette table

Trier sur l'index: Aucune

+ Options

				NE	nom	fonction
<input type="checkbox"/>	Modifier	Copier	Effacer	7566	JONES	MANAGER
<input type="checkbox"/>	Modifier	Copier	Effacer	7698	BLAKE	MANAGER
<input type="checkbox"/>	Modifier	Copier	Effacer	7782	CLARK	MANAGER

Tout cocher Pour la sélection : Modifier Effacer Exporter

Tout afficher | Nombre de lignes : 25 | Filtrer les lignes: Chercher dans cette table

Opérations sur les résultats de la requête

Version imprimable Version imprimable (avec textes complets) Exporter Afficher le graphique Créer une vue

➤ **Requête : Augmenter le salaire de l'employé 7566 de 50.**

On sélectionne l'employé 7566, puis on peut faire toutes les modifications qu'on veut.

The screenshot shows the phpMyAdmin interface for a MySQL database. The left sidebar shows the database structure with 'employeesTP01' selected, containing a table 'emp'. The main area displays the table structure and data for the selected table 'emp'.

Colonne	Type	Fonction	Null	Valeur
NE	int(11)			7566
NOM	varchar(10)			JONES
FONCTION	varchar(9)			MANAGER
DATEMB	date			1981-04-02
SAL	float(7,2)			2975.00
COMM	float(7,2)		<input checked="" type="checkbox"/>	
ND	int(11)			20
NEchef	int(11)		<input type="checkbox"/>	7839

Buttons: Afficher, Structure, SQL, Rechercher, Insérer, Exporter, Importer, Privilèges, Opérations, Déclencheurs

Buttons: Exécuter

Buttons: Sauvegarder, et ensuite, Retourner à la page précédente, Exécuter, Aperçu SQL, Réinitialiser



## 4. DATADIR et commandes de la calculette

### Datadir

La variable d'environnement DATADIR permet de définir et de connaître le répertoire de stockage des données.

On peut observer sa valeur avec un SHOW VARIABLE

```
mysql > show variable like '%datadir%'
```

### Contenu du répertoire des données

Les données de la BD sont stockées dans des répertoires et des fichiers dont le format est géré par la BD.

Il ne faut pas y toucher directement !

### Quelques commandes de la calculette

#### La commande SHOW

<http://dev.mysql.com/doc/refman/5.6/en/show.html>

help show : uniquement dans un terminal calculette.

Show databases, Show tables, Show create table, Show grants, Show variables

Show table status from nomBD ; // synthèse sur les tables de la BD

Select table\_schema, table\_name, table\_row from tables where table\_schema='maBD';

#### Fonctions d'informations

<https://dev.mysql.com/doc/refman/5.6/en/information-functions.html>

Select database();

Select user() ;

Select version() ;

Etc.

### TP

Afficher le « datadir » dans une calculette, MySQL ou WAMP.

Aller dans le « datadir ». On peut utiliser les outils de contrôle du serveur précédemment téléchargés.

## 5. Précisions sur la création des tables : CREATE TABLE

### Syntaxe SQL

```
CREATE TABLE NomTable (  
    attribut_1 type [contrainte d'intégrité],  
    attribut_2 type [contrainte d'intégrité],  
    ... ,  
    attribut_n type [contrainte d'intégrité]  
);
```

la contrainte est facultative.

La table est une coquille vide : c'est une structure. Elle permettra ensuite de créer des tuples.

### Exemple

Création de la table des employés :

```
CREATE TABLE EMPLOYES (  
    NE          integer primary key auto_increment,  
    NOM         varchar(10),  
    JOB         varchar(9),  
    DATEMB      date,  
    SAL         float(7,2),  
    COMM        float(7,2),  
    ND          integer not null  
);
```

### Les types des données

Les types des données correspondent pour la plupart aux types habituels des langages de programmation.

Pour avoir la liste et la signification de tous les types :

<http://dev.mysql.com/doc/refman/5.0/fr/column-types.html>

### « Moteurs » MyIsam et InnoDB

#### Notion de « moteur »

MySQL propose plusieurs « moteurs » pour gérer les tables. Les deux principaux sont : **MyISAM** et **InnoDB**.

Un moteur peut être vu comme un SGBD particulier pour gérer les tables.

#### Moteur par défaut : « default-storage-engine »

La variable default-storage-engine définit le moteur par défaut.

```
Show variable like '%engine%';
```

#### Moteur d'une table

➤ *Par défaut*

Quand on fait un CREATE TABLE, le moteur associé est celui par défaut

➤ **Explicitement**

A la création de la table, on peut préciser le moteur :

```
CREATE TABLE NomTable (  
    ...  
) ENGINE MyISAM;
```

On peut écrire ENGINE ou TYPE.

**Modification du moteur par défaut**

➤ **Par modification du fichier de configuration**

On peut préciser dans le fichier « my.ini » :

```
[mysqld]
```

```
default-storage-engine=MyISAM
```

Il faut alors relancer le serveur pour que la modification soit prise en compte.

**Choix d'un moteur**

Le moteur MyISAM est le moteur d'origine. Il est très rapide mais très permissif : il ne gère pas les contraintes d'intégrité référentielle.

Le moteur InnoDB gère l'intégrité référentielle.

**TP**

Vérifiez le moteur enregistré dans la BD pour la table « emp ».

Créer la table emp en MyISAM. Attention, créer cette table dans une nouvelle BD : empMyISAM.

Regarder où sont stockées les données.

A la fin, supprimer la BD empMyISAM.

## 6. Création des tuples : INSERT INTO

### Syntaxe SQL

```
INSERT INTO NomTable VALUES (val_att_1, ... , val_att_n)
```

On passe la liste des valeurs pour chaque attribut du tuple, dans l'ordre de la déclaration dans la table.

### Exemples

```
INSERT INTO EMPLOYES VALUES (7839, 'KING', 'PRESIDENT', '1981-11-17', 5000, NULL, 10);  
INSERT INTO EMPLOYES VALUES (7698, 'BLAKE', 'MANAGER', '1981-05-1', 2850, NULL, 30);  
INSERT INTO EMPLOYES VALUES (7782, 'CLARK', 'MANAGER', '1981-06-9', 2450, NULL, 10);  
etc.
```

### Auto-increment

La primary key peut être déclarée en auto-incrément dans le CREATE TABLE :

```
CREATE TABLE emp (  
    NE integer primary key auto_increment,  
    etc.
```

Dans ce cas, on peut faire un INSERT en passant la valeur NULL à la place d'un numéro explicite. Le serveur se charge de donner comme valeur le dernier numéro rentré incrémenté de 1.

### Afficher la situation de l'auto-increment

Pour connaître la situation de l'auto-incrément, on peut faire un SHOW CREATE TABLE.

```
show create table emp;
```

### TP

Vérifiez le numéro du prochain employé avec un auto-incrément pour la table « emp ».

## 7. Les utilisateurs

### Utilisateur courant

Quand on se connecte en tant que client, on est utilisateur de la BD. Un utilisateur à un nom et une machine hôte de son application cliente.

Pour connaître le nom de l'utilisateur connecté :

```
mysql > select user() ;
```

### root@localhost

Le nom d'un utilisateur a 2 parties : son nom et son hôte. L'hôte, c'est le nom de la machine à partir de laquelle l'utilisateur peut se connecter.

localhost veut dire « machine du serveur ».

root est donc défini comme utilisateur pouvant se connecter à partir de la machine du serveur.

### Tous les utilisateurs

Les utilisateurs de la BD sont listés dans la table « user » de la BD « mysql ».

On peut faire un « desc » de la table « user »

Elle a beaucoup d'attributs. On s'intéresse d'abord à « host », « user » et « password ».

```
mysql > use mysql  
mysql > Select host, user, password from user
```

ou

```
mysql > Select user, password from mysql.user
```

### Que faire si on a perdu le mot de passe de root ?

#### **Solution en passant par l'attribut password de la table mysql (serveurs 5.5 et 5.6 uniquement)**

- 1) On arrête le serveur
- 2) On redémarre le serveur en supprimant les droits : `mysqld --console --skip-grant-tables`.  
L'outil « `MYSQL_SERVEUR_sans_priv` » permet de faire ça.
- 3) On peut alors ouvrir une calculette sans préciser l'utilisateur : `mysql` directement.
- 4) L'outil « `CALCULETTE_sans_priv` » permet de faire ça.
- 5) On peut alors accéder au fichier `mysql.user`
- 6) On peut alors supprimer le mot de passe de root

```
UPDATE USER  
SET PASSWORD=''  
WHERE user='root' ;
```

- 7) On arrête le serveur
- 8) On redémarre le serveur normalement.

#### **Serveurs 5.7 et 8.0**

Ca ne fonctionne plus ! L'attribut Password n'est plus dans la table `mysql`.

## 8. Messages du serveur

Le serveur envoie des messages de validation et des messages d'erreur.

**Il faut analyser les messages du serveur!!!**

En cas d'erreur, le serveur renvoie un n° de ligne : il faut analyser ce numéro.

Exemples de message :

Query ok

Database changed

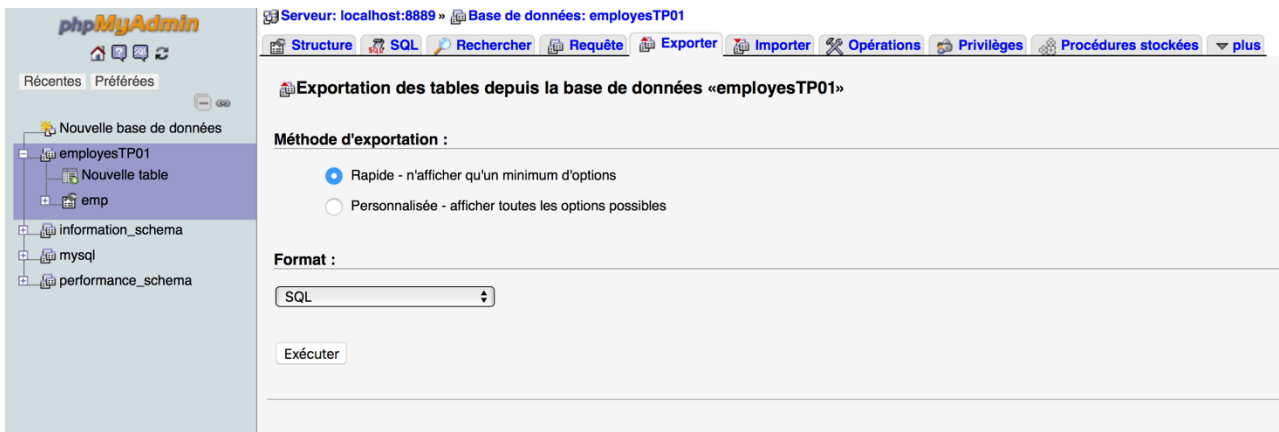
etc

On passe la liste des valeurs pour chaque attribut du tuple, dans l'ordre de la déclaration dans la table.

## 9. Sauvegarde

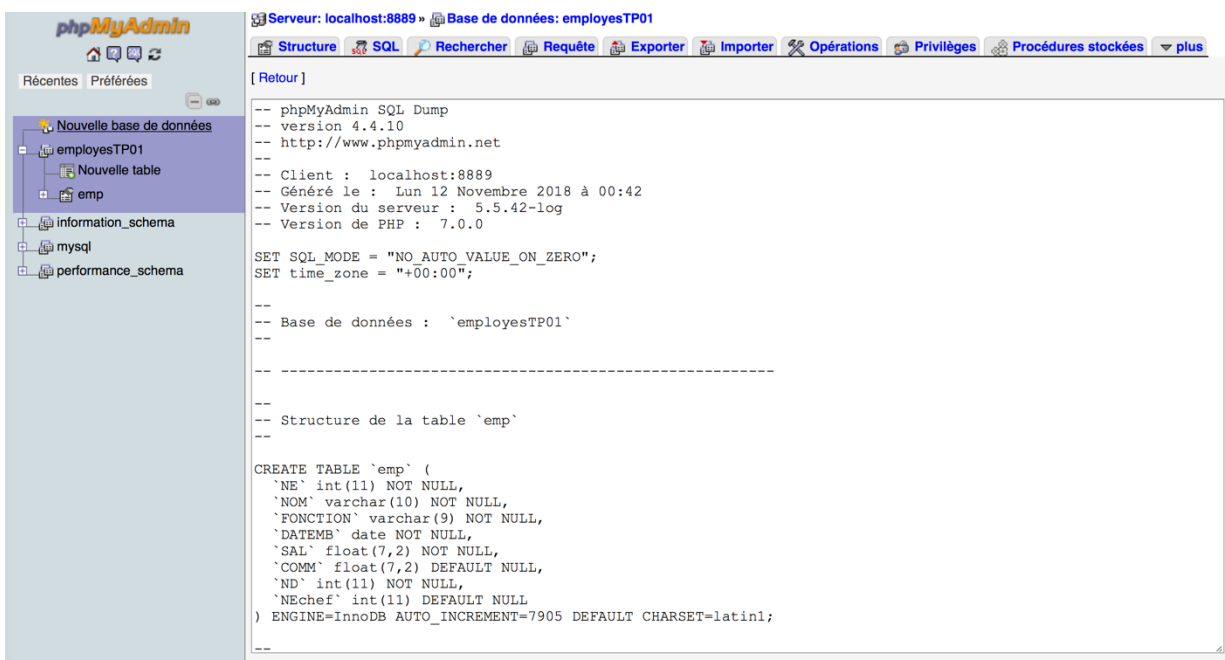
### Sauvegarder en utilisant PhpMyAdmin

#### Sélectionner la BD puis l'onglet « Exporter » :



#### Cliquez sur « Exécuter »

On récupère le code permettant de recréer la BD à l'identique :



#### Personnaliser

En choisissant l'option « Personnaliser », on peut par exemple choisir un fichier de destination pour sa sauvegarde.

#### TP

1. Sauvegarder la BD employesTP01 dans un fichier « dump.sql ».
2. Faites une sauvegarde du schéma de la base uniquement (le CREATE TABLE sans les INSERT) dans un fichier « dumpStructure.sql »
3. Consultez les fichiers produits.

## **Sauvegarder en utilisant Mysqldump**

<http://dev.mysql.com/doc/refman/5.0/fr/mysqldump.html>

Mysqldump est une commande qui permet de produire le code SQL permettant de recréer entièrement la BD à l'identique.

Une sauvegarde s'appelle aussi un « dump ».

### **Trois usages de mysqldump**

#### ➤ *Sauvegarder des BD*

```
C:\mysqldump [options] --databases DB1 [DB2 DB3...]
```

#### ➤ *Sauvegarder toutes les BD*

```
C:\mysqldump [options] --all-databases
```

### **Affichage à l'écran ou redirection dans un fichier**

```
C:\mysqldump -uroot -p nomBD
```

La commande affiche le code SQL de la BD nomBD.

```
C:\mysqldump -uroot -p nomBD > nomFichier
```

La commande écrit le code SQL de la BD nomBD dans le fichier nomFichier.

### **Usage courant de MYSQLDUMP pour sauvegarder et recharger une BD**

Commande usuelle de sauvegarde d'une BD :

```
C:\mysqldump --opt nomBD > backupFile.sql
```

L'option --opt inclut l'option --quick et l'option --lock-tables, entre autres.

Pour recharger le fichier de sauvegarde :

```
C:\mysql nomBD < backupFile.sql
```

### **Usage courant de MYSQLDUMP pour sauvegarder toutes les BDs**

Commande usuelle de sauvegarde de toutes les BD :

```
C:\mysqldump --opt --all-databases > allBD.sql
```

Pour recharger le fichier de sauvegarde de toutes les BD :

```
C:\mysql < allBD.sql
```

### **Usage spécial de MYSQLDUMP : ne sauvegarder que le schéma de la BD**

-- no-data



## 10. Import - Export

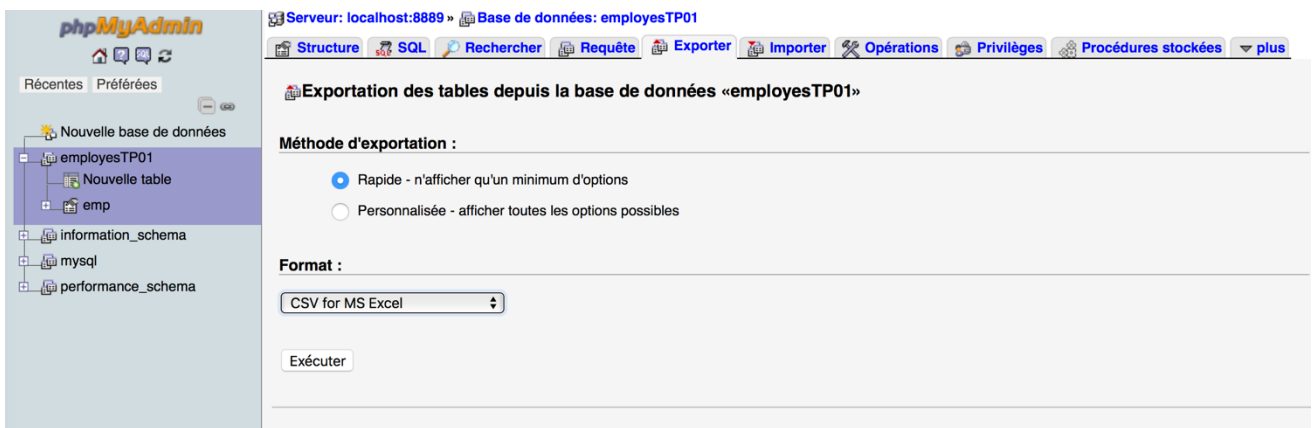
### Principes

On peut importer une BD sauvegardée ou importer des données provenant d'un fichier excel par exemple.

On peut aussi exporter des données au format excel par exemple.

### Exporter avec PhpMyAdmin

C'est comme pour une sauvegarde. Il faut juste préciser le format. Par exemple : CSV for MS Excel.

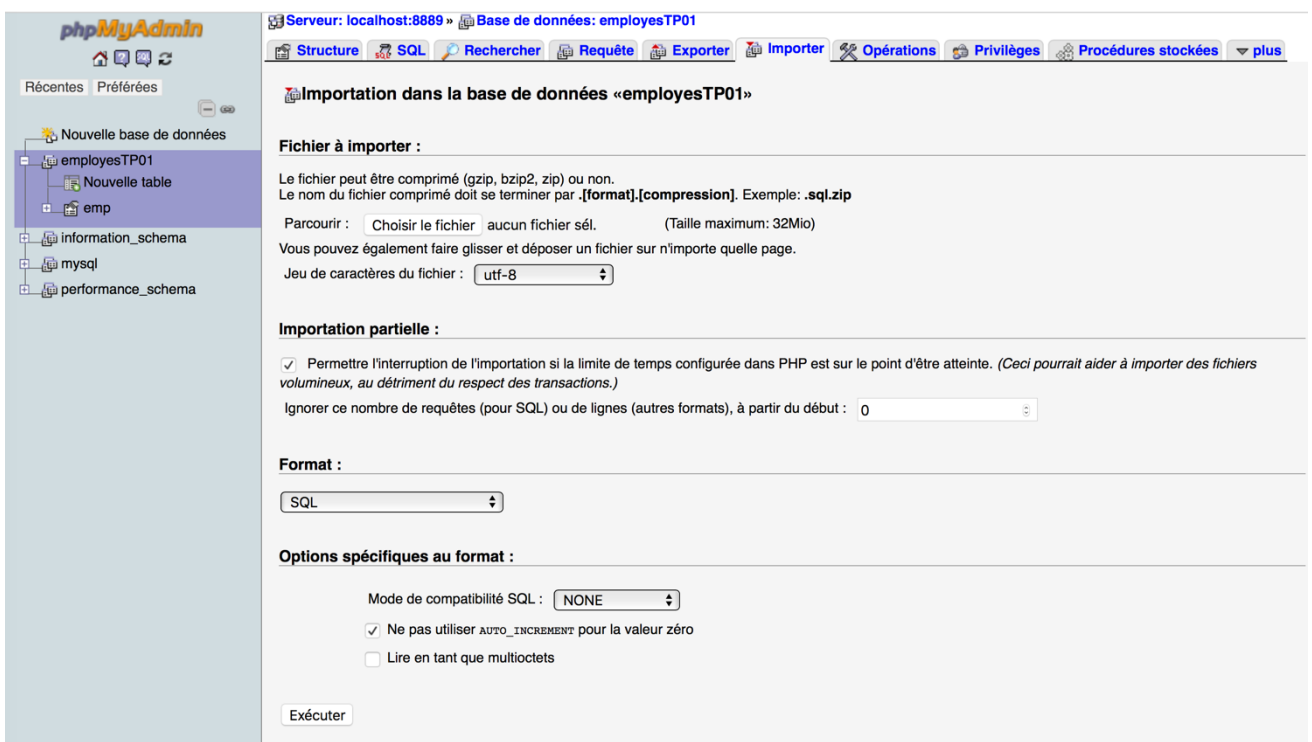


The screenshot shows the PhpMyAdmin interface. On the left, a tree view shows the database structure for 'employesTP01', including tables like 'emp', 'information\_schema', 'mysql', and 'performance\_schema'. The main panel is titled 'Exportation des tables depuis la base de données «employesTP01»'. It features a navigation bar with options like 'Structure', 'SQL', 'Rechercher', 'Requête', 'Exporter', 'Importer', 'Opérations', 'Privilèges', and 'Procédures stockées'. The 'Méthode d'exportation' section has two radio buttons: 'Rapide - n'afficher qu'un minimum d'options' (selected) and 'Personnalisée - afficher toutes les options possibles'. The 'Format' section has a dropdown menu set to 'CSV for MS Excel' and an 'Exécuter' button.

### Importer avec PhpMyAdmin

#### Sélectionner la BD puis l'onglet « Importer » :

Il reste à choisir le fichier à importer et le format (SQL ou Excel par exemple).



The screenshot shows the PhpMyAdmin interface for the import process. The left sidebar is the same as in the previous screenshot. The main panel is titled 'Importation dans la base de données «employesTP01»'. It features a navigation bar with options like 'Structure', 'SQL', 'Rechercher', 'Requête', 'Exporter', 'Importer', 'Opérations', 'Privilèges', and 'Procédures stockées'. The 'Fichier à importer' section includes instructions: 'Le fichier peut être comprimé (gzip, bzip2, zip) ou non. Le nom du fichier comprimé doit se terminer par .[format].[compression]. Exemple: .sql.zip'. It has a 'Parcourir' button, a text input for file name, and a 'Jeu de caractères du fichier' dropdown set to 'utf-8'. The 'Importation partielle' section has a checked checkbox for 'Permettre l'interruption de l'importation si la limite de temps configurée dans PHP est sur le point d'être atteinte.' and a text input for 'Ignorer ce nombre de requêtes (pour SQL) ou de lignes (autres formats), à partir du début' set to '0'. The 'Format' section has a dropdown menu set to 'SQL'. The 'Options spécifiques au format' section has a 'Mode de compatibilité SQL' dropdown set to 'NONE', a checked checkbox for 'Ne pas utiliser AUTO\_INCREMENT pour la valeur zéro', and an unchecked checkbox for 'Lire en tant que multioctets'. An 'Exécuter' button is at the bottom.

Pour importer un format Excel, il faut avoir enregistré le fichier Excel au format CSV.

## TP

1. Supprimer la BD chargée : EmployesTP01.
2. Importez la sauvegarde précédente : dump.sql.
3. Vérifiez que vous avez bien récupéré la BD.
4. Exportez les données de la table emp dans un fichier que vous serez ensuite capable de lire avec EXCEL.
5. Nommer ce fichier Excel emp2.
6. Importer les données de ce fichier emp2 dans la BD EmployesTP01. Dans ce cas, il faut préciser que les champs sont séparés par des ;

## Exporter les données avec la calculatrice : SELECT ... INTO OUTFILE

### A partir de mysql

Production d'un fichier « csv » exploitable sous Excel.

#### ➤ *Chemin absolu*

```
mysql> SELECT * INTO OUTFILE 'c:/dept.csv'  
fields terminated by ','  
FROM dept;
```

#### ➤ *Chemin relatif : DATADIR*

```
mysql> SELECT * INTO OUTFILE './dept.csv'  
fields terminated by ','  
FROM dept;
```

#### ➤ *Chemin relatif : DATADIR / BDused*

```
mysql> SELECT * INTO OUTFILE 'dept.csv'  
fields terminated by ','  
FROM dept;
```

### A partir du SE

```
C:\> mysql dbname -u user -p >output.tab
```

On passe en mode batch, l'affichage est redirigé dans le fichier output.tab

#### ➤ *Ou bien*

```
C:\> mysql dbname -u user -p -pager=cat >output.tab
```

**Remarque :** output.tab est écrasé au début de l'opération, mais toutes les commandes de la même session sont exécutées à la suite

#### ➤ *Autre usage :*

```
C:\> mysql dbname -u user -p < fichier.sql > output.tab
```

## Importer les données avec la calculette : LOAD DATA INFILE

<http://dev.mysql.com/doc/refman/5.0/fr/load-data.html>

On peut importer des données en précisant le format de séparation des champs.

Par exemple: les fichiers “csv” produit par Excel utilisent la virgule comme séparateur.

### **Attention :**

La table correspondant au fichier importé doit avoir le même schéma que celle dans laquelle les tuples sont insérés.

#### ➤ *Chemin absolu*

```
mysql> LOAD DATA INFILE 'c:/dept.csv' INTO TABLE dept
fields terminated by ',';
```

#### ➤ *Chemin relatif : DATADIR*

```
mysql> LOAD DATA INFILE './dept.csv' INTO TABLE dept
fields terminated by ',';
```

#### ➤ *Chemin relatif : DATADIR / BDused*

```
mysql> LOAD DATA INFILE 'dept.csv' INTO TABLE dept
fields terminated by ',';
```

#### ➤ *Chemin relatif : DATADIR / BD au choix*

```
mysql> LOAD DATA INFILE './empdept/dept.csv' INTO TABLE dept
fields terminated by ',';
```

Avec les options REPLACE ou IGNORE, quand une donnée rentre avec une clé primaire existant déjà, elle remplacera l'ancienne ou elle sera ignorée. Sinon, l'opération s'arrête en erreur.

```
LOAD DATA INFILE 'nomFichier' REPLACE INTO TABLE
nomTableOUTFILE
```