

# IMPLEMENTAREA UNEI BAZE DE DATE FOLOSIND MYSQL

## I. MODUL DE LUCRU

1. Transcrieti in limbaj formal structura bazei de date proiectate.
2. Deschideti serverele Apache si MySQL, respectiv programul de gestionare a BD din browserul de Internet, din pachetul de programe EasyPHP:  
<http://127.0.0.1:8080/home/mysql> sau ../phpmyadmin.
3. Scrieti comanda SQL de creare a bazei de date.
4. Creati tabelele din BD.
5. Creati conturile de utilizatori.
6. Stabiliti drepturile de acces la BD creata, de la terminalul local (localhost).
7. Introduceti valori de test in tabele.
8. Testati baza de date astfel creata.
9. Analizati tehnic BD (cale\_director, numar si tip de fisiere create, dimensiune).
10. Redefiniti drepturile de utilizatori pentru a putea fi accesata si de pe alte statii de lucru.

## II. INTRODUCERE IN MYSQL

MySQL este un limbaj de definire si manipulare a BD, respectiv a datelor (colectare, stocare, sortare, analiza).

Comanda VERSION – arata versiunea de server MySQL folosita.

Comanda USER – arata care este utilizatorul curent al serverului de BD;

MySQL permite crearea BD si a obiectelor din BD folosind comenzi cu sintaxa impusa, relativ simpla:

- Tabele
- Interogari
- Formulare
- Vederi

In aceste comenzi se folosesc diferite tipuri de date:

1. Numerice:

TINYINT pe 1B, SMALLINT pe 2B, MEDIUMINT pe 3B, INT pe 4B, LARGEINT pe 8B  
FLOAT (M,D) pe 4B, DOUBLE(M,D) pe 8B, REAL(M,D), DECIMAL[(M,D)]  
cu optiuni NOT NULL, UNIQUE, DEFAULT, UNSIGNED, ZEROFILL, AUTO\_INCREMENT.

2. data-timp:

DATE, DATETIME, TIME, YEAR(C)

3. caracter, sir de caractere

CHAR(N), VARCHAR(M) // nu pot fi folosite simultan deact la dimensiuni mici, de sub 4B.

4. BLOB (binary large object) case sensitive, folosit pentru introducerea de fisiere de tip imagine, sunet, text de dimensiuni mai mari.

5. TEXT case insensitive

6. ENUM("valoare1", "valoare2", "valoare3",...) de tip lista de cautare

### Observatii:

1. Dimensiunea implicita a bufferului folosit de serverul MySQL depinde de versiunea de program rulata (de exemplu, 24 MB), conform fisierului de configurare my.cfg. Pentru modificarea acesteia se foloseste comanda:

```
mysql>--set-variable max_allowed_packet=valoare (de exemplu, 64M).
```

2. Caracterul „underscore” \_ semnifica orice caracter si numai unul singur;

3. Caracterul „procent” % semnifica orice sir de caractere, cu orice lungime;

4. Valorile de tip sir de caractere trebuie scrise intre ghilimele, simple sau duble, atunci cand contin caractere speciale (inclusiv spatii albe).

➔ Marcheaza continuare unei comenzi SQL scrisa pe mai multe linii;

In instructiunile MySQL apar operatori:

- ✓ aritmetici
- ✓ de comparatie
- ✓ logici
- ✓ binari.

De asemenea, se pot apela si diverse functii MySQL:

- numerice,
- pentru siruri de caractere : LENGTH, CONCAT,
- TRIM, LTRIM, RTRIM – de stergere a spatiilor albe de la inceputul/sfarsitul unui sir;
- UPPER, LOWER, LIKE "\_x%";
- LAST\_INSERT\_ID(expresie) pentru a nu se modifica la reactualizarea tabelor coloanele cu numere curente s.a.

Functia PASSWORD ruleaza un algoritm de criptare pe diverse lungimi de cuvnt (16B, 41B etc ).

### III. COMENZI MYSQL UZUALE

```
CREATE DATABASE [IF NOT EXISTS] db;  
SHOW DATABASES;  
USE db;
```

```
CREATE [TEMPORARY] TABLE [IF NOT EXISTS] tabel  
    → (coloane atribute);
```

```
//      Atributele coloanelor stabilesc tipul, dimensiunile si constrangerile datelor.
```

```
LOAD DATA LOCAL INFILE "fisier.txt" INTO TABLE tabel  
    → LINES TERMINATED BY '\r\n';
```

```
//      \r\n este caracterul de terminare a liniei CR-LF pentru sisteme de operare Windows
```

```
DROP TABLE [IF EXISTS] tabel;
```

Putem face referiri la un tabel dintr-o anumita baza de date astfel:

```
db.tabel
```

sau la o anumita coloana dintr-un tabel:

```
tabel.coloana
```

Caracterul \* semnifica „orice” obiect (BD, tabel, coloana).

De exemplu, \*.\* poate sa insemne orice tabel din orice baza de date de pe server, sau orice coloana din orice tabel al bazei de date folosite.

```
SELECT coloane FROM tabel WHERE conditii ORDER BY optiuni GROUP BY valori LIMIT n;
```

Clauza LIMIT permite limitarea numarului de linii afisate odata;

```
SELECT CONCAT(coloana1, " ", coloana2) AS nume_nou_de_coloana, coloana3  
    → FROM tabel ... ;
```

```
INSERT INTO tabel VALUES (valoarea1, valoarea2 ...);
```

```
DELETE [LOW PRIORITY] FROM tabel WHERE coloana1=valoarea1 AND coloana2<=valoarea2;
```

// Optiunea LOW PRIORITY impune modificarea obiectelor atunci cand nu sunt folosite sau apelate de utilizatori.

```
GRANT privilegii coloane ON bd.tabel TO utilizator@'calculator_gazda'
```

```
→ IDENTIFIED BY "parola"
```

```
→ WITH GRANT OPTIONS; // deleaga altor utilizatori drepturile proprii.
```

```
FLUSH PRIVILEGES;
```

```
SHOW GRANTS FOR utilizator@'calculator_gazda';
```

```
REVOKE privilegii coloane ON bd.tabel TO utilizator@'calculator_gazda';
```

```
FLUSH PRIVILEGES;
```

```
SHOW GRANTS FOR utilizator@'calculator_gazda';
```

```
UPDATE [LOW PRIORITY] user SET PASSWORD=PASSWORD("parola")
```

```
-> WHERE User="utilizator";
```

Caracterul % semnifica orice utilizator sau orice calculator-gazda.

Se poate specifica numele sau adresa IP a gazdei.

Privilegiile sunt de mai multe tipuri:

1. SELECT, INSERT, CREATE, UPDATE, INDEX, DELETE, DROP, GRANT – manipularea BD.
2. SHUTDOWN, RELOAD, SHOW DATABASES, PROCESS - actiuni asupra serverului MySQL.
3. FILE - drept de acces la fisierele existente de server.

Acordarea privilegiilor trebuie facuta cu mare precautie si atentie, intrucat acestea sunt definite global pe server in tabelul user din BD mysql care se creeaza automat odata cu instalarea serverului. Acestea pot afecta si alte BD de pe server.

In mod implicit la instalarea serverului sunt create si conturi anonime cu drepturi de acces de pe orice statie din retea, pentru aplicatii comerciale sau de comunicare online de tip chat, messenger. Acestea trebuie sterse daca se doreste securizarea BD.