

Baze de date MySQL



## Introducere

Bazele de date sunt colecții de date, aranjate într-o anumită formă, asupra cărora se pot face diferite operații ca:

- Crearea bazei de date;
- Conectarea la baza de date;
- Inserarea datelor în baza de date;
- Stergerea datelor din baza de date;
- Adăugarea sau modificarea datelor;

Un tabel este alcătuit din coloane (numite câmpuri) și rânduri (numite înregistrări). Pentru implementarea unei baze de date este nevoie de un sistem de gestiune a bazelor de date (SGBD). Exemple de astfel de SGBD-uri:

- Microsoft Access;
- Visual Foxpro;
- MySQL;
- Oracle;
- etc.

MySQL este un sistem de gestiune a bazelor de date relationale, produs de compania suedeză MySQL AB și distribuit sub Licența Publică Generală GNU și este cel mai popular SGBD open source la ora actuală.

Deși este folosit foarte des împreună cu limbajul de programare PHP, cu MySQL se pot construi aplicații în orice limbaj major. Există multe scheme API disponibile pentru MySQL ce permit scrierea aplicațiilor în numeroase limbaje de programare pentru accesarea bazelor de date MySQL, cum ar fi: C, C++, C#, Java, Perl, PHP, Python, FreeBasic, etc., fiecare dintre acestea folosind un tip specific API. O interfață de tip ODBC denumită MyODBC permite altor limbaje de programare ce folosesc această interfață, să interacționeze cu bazele de date MySQL cum ar fi ASP sau Visual Basic. În sprijinul acestor limbaje de programare, unele companii produc componente de tip COM/COM+ sau .NET (pentru Windows) prin intermediul cărora respectivele limbaje să poată folosi

acest SGBD mult mai ușor decât prin intermediul sistemului ODBC. Aceste componente pot fi gratuite (ca de exemplu MyVBQL) sau comerciale.

Licența GNU GPL nu permite încorporarea MySQL în softuri comerciale; cei care doresc să facă acest lucru pot achiziționa, contra cost, o licență comercială de la compania producătoare, MySQL AB.

MySQL este componentă integrată test a platformelor LAMP sau WAMP (Linux/Windows-Apache-MySQL-PHP/Perl/Python). Popularitatea sa ca aplicație web este strâns legată de cea a PHP-ului care este adesea combinat cu MySQL și denumit Duo-ul Dinamic. În multe cărți de specialitate este precizat faptul ca MySQL este mult mai ușor de învățat și folosit decât multe din aplicațiile de gestiune a bazelor de date, ca exemplu comanda de ieșire fiind una simplă și evidentă: „exit” sau „quit”.

Pentru a administra bazele de date MySQL se poate folosi modul linie de comandă sau, prin descărcare de pe internet, o interfață grafică: MySQL Administrator (fig 3 si 4) și MySQL Query Browser. Un alt instrument de management al acestor baze de date este aplicația gratuită, scrisă în PHP, phpMyAdmin.

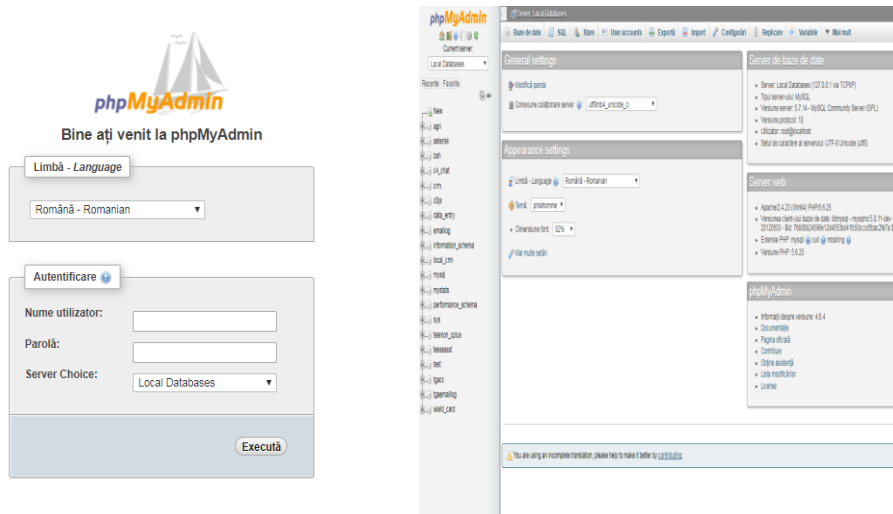


fig 3-4

MySQL poate fi rulat pe multe dintre platformele software existente: AIX, FreeBSD, GNU/Linux, Mac OS X, Net test BSD, Solaris, SunOS, Windows 9x/NT/2000/XP/Vista.

În general, o bază de date este alcătuită din una sau mai multe tabele, între acestea putându-se stabili diferite relații. Acest lucru ofera bazei

de date proprietatea de bază de date relațională. Un tabel este alcătuit din coloane test (numite câmpuri) și rânduri (numite înregistrări). Pentru implementarea unei baze de date este nevoie de un sistem de gestiune a bazelor de date(SGBD). Exemple de astfel de SGBD-uri: - Microsoft Access; - Visual Foxpro; - MySQL; - Oracle; - etc.

## Operatiuni uzuale cu bazele de date

CREATE	creaza o baza de date sau un tabel
DROP	sterge o baza de date sau un tabel
INSERT	adauga inregistrari intr-un tabel
DELETE	sterge inregistrari dintr-un tabel
UPDATE	updateaza inregistrările dintr-un tabel
SELECT	selecteaza un tabel
ALTER	alterarea unui tabel

O baza de date MySQL contine una sau mai multe tabele ce contin informatii struncturate in randuri si coloane. O tabela este un recipient ce contine informatiile stocate sub forma de randuri (rows) iar numarul de informatii ale a test cestor randuri sunt date de coloanele tabelelor (columns).

In imaginile urmatoare avem cate tabele dintr-o baza de date MySQL (fig 1) si de asemenea structura unui astfel de tabel (fig 2):

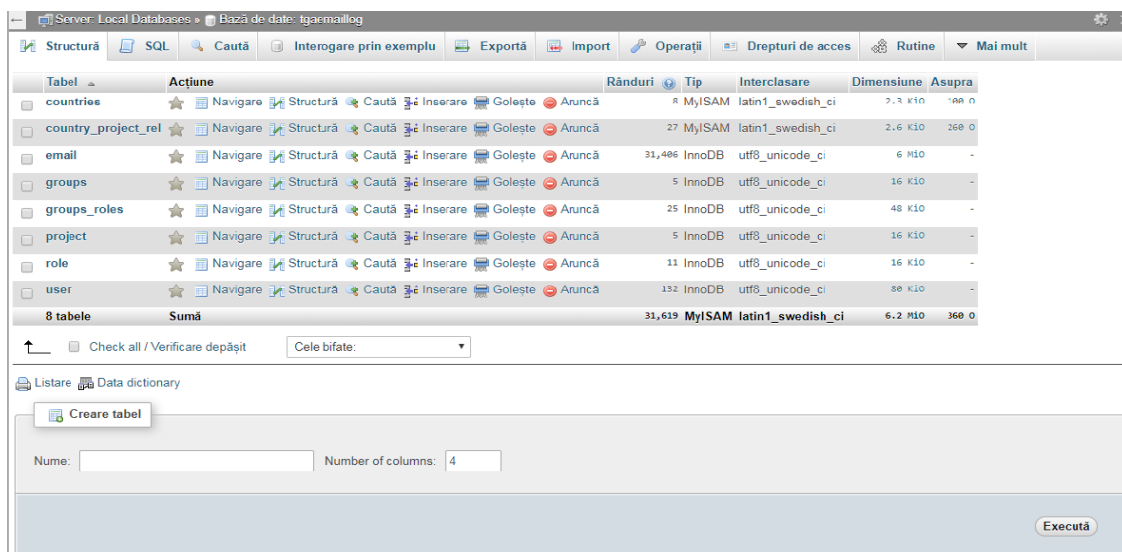


fig 1

The screenshot shows a database management interface with a table named 'user'. The table has 10 columns: id, username, email, password, salt, active, last\_login, author, group\_id, and disabled. Each column has specific properties like data type, length, and constraints. The 'id' column is the primary key and has an AUTO\_INCREMENT property. The 'password' and 'salt' columns are VARCHAR(255) and use UTF-8 encoding. The 'active' column is a TINYINT(1). The 'last\_login' column is a DATETIME. The 'author' column is a VARCHAR(255) and uses UTF-8 encoding. The 'group\_id' column is an INT(11). The 'disabled' column is an INT(2).

#	Nume	Tip	Interclasare	Proprietăți	Nul	Implicit	Comentarii	Extra	Acțiune
1	id	int(11)		Nu	None			AUTO_INCREMENT	Schimbă Aruncă Primar Unic Index Spatial Tot textul Distinct values
2	username	varchar(100)	utf8_unicode_ci	Nu	None				Schimbă Aruncă Primar Unic Index Spatial Tot textul Distinct values
3	email	varchar(200)	utf8_unicode_ci	Nu	None				Schimbă Aruncă Primar Unic Index Spatial Tot textul Distinct values
4	password	varchar(255)	utf8_unicode_ci	Nu	None				Schimbă Aruncă Primar Unic Index Spatial Tot textul Distinct values
5	salt	varchar(255)	utf8_unicode_ci	Nu	None				Schimbă Aruncă Primar Unic Index Spatial Tot textul Distinct values
6	active	tinyint(1)		Nu	None				Schimbă Aruncă Primar Unic Index Spatial Tot textul Distinct values
7	last_login	datetime		Da	NULL				Schimbă Aruncă Primar Unic Index Spatial Tot textul Distinct values
8	author	varchar(255)	utf8_unicode_ci	Nu	None				Schimbă Aruncă Primar Unic Index Spatial Tot textul Distinct values
9	group_id	int(11)		Da	NULL				Schimbă Aruncă Primar Unic Index Spatial Tot textul Distinct values
10	disabled	int(2)		Nu	0				Schimbă Aruncă Primar Unic Index Spatial Tot textul Distinct values

fig 2

Imaginea anterioara contine informatii despre structura unei tabele numita “user” si acest tabel contine urmatoarele coloane :

- id (identificatorul unic al fiecarui utilizator);
- username ( numele de utilizatorului / autentificare );
- email (email-ul utiliz test atorului);
- password (parola utilizatorului /autentificare , in momentul de fata parola este criptata sub formatul md5 );
- salt ( coloana salt poate fi folosita pentru intarirea parolei daca se doreste acest lucru dar nu este implementat in momentul de fata );
- active (coloana active poate fi folosita de administrator pentru a verifica daca utilizatorul este active sau nu in momentul de fata);
- last\_login ( in aceasta coloana se poate verifica data si ora ultimei logari in aplicatie);
- author ( coloana reprezinta numele utilizatorului cu drept de creare ce a creat acest agent/utilizator);
- group\_id( aceasta coloana contine un id din tabela “groups” ce asigneaza rolul unui utilizator ex admin);
- disabled( indica daca contul este blocat sau nu , 0 – NU si 1- Da);

## Caracteristici

Următoarea listă prezintă cele mai importante proprietăți ale MySQL. Această secțiune se adresează cititorului care are deja cunoștințe despre bazele de date relaționale. Vom folosi o terminologie din lumea bazelor de date relaționale test ale fără a ne defini exact termenii. Pe de altă parte, explicațiile ar trebui să facă posibil ca începătorii bazei de date să înțeleagă într-o oarecare măsură despre ce vorbim.

**Sistem relațional de baze de date:** Ca aproape toate celelalte sisteme de baze de date de pe piață, MySQL este un sistem de baze de date relaționale.

**Arhitectura client / server:** MySQL este un sistem client / server. Există un server de bază de date (MySQL) și în mod arbitrar mulți clienți (programe de aplicație) care comunică cu serverul; care sunt datele de interogare, salvarea modificărilor etc. Clienții pot rula pe același computer ca serverul sau pe alt computer (comunicare prin intermediul unei rețele locale sau prin Internet).

Aproape toate sistemele cunoscute de baze de date mari (Oracle, Microsoft SQL Server, etc.) sunt sisteme client / server. Acestea sunt în contrast cu sistemele de fișiere-server, care includ Microsoft Access, dBase și FoxPro. Dezavantajul decisiv al sistemelor de fișiere-server este acela că, atunci când rulează pe o rețea, acestea devin extrem de ineficiente pe măsură ce crește numărul de utilizatori.

SQL: 2003), deși cu restricții semnificative și un număr mare de extensii.

**SubSELECTs:** De la versiunea 4.1, MySQL este capabil să proceseze o interogare în formularul `SELECT * FROM table1 WHERE x IN (SELECT y FROM table2)` (Există, de asemenea, numeroase variante de sintaxă pentru subselectate).

**Proceduri stocate:** Aici avem de-a face cu codul SQL care este stocat în sistemul de baze de date.

Procedurile stocate (SP pe scurt) sunt utilizate în general pentru a simplifica anumiți pași, cum ar fi inserarea sau ștergerea unei înregistrări de date. Pentru programatorii de clienți acest lucru are avantajul că nu trebuie

să proceseze tabelele în mod direct, dar se pot baza pe SP. Ca și opinii, SP-urile ajută la administrarea proiectelor de baze de date mari. SP poate crește, de asemenea, eficiența. MySQL a sprijinit SP-urile de la versiunea 5.22412150.

**Declanșatoarele:** Declanșatoarele sunt comenzi SQL care sunt executate automat de către server în anumite operații de bază de date (INSERT, UPDATE și DELETE). MySQL a acceptat declanșatoarele într-o formă limitată de la versiunea 5.0, iar funcționalitatea suplimentară este promisă pentru versiunea 5.1.

**Interfața utilizatorului:** Există un număr de interfețe utilizator convenabile pentru administrarea unui server MySQL.

**Replicare:** Replicarea permite copierea conținutului unei baze de date (replicat) pe un număr de computere. În practică, acest lucru se întâmplă din două motive: pentru a crește protecția împotriva defecțiunilor sistemului (astfel încât, dacă un computer coboară, altul poate fi pus în funcțiune) și pentru a îmbunătăți viteza interogărilor bazei de date.

**Tranzacții:** În contextul unui sistem de baze de date, o tranzacție înseamnă executarea mai multor operațiuni de bază de date ca bloc. Sistemul de baze de date asigură fie că toate operațiile sunt executate corect, fie nici una dintre ele. Acest lucru se întâmplă chiar dacă în mijlocul unei tranzacții există o întrerupere a alimentării, se blochează computerul sau se produce un alt dezastru. Astfel, de exemplu, nu se poate întâmpla ca o sumă de bani să fie retrasă din contul A, dar nu este depusă în contul B din cauza unui anumit tip de eroare de sistem.

Tranzacțiile oferă de asemenea programatorilor posibilitatea de a întrerupe o serie de comenzi deja executate (un fel de revocare). În multe situații, acest lucru duce la o simplificare considerabilă a procesului de programare. În ciuda opiniei populare, MySQL a susținut tranzacțiile de mult timp. Trebuie să rețineți că MySQL poate stoca tabele într-o varietate de formate. Formatul de tabel implicit este numit MyISAM, iar acest format nu acceptă tranzacțiile. Dar există o serie de formate suplimentare care suportă tranzacțiile. Cel mai popular dintre acestea este InnoDB, care va fi descris extensiv în această carte.

**Constrângeri cheie externe:** Acestea sunt reguli care asigură că nu există referințe încrucișate în tabelele legate care duc la nicăieri. MySQL suportă constrângerile cheie cheie pentru tabelele InnoDB.

**Limbi de programare:** Există destul de multe interfețe API (interfețe de programare a aplicațiilor) și biblioteci pentru dezvoltarea aplicațiilor MySQL. Pentru programarea clientului puteți utiliza, printre altele, limbile C, C ++, Java, Perl, PHP, Python și Tcl.

**Viteza:** MySQL este considerat un program de baze de date foarte rapide. Această viteză a fost susținută de un număr mare de teste de referință (deși astfel de teste - indiferent de sursă - ar trebui luate în considerare cu o doză bună de scepticism).

Dezavantaje / dezavantajul MySQL:

Următoarele sunt câteva dezavantaje ale MySQL:

Versiunea MySQL mai mică de 5.0 nu suportă ROLE, COMMIT și procedura stocată. MySQL nu suportă o dimensiune foarte mare a bazei de date la fel de eficientă.

MySQL nu gestionează tranzacțiile foarte eficient și este predispus la coruperea datelor. MySQL este acuzat că nu are un bun instrument de dezvoltare și depanare comparativ cu bazele de date cu plată și de asemenea MySQL nu suportă constrângeri SQL de verificare.

## Concluziile mele

MySQL este cea mai populară bază de date open source din lume, permițând furnizarea eficientă de aplicații baze de date fiabile, de înaltă performanță și scalabile bazate pe Web și baze de date integrate. Este o bază de date compatibilă cu tranzacțiile integrate cu ACID, cu capacitate totală de comitere, răsturnare, recuperare accidentală și capabilități de blocare la nivel de linie.

MySQL oferă ușurința de utilizare, scalabilitatea și performanțele ridicate, precum și un costum complet de drivere de baze de date și instrumente vizuale pentru a ajuta dezvoltatorii și DBA-urile să-și construiască și să gestioneze aplicațiile MySQL critice pentru afaceri.



## **Bibliografie**

<http://agricolfalticeni.ro/acatrinei/programare%20web/bd/cap1.pdf>  
<https://ro.wikipedia.org/wiki/MySQL>

<http://searchitchannel.techtarget.com/feature/What-are-the-top-MYSQL-features-What-is-MYSQL>

<http://www.oracle.com/technetwork/database/mysql/index.html>