

Data Control Language (DCL) în MySQL - Creare și Ștergere Utilizatori

1 Introducere

Data Control Language (DCL) reprezintă un set de comenzi SQL utilizate pentru a gestiona permisiunile și controlul accesului la bazele de date. În MySQL, aceste comenzi sunt esențiale pentru securitate, asigurând că doar utilizatorii autorizați pot accesa sau modifica datele.

Principalele comenzi DCL includ:

- **CREATE USER:** Creează noi utilizatori în baza de date.
- **DROP USER:** Șterge utilizatori din baza de date.
- **GRANT:** Acordă permisiuni utilizatorilor.
- **REVOKE:** Revocă permisiunile acordate anterior.

Acest document se concentrează pe gestionarea utilizatorilor în MySQL, incluzând exemple de creare și ștergere a acestora.

2 Crearea Utilizatorilor

Comanda **CREATE USER** este utilizată pentru a crea noi utilizatori în baza de date.

2.1 Sintaxa Generală

```
CREATE USER 'utilizator'@'gazda'  
IDENTIFIED BY 'parola';
```

2.2 Exemple Practice

Exemplu 1: Crearea unui utilizator local

```
CREATE USER 'studentuser'@'localhost'  
IDENTIFIED BY 'parola123';
```

Acest utilizator poate accesa baza de date doar de pe gazda locală (**localhost**).

Exemplu 2: Crearea unui utilizator cu acces de la distanță

```
CREATE USER 'adminuser'@'%'  
IDENTIFIED BY 'adminpassword';
```

Simbolul % permite accesul acestui utilizator de la orice adresă IP.

Exemplu 3: Crearea unui utilizator cu acces restricționat

```
CREATE USER 'readonlyuser'@'192.168.1.100'  
IDENTIFIED BY 'readonlypassword';
```

Acest utilizator poate accesa baza de date doar de la adresa IP specificată.

3 Ștergerea Utilizatorilor

Comanda **DROP USER** este utilizată pentru a șterge utilizatori existenți din baza de date.

3.1 Sintaxa Generală

```
DROP USER 'utilizator' '@'gazda';
```

3.2 Exemple Practice

Exemplu 1: Ștergerea unui utilizator local

```
DROP USER 'studentuser' '@'localhost';
```

Exemplu 2: Ștergerea unui utilizator cu acces de la distanță

```
DROP USER 'adminuser' '@'%;
```

4 Gestionarea Utilizatorilor

În plus față de crearea și ștergerea utilizatorilor, MySQL permite modificarea utilizatorilor existenți prin intermediul comenzii **ALTER USER**.

4.1 Schimbarea Parolei

```
ALTER USER 'studentuser' '@'localhost '  
IDENTIFIED BY 'nouaparola';
```

4.2 Blocarea unui Utilizator

```
ALTER USER 'readonlyuser' '@'192.168.1.100 '  
ACCOUNT LOCK;
```

4.3 Deblocarea unui Utilizator

```
ALTER USER 'readonlyuser' '@'192.168.1.100 '  
ACCOUNT UNLOCK;
```

4.4 Exerciții

1. Creați un utilizator 'developeruser' care poate accesa baza de date doar de pe gazda locală (localhost), cu parola devpassword.
2. Creați un utilizator numit 'backupuser' care poate accesa baza de date de la orice adresă IP ('
3. Acordați permisiuni complete (ALL PRIVILEGES) utilizatorului 'adminuser' pentru baza de date universitate.
4. Revocați permisiunile de DELETE pentru utilizatorul 'developeruser' asupra tabeli Student.
5. Schimbați parola utilizatorului 'adminuser' în 'newadminpassword'.
6. Blocați accesul utilizatorului 'inactiveuser' utilizând comanda ALTER USER.

7. Creați un utilizator 'readonlyuser' care are doar permisiuni de SELECT asupra tuturor tabelor din baza de date proiect.
8. Acordați utilizatorului 'backupuser' permisiuni de SELECT și INSERT pentru tabela Curs.
9. Creați un utilizator 'testuser' care poate accesa baza de date doar de la adresa IP 192.168.0.10 și are parola 'testpassword'.
10. Revocați toate permisiunile utilizatorului 'readonlyuser' asupra bazei de date proiect.
11. Creați un utilizator 'guestuser' care poate vizualiza doar tabela Angajat, fără a avea permisiuni de modificare.
12. Ștergeți utilizatorul 'testuser' din baza de date.
13. Creați un utilizator 'manageruser' care are permisiuni de SELECT, INSERT, UPDATE și DELETE asupra bazei de date companie.
14. Revocați permisiunile de INSERT și DELETE pentru utilizatorul 'manageruser'.
15. Schimbați parola utilizatorului 'guestuser' în 'guestsecure'.
16. Creați un utilizator 'analyticsuser' care poate vizualiza și genera rapoarte din baza de date vanzari, dar nu poate modifica datele.
17. Utilizând comanda ALTER USER, deblocați utilizatorul 'inactiveuser'.
18. Acordați permisiuni de GRANT OPTION utilizatorului 'adminuser' astfel încât să poată transfera permisiuni altor utilizatori.
19. Creați un utilizator 'reportuser' care poate accesa baza de date doar în mod citire (SELECT) și poate utiliza parola 'reportpassword'.
20. Configurați utilizatorul 'developeruser' astfel încât să aibă acces doar la tabela Proiecte din baza de date dezvoltare.

5 Concluzie

Gestionarea utilizatorilor este o parte esențială a administrării bazelor de date MySQL. Comenzile **CREATE USER**, **DROP USER**, și **ALTER USER** permit administratorilor să controleze accesul și să implementeze măsuri de securitate pentru a proteja datele și resursele bazei de date.