

Lucrari de seminar: Introducere in SQL pentru MySQL

Aplicatii

1. Crearea unei baze de date

Creati o baza de date numita **biblioteca**.

Solutie:

```
— Crearea unei baze de date numite 'biblioteca'
CREATE DATABASE biblioteca;
```

2. Selectarea unei baze de date

Selectati baza de date **biblioteca** pentru utilizare.

Solutie:

```
— Selectarea bazei de date 'biblioteca' pentru a
  lucra cu ea
USE biblioteca;
```

3. Crearea unei tabele

In baza de date **biblioteca**, creati o tabela numita **carti** cu urmatoarele coloane:

- **id** INT AUTO_INCREMENT PRIMARY KEY
- **titlu** VARCHAR(255) NOT NULL
- **autor** VARCHAR(255) NOT NULL
- **an_publicare** YEAR
- **gen** VARCHAR(100)

Solutie:

```
— Crearea tabelei 'carti' cu specificatiile date
CREATE TABLE carti (
  id INT AUTO_INCREMENT PRIMARY KEY,      —
      Identificator unic pentru fiecare carte
  titlu VARCHAR(255) NOT NULL,              — Titlul
      cartii, nu poate fi NULL
  autor VARCHAR(255) NOT NULL,             — Numele
      autorului, nu poate fi NULL
  an_publicare YEAR,                        — Anul
      publicarii
  gen VARCHAR(100)                          — Genul
      literar
);
```

4. Inserarea de date

Inserati urmatoarele inregistrari in tabelul **carti**:

- (titlu: 'Mandrie si prejudecata', autor: 'Jane Austen', an_publicare: 1813, gen: 'Romantic')
- (titlu: '1984', autor: 'George Orwell', an_publicare: 1949, gen: 'Distopie')

Solutie:

— *Inserarea inregistrarilor in tabelul 'carti'*
INSERT INTO carti (titlu , autor , an_publicare ,
 gen) **VALUES**
 ('Mandrie_si_prejudecata' , 'Jane_Austen' , 1813 ,
 'Romantic') ,
 ('1984' , 'George_Orwell' , 1949 , 'Distopie');

5. Selectarea datelor

Selectati toate inregistrările din tabelul **carti**.

Solutie:

— *Selectarea tuturor inregistrarilor din tabelul 'carti'*
SELECT * FROM carti;

6. Filtrarea datelor

Selectati toate cartile scrise de 'Jane Austen'.

Solutie:

— *Selectarea cartilor unde autorul este 'Jane Austen'*
SELECT * FROM carti **WHERE** autor = 'Jane_Austen';

7. Actualizarea datelor

Modificati genul cartii '1984' la 'Science Fiction'.

Solutie:

— *Actualizarea genului pentru cartea '1984'*
UPDATE carti **SET** gen = 'Science_Fiction' **WHERE**
 titlu = '1984';

8. Stergerea datelor

Stergeti toate cartile publicate inainte de anul 1900.

Solutie:

— *Stergerea cartilor publicate inainte de 1900*
DELETE FROM carti **WHERE** an_publicare < 1900;

9. Adaugarea unei noi coloane

Adaugati o coloana numita **numar_pagini** de tip **INT** in tabelul **carti**.

Solutie:

— *Adaugarea coloanei 'numar_pagini' in tabelul 'carti'*
ALTER TABLE carti **ADD COLUMN** numar_pagini **INT**;

10. Ordine in rezultatele interogarii

Selectati toate cartile ordonate crescator dupa **an_publicare**.

Solutie:

— *Selectarea cartilor ordonate dupa anul publicarii*
SELECT * FROM carti ORDER BY an_publicare ASC;

11. Limitarea rezultatelor

Selectati primele 5 carti din tabelul **carti**.

Solutie:

— *Selectarea primelor 5 inregistrari din 'carti'*
SELECT * FROM carti LIMIT 5;

12. Functii agregate

Numarati cate carti exista in tabelul **carti**.

Solutie:

— *Numararea totalului de carti din tabel*
SELECT COUNT(*) FROM carti;

13. Gruparea datelor

Selectati numarul de carti pentru fiecare gen.

Solutie:

— *Numararea cartilor pentru fiecare gen*
SELECT gen, COUNT(*) FROM carti GROUP BY gen;

14. Conditii de grupare

Selectati genurile care au mai mult de 2 carti in tabel.

Solutie:

— *Selectarea genurilor cu mai mult de 2 carti*
**SELECT gen, COUNT(*) FROM carti
 GROUP BY gen
 HAVING COUNT(*) > 2;**

15. Utilizarea operatorului LIKE

Selectati toate cartile al caror titlu incepe cu 'The'.

Solutie:

— *Selectarea cartilor al caror titlu incepe cu 'The'*
SELECT * FROM carti WHERE titlu LIKE 'The%';

16. Utilizarea operatorului BETWEEN

Selectati toate cartile publicate intre anii 1950 si 2000.

Solutie:

— *Selectarea cartilor publicate intre 1950 si 2000*
**SELECT * FROM carti WHERE an_publicare BETWEEN
 1950 AND 2000;**

17. Crearea unei tabele noi dintr-o interogare

Creati o noua tabela numita `carti_moderne` care contine toate cartile publicate dupa anul 2000.

Solutie:

```
— Crearea tabelei 'carti_moderne' cu cartile
publicate dupa 2000
CREATE TABLE carti_moderne AS
SELECT * FROM carti WHERE an_publicare > 2000;
```

18. Unirea tabelelor (JOIN)

Creati o tabela noua numita `autori` cu urmatoarele coloane:

- `id` `INT AUTO_INCREMENT PRIMARY KEY`
- `nume` `VARCHAR(255) NOT NULL`
- `tara_origine` `VARCHAR(100)`

Inserati datele corespunzatoare autorilor din tabelul `carti`. Apoi, selectati titlurile cartilor impreuna cu tara de origine a autorilor folosind un `JOIN` intre `carti` si `autori`.

Solutie:

Crearea tabelei `autori`:

```
— Crearea tabelei 'autori'
CREATE TABLE autori (
id INT AUTO_INCREMENT PRIMARY KEY,
nume VARCHAR(255) NOT NULL,
tara_origine VARCHAR(100)
);
```

Inserarea datelor:

```
— Inserarea autorilor in tabelul 'autori'
INSERT INTO autori (nume, tara_origine) VALUES
('Jane_Austen', 'Marea_Britanie'),
('George_Orwell', 'Marea_Britanie');
```

Interogarea cu `JOIN`:

```
— Selectarea titlurilor si a tarilor de origine
folosind JOIN
SELECT carti.titlu, autori.tara_origine
FROM carti
JOIN autori ON carti.autor = autori.nume;
```

19. Utilizarea subinterogarilor

Selectati toate cartile care apartin genurilor cu mai mult de 3 carti in tabel.

Solutie:

```
— Selectarea cartilor din genurile cu peste 3
carti
SELECT * FROM carti WHERE gen IN (
SELECT gen FROM carti GROUP BY gen HAVING
COUNT(*) > 3
);
```

20. Crearea de vizualizari (VIEW)

Creati o vizualizare numita `viz_carti_autori` care contine titlurile cartilor si numele autorilor.

Solutie:

```
— Crearea vizualizarii 'viz_carti_autori'
CREATE VIEW viz_carti_autori AS
SELECT titlu , autor FROM carti;
```

21. Indexarea unei coloane

Adaugati un index pe coloana `titlu` din tabelul `carti`.

Solutie:

```
— Crearea unui index pe coloana 'titlu' pentru
performanta
CREATE INDEX idx_titlu ON carti(titlu);
```

22. Constrangeri (Constraints)

Modificati tabelul `carti` pentru a adauga o constrangere care sa nu permita valoarea `NULL` in coloana `an_publicare`.

Solutie:

```
— Modificarea coloanei 'an_publicare' pentru a
nu permite NULL
ALTER TABLE carti MODIFY an_publicare YEAR NOT
NULL;
```

23. Utilizarea functiilor SQL

Selectati titlul cartii si lungimea (numarul de caractere) titlului.

Solutie:

```
— Selectarea titlului si a lungimii acestuia
SELECT titlu , LENGTH(titlu) AS lungime_titlu FROM
carti;
```

24. Tranzactii

Explicati cum se realizeaza o tranzactie simpla in MySQL care insereaza o inregistrare in tabelul `carti` si apoi confirma modificarile.

Solutie:

```
— Inceperea tranzactiei
START TRANSACTION;

— Inserarea unei noi carti
INSERT INTO carti (titlu , autor , an_publicare ,
gen)
VALUES ( 'Marele_Gatsby' , 'F._Scott_Fitzgerald' ,
1925 , 'Roman' );

— Confirmarea tranzactiei
COMMIT;
```

25. Functia IFNULL

Selectati titlul si numarul de pagini al cartilor, afisand 'Necunoscut' daca `numar_pagini` este NULL.

Solutie:

```
— Utilizarea IFNULL pentru a inlocui valorile
NULL
SELECT titlu , IFNULL(numar_pagini , 'Necunoscut')
AS numar_pagini
FROM carti;
```

26. Stergerea unei tabele

Stergeti tabelul `carti_moderne`.

Solutie:

```
— Stergerea tabelei 'carti_moderne'
DROP TABLE carti_moderne;
```

27. Alterarea unei tabele

Redenumiti tabelul `carti` in `biblioteca_carti`.

Solutie:

```
— Redenumirea tabelei 'carti' in
'biblioteca_carti'
RENAME TABLE carti TO biblioteca_carti;
```

28. Importul si exportul de date

Descrieti cum puteti exporta tabelul `biblioteca_carti` intr-un fisier CSV si apoi cum sa importati datele dintr-un fisier CSV intr-o tabela.

Solutie:

Pentru export:

```
— Exportarea datelor din 'biblioteca_carti'
intr-un fisier CSV
SELECT * FROM biblioteca_carti INTO OUTFILE
'/path/to/file.csv'
FIELDS TERMINATED BY ',' ENCLOSED BY '"' LINES
TERMINATED BY '\n';
```

Pentru import:

```
— Importarea datelor dintr-un fisier CSV in
'biblioteca_carti'
LOAD DATA INFILE '/path/to/file.csv' INTO TABLE
biblioteca_carti
FIELDS TERMINATED BY ',' ENCLOSED BY '"' LINES
TERMINATED BY '\n';
```

29. Utilizarea functiei DATE

Selectati toate cartile si adaugati o coloana calculata care arata varsta cartii (in ani) fata de anul curent.

Solutie:

```

— Calcularea varstei cartilor
SELECT titlu, an_publicare, YEAR(CURDATE()) -
    an_publicare AS varsta
FROM biblioteca_carti;

```

30. Gestionarea utilizatorilor si permisiunilor

Creati un utilizator numit `student` cu parola `parola123` si acordati-i permisiuni de `SELECT` pe baza de date `biblioteca`.

Solutie:

```

— Crearea utilizatorului 'student' cu parola
  'parola123'
CREATE USER 'student'@'localhost' IDENTIFIED BY
  'parola123';

— Acordarea permisiunilor de SELECT pe baza de
  date 'biblioteca'
GRANT SELECT ON biblioteca.* TO
  'student'@'localhost';

```

31. Functii definite de utilizator

Creati o functie numita `calculate_age` care primeste ca parametru un an si returneaza varsta (in ani) fata de anul curent.

Solutie:

```

— Crearea functiei 'calculate_age'
DELIMITER //
CREATE FUNCTION calculate_age(an INT)
RETURNS INT DETERMINISTIC
BEGIN
RETURN YEAR(CURDATE()) - an;
END //
DELIMITER ;

— Utilizarea functiei 'calculate_age'
SELECT titlu, calculate_age(an_publicare) AS
    varsta FROM biblioteca_carti;

```