



- Lemnori 2 - Data Manipulation Language

Cele mai folosite instructiuni DML sunt:

- INSERT → inserarea înregistrărilor noi
- UPDATE → actualizarea înregistrărilor
- DELETE → ștergere înregistrări
- SELECT → extragere înregistrări

I INSERT

INSERT INTO Lectiuni (nume, descriere)
VALUES ('sectiunea 1', 'cea mai mare sectiune');

specificarea este coloanelor opțională

II UPDATE

UPDATE Lectiuni SET descriere = 'cea mai reche sectiune'
WHERE nume = 'sectiunea 1';

*omiterea clauzei
duce la actuali-
zarea tuturor
înregistrărilor*

III DELETE

DELETE FROM Lectiuni WHERE nume = 'sectiunea 1';

*omiterea clauzei
rezultă în ștergerea tuturor
înregistrărilor*

IV SELECT

SELECT * FROM Lectiuni;
SELECT cod, nume, descriere FROM Lectiuni;

*DISTINCT, dacă vrem să extragem
valorile distincte din coloană*

SELECT nume, vârstă_min FROM Atracții WHERE nume <> 'Castelul Algori';

*↳ diferit
↳ returnează doar înregistrări care îndeplinesc condițiile*

SELECT * FROM Atracții WHERE vârstă_min BETWEEN 12 AND 15;

At. să fie INT NOT BETWEEN ... AND ...;

↳ specifică un interval închis

SELECT * FROM Atracții WHERE vârstă_min IN (12, 13, 14);

*↳ care au vârsta egală cu 12, 13 sau 14
NOT IN (diferite de ...)*

- pt. a folosi glosare de căutare într-o coloană, folosim LIKE

SELECT * FROM vizitatori WHERE nume LIKE 'A%'; (numele începe cu A)

'%a'; (se termină cu a)

'%ana%'; (numele conține 'ana')

'-ma'; (numele se termină cu 'ma' și este format din 3 caractere)

! OBS. Caracterul - înlocuiește un
singur caracter
Caracterul % înlocuiește zero sau mai
multe caractere

SELECT * FROM Vizitatori WHERE nume LIKE '[ABC] %'; (începe cu unul din caracterele enumerate)
'[^ABC] %'; (nu începe cu vreunul din caractere)

SELECT * FROM Vizitatori WHERE email IS NULL (core de valoarea NULL)
IS NOT NULL (nu core nu)

alias pt. coloană

SELECT B.nume AS categorie, V.nume, V.email FROM Categorie C, Vizitatori V WHERE B.cod-c = V.cod-c;

SELECT B.nume AS categorie, V.nume, V.email FROM Categorie C INNER JOIN Vizitatori V ON B.cod-c = V.cod-c;

alias pt. tabel

- putem pune și LEFT JOIN → pt. a include și categoriile care nu au vizitatori asociați (dar nu și vizitatorii care nu aparțin unei categorii)

- putem pune și RIGHT JOIN → pt. a include vizitatorii care nu aparțin unei categorii (dar nu și categoriile care nu au vizitatori asociați)

FULL JOIN → ce ne le confirmă pe toate

SELECT COUNT(*) FROM Categorie; (ret. nr. total de înregistrări din tabelul Categorie)
↳ funcție de agregare

- la utilizăm de obicei cu GROUP BY și HAVING
- Ex: COUNT(), SUM(), AVG(), MIN(), MAX()

SELECT B.nume, COUNT(cod-v) nr. vizitatori FROM Categorie B INNER JOIN Vizitatori V ON B.cod-c = V.cod-c GROUP BY B.cod-c, B.nume; (găsim numele categoriilor și nr. de vizitatori pt. fiecare categorie care are cel puțin un vizitator)

SELECT B.nume, COUNT(cod-v) nr. vizitatori FROM Categorie B LEFT JOIN Vizitatori V ON B.cod-c = V.cod-c GROUP BY B.cod-c, B.nume; (găsim numele categoriilor și nr. de vizitatori pt. fiecare categorie)

Subinterogări

- punem o subinterogare în clauza WHERE pt. a găsi înregistrările dintr-un tabel care se potrivească cu înregistrările din alt tabel folosind JOIN

```
SELECT nume FROM Categorie  
WHERE cod-c IN (SELECT cod-c FROM Vizitatori);
```

Putem rescrie interogarea folosind operatorul EXISTS:

SELECT DISTINCT B.nume FROM Categorie B INNER JOIN Vizitatori V ON B.cod-c = V.cod-c;
↳ SELECT B.nume FROM Categorie B
WHERE EXISTS (SELECT * FROM Vizitatori V WHERE V.cod-c = B.cod-c);

SELECT B.nume FROM Categorie B WHERE NOT EXISTS (SELECT * FROM Vizitatori V WHERE V.cod-c = B.cod-c);

SELECT nume FROM Note
WHERE cod-a IN (SELECT cod-a FROM Note WHERE nota = 9);

```
SELECT nume FROM Atrocity WHERE cod_a NOT IN (SELECT cod_a FROM Note WHERE nota=9) AND cod_a IN (SELECT cod_a FROM Note);
```

```
SELECT nume FROM Atrocity  
WHERE cod_a IN (SELECT cod_a FROM Note WHERE nota=10)  
UNION ALL / EXCEPT / INTERSECT  
SELECT nume FROM Atrocity  
WHERE cod_a IN (SELECT cod_a FROM Note WHERE nota=9);
```

(numele structurilor care au primit cel puțin o dată nota 9 sau cel puțin o dată nota 10)

(numele structurilor care nu au primit niciodată nota 9 și au primit cel puțin o dată nota 10)

(numele structurilor care au primit cel puțin o dată nota 9 și cel puțin o dată nota 10)