

SQL - 1

CERERI SELECT PE O TABELA

STUD

MATR	NUME	AN	GRUPA	DATAN	LOC	TUTOR	PUNCTAJ	CODS
----	-----	--	-----	-----	-----	-----	-----	----
1456	GEORGE	4	1141A	12-MAR-82	BUCURESTI		2890	11
1325	VASILE	2	1122A	05-OCT-84	PITESTI	1456	390	11
1645	MARIA	3	1131B	17-JUN-83	PLOIESTI		1400	11
3145	ION	1	2112B	24-JAN-85	PLOIESTI	3251	1670	21
2146	STANCA	4	2141A	15-MAY-82	BUCURESTI		620	21
3251	ALEX	5	2153B	07-NOV-81	BRASOV		1570	21
2215	ELENA	2	2122A	29-AUG-84	BUCURESTI	2146	890	21
4311	ADRIAN	3	2431A	31-JUL-83	BUCURESTI		450	24
3514	FLOREA	5	2452B	03-FEB-81	BRASOV		3230	24
1925	OANA	2	2421A	20-DEC-84	BUCURESTI	4311	760	24
2101	MARIUS	1	2412B	02-SEP-85	PITESTI	3514	310	24
4705	VOICU	2	2421B	19-APR-84	BRASOV	4311	1290	24

SPEC si BURSA

CODS	NUME	DOMENIU			

11	MATEMATICA	STIINTE EXACTE			
21	GEOGRAFIE	UMANIST			
24	ISTORIE	UMANIST			
TIP			PMIN	PMAX	SUMA

FARA BURSA			0	399	
BURSA SOCIALA			400	899	100
BURSA DE STUDIU			900	1799	150
BURSA DE MERIT			1800	2499	200
BURSA DE EXCEPTIE			2500	9999	300

SINTAXA

```
SELECT [DISTINCT] lista_de_expresii
FROM nume_tabela
WHERE conditie_linie
      -- clauza optionala
ORDER BY criterii_sortare_rezultat;
      -- clauza optionala
```

EFFECT

- ◆ Se parcurg rând pe rând liniile tabelii specificate pe clauza FROM.
- ◆ Din fiecare linie conținând date pentru care condiția aflată pe clauza WHERE este adevărată va rezulta o linie în rezultatul cererii. În cazul în care WHERE lipsește, toate liniile tabelii FROM vor avea o linie corespondentă în rezultatul cererii.
- ◆ Linia de rezultat este compusă pe baza listei de expresii aflată pe clauza SELECT.

EFFECT

- ◆ Dacă există cuvântul cheie DISTINCT, din rezultat se elimină liniile duplicate.
- ◆ Înainte de a trimite rezultatul, serverul îl sortează în funcție de criteriile specificate de clauza ORDER BY.
- ◆ În cazul în care ORDER BY lipsește, liniile din rezultat sunt într-o ordine independentă de conținutul lor sau de ordinea în care ele au fost adăugate în tabelă.

REZULTAT

- ◆ Numărul coloanelor din rezultat este egal cu numărul expresiilor din lista aflată pe clauza SELECT. Aceste expresii dau și numele coloanelor din rezultat.
- ◆ În lipsa clauzei DISTINCT, numărul de linii din rezultat este egal cu numărul liniilor din tabelă care îndeplinesc condiția WHERE sau, când clauza respectivă lipsește, cu numărul total de linii din tabelă.

REZULTAT

- ◆ Evaluarea valorii de adevăr a condiției din `WHERE` se face doar pe baza datelor aflate pe linia respectivă.
- ◆ Deoarece parcurgerea liniilor specificată de o cerere `SELECT` se face după un plan de execuție generat de server, folosirea clauzei `ORDER BY` este obligatorie în cazul în care se dorește un rezultat sortat după anumite criterii.

LISTA SELECT

◆ Nume de coloane sau *

```
SELECT NUME, DOMENIU  
FROM SPEC;
```

```
SELECT *  
FROM STUD;
```

LISTA SELECT

◆ Constante:

```
SELECT 'Specializarea ', NUME,  
       ' infiintata in ', 1995  
FROM SPEC
```

LISTA SELECT

◆ Expresii aritmetice:

```
SELECT TIP, SUMA, (SUMA+20)*1.1  
FROM BURSA;
```

Functia NVL (MySQL: IFNULL)

```
SELECT TIP, SUMA,  
        NVL((SUMA+20)*1.1, 0)  
FROM BURSA;
```

LISTA SELECT

◆ Expresii concatenate:

```
SELECT 'Specializarea ' || NUME ||  
       ' are codul ', CODS
```

```
FROM SPEC;
```

Cu valori nule:

```
SELECT TIP, ' are valoarea ' || SUMA ||  
       '.Lei'
```

```
FROM BURSA;
```

LISTA SELECT

Alias de coloana:

- ◆ Nu poate fi mai lung de 30 de caractere.
- ◆ Incepe cu o literă, conține decît litere, cifre, `_`, `#` și `$` sau e pus între ghilimele (tot max. 30 caractere între ghilimele).
- ◆ Între ghilimele literele mici sunt considerate diferite de literele mari.
- ◆ Nu poate fi folosit decît în cererea curentă. Sistemul nu stochează în baza de date sau altundeva aceste nume alternative.
- ◆ Nu poate fi folosit în alte clauze ale cererii (doar în `SELECT` și `ORDER BY`).

LISTA SELECT

Alias de coloana:

```
SELECT TIP AS "Tip bursa",  
       ' are valoarea ' || SUMA ||  
       '.Lei' AS Descriere  
FROM BURSA;
```

Tip bursa	DESCRIERE
FARA BURSA	are valoarea .Lei
BURSA SOCIALA	are valoarea 100.Lei
.

LISTA SELECT

DISTINCT: Elimina liniile duplicate din rezultat:

```
SELECT CODS  
FROM STUD;
```

```
SELECT DISTINCT CODS  
FROM STUD;
```

```
SELECT DISTINCT CODS, AN  
FROM STUD;
```

CLAUZA WHERE

Sintaxa: WHERE expresie_logica

Exemplu:

```
SELECT NUME, GRUPA, CODS  
FROM STUD  
WHERE AN = 4;
```


CLAUZA WHERE

Operatori de
comparatie:

Operator	Semnificație
=	Egal
>	Mai mare
>=	Mai mare sau egal
<	Mai mic
<=	Mai mic sau egal
<>	Diferit
!=	Diferit
^=	Diferit

CLAUZA WHERE

Conditii compuse (AND, OR, NOT) si paranteze

◆ AN=2 AND PUNCTAJ>500 OR
CODS=11

◆ AN=2 AND (PUNCTAJ>500 OR
CODS=11)

CLAUZA WHERE

Operatorul BETWEEN:

Sintaxa:

expresie BETWEEN valoare_minima AND
valoare_maxima

Exemplu:

```
SELECT NUME, AN, PUNCTAJ
```

```
FROM STUD
```

```
WHERE PUNCTAJ BETWEEN 2000 AND  
4000;
```

CLAUZA WHERE

BETWEEN: Alte exemple

```
SELECT NUME, AN, PUNCTAJ  
FROM STUD
```

```
WHERE PUNCTAJ + 100 BETWEEN TUTOR -  
2000 AND TUTOR + 1000;
```

```
SELECT NUME, LOC, DATAN  
FROM STUD
```

```
WHERE LOC BETWEEN 'A' AND 'L' AND  
DATAN BETWEEN '1-JAN-82' AND '31-DEC-  
82';
```

CLAUZA WHERE

Operatorul IN:

Sintaxa:

expresie IN (val_1, val_2, ..., val_n)

Exemple:

```
SELECT NUME, AN, DATAN  
FROM STUD
```

```
WHERE TUTOR IN (1456, 2146);
```

IN ignora valorile nule din lista:

```
SELECT NUME, AN, GRUPA, TUTOR  
FROM STUD
```

```
WHERE TUTOR IN (NULL, 1456, 2146);
```

CLAUZA WHERE

NOT IN intoarce fals daca lista contine valori nule:

```
SELECT NUME, AN, GRUPA, TUTOR  
FROM STUD
```

```
WHERE TUTOR NOT IN (NULL, 1456,  
2146);
```

IN este operator derivat:

```
SELECT NUME, AN, DATAN  
FROM STUD
```

```
WHERE TUTOR=1456 OR TUTOR=2146;
```

CLAUZA WHERE

Operatorul IN. Alte exemple:

```
SELECT NUME, PUNCTAJ, CODS  
FROM STUD
```

```
WHERE PUNCTAJ + 10 IN (CODS*30+70,  
    CODS*200+700);
```

```
SELECT NUME, LOC, DATAN  
FROM STUD
```

```
WHERE LOC IN ('BUCURESTI', 'PLOIESTI') OR  
    DATAN IN ('02-SEP-85', '19-APR-84', '29-  
    AUG-84');
```

CLAUZA WHERE

Operatorul LIKE:

Sintaxa:

expresie LIKE 'SABLON' [ESCAPE
'character']

Caractere de inlocuire in sablon:

Caracter	Semnificație
----------	--------------

_	Orice caracter
---	----------------

%	Orice șir de caractere, inclusiv șirul vid
---	--

CLAUZA WHERE

Operatorul LIKE: Exemple

```
SELECT NUME, AN, GRUPA  
FROM STUD  
WHERE NUME LIKE 'A%';
```

```
SELECT NUME, GRUPA  
FROM STUD  
WHERE NUME LIKE '____';
```

```
SELECT NUME, DOMENIU  
FROM SPEC  
WHERE DOMENIU LIKE '% %';
```

CLAUZA WHERE

Operatorul LIKE: Alte exemple:

```
SELECT NUME, DOMENIU
```

```
FROM SPEC
```

```
WHERE NUME LIKE '%A%I_';
```

```
SELECT NUME||'_'||DOMENIU AS  
      NUMESIDOMENIU
```

```
FROM SPEC
```

```
WHERE NUME||'_'||DOMENIU LIKE '%\_U%'  
      ESCAPE '\'
```

CLAUZA WHERE

Operatorul LIKE pentru numere, siruri, date:

```
SELECT NUME, DATAN, PUNCTAJ  
FROM STUD
```

```
WHERE DATAN LIKE '%84' AND PUNCTAJ LIKE '%9_'
```

Valorile nule nu sunt considerate sirul vid

```
SELECT NUME, TUTOR  
FROM STUD  
WHERE TUTOR LIKE '%' OR  
       TUTOR NOT LIKE '%';
```

Sablocul se poate obtine dintr-o expresie

```
SELECT NUME, 'A' || '%' || TUTOR AS SABLON  
FROM STUD  
WHERE NUME LIKE 'A' || '%' || TUTOR;
```

CLAUZA WHERE

Operatorul LIKE va considera literele mici si mari diferite in sablon (Oracle):

```
SELECT NUME, DATAN  
FROM STUD  
WHERE DATAN LIKE '%oct%';
```

```
SELECT NUME, DATAN  
FROM STUD  
WHERE DATAN LIKE '%OCT%';
```

CLAUZA WHERE

Operatorul IS NULL:

Sintaxa:

expresie IS NULL -- iar negata este:
expresie IS NOT NULL

Valorile nule nu se pot compara cu =, <>:

```
SELECT NUME, TUTOR  
FROM STUD
```

```
WHERE TUTOR = NULL; -- fals mereu
```

```
SELECT NUME, TUTOR  
FROM STUD
```

```
WHERE TUTOR <> NULL; -- fals mereu
```

CLAUZA WHERE

Operatorul IS NULL: Exemple:

```
SELECT NUME, TUTOR
```

```
FROM STUD
```

```
WHERE TUTOR IS NULL;
```

```
SELECT NUME, TUTOR
```

```
FROM STUD
```

```
WHERE TUTOR IS NOT NULL;
```

CLAUZA ORDER BY

Sintaxa:

```
ORDER BY criteriu1 [DESC] [,criteriu2  
[DESC]...]
```

Cuvântul cheie opțional DESC (de la englezescul *descending*) specifică inversarea ordinii de sortare implicite pentru criteriul respectiv (ordinea ascendentă, crescătoare) astfel încât sortarea se face descendent (descrescător).

EFFECT

- ◆ În cazul în care ORDER BY conține mai multe criterii de sortare, **ele nu sunt echivalente** ci se iau în considerare în ordinea specificată:
- ◆ Se sortează rezultatul **după primul criteriu**
- ◆ **Pentru valori egale pentru primul criteriu** se ia în considerare al doilea criteriu
- ◆ Pentru valori egale pentru primele două criterii se ia în considerare al treilea criteriu, s.a.m.d.

ORDER BY – coloane din rezultat

```
SELECT NUME, DOMENIU, CODS  
FROM SPEC  
ORDER BY NUME;
```

```
SELECT NUME, AN, GRUPA, DATAN,  
        CODS  
FROM STUD  
ORDER BY AN DESC, NUME
```

ORDER BY – alias de coloana

```
SELECT NUME, PUNCTAJ,  
       (PUNCTAJ+20)*1.1 PMARIT  
FROM STUD  
WHERE CODS=11  
ORDER BY PMARIT;
```

ORDER BY – expresii (coloane si aliasuri)

```
SELECT NUME, PUNCTAJ, (PUNCTAJ+20)*1.1  
    PMARIT  
FROM STUD  
WHERE CODS=11  
ORDER BY (PUNCTAJ+20)*1.1;
```

```
SELECT NUME, PUNCTAJ, (PUNCTAJ+20)*1.1  
    PMARIT  
FROM STUD  
WHERE CODS=11  
ORDER BY PUNCTAJ-PMARIT;
```

ORDER BY – coloane care nu apar in rezultat

```
SELECT NUME, AN, GRUPA  
FROM STUD  
WHERE AN=2  
ORDER BY LOC DESC, (PUNCTAJ/10);
```

ORDER BY – coloane care nu apar in rezultat (1)

```
SELECT MATR, NUME, AN  
FROM STUD  
ORDER BY 3 DESC, 2;
```

```
SELECT MATR, NUME, AN  
FROM STUD  
ORDER BY 3 DESC, NUME;
```

ORDER BY – coloane care nu apar in rezultat (2)

Numarul de coloana nu se poate da printr-o expresie:

```
SELECT MATR, NUME, AN  
FROM STUD  
ORDER BY 2+1 DESC, NUME;
```

ORDER BY – Valori nule (1)

Sunt considerate mai mari decat orice
valoare (Oracle):

```
SELECT TIP, SUMA  
FROM BURSA  
ORDER BY SUMA
```

ORDER BY – Valori nule (2)

Rezultat:

TIP	SUMA
-----	-----
BURSA SOCIALA	100
BURSA DE STUDIU	150
BURSA DE MERIT	200
BURSA DE EXCEPTIE	300
FARA BURSA	

Sfarsitul capitolului

“Cereri SELECT pe o tabela”