

# BAZE DE DATE - 2

## Autor

id_autor	nume_autor
1	Petru
2	Diuma
3	Vieru
4	Lena
5	Ioana
6	Ghita
7	Colea
8	Nicu
9	Valea

## Carti

idcarte	autor	titlu	pret	cantitatea	id_autor
2-2222-222-10	Petru	GFGF Laborator Mysql-Php vbvbnb	950.00	23	1
2-2222-222-13	Vieru	MAMA	2000.00	200	3
2-2222-2222-22	Eminescu	Ghid Php	1000.00	3	2
2-2222-2222-23	Eminescu	Ghid Php	1000.00	3	2
2-2222-2222-6	Vica	Cei trei muschetari Php	930.00	40	4
2-2222-2222-8	Eminescu	Ghid Php	1000.00	3	6
2-2222-2222-9	Eminescu	Ghid Php	1000.00	3	2

# Limbajul SQL

# Limbaajul SQL

## Cereri SELECT pe o tabelă

- 1. SELECT. Sintaxa. Efect. Rezultat**
- 2. Lista SELECT**
- 3. Clauza WHERE**
- 4. Clauza ORDER BY**

# SINTAXA

**SELECT** [**DISTINCT**] lista\_de\_expresii

**FROM** nume\_tabela

**WHERE** conditie\_linie            -- clauza optionala

**ORDER BY** criterii\_sortare\_rezultat; -- clauza optionala

# SINTAXA

```
SELECT [DISTINCT | ALL] { * |  
[fieldExpression [AS newName]}  
FROM tableName [alias]  
[WHERE condition]  
[GROUP BY fieldName(s)]  
[HAVING condition]  
ORDER BY fieldName(s)
```



## EFFECT

- Se parcurg rând pe rând liniile tablei specificate în 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 tablei **FROM** vor avea o linie corespondentă în rezultatul cererii.

*Linia de rezultat este compusă pe baza listei de expresii aflată pe clauza **SELECT**.*



# EFFECT

- *Daca* exista cuvântul cheie **DISTINCT**, *din rezultat se elimina liniile duplicat.*
- *Înainte de a trimite rezultatul*, serverul îl sorteaza în functie de criteriile specificate de clauza **ORDER BY**.
- În cazul în care **ORDER BY** lipseste, *liniile din rezultat sunt într-o ordine independenta de continutul lor sau de ordinea în care ele au fost adaugate în tabela.*

# REZULTAT



<https://metanit.com/sql/mysql/6.3.php>

**Numarul coloanelor din rezultat** este egal cu numarul expresiilor din lista aflata pe clauza **SELECT**. **SELECT 1+1,5;**


**Aceste expresii dau si numele coloanelor din rezultat.** **SELECT 1+SIN(PI());**

În lipsa clauzei **DISTINCT**, numarul de linii din rezultat este egal cu numarul liniilor din tabela care îndeplinesc conditia **WHERE** sau, când clauza respectiva lipseste, cu numarul total de linii din tabela. **SELECT NOW(); CURDATE();**  
**SELECT CURRENT\_DATE();**



# REZULTAT

Evaluarea valorii de adevar a conditiei din **WHERE** se face doar pe baza datelor aflate pe linia respectiva.



Deoarece parcurgerea liniilor specificata de o cerere **SELECT** *se face dupa un plan de executie generat de server*, folosirea clauzei **ORDER BY** *este obligatorie* în cazul în care se doreste un rezultat *sortat dupa anumite criterii*.

# Limbaajul SQL

## Cereri SELECT pe o tabelă

1. SELECT. Sintaxa. Efect. Rezultat
2. **Lista SELECT**
3. Clauza WHERE
4. Clauza ORDER BY

# Exemple

Considerăm următoarele tabele:

1. **STUDENT** având structura **MATR**, NUME, AN, GRUPA, DATAN, LOC, INDRUMATOR, PUNCTAJ, CODS
2. **SPECIALIZARE** având structura **CODS**, NUME, DOMENIU
3. **BURSA** având structura **TIP**, PMIN, PMAX, SUMA

# Tabela STUDENT

MATR	NUME	AN	GRUPA	DATAN	LOC	INDRUMATOR	PUNCTAJ	CODS
1234	POPA MARCEL	1	114A	12-03-87	CHISINAU	1001	2345	1
1235	POPESCU ION	2	121B	02-04-89	ANENI	1003	1300	1
1236	AVRAM NICOLAE	1	115A	21-03-68	ANENI	1001	3000	2
1237	IONESCU MARIANA	2	116C	05-05-89	CHISINAU	1003	1234	3
1256	POPESCU GINA	3	114A	06-09-90	CAHUL	1002	3456	2

# Tabelele SPECIALIZARE si BURSA

<b>CODS</b>	<b>NUME</b>	<b>DOMENIU</b>
1	AUTOMATICA	CALCULATOARE
2	ENERGETICA	INGINERIE ELECTRICA
3	MECANICA	INGINERIE MECANICA

<b>Tip</b>	<b>Pmin</b>	<b>Pmax</b>	<b>Suma</b>
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	3999	300

# LISTA SELECT



Nume de coloane sau \*

**Exemplu 1:**

**SELECT** NUME, DOMENIU  
**FROM** SPECIALIZARE;

**Exemplu 2:**

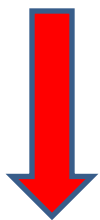
**SELECT \***  
**FROM** STUD;

idcarte	autor	titlu
2-2222-222-10	Petru GFGF	Laborator Mysql-Php vbvbnb
2-2222-222-13	Vieru	MAMA
2-2222-2222-22	Eminescu	Ghid Php
2-2222-2222-23	Eminescu	Ghid Php
2-2222-2222-6	Vica	Cei trei muschetari Php
2-2222-2222-8	Eminescu	Ghid Php
2-2222-2222-9	Eminescu	Ghid Php

id_autor	nume_autor
1	Petru
2	Diuma
3	Vieru
4	Lena
5	Ioana
6	Ghita
7	Colea
8	Nicu
9	Valea

Considerăm următoarele tabele:

1. **STUDENT** având structura **MATR**, NUME, AN, GRUPA, DATAN, LOC, INDRUMATOR, PUNCTAJ, CODS
2. **SPECIALIZARE** având structura **CODS**, NUME, DOMENIU
3. **BURSA** având structura **TIP**, PMIN, PMAX, SUMA



Constante:

Exemplu 3:

**SELECT** 'Specializarea ', NUME, ' infiintata in ',  
**1995 FROM SPECIALIZARE**

**SELECT** 'NUME ', AUTOR, ' NASCUT IN ',  
**1995 FROM CARTI**

NUME	AUTOR	NASCUT IN	1995
NUME	Petru GFGF	NASCUT IN	1995
NUME	GHITA	NASCUT IN	1995
NUME	Eminescu	NASCUT IN	1995
NUME	Eminescu	NASCUT IN	1995
NUME	Vica	NASCUT IN	1995
NUME	Eminescu	NASCUT IN	1995
NUME	Eminescu	NASCUT IN	1995

# LISTA SELECT

id_autor	nume_autor
1	Petru
2	Diuma
3	Vieru
4	Lena
5	Ioana
6	Ghita
7	Colea
8	Nicu
9	Valea

idcarte	autor	titlu
2-2222-222-10	Petru GFGF	Laborator Mysql-Php vbvbvb
2-2222-222-13	Vieru	MAMA
2-2222-2222-22	Eminescu	Ghid Php
2-2222-2222-23	Eminescu	Ghid Php
2-2222-2222-6	Vica	Cei trei muschetari Php
2-2222-2222-8	Eminescu	Ghid Php
2-2222-2222-9	Eminescu	Ghid Php

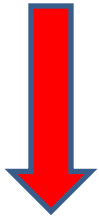
Considerăm următoarele tabele:

1. **STUDENT** având structura **MATR**, NUME, AN, GRUPA, DATAN, LOC, INDRUMATOR, PUNCTAJ, CODS

2. **SPECIALIZARE** având structura **CODS**, NUME, DOMENIU

3. **BURSA** având structura **TIP**, PMIN, PMAX, SUMA

# LISTA SELECT



## Expresii aritmetice:

Exemplu 4:

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

id_autor	nume_autor
1	Petru
2	Diuma
3	Vieru
4	Lena
5	Ioana
6	Ghita
7	Colea
8	Nicu
9	Valea

idcarte	autor	titlu
2-2222-222-10	Petru GFGF	Laborator Mysql-Php
2-2222-222-13	Vieru	MAMA
2-2222-2222-22	Eminescu	Ghid Php
2-2222-2222-23	Eminescu	Ghid Php
2-2222-2222-6	Vica	Cei trei muschetari Ph
2-2222-2222-8	Eminescu	Ghid Php
2-2222-2222-9	Eminescu	Ghid Php

Considerăm următoarele tabele:

1. **STUD** având structura MATR, NUME, AN, GRUPA, DATAN, LOC, INDRUMATOR, PUNCTAJ, CODS
2. **SPECIALIZARE** având structura CODS, NUME, DOMENIU
3. **BURSA** având structura TIP, PMIN, PMAX, SUMA





# LISTA SELECT

## Expresii concatenate:

```
SELECT CONCAT('FIRST ', 'SECOND');
```

Exemplu 5:

```
SELECT 'Specializarea ' || NUME || ' are codul ', CODS  
FROM SPECIALIZARE;
```

Exemplu 6:

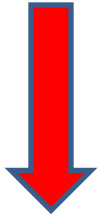
Cu valori nule:

```
SELECT TIP, ' are valoarea ' || SUMA || '.Lei'  
FROM BURSA; → SELECT autor, concat(' are valoarea  
, pret, '.Lei') FROM carti
```

idcarte	autor	titlu
2-2222-222-10	Petru GFGF	Laborator Mysql-Php vbvbyv
2-2222-222-13	Vieru	MAMA
2-2222-2222-22	Eminescu	Ghid Php
2-2222-2222-23	Eminescu	Ghid Php
2-2222-2222-6	Vica	Cei trei muschetari Php
2-2222-2222-8	Eminescu	Ghid Php
2-2222-2222-9	Eminescu	Ghid Php

Considerăm următoarele tabele:

1. **STUDENT** având structura **MATR**, NUME, AN, GRUPA, DATAN, LOC, INDRUMATOR, PUNCTAJ, CODS
2. **SPECIALIZARE** având structura **CODS**, NUME, DOMENIU
3. **BURSA** având structura **TIP**, PMIN, PMAX, SUMA



# LISTA SELECT

SELECT autor, **concat**('are valoarea ',pret, '.Lei') FROM carti

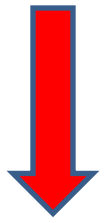
autor	concat('are valoarea ',pret, '.Lei')
Petru GFGF	are valoarea 950.00.Lei
GHITA	are valoarea 2000.00.Lei
Eminescu	are valoarea 1000.00.Lei
Eminescu	are valoarea 1000.00.Lei
Vica	are valoarea 930.00.Lei
Eminescu	are valoarea 1000.00.Lei
Eminescu	are valoarea 1000.00.Lei

idcarte	autor	titlu
2-2222-222-10	Petru GFGF	Laborator Mysql-Php vbvbyv
2-2222-222-13	Vieru	MAMA
2-2222-2222-22	Eminescu	Ghid Php
2-2222-2222-23	Eminescu	Ghid Php
2-2222-2222-6	Vica	Cei trei muschetari Php
2-2222-2222-8	Eminescu	Ghid Php
2-2222-2222-9	Eminescu	Ghid Php

id_a

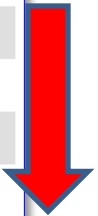
# LISTA SELECT

## Alias de coloana:



1. ***Nu poate fi*** mai lung de 30 de caractere.
2. ***Începe cu o litera***, contine numai **litere**, cifre, **\_**, **#** si **\$** sau e pus intre ghilimele (tot max. 30 caractere intre ghilimele).
3. ***Între ghilimele literele mici sunt*** considerate ***diferite de literele mari.***
4. ***Nu poate fi*** folosit decât în cererea curenta.
5. ***Sistemul nu*** stocheaza în baza de date sau altundeva aceste nume alternative.
6. ***Nu poate fi*** folosit în alte clauze ale cererii (doar in **SELECT** si **ORDER BY**)

2000.00	200	3	
1000.00	3	2	
1000.00	3	2	
chetari Php	930.00	40	4
1000.00	3	6	
1000.00	3	2	



# LISTA SELECT

## Alias de coloana:

Exemplu 7:

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

Rezultat:

Tip bursa

-----  
 FARA BURSA  
 BURSA SOCIALA

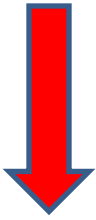
Considerăm următoarele tabele:

1. **STUDENT** având structura MATR, NUME, AN, GRUPA, DATAN, LOC, INDRUMATOR, PUNCTAJ, CODS
2. **SPECIALIZARE** având structura CODS, NUME, DOMENIU
3. **BURSA** având structura TIP, PMIN, PMAX, SUMA

DESCRIERE

-----  
 are valoarea .Lei  
 are valoarea 100.Lei

id_autor	num
1	Petr
2	Diur
3	Vier
4	Len
5	Ioan
6	Ghit
7	Cole
8	Nicu
9	Vale



# LISTA SELECT

**DISTINCT:** Elimina liniile duplicat din rezultat:

id_autor	num
1	Petr
2	Diur
3	Vier
4	Len
5	Ioan
6	Ghit
7	Cole
8	Nicu
9	Vale

Considerăm următoarele tabele:

Exemplu 8:

```
SELECT CODS  
FROM STUD;
```

1. **STUDENT** având structura MATR, NUME, AN, GRUPA, DATAN, LOC, INDRUMATOR, PUNCTAJ, CODS

2. **SPECIALIZARE** având structura CODS, NUME, DOMENIU

Exemplu 9:

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

3. **BURSA** având structura TIP, PMIN, PMAX, SUMA

Exemplu 10:

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

idcarte	autor	titlu	pret	cantitate
2-2222-222-10	Petru	GFGF Laborator Mysql-Php vbvbvb	950.00	2
2-2222-222-13	Vieru	MAMA	2000.00	20
2-2222-2222-22	Eminescu	Ghid Php	1000.00	
2-2222-2222-23	Eminescu	Ghid Php	1000.00	
2-2222-2222-6	Vica	Cei trei muschetari Php	930.00	4
2-2222-2222-8	Eminescu	Ghid Php	1000.00	
2-2222-2222-9	Eminescu	Ghid Php	1000.00	

# Limbajul SQL

## Cereri SELECT pe o tabelă

1. **SELECT. Sintaxa. Efect. Rezultat**
2. **Lista SELECT**
3. **Clauza WHERE**
4. **Clauza ORDER BY**

id_autor	nume_autor
1	Petru
2	Diuma
3	Vieru
4	Lena
5	Ioana
6	Ghita
7	Colea
8	Nicu
9	Valea

# CLAUZA WHERE

idcarte	autor	titlu
2-2222-222-10	Petru GFGF	Laborator Mysq-
2-2222-222-13	Vieru	MAMA
2-2222-2222-22	Eminescu	Ghid Php
2-2222-2222-23	Eminescu	Ghid Php
2-2222-2222-6	Vica	Cei trei muscheta
2-2222-2222-8	Eminescu	Ghid Php
2-2222-2222-9	Eminescu	Ghid Php

Sintaxa:

**WHERE** expresie\_logica

Exemplu 11:

**SELECT** NUME, GRUPA, CODS

**FROM** STUD

**WHERE** AN = 4;

Considerăm următoarele tabele:

1. **STUDENT** având structura MATR, NUME, AN, GRUPA, DATAN, LOC, INDRUMATOR, PUNCTAJ, CODS

2. **SPECIALIZARE** având structura CODS, NUME, DOMENIU

3. **BURSA** având structura TIP, PMIN, PMAX, SUMA

# CLAUZA WHERE

## Operatori de comparare

<	Mai mic
<=	Mai mic sau egal
>	Mai mare
>=	Mai mare sau egal
<>	Diferit
!=	Diferit
^=	Diferit



# CLAUZA WHERE

Conditii compuse (**AND, OR, NOT**) si paranteze

**AN=2 AND PUNCTAJ>500 OR CODS=11**

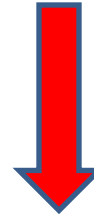
**AN=2 AND (PUNCTAJ>500 OR CODS=11)**

Considerăm următoarele tabele:

- 1. STUDENT** având structura **MATR**, NUME, AN, GRUPA, DATAN, LOC, INDRUMATOR, PUNCTAJ, CODS
- 2. SPECIALIZARE** având structura **CODS**, NUME, DOMENIU
- 3. BURSA** având structura **TIP**, PMIN, PMAX, SUMA

1	Petru
2	Diuma
3	Vieru
4	Lena
5	Ioana
6	Ghita
7	Colea
8	Nicu
9	Valea

# CLAUZA WHERE



**Operatorul BETWEEN:**

idcarte	autor	titlu
2-2222-222-10	Petru GFGF	Laborator Mysql-Php v
2-2222-222-13	Vieru	MAMA
2-2222-2222-22	Eminescu	Ghid Php
2-2222-2222-23	Eminescu	Ghid Php
2-2222-2222-6	Vica	Cei trei muschetari Php
2-2222-2222-8	Eminescu	Ghid Php
2-2222-2222-9	Eminescu	Ghid Php

Sintaxa:

expresie **BETWEEN** valoare\_minima **AND**  
valoare\_maxima

Exemplu 12:

**SELECT NUME, AN, PUNCTAJ**

**FROM STUD**

**WHERE PUNCTAJ BETWEEN 2000 AND 3999;**

Considerăm următoarele tabele:

1. **STUDENT** având structura **MATR**, NUME, AN, GRUPA, DATAN, LOC, INDRUMATOR, PUNCTAJ, CODS
2. **SPECIALIZARE** având structura **CODS**, NUME, DOMENIU
3. **BURSA** având structura **TIP**, PMIN, PMAX, SUMA

2	Diuma
3	Vieru
4	Lena
5	Ioana
6	Ghita
7	Colea
8	Nicu
9	Valea

# CLAUZA WHERE



2-2222-222-13	Vieru	MAMA
2-2222-2222-22	Eminescu	Ghid PH
2-2222-2222-23	Eminescu	Ghid PH
2-2222-2222-6	Vica	Cei trei
2-2222-2222-8	Eminescu	Ghid PH
2-2222-2222-9	Eminescu	Ghid PH

Considerăm următoarele tabele:

1. **STUDENT** având structura **MATR**, NUME, AN, GRUPA, DATAN, LOC, INDRUMATOR, PUNCTAJ, CODS
2. **SPECIALIZARE** având structura **CODS**, NUME, DOMENIU
3. **BURSA** având structura **TIP**, PMIN, PMAX, SUMA

**BETWEEN:** Alte exemple

Exemplu 13:

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

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

Exemplu 14:

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

```
WHERE LOC BETWEEN 'A' AND 'L'
AND DATAN BETWEEN '1-JAN-89' AND '31-DEC-89';
```

# CLAUZA WHERE



## Operatorul IN:

Sintaxa: **expresie IN (val\_1, val\_2, ..., val\_n)**

id_autor	nume_autor
1	Petru
2	Diuma
3	Vieru
4	Lena
5	Ioana
6	Ghita
7	Colea
8	Nicu
9	Valea

Considerăm următoarele tabele:

1. **STUDENT** având structura MATR, NUME, AN, GRUPA, DATAN, LOC, INDRUMATOR, PUNCTAJ, CODS
2. **SPECIALIZARE** având structura CODS, NUME, DOMENIU
3. **BURSA** având structura TIP, PMIN, PMAX, SUMA

Exemplu 15:

**SELECT NUME, AN, DATAN**

**FROM STUD**

**WHERE INDRUMATOR IN (1001, 1234);**

id_autor	nume_autor
1	Petru
2	Diuma
3	Vieru
4	Lena
5	Ioana
6	Ghita
7	Colea
8	Nicu
9	Valea

# CLAUZA WHERE

Considerăm următoarele tabele:

1. **STUDENT** având structura **MATR**, NUME, AN, GRUPA, DATAN, LOC, INDRUMATOR, PUNCTAJ, CODS
2. **SPECIALIZARE** având structura **CODS**, NUME, DOMENIU
3. **BURSA** având structura **TIP**, PMIN, PMAX, SUMA

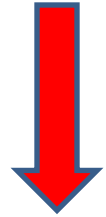
*IN ignora valorile NULL din lista:*

Exemplu 16 (pentru a lua in considerare valorile nule):

```
SELECT NUME, AN, GRUPA, INDRUMATOR
FROM STUD
WHERE INDRUMATOR IN (NULL, 1001, 1234);
```

idcarte	autor
2-2222-222-10	Petru GF
2-2222-222-13	Vieru
2-2222-2222-22	Eminescu
2-2222-2222-23	Eminescu
2-2222-2222-6	Vica
2-2222-2222-8	Eminescu
2-2222-2222-9	Eminescu

# CLAUZA WHERE



id_autor	nume_autor
1	Petru
2	Diuma
3	Vieru
4	Lena
5	Ioana
6	Ghita
7	Colea
8	Nicu
9	Valea

**NOT IN** intoarce **fals** daca lista contine valori nule:

Exemplu 17:

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

```
WHERE INDRUMATOR NOT IN (NULL, 1001, 1234);
```

	autor	titlu	pret	cantitatea	id_autor	
10	Petru	GFGF	Laborator Mysql-Php vbvbnb	950.00	23	1
13	Vieru	MAMA		2000.00	200	3
2-22	Eminescu	Ghid Php		1000.00	3	2
2-23	Eminescu	Ghid Php		1000.00	3	2
2-6	Vica	Cei trei muschetari Php		930.00	40	4
2-8	Eminescu	Ghid Php		1000.00	3	6
2-9	Eminescu	Ghid Php		1000.00	3	2

Considerăm următoarele tabele:

- 1. STUDENT** având structura **MATR**, NUME, AN, GRUPA, DATAN, LOC, INDRUMATOR, PUNCTAJ, CODS
- 2. SPECIALIZARE** având structura **CODS**, NUME, DOMENIU
- 3. BURSA** având structura **TIP**, PMIN, PMAX, SUMA

1	Petru
2	Diuma
3	Vieru
4	Lena
5	Ioana
6	Ghita
7	Colea
8	Nicu
9	Valea

# CLAUZA WHERE

idcarte	autor
2-2222-222-10	Petru
2-2222-222-13	Vieru
2-2222-2222-22	Emine
2-2222-2222-23	Emine
2-2222-2222-6	Vica
2-2222-2222-8	Emine
2-2222-2222-9	Emine

**NOT IN** intoarce **fals** daca lista contine valori  
nule:

Considerăm următoarele tabele:

- STUDENT** având structura MATR, NUME, AN, GRUPA, DATAN, LOC, INDRUMATOR, PUNCTAJ, CODS
- SPECIALIZARE** având structura CODS, NUME, DOMENIU
- BURSA** având structura TIP, PMIN, PMAX, SUMA

Exemplu 18:

**SELECT NUME, AN, DATAN**

**FROM STUD**

**WHERE INDRUMATOR=1001 OR INDRUMATOR=1234;**

1	Petru
2	Diuma
3	Vieru
4	Lena
5	Ioana
6	Ghita
7	Colea
8	Nicu
9	Valea

# CLAUZA WHERE



## Operatorul IN.

Exemplu 19:

**SELECT NUME, PUNCTAJ, CODS  
FROM STUD**

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

Exemplu 20:

**SELECT NUME, LOC, DATAN  
FROM STUD**

**WHERE LOC IN ('CHISINAU', 'ANENI')**

**OR DATAN IN ('02-SEP-1995', '19-APR-1994', '29-AUG-1994');**

idcarte	autor	titlu
2-2222-222-10	Petru GFGF	Laborator Mysql-Php vbvbyv
2-2222-222-13	Vieru	MAMA
2-2222-2222-22	Eminescu	Ghid Php
2-2222-2222-23	Eminescu	Ghid Php
2-2222-2222-6	Vica	Cei trei muschetari Php
2-2222-2222-8	Eminescu	Ghid Php
2-2222-2222-9	Eminescu	Ghid Php

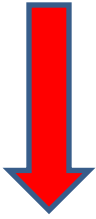
Considerăm următo

1. **STUDENT** având st  
DATAN, LOC, INDR

2. **SPECIALIZARE** având structura **CODS**, NUME, DOMENIU

3. **BURSA** având structura **TIP**, PMIN, PMAX, SUMA





# Clauza WHERE

Clauza **WHERE** poate compara valori în coloană, valori literale, expresii aritmetice sau funcții.

**WHERE** - restricționează cererea la rândurile care îndeplinesc o condiție;

*Condiție* este alcătuită din trei elemente:

1. Un nume de coloană
2. Un operator de comparație
3. Un nume de coloană, constantă sau listă de valori

# CLAUZA WHERE

## A. Operatori de comparare (operatori logici)

<	Mai mic
<=	Mai mic sau egal
>	Mai mare
>=	Mai mare sau egal
<>	Diferit
!=	Diferit
^=	Diferit

# Clauza WHERE

Exemplu:

Listează toți angajații  
care au salariul mai mic  
sau egal cu 1500.

```
SELECT nume, functie  
FROM angajati  
WHERE salariu<=1500
```

Sa se creeze urmatoarele tabele:

## Tabela DEPARTAMENTE

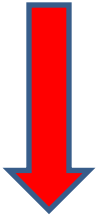
Id\_dept number(3) cheie primara (PK)  
Den\_dept varchar2(20)  
Id\_manager varchar2(3)  
Locatie varchar2(100)

## Tabela ANGAJATI

Id\_angajat number(3) cheie primara (PK)  
Id\_dept number(3) referinta (FK) la tabela DEPARTAMENTE  
Nume varchar2(40)  
Prenume varchar2(40)  
Functie varchar2(25)  
Salariu number(7)  
Id\_manager varchar2(3)  
Data\_ang date  
Comision number(5)

id_autor	nume_autor
1	Petru
2	Diuma
3	Vieru
4	Lena
5	Ioana
6	Ghita
7	Colea
8	Nicu
9	Valea

idcarte	autor	titlu
2-2222-222-10	Petru GFGF	Laborator Mysql-Php v
2-2222-222-13	Vieru	MAMA
2-2222-2222-22	Eminescu	Ghid Php
2-2222-2222-23	Eminescu	Ghid Php
2-2222-2222-6	Vica	Cei trei muschetari Php
2-2222-2222-8	Eminescu	Ghid Php
2-2222-2222-9	Eminescu	Ghid Php



# Clauza WHERE

## B. Șirurile de caractere și date calendaristice

Pentru a putea folosi în clauza **WHERE** șiruri de caractere și date calendaristice acestea trebuie introduse între ghilimele simple ( ' '), singura excepție fiind constantele numerice.

	pret	cantitatea	id_autor
orator Mysql-Php vbvbnb	950.00	23	1
MA	2000.00	200	3
d Php	1000.00	3	2
d Php	1000.00	3	2
trei muschetari Php	930.00	40	4
d Php	1000.00	3	6
d Php	1000.00	3	2

# Clauza WHERE

id_autor	num
1	Petr
2	Diur
3	Vier
4	Len
5	Ioan
6	Ghit
7	Cole
8	Nicu
9	Vale

Exemplu:

Listează toți angajații care sunt pe poziția de 'CONTABIL'.

```
SELECT *
FROM angajati
WHERE functie='CONTABIL'
```

Sa se creeze urmatoarele tabele:

## Tabela DEPARTAMENTE

Id\_dept number(3) cheie primara (PK)  
 Den\_dept varchar2(20)  
 Id\_manager varchar2(3)  
 Locatie varchar2(100)

## Tabela ANGAJATI

Id\_angajat number(3) cheie primara (PK)  
 Id\_dept number(3) referinta (FK) la tabela DEPARTAMENTE  
 Nume varchar2(40)  
 Prenume varchar2(40)  
 Functie varchar2(25)  
 Salariu number(7)  
 Id\_manager varchar2(3)  
 Data\_ang date  
 Comision number(5)

1	Petru
2	Diuma
3	Vieru
4	Lena
5	Ioana
6	Ghita
7	Colea
8	Nicu
9	Valea

# Clauza WHERE

Exemplu:

Listează toți angajații care au data de angajare 17-DEC-1990.

```
SELECT *
FROM angajati
WHERE data_ang='17-
DEC-1990'
```

idcarte	autor	titlu
2-2222-222-10	Petru GFGF	Laborator M
2-2222-222-13	Vieru	MAMA
2-2222-2222-22	Eminescu	Ghid Php
2-2222-2222-23	Eminescu	Ghid Php
2-2222-2222-6	Vica	Cei trei mu
2-2222-2222-8	Eminescu	Ghid Php
2-2222-2222-9	Eminescu	Ghid Php

Sa se creeze urmatoarele tabele:

## Tabela DEPARTAMENTE

Id\_dept number(3) cheie primara (PK)  
 Den\_dept varchar2(20)  
 Id\_manager varchar2(3)  
 Locatie varchar2(100)

## Tabela ANGAJATI

Id\_angajat number(3) cheie primara (PK)  
 Id\_dept number(3) referinta (FK) la tabela DEPARTAMENTE  
 Nume varchar2(40)  
 Prenume varchar2(40)  
 Functie varchar2(25)  
 Salariu number(7)  
 Id\_manager varchar2(3)  
 Data\_ang date  
 Comision number(5)

1	Petru
2	Diuma
3	Vieru
4	Lena
5	Ioana
6	Ghita
7	Colea
8	Nicu
9	Valea

# Clauza WHERE

Exemplu:

Listeaza toti angajatii  
care sunt in  
departamentul 10.

```
SELECT nume, functie,
salariu
FROM angajati
WHERE id_dept=10
```

Sa se creeze urmatoarele tabele:

## Tabela DEPARTAMENTE

Id\_dept number(3) cheie primara (PK)  
Den\_dept varchar2(20)  
Id\_manager varchar2(3)  
Locatie varchar2(100)

## Tabela ANGAJATI

Id\_angajat number(3) cheie primara (PK)  
Id\_dept number(3) referinta (FK) la tabela DEPARTAMENTE  
Nume varchar2(40)  
Prenume varchar2(40)  
Functie varchar2(25)  
Salariu number(7)  
Id\_manager varchar2(3)  
Data\_ang date  
Comision number(5)

idcarte	autor	titlu
2-2222-222-10	Petru GFGF	Laborator Mysq-
2-2222-222-13	Vieru	MAMA
2-2222-2222-22	Eminescu	Ghid Php
2-2222-2222-23	Eminescu	Ghid Php
2-2222-2222-6	Vica	Cei trei muscheta
2-2222-2222-8	Eminescu	Ghid Php
2-2222-2222-9	Eminescu	Ghid Php

# Clauza WHERE

```
SELECT DATE_FORMAT(CURDATE(), '%d/%m/%Y')
```



```
SELECT CURDATE()
```

Show all | Num

+ Options

CURDATE()

2021-11-04



Atenție! /IN MYSQL/

*Contează dacă caracterele sunt scrise cu litere mari sau cu litere mici.*

Datele calendaristice sunt înregistrate în baza de date într-un *format numeric intern*:

**secol, an, luna, ziua, ora, minute, secunde**

Formatul de afișare este: **DD-MON-RR**.

Acesta poate fi schimbat (vedeți în alt curs).

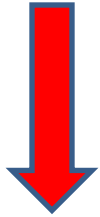


# Clauza WHERE

## C. Condiții de comparare

În **SQL** sunt 4 operatori care pot fi folosiți pentru toate tipurile de date:

Operator	Semnificație
<b>BETWEEN...AND...</b>	între 2 valori (inclusiv)
<b>IN (lista)</b>	compară cu o listă de valori
<b>LIKE</b>	compară cu un model de tip caracter
<b>IS NULL</b>	este o valoare nulă

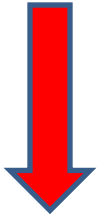


# Clauza WHERE

## 1. BETWEEN...AND...

Condiția **BETWEEN** poate fi folosită pentru a selecta rânduri pe baza unui interval de valori (conținut în condiție).

Intervalul este inclusiv, are o limită inferioară și o limită superioară și ***neapărat prima specificată trebuie să fie limita inferioară.***



# Clauza WHERE

## Exemplu:

Listează toți angajații care au salariul între 1000 și 2000.

```
SELECT *  
FROM angajati  
WHERE salariu BETWEEN  
1000 AND 2000
```

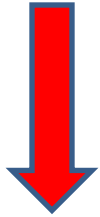
Sa se creeze urmatoarele tabele:

### Tabela DEPARTAMENTE

Id\_dept number(3) cheie primara (PK)  
Den\_dept varchar2(20)  
Id\_manager varchar2(3)  
Locatie varchar2(100)

### Tabela ANGAJATI

Id\_angajat number(3) cheie primara (PK)  
Id\_dept number(3) referinta (FK) la tabela DEPARTAMENTE  
Nume varchar2(40)  
Prenume varchar2(40)  
Functie varchar2(25)  
Salariu number(7)  
Id\_manager varchar2(3)  
Data\_ang date  
Comision number(5)



# Clauza WHERE

## 2. IN

Condiția **IN** testează valorile dintr-o listă specificată.

Pot fi folosite orice tipuri de date, cu precizarea că pentru *șirurile de caractere și date calendaristice trebuie folosite ghilimelele simple* (' ').

# Clauza WHERE

Exemplu:

Listează toți angajații care au salariul în lista (3631, 1432, 4000, 5000).

```
SELECT nume, functie, salariu  
FROM angajati  
WHERE salariu IN (3631,  
1432, 4000, 5000)
```

id_autor	nume_autor
1	Petru
2	Diuma
3	Vieru
4	Lena
5	Ioana
6	Ghita
7	Colea
8	Nicu
9	Valea

Sa se creeze urmatoarele tabele:

## Tabela DEPARTAMENTE

Id\_dept number(3) cheie primara (PK)  
Den\_dept varchar2(20)  
Id\_manager varchar2(3)  
Locatie varchar2(100)

## Tabela ANGAJATI

Id\_angajat number(3) cheie primara (PK)  
Id\_dept number(3) referinta (FK) la tabela DEPARTAMENTE  
Nume varchar2(40)  
Prenume varchar2(40)  
Functie varchar2(25)  
Salariu number(7)  
Id\_manager varchar2(3)  
Data\_ang date  
Comision number(5)

idcarte	autor	titlu
2-2222-222-10	Petru	GFGF Laborator Mysql-Php v
2-2222-222-13	Vieru	MAMA
2-2222-2222-22	Eminescu	Ghid Php
2-2222-2222-23	Eminescu	Ghid Php
2-2222-2222-6	Vica	Cei trei muschetari Php
2-2222-2222-8	Eminescu	Ghid Php
2-2222-2222-9	Eminescu	Ghid Php

# Clauza WHERE

Exemplu:

Listează toți angajații care au numele în lista ('POPA', 'IONESCU', 'POPESCU').

```
SELECT *  
FROM angajati  
WHERE nume IN ('POPA',  
'IONESCU', 'POPESCU')
```

Sa se creeze urmatoarele tabele:

## Tabela DEPARTAMENTE

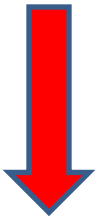
Id\_dept number(3) cheie primara (PK)  
Den\_dept varchar2(20)  
Id\_manager varchar2(3)  
Locatie varchar2(100)

## Tabela ANGAJATI

Id\_angajat number(3) cheie primara (PK)  
Id\_dept number(3) referinta (FK) la tabela DEPARTAMENTE  
Nume varchar2(40)  
Prenume varchar2(40)  
Funcctie varchar2(25)  
Salariu number(7)  
Id\_manager varchar2(3)  
Data\_ang date  
Comision number(5)

id_autor	nume_autor
1	Petru
2	Diuma
3	Vieru
4	Lena
5	Ioana
6	Ghita
7	Colea
8	Nicu
9	Valea

idcarte	autor	titlu
2-2222-222-10	Petru	GFGF Laborator Mysql-Php v
2-2222-222-13	Vieru	MAMA
2-2222-2222-22	Eminescu	Ghid Php
2-2222-2222-23	Eminescu	Ghid Php
2-2222-2222-6	Vica	Cei trei muschetari Php
2-2222-2222-8	Eminescu	Ghid Php
2-2222-2222-9	Eminescu	Ghid Php



# Clauza WHERE

## 3. LIKE

Dacă nu se cunoaște valoarea exactă căutată, cu ajutorul condiției **LIKE** putem să selectăm rândurile care se potrivesc cu un model specificat de caractere.

Operația de căutare după un model poate fi asemănată cu o căutare "**wildcard**".

Pentru construirea modelului șirului căutat pot fi folosite 2 simboluri:

**%** : orice secvență de 0 sau mai multe caractere

**\_** : un singur caracter (oarecare)

***Aceste simboluri pot fi folosite în orice combinație de caractere literale.***

id_autor	nume_autor
1	Petru
2	Diuma
3	Vieru
4	Lena
5	Ioana
6	Ghita
7	Colea
8	Nicu
9	Valea

# Clauza WHERE

## Exemplu:

Listează toți angajații  
al căror nume începe cu  
litera S.

```
SELECT *
FROM angajati
WHERE nume LIKE 'S%'
```

idcarte	autor	titlu
2-2222-222-10	Petru GFGF	Laborator Mys
2-2222-222-13	Vieru	MAMA
2-2222-2222-22	Eminescu	Ghid Php
2-2222-2222-23	Eminescu	Ghid Php
2-2222-2222-6	Vica	Cei trei musch
2-2222-2222-8	Eminescu	Ghid Php
2-2222-2222-9	Eminescu	Ghid Php

Sa se creeze urmatoarele tabele:

### Tabela DEPARTAMENTE

```
Id_dept number(3) cheie primara (PK)
Den_dept varchar2(20)
Id_manager varchar2(3)
Locatie varchar2(100)
```

### Tabela ANGAJATI

```
Id_angajat number(3) cheie primara (PK)
Id_dept number(3) referinta (FK) la tabela DEPARTAMENTE
Nume varchar2(40)
Prenume varchar2(40)
Functie varchar2(25)
Salariu number(7)
Id_manager varchar2(3)
Data_ang date
Comision number(5)
```



id_autor	nume_autor
1	Petru
2	Diuma
3	Vieru
4	Lena
5	Ioana
6	Ghita
7	Colea
8	Nicu
9	Valea

# Clauza WHERE

Exemplu:

Listează toți angajații care au numele de 4 caractere.

```
SELECT nume
FROM angajati
WHERE nume LIKE '____'
```

idcarte	autor	titlu
2-2222-222-10	Petru GFGF	Laborator
2-2222-222-13	Vieru	MAMA
2-2222-2222-22	Eminescu	Ghid Php
2-2222-2222-23	Eminescu	Ghid Php
2-2222-2222-6	Vica	Cei trei m
2-2222-2222-8	Eminescu	Ghid Php
2-2222-2222-9	Eminescu	Ghid Php

Sa se creeze urmatoarele tabele:

## Tabela DEPARTAMENTE

Id\_dept number(3) cheie primara (PK)  
 Den\_dept varchar2(20)  
 Id\_manager varchar2(3)  
 Locatie varchar2(100)

## Tabela ANGAJATI

Id\_angajat number(3) cheie primara (PK)  
 Id\_dept number(3) referinta (FK) la tabela DEPARTAMENTE  
 Nume varchar2(40)  
 Prenume varchar2(40)  
 Functie varchar2(25)  
 Salariu number(7)  
 Id\_manager varchar2(3)  
 Data\_ang date  
 Comision number(5)

id_autor	nume_autor
1	Petru
2	Diuma
3	Vieru
4	Lena
5	Ioana
6	Ghita
7	Colea
8	Nicu
9	Valea

# Clauza WHERE

idcarte	autor	titlu
2-2222-222-10	Petru GFGF	Laborator My
2-2222-222-13	Vieru	MAMA
2-2222-2222-22	Eminescu	Ghid Php
2-2222-2222-23	Eminescu	Ghid Php
2-2222-2222-6	Vica	Cei trei musc
2-2222-2222-8	Eminescu	Ghid Php
2-2222-2222-9	Eminescu	Ghid Php

## Exemplu

Listeaza toți angajații care al doilea caracter din nume 'o'.

```
SELECT nume, functie,
       data_ang
FROM angajati
WHERE nume LIKE '_o%'
```

Sa se creeze urmatoarele tabele:

### Tabela DEPARTAMENTE

```
Id_dept number(3) cheie primara (PK)
Den_dept varchar2(20)
Id_manager varchar2(3)
Locatie varchar2(100)
```

### Tabela ANGAJATI

```
Id_angajat number(3) cheie primara (PK)
Id_dept number(3) referinta (FK) la tabela DEPARTAMENTE
Nume varchar2(40)
Prenume varchar2(40)
Functie varchar2(25)
Salariu number(7)
Id_manager varchar2(3)
Data_ang date
Comision number(5)
```

# Clauza WHERE

**ESCAPE** - când sunt căutate chiar caracterele % sau \_. Acesta specifică caracterul care este "sărit".

Sintaxa:

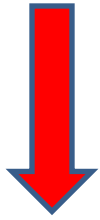
**SELECT** coloana

**FROM** tabela

**WHERE** coloana **LIKE** '%sa\\_%' **ESCAPE** '\';

**ESCAPE** identifică '\' drept caracterul care trebuie "sărit".

El precede caracterul '\_' astfel încât acesta va fi considerat drept literal.



# Clauza WHERE

## 4. Conditia IS NULL

Pentru a verifica valorile de tip **NULL** există condiția **IS NULL** sau negarea acesteia **IS NOT NULL**.

O valoare nulă este o valoare care este sau incorectă, sau necunoscută, sau inaplicabilă de aceea nu poate fi testată cu "=".

O valoare nulă nu este la fel cu "zero" care este un număr.

Dacă valoarea **NULL** este utilizată într-o comparație, atunci operatorul de comparație trebuie să fie **IS NULL** sau **IS NOT NULL** altfel rezultatul este întotdeauna FALSE.

1	Petru
2	Diuma
3	Vieru
4	Lena
5	Ioana
6	Ghita
7	Colea
8	Nicu
9	Valea

# Clauza WHERE

Exemplu:

Listează toți angajații care nu au comision.

```
SELECT *
FROM angajati
WHERE comision IS NULL
```

idcarte	autor	titlu
2-2222-222-10	Petru GFGF	Laborator Mys
2-2222-222-13	Vieru	MAMA
2-2222-2222-22	Eminescu	Ghid Php
2-2222-2222-23	Eminescu	Ghid Php
2-2222-2222-6	Vica	Cei trei musch
2-2222-2222-8	Eminescu	Ghid Php
2-2222-2222-9	Eminescu	Ghid Php

Sa se creeze urmatoarele tabele:

## Tabela DEPARTAMENTE

```
Id_dept number(3) cheie primara (PK)
Den_dept varchar2(20)
Id_manager varchar2(3)
Locatie varchar2(100)
```

## Tabela ANGAJATI

```
Id_angajat number(3) cheie primara (PK)
Id_dept number(3) referinta (FK) la tabela DEPARTAMENTE
Nume varchar2(40)
Prenume varchar2(40)
Functie varchar2(25)
Salariu number(7)
Id_manager varchar2(3)
Data_ang date
Comision number(5)
```

# Clauza WHERE

## 5.Negarea expresiilor

Operator	Semnificatie
<b>!=</b>	diferit de (pt anumite S.O.)
<b>^=</b>	diferit de (pt anumite S.O.)
<b>&lt;&gt;</b>	diferit de
<b>NOT BETWEEN</b>	nu se afla intre 2 valori date
<b>NOT IN</b>	nu se afla intr-o lista
<b>NOT LIKE</b>	diferit de sirul
<b>IS NOT NULL</b>	nu este o valoare nula

# Clauza WHERE

## D. Prioritatea de execuție

Ordinea de execuție	Operator
1.	Operatorii aritmetici
2.	Operatorii de concatenare
3.	Condițiile de comparare
4.	IS [NOT] NULL, LIKE, [NOT] IN
5.	[NOT] BETWEEN
6.	Operatorul logic NOT
7.	Operatorul logic AND
8.	Operatorul logic OR

1	Petru
2	Diuna
3	Vieru
4	Lena
5	Ioana
6	Ghita
7	Colea
8	Nicu
9	Valea

# Clauza WHERE

idcarte	autor	titlu
2-2222-222-10	Petru GFGF	Laborator M
2-2222-222-13	Vieru	MAMA
2-2222-2222-22	Eminescu	Ghid Php
2-2222-2222-23	Eminescu	Ghid Php
2-2222-2222-6	Vica	Cei trei mu
2-2222-2222-8	Eminescu	Ghid Php
2-2222-2222-9	Eminescu	Ghid Php

Exemplu:

```

SELECT *
FROM angajati
WHERE functie = 'DIRECTOR'
OR functie = 'CONTABIL'
AND salariu < 1500

```

Sa se creeze urmatoarele tabele:

## Tabela DEPARTAMENTE

Id\_dept number(3) cheie primara (PK)  
Den\_dept varchar2(20)  
Id\_manager varchar2(3)  
Locatie varchar2(100)

## Tabela ANGAJATI

Id\_angajat number(3) cheie primara (PK)  
Id\_dept number(3) referinta (FK) la tabela DEPARTAMENTE  
Nume varchar2(40)  
Prenume varchar2(40)  
Functie varchar2(25)  
Salariu number(7)  
Id\_manager varchar2(3)  
Data\_ang date  
Comision number(5)



1	Petru
2	Diuma
3	Vieru
4	Lena
5	Ioana
6	Ghita
7	Colea
8	Nicu
9	Valea

# Clauza WHERE

Dar ordinea se poate modifica dacă se folosesc paranteze.

```

SELECT *
FROM angajati
WHERE (functie = 'DIRECTOR'
OR functie = 'CONTABIL')
AND salariu < 1500

```

Sa se creeze urmatoarele tabele:

## Tabela DEPARTAMENTE

Id\_dept number(3) cheie primara (PK)  
Den\_dept varchar2(20)  
Id\_manager varchar2(3)  
Locatie varchar2(100)

## Tabela ANGAJATI

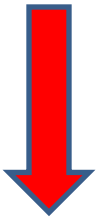
Id\_angajat number(3) cheie primara (PK)  
Id\_dept number(3) referinta (FK) la tabela DEPARTAMENTE  
Nume varchar2(40)  
Prenume varchar2(40)  
Functie varchar2(25)  
Salariu number(7)  
Id\_manager varchar2(3)  
Data\_ang date  
Comision number(5)

idcarte	autor	titlu
2-2222-222-10	Petru GFGF	Laborator My
2-2222-222-13	Vieru	MAMA
2-2222-2222-22	Eminescu	Ghid Php
2-2222-2222-23	Eminescu	Ghid Php
2-2222-2222-6	Vica	Cei trei musc
2-2222-2222-8	Eminescu	Ghid Php
2-2222-2222-9	Eminescu	Ghid Php

# Limbajul SQL

## Cereri SELECT pe o tabelă

1. **SELECT. Sintaxa. Efect. Rezultat**
2. **Lista SELECT**
3. **Clauza WHERE**
4. **Clauza ORDER BY**



# Clauza ORDER BY

În mod normal (fără clauza **ORDER BY**) rândurile sunt returnate într-o ordine convenită de SGBD însă ea fiind consistentă de la cerere la cerere.

Cu ajutorul clauzei **ORDER BY** rândurile vor fi afișate în ordinea solicitată (cu toate acestea nu se va modifica ordinea internă a rândurilor din baza de date).

**Clauza ORDER BY trebuie să fie ultima clauză din cerere (interogare).**

# Clauza ORDER BY

Sintaxa:

**SELECT** coloana

**FROM** tabela

[**WHERE** conditie]

[**ORDER BY** {coloana, expresie}]

[**ASC/DESC**];

unde:

**ORDER BY** - Specifică ordinea în care sunt ordonate rândurile

**ASC** - Ordonează rândurile crescător

**DESC** - Ordonează rândurile descrescător

1	Petru
2	Diuma
3	Vieru
4	Lena
5	Ioana
6	Ghita
7	Colea
8	Nicu
9	Valea

# Clauza ORDER BY

idcarte	autor	titlu
2-2222-222-10	Petru GFGF	Laborator My
2-2222-222-13	Vieru	MAMA
2-2222-2222-22	Eminescu	Ghid Php
2-2222-2222-23	Eminescu	Ghid Php
2-2222-2222-6	Vica	Cei trei musc
2-2222-2222-8	Eminescu	Ghid Php
2-2222-2222-9	Eminescu	Ghid Php

Exemplu:

Listați toți angajații ordonându-i după data de angajare (în ordine crescătoare).

```
SELECT *
FROM angajati
ORDER BY data_ang
```

Sa se creeze urmatoarele tabele:

## Tabela DEPARTAMENTE

Id\_dept number(3) cheie primara (PK)  
 Den\_dept varchar2(20)  
 Id\_manager varchar2(3)  
 Locatie varchar2(100)

## Tabela ANGAJATI

Id\_angajat number(3) cheie primara (PK)  
 Id\_dept number(3) referinta (FK) la tabela DEPARTAMENTE  
 Nume varchar2(40)  
 Prenume varchar2(40)  
 Functie varchar2(25)  
 Salariu number(7)  
 Id\_manager varchar2(3)  
 Data\_ang date  
 Comision number(5)

2	Diuma
3	Vieru
4	Lena
5	Ioana
6	Ghita
7	Colea
8	Nicu
9	Valea

# Clauza ORDER BY

Exemplu:

Listați toți angajații ordonându-i după data de angajare (în ordine descrescătoare).

```
SELECT *
FROM angajati
ORDER BY data_ang DESC
```

idcarte	autor	titlu
2-2222-222-10	Petru GFGF	Labor
2-2222-222-13	Vieru	MAMA
2-2222-2222-22	Eminescu	Ghid f
2-2222-2222-23	Eminescu	Ghid f
2-2222-2222-6	Vica	Cei tre
2-2222-2222-8	Eminescu	Ghid f
2-2222-2222-9	Eminescu	Ghid f

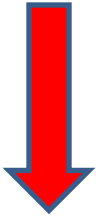
Sa se creeze urmatoarele tabele:

## Tabela DEPARTAMENTE

Id\_dept number(3) cheie primara (PK)  
 Den\_dept varchar2(20)  
 Id\_manager varchar2(3)  
 Locatie varchar2(100)

## Tabela ANGAJATI

Id\_angajat number(3) cheie primara (PK)  
 Id\_dept number(3) referinta (FK) la tabela DEPARTAMENTE  
 Nume varchar2(40)  
 Prenume varchar2(40)  
 Functie varchar2(25)  
 Salariu number(7)  
 Id\_manager varchar2(3)  
 Data\_ang date  
 Comision number(5)



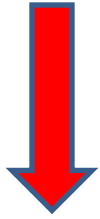
# Clauza ORDER BY

Mai putem ordona randurile returnate cu ajutorul alias-urilor.

Exemplu:

Listati toti angajatii ordonandu-i dupa salariul anual: sal\*12 salannual (in ordine crescatoare).

```
SELECT nume, functie, salariu*12 salariuanual  
FROM angajati  
ORDER BY salariuanual
```



# Clauza ORDER BY

1	Petru
2	Diuma
3	Vieru
4	Lena
5	Ioana
6	Ghita
7	Colea
8	Nicu
9	Valea

Rândurile mai pot fi ordonate și după mai multe coloane.

Numărul de coloane după care se pot ordona rândurile întoarse este numărul maxim de coloane existente în tabelă.

Exemplu

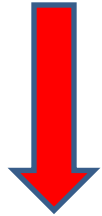
Listați toți angajații ordonându-i după data de angajare, salariu și comision (în ordine crescătoare).

idcarte	autor	titlu
2-2222-222-10	Petru	GFGF Laborator Mysql-Php vbvbyv
2-2222-222-13	Vieru	MAMA
2-2222-2222-22	Eminescu	Ghid Php
2-2222-2222-23	Eminescu	Ghid Php
2-2222-2222-6	Vica	Cei trei muschetari Php
2-2222-2222-8	Eminescu	Ghid Php
2-2222-2222-9	Eminescu	Ghid Php

```
SELECT *  
FROM angajati  
ORDER BY data_ang, salariu, comision
```



1	Petru
2	Diuma
3	Vieru
4	Lena
5	Ioana
6	Ghita
7	Colea
8	Nicu
9	Valea



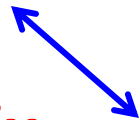
# Clauza ORDER BY

În clauza **ORDER BY** *se pot folosi coloane care nu sunt conținute în SELECT.*

Exemplu:

Listați toți angajații ordonându-i după salariu (în ordine crescătoare).

**SELECT** nume, functie  
**FROM** angajati  
**ORDER BY** *salariu*



idcarte	autor	titlu	pret	cantitatea	id_autor	
2-2222-222-10	Petru	GFGF	Laborator Mysql-Php vbvbnb	950.00	23	1
2-2222-222-13	Vieru	MAMA		2000.00	200	3
2-2222-2222-22	Eminescu	Ghid Php		1000.00	3	2
2-2222-2222-23	Eminescu	Ghid Php		1000.00	3	2
2-2222-2222-6	Vica	Cei trei muschetari Php		930.00	40	4
2-2222-2222-8	Eminescu	Ghid Php		1000.00	3	6
2-2222-2222-9	Eminescu	Ghid Php		1000.00	3	2

1	Petru
2	Diuma
3	Vieru
4	Lena
5	Ioana
6	Ghita
7	Colea
8	Nicu
9	Valea

## Exercitii

2-2222-222-13	Vieru	MAMA	2000.00
2-2222-2222-22	Eminescu	Ghid Php	1000.00
2-2222-2222-23	Eminescu	Ghid Php	1000.00
2-2222-2222-6	Vica	Cei trei muschetari Php	930.00
2-2222-2222-8	Eminescu	Ghid Php	1000.00
2-2222-2222-9	Eminescu	Ghid Php	1000.00

1. Listați toți angajații care au salariul între 1500 și 3000.
2. Listați numele angajaților ordonându-i în ordine alfabetică.
3. Listați toți angajații ai căror nume conțin NE sau LL în interior.
4. Listați toți angajații care nu au manager.

# Exercitii

5. Listați toți angajații care au fost angajați în anul 2010.
6. Afișați numele, salariul anual și comisionul pentru toți vânzătorii ai căror salariu lunar este mai mare decât comisionul lor.

leșirea va fi ordonată după salariu, cele mai mari primele.

Dacă doi sau mai mulți angajați au același salariu trebuie sortați după nume în ordinea celor mai mari salarii.

