

**Baza de date FIRMA**

**Conține următoarele tabele:**

**Angajati** (*ida number(3), nume varchar2(30), profesie varchar2(30), data\_nastere date, data\_angajare date, salariu number(5), iddep number(3)* )

*create table Angajati (ida number(3), nume varchar2(30), profesie varchar2(30), data\_nastere date, data\_angajare date, salariu number(5), iddep number(3) )*

**Departamente** ( *iddep number(3), denumire varchar2(30), etaj number(2), idsef number(3)* )

*create table Departamente ( iddep number(3), denumire varchar2(30), etaj number(2), idsef number(3) )*

**Următoarele zece probleme fac referire la baza de date Firma**

**Subiectul nr. 1**

Afișați numele, profesia și salariul angajaților descrescător după salariu.

SELECT nume, profesie, salariu FROM angajati ORDER BY salariu DESC

Afișați numele, profesia și departamentul în care lucrează fiecare angajat

SELECT nume, profesie, denumire FROM angajati ang, departamente dep  
WHERE ang.iddep=dep.iddep

**Subiectul nr. 2**

Afișați numele, profesia și data angajării pentru cel mai nou angajat.

SELECT nume, profesie, data\_angajare FROM angajati  
WHERE data\_angajare=(SELECT max(data\_angajare) FROM angajati)

Afișați numele, profesia și departamentul pentru fiecare șef de departament

SELECT nume, profesie, denumire FROM angajati, departamente  
WHERE ida=idsef

**Subiectul nr. 3**

Afișați numele, profesia și data angajării pentru angajații cu o vechime de peste 10 ani.

SELECT nume, profesie, data\_angajare FROM angajati  
WHERE ADD\_MONTHS(data\_angajare, 12\*10) < SYSDATE

Afișați numele, profesia și salariul pentru angajații din departamentul IT

SELECT nume, profesie, salariu FROM angajati ang, departamente dep  
WHERE ang.iddep=dep.iddep AND UPPER(dep.denumire) = 'IT'



#### **Subiectul nr. 4**

Afișați numele, profesia și data nașterii pentru angajații care au sub 30 de ani.

```
SELECT nume, profesie, data_nastere FROM angajati  
WHERE ADD_MONTHS (data_nastere, 30*12) > SYSDATE
```

Afișați toți subordonații lui Avram Mihai. Se vor afișa numele și profesia

```
SELECT nume, profesie FROM angajati ang, departamente dep  
WHERE ang.iddep=dep.iddep AND idsef=(SELECT ida FROM angajati WHERE UPPER(nume)  
='AVRAM MIHAI') AND UPPER(nume)<>'AVRAM MIHAI'
```

#### **Subiectul nr. 5**

Să se majoreze cu 10% salariul celui mai vechi angajat.

```
UPDATE angajati SET salariu=1.1*salariu  
WHERE data_angajare=(SELECT min(data_angajare) FROM angajati)
```

Să se afișeze pentru toți angajații care lucrează la etajul 2 numele, profesia și departamentul.

```
SELECT nume, profesie, denumire FROM angajati ang, departamente dep  
WHERE ang.iddep=dep.iddep AND etaj=2
```

#### **Subiectul nr. 6**

Să se elimine din tabela **angajați** angajatul Vasilescu Andrei

```
DELETE FROM angajati WHERE UPPER(nume)='VASILESCU ANDREI'
```

Afișați denumirea departamentului condus de Avram Mihai

```
SELECT denumire FROM departamente  
WHERE idsef=( SELECT ida FROM angajati WHERE UPPER(nume)='AVRAM MIHAI')
```

#### **Subiectul nr. 7**

Afișați numele și data nașterii angajaților care își sărbătoresc azi data de naștere

```
SELECT nume, data_nastere FROM angajati  
WHERE EXTRACT(MONTH FROM SYSDATE) = EXTRACT(MONTH FROM data_nastere)  
AND EXTRACT(DAY FROM SYSDATE) = EXTRACT(DAY FROM data_nastere)
```

Afișați descrescător după salarii numele, profesiile și salariile angajaților de la departamentul economic

```
SELECT nume, profesie, salariu FROM angajati ang, departamente dep  
WHERE ang.iddep=dep.iddep AND UPPER(dep.denumire)= 'ECONOMIC'
```

#### **Subiectul nr. 8**

Afișați numele colegilor de departament pentru angajatul Costin Andrei

```
SELECT nume FROM angajati WHERE iddep=(SELECT iddep FROM angajati  
WHERE UPPER(nume)='COSTIN ANDREI') AND UPPER(nume)<>'COSTIN ANDREI'
```

Afișați numele celui mai tânăr șef de departament

```
SELECT nume FROM angajati  
WHERE data_nastere=(SELECT MAX(data_nastere) FROM angajati WHERE ida IN (SELECT  
idsef FROM departamente))
```



### Subiectul nr. 9

Afișați numele șefilor de departamente

```
SELECT nume FROM angajati WHERE ida IN (SELECT idsef FROM departamente)
```

Afișați denumirea fiecărui departament și salariul maxim din acel departament

```
SELECT denumire, MAX(salariu) FROM angajati ang, departamente dep  
WHERE ang.iddep=dep.iddep GROUP BY denumire
```

### Subiectul nr. 10

Să se majoreze cu 10% salariile angajaților din departamentul IT

```
UPDATE angajati SET salariu=1.1*salariu WHERE iddep=(SELECT iddep FROM departamente  
WHERE UPPER(denumire)= 'IT')
```

Afișați numele șefilor de departamente și denumirile departamentelor pe care le conduc

```
SELECT nume, denumire FROM angajati ang, departamente dep WHERE ang.iddep=dep.iddep  
AND ida=idsef
```

**Baza de date MAGAZIN** conține următoarele tabele:

**Produse** (idp number(5), denumire varchar2(30), categorie varchar2(30), data\_fab date, stoc number(5), pret number(8,2) )

*create table Produse (idp number(5), denumire varchar2(30), categorie varchar2(30), data\_fab date, stoc number(5), pret number(8,2) )*

**Vanzari** (idv number(5), data\_vanzare date, idp number(5), numar\_bucati number(3) )

*create table Vanzari (idv number(5), data\_vanzare date, idp number(5), numar\_bucati number(3) )*

Următoarele zece probleme fac referire la baza de date magazin

### Subiectul nr. 11

Să se majoreze cu 10% produsele *lactate*

```
UPDATE pret SET pret=1.1*pret WHERE UPPER(categorie)= 'LACTATE'
```

Să se afișeze toate vânzările (denumire produs, data, număr de bucăți).

```
SELECT denumire, data_vanzare, numar_bucati FROM produse pd, vanzari vz WHERE  
pd.idp=vz.idp
```

### Subiectul nr. 12

Să se afișeze denumirile produselor vândute azi

```
SELECT DISTINCT(denumire) FROM produse pd, vanzari vz WHERE pd.idp=vz.idp AND  
TO_CHAR(data_vanzare)=TO_CHAR(SYSDATE)
```

Să se afișeze categoria cu cele mai multe produse pe stoc (ca număr de bucăți)

```
SELECT categorie, SUM(stoc) FROM produse GROUP BY categorie HAVING SUM(stoc)=  
(SELECT MAX (SUM(stoc) ) FROM produse GROUP BY categorie)
```



### **Subiectul nr. 13**

Să se afișeze denumirile produselor cu prețul cel mai mic

```
SELECT denumire, pret FROM produse  
WHERE pret=(SELECT min(pret) FROM produse)
```

Pentru fiecare produs vândut să se afișeze denumirea, totalul valorilor încasate precum și totalul de cantitate vândută.

```
SELECT denumire, SUM(pret*numar_bucati), SUM(numar_bucati) FROM produse pd, vanzari vz  
WHERE pd.idp=vz.idp GROUP BY denumire
```

### **Subiectul nr. 14**

Să se afișeze categoriile produselor, prețul minim și prețul maxim pentru fiecare categorie

```
SELECT categorie, MIN(pret), MAX(pret) FROM produse GROUP BY categorie
```

Să se afișeze în ordine alfabetică denumirea produselor care nu s-au vândut.

```
SELECT denumire FROM produse  
WHERE idp NOT IN (SELECT DISTINCT(idp) FROM vanzari)
```

### **Subiectul nr. 15**

Să se afișeze produsele cu stoc epuizat

```
SELECT denumire FROM produse WHERE stoc=0
```

Să se determine câte bucăți s-au vândut azi din produsul *Fanta*

```
SELECT SUM(numar_bucati) FROM produse pd, vanzari vz  
WHERE pd.idp=vz.idp AND UPPER (denumire) = 'FANTA'  
AND TO_CHAR(data_vanzare)=TO_CHAR(SYSDATE)
```

### **Subiectul nr. 16**

Să se afișeze denumirea și prețul produselor cu prețul cuprins între 5 și 10 lei.

```
SELECT denumire, pret FROM produse WHERE pret >=5 AND pret <= 10
```

Să se afișeze toate încasările realizate la produsele lactate

```
SELECT denumire, SUM(numar_bucati*pret) FROM produse pd, vanzari vz  
WHERE pd.idp=vz.idp AND UPPER(categorie)='LACTATE' GROUP BY denumire
```

### **Subiectul nr. 17**

Să se afișeze denumirile produselor lactate care expiră mâine știind că acestea expiră după 30 de zile de la data fabricației

```
SELECT denumire, data_fab FROM produse WHERE data_fab+30 <= SYSDATE+1
```

Să se afișeze produsele și numărul bucăților vândute descrescător după numărul bucăților vândute

```
SELECT denumire, SUM(numar_bucati) FROM produse pd, vanzari vz  
WHERE pd.idp=vz.idp GROUP BY denumire ORDER BY SUM(numar_bucati) DESC
```



### **Subiectul nr. 18**

Să se șteargă produsele cu stoc 0

```
DELETE FROM produse WHERE stoc=0
```

Să se afișeze denumirile produselor vândute luna aceasta

```
SELECT DISTINCT(denumire) FROM produse pd, vanzari vz  
WHERE pd.idp=vz.idp AND TO_CHAR(data_vanzare,'mm')=TO_CHAR(SYSDATE,'mm')
```

### **Subiectul nr. 19**

Să se afișeze categoria și valoare încasărilor pentru fiecare categorie de produse

```
SELECT categorie, SUM(numar_bucati*pret) FROM produse pd, vanzari vz WHERE pd.idp=vz.idp  
GROUP BY categorie
```

Să se micșoreze cu 10% prețul produselor cu preț maxim din categoria *dulciuri*

```
UPDATE produse SET pret=0.9*pret  
WHERE UPPER(categorie)= 'DULCIURI'
```

### **Subiectul nr. 20**

Să se afișeze produsele ordonate pe categorii, denumirile lor și prețul acestora ordonate descrescător după preț în cadrul fiecărei categorii

```
SELECT categorie, denumire, pret FROM produse ORDER BY categorie, pret DESC
```

Să se afișeze denumirea/denumirile produselor pentru care s-au realizat cele mai multe vânzări (ca număr de bucăți cumulate în toate vânzările)

```
SELECT denumire FROM produse  
WHERE idp IN  
(SELECT idp FROM vanzari GROUP BY idp HAVING SUM(numar_bucati)=  
(SELECT MAX(SUM(numar_bucati)) FROM vanzari GROUP BY idp) )
```

**Baza de date ȘCOALA** conține următoarele tabele:

**Elev** (**ide** number(3), **nume** varchar2(30), **data\_nastere** date, **medie** number(5,2), **cls** varchar2(5))

*create table Elev (ide number(3), nume varchar2(30), data\_nastere date, medie number(5,2), cls varchar2(5))*

**Clasa** (**cls** varchar2(5), **specializare** varchar2(30), **sala** number(2), **id\_dirig** number(3))

*create table Clasa (cls varchar2(5), specializare varchar2(30), sala number(2), id\_dirig number(3))*

**Profesor** (**idp** number(3), **nume** varchar2(30), **disciplina** varchar2(30), **data\_ang** date, **salariu** number(5))

*create table Profesor (idp number(3), nume varchar2(30), disciplina varchar2(30), data\_ang date, salariu number(5))*

Următoarele zece probleme fac referire la baza de date Școala



### **Subiectul nr. 21**

Afișați numele, disciplina predată și salariul profesorilor, descrescător după salariu.

```
SELECT nume, disciplina, salariu FROM profesor ORDER BY salariu DESC
```

Afișați numele și clasa elevilor de la profilul matematica-informatica

```
SELECT nume, e.cls FROM elev e, clasa c  
WHERE e.cls=c.cls AND specializare='matematica-informatica'
```

### **Subiectul nr. 22**

Afișați disciplinele predate în școală, în ordine alfabetică.

```
SELECT DISTINCT disciplina FROM profesor ORDER BY disciplina
```

Afișați numele și clasa elevilor care au media 10, precum și diriginții acestora.

```
SELECT e.numa AS nume_elev, e.cls, p.numa AS nume_diriginte FROM elev e, clasa c, profesor p  
WHERE e.cls=c.cls AND idp=id_dirig AND medie=10
```

### **Subiectul nr. 23**

Afișați salariul mediu al profesorilor de matematică.

```
SELECT AVG(salariu) FROM profesor WHERE disciplina='matematica'
```

Afișați elevii din clasele a 9-a și sălile lor de clasă.

```
SELECT e.numa, e.cls FROM elev e, clasa c WHERE e.cls=c.cls AND e.cls LIKE '9%'
```

### **Subiectul nr. 24**

Câți profesori de matematică și informatică sunt în școală?

```
SELECT COUNT(*) FROM profesor WHERE disciplina='matematica' OR disciplina='informatica'
```

Afișați elevii care au ca diriginte profesorul cu cea mai mică vechime.

```
SELECT e.numa FROM elev e, clasa c, profesor p  
WHERE e.cls=c.cls AND idp=id_dirig AND data_ang=(SELECT MAX(data_ang) FROM profesor)
```

### **Subiectul nr. 25**

Afișați numele, salariul și data angajării pentru profesorii cu vechime mai mare de 10 ani.

```
SELECT numa, salariu, data_ang FROM profesor  
WHERE ADD_MONTHS(data_ang, 12*10) < SYSDATE
```

Afișați numele elevilor care au ca diriginți profesori de matematică.

```
SELECT e.numa FROM elev e, clasa c, profesor p  
WHERE e.cls=c.cls AND idp=id_dirig AND disciplina='matematica'
```

### **Subiectul nr. 26**

Care sunt clasele care nu au specializarea FILOLOGIE?

```
SELECT cls, specializare FROM clasa  
WHERE LOWER(specializare)!='filologie'
```

Afișați mediile elevilor pe clase.

```
SELECT e.cls, AVG(medie) FROM elev e, clasa c, profesor p  
WHERE e.cls=c.cls AND idp=id_dirig GROUP BY e.cls
```

### **Subiectul nr. 27**

Să se șteargă elevii cu media mai mică decât 5.

```
DELETE FROM elev WHERE medie<5
```

Să se afișeze diriginții ai căror elevi au medii între 8 și 9.

```
SELECT e.num AS elev, e.medie, p.num AS profesor FROM elev e, clasa c, profesor p WHERE  
e.cls=c.cls AND idp=id_dirig AND medie BETWEEN 8 AND 9
```

### **Subiectul nr. 28**

Să se afișeze elevii care își serbează ziua de naștere în luna curentă.

```
SELECT nume FROM elev  
WHERE EXTRACT(MONTH FROM data_nastere)=EXTRACT(MONTH FROM SYSDATE)
```

Să se afișeze profesorii care au salariul mai mare decât salariul dirigintelui de la clasa 9B.

```
SELECT nume, salariu FROM profesor  
WHERE salariu>(SELECT salariu FROM profesor, clasa WHERE id_dirig=idp AND cls='9B')
```

### **Subiectul nr. 29**

Câți elevi învață la etajul 1 (sălile de la 10 la 19)?

```
SELECT COUNT(*) FROM elev e, clasa c  
WHERE e.cls=c.cls AND sala BETWEEN 10 AND 19
```

Afișați salariul celui mai vechi angajat.

```
SELECT nume, salariu FROM profesor WHERE data_ang=(SELECT MIN(data_ang) FROM  
profesor)
```

### **Subiectul nr. 30**

Să se afișeze numele elevilor și clasele din care fac parte, împreună cu numerele lor de ordine din tabela ELEV.

```
SELECT ROWNUM AS nr_crt, nume, cls FROM elev
```

Afișați elevii care au aceeași specializare cu Popa Daniela.

```
SELECT nume, e.cls FROM elev e, clasa c  
WHERE c.cls=e.cls AND UPPER(nume)<>'POPA DANIELA' AND specializare=  
(SELECT specializare FROM clasa c, elev e  
WHERE c.cls=e.cls AND UPPER(nume)='POPA DANIELA')
```



## **Baza de date CLINICA**

**Conține următoarele tabele:**

**Medic** (**idm** *number*(3), **nume** *varchar2*(30), **data\_ang** *date*, **salariu** *number*(8,2), **specializare** *varchar2*(30))

*create table* **Medic** (**idm** *number*(3), **nume** *varchar2*(30), **data\_ang** *date*, **salariu** *number*(8,2), **specializare** *varchar2*(30))

**Pacient** (**idp** *number*(4), **nume** *varchar2*(30), **data\_n** *date*, **adresa** *varchar2*(20), **statut** *varchar2*(20), **tel** *varchar2*(10))

*create table* **Pacient** (**idp** *number*(4), **nume** *varchar2*(30), **data\_n** *date*, **adresa** *varchar2*(20), **statut** *varchar2*(20), **tel** *varchar2*(10))

**Consultatie** (**idp** *number*(4), **idm** *number*(3), **data\_cons** *date*, **diagnostic** *varchar2*(30))

*create table* **Consultatie** (**idp** *number*(4), **idm** *number*(3), **data\_cons** *date*, **diagnostic** *varchar2*(30))

**Următoarele zece probleme fac referire la baza de date CLINICA**

### **Subiectul nr. 31**

Afișați numele și salariile tuturor medicilor, în ordine alfabetică

SELECT nume, salariu FROM medic ORDER BY nume

Să se afișeze pentru fiecare consultație atât data consultației, cât și datele pacienților (nume, adresă, statut) și ale medicilor (nume, specializare). Numele se vor afișa cu majuscule.

SELECT data\_cons, specializare, UPPER(m.nume), diagnostic, UPPER(p.nume), adresa, statut  
FROM medic m, pacient p, consultatie c WHERE m.idm=c.idm AND p.idp=c.idp

### **Subiectul nr. 32**

Să se afișeze pacienții din Galați care sunt elevi sau studenți

SELECT \* FROM pacient WHERE (statut='elev' OR statut='student') AND adresa LIKE 'Galati'

Să se afișeze consultațiile pacientei Barbu Ionela (data consultației, numele medicilor, specializările, diagnosticul)

SELECT data\_cons, specializare, UPPER(m.nume), diagnostic FROM medic m, pacient p, consultatie c WHERE m.idm=c.idm AND p.idp=c.idp AND p.nume='Barbu Ionela'

### **Subiectul nr. 33**

Câți pacienți care nu sunt pensionari sunt în tabela PACIENT?

SELECT COUNT(\*) FROM pacient WHERE statut NOT LIKE 'pensionar'

Să se afișeze toate consultatiile medicilor cu salariu mai mare decât salariul mediu al tuturor medicilor.

SELECT nume, specializare, data\_cons, diagnostic FROM medic m, consultatie c WHERE c.idm=m.idm AND salariu>(SELECT AVG(salariu) FROM medic)





### **Subiectul nr. 34**

Să se afișeze toate datele medicului / medicilor cu salariu maxim

```
SELECT * FROM medic  
WHERE salariu=(SELECT MAX(salariu) FROM medic)
```

Să se afișeze, pentru fiecare diagnostic, numărul de pacienți diagnosticați.

```
SELECT diagnostic, COUNT(*) AS nr_pacienti FROM consultatie GROUP BY diagnostic
```

### **Subiectul nr. 35**

Câte consultații au fost cu fractură sau entorsă?

```
SELECT COUNT(*) FROM consultatie  
WHERE diagnostic='fractura' OR diagnostic='entorsa'
```

Care este suma salariilor medicilor care nu au acordat consultații în luna octombrie?

```
SELECT SUM(salariu) FROM medic WHERE idm NOT IN (SELECT idm FROM consultatie  
WHERE EXTRACT(MONTH FROM data_cons)=10)
```

### **Subiectul nr. 36**

Să se afișeze medicii cu o vechime mai mare de 25 de ani.

```
SELECT nume, data_ang FROM medic  
WHERE ADD_MONTHS(data_ang, 25*12) <= SYSDATE
```

Care este salariul maxim al medicilor care au consultat elevi sau copii în luna curentă?

```
SELECT MAX(salariu) FROM medic m, consultatie c, pacient p  
WHERE m.idm=c.idm AND c.idp=p.idp AND (statut='copil' OR statut='elev') AND EXTRACT  
(MONTH FROM data_cons)=EXTRACT (MONTH FROM SYSDATE)
```

### **Subiectul nr. 37**

Să se afișeze cel mai mic și cel mai mare salariu al medicilor

```
SELECT MAX(salariu), MIN(salariu) FROM medic
```

Câți pacienți diferiți au avut consultații în această lună? (se raportează la data curentă)

```
SELECT COUNT(DISTINCT p.idp) FROM pacient p, consultatie c  
WHERE p.idp = c.idp AND EXTRACT (MONTH FROM data_cons) = EXTRACT (MONTH  
FROM SYSDATE)
```

### **Subiectul nr. 38**

Să se afișeze numărul de consultații din fiecare lună.

```
SELECT COUNT(*), TO_CHAR(data_cons,'MONTH') FROM consultatie GROUP BY  
TO_CHAR(data_cons,'MONTH')
```

Care sunt pacienții care au fost consultați de medicul cel mai recent angajat?

```
SELECT p.nume FROM medic m, consultatie c, pacient p  
WHERE m.idm=c.idm AND c.idp=p.idp AND data_ang=(SELECT MAX(data_ang) FROM  
medic)
```



### **Subiectul nr. 39**

Să se afișeze datele pacienților care sunt pensionari

```
SELECT * FROM pacient WHERE statut='pensionar'
```

Să se afișeze alfabetic numele pacienților care au avut consultații și diagnosticele primite.

```
SELECT p.nume, diagnostic FROM medic m, consultatie c, pacient p  
WHERE m.idm=c.idm AND c.idp=p.idp ORDER BY diagnostic
```

### **Subiectul nr. 40**

Să se afișeze suma salariilor tuturor medicilor.

```
SELECT SUM(salariu) FROM medic
```

Să se afișeze toți pacienții care au avut aceleași diagnostice ca pacientul Costin Ion.

```
SELECT nume, diagnostic FROM consultatie c, pacient p  
WHERE c.idp=p.idp AND diagnostic IN  
(SELECT diagnostic FROM consultatie c, pacient p
```

