Proiect final SBD

Mihai Andrei-Alexandru, Grupa 405

CUPRINS

[Cerinta 1 2](#_Toc188904674)

[Cerinta 2 2](#_Toc188904675)

[Cerinta 3 3](#_Toc188904676)

[Cerinta 4 3](#_Toc188904677)

[Cerinta 5 3](#_Toc188904678)

[Cerinta 6 4](#_Toc188904679)

[Cerinta 8 15](#_Toc188904680)

[Cerinta 9 15](#_Toc188904681)

[Cerinta 10 15](#_Toc188904682)

CERINȚE

Cerinta 1

*Prezentarea concisă a bazei de date (utilizarea ei).*

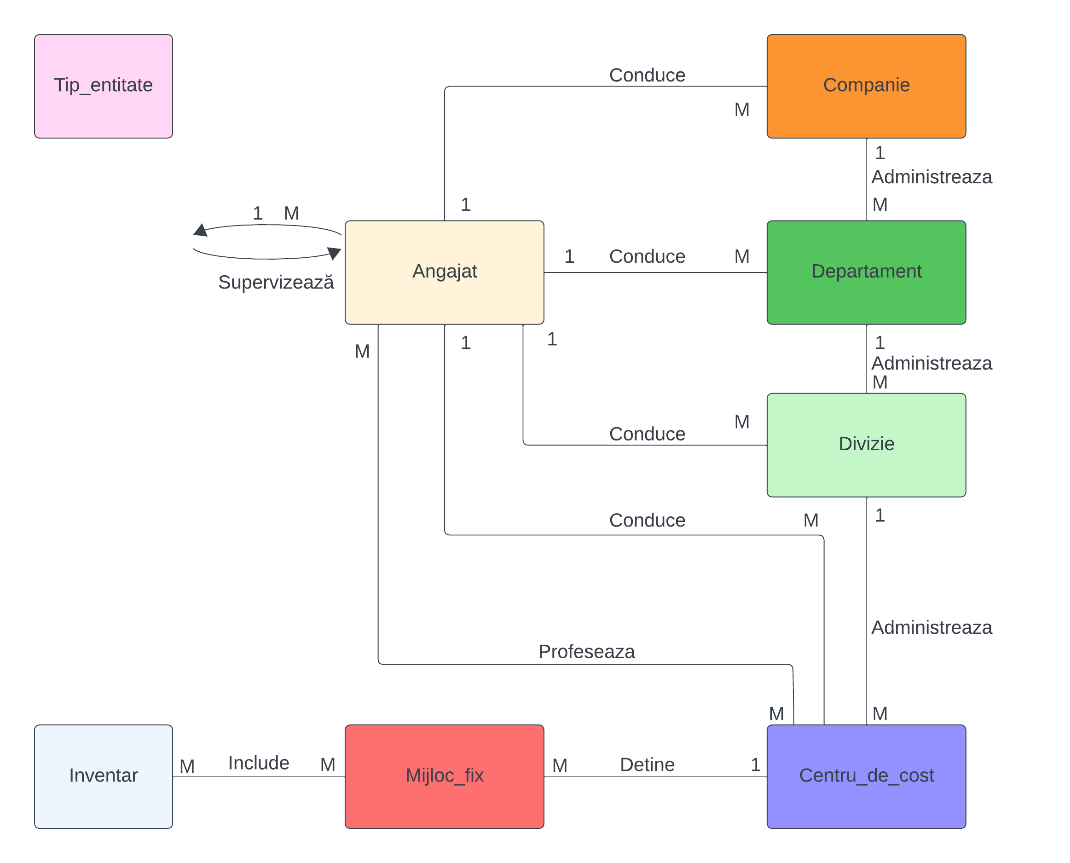
Această bază de date este destinată gestionării unei organizații complexe, având în vedere angajații, structura organizațională, resursele financiare și materiale, precum și inventarierea acestora.

**Scopul bazei de date:**

1. **Gestionarea angajaților** - Evidență completă a angajaților, ierarhiilor, și asocierea lor cu diferite structuri și resurse.
2. **Structurarea organizațională** - Organizarea companiei pe mai multe niveluri ierarhice: companie, departament, divizie, centre de cost.
3. **Managementul resurselor** - Gestionarea mijloacelor fixe și a inventarului cu trasabilitate clară.
4. **Urmărirea mesajelor și notificărilor** - Gestionarea mesajelor pentru evenimente operaționale sau erori.

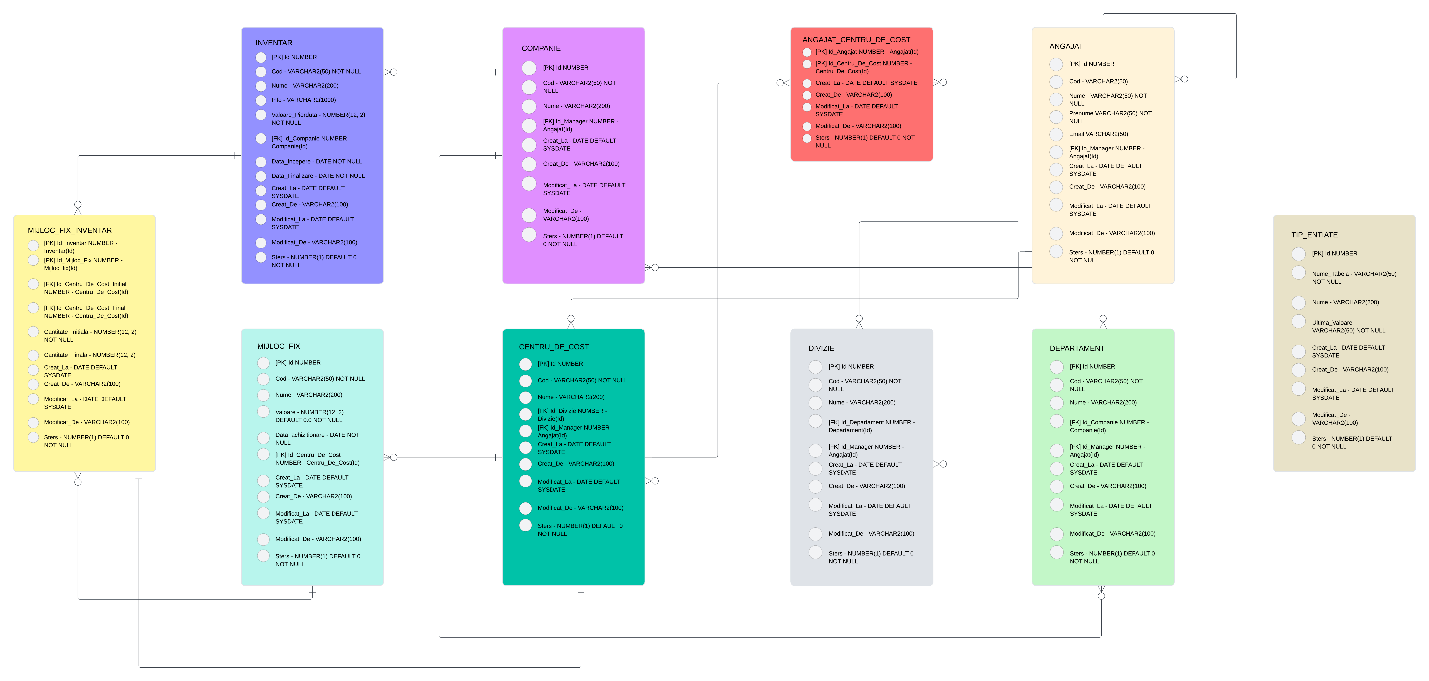
Cerinta 2

*Realizarea diagramei entitate-relație (ERD): entitățile, relațiile și atributele trebuie definite în limba română.*



Cerinta 3

*Realizarea diagramei conceptuale pornind de la diagrama entitate-relație.*



Cerinta 4

Transformarea sistemului conceptual într-un design logic, subliniind relaţiile dintre tabele, cheile primare şi străine (externe).

**Angajat**: are relație recursivă:

Fiecare angajat poate avea un manager (care îl supervizează).

**FK**: id\_manager → Angajat(id).

**Companie**: este condusă de un angajat:

**FK**: id\_manager → Angajat(id).

**Departament**: aparține unei companii și este condus de un manager:

**FK**: id\_companie → Companie(id).

**FK**: id\_manager → Angajat(id).

**Divizie**: aparține unui departament și este condusă de un manager:

**FK**: id\_departament → Departament(id).

**FK**: id\_manager → Angajat(id).

**Centru\_de\_Cost**: aparține unei divizii și este condus de un manager:

**FK**: id\_divizie → Divizie(id).

**FK**: id\_manager → Angajat(id).

**Angajat\_Centru\_de\_Cost**: leagă angajații de centrele de cost (relație M:M):

**PK**: id\_angajat, id\_centru\_de\_cost.

**FK**: id\_angajat → Angajat(id).

**FK**: id\_centru\_de\_cost → Centru\_de\_Cost(id).

**Inventar**: este desfășurat de către o companie:

**FK**: id\_companie → Companie(id).

**Mijloc\_fix**: este asociat unui centru de cost:

**FK**: id\_centru\_de\_cost → Centru\_de\_Cost(id).

**Mijloc\_fix\_inventar**: leagă mijloacele fixe de inventare (relație M:M):

**PK**: id\_inventar, id\_mijloc\_fix.

**FK**: id\_inventar → Inventar(id).

**FK**: id\_mijloc\_fix → Mijloc\_fix(id).

**FK**: id\_centru\_de\_cost\_initial → Centru\_de\_Cost(id).

**FK**: id\_centru\_de\_cost\_final → Centru\_de\_Cost(id) (opțional).

**Tip\_entitate**: stochează informații despre tabele:

Nu are relații directe cu alte entități.

Cerinta 5

Transformarea design-ului logic într-un design fizic astfel încât sistemul rezultat la punctul 4 să fie în FN3

• Dați un exemplu de atribut repetitiv (multivaloare) al unei entități din diagramă.

Posibil exemplu: în tabela **Angajat**, ar putea exista un atribut precum numere\_telefon, care permite stocarea mai multor numere de telefon pentru un singur angajat. Astfel, se creează o nouă tabelă pentru a stoca numerele de telefon, legată de tabela **Angajat**:

|  |
| --- |
| *CREATE TABLE* Angajat\_Telefon (  id\_angajat *NUMBER NOT NULL*,  telefon *VARCHAR2*(20) *NOT NULL*,  *CONSTRAINT* fk\_angajat\_telefon *FOREIGN KEY* (id\_angajat) *REFERENCES* Angajat(*id*),  *PRIMARY KEY* (id\_angajat, telefon) ); |

• Dați un exemplu de tabel relațional din diagramă care este în FN1, dar nu în FN2.

Să se aducă tabelul în FN2.

• Dați un exemplu de tabel relațional din diagramă care este în FN2, dar nu în FN3.

Să se aducă tabelul în FN3.

Cerinta 6

Implementarea tabelelor în Oracle, folosind chei primare, constrângeri de referinţa

domeniu. Adăugarea de informații coerente (minim 5 înregistrări pentru fiecare entitat

independentă; minim 10 înregistrări pentru fiecare tabelă asociativă.

|  |
| --- |
| *-- Tabela: Angajat CREATE TABLE* Angajat (  *id NUMBER GENERATED BY DEFAULT ON NULL AS IDENTITY PRIMARY KEY*,  cod *VARCHAR2*(50),  nume *VARCHAR2*(50) *NOT NULL*,  prenume *VARCHAR2*(50) *NOT NULL*,  email *VARCHAR2*(50),  id\_manager *NUMBER*,  creat\_la *DATE DEFAULT SYSDATE*,  creat\_de *VARCHAR2*(100),  modificat\_la *DATE DEFAULT SYSDATE*,  modificat\_de *VARCHAR2*(100),  sters *NUMBER*(1) *DEFAULT* 0 *NOT NULL*,  *CONSTRAINT* fk\_angajat\_manager *FOREIGN KEY* (id\_manager) *REFERENCES* Angajat(*id*) );  *-- Tabela: Companie CREATE TABLE* Companie (  *id NUMBER GENERATED BY DEFAULT ON NULL AS IDENTITY PRIMARY KEY*,  cod *VARCHAR2*(50) *NOT NULL*,  nume *VARCHAR2*(200),  id\_manager *NUMBER NOT NULL*,  creat\_la *DATE DEFAULT SYSDATE*,  creat\_de *VARCHAR2*(100),  modificat\_la *DATE DEFAULT SYSDATE*,  modificat\_de *VARCHAR2*(100),  sters *NUMBER*(1) *DEFAULT* 0 *NOT NULL*,  *CONSTRAINT* fk\_companie\_manager *FOREIGN KEY* (id\_manager) *REFERENCES* Angajat(*id*) );  *-- Tabela: Departament CREATE TABLE* Departament (  *id NUMBER GENERATED BY DEFAULT ON NULL AS IDENTITY PRIMARY KEY*,  cod *VARCHAR2*(50) *NOT NULL*,  nume *VARCHAR2*(200),  id\_companie *NUMBER NOT NULL*,  id\_manager *NUMBER NOT NULL*,  creat\_la *DATE DEFAULT SYSDATE*,  creat\_de *VARCHAR2*(100),  modificat\_la *DATE DEFAULT SYSDATE*,  modificat\_de *VARCHAR2*(100),  sters *NUMBER*(1) *DEFAULT* 0 *NOT NULL*,  *CONSTRAINT* fk\_departament\_companie *FOREIGN KEY* (id\_companie) *REFERENCES* Companie(*id*),  *CONSTRAINT* fk\_departament\_manager *FOREIGN KEY* (id\_manager) *REFERENCES* Angajat(*id*) );  *-- Tabela: Divizie CREATE TABLE* Divizie (  *id NUMBER GENERATED BY DEFAULT ON NULL AS IDENTITY PRIMARY KEY*,  cod *VARCHAR2*(50) *NOT NULL*,  nume *VARCHAR2*(200),  id\_departament *NUMBER NOT NULL*,  id\_manager *NUMBER NOT NULL*,  creat\_la *DATE DEFAULT SYSDATE*,  creat\_de *VARCHAR2*(100),  modificat\_la *DATE DEFAULT SYSDATE*,  modificat\_de *VARCHAR2*(100),  sters *NUMBER*(1) *DEFAULT* 0 *NOT NULL*,  *CONSTRAINT* fk\_divizie\_departament *FOREIGN KEY* (id\_departament) *REFERENCES* Departament(*id*),  *CONSTRAINT* fk\_divizie\_manager *FOREIGN KEY* (id\_manager) *REFERENCES* Angajat(*id*) );  *-- Tabela: Centru\_de\_Cost CREATE TABLE* Centru\_de\_Cost (  *id NUMBER GENERATED BY DEFAULT ON NULL AS IDENTITY PRIMARY KEY*,  cod *VARCHAR2*(50) *NOT NULL*,  nume *VARCHAR2*(200),  id\_divizie *NUMBER NOT NULL*,  id\_manager *NUMBER NOT NULL*,  creat\_la *DATE DEFAULT SYSDATE*,  creat\_de *VARCHAR2*(100),  modificat\_la *DATE DEFAULT SYSDATE*,  modificat\_de *VARCHAR2*(100),  sters *NUMBER*(1) *DEFAULT* 0 *NOT NULL*,  *CONSTRAINT* fk\_centru\_cost\_divizie *FOREIGN KEY* (id\_divizie) *REFERENCES* Divizie(*id*),  *CONSTRAINT* fk\_centru\_cost\_manager *FOREIGN KEY* (id\_manager) *REFERENCES* Angajat(*id*) );  *-- Tabela: Angajat\_Centru\_De\_Cost CREATE TABLE* Angajat\_Centru\_De\_Cost (  id\_angajat *NUMBER NOT NULL*,  id\_centru\_de\_cost *NUMBER NOT NULL*,  creat\_la *DATE DEFAULT SYSDATE*,  creat\_de *VARCHAR2*(100),  modificat\_la *DATE DEFAULT SYSDATE*,  modificat\_de *VARCHAR2*(100),  sters *NUMBER*(1) *DEFAULT* 0 *NOT NULL*,  *PRIMARY KEY* (id\_angajat, id\_centru\_de\_cost),  *CONSTRAINT* fk\_angajat\_cdc\_angajat *FOREIGN KEY* (id\_angajat) *REFERENCES* Angajat(*id*),  *CONSTRAINT* fk\_angajat\_cdc\_centru *FOREIGN KEY* (id\_centru\_de\_cost) *REFERENCES* Centru\_de\_Cost(*id*) );  *-- Tabela: Inventar CREATE TABLE* Inventar (  *id NUMBER GENERATED BY DEFAULT ON NULL AS IDENTITY PRIMARY KEY*,  cod *VARCHAR2*(50) *NOT NULL*,  nume *VARCHAR2*(200),  info *VARCHAR2*(1000),  valoare\_pierduta *NUMBER*(12,2) *DEFAULT* 0.0,  id\_companie *NUMBER NOT NULL*,  data\_incepere *DATE NOT NULL*,  data\_finalizare *DATE NOT NULL*,  creat\_la *DATE DEFAULT SYSDATE*,  creat\_de *VARCHAR2*(100),  modificat\_la *DATE DEFAULT SYSDATE*,  modificat\_de *VARCHAR2*(100),  sters *NUMBER*(1) *DEFAULT* 0 *NOT NULL*,  *CONSTRAINT* fk\_inventar\_companie *FOREIGN KEY* (id\_companie) *REFERENCES* Companie(*id*) );  *-- Tabela: Mijloc\_fix CREATE TABLE* Mijloc\_fix (  *id NUMBER GENERATED BY DEFAULT ON NULL AS IDENTITY PRIMARY KEY*,  cod *VARCHAR2*(50) *NOT NULL*,  nume *VARCHAR2*(200),  valoare *NUMBER*(12, 2) *DEFAULT* 0.0 *NOT NULL*,  data\_achizitionare *DATE NOT NULL*,  id\_centru\_de\_cost *NUMBER NOT NULL*,  creat\_la *DATE DEFAULT SYSDATE*,  creat\_de *VARCHAR2*(100),  modificat\_la *DATE DEFAULT SYSDATE*,  modificat\_de *VARCHAR2*(100),  sters *NUMBER*(1) *DEFAULT* 0 *NOT NULL*,  *CONSTRAINT* fk\_mijloc\_fix\_centru *FOREIGN KEY* (id\_centru\_de\_cost) *REFERENCES* Centru\_de\_Cost(*id*) );  *-- Tabela: Mijloc\_fix\_inventar CREATE TABLE* Mijloc\_fix\_inventar (  id\_inventar *NUMBER NOT NULL*,  id\_mijloc\_fix *NUMBER NOT NULL*,  id\_centru\_de\_cost\_initial *NUMBER NOT NULL*,  id\_centru\_de\_cost\_final *NUMBER*,  cantitate\_initiala *NUMBER*(12, 2) *NOT NULL*,  cantitate\_finala *NUMBER*(12, 2),  creat\_la *DATE DEFAULT SYSDATE*,  creat\_de *VARCHAR2*(100),  modificat\_la *DATE DEFAULT SYSDATE*,  modificat\_de *VARCHAR2*(100),  sters *NUMBER*(1) *DEFAULT* 0 *NOT NULL*,  *PRIMARY KEY* (id\_inventar, id\_mijloc\_fix),  *CONSTRAINT* fk\_mfi\_inventar *FOREIGN KEY* (id\_inventar) *REFERENCES* Inventar(*id*),  *CONSTRAINT* fk\_mfi\_mijloc\_fix *FOREIGN KEY* (id\_mijloc\_fix) *REFERENCES* Mijloc\_fix(*id*),  *CONSTRAINT* fk\_mfi\_centru\_initial *FOREIGN KEY* (id\_centru\_de\_cost\_initial) *REFERENCES* Centru\_de\_Cost(*id*),  *CONSTRAINT* fk\_mfi\_centru\_final *FOREIGN KEY* (id\_centru\_de\_cost\_final) *REFERENCES* Centru\_de\_Cost(*id*) );  *-- Tabela: Tip\_entitate CREATE TABLE* Tip\_entitate (  *id NUMBER GENERATED BY DEFAULT ON NULL AS IDENTITY PRIMARY KEY*,  nume\_tabela *VARCHAR2*(50) *NOT NULL*,  ultima\_valoare *VARCHAR2*(50) *NOT NULL*,  creat\_la *DATE DEFAULT SYSDATE*,  creat\_de *VARCHAR2*(100),  modificat\_la *DATE DEFAULT SYSDATE*,  modificat\_de *VARCHAR2*(100),  sters *NUMBER*(1) *DEFAULT* 0 *NOT NULL* );  Cerinta 7  Scrierea a 15 interogări, cât mai complexe, care să ilustreze toate aspectele învăţ  lista (formulați în limbaj natural problemele ce urmează a fi rezolvate). |

|  |
| --- |
| *--1. Top 5 departamente cu cele mai multe mijloace fixe șterse.  select* d.COD, d.NUME, count(\*) *as* Numar\_mijloace\_fixe\_sterse *from* DEPARTAMENT d *inner join* DIVIZIE div *on* div.ID\_DEPARTAMENT = d.ID *and* div.STERS = 0 *inner join* CENTRU\_DE\_COST cc *on* cc.ID\_DIVIZIE = div.ID *and* cc.STERS = 0 *inner join* MIJLOC\_FIX mf *on* mf.ID\_CENTRU\_DE\_COST = cc.ID *and* mf.STERS = 1 *where* d.STERS = 0 *group by* d.COD, d.NUME *order by* count(\*) *desc FETCH FIRST* 5 *ROWS ONLY*; |
| *--2. Lista centrelor de cost care au fost modificate de la prima introducere in sistem, împreună cu numărul de mijloace fixe atribuite.  select* cc.cod, cc.nume, cc.MODIFICAT\_LA, x.CountMF *as* NumarMF *from* centru\_de\_cost cc *inner join* (  *select* ID\_CENTRU\_DE\_COST, count(\*) *as* CountMF  *from* MIJLOC\_FIX  *where* STERS = 0  *group by* ID\_CENTRU\_DE\_COST ) X *on* X.ID\_CENTRU\_DE\_COST = cc.ID *where* (cc.CREAT\_LA *is not null and* cc.MODIFICAT\_LA *is not null*)  *and* cc.MODIFICAT\_LA > cc.CREAT\_LA |
| *--3. Diviziile grupate in functie de media valorii mijloacelor fixe fata de divizia cu maximul pe media valorii mijloacelor fixe  SELECT* div.COD,  div.NUME,  *CASE  WHEN* avg(mf.VALOARE) =  (  *SELECT MAX*(valoare\_medie)  *FROM* (  *SELECT* div.id *AS* id\_divizie, AVG(mf.VALOARE) *AS* valoare\_medie  *FROM* DIVIZIE div  *FULL JOIN* CENTRU\_DE\_COST cc *ON* cc.ID\_DIVIZIE = div.ID  *FULL JOIN* MIJLOC\_FIX mf *ON* mf.ID\_CENTRU\_DE\_COST = cc.ID  *GROUP BY* div.id  ) vm\_pe\_div) *THEN* TO\_CHAR(avg(mf.VALOARE)) || ' ='  *ELSE* TO\_CHAR(avg(mf.VALOARE)) || ' <'  *END AS* comparatie *FROM* DIVIZIE div *FULL JOIN* CENTRU\_DE\_COST cc *ON* cc.ID\_DIVIZIE = div.ID *FULL JOIN* MIJLOC\_FIX mf *ON* mf.ID\_CENTRU\_DE\_COST = cc.ID *WHERE* div.STERS = 0 *GROUP BY* div.COD, div.NUME; |
| *--4. Top 3 manageri cu cele mai multe mijloace fixe înregistrate dupa 2017.  select  a*.EMAIL *as* email\_angajat,  *COALESCE*(COMP.numar\_mf, DEP.numar\_mf, DIV.numar\_mf, CC.numar\_mf, 0) *as* numar\_mf *from* ANGAJAT *a left join* (  *select a*.id *as* id\_angajat, count(mf.ID) *as* numar\_mf  *from* Angajat *a  inner join* Companie *c on c*.ID\_MANAGER = *a*.ID *and a*.STERS = 0  *full join* DEPARTAMENT d *on* d.ID\_COMPANIE = *c*.ID  *full join* DIVIZIE div *on* div.ID\_DEPARTAMENT = d.ID *and* div.STERS = 0  *full join* CENTRU\_DE\_COST cc *on* cc.ID\_DIVIZIE = div.ID *and* cc.STERS = 0  *full join* MIJLOC\_FIX mf *on* mf.ID\_CENTRU\_DE\_COST = cc.ID *and* mf.STERS = 0  *where* mf.DATA\_ACHIZITIONARE > TO\_DATE('01-JAN-2017', 'DD-MON-YYYY')  *group by a*.id  *having a*.id *is not null* ) comp *on* comp.id\_angajat = *a*.ID *left join* (  *select a*.id *as* id\_angajat, count(mf.ID) *as* numar\_mf  *from* Angajat *a  inner join* DEPARTAMENT d *on* d.ID\_MANAGER = *a*.ID  *full join* DIVIZIE div *on* div.ID\_DEPARTAMENT = d.ID *and* div.STERS = 0  *full join* CENTRU\_DE\_COST cc *on* cc.ID\_DIVIZIE = div.ID *and* cc.STERS = 0  *full join* MIJLOC\_FIX mf *on* mf.ID\_CENTRU\_DE\_COST = cc.ID *and* mf.STERS = 0  *where* mf.DATA\_ACHIZITIONARE > TO\_DATE('01-JAN-2017', 'DD-MON-YYYY')  *group by a*.id  *having a*.id *is not null* ) dep *on* dep.id\_angajat = *a*.ID *left join* (  *select a*.id *as* id\_angajat, count(mf.ID) *as* numar\_mf  *from* Angajat *a  inner join* DIVIZIE div *on* div.ID\_MANAGER = *a*.ID *and* div.STERS = 0  *full join* CENTRU\_DE\_COST cc *on* cc.ID\_DIVIZIE = div.ID *and* cc.STERS = 0  *full join* MIJLOC\_FIX mf *on* mf.ID\_CENTRU\_DE\_COST = cc.ID *and* mf.STERS = 0  *where* mf.DATA\_ACHIZITIONARE > TO\_DATE('01-JAN-2017', 'DD-MON-YYYY')  *group by a*.id  *having a*.id *is not null* ) div *on* div.id\_angajat = *a*.ID *left join* (  *select a*.id *as* id\_angajat, count(mf.ID) *as* numar\_mf  *from* Angajat *a  inner join* CENTRU\_DE\_COST cc *on* cc.ID\_MANAGER = *a*.ID *and* cc.STERS = 0  *full join* MIJLOC\_FIX mf *on* mf.ID\_CENTRU\_DE\_COST = cc.ID *and* mf.STERS = 0  *where* mf.DATA\_ACHIZITIONARE > TO\_DATE('01-JAN-2017', 'DD-MON-YYYY')  *group by a*.id  *having a*.id *is not null* ) cc *on* cc.id\_angajat = *a*.ID *order by COALESCE*(COMP.numar\_mf, DEP.numar\_mf, DIV.numar\_mf, CC.numar\_mf, 0) *desc fetch next* 3 *rows only*; |
| *--5. Situatia scriptic vs faptic in cazul tuturor inventarelor din sistem.   select* NUME, DATA\_INCEPERE, DATA\_FINALIZARE,  scriptic.numar\_mf *as* SCRIPTIC,  faptic.numar\_mf *as* FAPTIC *from* INVENTAR i *left join* (  *select* ID\_INVENTAR, count(ID\_MIJLOC\_FIX) *as* numar\_mf  *from* MIJLOC\_FIX\_INVENTAR  *where* ID\_CENTRU\_DE\_COST\_INITIAL *is not null  group by* ID\_INVENTAR ) scriptic *on* scriptic.ID\_INVENTAR = i.ID *left join* (  *select* ID\_INVENTAR, count(ID\_MIJLOC\_FIX) *as* numar\_mf  *from* MIJLOC\_FIX\_INVENTAR  *where* ID\_CENTRU\_DE\_COST\_FINAL *is not null  group by* ID\_INVENTAR ) faptic *on* faptic.ID\_INVENTAR = i.ID *where* i.STERS = 0 |
| *--6. Diferentele de cantitate ale produselor si suma acestora pentru inventarele care au data de -- finalizare in 2 luni fata de data sistemului  select* i.nume, mf.NUME,  *abs*(mfi.CANTITATE\_FINALA - mfi.CANTITATE\_INITIALA) *as* DIFERENTE,  sum(*abs*(mfi.CANTITATE\_FINALA - mfi.CANTITATE\_INITIALA)) *over*(*partition by* i.ID) *as* SUMA\_DIFERENTE\_PER\_INVENTAR *from* INVENTAR i *inner join* MIJLOC\_FIX\_INVENTAR mfi *on* mfi.ID\_INVENTAR = i.ID *inner join* MIJLOC\_FIX mf *on* mf.ID = mfi.ID\_MIJLOC\_FIX *and* mf.STERS = 0 *and* mfi.STERS = 0 *where* mfi.ID\_CENTRU\_DE\_COST\_FINAL *is not null  and* mfi.CANTITATE\_INITIALA != mfi.CANTITATE\_FINALA  *and* add\_months(*sysdate*, 2) < i.DATA\_FINALIZARE *order by* i.NUME, *abs*(mfi.CANTITATE\_FINALA - mfi.CANTITATE\_INITIALA) *desc* |
| *--7. Pierderile financiare pentru inventarul cu data minima de incepere  select* i.nume, mf.NUME,  *COALESCE*(*abs*(mfi.CANTITATE\_FINALA - mfi.CANTITATE\_INITIALA), 0) \* mf.VALOARE *as* PIERDERI *from* INVENTAR i *inner join* MIJLOC\_FIX\_INVENTAR mfi *on* mfi.ID\_INVENTAR = i.ID *inner join* MIJLOC\_FIX mf *on* mf.ID = mfi.ID\_MIJLOC\_FIX *and* mf.STERS = 0 *and* mfi.STERS = 0 *where* mfi.ID\_CENTRU\_DE\_COST\_FINAL *is not null  and* mfi.CANTITATE\_FINALA *is not null  and* mfi.CANTITATE\_FINALA < MFI.CANTITATE\_INITIALA  *and* i.DATA\_INCEPERE = (*select min*(i.DATA\_INCEPERE) *from* INVENTAR i) *order by* i.NUME, *abs*(mfi.CANTITATE\_FINALA - mfi.CANTITATE\_INITIALA) \* mf.VALOARE *desc* |
| *--8. Ierarhia angajatilor pentru companiile care au in nume cel putin 2 cuvinte  SELECT a*.NUME, *a*.PRENUME, *a*.EMAIL *FROM* Angajat *a inner join* COMPANIE *c on c*.ID\_MANAGER = *a*.ID *and* INSTR(lower(*c*.NUME), ' ') > 0 *START WITH  a*.ID\_MANAGER *IS NULL CONNECT BY  PRIOR a*.ID = *a*.ID\_MANAGER; |
| *--9. Listează toate companiile care au maxim 3 divizii si minim un angajat alocat.  select* UPPER(*c*.nume) *from* COMPANIE *c inner join* (  *select distinct* z.id\_companie  *from* DIVIZIE div  *inner join* (  *select* ID\_DIVIZIE, count(ID\_ANGAJAT) *as* numar\_angajati  *from* ANGAJAT\_CENTRU\_DE\_COST acc  *inner join* CENTRU\_DE\_COST cc *on* cc.ID = acc.ID\_CENTRU\_DE\_COST  *inner join* DIVIZIE div *on* div.ID = cc.ID\_DIVIZIE  *where* acc.STERS = 0  *group by* ID\_DIVIZIE  ) X *on* X.ID\_DIVIZIE = div.ID *and* X.numar\_angajati > 0  *inner join* (  *select* div.id *as* id\_divizie, *c*.id *as* id\_companie  *from* COMPANIE *c  inner join* DEPARTAMENT d *on* d.ID\_COMPANIE = *c*.ID  *inner join* DIVIZIE div *on* div.ID\_DEPARTAMENT = d.ID  ) Y *ON* Y.id\_divizie = div.ID  *inner join* (  *select c*.id *as* id\_companie, count(div.id) *as* numar\_divizii *from* DIVIZIE div  *inner join* DEPARTAMENT d *on* d.id = div.ID\_DEPARTAMENT  *inner join* COMPANIE *c on c*.id = d.ID\_COMPANIE  *group by c*.id  ) Z *on* z.id\_companie = y.id\_companie *and* Z.numar\_divizii < 4 ) dist *on c*.ID = dist.id\_companie |
| *--10. Companiile care au inventar dar nu au departamente select c*.COD, *c*.NUME *from* COMPANIE *c inner join* INVENTAR i *on* i.ID\_COMPANIE = *c*.ID *where c*.STERS = 0 *MINUS select c*.cod, *c*.nume *from* COMPANIE *c right join* (  *select distinct* d.ID\_COMPANIE  *from* DEPARTAMENT d  *where* d.STERS = 0 ) x *on c*.ID = x.id\_companie |
| *--11. Lista mijloacelor fixe active într-un inventar dar șterse în sistem  select* mf.ID, mf.COD, mf.NUME, mf.VALOARE *from* MIJLOC\_FIX mf *inner join* (  *select distinct* mf.Id *as* id\_mijloc\_fix *from* MIJLOC\_FIX mf  *inner join* MIJLOC\_FIX\_INVENTAR MFI *on* mf.ID = MFI.ID\_MIJLOC\_FIX  *inner join* INVENTAR i *on* i.ID = mfi.ID\_INVENTAR *and* i.sters = 0 ) x *on* x.id\_mijloc\_fix = mf.ID *INTERSECT select* mf.ID, mf.COD, mf.NUME, mf.VALOARE *from* MIJLOC\_FIX mf *where* mf.STERS = 1 |
| *--12. Sa se estimeze valoarea fiecarei companii, in raportul cu mijloacele fixe detinute. --Estimari: --Sub 10000 - MIC --Intre 10000 si 20000 - MEDIU --Peste 20000 - MARE  SELECT  C*.COD, *C*.NUME,  *CASE  WHEN* X.valoare\_companie < 10000 *THEN* 'MIC'  *WHEN* X.valoare\_companie >= 10000 *AND* X.valoare\_companie < 20000 *THEN* 'MEDIU'  *ELSE* 'MARE'  *END AS* Clasificare *FROM* COMPANIE *C INNER JOIN* (  *SELECT C*.ID *AS* id\_companie, NVL(SUM(MF.VALOARE), 0) *as* valoare\_companie  *from* COMPANIE *c  full join* DEPARTAMENT d *on* d.ID\_COMPANIE = *c*.ID *and c*.STERS = 0  *full join* DIVIZIE div *on* div.ID\_DEPARTAMENT = d.ID *and* d.STERS = 0  *full join* CENTRU\_DE\_COST cc *on* cc.ID\_DIVIZIE = div.ID *and* div.STERS = 0  *full join* MIJLOC\_FIX mf *on* mf.ID\_CENTRU\_DE\_COST = cc.ID *and* cc.STERS = 0  *group by c*.ID ) X *ON* X.id\_companie = *C*.ID |
| *--13. Sa se afiseze toate inventarele care au fost adaugate in db in mod gresit, alaturi de o codalitate de contact pentru managerul companiei --pentru care se face invetarul.  --Nereguli: --Data de finalizare < data de incepere sau oricare dintre ele nu este introdusa --Durata de executie a inventarului depaseste 2 saptamani --Niciun mijlloc fix asignat pe invetar  select* i.COD, i.NUME, *c*.NUME *as* nume\_companie, *a*.EMAIL *as* email\_manager, 'Eroare la data de incepere sau cea de finalizare a inventarului!' *as* cauza *from* INVENTAR i *inner join* COMPANIE *c on* i.ID\_COMPANIE = *c*.ID *inner join* ANGAJAT *a on a*.ID = *c*.ID\_MANAGER *where* (DATA\_INCEPERE *is null*) *or* (DATA\_FINALIZARE *is null*) *or* (DATA\_FINALIZARE < DATA\_INCEPERE) *or* (DATA\_FINALIZARE - DATA\_INCEPERE > 14) *union all select* i.COD, i.NUME, *c*.NUME *as* nume\_companie, *a*.EMAIL *as* email\_manager, 'Nu exista niciun mijloc fix in inventar!' *as* cauza *from* INVENTAR i *inner join* COMPANIE *c on* i.ID\_COMPANIE = *c*.ID *inner join* ANGAJAT *a on a*.ID = *c*.ID\_MANAGER *left join* (  *select* i.id, count(\*) *as* numar\_mf  *from* INVENTAR i  *inner join* MIJLOC\_FIX\_INVENTAR mfi *on* mfi.ID\_INVENTAR = i.ID  *where* mfi.STERS = 0  *group by* i.id ) x *on* x.ID = i.ID *where* x.ID *is null* |
| *--14. Sa se afiseze pentru fiecare divizie codurile centrelor de cost care nu au suferit modificari de cand au fost introduse in sistem --Pentru fiecare divizie, sa se afiseze o singura linie  SELECT* d.ID *AS* divizie\_id,  d.Nume *AS* divizie\_nume,  LISTAGG(cc.Nume, ' | ') *WITHIN GROUP* (*ORDER BY* cc.Id) *AS* Centre\_De\_Cost *FROM* DIVIZIE d *INNER JOIN* CENTRU\_DE\_COST cc *ON* cc.ID\_DIVIZIE = d.ID *AND* cc.STERS = 0 *WHERE* d.STERS = 0 *and* d.CREAT\_LA = d.MODIFICAT\_LA *GROUP BY* d.ID, d.Nume *ORDER BY* d.ID; |
| *--Obțineți toți angajații care sunt asociați cu toate centrele de cost dintr-o anumită divizie  select a*.ID, *a*.NUME, *a*.PRENUME, *a*.EMAIL *from* ANGAJAT *a inner join* (  *SELECT DISTINCT* ac.id\_angajat  *FROM* Angajat\_Centru\_De\_Cost ac  *WHERE NOT EXISTS* (  *SELECT* cc.id  *FROM* Centru\_de\_Cost cc  *inner join* DIVIZIE div *on* div.ID = cc.ID\_DIVIZIE *and* div.STERS = 0  *where* div.COD = 'DIV-1'  *AND* cc.sters = 0  *AND NOT EXISTS* (  *SELECT* 1  *FROM* Angajat\_Centru\_De\_Cost ac2  *WHERE* ac2.id\_angajat = ac.id\_angajat  *AND* ac2.id\_centru\_de\_cost = cc.id  )  ) ) X *on* x.ID\_ANGAJAT = *a*.ID |

Cerinta 8

Crearea un tabel de mesaje cu următoarea structură: ...

|  |
| --- |
| *CREATE TABLE* Mesaje (  cod\_mesaj *NUMBER PRIMARY KEY*,  mesaj *VARCHAR2*(255),  tip\_mesaj *VARCHAR2*(1) *CHECK* (tip\_mesaj *IN* ('E', 'W', 'I')),  creat\_de *VARCHAR2*(40) *NOT NULL*,  creat\_la *DATE NOT NULL* ); |

Cerinta 9

Cerinta 10