Baza de date a unui retele de aeroporturi

Realizat de: Iftimi Ruxandra

Cuprins

1. Descrierea modelului
2. Constrangeri (restrictii, reguli)
3. Entitati
4. Relatii
5. Atribute
6. Diagrama Entitate-Relatie
7. Diagrama Conceptuala
8. Schema Relationala
9. Normalizarea
   1. NON FN1 – FN1
   2. NON FN2 – FN2
   3. NON FN3 – FN3
10. Crearea si inserarea datelor
11. Cereri SQL
    1. Cerinta1 + Rezolvare1
    2. Cerinta2 + Rezolvare2
    3. Cerinta3 + Rezolvare3
    4. Cerinta4 + Rezolvare4
    5. Cerinta5 + Rezolvare5
12. Actualizare si Suprimare

1. Descrierea modelului real, a utilității acestuia și a regulilor de funcționare.

Pentru acest proiect, am ales realizarea bazei de date a unei retele de aeroporturi, formata din Aeroportul “Henri Coanda”, Aeroportul “Schphol”, Aeroportul “Charles des Gaulle”, Aeroportul ”Heathrow”, Aeroportul “John F. Kennedy” si Aeroportul “Beijing”. Aceasta are ca entitate principala “AEROPORT” si elementele sale principale componente: pasagerii ce tranziteaza aeroporturile, companiile aeriene ce opereaza pe acele aeroporturi, departamentele ce le alcatuiesc si turnurile lor de control. Pasagerii dispun de bilete individuale, cu bagajul inclus sau fara, si vor sa ajunga la destinatie. Acest lucru se poate realiza cu ajutorul companiilor aeriene care vand biletele de avion pentru zborurile lor, realizate cu avioane din flota acestora, la care lucreaza un numar mare de personal, in vederea prestarii serviciilor care sa se ridice la standardele asteptate. Activitatea aeriana a unui aeroport se realizeaza cu ajutorul turnului sau de control ce se ocupa de desfasurarea in siguranta a zborurilor si coordonarea avioanelor de pe fiecare aeroport in parte. Totodata, la baza activitatii de pe fiecare aeroport, se afla departamentele sale si angajatii lor, menite sa organizeze activitatea interna a acestuia, astfel incat sa nu existe deficiente. Utilitatea principala a acestei baze de date consta in faptul ca, in orice moment, mai ales in cele de urgenta, se pot afla toate informatiile necesare pentru a se detecta problema.

1. Prezentarea constrângerilor (restricții, reguli) impuse asupra modelului.

Restrictii si reguli:

* Departamentele difera in functie de fiecare aeroport
* Biletul poate fi vandut si prin intermediul unei firme/agentii de turism, nu neaparat de o companie aeriana
* Un pasager se identifica unic pentru fiecare zbor la care participa (pentru a nu se suprapune cu un eventual id al altui pasager al altui zbor)
* Fiecare pasager trebuie sa aiba un singur bagaj de cala (bagajele de mana nu intra in aceasta categorie, intrucat ele vor ajunge in cabina)
* Destinatia pasagerilor reprezinta punctul final al calatoriei, deci este unica pentru fiecare dintre ei
* Locatia aeroportului este unica, atat prin identificarea acestuia, cat si prin cea a destinatiei (alegem cel mult un aeroport per oras cu care colaboram)
* Toate zborurile catre o anumita destinatie vor fi unice, intrucat ele vor depinde de identificarea lor din cadrul companiei aeriene si de id-ul acesteia
* Personalul care lucreaza in cadrul companiei aeriene semneaza un contract de angajare care ii impune faptul ca nu mai are voie sa lucreze la alta companie aeriana
* Zborurile sunt unice pentru fiecare aeroport, intrucat depind de id-ul acestuia si id-ul companiei
* Companiile aeriene care opereaza pe un anumit aeroport sunt unice pentru acesta, intrucat isi desfasoara activitatea in functie de conditiile aeroportului(adica TAROM Otopeni are id diferit de TAROM Londra)
* Turnul de control este unic pentru fiecare aeroport
* Un aeroport este alcatuit din cel putin un department care are cel putin un angajat
* Pe un aeroport exista posibilitatea sa nu existe niciun pasager, dar sa existe zboruri de marfuri ce garanteaza activitatea aeroportului
* Pe un aeroport exista posibilitatea sa nu ajunga niciun bagaj daca nu ajunge niciun pasager
* Biletul de avion este unic pentru fiecare pasager si nu este transmisibil
* O companie aeriana trebuie sa aiba cel putin un zbor ca sa opereze
* In cadrul companiei aeriene va exista cel putin un individ care face parte din personal
* Pasagerii pot sa aiba mai multe bilete (pentru escale)
* Biletul de avion poate fi si castigat, deci persoana care intra in posesia lui nu va plati

Constrangeri:

* Cheie primara – PK
* Cheie externa – FK

1. Descrierea entităților, incluzând precizarea cheii primare.

***AEROPORT*** (ID\_AEROPORT#) – entitate ce va contine detaliile despre un aeroport care face parte din reteaua de aeroporturi;

***DEPARTAMENTE*** (ID\_DEPARTAMENT#) – entitate ce va contine detalii despre departamentele din cadrul unui aeroport;

***ANGAJATI***(ID\_ANGAJAT#) – entitate ce va contine detalii despre angajatii dintr-un department anume;

***TURN\_CONTROL*** (ID\_TURN\_CONTROL#) – entitate ce va contine detalii despre turnul de control al unui aeroport;

***COMPANII\_AERIENE*** (ID\_COMPANIE#) – entitate ce va contine detalii despre o companie aeriana ce opereaza pe un anumit aeroport;

***AVIOANE*** (ID\_AVION#) – entitate ce va contine detalii despre avioanele detinute de o companie aeriana;

***ZBORURI*** (ID\_ZBOR#) – entitate ce va contine detalii despre zborurile unei companii aeriene de pe un aeroport;

***PERSONAL*** (ID\_PERSONAL#) – entitate ce va contine detalii despre personalul din cadrul companiei aeriene;

***PERSONAL\_ORGANIZATORIC***(ID\_PERSONAL\_ ORGANIZATORIC#) – entitate ce va contine detalii despre personalul organizatoric din cadrul companiei;

***PERSONAL\_ZBOR*** (ID\_PERSONAL\_ ZBOR#) – entitate ce va contine detalii despre personalul de zbor din cadrul companiei;

***BILETE*** (ID\_BILET#) – entitate ce va contine detalii despre biletul unui pasager;

***PASAGERI*** (ID\_PASAGER#) – entitate ce va contine detalii despre pasagerii care tranziteaza un aeroport anume;

***BAGAJE*** (ID\_BAGAJ#) – entitate ce va contine detalii despre bagajul unui pasager;

***DESTINATIE*** (ID\_DESTINATIE#) – entitate ce va contine detalii despre destinatia pasagerilor;

1. Descrierea relațiilor, incluzând precizarea cardinalității acestora.

***AEROPORT--DEPARTAMENTE***: un aeroport este alcatuit din mai multe departamente, dar un department este unic pentru un aeroport.

* Tipul relatiei: many-one
* Cardinalitate: M(1)-1

***DEPARTAMENTE--ANGAJATI***: intr-un departament pot sa lucreze mai multi angajati, dar un angajat poate sa lucreze in cel mult un department.

* Tipul relatiei: many-one
* Cardinalitate: M(1)-1

***AEROPORT--PASAGERI***: pe un aeroport ajung mai multi pasageri, dar si un pasager poate ajunge pe mai multe aeroporturi.

* Tipul relatiei: many-many
* Cardinalitate: M(0)-M(1)

***PASAGERI--BAGAJE***: pasagerii pot sa aiba mai multe bagaje, dar un bagaj trebuie sa fie detinut numai de o singura persoana.

* Tipul relatiei: many-one
* Cardinalitate: M(1)-1

***AEROPORT--BAGAJE***: pe un aeroport ajung mai multe bagaje, dar si un bagaje poate ajunge pe mai multe aeroporturi.

* Tipul relatiei: many-many
* Cardinalitate: M(1)-M(0)

***PASAGERI--BILETE***: pasagerii pot sa aiba mai multe bilete, dar biletul este individual.

* Tipul relatiei: many-one
* Cardinalitate: M(1)-1

***PASAGERI--DESTINATIE***: pasagerii poate sa aiba o unica destinatie, dar o destinatie poate sa aiba mai multi pasageri.

* Tipul relatiei: one-many
* Cardinalitate: 1-M(1)

***AEROPORT--DESTINATIE***: aeroportul este situat intr-o singura destinatie, dar si destinatia este unica pentru aeroport.

* Tipul relatiei: one-one
* Cardinalitate: 1-1

***AEROPORT--TURN\_CONTROL***: un aeroport poate sa aiba un singur turn de control, dar turnul de control este unic pentru fiecare aeroport.

* Tipul relatiei: one-one
* Cardinalitate: 1-1

***AEROPORT--COMPANII\_AERIENE***: un aeroport poate sa aiba multe companii aeriene partenere, dar si aceeasi companie aeriana poate sa opereze pe mai multe aeroporturi. (constrangere)

* Tipul relatiei: many-one
* Cardinalitate: M(1)-1

***COMPANII\_AERIENE--ZBORURI***: o companie aeriana poate sa aiba mai multe zboruri, dar un zbor poate fi operat numai de o companie aeriana.

* Tipul relatiei: many-one
* Cardinalitate: M(1)-1

***COMPANII\_AERIENE--AVIOANE***: o companie aeriana poate sa detina mai multe avioane, dar un avion poate sa fie detinut numai de o singura companie aeriana.

* Tipul relatiei: many-one
* Cardinalitate: M(1)-1

***COMPANII\_AERIENE--PERSONAL***: o companie poate sa aiba un numar mare de personal, dar un membru al personalului poate sa lucreze numai la o singura companie aeriana.

* Tipul relatiei: many-one
* Cardinalitate: M(1)-1

***AEROPORT--ZBORURI***: pe un aeroport ajung mai multe zboruri, dar un zbor ajunge pe un singur aeroport.

* Tipul relatiei: many-one
* Cardinalitate: M(1)-1

***PERSONAL--PERSONAL\_ORGANIZATORIC***: personalul poate fi organizatoric.

* Tipul relatiei: ISA
* Cardinalitate: 1-1(0)

***PERSONAL--PERSONAL\_ZBOR***: personalul poate fi pentru zbor.

* Tipul relatiei: ISA
* Cardinalitate: 1-1(0)

1. Descrierea atributelor, incluzând tipul de date și eventualele constrângeri, valori implicite, valori posibile ale atributelor.

***AEROPORT***

ID\_AEROPORT# int not null

NUME\_AEROPORT varchar2(50)

LOCATIE\_AEROPORT varchar2(50)

NUMAR\_ZBORURI\_AN int

NUMAR\_CALATORI\_AN long

ID\_AEROPORT# => cheie primara de tip int, diferita de null, care identifica unic un aeroport din cadrul retelei de aeroporturi (valori de 4 cifre)

NUME\_AEROPORT => atribut de tip sir de caractere, de lungime maxima 50, ce va contine numele aeroportului

LOCATIE\_AEROPORT => atribut de tip sir de caractere, de lungime maxima 50, ce va contine orasul in care se afla aeroportul

NUMAR\_ZBORURI\_AN => atribut de tip int ce va contine numarul de zboruri pe an ale aeroportului

NUMAR\_CALATORI\_AN => atribut de tip long ce va contine numarul de pasageri care tranziteaza aeroportul pe an

***DEPARTAMENTE***

ID\_DEPARTAMENT# int not null

NUME\_DEPARTAMENT varchar2(30)

NUMAR\_ANGAJATI int

FK ID\_AEROPORT

ID\_DEPARTAMENT # => cheie primara de tip int, diferita de null, care identifica unic un departament din cadrul unui aeroport (valori de 4 cifre)

NUME\_DEPARTAMENT => atribut de tip sir de caractere, de lungime maxima 30, ce va contine numele departamentului

NUMAR\_ANGAJATI => atribut de tip int ce va contine numarul de angajati dintr-un departament

FK ID\_AEROPORT => cheie externa ce va identifica departamentele care se afla pe un anumit aeroport (constrangere -vezi exercitiul 2- departamentele difera in functie de fiecare aeroport)

***ANGAJATI***

ID\_ANGAJAT# int not null

NUME\_ANGAJAT varchar2(20)

DATA\_ANGAJARII varchar2(30)

SALARIU\_ANGAJATI int

FK ID\_DEPARTAMENT

ID\_ANGAJAT # => cheie primara de tip int, diferita de null, care identifica unic un angajat din cadrul unui departament (valori de 3 cifre)

NUME\_ANGAJAT => atribut de tip sir de caractere, de lungime maxima 20, ce va contine numele de familie al angajatului

DATA\_ANGAJARII => atribut de tip sir de caractere, de lungime maxima 30, ce va contine data angajarii a persoanei respective

SALARIU\_ANGAJATI => atribut de tip int ce va contine salariul angajatului

FK ID\_DEPARTAMENT => cheie externa ce va identifica angajatii care lucreaza intr-un anumit department

***TURN\_CONTROL***

ID\_TURN\_CONTROL# int not null

NUMAR\_AERONAVE\_COORDONATE long

NUMAR\_CONTROLORI\_TRAFIC int

FK ID\_AEROPORT

ID\_TURN\_CONTROL # => cheie primara de tip int, diferita de null, care identifica unic turnul de control al unui aeroport din cadrul retelei de aeroporturi (valori de 3 cifre)

NUMAR\_AERONAVE\_COORDONATE => atribut de tip long ce va contine numarul de avioane coordinate de turnul de control

NUMAR\_CONTROLORI\_TRAFIC => atribut de tip int ce va contine numarul de controlori care lucreaza in turnul de control respectiv

FK ID\_AEROPORT => cheie externa ce va identifica turnul de control care se afla pe un anumit aeroport

***COMPANII\_AERIENE***

ID\_COMPANIE# int not null

NUME\_COMPANIE varchar2(50)

NUMAR\_AVIOANE\_COMPANIE long

ORAS\_SEDIU varchar2(30)

FK ID\_AEROPORT

ID\_COMPANIE # => cheie primara de tip int, diferita de null, care identifica unic o campanie aeriana care se afla pe un aeroport anume (valori de 3 cifre; constrangere -vezi exercitiul 2- companiile aeriene care opereaza pe un anumit aeroport sunt unice pentru acesta, intrucat isi desfasoara activitatea in functie de conditiile aeroportului)

NUME\_COMPANIE => atribut de tip sir de caractere, de lungime maxima 50, ce va contine numele companiei aeriene

NUMAR\_AVIOANE\_COMPANIE => atribut de tip long ce va contine numar de avioane detinute de companie

ORAS\_SEDIU => atribut de tip sir de caractere, de lungime maxima 30, ce va contine orasul sediului companiei aeriene

FK ID\_AEROPORT => cheie externa ce va identifica aeroportul pe care se opereaza compania aeriana

***AVIOANE***

ID\_AVION# int not null

TIP\_AVION varchar2(20)

FK ID\_COMPANIE

ID\_AVION # => cheie primara de tip int, diferita de null, care identifica unic avion (valori de 2 cifre)

TIP\_AVION => atribut de tip sir de caractere, de lungime maxima 20, ce va contine tipul avionului

FK ID\_ COMPANIE => cheie externa ce va identifica compania care detine avionul

***ZBORURI***

ID\_ZBOR# int not null

DESTINATIE\_ZBOR varchar2(30)

DURATA\_CALATORIEI\_IN\_ORE int

FK ID\_COMPANIE

FK ID\_AEROPORT

ID\_ZBOR # => cheie primara de tip int, diferita de null, care identifica unic un zbor (valori de 4 cifre)

DESTINATIE\_ZBOR => atribut de tip sir de caractere, de lungime maxima 30, ce va contine orasul in care va ajunge zborul

DURATA\_CALATORIEI\_IN\_ORE => atribut de tip int ce va contine durata calatoriei

FK ID\_ COMPANIE => cheie externa ce va compania care opereaza zborul

FK ID\_AEROPORT => cheie externa ce va identifica aeroportul pe care va ateriza zborul

***PERSONAL***

ID\_PERSONAL# int not null

NUME\_PERSONAL varchar2(20)

DEPARTAMENT\_PERSONAL varchar2(20)

SALARIU\_PERSONAL int

FK ID\_COMPANIE

ID\_PERSONAL # => cheie primara de tip int, diferita de null, care identifica unic un individ care face parte din personal (valori de 3 cifre)

NUME\_PERSONAL => atribut de tip sir de caractere, de lungime maxima 20, ce va contine numele de familie al unui individ care face parte din personal

DEPARTAMENT\_PERSONAL => atribut de tip sir de caractere, de lungime maxima 20, ce va contine departamentul din care face parte un individ care face parte din personal

SALARIU\_PERSONAL => atribut de tip int ce va contine salariul mediu dintr-un departament

FK ID\_ COMPANIE => cheie externa ce va identifica compania din cadrul careia face parte un individ care face parte din personal

***PERSONAL\_ORGANIZATORIC***

ID\_PERSONAL\_ORGANIZATORIC# int not null

NUMAR\_ORE\_LUCRATE int

FUNCTIE\_ORGANIZATORIC varchar2(20)

DEPARTAMENT\_ORGANIZATORIC varchar2(20)

FK ID\_PERSONAL

ID\_PERSONAL\_ORGANIZATORIC # => cheie primara de tip int, diferita de null, care identifica unic angajat din cadrul personalului organizatoric (valori de 3 cifre)

NUMAR\_ORE\_LUCRATE => atribut de tip int ce va contine numarul de ore lucrate in cadrul departamentului din personalul organizatoric

FUNCTIE\_ORGANIZATORIC => atribut de tip sir de caractere, de lungime maxima 20, ce va contine functia unui angajat in cadrul personalului organizatoric

DEPARTAMENT\_ORGANIZATORIC => atribut de tip sir de caractere, de lungime maxima 20, ce va contine departamentul unui angajat in cadrul personalului organizatoric

FK ID\_ PERSONAL => cheie externa ce va identifica angajatul si in cadrul personalului

***PERSONAL\_ZBOR***

ID\_PERSONAL\_ZBOR# int not null

CATEGORIE\_PERSONAL\_ZBOR varchar2(20)

NUMAR\_ZBORURI\_ACTIV int

ORE\_EXPERIENTA int

FK ID\_PERSONAL

ID\_PERSONAL\_ZBOR # => cheie primara de tip int, diferita de null, care identifica unic un angajat din cadrul personalului de zbor (valori de 3 cifre)

CATEGORIE\_PERSONAL\_ZBOR => atribut de tip sir de caractere, de lungime maxima 20, ce va contine categoria in cadrul personalului de zbor a unui individ care lucreaza in cadrul personalului de zbor

NUMAR\_ZBORURI\_ACTIV => atribut de tip int ce va contine numarul de zboruri pe care le-a efectuat/la care a luat parte un individ care lucreaza in cadrul personalului de zbor

ORE\_EXPERIENTA => atribut de tip int ce va contine numarul de ore de experienta ale unui unui individ care lucreaza in cadrul personalului de zbor

FK ID\_ PERSONAL => cheie externa ce va identifica angajatul si in cadrul personalului

***PASAGERI***

ID\_PASAGER# int not null

NUME\_PASAGER varchar2(20)

NATIONALITATE varchar2(30)

TARA\_DE\_PROVENIENTA varchar2(30)

FK ID\_AEROPORT

ID\_PASAGER # => cheie primara de tip int, diferita de null, care identifica unic un pasager care ajunge pe unul din aeroporturile din cadrul rete;ei de aeroporturi (valori de 2/3 cifre)

NUME\_PASAGER => atribut de tip sir de caractere, de lungime maxima 20, ce va contine numelede familike al pasagerului

NATIONALITATE => atribut de tip sir de caractere, de lungime maxima 30, ce va contine nationalitatea pasagerului

TARA\_DE\_PROVENIENTA => atribut de tip sir de caractere, de lungime maxima 30, ce va contine tara de provenienta a pasagerului

FK ID\_AEROPORT => cheie externa ce va identifica aeroportul pe care ajunge pasagerul

***BILETE***

ID\_BILET# int not null

OPERATOR\_BILET varchar2(50)

PRET int

FK ID\_PASAGER

ID\_BILET # => cheie primara de tip int, diferita de null, care identifica unic un bilet de avion (valori de 3 cifre)

OPERATOR\_BILET => atribut de tip sir de caractere, de lungime maxima 50, ce va operatorul care a vandut biletul de avion

PRET => atribut de tip int ce va contine pretul biletului

FK ID\_PASAGER => cheie externa ce va identifica pasagerul caruia ii apartine biletul

***BAGAJE***

ID\_BAGAJ# int not null

DESTINATIE\_FINALA\_BAGAJ varchar2(30)

FK ID\_PASAGER

FK ID\_AEROPORT

ID\_BAGAJ # => cheie primara de tip int, diferita de null, care identifica unic un bagaj (valori de 2 cifre)

DESTINATIE\_FINALA\_BAGAJ => atribut de tip sir de caractere, de lungime maxima 30, ce va contine destinatia finala a bagajului

FK ID\_PASAGER => cheie externa ce va identifica pasagerul caruia ii apartine bagajul

FK ID\_ AEROPORT => cheie externa ce va identifica aeroportul pe care va ajunge bagajul

***DESTINATIE***

ID\_DESTINATIE# int not null

ORA\_SOSIRII int

FK ID\_PASAGER

FK ID\_AEROPORT

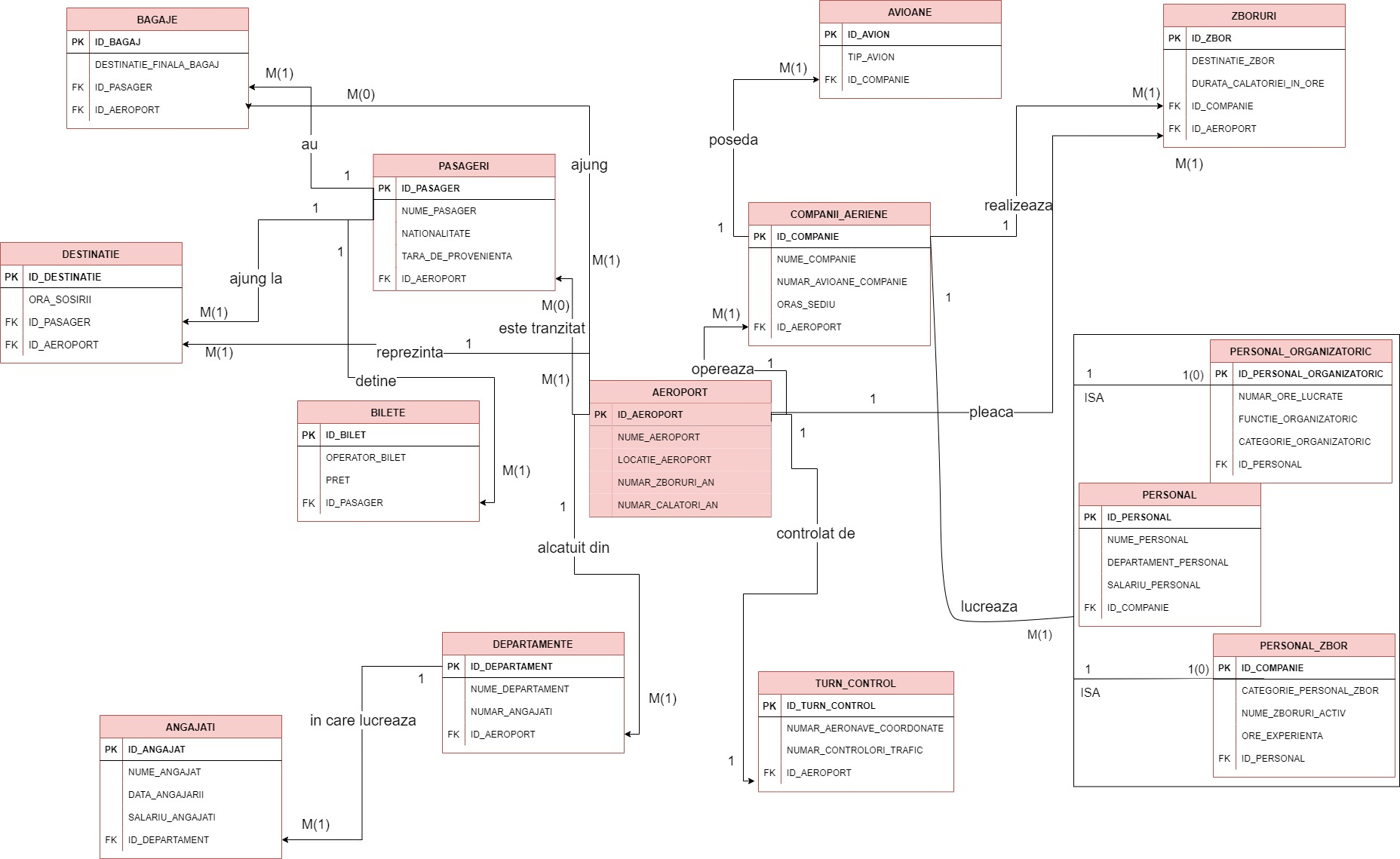
ID\_DESTINATIE # => cheie primara de tip int, diferita de null, care identifica unic o destinatie (valori de 4 cifre)

ORA\_SOSIRII => atribut de tip int ce va ora sosirii la destinatie

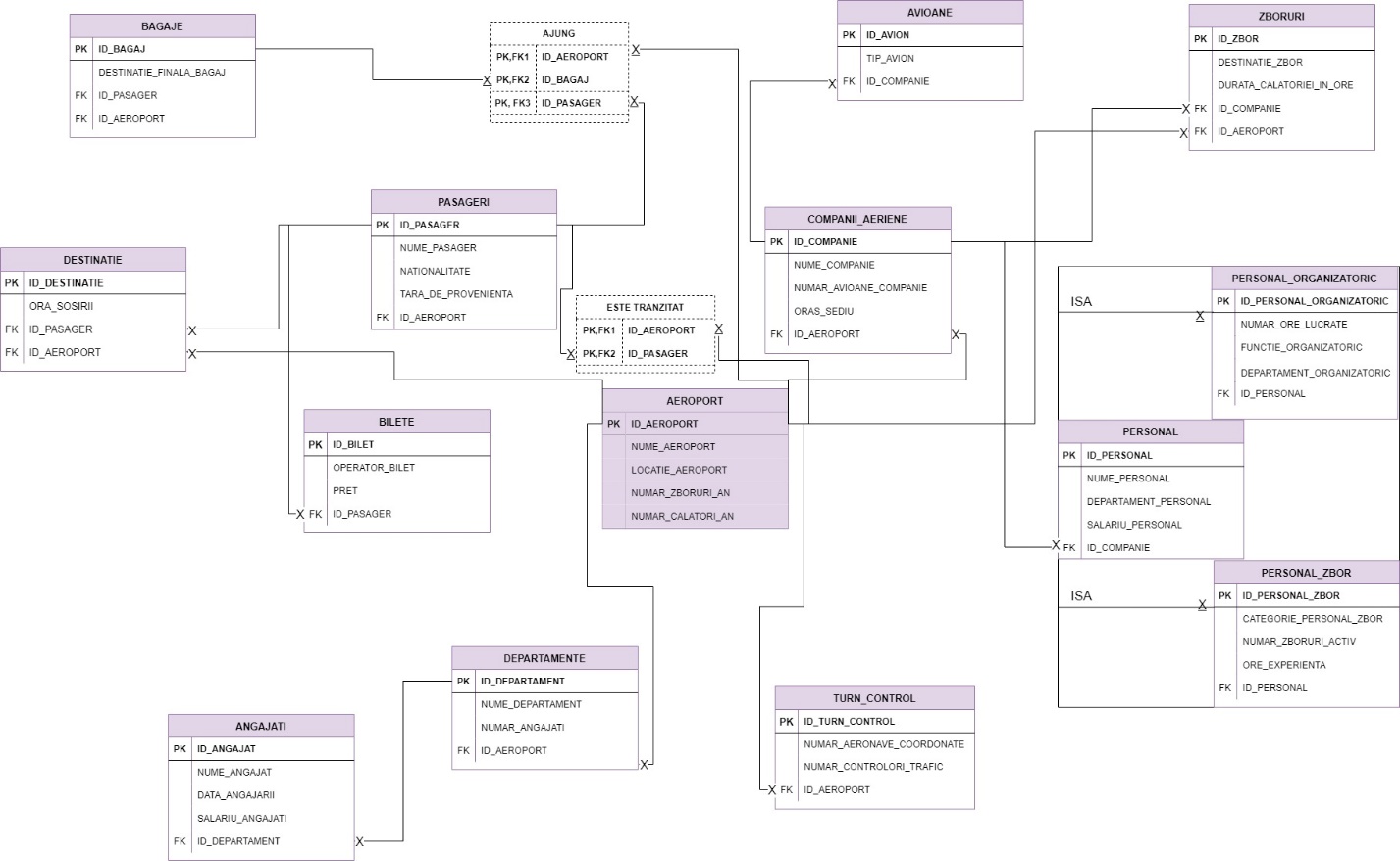
FK ID\_PASAGER => cheie externa ce va identifica pasagerul care ajunge la destinatie

FK ID\_ AEROPORT => cheie externa ce va identifica aeroportul care se afla la destinatie

1. Realizarea diagramei entitate-relație corespunzătoare descrierii de la punctele 3-5.



1. Realizarea diagramei conceptuale corespunzătoare diagramei entitate-relație proiectate la punctul 6. Diagrama conceptuală obținută trebuie să conțină minimum 6 tabele (fără considerarea subentităților), dintre care cel puțin un tabel asociativ.



1. Enumerarea schemelor relaționale corespunzătoare diagramei conceptuale proiectate la punctul 7.

***AEROPORT***

(ID\_AEROPORT#, NUME\_AEROPORT, LOCATIE\_AEROPORT, NUMAR\_ZBORURI\_AN, NUMAR\_CALATORI\_AN)

***DEPARTAMENTE***

(ID\_DEPARTAMENT#, NUME\_DEPARTAMENT, NUMAR\_ANGAJATI, ID\_AEROPORT)

***ANGAJATI***

(ID\_ANGAJAT#, NUME\_ANGAJAT, DATA\_ANGAJARII, SALARIU\_ANGAJATI, ID\_DEPARTAMENT)

***TURN\_CONTROL***

(ID\_TURN\_CONTROL#, NUMAR\_AERONAVE\_COORDONATE, NUMAR\_CONTROLORI\_TRAFIC, ID\_AEROPORT)

***COMPANII\_AERIENE***

(ID\_COMPANIE#, NUME\_COMPANIE, NUMAR\_AVIOANE\_COMPANIE, ORAS\_SEDIU, ID\_AEROPORT)

***AVIOANE***

(ID\_AVION#, TIP\_AVION, ID\_COMPANIE)

***ZBORURI***

(ID\_ZBOR#, DESTINATIE\_ZBOR, DURATA\_CALATORIEI\_IN\_ORE, ID\_COMPANIE, ID\_AEROPORT)

***PERSONAL***

(ID\_PERSONAL#, NUME\_PERSONAL, DEPARTAMENT\_PERSONAL, SALARIU\_PERSONAL, ID\_COMPANIE)

***PERSONAL\_ORGANIZATORIC***

(ID\_PERSONAL\_ ORGANIZATORIC#, NUMAR\_ORE\_LUCRATE, FUNCTIE\_ORGANIZATORIC, CATEGORIE\_ORGANIZATORIC, ID\_PERSONAL)

***PERSONAL\_ZBOR***

(ID\_PERSONAL\_ ZBOR#, CATEGORIE\_PERSONAL\_ZBOR, NUMAR\_ZBORURI\_ACTIV, ORE\_EXPERIENTA, ID\_PERSONAL)

***BILETE***

(ID\_BILET#, OPERATOR\_BILET, PRET, ID\_PASAGER)

***PASAGERI***

(ID\_PASAGER#, NUME\_PASAGER, NATIONALITATE, TARA\_DE\_PROVENIENTA, ID\_AEROPORT)

***BAGAJE***

(ID\_BAGAJ#, DESTINATIE\_FINALA\_BAGAJ, ID\_PASAGER, ID\_AEROPORT)

***DESTINATIE***

(ID\_DESTINATIE#, ORA\_SOSIRII, ID\_PASAGER, ID\_AEROPORT)

***ESTE TRANZITAT***

(ID\_PASAGER#, ID\_AEROPORT#)

***AJUNG***

(ID\_AEROPORT#, ID\_BAGAJ#, ID\_PASAGER#)

1. Realizarea normalizării până la forma normală 3 (FN1-FN3).

**NON FN1:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID\_PASAGER | NUME\_INTREG\_PASAGER | NATIONALITATE | TARA\_DE\_PROVENIENTA | ID\_AEROPORT |
| 100 | Stefan King | Romana | Italia | 1000 |
| 101 | Cassie Liu | Chineza | Australia | 1500 |
| 102 | Ernest Lott | Britanica | SUA | 1200 |
| 103 | Noel Bishop | Canadiana | Africa de Sud | 1100 |
| 104 | Macy Medina | Italiana | Franta | 1300 |
| 105 | August Franks | Americana | Ucraina | 1400 |
| 90 | Constantin Popa | Romana | Germania | 1000 |
| 91 | Sameera Phan | Chineza | Japonia | 1500 |
| 92 | Neha Edwards | Irlandeza | Argentina | 1200 |
| 93 | Koa Bass | Britanica | Egipt | 1100 |
| 94 | Arooj Senior | Canadiana | Ungaria | 1300 |
| 95 | Elissa Mcclain | Americana | Finlanda | 1400 |

**FN1:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID\_PASAGER | PRENUME\_  PASAGER | NUME\_  PASAGER | NATIONALITATE | TARA\_DE\_  PROVENIENTA | ID\_  AEROPORT |
| 100 | Stefan | King | Romana | Italia | 1000 |
| 101 | Cassie | Liu | Chineza | Australia | 1500 |
| 102 | Ernest | Lott | Britanica | SUA | 1200 |
| 103 | Noel | Bishop | Canadiana | Africa de Sud | 1100 |
| 104 | Macy | Medina | Italiana | Franta | 1300 |
| 105 | August | Franks | Americana | Ucraina | 1400 |
| 90 | Constantin | Popa | Romana | Germania | 1000 |
| 91 | Sameera | Phan | Chineza | Japonia | 1500 |
| 92 | Neha | Edwards | Irlandeza | Argentina | 1200 |
| 93 | Koa | Bass | Britanica | Egipt | 1100 |
| 94 | Arooj | Senior | Canadiana | Ungaria | 1300 |
| 95 | Elissa | Mcclain | Americana | Finlanda | 1400 |

**NON FN2:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID\_DEPARTAMENT | NUME\_DEPARTAMENT | NUMAR\_ANGAJATI | SALARIU\_MEDIU | ID\_AEROPORT |
| 1001 | RESURSE UMANE | 100 | 5000 | 1000 |
| 1002 | PR | 100 | 5000 | 1000 |
| 1003 | LOGISTICA | 50 | 3000 | 1100 |
| 1004 | COMERCIAL | 40 | 3500 | 1100 |
| 1005 | MARKETING | 25 | 4000 | 1200 |
| 1006 | CONTROL | 10 | 700 | 1200 |
| 1007 | SECURITARE | 5 | 10000 | 1300 |
| 1008 | OPERATIUNI | 80 | 5500 | 1300 |
| 1009 | MANAGEMENT | 45 | 6000 | 1400 |
| 1010 | ACHIZITII | 30 | 4400 | 1400 |
| 1011 | FINANTE | 100 | 6500 | 1500 |
| 1012 | DEZVOLTARE | 30 | 8000 | 1500 |
| 1013 | DIRECTOR\_1000 | 5 | 10000 | 1000 |
| 1014 | DIRECTOR\_1100 | 80 | 5500 | 1100 |
| 1015 | DIRECTOR\_1200 | 45 | 6000 | 1200 |
| 1016 | DIRECTOR\_1300 | 30 | 4400 | 1300 |
| 1017 | DIRECTOR\_1400 | 100 | 6500 | 1400 |
| 1018 | DIRECTOR\_1500 | 30 | 8000 | 1500 |

**FN2:**

|  |  |
| --- | --- |
| ID\_DEPARTAMENT | SALARIU\_MEDIU |
| 1001 | 5000 |
| 1002 | 5000 |
| 1003 | 3000 |
| 1004 | 3500 |
| 1005 | 4000 |
| 1006 | 700 |
| 1007 | 10000 |
| 1008 | 5500 |
| 1009 | 6000 |
| 1010 | 4400 |
| 1011 | 6500 |
| 1012 | 8000 |
| 1013 | 10000 |
| 1014 | 5500 |
| 1015 | 6000 |
| 1016 | 4400 |
| 1017 | 6500 |
| 1018 | 8000 |

|  |  |  |  |
| --- | --- | --- | --- |
| ID\_DEPARTAMENT | NUME\_DEPARTAMENT | NUMAR\_ANGAJATI | ID\_AEROPORT |
| 1001 | RESURSE UMANE | 100 | 1000 |
| 1002 | PR | 100 | 1000 |
| 1003 | LOGISTICA | 50 | 1100 |
| 1004 | COMERCIAL | 40 | 1100 |
| 1005 | MARKETING | 25 | 1200 |
| 1006 | CONTROL | 10 | 1200 |
| 1007 | SECURITARE | 5 | 1300 |
| 1008 | OPERATIUNI | 80 | 1300 |
| 1009 | MANAGEMENT | 45 | 1400 |
| 1010 | ACHIZITII | 30 | 1400 |
| 1011 | FINANTE | 100 | 1500 |
| 1012 | DEZVOLTARE | 30 | 1500 |
| 1013 | DIRECTOR\_1000 | 5 | 1000 |
| 1014 | DIRECTOR\_1100 | 80 | 1100 |
| 1015 | DIRECTOR\_1200 | 45 | 1200 |
| 1016 | DIRECTOR\_1300 | 30 | 1300 |
| 1017 | DIRECTOR\_1400 | 100 | 1400 |
| 1018 | DIRECTOR\_1500 | 30 | 1500 |

**NON FN3:**

|  |  |  |  |
| --- | --- | --- | --- |
| ID\_AVION | TIP\_AVION | LOCURI\_AVION | ID\_COMPANIE |
| 50 | Boeing 707 | 20 | 800 |
| 51 | Boeing 787 | 25 | 800 |
| 52 | Airbus A300 | 40 | 810 |
| 53 | Airbus A340 | 30 | 810 |
| 54 | Airbus A310 | 65 | 820 |
| 55 | Airbus A330 | 55 | 820 |
| 56 | Boeing 717 | 35 | 830 |
| 57 | ATR 42 | 20 | 830 |
| 58 | Airbus A380 | 43 | 840 |
| 59 | Airbus A320 | 48 | 840 |
| 60 | Boeing 777 | 33 | 850 |
| 61 | Boeing 747 | 27 | 850 |

**FN3:**

|  |  |  |
| --- | --- | --- |
| ID\_AVION | TIP\_AVION | ID\_COMPANIE |
| 50 | Boeing 707 | 800 |
| 51 | Boeing 787 | 800 |
| 52 | Airbus A300 | 810 |
| 53 | Airbus A340 | 810 |
| 54 | Airbus A310 | 820 |
| 55 | Airbus A330 | 820 |
| 56 | Boeing 717 | 830 |
| 57 | ATR 42 | 830 |
| 58 | Airbus A380 | 840 |
| 59 | Airbus A320 | 840 |
| 60 | Boeing 777 | 850 |
| 61 | Boeing 747 | 850 |

1. Crearea tabelelor în SQL și inserarea de date coerente în fiecare dintre acestea (minimum 5 înregistrări în fiecare tabel neasociativ; minimum 10 înregistrări în tabelele asociative).

--CREARE--

create table AEROPORT(

ID\_AEROPORT int not null primary key,

NUME\_AEROPORT varchar2(50),

LOCATIE\_AEROPORT varchar2(50),

NUMAR\_ZBORURI\_AN int,

NUMAR\_CALATORI\_AN long

);

--drop table AEROPORT;

create table DEPARTAMENTE(

ID\_DEPARTAMENT int not null primary key,

NUME\_DEPARTAMENT varchar2(30),

NUMAR\_ANGAJATI int,

ID\_AEROPORT not null,

foreign key (ID\_AEROPORT) references AEROPORT(ID\_AEROPORT)

);

--drop table DEPARTAMENTE;

create table ANGAJATI(

ID\_ANGAJAT int not null primary key,

NUME\_ANGAJAT varchar2(20),

DATA\_ANGAJARII varchar2(30),

SALARIU\_ANGAJATI int,

ID\_DEPARTAMENT not null,

foreign key(ID\_DEPARTAMENT) references DEPARTAMENTE(ID\_DEPARTAMENT)

);

--drop table ANGAJATI;

create table TURN\_CONTROL(

ID\_TURN\_CONTROL int not null primary key,

NUMAR\_AERONAVE\_COORDONATE long,

NUMAR\_CONTROLORI\_TRAFIC int,

ID\_AEROPORT not null,

foreign key(ID\_AEROPORT) references AEROPORT(ID\_AEROPORT)

);

--drop table TURN\_CONTROL;

create table COMPANII\_AERIENE(

ID\_COMPANIE int not null primary key,

NUME\_COMPANIE varchar2(50),

NUMAR\_AVION\_COMPANIE long,

ORAS\_SEDIU varchar2(30),

ID\_AEROPORT not null,

foreign key(ID\_AEROPORT) references AEROPORT(ID\_AEROPORT)

);

--drop table COMPANII\_AERIENE;

create table AVIOANE(

ID\_AVION int not null primary key,

TIP\_AVION varchar2(20),

ID\_COMPANIE not null,

foreign key(ID\_COMPANIE) references COMPANII\_AERIENE(ID\_COMPANIE)

);

--drop table AVIOANE;

create table ZBORURI(

ID\_ZBOR int not null primary key,

DESTINATIE\_ZBOR varchar2(30),

DURATA\_CALATORIEI\_IN\_ORE int,

ID\_AEROPORT not null,

ID\_COMPANIE not null,

foreign key(ID\_COMPANIE) references COMPANII\_AERIENE(ID\_COMPANIE),

foreign key(ID\_AEROPORT) references AEROPORT(ID\_AEROPORT)

);

--drop table ZBORURI;

create table PERSONAL(

ID\_PERSONAL int not null primary key,

NUME\_PERSONAL varchar2(20),

DEPARTAMENT\_PERSONAL varchar2(20),

SALARIU\_PERSONAL int,

ID\_COMPANIE not null,

foreign key(ID\_COMPANIE) references COMPANII\_AERIENE(ID\_COMPANIE)

);

--drop table PERSONAL;

create table PERSONAL\_ORGANIZATORIC (

ID\_PERSONAL\_ORGANIZATORIC int not null primary key,

NUMAR\_ORE\_LUCRATE int,

FUNCTIE\_ORGANIZATORIC varchar2(20),

DEPARTAMENT\_ORGANIZATORIC varchar2(20),

ID\_PERSONAL not null,

foreign key(ID\_PERSONAL) references PERSONAL(ID\_PERSONAL)

);

--drop table PERSONAL\_ORGANIZATORIC;

create table PERSONAL\_ZBOR (

ID\_PERSONAL\_ZBOR int not null primary key,

CATEGORIE\_PERSONAL\_ZBOR varchar2(20),

NUMAR\_ZBORURI\_ACTIV int,

ORE\_EXPERIENTA int,

ID\_PERSONAL not null,

foreign key(ID\_PERSONAL) references PERSONAL(ID\_PERSONAL)

);

--drop table PERSONAL\_ZBOR;

create table PASAGERI (

ID\_PASAGER int not null primary key,

NUME\_PASAGER varchar2(20),

NATIONALITATE varchar2(30),

TARA\_DE\_PROVENIENTA varchar2(30),

ID\_AEROPORT not null,

foreign key(ID\_AEROPORT) references AEROPORT(ID\_AEROPORT)

);

--drop table PASAGERI;

create table BILETE (

ID\_BILET int not null primary key,

OPERATOR\_BILET varchar2(50),

PRET int,

ID\_PASAGER not null,

foreign key(ID\_PASAGER) references PASAGERI(ID\_PASAGER)

);

--drop table BILETE;

create table BAGAJE (

ID\_BAGAJ int not null primary key,

DESTINATIE\_FINALA\_BAGAJ varchar2(30),

ID\_AEROPORT not null,

ID\_PASAGER not null,

foreign key(ID\_AEROPORT) references AEROPORT(ID\_AEROPORT),

foreign key(ID\_PASAGER) references PASAGERI(ID\_PASAGER)

);

--drop table BAGAJE;

create table DESTINATIE (

ID\_DESTINATIE int not null primary key,

ORA\_SOSIRII int,

ID\_AEROPORT not null,

ID\_PASAGER not null,

foreign key(ID\_AEROPORT) references AEROPORT(ID\_AEROPORT),

foreign key(ID\_PASAGER) references PASAGERI(ID\_PASAGER)

);

--drop table DESTINATIE;

create table ESTE\_TRANZITAT (

ID\_PASAGER int not null,

ID\_AEROPORT int not null,

constraint pk\_este\_tranzitat primary key(ID\_PASAGER, ID\_AEROPORT)

);

-- drop table ESTE\_TRANZITAT;

create table AJUNG (

ID\_PASAGER int not null,

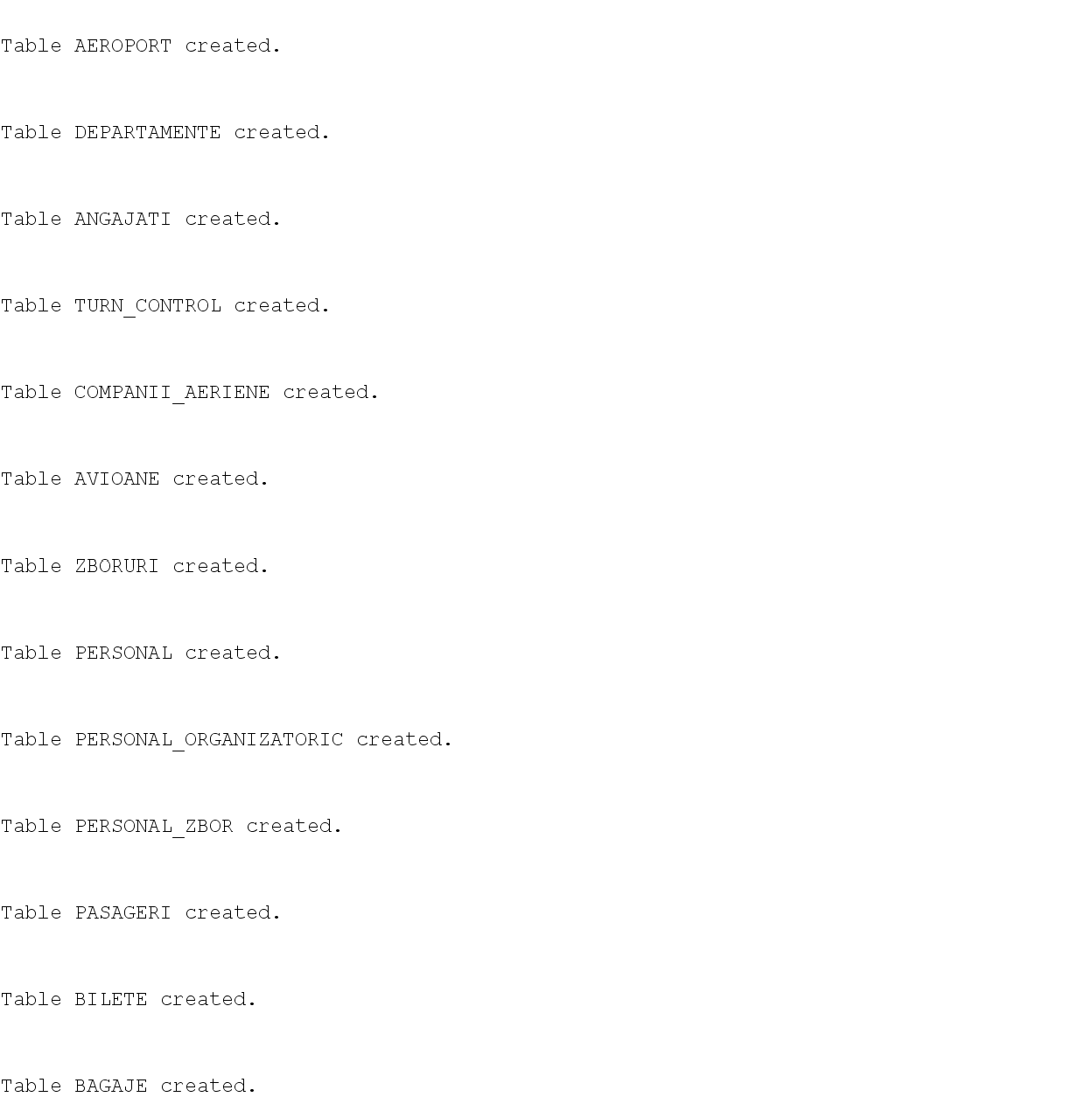
ID\_AEROPORT int not null,

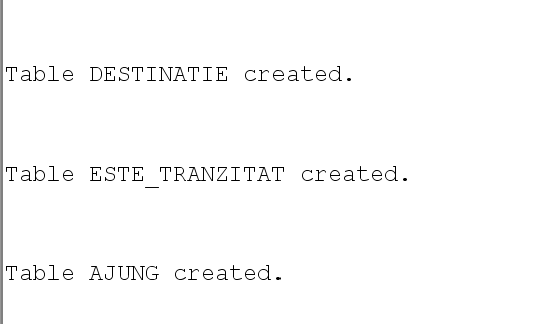
ID\_BAGAJ int not null,

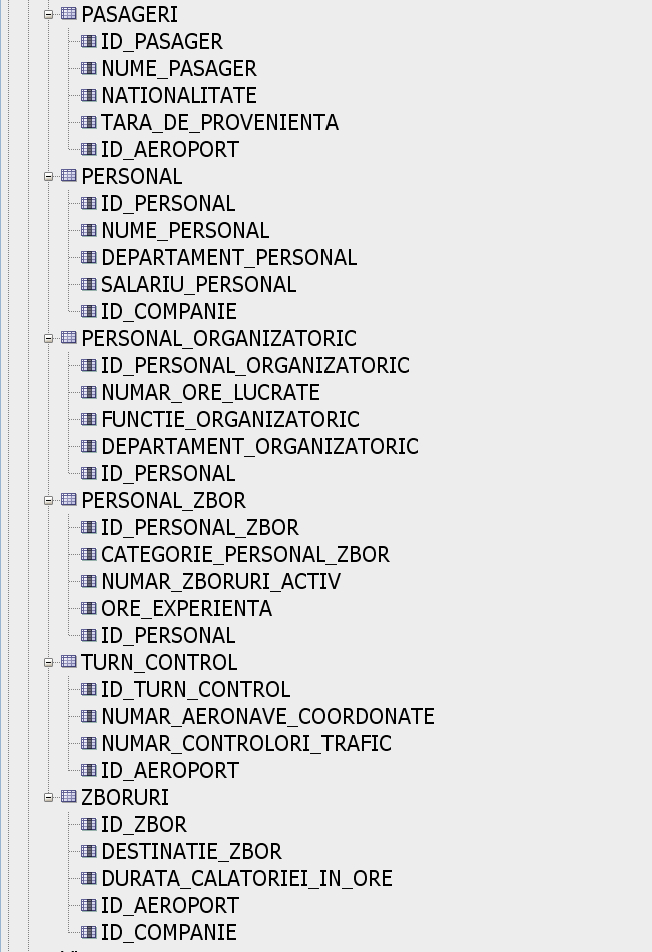
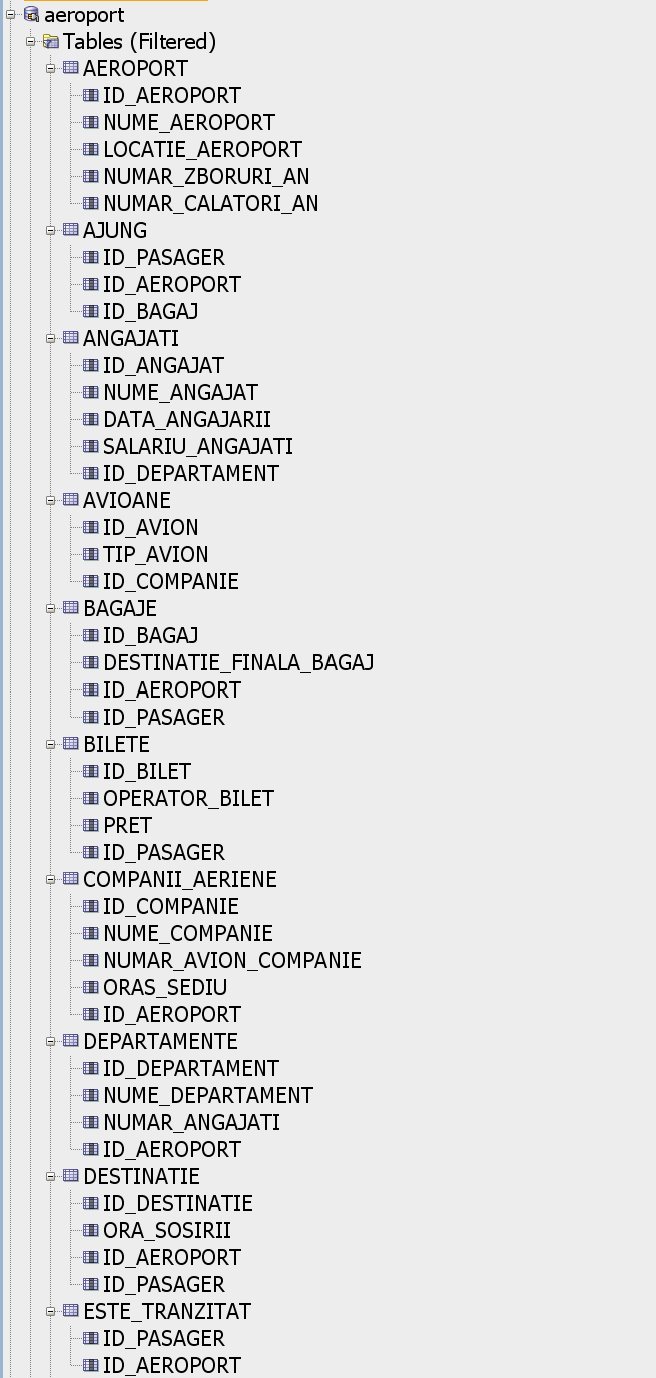
constraint pk\_ajung primary key(ID\_PASAGER, ID\_AEROPORT, ID\_BAGAJ)

);

--drop table AJUNG;







--INSERARE--

insert into AEROPORT values (1000, 'Henri Coanda', 'Bucuresti', 500, 20000);

insert into AEROPORT values (1100, 'Schipol', 'Amsterdam', 1000, 100500);

insert into AEROPORT values (1200, 'Heathrow', 'Londra', 1500, 300000);

insert into AEROPORT values (1300, 'Charles de Gaulle', 'Paris', 1100, 100000);

insert into AEROPORT values (1400, 'John F. Kennedy', 'New York', 2000, 900000);

insert into AEROPORT values (1500, 'Beijing', 'Beijing', 3000, 1000000);

select \* from AEROPORT;



insert into DEPARTAMENTE values (1001, 'HR' ,100, 1000);

insert into DEPARTAMENTE values (1002, 'PR' ,100, 1000);

insert into DEPARTAMENTE values (1003, 'Logistica', 50, 1100);

insert into DEPARTAMENTE values (1004, 'Comercial', 40, 1100);

insert into DEPARTAMENTE values (1005, 'Marketing', 25, 1200);

insert into DEPARTAMENTE values (1006, 'Control', 10, 1200);

insert into DEPARTAMENTE values (1007, 'Securitate', 5, 1300);

insert into DEPARTAMENTE values (1008, 'Operatiuni', 80, 1300);

insert into DEPARTAMENTE values (1009, 'Management', 45, 1400);

insert into DEPARTAMENTE values (1010, 'Achizitii', 30, 1400);

insert into DEPARTAMENTE values (1011, 'Finante', 100, 1500);

insert into DEPARTAMENTE values (1012, 'Dezvoltare', 30, 1500);

insert into DEPARTAMENTE values (1013, 'Director\_1000', 5, 1000);

insert into DEPARTAMENTE values (1014, 'Director\_1100', 80, 1100);

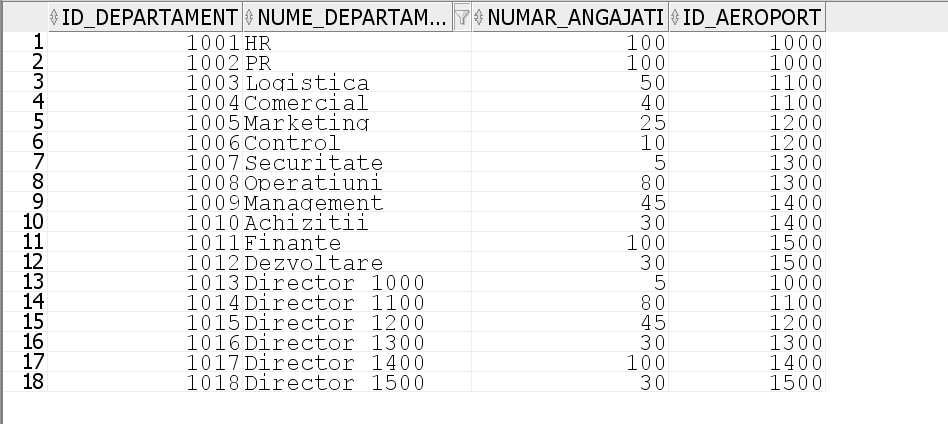
insert into DEPARTAMENTE values (1015, 'Director\_1200', 45, 1200);

insert into DEPARTAMENTE values (1016, 'Director\_1300', 30, 1300);

insert into DEPARTAMENTE values (1017, 'Director\_1400', 100, 1400);

insert into DEPARTAMENTE values (1018, 'Director\_1500', 30, 1500);

select \* from DEPARTAMENTE;



insert into ANGAJATI values (111, 'Popescu', '21-03-2010', 2000, 1001);

insert into ANGAJATI values (112, 'Marinescu', '18-09-2009', 5000, 1002);

insert into ANGAJATI values (113, 'Popa', '20-05-2011', 4500, 1003);

insert into ANGAJATI values (114, 'Spataru', '03-03-2010', 6300, 1004);

insert into ANGAJATI values (115, 'Nastase', '04-05-2003', 5200, 1005);

insert into ANGAJATI values (116, 'Afrim', '01-08-2013', 3900, 1006);

insert into ANGAJATI values (117, 'Alexa', '15-12-2015', 4800, 1007);

insert into ANGAJATI values (118, 'Ionita', '21-10-2012', 6700, 1008);

insert into ANGAJATI values (119, 'Bercu', '17-02-2005', 5400, 1009);

insert into ANGAJATI values (120, 'Nita', '18-08-2013', 5500, 1010);

insert into ANGAJATI values (121, 'Plesa', '27-06-2014', 5500, 1011);

insert into ANGAJATI values (122, 'Duta', '05-07-2004', 7200, 1012);

insert into ANGAJATI values (123, 'Hobbs', '03-03-2010', 8900, 1013);

insert into ANGAJATI values (124, 'Sargent', '21-09-2012', 9200, 1014);

insert into ANGAJATI values (125, 'Galloway', '02-02-2006', 10000, 1015);

insert into ANGAJATI values (126, 'Connolly', '04-08-2013', 9500, 1016);

insert into ANGAJATI values (127, 'Holcomb', '05-06-2014', 9800, 1017);

insert into ANGAJATI values (128, 'Sheehan', '08-09-2004', 8800, 1018);

select \* from ANGAJATI;



insert into TURN\_CONTROL values (900, 600, 30, 1000);

insert into TURN\_CONTROL values (910, 1500, 50, 1100);

insert into TURN\_CONTROL values (920, 2000, 100, 1200);

insert into TURN\_CONTROL values (930, 1900, 80, 1300);

insert into TURN\_CONTROL values (940, 2500, 150, 1400);

insert into TURN\_CONTROL values (950, 4000, 200, 1500);

select \* from TURN\_CONTROL;



insert into PASAGERI values (90, 'Popa', 'Romana', 'Germania', 1000);

insert into PASAGERI values (91, 'Phan', 'Chineza', 'Japonia', 1500);

insert into PASAGERI values (92, 'Edwards', 'Irlandeza', 'Argentina', 1200);

insert into PASAGERI values (93, 'Bass', 'Britanica', 'Egipt', 1100);

insert into PASAGERI values (94, 'Senior', 'Canadiana', 'Ungaria', 1300);

insert into PASAGERI values (95, 'Mcclain', 'Americana', 'Finlanda', 1400);

insert into PASAGERI values (100, 'King', 'Romana', 'Italia', 1000);

insert into PASAGERI values (101, 'Liu', 'Chineza', 'Australia', 1500);

insert into PASAGERI values (102, 'Lott', 'Britanica', 'SUA', 1200);

insert into PASAGERI values (103, 'Bishop', 'Canadiana', 'Africa de Sud', 1100);

insert into PASAGERI values (104, 'Medina', 'Italiana', 'Franta', 1300);

insert into PASAGERI values (105, 'Franks', 'Americana', 'Ucraina', 1400);

select \* from PASAGERI;



insert into BILETE values (500, 'TAROM', 100, 100);

insert into BILETE values (501, 'KLM', 300, 101);

insert into BILETE values (502, 'British Airways', 200, 102);

insert into BILETE values (503, 'AirChina', 500, 103);

insert into BILETE values (504, 'Delta', 700, 104);

insert into BILETE values (505, 'AirFrance', 400, 105);

insert into BILETE values (506, 'TAROM', 200, 90);

insert into BILETE values (507, 'KLM', 400, 91);

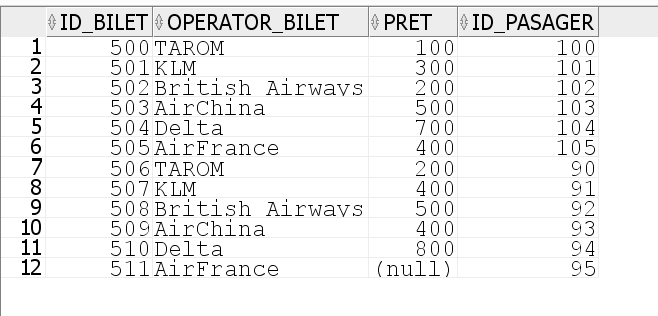
insert into BILETE values (508, 'British Airways', 500, 92);

insert into BILETE values (509, 'AirChina', 400, 93);

insert into BILETE values (510, 'Delta', 800, 94);

insert into BILETE values (511, 'AirFrance', null, 95);

select \* from BILETE;



insert into BAGAJE values (10, 'Bucuresti', 1000, 100);

insert into BAGAJE values (11, 'Beijing', 1500, 101);

insert into BAGAJE values (12, 'Londra', 1200, 102);

insert into BAGAJE values (13, 'Amsterdam', 1100, 103);

insert into BAGAJE values (14, 'Paris', 1300, 104);

insert into BAGAJE values (15, 'New York', 1400, 105);

insert into BAGAJE values (16, 'Bucuresti', 1000, 90);

insert into BAGAJE values (17, 'Beijing', 1500, 91);

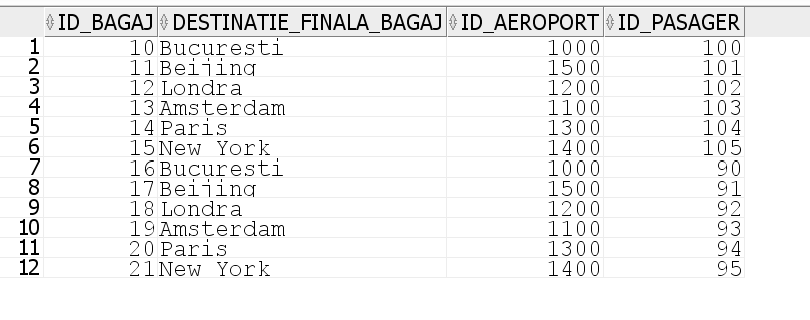
insert into BAGAJE values (18, 'Londra', 1200, 92);

insert into BAGAJE values (19, 'Amsterdam', 1100, 93);

insert into BAGAJE values (20, 'Paris', 1300, 94);

insert into BAGAJE values (21, 'New York', 1400, 95);

select \* from BAGAJE;



insert into DESTINATIE values (3000, 20, 1000, 100);

insert into DESTINATIE values (3100, 15, 1500, 101);

insert into DESTINATIE values (3200, 01, 1200, 102);

insert into DESTINATIE values (3300, 10, 1100, 103);

insert into DESTINATIE values (3400, 19, 1300, 104);

insert into DESTINATIE values (3500, 12, 1400, 105);

insert into DESTINATIE values (3001, 07, 1000, 90);

insert into DESTINATIE values (3101, 15, 1500, 91);

insert into DESTINATIE values (3201, 23, 1200, 92);

insert into DESTINATIE values (3301, 02, 1100, 93);

insert into DESTINATIE values (3401, 21, 1300, 94);

insert into DESTINATIE values (3501, 11, 1400, 95);

select \* from DESTINATIE;



insert into COMPANII\_AERIENE values (800, 'TAROM', 50, 'Bucuresti', 1000);

insert into COMPANII\_AERIENE values (810, 'KLM', 300, 'Amsterdam', 1100);

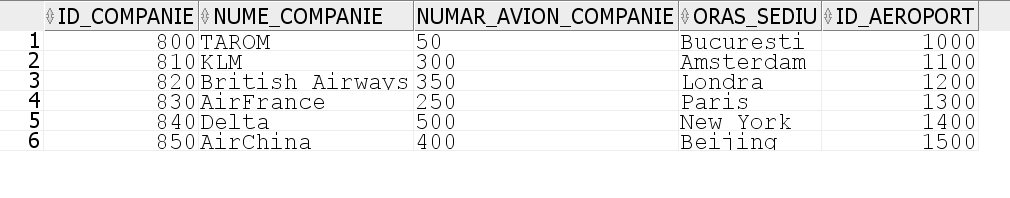
insert into COMPANII\_AERIENE values (820, 'British Airways', 350, 'Londra', 1200);

insert into COMPANII\_AERIENE values (830, 'AirFrance', 250, 'Paris', 1300);

insert into COMPANII\_AERIENE values (840, 'Delta', 500, 'New York', 1400);

insert into COMPANII\_AERIENE values (850, 'AirChina', 400, 'Beijing', 1500);

select \* from COMPANII\_AERIENE;



insert into AVIOANE values (50, 'Boeing 707', 800);

insert into AVIOANE values (51, 'Boeing 787', 800);

insert into AVIOANE values (52, 'Airbus A300', 810);

insert into AVIOANE values (53, 'Airbus A340', 810);

insert into AVIOANE values (54, 'Airbus A310', 820);

insert into AVIOANE values (55, 'Airbus A330', 820);

insert into AVIOANE values (56, 'Boeing 717', 830);

insert into AVIOANE values (57, 'ATR 42', 830);

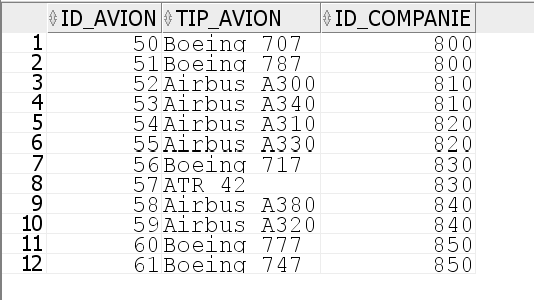
insert into AVIOANE values (58, 'Airbus A380', 840);

insert into AVIOANE values (59, 'Airbus A320', 840);

insert into AVIOANE values (60, 'Boeing 777', 850);

insert into AVIOANE values (61, 'Boeing 747', 850);

select \* from AVIOANE;



insert into ZBORURI values (5000, 'Bucuresti', 2, 1000, 800);

insert into ZBORURI values (5100, 'Amsterdam', 3, 1100, 810);

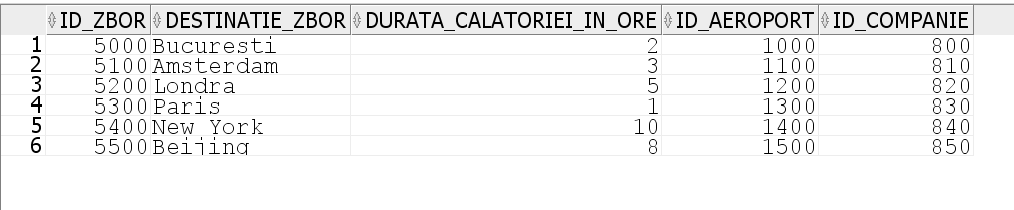
insert into ZBORURI values (5200, 'Londra', 5, 1200, 820);

insert into ZBORURI values (5300, 'Paris', 1, 1300, 830);

insert into ZBORURI values (5400, 'New York', 10, 1400, 840);

insert into ZBORURI values (5500, 'Beijing', 8, 1500, 850);

select \* from ZBORURI;



insert into PERSONAL values (300, 'Turner', 'Conducere\_800', 8000, 800);

insert into PERSONAL values (301, 'Pruitt', 'Conducere\_810', 9000, 810);

insert into PERSONAL values (302, 'Stamp', 'Conducere\_820', 7000, 820);

insert into PERSONAL values (303, 'Devine', 'Conducere\_830', 10000, 830);

insert into PERSONAL values (304, 'Hudson', 'Conducere\_840', 6000, 840);

insert into PERSONAL values (305, 'Oritz', 'Conducere\_850', 12000, 850);

insert into PERSONAL values (306, 'Parsons', 'Zbor', 10000, 800);

insert into PERSONAL values (307, 'Rankin', 'Zbor', 5000, 810);

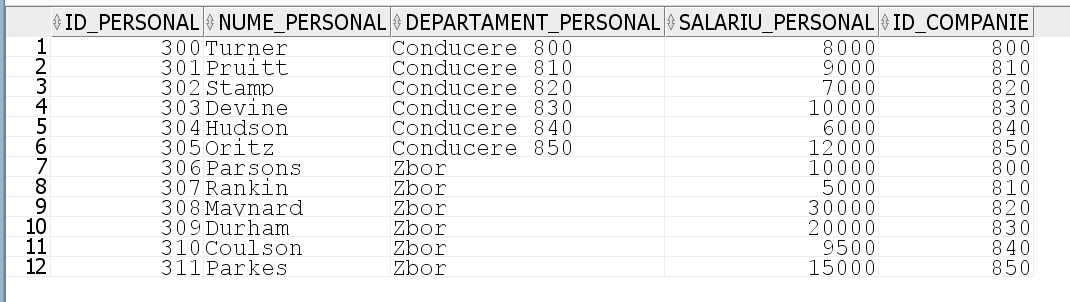
insert into PERSONAL values (308, 'Maynard', 'Zbor', 30000, 820);

insert into PERSONAL values (309, 'Durham', 'Zbor', 20000, 830);

insert into PERSONAL values (310, 'Coulson', 'Zbor', 9500, 840);

insert into PERSONAL values (311, 'Parkes', 'Zbor', 15000, 850);

select \* from PERSONAL;



insert into PERSONAL\_ORGANIZATORIC values (320, 100, 'Manager', 'HR', 300);

insert into PERSONAL\_ORGANIZATORIC values (321, 150, 'Manager', 'PR', 301);

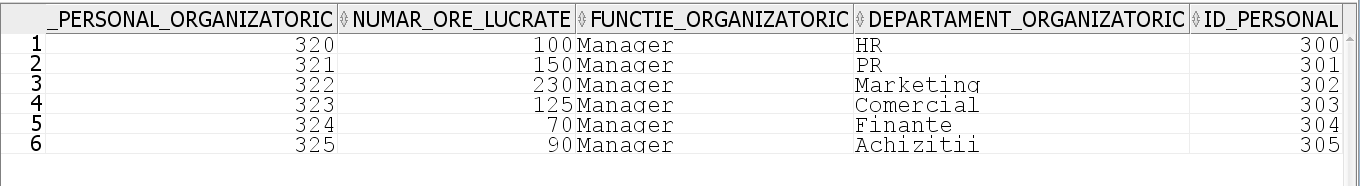
insert into PERSONAL\_ORGANIZATORIC values (322, 230, 'Manager', 'Marketing', 302);

insert into PERSONAL\_ORGANIZATORIC values (323, 125, 'Manager', 'Comercial', 303);

insert into PERSONAL\_ORGANIZATORIC values (324, 70, 'Manager', 'Finante', 304);

insert into PERSONAL\_ORGANIZATORIC values (325, 90, 'Manager', 'Achizitii', 305);

select \* from PERSONAL\_ORGANIZATORIC;



insert into PERSONAL\_ZBOR values (326, 'Pilot', 200, 1000, 306);

insert into PERSONAL\_ZBOR values (327, 'Pilot', 200, 1200, 307);

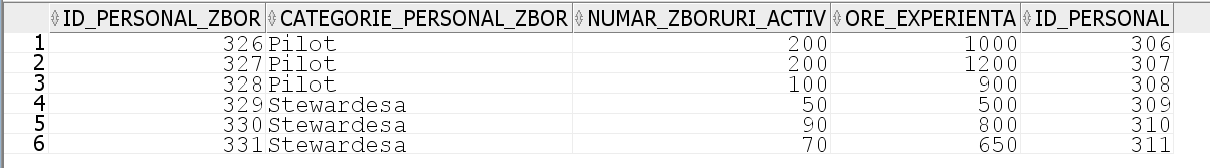
insert into PERSONAL\_ZBOR values (328, 'Pilot', 100, 900, 308);

insert into PERSONAL\_ZBOR values (329, 'Stewardesa', 50, 500, 309);

insert into PERSONAL\_ZBOR values (330, 'Stewardesa', 90, 800, 310);

insert into PERSONAL\_ZBOR values (331, 'Stewardesa', 70, 650, 311);

select \* from PERSONAL\_ZBOR;



insert into AJUNG values (1000, 10, 100);

insert into AJUNG values (1100, 11, 101);

insert into AJUNG values (1200, 12, 102);

insert into AJUNG values (1300, 13, 103);

insert into AJUNG values (1400, 14, 104);

insert into AJUNG values (1500, 15, 105);

insert into AJUNG values (1000, 16, 90);

insert into AJUNG values (1100, 17, 91);

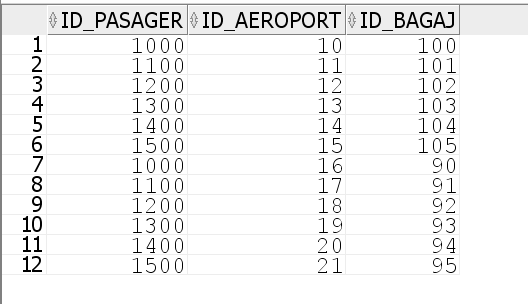
insert into AJUNG values (1200, 18, 92);

insert into AJUNG values (1300, 19, 93);

insert into AJUNG values (1400, 20, 94);

insert into AJUNG values (1500, 21, 95);

select \* from AJUNG;



insert into ESTE\_TRANZITAT values (1000, 10);

insert into ESTE\_TRANZITAT values (1100, 11);

insert into ESTE\_TRANZITAT values (1200, 12);

insert into ESTE\_TRANZITAT values (1300, 13);

insert into ESTE\_TRANZITAT values (1400, 14);

insert into ESTE\_TRANZITAT values (1500, 15);

insert into ESTE\_TRANZITAT values (1000, 16);

insert into ESTE\_TRANZITAT values (1100, 17);

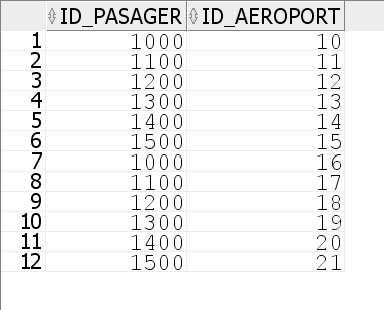
insert into ESTE\_TRANZITAT values (1200, 18);

insert into ESTE\_TRANZITAT values (1300, 19);

insert into ESTE\_TRANZITAT values (1400, 20);

insert into ESTE\_TRANZITAT values (1500, 21);

select \* from ESTE\_TRANZITAT;



1. Formulați în limbaj natural și implementați 5 cereri SQL complexe ce vor utiliza, în ansamblul lor, următoarele elemente:

• operație *join* pe cel puțin 4 tabele -

* + 1. • filtrare la nivel de linii -
    2. • subcereri sincronizate în care intervin cel puțin 3 tabele
    3. • subcereri nesincronizate în care intervin cel puțin 3 tabele
    4. • grupări de date, funcții grup, filtrare la nivel de grupuri
    5. • ordonări -
    6. • utilizarea a cel puțin 2 funcții pe șiruri de caractere, -
    7. 2 funcții pe date calendaristice,
    8. a funcțiilor NVL și DECODE,
    9. a cel puțin unei expresii CASE
    10. • utilizarea a cel puțin 1 bloc de cerere (clauza WITH)

**11.1 Exercitiul 1**

Sa se selecteze id-ul biletelor si numele pasagerilor, cu majuscule si ordonati alfabetic, ale caror bagaje ajung pe aeroportul unde opereaza compania aeriana 'Klm'.

select upper(p.nume\_pasager) as Nume, bil.id\_bilet

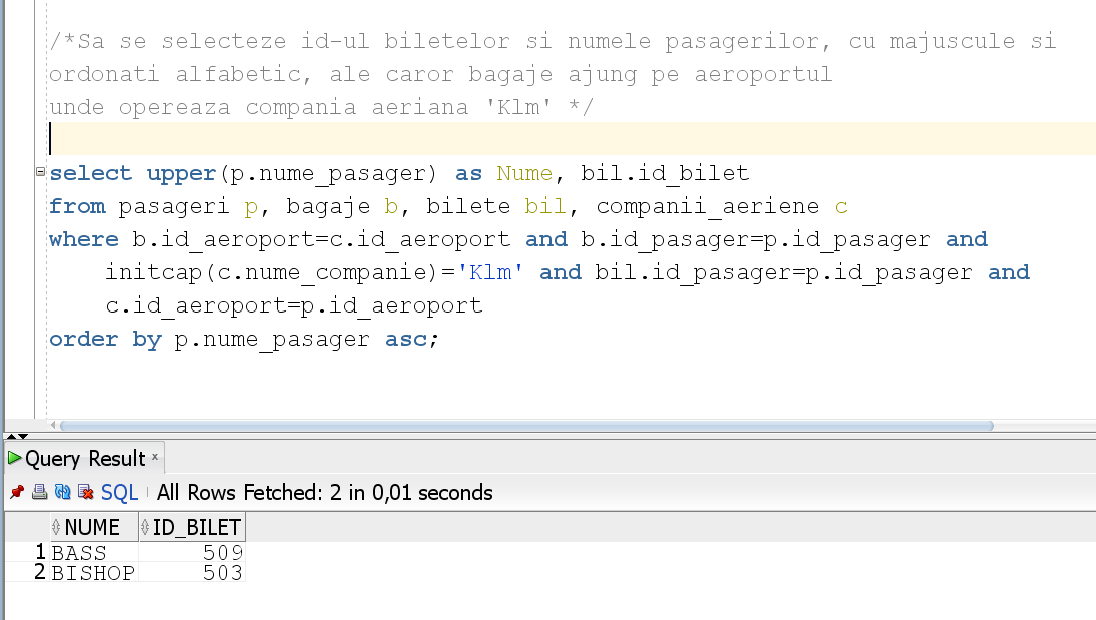
from pasageri p, bagaje b, bilete bil, companii\_aeriene c

where b.id\_aeroport=c.id\_aeroport and b.id\_pasager=p.id\_pasager and

initcap(c.nume\_companie)='Klm' and bil.id\_pasager=p.id\_pasager and

c.id\_aeroport=p.id\_aeroport

order by p.nume\_pasager asc;



**11.2 Exercitiul 2**

Sa se realizeze un bloc de cerere denumit "situatie" in care se vor selecta numele pasagerului, id-ul, pretul si operatorul biletului pentru care pretul sau este mai mare decat pretul mediu al tuturor biletelor ai caror operatori sunt companiile al caror sediu se afla in orasul in care id-ul aeroportului pe care ajung bagajele este 1100. Pasagerilor care indeplinesc aceasta conditie si au zborul operat de 'KLM' li se va aplica o reducere de 10%, noua coloanal fiind denumita NOUL\_PRET.

with situatie as(

select p.nume\_pasager, bil.id\_bilet, bil.pret, bil.operator\_bilet

from bilete bil , pasageri p

where bil.id\_pasager= p.id\_pasager and bil.pret>(select round(avg(pret))

from bilete

where operator\_bilet=( select nume\_companie

from companii\_aeriene

where oras\_sediu in ( select destinatie\_finala\_bagaj

from bagaje

where id\_aeroport=1100

)

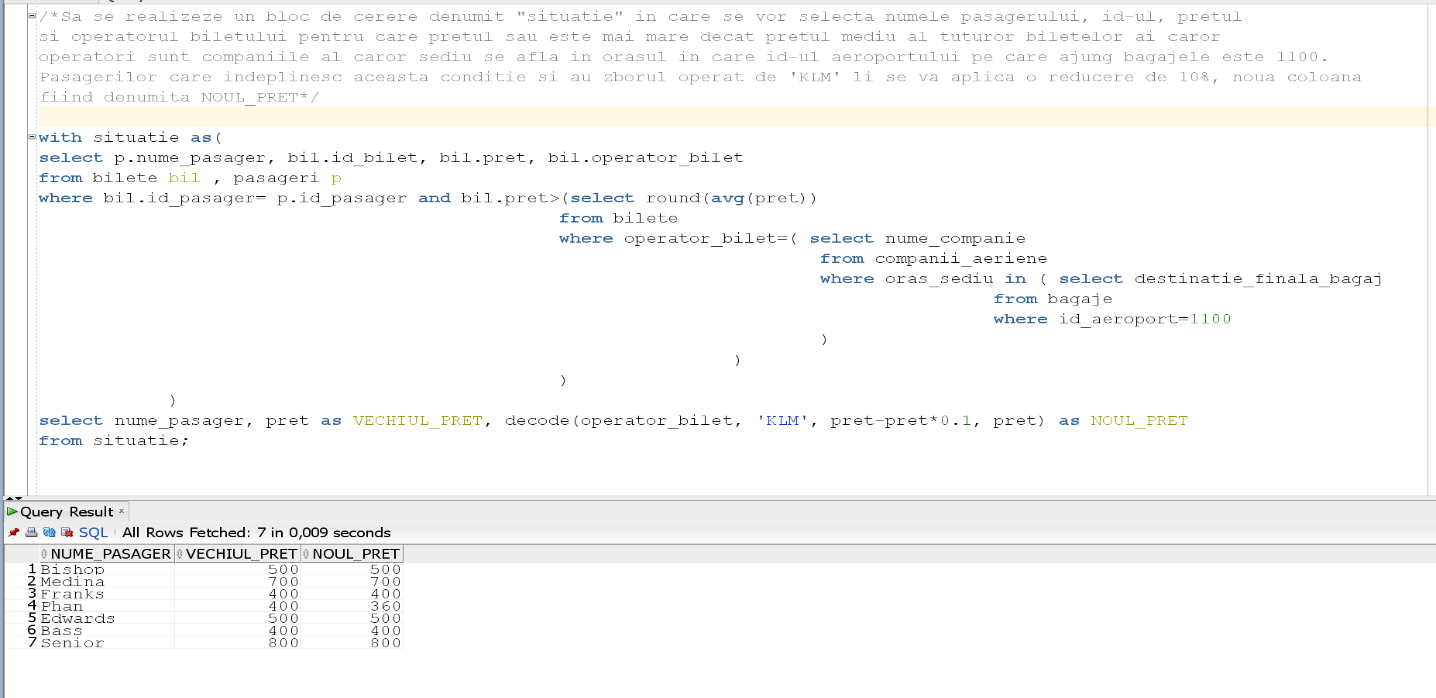
)

)

)

select nume\_pasager, pret as VECHIUL\_PRET, decode(operator\_bilet, 'KLM', pret-pret\*0.1, pret) as NOUL\_PRET

from situatie;



**11.3 Exercitiul 3**

Sa se selecteze numele angajatului, salariul sau vechi cu numele de VECHIUL\_SALARIU si noul lor salariu ca NOUL\_SALARIU, stiind ca cei care au salariul mai mic de 700 vor primi o marire de 20%. Acest lucru se aplica numai angajatilor care lucreaza de mai mult de 3000 de zile in departamentele lor, iar ultima zi a lunii in care au fost angajati este 31.03.2010. Sa se ordoneze alfabetic rezultatele obtinute.

select nume\_angajat, salariu\_angajati as VECHIUL\_SALARIU,

case

when salariu\_angajati <7000 then

salariu\_angajati+salariu\_angajati\*0.2

else salariu\_angajati

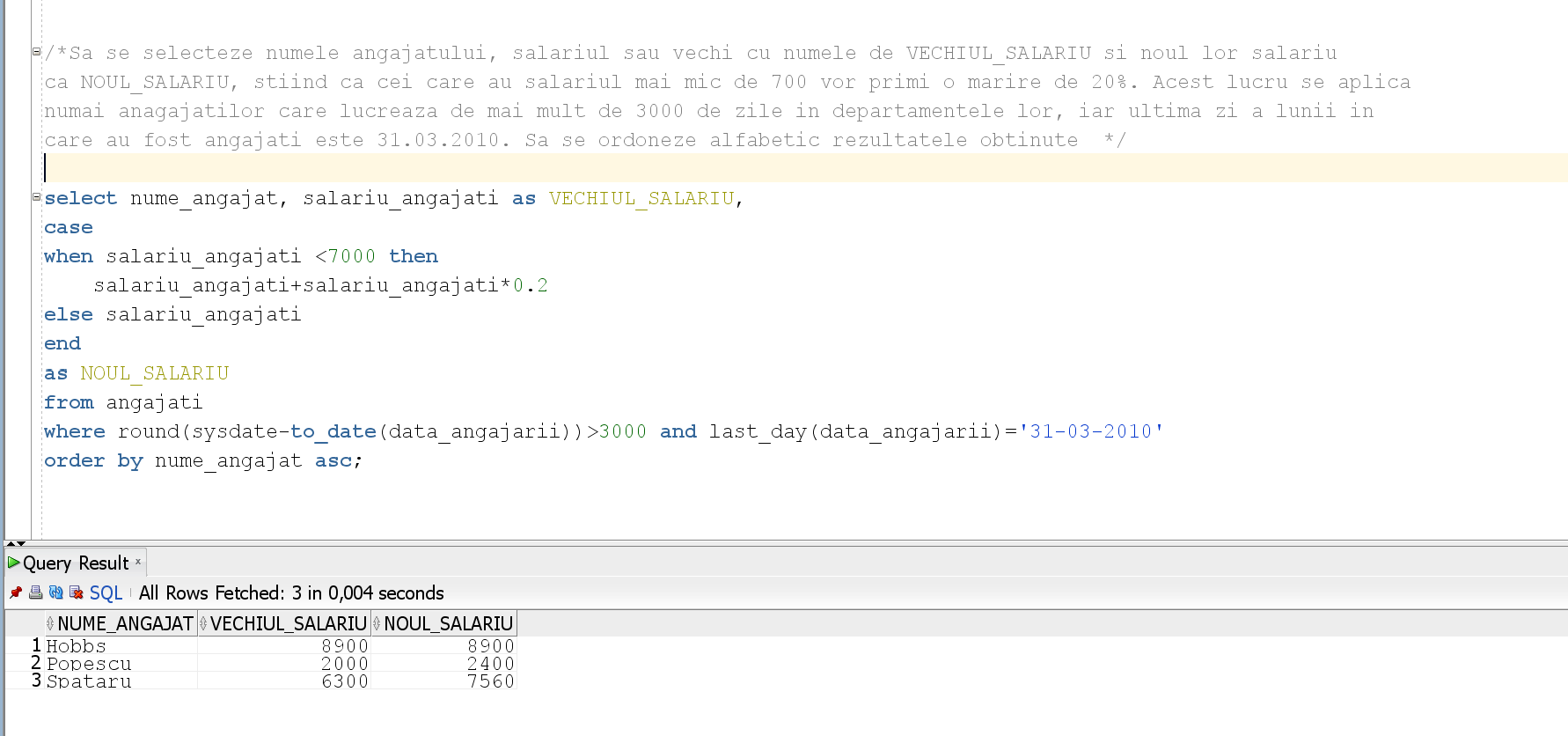
end

as NOUL\_SALARIU

from angajati

where round(sysdate-to\_date(data\_angajarii))>3000 and last\_day(data\_angajarii)='31-03-2010'

order by nume\_angajat asc;

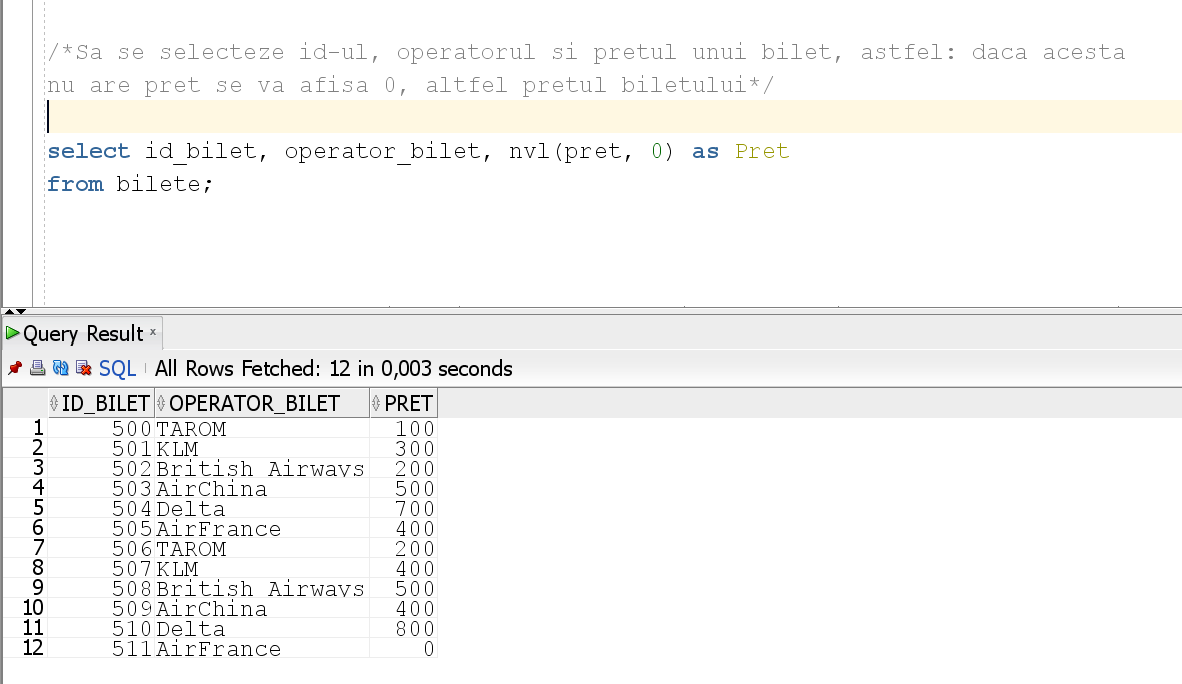


**11.4 Exercitiul 4**

Sa se selecteze id-ul, operatorul si pretul unui bilet, astfel: daca acesta nu are pret se va afisa 0, altfel pretul biletului.

select id\_bilet, operator\_bilet, nvl(pret, 0) as Pret

from bilete;



**11.5 Exercitiul 5**

Sa se se selecteze nationalitatea care contine 'ana' si numarul de pasageri cu acea nationalitate pentru pasagerii care tranziteaza aeroporturile ale caror turnuri de control au id-rile 900, respectiv 910. Ulterior, se va selecta numarul maxim de pasageri din rezultatele obtinute.

select \* from pasageri;

select max(numar)as Maxim

from(

select nationalitate, count(\*) as numar

from pasageri p

where p.id\_aeroport in (

select a.id\_aeroport

from aeroport a

where p.id\_aeroport=a.id\_aeroport and

a.id\_aeroport in (

select t.id\_aeroport

from turn\_control t

where t.id\_aeroport=a.id\_aeroport

and t.id\_turn\_control=910

or t.id\_turn\_control=900

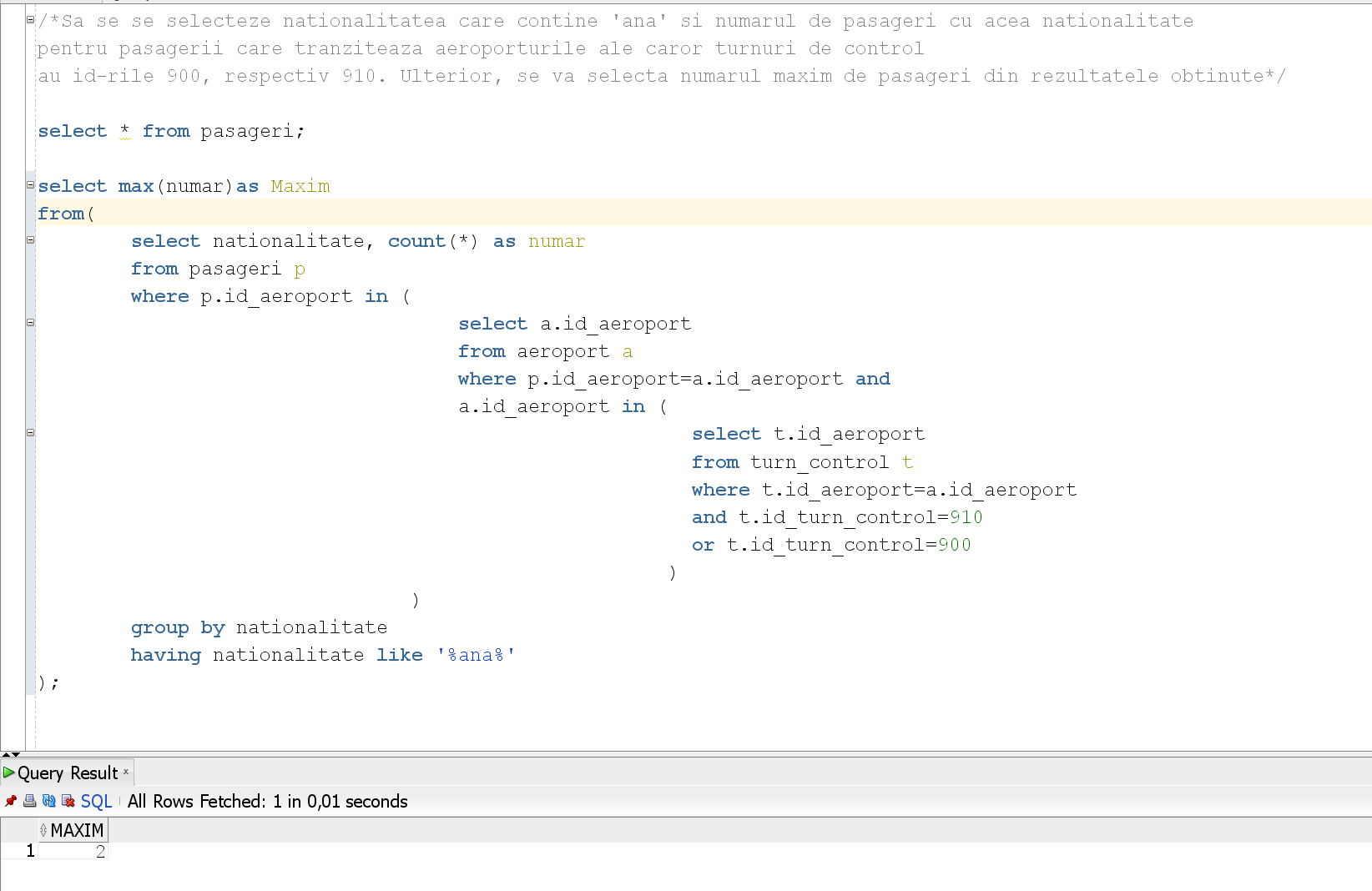
)

)

group by nationalitate

having nationalitate like '%ana%'

);



1. Implementarea a 3 operații de actualizare și de suprimare a datelor utilizând subcereri.

/\*Sa se stearga biletele de la operatorul care opereaza pe aeroportul cu id-ul 1200.\*/

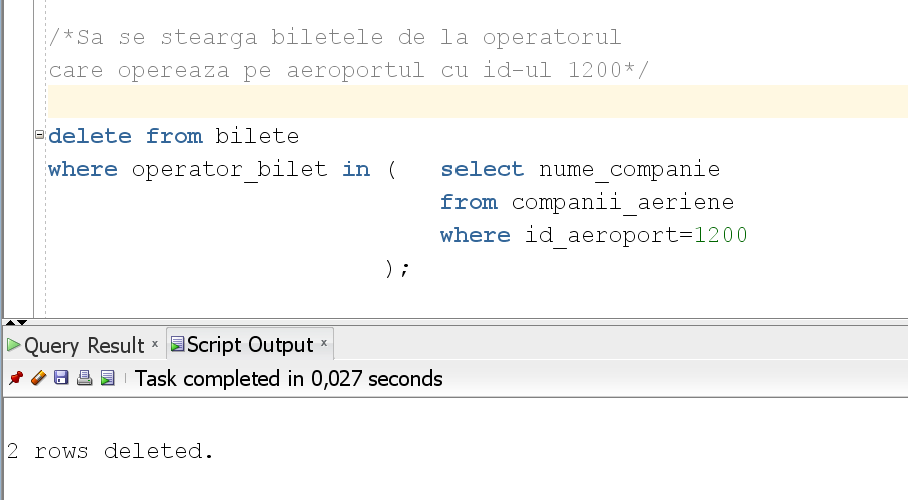
delete from bilete

where operator\_bilet in ( select nume\_companie

from companii\_aeriene

where id\_aeroport=1200

);



/\*Sa se modifice salariul angajatilor din tabela de angajati care au salariul mai mare de 5000 cu media salariilor din tabela de personal.\*/

update angajati

set salariu\_angajati=( select round(avg(salariu\_personal))

from personal)

where salariu\_angajati > 5000;



/\*Sa se stearga bagajele paserilor care contin ‘a’ in nume.\*/

delete from bagaje

where id\_pasager in ( select id\_pasager

from pasageri

where nume\_pasager like '%a%'

);

