**Introducere în Business Analytics**

**Proiect 1 – Proc SQL**

**Business Analytics și Managementul Informațiilor**

**Popoviciu Andreea**

**Anul I, 2023-2024**

**Descrierea date**

Mai jos este structura datelor pentru fiecare tabel folosit în cadrul proiectului. Structura conține numele coloanelor, descrierea și cerințe specifice pentru valoarile coloanelor.

**A screenshot of a computer

Description automatically generated**

**A screenshot of a computer screen

Description automatically generated**

**A screenshot of a computer

Description automatically generated**

**A screenshot of a computer screen

Description automatically generated**

**A screenshot of a computer

Description automatically generated**

**A screenshot of a computer

Description automatically generated**

**Cerințe**

**Lecția1.** Afișați cel mult 15 înregistrări din fiecare tabelă, cu menționarea numărului înregistrării. La o tabelă afișați toate coloanele, iar la celelalte două selectiv. Să se afișeze structura unei tabele;

1. Să se afișeze conținutul tabelului boarding2013\_2016.

proc sql inobs=15 number;

select \* from p1.boarding2013\_2016;

quit;

A screenshot of a computer

Description automatically generated

1. În cadrul tabelei claimsraw să se afișeze informațiile legate de Claim\_Number, Date\_Received, Incident\_Date, Claim\_Type și Claim\_Site.

proc sql inobs=15 number;

select Claim\_Number, Date\_Received, Incident\_Date, Claim\_Type, Claim\_Site

from p1.claimsraw;

quit;

A screenshot of a document

Description automatically generated

1. În cadrul tabelei enplanement2017 să se afișeze informațiile legate de LocID și Enplanement.

proc sql inobs=15 number;

select LocID, Enplanement

from p1.enplanement2017;

quit;

A screenshot of a computer screen

Description automatically generated

1. Să se afișeze structura tabelei claimsraw.

proc sql;

describe table p1.claimsraw;

A screenshot of a computer code

Description automatically generatedquit;

**Lecția 2.** În interogări distincte valorificați câte 3-4 clauze din comanda SELECT, astfel încât să nu rămână vreuna dintre clauze nefolosită. Afișați, fără duplicate, împreună cu coloanele și diverse expresii de calcul aritmetic , dar și bazate pe formule de calcul (obligatoriu una de extragere a unor elemente dintr-o dată calendaristică), precum și funcții de agregare. Să aveți un exemplu și cu CASE. Puneți condiții compuse în care să folosiți operatorii IN, NOT IN, IS MISSING, BETWEEN, LIKE. Îmbogățiți raportarea cu titluri, antete, subsoluri, etichete, formatări. Creați o tabelă nouă în care să adăugați date prin 2 metode distincte de utilizare a INSERT-ului. Faceți pe noua tabelă 2 actualizări și o ștergere;

1. Să se afișeze primele 15 reclamații distincte primite din partea pasagerilor înregistrate în lunile noiembrie și decembrie ordonate descrescător în funcție de data la care au fost înregistrate.

title1 "Reclamații primite";

footnote1 "Noiembrie-Decembrie";

proc sql outobs=15 number;

select distinct Claim\_Number label="Claim Number",

Date\_Received label="Date Received",

Incident\_Date label="Incident Date",

Airport\_Code label="Airport Code",

Claim\_Type label="Claim Type",

Claim\_Site label="Claim Site",

Close\_Amount label="Close Amount" format=DOLLAR16.2,

Disposition

from p1.claimsraw

where month(Incident\_Date) IN (11,12) AND Close\_Amount IS NOT MISSING

order by Incident\_Date DESC;

quit;

title;

A screenshot of a computer

Description automatically generatedfootnote;

1. În cadrul tabelei boarding2013\_2016 să se afișeze numărul de îmbarcări din fiecare aeroport pentru toți ani înafară de 2013 și 2014.

title1 "Îmbarcări 2015-2016";

proc sql number;

select LocID, Year, Boarding format=COMMA15.

from p1.boarding2013\_2016

where Year NOT IN (2013, 2014)

group by Year

order by LocID;

quit;

title;

A screenshot of a computer

Description automatically generated

1. Să se calculeze și afișeze numărul de reclamați primite pentru fiecare aeroport alături de suma minimă, maximă și media pentru despăgubirile acordate pasagerilor în vederea soluționări reclamațiilor.

title1 "Reclamați primite pe aeroport";

title2 "Suma minimă, maximă și media despăgubirilor acordare";

proc sql number;

select distinct Airport\_Code, Airport\_Name,

count(Airport\_Code) AS Number\_of\_claims,

avg(Close\_Amount) AS Avg\_Close\_Amount format=DOLLAR16.2,

min(Close\_Amount) AS Min\_Close\_Amount format=DOLLAR16.2,

max(Close\_Amount) AS Max\_Close\_Amount format=DOLLAR16.2

from p1.claimsraw

where Close\_Amount IS NOT NULL

group by Airport\_Code

having calculated Number\_of\_claims > 0

order by Airport\_Code;

quit;

title2;

title1;

A screenshot of a computer screen

Description automatically generated

1. Să se afișeze primele 300 de reclamați primite din partea pasagerilor care nu au fost respinse și pentru care există valori înregistrate privind valoarea despăgubirilor acordate. Să se calculeze numărul de zile necesare companiei pentru soluționarea reclamațiilor și să se idetifice calitatea serviciilor în funcție de numărul de zile necesare pentru soluționare.

title1 "Reclamații soluționate";

proc sql outobs=300 number;

select Claim\_Number, Date\_Received, Incident\_Date, Airport\_Code,

Claim\_Type, Claim\_Site, Close\_Amount, Disposition,

datdif(Incident\_Date, Date\_Received, '30/360') as Days,

case

when calculated Days <= 5 then "Excellent"

when calculated Days <= 10 then "Good"

when calculated Days <= 30 then "Medium"

when calculated Days >= 31 then "Bad"

else "Unknown"

end as Quality\_of\_Service

from p1.claimsraw

where Disposition NOT IN ("Deny") AND Close\_Amount IS NOT MISSING

group by Airport\_Code

order by Incident\_Date DESC;

quit;

title;

A screenshot of a computer

Description automatically generated

1. În cadrul tabelei enplanement2017 să se afișeze aeroporturile pentru care a existat un număr cuprins între 10.000.00 și 20.000.000 de pasageri îmbarcați în aeronave.

title "Aeroporturi cu 10.000.000-20.000.000 de îmbarcări în anul 2017";

proc sql outobs=100 number;

select \*

from p1.enplanement2017

where Enplanement BETWEEN 10000000 AND 20000000

order by LocID;

quit;

title;

A screenshot of a computer screen

Description automatically generated

1. În cadrul tabelei boarding2013\_2016 să se afișeze aeroporturile a căror denumire începe cu litera „A”.

title "Aeroporturile a căror denumire începe cu litera A";

proc sql number;

select \*

from p1.enplanement2017

where LocID LIKE "A%"

group by LocID;

quit;

title;

A screenshot of a computer

Description automatically generated

1. Creați o tabelă în care se va stoca informați cu privire la rezervările pasagerilor. Adăugați date folosind 2 metoded diferite ale INSERT-ului. Efectuați două actualizări și o ștergere în cadrul tabelului.

proc sql;

create table p1.reservations

(ID num(5),

Flight\_Code char(8),

LocID char(14) format=$3.,

Passenger\_Full\_Name char(50),

Reservation\_Entry\_Date num format=DATETIME19.,

Reserved\_Date num format=DATETIME19.,

State char(25),

City char(25),

Class char(25),

Status char(20)

);

insert into p1.reservations

(ID, Flight\_Code, LocID, Passenger\_Full\_Name, Reservation\_Entry\_Date, Reserved\_Date, State, City, Class, Status)

values (12345, "11223344", "JFK", "John Smith", "01Dec2017:01:25:11"dt, "05Dec2017:08:15:00"dt, "NEW YORK", "NEW YORK", "Business", "In Process")

values (11000, "00112233", "SAN", "Sarah Davis", "20Nov2017:05:22:50"dt, "05Dec2017:07:30:00"dt, "WASHINGTON", "SEATTLE", "Economy","Checked-In");

insert into p1.reservations

set ID = 2222,

Flight\_Code = "22334411",

LocID = "DEN",

Passenger\_Full\_Name = "Emily Brown",

Reservation\_Entry\_Date = "25Nov2017:15:22:45"dt,

Reserved\_Date = "02Dec2017:10:00:00"dt,

State = "CALIFORNIA",

City = "SAN DIEGO",

Class = "Business",

Status = "Checked-In";

quit;

A screenshot of a computer

Description automatically generated

proc sql;

update p1.reservations

set Status = "Confirmed" where ID = 12345;

alter table p1.reservations

modify ID num format=z5.;

delete from p1.reservations

where ID = 11000;

quit;

A screenshot of a computer code

Description automatically generated

**Lecția 3.** Realizați interogări în care să folosiți JOIN-uri pe baza a 2-3 tabele. Să aveți: INNER JOIN, join natural, produs cartezian, cele 3 tipuri de join extern, un autojoin (self join) + o interogare în care să returnați DOAR înregistrările pentru care nu există potriviri (nonmatching rows) față de cealaltă tabelă;

1. Folosind INNER JOIN selectați informațiile despre fiecare aeroport din tabelele enplanement2017 și claimsraw.

proc sql outobs=200 number;

select distinct LocID, c.Airport\_Name, c.StateName, c.City, Year, Enplanement

from p1.enplanement2017 as e inner join p1.claimsraw as c

on e.LocID = c.Airport\_Code

group by e.LocID;

quit;

A screenshot of a computer

Description automatically generated

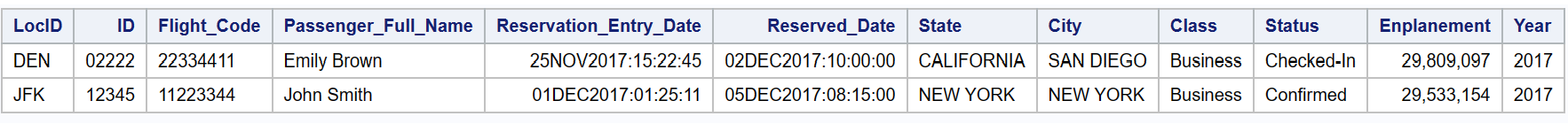
1. Folosind NATURAL JOIN selectați aeroporturile și pasageri care au rezervat locuri pentru zboruri.

proc sql outobs=100 feedback;

select \*

from p1.enplanement2017 as e NATURAL JOIN p1.reservations as r;

quit;



1. Generați Produsul Cartezian între tabelele boarding2013\_2016 și enplanement2017.

proc sql inobs=10 number;

select \* from p1.boarding2013\_2016, p1.enplanement2017;

quit;

A screenshot of a computer screen

Description automatically generated

1. Folosind LEFT JOIN pentru tabelele boarding2013\_2016 si claimsraw selectați ID-ul locațiilor și suma maximă acordată pentru soluționarea reclamațiilor în locația respectivă.

proc sql outobs=200 number;

select distinct LocID, Max(c.Close\_Amount) as Max\_Claim

from p1.boarding2013\_2016 as b left join p1.claimsraw as c

on b.LocID = c.Airport\_Code and LocID IS NOT NULL

group by c.Airport\_Code;

quit;

A screenshot of a computer

Description automatically generated

1. Folosind RIGHT JOIN pentru tabelele boarding2013\_2016 si claimsraw selectați sumele maxime acordate pentru soluționarea reclamaților alături de aeroportul corespunzător.

proc sql outobs=200 number;

select distinct LocID, Max(c.Close\_Amount) as Max\_Claim

from p1.boarding2013\_2016 as b right join p1.claimsraw as c

on b.LocID = c.Airport\_Code and LocID IS NOT NULL

group by c.Airport\_Code;

quit;

A screenshot of a computer

Description automatically generated

1. Folosind FULL JOIN pentru tabelele boarding2013\_2016, claimsraw si enplanement2017 selectați codul, numele, statul și numărul total de reclamați primite în fiecare an alături de valorile corespunzătoare pentru Boarding și Enplanement.

proc sql outobs=200 number;

select distinct c.Airport\_Code,

c.Airport\_Name, c.StateName,

count(Airport\_Code) AS Number\_of\_claims,

b.Boarding format=COMMA15.,

e.Enplanement,

year(c.Incident\_Date) as Year

from p1.claimsraw as c full join p1.boarding2013\_2016 as b

on c.Airport\_Code = b.LocID and year(c.Incident\_Date) = b.Year full join p1.enplanement2017 as e

on c.Airport\_Code = e.LocID and year(c.Incident\_Date) = 2017

where c.Airport\_Code IS NOT NULL

group by c.Airport\_Code, calculated Year

order by c.Airport\_Code, Year;

quit;

1. A screenshot of a computer

   Description automatically generatedFolosind Autojoin selectați reclamațiile care au fost primite și soluționate în ziua în care a avut loc incidentul.

proc sql outobs= 200 number;

select c.Claim\_Number, c.Date\_Received, c.Incident\_Date,

c.Airport\_Code, c.Claim\_Type, c.Close\_Amount, c.Disposition

from p1.claimsraw as c inner join p1.claimsraw as cl

on c.Date\_Received = cl.Incident\_Date and c.Claim\_Number = cl.Claim\_Number

where c.Close\_Amount is not null;

A screenshot of a computer

Description automatically generatedquit;

1. Selectați înregistrările pentru care nu există potriviri în tabelele boarding2013\_2016 și enplanement2017.

proc sql number;

select \*

from p1.enplanement2017 as e left join p1.boarding2013\_2016 as b

on e.LocID = b.LocID

where b.LocID is null;

quit;

A screenshot of a table

Description automatically generated

**Lecția 4.** Realizați interogări care să conțină subcereri pentru fiecare clauză. Realizați o vedere pe care să o utilizați într-o operațiune de join cu una dintre tabelele de bază;

1. Selectati reclamațiile pentru care a fost acordată o sumă de bani mai mare decât media în vederea soluționări.

proc sql outobs=200 number;

select Claim\_Number, Date\_Received, Incident\_Date, Airport\_Code,

Claim\_Type, Claim\_Site, Close\_Amount, Disposition, StateName

from p1.claimsraw

where Close\_Amount > (select avg(Close\_Amount)

from p1.claimsraw

) and StateName is not null

order by Close\_Amount desc;

quit;

A screenshot of a computer

Description automatically generated

1. Selectați reclamațiile apărute în aeroporturi pentru care nu există informați înregistrate în tabela boarding2013\_2016.

proc sql number;

select Claim\_Number, Date\_Received, Incident\_Date, Airport\_Code,

Claim\_Type, Claim\_Site, Close\_Amount, Disposition, StateName

from p1.claimsraw

where Airport\_Code not in (select LocID from p1.boarding2013\_2016) and

Airport\_Code is not null and

Close\_Amount is not null and

StateName is not null;

A screenshot of a computer

Description automatically generatedquit;

1. Selectați aeroporturile pentru care media numărului de boarding din cei 4 ani este mai mare decât media numărului total de boarding pentru toate aeroporturile.

proc sql number;

select LocID, avg(Boarding) as AvgBoarding2013\_2016 format=COMMA15.

from p1.boarding2013\_2016

group by LocID

having AvgBoarding2013\_2016 > (select avg(Boarding)

from p1.boarding2013\_2016)

order by AvgBoarding2013\_2016 desc;

quit;

A screenshot of a computer screen

Description automatically generated

1. Calculați procentul de enplanement pentru fiecare aeroport raportat la numărul total de îmbarcări.

proc sql number;

select LocID, Enplanement/(select sum(Enplanement)

from p1.enplanement2017)

as ProcEnplanement format=percent7.2

from p1.enplanement2017

order by ProcEnplanement desc;

quit;

A screenshot of a table

Description automatically generated

1. Să se afișeze numărul total de boarding pentru cei 4 ani și numărul total de reclamați pentru fiecare aeroport în parte.

proc sql outobs=300 number;

select distinct b.LocID,

sum(bd.Boarding) as TotalBoarding format=COMMA15.,

c.TotalClaims

from (select Airport\_Code, count(\*) as TotalClaims

from p1.claimsraw

group by Airport\_Code) as c inner join

p1.boarding2013\_2016 as b

on c.Airport\_Code = b.LocID inner join p1.boarding2013\_2016 as bd

on b.LocID = bd.LocID

group by b.LocID

order by b.LocID;

quit;

A screenshot of a computer screen

Description automatically generated

1. Creare View

proc sql;

create view p1.totalClaims as

select Airport\_Code, count(\*) as TotalClaims

from p1.claimsraw

group by AIrport\_Code;

quit;

1. A screenshot of a computer

   Description automatically generatedUtilizare View în interogare

proc sql number;

select distinct b.LocID,

sum(bd.Boarding) as TotalBoarding format=COMMA15.,

c.TotalClaims

from p1.totalClaims as c inner join p1.boarding2013\_2016 as b

on c.Airport\_Code = b.LocID inner join p1.boarding2013\_2016 as bd

on b.LocID = bd.LocID

group by b.LocID

order by LocID;

quit;

A screenshot of a computer screen

Description automatically generated

**Lecția 5.** În cadrul unor interogări utilizați operatorii pe mulțimi: INTERSECT, EXCEPT, UNION și OUTER UNION. Puteți crea în prealabil tabele temporare. Forțați rezultatul interogărilor și în cazul în care nu sunt toate coloanele identice;

1. Selectați aeroporturile pentru care s-au înregistrat reclamații care există în tabela boarding2013\_2016.

proc sql number;

select Airport\_Code

from p1.claimsraw

intersect

select LocID

from p1.boarding2013\_2016;

quit;

A screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generated

1. Selectati aeroporturile pentru care s-au inregistrat doar informații despre Enplanement incepând cu anul 2017.

proc sql number;

select LocID

from p1.enplanement2017

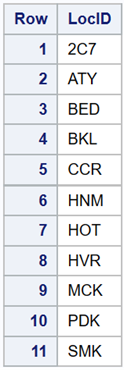
except

select LocID

from p1.boarding2013\_2016;

quit;

A screenshot of a computer

Description automatically generated 

1. Selectați aeroporturile pentru care s-au înregistrat reclamații dar nu există informații despre Boarding sau Enplanement.

proc sql number;

select Airport\_Code

from p1.claimsraw

except

(select LocID

from p1.enplanement2017

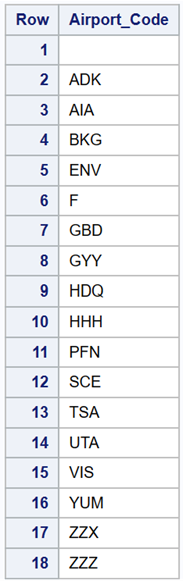
union

select LocID

from p1.boarding2013\_2016);

quit;

A screenshot of a computer

Description automatically generated

1. Selectati aeroporturile pentru care s-au inregistrat informatii despre Boarding sau Enplanement.

proc sql number;

select LocID

from p1.enplanement2017

union

select LocID

from p1.boarding2013\_2016;

quit;

A screenshot of a computer

Description automatically generated A screenshot of a computer

Description automatically generated

1. Uniți și corelați înregistrările din tabelele boarding2013\_2016 și enplanement2017.

proc sql number;

select LocID, input(Year, NUMX4.) as Year, Enplanement

from p1.enplanement2017

outer union corr

select LocID, Year, Boarding as Enplanement

from p1.boarding2013\_2016

group by LocID;

quit;

A screenshot of a table

Description automatically generated A screenshot of a computer screen

Description automatically generated

1. Creare table

proc sql number;

create table totalEnplanements as

select LocID, input(Year, NUMX4.) as Year, Enplanement

from p1.enplanement2017

outer union corr

select LocID, Year, Boarding as Enplanement

from p1.boarding2013\_2016

group by LocID;

quit;

A screenshot of a computer

Description automatically generated

**Lecția 6.** Definiți și utilizați câte 3 macrovariabile definite de utilizator, respectiv rezultate din date;

1. Selectați aeroporturile din statul Florida pentru care au fost înregistrate mai mult de 40 de reclamați.

%let State=FL;

%let MinClaims=40;

proc sql number;

select distinct Airport\_Code, Airport\_Name, StateName,

State, County, City, Count(Airport\_Code) as NumberClaims

from p1.claimsraw

where State = "&State"

group by Airport\_Code

having calculated NumberClaims > &MinClaims

order by calculated NumberClaims desc;

A table with names and numbers

Description automatically generated with medium confidencequit;

1. Selectați reclamațiile pentru care s-a acordat o sumă de bani în vederea soluționări mai mare de 1000 unități monetare.

%let MinAmount=1000;

proc sql number;

select Claim\_Number, Date\_Received, Incident\_Date, Airport\_Code,

Claim\_Type, Claim\_Site, Close\_Amount, Disposition, StateName

from p1.claimsraw

where Close\_Amount > &MinAmount and StateName is not null

order by Close\_Amount desc;

A screenshot of a computer screen

Description automatically generatedquit;

1. Selectați aeroporturile pentru care s-a înregistrat un număr de Enplanement mai mic de 20000.

%let MaxEnplanement=20000;

proc sql number;

select \*

from p1.enplanement2017

where Enplanement < &MaxEnplanement;

quit;

A table with numbers and letters

Description automatically generatedA screenshot of a table

Description automatically generated

1. Creați trei macrovariabile rezultate din date în care se va stoca minimul valorilor pentru Enplanement, media valorilor pentru Boarding și suma maximă acordată pentru soluționarea reclamațiilor.

proc sql;

select min(Enplanement), avg(Boarding)

into :MinEnplanement, :AvgBoarding

from p1.enplanement2017, p1.boarding2013\_2016;

quit;

A white rectangular box with black numbers

Description automatically generated

proc sql;

select max(Close\_Amount)

into :MaxCloseAmount

from p1.claimsraw;

quit;

A white rectangular object with black numbers

Description automatically generated

1. Afișați aeroporturile care au o valoare pentru Enplanement egală cu minimul.

proc sql number;

select LocID, Year, Enplanement

from p1.enplanement2017

where Enplanement = &MinEnplanement;

quit;

A calendar with numbers and words

Description automatically generated

1. Afișați aeroporturile care au o valoare pentru Boarding mai mare decât media.

proc sql number;

select LocID, Year, Boarding

from p1.boarding2013\_2016

where Boarding > &AvgBoarding;

quit;

A table with numbers and letters

Description automatically generated

1. Selectați reaclamațiile pentru care s-a acordat, în vederea soluționări, suma maximă de unități monetare.

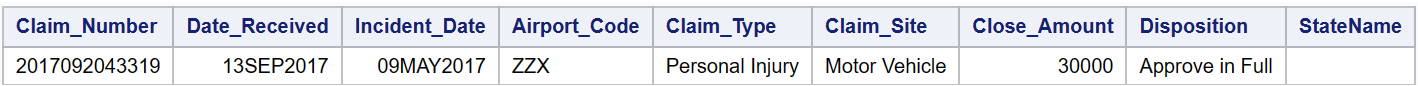
proc sql;

select Claim\_Number, Date\_Received, Incident\_Date, Airport\_Code,

Claim\_Type, Claim\_Site, Close\_Amount, Disposition, StateName

from p1.claimsraw

where Close\_Amount = &MaxCloseAmount;

quit;

**Lecția 7.** Simulați o interogare Access pentru baza de date */ESQ1M6/database/SQL\_DB.accdb*. Scrieți o interogare SQL FedSQL.

1. Simulați o interogare Access pentru baza de date */ESQ1M6/database/SQL\_DB.accdb.*

libname db pcfiles path="&path/database/SQL\_DB.accdb" dbpassword=sastest;

proc sql outobs=10;

select Region, Division, Name, PopEstimate1

from db.statepopulation

where PopEstimate1 > 10000000;

quit;

libname db clear;

A screen shot of a computer

Description automatically generated

1. Scrieți o interogare SQL FedSQL.

proc fedsql number;

select Region, Division, Name, PopEstimate1

from sq.statepopulation

where PopEstimate1 > 10000000;

quit;

A table with numbers and names

Description automatically generated