**UVT**

****

prog studii

INFORMATICA ECONOMICA

**PROIECT SISTEME INFORMATICE PENTRU ASISTAREA DECIZIILOR**

Analiza și vizualizarea datelor despre vanzări

pentru suportul decizional

**STUDENT:**

Ianculescu Alexandra Mihaela

**Timisoara**

**2024**

Contents

[**1.** **Prelucrari in SQL Server** 3](#_Toc172232988)

[**1.1.** **Baza de date** 3](#_Toc172232989)

[**1.2.** **Depozitul de date** 5](#_Toc172232990)

[**1.3.** **Interogari SQL, functii, proceduri si triggere** 7](#_Toc172232991)

[**2.** **Prelucrari in MS Power BI** 12](#_Toc172232992)

[**2.1.** **Importul datelor din SQL Server** 12](#_Toc172232993)

[**2.2.** **Utilizarea unor functii DAX. Coloane calculate si masuri KPI** 12](#_Toc172232994)

[**2.3.** **Utilizarea unor script-uri Python** 14](#_Toc172232995)

[**2.4.** **Dashboard-uri** 17](#_Toc172232996)

[**3.** **Generarea de rapoarte utilizand Report Viewer Control for .Net Windows** 21](#_Toc172232997)

[**3.1.** **Raport 1 ( cu parametru)** 22](#_Toc172232998)

[**3.2.** **Raport 2 (fara parametru)** 23](#_Toc172232999)

[**3.3.** **Raport 3 (indicatori KPI)** 25](#_Toc172233000)

[**3.4.** **Raport 4 (Grafic)** 28](#_Toc172233001)

[**4.** **Prelucrari utilizand MS PowerPivot** 30](#_Toc172233002)

# **Prelucrari in SQL Server**

## **Baza de date**

Am create o baza de date cu urmatoarele tabele si campuri:

* **geo**: Stochează informații geografice.
* **GeoID**: Identificator unic pentru o locație geografică
* **Geo**: Nnume al locației geografice
* **Region:** Regiune asociată cu locația geografică
* **people**: Stochează informații despre vânzători.
* **Salesperson**: Numele vânzătorului
* **SPID**: Identificator unic pentru un vânzător
* **Team**: Echipa sau grupul din care face parte vânzătorul
* **Loc**: Locația sau zona de operare a vânzătorului
* **products**: Stochează informații despre produse.
  + **pid:** Identificator unic pentru un produs
  + **product**: Numele produsului
  + **category**: Categorie sau tipul produsului
  + **size**: Specificația dimensiunii produsului
  + **cost\_per\_box**: Costul produsului per cutie
* **sales**: Stochează datele de vânzări.
  + **spid:** Face referire la SPID din tabela people, indicând care vânzător a realizat vânzarea.
  + **geoid**: Face referire la GeoID din tabela geo, specificând locația geografică a vânzării.
  + **pid**: Face referire la pid din tabela products, indicând care produs a fost vândut.
  + **saledate**: Data și ora vânzării.
  + **customers**: Numărul de clienți implicați în vânzare.
  + **boxes:** Numărul de cutii ale produsului vândute.

Datele sunt importate din fisiere csv.

-- CREAREA TABELELOR

USE SIAD

CREATE TABLE geo (

GeoID VARCHAR(2) PRIMARY KEY,

Geo VARCHAR(20),

Region VARCHAR(15)

);

CREATE TABLE people (

Salesperson VARCHAR(30),

SPID VARCHAR(4) PRIMARY KEY,

Team VARCHAR(20),

Loc VARCHAR(20)

);

CREATE TABLE products (

pid VARCHAR(3) PRIMARY KEY,

product VARCHAR(40),

category VARCHAR(15),

size VARCHAR(6),

cost\_per\_box FLOAT

);

CREATE TABLE sales (

spid VARCHAR(4) REFERENCES people(spid),

geoid VARCHAR(2) REFERENCES geo(geoid),

pid VARCHAR(3) REFERENCES products(pid),

saledate DATETIME,

customers INT,

boxes INT

);

-- COMPLETAREA CU DATE DIN FISIERE CSV

BULK INSERT geo

FROM 'C:\Users\iancu\OneDrive\Desktop\proiectSIAD\geo.csv'

WITH (

FIELDTERMINATOR = ',',

ROWTERMINATOR = '\n',

FIRSTROW = 2

);

BULK INSERT people

FROM 'C:\Users\iancu\OneDrive\Desktop\proiectSIAD\people.csv'

WITH (

FIELDTERMINATOR = ',',

ROWTERMINATOR = '\n',

FIRSTROW = 2

);

BULK INSERT products

FROM 'C:\Users\iancu\OneDrive\Desktop\proiectSIAD\products.csv'

WITH (

FIELDTERMINATOR = ',',

ROWTERMINATOR = '\n',

FIRSTROW = 2

);

BULK INSERT sales

FROM 'C:\Users\iancu\OneDrive\Desktop\proiectSIAD\sales.csv'

WITH (

FIELDTERMINATOR = ',',

ROWTERMINATOR = '\n',

FIRSTROW = 2

);

A screenshot of a computer

Description automatically generated

## **Depozitul de date**

Pe baza modelului de mai sus, am create un deposit de date care contine o tablea de fapte si cinci dimensiuni:

* **analiza**: Aceasta este tabla de fapte unde metricile agregate sunt calculate pentru analiza datelor. Aici, am inclus metrici precum numărul maxim și minim de cutii vândute (max\_boxes, min\_boxes), totalul cutiilor (total\_boxes), media și totalul clienților (avg\_customers, total\_customers), precum și suma totală (amount) bazată pe costul per cutie.
* **dim\_year:** O dimensiune de timp care reține anii unici
* **dim\_month**: O dimensiune de timp care reține luniile unice
* **dim\_geo:** O dimensiune geografică care va stoca informații despre regiuni.
* **dim\_people**: O dimensiune pentru vânzători, care va conține detalii despre fiecare vânzător și echipa lor.
* **dim\_products**: O dimensiune pentru produse, care va conține informații despre fiecare produs.

--CREAREA DEPOZITULUI DE DATE

CREATE TABLE dim\_year ( an INT PRIMARY KEY);

CREATE TABLE dim\_month (luna INT PRIMARY KEY);

CREATE TABLE dim\_geo (geoid VARCHAR(2) PRIMARY KEY, Geo VARCHAR(20) );

CREATE TABLE dim\_people ( spid VARCHAR(4) PRIMARY KEY, Salesperson VARCHAR(30), team VARCHAR(20) );

CREATE TABLE dim\_products ( pid VARCHAR(3) PRIMARY KEY, product VARCHAR(40), category VARCHAR(15));

CREATE TABLE analiza (

spid VARCHAR(4) REFERENCES dim\_people(spid),

geoid VARCHAR(2) REFERENCES dim\_geo(geoid),

pid VARCHAR(3) REFERENCES dim\_products(pid),

an INT REFERENCES dim\_year(an),

luna INT REFERENCES dim\_month(luna),

max\_boxes INT,

min\_boxes INT,

total\_boxes INT,

avg\_customers INT,

total\_customers INT,

amount FLOAT,

PRIMARY KEY (spid, geoid, pid, an, luna)

);

--POPULAREA DEPOZITULUI DE DATE

INSERT INTO dim\_year SELECT DISTINCT YEAR(saledate) FROM sales;

INSERT INTO dim\_month SELECT DISTINCT MONTH(saledate) FROM sales;

INSERT INTO dim\_geo SELECT geoid, geo FROM geo;

INSERT INTO dim\_people SELECT spid, salesperson, team FROM people;

INSERT INTO dim\_products SELECT pid, product, category FROM products;

INSERT INTO analiza

SELECT spid, geoid, sales.pid, YEAR(saledate)as an, MONTH(saledate) as luna, MAX(boxes)as max\_boxes, Min(boxes) as min\_boxes, SUM(boxes) as total\_boxes,

AVG(customers) as avg\_customers , SUM(customers) as total\_customers, SUM(boxes\*cost\_per\_box) as amount

FROM sales, products

WHERE sales.pid=products.pid

GROUP BY spid, geoid, sales.pid, YEAR(saledate), MONTH(saledate) ;

A screenshot of a computer

Description automatically generated

## **Interogari SQL, functii, proceduri si triggere**

* Afișare vânzătorilor care nu au vândut un anumit produs într-o anumită lună:

SELECT

dp.Salesperson,

dpro.product,

dm.luna

FROM

dim\_people dp

CROSS JOIN

dim\_products dpro

CROSS JOIN

dim\_month dm

LEFT JOIN

analiza a ON dp.spid = a.spid AND dpro.pid = a.pid AND dm.luna = a.luna

WHERE

a.spid IS NULL -- Vânzătorul nu a vândut acest produs în această lună

ORDER BY

dp.Salesperson, dpro.product, dm.luna;

A table with numbers and percentages

Description automatically generated

* Afișare mediei lunare a vânzărilor pentru fiecare categorie de produse:

SELECT

dp.category,

ROUND(AVG(a.amount),2) AS Avg\_Monthly\_Sales

FROM

analiza a

JOIN

dim\_products dp ON a.pid = dp.pid

GROUP BY

dp.category

ORDER BY

Avg\_Monthly\_Sales DESC;

A screenshot of a computer

Description automatically generated

* Funcția f2 primește trei parametrii, id-ul unui produs (@pid) și un interval de timp

(@startDate și @endDate). Aceasta extrage din tabelul ‘analiza’ datele care corespund produsului și intervalului de date specificat, stocând aceste informații într-un tabel temporar (@ResultTable).

De exeemplu, funcția este apelată pentru produsul care are id-ul 'P01' și intervalul de date '2022-01-01' până la '2022-12-31', iar rezultatele sunt stocate și apoi afișate din tabelul temporar @Results.

--FUNCTIE PENTRU RETURNAREA VANZARILOR UNUI ANUMIT PRODUS PE O ANUMITA PERIOADA DE TIMP

CREATE OR ALTER FUNCTION f2(@pid VARCHAR(3), @startDate DATE, @endDate DATE)

RETURNS @ResultTable TABLE (

spid VARCHAR(4),

geoid VARCHAR(2),

pid VARCHAR(3),

an INT,

luna INT,

total\_boxes INT,

amount FLOAT

)

AS

BEGIN

DECLARE @spid VARCHAR(4),

@geoid VARCHAR(2),

@pidInternal VARCHAR(3),

@an INT,

@luna INT,

@total\_boxes INT,

@amount FLOAT;

-- Definirea cursorului

DECLARE c CURSOR FOR

SELECT spid, geoid, pid, an, luna, total\_boxes, amount

FROM analiza

WHERE pid = @pid AND CONVERT(DATE, CONCAT(an, '-', luna, '-01')) BETWEEN @startDate AND @endDate;

-- Deschiderea cursorului

OPEN c;

-- Fetch din cursor în variabile și inserarea în tabelul de rezultate

FETCH NEXT FROM c INTO @spid, @geoid, @pidInternal, @an, @luna, @total\_boxes, @amount;

WHILE @@FETCH\_STATUS = 0

BEGIN

INSERT INTO @ResultTable (spid, geoid, pid, an, luna, total\_boxes, amount)

VALUES (@spid, @geoid, @pidInternal, @an, @luna, @total\_boxes, @amount);

FETCH NEXT FROM c INTO @spid, @geoid, @pidInternal, @an, @luna, @total\_boxes, @amount;

END;

-- Închiderea cursorului

CLOSE c;

DEALLOCATE c;

-- Returnarea tabelului de rezultate

RETURN;

END;

-- Apelarea funcției și stocarea rezultatelor într-un tabel temporar

DECLARE @Results TABLE (

spid VARCHAR(4),

geoid VARCHAR(2),

pid VARCHAR(3),

an INT,

luna INT,

total\_boxes INT,

amount FLOAT

);

-- Apelarea funcției cu parametrii specificați

INSERT INTO @Results

SELECT \* FROM f2('P01', '2022-01-01', '2022-12-31');

-- Selectarea rezultatelor din tabelul temporar

SELECT \* FROM @Results;

O imagine care conține text, captură de ecran, număr, Font

Descriere generată automat

* Funcția f11 calculează suma totală a vânzărilor pentru o regiune specificată de

codul @geoid din tabelul ‘analiza’.

Apelând funcția cu @geoid = 'G1', se obține suma totală a vânzărilor pentru

această regiune, care este apoi afișată.

--FUNCTIE PENTRU CALCULAREA VANZARILOR TOTALE INTR-O ANUMITA REGIUNE

CREATE OR ALTER FUNCTION f11(@geoid VARCHAR(2))

RETURNS FLOAT

AS

BEGIN

DECLARE @totalAmount FLOAT;

SELECT @totalAmount = SUM(amount)

FROM analiza

WHERE geoid = @geoid;

RETURN @totalAmount;

END;

--apelarea functiei f11

DECLARE @result FLOAT;

SET @result = dbo.f11('G1');

SELECT @result AS TotalAmount;

O imagine care conține text, captură de ecran, Font, linie

Descriere generată automat

* Această procedură, denumită p11, utilizează funcția f11 pentru a afișa suma totală a

vânzărilor pentru o anumită regiune identificată prin codul @geoid

--PROCEDURA CARE FOLOSESTE FUNCTIA f11 (AFISEAZA VANZARILE TOTALE DINTR-O ANUMITA REGIUNE)

CREATE OR ALTER PROCEDURE p11

@geoid VARCHAR(2)

AS

BEGIN

DECLARE @totalAmount FLOAT;

DECLARE @regionName VARCHAR(20);

-- Obținerea numelui regiunii pentru geoid-ul furnizat

SELECT @regionName = Geo

FROM dim\_geo

WHERE geoid = @geoid;

-- Verificare dacă s-a găsit un nume de regiune

IF @regionName IS NULL

BEGIN

PRINT 'Regiunea data nu exista in baza de date existenta ';

RETURN;

END

-- Apelarea funcției f11 cu @geoid-ul furnizat

SET @totalAmount = dbo.f11(@geoid);

-- Afișarea rezultatului

PRINT 'Totalul vanzarilor pentru ' + @regionName + ' (geoid ' + @geoid + '): ' + CAST(@totalAmount AS VARCHAR(20));

END;

-- Apelarea procedurii cu un anumita regiune

EXEC p11 'G3';

O imagine care conține text, Font, captură de ecran, linie

Descriere generată automat

* Acest trigger, denumit stergere\_produse, este conceput pentru a preveni ștergerea

unui produs din tabelul dim\_products dacă există vânzări asociate cu acel produs în tabelul analiza

--TRIGGER PENTRU ASIGURAREA INTEGRITĂȚII DATELOR, CUM AR FI PREVENIREA ȘTERGERII UNUI PRODUS CARE ARE VÂNZĂRI ASOCIATE.

CREATE TRIGGER stergere\_produse

ON dim\_products

INSTEAD OF DELETE

AS

BEGIN

-- Verifica dacă există înregistrări asociate în tabelul analiza pentru produsele pe care încercăm să le ștergem

IF EXISTS (SELECT 1 FROM analiza WHERE pid IN (SELECT pid FROM deleted))

BEGIN

-- Dacă există înregistrări asociate, generăm o eroare și nu permitem ștergerea

RAISERROR ('Produsul nu se poate sterge. Exista vanzari asociate lui.', 16, 1);

END

ELSE

BEGIN

-- Dacă nu există înregistrări asociate, permitem ștergerea produsului din tabelul dim\_products

DELETE p

FROM dim\_products p

INNER JOIN deleted d ON p.pid = d.pid;

END

END;

--incercam sa stergem produsul P01 pentru care exista vanzari

DELETE FROM dim\_products WHERE pid = 'P01';

O imagine care conține text, captură de ecran, Font, linie

Descriere generată automat

# **Prelucrari in MS Power BI**

## **Importul datelor din SQL Server**

Am important din SQL Server modelul initial, resprctiv tabelele geo, people, products, sales.

A screenshot of a computer

Description automatically generated

## **Utilizarea unor functii DAX. Coloane calculate si masuri KPI**

* In tabelul “sales” am adaugat o noua coloana “amount” in care se calculeaza valoarea vanzarilor (nr cutii vandute \* cost pe cutie)

Amount = sales[boxes] \* RELATED(products[cost\_per\_box])

A screenshot of a computer

Description automatically generated

* Masuri
* numărul total de expedieri

shipment count = COUNTROWS(sales)

* numărul de expedieri care au avut mai puțin de 50 de cutii

Low box shipment = CALCULATE([shipment count],sales[Boxes]<50)

* numărul de expedieri care au avut mai mult de 300 de cutii
* High box shipment = CALCULATE([shipment count],sales[Boxes]>300)
* procentajul de expedieri cu mai puțin de 50 de cutii raportat la numărul total de expedieri

LBS % = DIVIDE([Low box shipment], [shipment count])

* suma totală a cutiilor vandute

Total Boxes = sum(sales[Boxes])

* valoarea totală a vânzărilor

Total Sales = Sum(sales[Amount])

## **Utilizarea unor script-uri Python**

**Analiza cluster a vanzarilor (cu Automatically find clusters)**

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

**Analiza vanzarilor prin clasificare**

# 'dataset' holds the input data for this script

import pandas as pd

import pickle

import numpy as np

from sklearn.neighbors import KNeighborsClassifier

# stabilirea setului de date, a celui de antrenare si de testare

from sklearn.model\_selection import train\_test\_split

X = dataset[['Total Sales', 'Total Boxes']]

Y = dataset[['clasa']]

Y1 = np.ravel(Y)

from sklearn.model\_selection import train\_test\_split

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, Y1, test\_size=0.25, random\_state=101)

# invatarea: antrenarea si crearea modelului

model = KNeighborsClassifier(n\_neighbors=3)

model.fit(X\_train,y\_train)

# modelul se salveaza in fisierul knn.pkl

cale = r"C:\Users\iancu\OneDrive\Desktop\proiectSIAD\cluster\_csv.pkl"

with open(cale, 'wb') as f:

pickle.dump(model, f)

# testarea se va efectua ptr tot X in care se va genera coloana ‘clasa’ cu predictia

loaded\_model = pickle.load(open(cale, 'rb'))

dataset ['clasa\_predictie']=loaded\_model.predict(X)

A computer screen shot of a computer screen

Description automatically generated

A screenshot of a computer

Description automatically generated

Am completat un fisier excel (date\_test.xlsx) un nou set de date cu campurile Salesperson, Total Sales, Total Boxes si clasa, asupra caruia am realizat o predictie ruland urmatorul algoritm care clasifica datele:

# 'dataset' holds the input data for this script

import pandas as pd

import pickle

from sklearn.neighbors import KNeighborsClassifier

X = dataset[['Total Sales', 'Total Boxs']]

cale = r"C:\Users\iancu\OneDrive\Desktop\proiectSIA\date\_test.pkl"

loaded\_model = pickle.load(open(cale, 'rb'))

dataset ['clasa']=loaded\_model.predict(X)

A screenshot of a computer program

Description automatically generated

A screenshot of a computer

Description automatically generated

## **Dashboard-uri**

Am creat un Dashboard bazat pe datele clusterizate, care include:

* un tabel ce prezintă informații despre vânzători, vânzări totale, numărul total de cutii și clusterele asociate.
* un tabel ce arată clusterele și numărul specific de elemente din fiecare cluster.
* un grafic de dispersie care evidențiază cele patru clustere distincte
* un donut chart care evidentieaza marimea celor patru clustere distincte.

A screenshot of a computer

Description automatically generated

* În Dashboard ul de mai jos, sunt prezentate diferite vizualizări care oferă o perspectivă cuprinzătoare asupra performanței vânzărilor din diverse unghiuri:
* Patru Carduri Vizuale: Aceste carduri sintetizează informații cruciale despre vânzări, acoperind aspecte precum vânzările totale, numărul de cutii vândute, numărul de vânzări efectuate și procentajul de expedieri cu mai puțin de 50 de cutii în raport cu totalul.
* Stacked Bar Chart - Categorii de Produse: Acest grafic permite vizualizarea distribuției vânzărilor pe categorii de produse, evidențiind diferențele de performanță între diferitele tipuri de produse.
* Stacked Bar Chart - Zone Geografice: Acest grafic oferă o perspectivă asupra vânzărilor în diferite regiuni geografice, facilitând compararea performanței dintre diverse zone.
* Line Chart - Trendul Vânzărilor: Acest grafic linie analizează evoluția vânzărilor în timp, permițând identificarea tendințelor și a factorilor care influențează performanța pe parcursul unei perioade.
* Stacked Column Chart - Echipa Vânzătorilor: Ultima vizualizare pune în evidență performanța echipelor de vânzători, prezentând vânzările realizate de fiecare echipă și facilitând compararea și analiza performanței acestora.

Prin combinarea acestor vizualizări, utilizatorul are posibilitatea să înțeleagă mai profund și să analizeze eficient performanța vânzărilor din diverse perspective și dimensiuni relevante.

A screenshot of a computer

Description automatically generated

* Urmatorul oferă o analiză detaliată și cuprinzătoare a vânzărilor, folosind următoarele vizualizări:
* Card Vizual: Acest card furnizează informații rapide despre volumul de produse vândute, concentrându-se pe numărul total de cutii comercializate într-un anumit interval sau context specific.
* Diagramă Tip Pie Chart: Prin această vizualizare, distribuția numărului total de cutii vândute în diferite categorii de produse este prezentată vizual, permitând utilizatorului să identifice cu ușurință care segmente contribuie cel mai mult sau cel mai puțin la vânzări.
* Hartă (Map): Acest instrument cartografic analizează modul în care numărul total de cutii vândute este distribuit pe regiuni geografice. Culorile și dimensiunile diferitelor zone permit identificarea rapidă a regiunilor cu performanță înaltă sau scăzută.
* Line Chart: Acest grafic linie oferă o analiză a tendințelor legate de numărul total de cutii vândute în funcție de data vânzării. Acesta permite detectarea sezonierității, fluctuațiilor și altor modele temporale în performanța vânzărilor.
* Decomposition Tree: Cu ajutorul acestei vizualizări, vânzările totale de cutii pot fi analizate și descompuse într-o structură ierarhică. Acest lucru permite explorarea detaliată a vânzărilor în funcție de diferiți parametri, inclusiv agentul de vânzări, categoria și tipul de produs, oferind astfel o înțelegere complexă și detaliată a performanței într-o manieră structurată și clară.

A screenshot of a computer

Description automatically generated

* Dashboard ul 4 (KPI)se concentrează pe monitorizarea și analizarea vânzărilor în funcție de numărul de cutii. Aceasta oferă:
* o comparație între numărul total de cutii vândute (ținta) și numărul de vânzări unde numărul de cutii este mai mic de 50 (valoarea), pe baza masurii

Low box shipment =CALCULATE([shipment count],sales[Boxes]<50) )

* o comparație între numărul total de cutii vândute (ținta) și numărul de vânzări unde numărul de cutii este mai mare decat 300 (valoarea), pe baza masurii

High box shipment = CALCULATE([shipment count],sales[Boxes]>300)

A screenshot of a computer

Description automatically generated

# **Generarea de rapoarte utilizand Report Viewer Control for .Net Windows**

O imagine care conține text, captură de ecran, afișaj, software

Descriere generată automat

## **Raport 1 ( cu parametru)**

## 

Interogarea oferă o analiză a performanței de vânzare a fiecărui vânzător în funcție de lună și an, evidențiind valorile maxime, minime și totale ale cutiilor vândute. Mai exact:

* Calculează maximul (max\_boxes) și minimul (min\_boxes) de cutii vândute de fiecare vânzător într-o anumită lună și an.
* Suma totală (total\_boxes) a cutiilor vândute de fiecare vânzător pentru fiecare combinație de lună și an.
* Rezultatele sunt filtrate pentru un anumit vânzător specificat de variabila @salesperson.

SELECT dim\_people.spid, dim\_people.Salesperson, dim\_month.luna, dim\_year.an, MAX(analiza.max\_boxes) AS max\_boxes, MIN(analiza.min\_boxes) AS min\_boxes, SUM(analiza.total\_boxes) AS total\_boxes

FROM analiza INNER JOIN

dim\_month ON analiza.luna = dim\_month.luna INNER JOIN

dim\_year ON analiza.an = dim\_year.an INNER JOIN

dim\_people ON analiza.spid = dim\_people.spid

WHERE (dim\_people.spid = @salesperson)

GROUP BY dim\_people.spid, dim\_people.Salesperson, dim\_month.luna, dim\_year.an

ORDER BY dim\_people.Salesperson, dim\_month.luna, dim\_year.an

O imagine care conține text, software, captură de ecran, calculator

Descriere generată automat

O imagine care conține text, captură de ecran, software, Software multimedia

Descriere generată automat

O imagine care conține text, captură de ecran, afișaj, software

Descriere generată automat

## **Raport 2 (fara parametru)**

Interogarea analizează datele din două tabele, analiza și dim\_geo, pentru a calcula media, suma și alte agregări legate de numărul de clienți, cutii și sume, grupate după locația geografică

SELECT dim\_geo.geoid, dim\_geo.Geo, AVG(analiza.avg\_customers) AS avg\_customers, SUM(analiza.total\_customers) AS total\_customers, SUM(analiza.total\_boxes) AS total\_boxes, SUM(analiza.amount) AS amount

FROM analiza INNER JOIN

dim\_geo ON analiza.geoid = dim\_geo.geoid

GROUP BY dim\_geo.geoid, dim\_geo.Geo

ORDER BY dim\_geo.Geo

O imagine care conține text, software, captură de ecran, Software multimedia

Descriere generată automat

O imagine care conține text, captură de ecran, software, electronice

Descriere generată automat

O imagine care conține text, electronice, afișaj, captură de ecran

Descriere generată automat

## **Raport 3 (indicatori KPI)**

Analiza datele din tabela analiza în contextul informațiilor din tabelele dim\_year și dim\_month. Analiza se concentrează pe următoarele aspecte:

-Total Boxes (total\_boxes): Se calculează suma totală a cutiilor pentru fiecare combinație de an și lună.

-Total Customers (total\_customers): Se calculează suma totală a clienților pentru fiecare combinație de an și lună.

-Amount: Se calculează suma totală a unei vanzarilor pentru fiecare combinație de an și lună.

SELECT dim\_year.an, dim\_month.luna, SUM(analiza.total\_boxes) AS total\_boxes, SUM(analiza.total\_customers) AS total\_customers, ROUND(SUM(analiza.amount), 2) AS amount

FROM analiza INNER JOIN

dim\_year ON analiza.an = dim\_year.an INNER JOIN

dim\_month ON analiza.luna = dim\_month.luna

GROUP BY dim\_year.an, dim\_month.luna

ORDER BY dim\_year.an, dim\_month.luna

A screenshot of a computer

Description automatically generated

O imagine care conține captură de ecran, text, software, Software multimedia

Descriere generată automat

Indicator KPI care analizeaza campul amount, adica suma finala a vanzarilor pe fiacare an si luna, care va avea urmatoarele stari:

* Rosu- minim 500 000
* Galben intre 500 000-700 000
* Verde intre 700 000 – 1 500 000.

O imagine care conține text, software, Software multimedia, Pictogramă computer

Descriere generată automat

A screenshot of a computer

Description automatically generated

## **Raport 4 (Grafic)**

Analiza care prezintă totalurile pentru fiecare categorie de produse pentru o anumită lună și an specificat.

SELECT dim\_products.category, SUM(analiza.total\_boxes) AS total\_boxes, SUM(analiza.total\_customers) AS total\_customers, ROUND(SUM(analiza.amount), 2) AS amount

FROM analiza INNER JOIN

dim\_products ON analiza.pid = dim\_products.pid INNER JOIN

dim\_month ON analiza.luna = dim\_month.luna INNER JOIN

dim\_year ON analiza.an = dim\_year.an

WHERE (dim\_year.an = @y) AND (dim\_month.luna = @m)

GROUP BY dim\_products.category, dim\_month.luna

O imagine care conține text, captură de ecran, software, Software multimedia

Descriere generată automat

O imagine care conține text, captură de ecran, software, Pictogramă computer

Descriere generată automat

O imagine care conține text, captură de ecran, afișaj, software

Descriere generată automat

# **Prelucrari utilizand MS PowerPivot**

**Analiza performanței vânzătorilor în funcție de produse și regiuni:**

SELECT d\_people.Salesperson, d\_geo.Geo, SUM(a.amount) as TotalAmount

FROM analiza a

JOIN dim\_people d\_people ON a.spid = d\_people.spid

JOIN dim\_geo d\_geo ON a.geoid = d\_geo.geoid

GROUP BY d\_people.Salesperson, d\_geo.Geo

ORDER BY TotalAmount DESC;

O imagine care conține text, electronice, captură de ecran, afișaj

Descriere generată automat

**Camp calculat pentru diferența dintre vânzările unui vânzător și media totală**

=[Suma pentru TotalAmount] - AVERAGE([TotalAmount])

A screenshot of a computer

Description automatically generated

**Măsura pentru calcularea procentajul vânzărilor fiecărui vânzător din totalul general**

=ROUND(DIVIDE( [Suma pentru TotalAmount], CALCULATE(SUM([TotalAmount]), ALL(Interogare[Salesperson]))) \* 100, 2)

O imagine care conține text, captură de ecran, software, afișaj

Descriere generată automat

O imagine care conține text, captură de ecran, software, Pictogramă computer

Descriere generată automat

O imagine care conține text, captură de ecran, număr, software

Descriere generată automat

A screenshot of a computer

Description automatically generated