IF EXISTS (SELECT \* FROM sys.databases WHERE name = 'LucruIndividual')

BEGIN

USE master;

DROP DATABASE LucruIndividual;

END;

CREATE DATABASE LucruIndividual;

-- Creare tabela Magazin

CREATE TABLE LucruIndividual.dbo.Magazin (

MagazinID TINYINT IDENTITY(1,1) PRIMARY KEY,

NumeMagazin VARCHAR(255),

Adresa VARCHAR(255)

);

-- Inserare date in tabela Magazin

INSERT INTO LucruIndividual.dbo.Magazin (NumeMagazin, Adresa)

VALUES

('Mega Store', 'Strada Primaverii, nr. 123'),

('SuperMart', 'Bulevardul Victoriei, nr. 456'),

('Fresh Market', 'Aleea Fructelor, nr. 789'),

('Bonus', 'Bulevardul Unirii, nr. 789'),

('Nr1', 'Strada Luminii, nr. 456'),

('Linella', 'Aleea Viselor, nr. 123'),

('Kaufland', 'Strada Soarelui, nr. 101'),

('Local', 'Bulevardul Lunii, nr. 222'),

('Metro', 'Aleea Stelelor, nr. 333'),

('City Supermarket', 'Strada Norilor, nr. 444'),

('Mall', 'Bulevardul Ploilor, nr. 555'),

('Sun City', 'Aleea Fulgilor, nr. 666'),

('Blue Sky Market', 'Strada Vantului, nr. 777'),

('Nr2', 'Sarmisegetusa, nr.43'),

('Metro 2', 'Dacia, nr21');

-- Creare tabela Furnizori

CREATE TABLE LucruIndividual.dbo.Furnizori (

FurnizorID SMALLINT PRIMARY KEY,

Denumire VARCHAR(255),

Adresa VARCHAR(255)

);

-- Inserare date in tabela Furnizori

INSERT INTO LucruIndividual.dbo.Furnizori (FurnizorID, Denumire, Adresa)

VALUES

(1, 'Flroare', 'Strada Fagilor, nr. 10'),

(2, 'Carpet', 'Aleea Florilor, nr. 25'),

(3, 'Alimente SRL', 'Bulevardul Rozelor, nr. 40'),

(4, 'Delicios Foods SA', 'Aleea Miresmeilor, nr. 15'),

(5, 'Fresco Distributie Alimentara', 'Strada Perlelor, nr. 30'),

(6, 'BioFarm Produse Organice', 'Bulevardul Colibelor, nr. 45'),

(7, 'Gusturi Traditionale Import-Export', 'Aleea Bijuteriilor, nr. 60'),

(8, 'Marele Pantry Importer', 'Strada Elegantei, nr. 75'),

(9, 'ProNatura Distributie', 'Bulevardul Refined, nr. 90'),

(10, 'Alimex Companie Alimentara', 'Aleea Preciosilor, nr. 105'),

(11, 'EcoBio Distributie Bio', 'Strada Eterna, nr. 120'),

(12, 'MegaFoods Importer', 'Bulevardul Modernitatii, nr. 135'),

(13, 'TopGusturi Distributie', 'Aleea Impecabililor, nr. 150'),

(14, 'ProdusLocal SA', 'Dacia, nr32'),

(15, 'ImportExport', 'Independentei, nr 322');

-- Creare tabela Cumparatori

CREATE TABLE LucruIndividual.dbo.Cumparatori (

CumparatorID INT PRIMARY KEY,

Nume VARCHAR(255),

Prenume VARCHAR(255),

Telefon VARCHAR(15)

);

-- Inserare date in tabela Cumparatori

INSERT INTO LucruIndividual.dbo.Cumparatori (CumparatorID, Nume, Prenume, Telefon)

VALUES

(1, 'Ion', 'Popescu', '062048221'),

(2, 'Ana', 'Ionescu', '079342896'),

(3, 'Mihai', 'Georgescu', '062097521'),

(4, 'Elena', 'Dumitru', '062048982'),

(5, 'Adrian', 'Popa', '062918221'),

(6, 'Cristina', 'Vasilescu', '069248221'),

(7, 'George', 'Stancu', '062872121'),

(8, 'Andreea', 'Munteanu', '062049989'),

(9, 'Alex', 'Constantinescu', '069898221'),

(10, 'Laura', 'Moldovan', '062098321'),

(11, 'Vlad', 'Alexandrescu', '068676221'),

(12, 'Raluca', 'Radu', '062088221'),

(13, 'Ciprian', 'Iancu', '062023211'),

(14, 'Catalin', 'Popa', '062028456'),

(15, 'Ion', 'Mardarenco', '079325323');

-- Creare tabela Functii

CREATE TABLE LucruIndividual.dbo.Functii (

FunctieID INT PRIMARY KEY,

Denumire VARCHAR(255),

SefFunctie VARCHAR(255)

);

-- Inserare date in tabela Functii

INSERT INTO LucruIndividual.dbo.Functii (FunctieID, Denumire, SefFunctie)

VALUES

(1, 'Manager', 'Marius Popescu'),

(2, 'Vanzator', 'Andreea Ionescu'),

(3, 'Casier', 'George Dumitrescu'),

(4, 'Receptioner', 'Marius Popescu'),

(5, 'Vanzator secundar', 'Mihai Patic'),

(6, 'Casier principal', 'George Petrache'),

(7, 'Logistician', 'Marius Popescu'),

(8, 'Vanzator principal', 'Cristina Ioana'),

(9, 'Hamal', 'George Ionescu'),

(10, 'Supraveghetor', 'Marius Popescu'),

(11, 'Consultant', 'Catalin Popa'),

(12, 'Sofer', 'George Patic'),

(13, 'Paznic', 'Alexandru Chirica');

-- Creare tabela Angajati

CREATE TABLE LucruIndividual.dbo.Angajati (

AngajatID SMALLINT PRIMARY KEY,

Nume VARCHAR(255),

Prenume VARCHAR(255),

AnulNasterii INT,

FunctieID INT,

NumeComplet AS (Nume + ' ' + Prenume) PERSISTED

FOREIGN KEY (FunctieID) REFERENCES Functii(FunctieID)

);

-- Inserare date in tabela Angajati

INSERT INTO LucruIndividual.dbo.Angajati (AngajatID, Nume, Prenume, AnulNasterii, FunctieID)

VALUES

(1, 'Popescu', 'Ion', 1980, 1),

(2, 'Negura', 'Ana', 1990, 2),

(3, 'Dumitru', 'Mihai', 1985, 3),

(4, 'Gabriela', 'Vasile', 1982, 4),

(5, 'Radu', 'Elena', 1995, 5),

(6, 'Ana', 'Adrian', 1988, 6),

(7, 'Vasile', 'Cristina', 1980, 7),

(8, 'Cristian', 'Mihai', 1992, 8),

(9, 'Mihai', 'Alexandru', 1985, 9),

(10, 'Adrian', 'Laura', 1987, 10),

(11, 'Elena', 'Vlad', 1990, 11),

(12, 'Iuliana', 'Raul', 1983, 12),

(13, 'Valentin', 'Ciprian', 1986, 13),

(14, 'Popa', 'Galina', 1979, 11),

(15, 'Grigorenco', 'Ion', 1899, 13);

-- Creare tabela Produse

CREATE TABLE LucruIndividual.dbo.Produse (

ProdusID INT PRIMARY KEY,

NumeProdus VARCHAR(255),

MagazinID TINYINT,

PretProdus DECIMAL(10, 2),

FurnizorID SMALLINT,

FOREIGN KEY (MagazinID) REFERENCES Magazin(MagazinID),

FOREIGN KEY (FurnizorID) REFERENCES Furnizori(FurnizorID)

);

-- Inserare date in tabela Produse

INSERT INTO LucruIndividual.dbo.Produse (ProdusID, NumeProdus, MagazinID, PretProdus, FurnizorID)

VALUES

(1, 'Făină integrală bio', 1, 50.00, 1),

(2, 'Ovăz ecologic', 4, 30.00, 2),

(3, 'Miere de albine pură', 3, 25.00, 3),

(4, 'Quinoa organica', 7, 40.00, 4),

(5, 'Ulei de măsline extravirgin', 5, 28.00, 5),

(6, 'Nuci organice', 8, 22.00, 6),

(7, 'Ciocolată neagră bio', 9, 60.00, 7),

(8, 'Fructe uscate mix', 9, 35.00, 8),

(9, 'Ceai verde organic', 9, 18.00, 9),

(10, 'Mazăre congelată', 10, 45.00, 10),

(11, 'Oțet balsamic premium', 11, 32.00, 11),

(12, 'Lapte de migdale', 12, 25.00, 12),

(13, 'Sos de roșii organic', 8, 55.00, 13),

(14, 'Peste rosu', 13, 52, 4),

(15, 'Apa minerala', 4, 12, 2);

CREATE TABLE LucruIndividual.dbo.Valabilitate (

ProdusID INT PRIMARY KEY,

DataProducere DATE,

DataExpirare DATE,

FOREIGN KEY (ProdusID) REFERENCES Produse(ProdusID)

);

-- Inserare date in tabela Valabilitate

INSERT INTO LucruIndividual.dbo.Valabilitate (ProdusID, DataProducere, DataExpirare)

VALUES

(1, '2022-01-01', '2022-12-31'),

(2, '2021-05-15', '2022-05-15'),

(3, '2022-03-10', '2023-03-10'),

(4, '2021-11-15', '2022-11-15'),

(5, '2022-06-20', '2023-06-20'),

(6, '2021-08-05', '2022-08-05'),

(7, '2022-02-28', '2023-02-28'),

(8, '2021-12-10', '2022-12-10'),

(9, '2022-04-25', '2023-04-25'),

(10, '2022-07-01', '2023-07-01'),

(11, '2021-09-12', '2022-09-12'),

(12, '2022-01-05', '2023-01-05'),

(13, '2022-10-30', '2023-10-30'),

(14, '2022-11-30', '2024-10-30'),

(15, '2023-12-02', '2024-10-30');

-- Creare tabela Bon

CREATE TABLE LucruIndividual.dbo.Bon (

BonID INT PRIMARY KEY,

CumparatorID INT,

AngajatID SMALLINT,

SumaTotala DECIMAL(10, 2),

FOREIGN KEY (CumparatorID) REFERENCES Cumparatori(CumparatorID),

FOREIGN KEY (AngajatID) REFERENCES Angajati(AngajatID)

);

-- Inserare date in tabela Bon

INSERT INTO LucruIndividual.dbo.Bon (BonID, CumparatorID, AngajatID, SumaTotala)

VALUES

(1, 1, 1, 110.00),

(2, 2, 2, 75.00),

(3, 3, 13, 50.00),

(4, 2, 2, 130.00),

(5, 5, 3, 90.00),

(6, 13, 10, 115.00),

(7, 8, 2, 78.00),

(8, 9, 9, 55.00),

(9, 8, 1, 75.00),

(10, 6, 2, 105.00),

(11, 3, 2, 68.00),

(12, 13, 2, 40.00),

(13, 12, 6, 50.00),

(14, 15, 5, 12),

(15, 3, 1, 100);

-- Creare tabela ProduseBon

CREATE TABLE LucruIndividual.dbo.ProduseBon (

BonID INT,

ProdusID INT,

Cantitate INT,

PRIMARY KEY (BonID, ProdusID),

FOREIGN KEY (BonID) REFERENCES Bon(BonID),

FOREIGN KEY (ProdusID) REFERENCES Produse(ProdusID)

);

-- Inserare date in tabela ProduseBon

INSERT INTO LucruIndividual.dbo.ProduseBon (BonID, ProdusID, Cantitate)

VALUES

(1, 1, 2),

(1, 2, 1),

(2, 3, 3),

(2, 2, 2),

(3, 1, 1),

(3, 3, 2),

(4, 4, 3),

(4, 5, 1),

(5, 6, 2),

(5, 7, 1),

(6, 8, 3),

(6, 9, 1),

(7, 10, 2);

select \* from LucruIndividual.dbo.Angajati

select \* from LucruIndividual.dbo.Bon

select \* from LucruIndividual.dbo.Cumparatori

select \* from LucruIndividual.dbo.Functii

select \* from LucruIndividual.dbo.Furnizori

select \* from LucruIndividual.dbo.Magazin

select \* from LucruIndividual.dbo.Produse

select \* from LucruIndividual.dbo.ProduseBon

select \* from LucruIndividual.dbo.Valabilitate

--va afisa produsul si denumirea furnizorului

SELECT a.NumeProdus, b.Denumire as furnizor

FROM LucruIndividual.dbo.Produse a

INNER JOIN LucruIndividual.dbo.Furnizori b ON a.FurnizorID = b.FurnizorID;

--va afisa cumaraotrii si suma totala a bonuriloe

SELECT Cumparatori.Nume, Cumparatori.Prenume, Bon.SumaTotala

FROM LucruIndividual.dbo.Cumparatori

LEFT JOIN LucruIndividual.dbo.Bon ON Cumparatori.CumparatorID = Bon.CumparatorID;

--Afișează denumirea funcțiilor și numele complet al angajaților. Funcțiile fără angajați vor apărea și ele în rezultat.

SELECT Functii.Denumire, Angajati.NumeComplet

FROM LucruIndividual.dbo.Functii

RIGHT JOIN LucruIndividual.dbo.Angajati ON Functii.FunctieID = Angajati.FunctieID;

--Afișează numele magazinelor, numele produselor și prețurile produselor pentru acele produse care au un preț mai mare de 30.00.

SELECT Magazin.NumeMagazin, Produse.NumeProdus, Produse.PretProdus

FROM LucruIndividual.dbo.Magazin

INNER JOIN LucruIndividual.dbo.Produse ON Magazin.MagazinID = Produse.MagazinID

WHERE Produse.PretProdus > 30.00;

--Afișează numele complet al angajaților, denumirea funcțiilor și suma totală a bonurilor asociate.

SELECT Angajati.NumeComplet, Functii.Denumire, sum(Bon.SumaTotala)

FROM LucruIndividual.dbo.Angajati

INNER JOIN LucruIndividual.dbo.Functii ON Angajati.FunctieID = Functii.FunctieID

INNER JOIN LucruIndividual.dbo.Bon ON Angajati.AngajatID = Bon.AngajatID

group by Angajati.NumeComplet, Functii.Denumire;

--Afișează numele și prenumele cumpărătorilor care au un număr de telefon care începe cu '062'.

SELECT Nume, Prenume, Telefon

FROM LucruIndividual.dbo.Cumparatori

WHERE Telefon LIKE '062%';

--Afișează media duratei de valabilitate (în zile) a produselor din tabela de valabilitate.

SELECT AVG(DATEDIFF(day, DataProducere, DataExpirare)) AS MedieDurataValabilitate

FROM LucruIndividual.dbo.Valabilitate;

--Afișează denumirile și adresele furnizorilor și magazinelor într-un singur set distinct de rezultate.

SELECT Denumire, Adresa FROM LucruIndividual.dbo.Furnizori

UNION

SELECT NumeMagazin, Adresa FROM LucruIndividual.dbo.Magazin;

--Afișează numele complet al angajaților care au asociate bonuri în tabela Bon.

SELECT NumeComplet

FROM LucruIndividual.dbo.Angajati AS A

WHERE EXISTS (

SELECT 1

FROM LucruIndividual.dbo.Bon AS B

WHERE A.AngajatID = B.AngajatID

);

--Afișează numărul de produse pentru fiecare bon, folosind funcția COUNT și GROUP BY.

SELECT a.BonID, COUNT(a.ProdusID) AS NumarProduse

FROM LucruIndividual.dbo.ProduseBon a

GROUP BY a.BonID;

--Afișează numele și prenumele cumpărătorilor care nu au asociate bonuri.

SELECT Cumparatori.Nume, Cumparatori.Prenume

FROM LucruIndividual.dbo.Cumparatori

LEFT JOIN LucruIndividual.dbo.Bon ON Cumparatori.CumparatorID = Bon.CumparatorID

WHERE Bon.BonID IS NULL;

--Afișează produsul și data maximă de expirare asociată fiecărui produs.

SELECT ProdusID, MAX(DataExpirare) AS DataExpirareMaxima

FROM LucruIndividual.dbo.Valabilitate

GROUP BY ProdusID;

--Afișează numărul total de produse cumpărate de fiecare cumpărător.

SELECT B.CumparatorID, COUNT(PB.ProdusID) AS NumarProduse

FROM LucruIndividual.dbo.Bon AS B

INNER JOIN LucruIndividual.dbo.ProduseBon AS PB ON B.BonID = PB.BonID

GROUP BY B.CumparatorID;

--Afișează numele și prețul produselor care au un preț mai mare decât media prețurilor produselor.

SELECT NumeProdus, PretProdus

FROM LucruIndividual.dbo.Produse

WHERE PretProdus > (SELECT AVG(PretProdus) FROM LucruIndividual.dbo.Produse);

--Afișează ID-ul cumpărătorului și suma totală a cumpărăturilor pentru cei cu cumpărături mai mari de 80.00

SELECT B.CumparatorID, SUM(P.PretProdus) AS SumaTotalaCumparaturi

FROM LucruIndividual.dbo.Bon AS B

INNER JOIN LucruIndividual.dbo.ProduseBon AS PB ON B.BonID = PB.BonID

INNER JOIN LucruIndividual.dbo.Produse AS P ON PB.ProdusID = P.ProdusID

GROUP BY B.CumparatorID

HAVING SUM(P.PretProdus) > 80.00;

--Afișează numele complet al angajaților și vârsta lor actuală, calculată pe baza anului de naștere.

SELECT NumeComplet, DATEPART(YEAR, GETDATE()) - AnulNasterii AS Varsta

FROM LucruIndividual.dbo.Angajati

where DATEPART(YEAR, GETDATE()) - AnulNasterii < 50;

--Afișează numele complet al angajaților folosind funcția CONCAT.

SELECT CONCAT(Nume, ' ', Prenume) AS NumeComplet

FROM LucruIndividual.dbo.Angajati;

--Afișează numele și prenumele cumpărătorilor care au asociate bonuri.

SELECT Nume, Prenume

FROM LucruIndividual.dbo.Cumparatori

WHERE CumparatorID IN (SELECT CumparatorID FROM LucruIndividual.dbo.Bon);

--Afișează produsele și perioada de valabilitate care se află între anumite date de producere.

SELECT ProdusID, DataProducere, DataExpirare

FROM LucruIndividual.dbo.Valabilitate

WHERE DataProducere BETWEEN '2022-01-01' AND '2022-12-31';