**PROIECT SGBD**

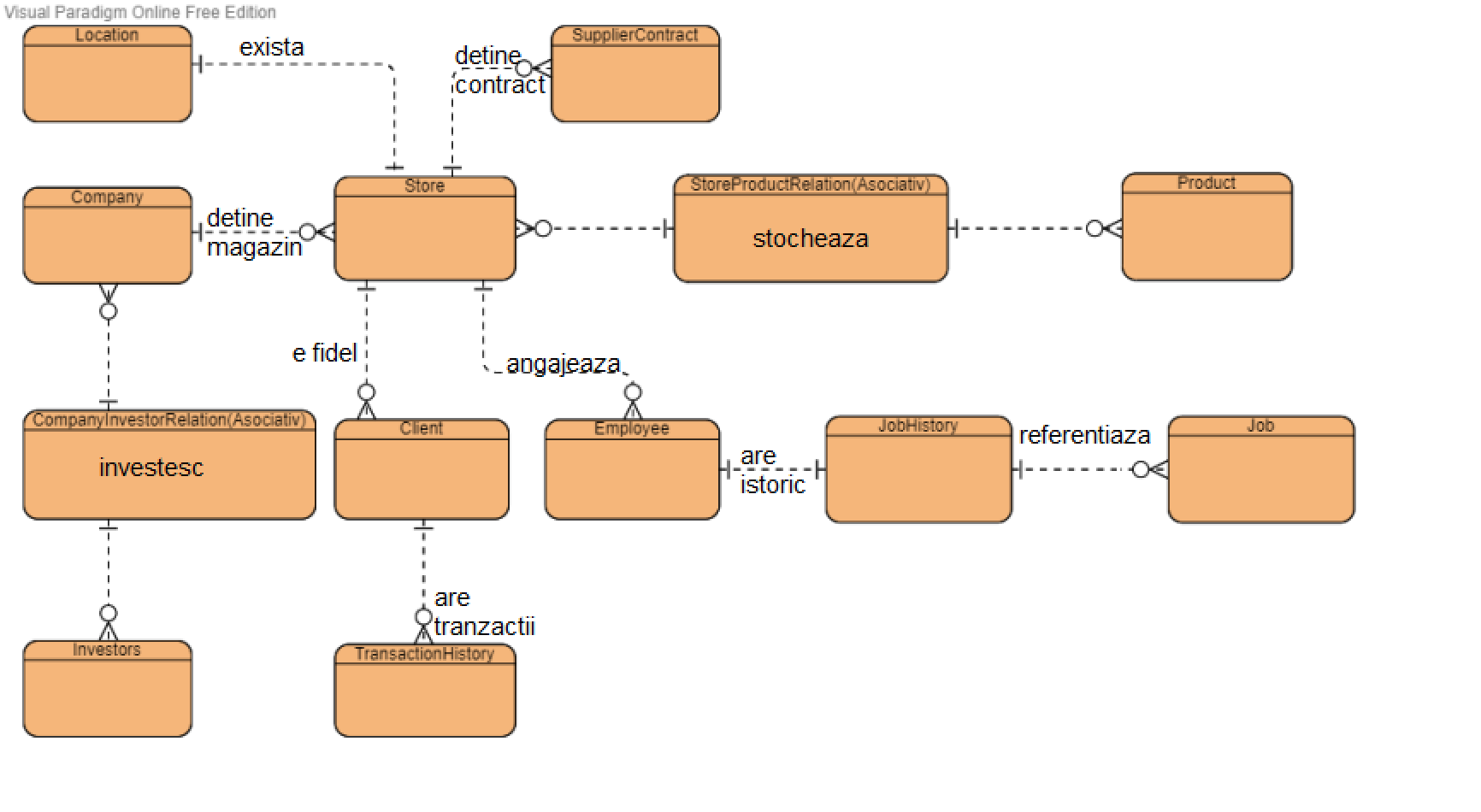
RADA PAUL

GRUPA 241

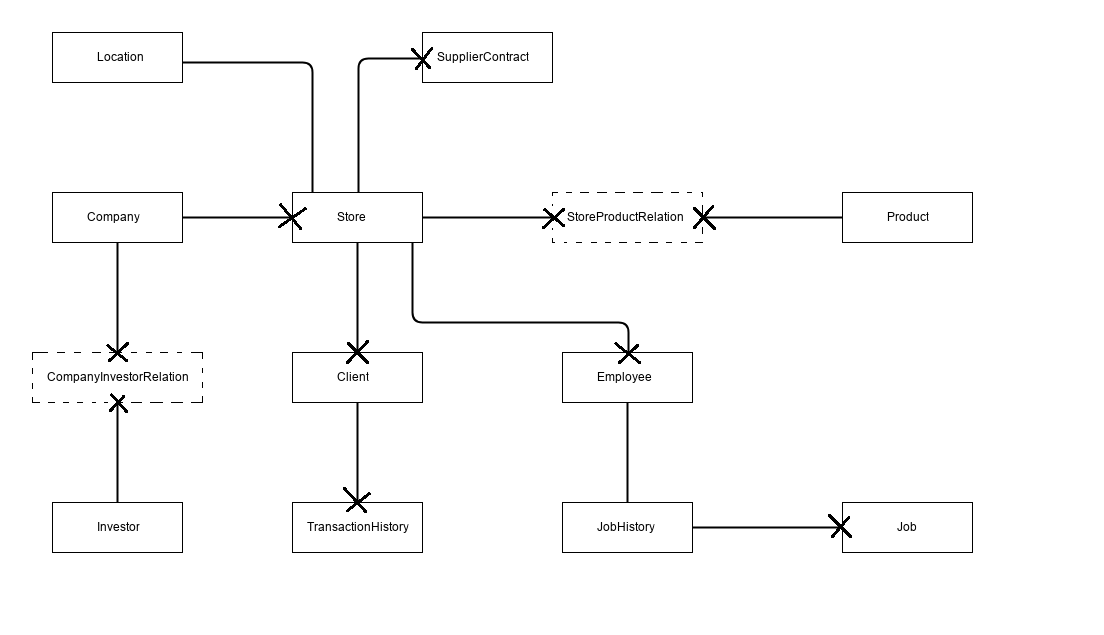
1. Prezentați pe scurt baza de date (utilitatea ei).

Avem un **lant de companii (tCompany)** .Fiecare companie are un numar de **magazine (tStore)** si anumiti **sponsori (tInvestor)** . Fiecare magazin are o **locatie (tLocation)** , **contracte cu furnizori (tSupplierContract)**, **clienti (tClient)** , **angajati (tEmployee)** si **produse (tProduct)** . Un client are un **istoric al tranzactiilor (tTransactionHistory)** . Angajatii detin un **istoric al joburilor (tJobHistory)**. Istoricul joburilor face referinta la **job (tJob)**. Avem tabele asociative intre **companii si sponsori (tCompanyInvestorRelation)** si intre **magazine si produse (tProductStoreRelation)**.

2. Realizați diagrama entitate-relație (ERD).



3. Pornind de la diagrama entitate-relație realizați diagrama conceptuală a modelului propus, integrând toate atributele necesare.



--4+5

REM   Script: DATABASE CREATION

REM   create database

DROP SEQUENCE company\_seq;

DROP TABLE tLocation CASCADE CONSTRAINTS;

DROP TABLE tProduct CASCADE CONSTRAINTS;

DROP TABLE tClient CASCADE CONSTRAINTS;

DROP TABLE tInvestor CASCADE CONSTRAINTS;

DROP TABLE tStore CASCADE CONSTRAINTS;

DROP TABLE tJobHistory CASCADE CONSTRAINTS;

DROP TABLE tJob CASCADE CONSTRAINTS;

DROP TABLE tEmployee CASCADE CONSTRAINTS;

DROP TABLE tCompany CASCADE CONSTRAINTS;

DROP TABLE tSupplierContract CASCADE CONSTRAINTS;

DROP TABLE tTransactionHistory CASCADE CONSTRAINTS;

DROP TABLE tProductStoreRelation CASCADE CONSTRAINTS;

DROP TABLE tCompanyInvestorRelation CASCADE CONSTRAINTS;

CREATE TABLE tCompany (

    id                  INT             NOT NULL,

    name                VARCHAR(50)     NOT NULL,

    ceo                 VARCHAR(50)     NOT NULL,

    cr\_date             DATE                    ,

    value               DECIMAL(18,2)           ,

    PRIMARY KEY (id)

);

CREATE TABLE tStore (

    id                  INT             NOT NULL,

    id\_location         INT             NOT NULL,

    type                VARCHAR(10)     NOT NULL,

    id\_company          INT             NOT NULL,

    PRIMARY KEY (id)

);

CREATE TABLE tSupplierContract(

    id                  INT             NOT NULL,

    name                VARCHAR(50)     NOT NULL,

    price               DECIMAL(9,2)    NOT NULL,

    duration\_months     INT             NOT NULL,

    sign\_date           DATE            NOT NULL,

    id\_store            INT             NOT NULL,

    PRIMARY KEY (id)

);

CREATE TABLE tLocation(

    id                  INT             NOT NULL,

    county              VARCHAR(50)     NOT NULL,

    city                VARCHAR(50)     NOT NULL,

    street              VARCHAR(50)     NOT NULL,

    postal\_code         INT                     ,

    PRIMARY KEY (id)

);

CREATE TABLE tClient(

    id                  INT             NOT NULL,

    id\_store            INT             NOT NULL,

    type                VARCHAR(50)             ,

    first\_name          VARCHAR(50)     NOT NULL,

    last\_name           VARCHAR(50)     NOT NULL,

    email               VARCHAR(50)             ,

    phone\_number        VARCHAR(50)     NOT NULL,

    PRIMARY KEY (id)

);

CREATE TABLE tTransactionHistory(

    id                  INT             NOT NULL,

    id\_client           INT             NOT NULL,

    sum                 DECIMAL(9,2)    NOT NULL,

    cr\_date             TIMESTAMP       NOT NULL,

    PRIMARY KEY (id)

);

CREATE TABLE tEmployee (

    id                  INT             NOT NULL,

    id\_store            INT             NOT NULL,

    id\_job\_history      INT             NOT NULL,

    salary              DECIMAL(11,2)   NOT NULL,

    surname             VARCHAR(50)             ,

    name                VARCHAR(50)             ,

    id\_manager          INT                     ,

    email               VARCHAR(100)            ,

    phone\_number        VARCHAR(10)             ,

    PRIMARY KEY (id)

);

CREATE TABLE tJobHistory(

    id                  INT             NOT NULL,

    id\_job              INT             NOT NULL,

    start\_date          DATE            NOT NULL,

    end\_date            DATE                    ,

    PRIMARY KEY (id)

);

CREATE TABLE tJob(

    id                  INT             NOT NULL,

    title               VARCHAR(50)     NOT NULL,

    min\_salary          INT             NOT NULL,

    max\_salary          INT             NOT NULL,

    PRIMARY KEY (id)

);

CREATE TABLE tProduct (

    id                  INT             NOT NULL,

    name                VARCHAR(20)     NOT NULL,

    price               DECIMAL(9,2)    NOT NULL,

    PRIMARY KEY (id)

);

CREATE TABLE tProductStoreRelation (

    id                  INT             NOT NULL,

    id\_product          INT             NOT NULL,

    id\_store            INT             NOT NULL,

    PRIMARY KEY (id)

);

CREATE TABLE tInvestor (

    id                  INT             NOT NULL,

    name                VARCHAR(50)     NOT NULL,

    PRIMARY KEY (id)

);

CREATE TABLE tCompanyInvestorRelation (

    id                  INT             NOT NULL,

    id\_company          INT             NOT NULL,

    id\_investor         INT             NOT NULL,

    PRIMARY KEY (id)

);

ALTER TABLE tStore ADD CONSTRAINT fk\_id\_company FOREIGN KEY (id\_company) REFERENCES tCompany(id);

ALTER TABLE tStore ADD CONSTRAINT fk\_id\_location FOREIGN KEY (id\_location) REFERENCES tLocation(id);

ALTER TABLE tSupplierContract ADD CONSTRAINT fk\_id\_store FOREIGN KEY (id\_store) REFERENCES tStore(id);

ALTER TABLE tClient ADD CONSTRAINT fk\_id\_store\_tClient FOREIGN KEY (id\_store) REFERENCES tStore(id);

ALTER TABLE tTransactionHistory ADD CONSTRAINT fk\_id\_client FOREIGN KEY (id\_client) REFERENCES tClient(id);

ALTER TABLE tEmployee ADD CONSTRAINT fk\_id\_store\_tEmployee FOREIGN KEY (id\_store) REFERENCES tStore(id);

ALTER TABLE tEmployee ADD CONSTRAINT fk\_id\_job\_history FOREIGN KEY (id\_job\_history) REFERENCES tJobHistory(id);

ALTER TABLE tJobHistory ADD CONSTRAINT fk\_id\_job FOREIGN KEY (id\_job) REFERENCES tJob(id);

ALTER TABLE tProductStoreRelation ADD CONSTRAINT fk\_id\_product FOREIGN KEY (id\_product) REFERENCES tProduct(id);

ALTER TABLE tProductStoreRelation ADD CONSTRAINT fk\_id\_store\_tProductStoreRelation FOREIGN KEY (id\_store) REFERENCES tStore(id);

ALTER TABLE tCompanyInvestorRelation ADD CONSTRAINT fk\_id\_company\_tCompanyInvestorRelation FOREIGN KEY (id\_company) REFERENCES tCompany(id);

ALTER TABLE tCompanyInvestorRelation ADD CONSTRAINT fk\_id\_investor FOREIGN KEY (id\_investor) REFERENCES tInvestor(id);

ALTER TABLE tStore

ADD CONSTRAINT tStore\_c1

CHECK (

    type in (

        'NONSTOP',

        'EXPRESS',

        'SUPERSTORE',

        'NORMAL'

    )

);

ALTER TABLE tClient

ADD CONSTRAINT tClient\_c1

CHECK (

    type in (

        'Aur',

        'Platina',

        'Argint',

        'Diamant',

        'Bronz'

    )

);

ALTER TABLE tJob

ADD CONSTRAINT tJob\_c1

CHECK (

    title in (

        'Programator',

        'Vanzator',

        'Securitate',

        'Electrician',

        'Om de serviciu',

        'Manager',

        'HR',

        'Bucatar',

        'QA',

        'Lucrator la depozit'

    )

);

    CREATE SEQUENCE company\_seq START WITH 0 INCREMENT BY 1 MINVALUE 0;

    INSERT INTO tCompany

        (id, name, ceo, cr\_date, value)

    VALUES

        (company\_seq.nextval, 'Lorsus', 'Maria Ionescu', DATE '2009-06-24', 19173810);

    INSERT INTO tCompany

        (id, name, ceo, cr\_date, value)

    VALUES

        (company\_seq.nextval, 'Platinum Home', 'Ioana Ungureanu', DATE '2007-10-31', 7276086171);

    INSERT INTO tCompany

        (id, name, ceo, cr\_date, value)

    VALUES

        (company\_seq.nextval, 'Xaris', 'Paula Popescu', DATE '2002-03-04', 24140019);

    INSERT INTO tCompany

        (id, name, ceo, cr\_date, value)

    VALUES

        (company\_seq.nextval, 'Horrisons', 'Daria Moldovan', DATE '2002-03-25', 5546498854);

    INSERT INTO tCompany

        (id, name, ceo, cr\_date, value)

    VALUES

        (company\_seq.nextval, 'Micro', 'Ada Simon', DATE '1991-01-10', 65636840);

    INSERT INTO tCompany

        (id, name, ceo, cr\_date, value)

    VALUES

        (company\_seq.nextval, 'KIDL', 'Catalina Rusu', DATE '1998-06-27', 76758296);

    INSERT INTO tCompany

        (id, name, ceo, cr\_date, value)

    VALUES

        (company\_seq.nextval, 'Rafus', 'Marian Marginean', DATE '1997-04-11', 56217890);

    INSERT INTO tCompany

        (id, name, ceo, cr\_date, value)

    VALUES

        (company\_seq.nextval, 'AZDA', 'Alexandra Costea', DATE '2009-02-24', 723619940);

    INSERT INTO tCompany

        (id, name, ceo, cr\_date, value)

    VALUES

        (company\_seq.nextval, 'Aqua', 'Alexandra Popa', DATE '1997-05-10', 39473874);

    INSERT INTO tCompany

        (id, name, ceo, cr\_date, value)

    VALUES

        (company\_seq.nextval, 'Balors', 'Razvan Tudor', DATE '1994-01-14', 185560072);

    INSERT INTO tInvestor

        (id, name)

    VALUES

        (0, 'Dante');

    INSERT INTO tInvestor

        (id, name)

    VALUES

        (1, 'Play');

    INSERT INTO tInvestor

        (id, name)

    VALUES

        (2, 'Foint');

    INSERT INTO tInvestor

        (id, name)

    VALUES

        (3, 'Emporium');

    INSERT INTO tInvestor

        (id, name)

    VALUES

        (4, 'Bochan');

    INSERT INTO tInvestor

        (id, name)

    VALUES

        (5, 'Erlin');

    INSERT INTO tInvestor

        (id, name)

    VALUES

        (6, 'Jankus');

    INSERT INTO tInvestor

        (id, name)

    VALUES

        (7, 'Helbis');

    INSERT INTO tInvestor

        (id, name)

    VALUES

        (8, 'Nezlo');

    INSERT INTO tInvestor

        (id, name)

    VALUES

        (9, 'Experience');

    INSERT INTO tCompanyInvestorRelation

        (id, id\_investor, id\_company)

    VALUES

        (0, 8, 2);

    INSERT INTO tCompanyInvestorRelation

        (id, id\_investor, id\_company)

    VALUES

        (1, 5, 9);

    INSERT INTO tCompanyInvestorRelation

        (id, id\_investor, id\_company)

    VALUES

        (2, 1, 2);

    INSERT INTO tCompanyInvestorRelation

        (id, id\_investor, id\_company)

    VALUES

        (3, 2, 4);

    INSERT INTO tCompanyInvestorRelation

        (id, id\_investor, id\_company)

    VALUES

        (4, 8, 7);

    INSERT INTO tCompanyInvestorRelation

        (id, id\_investor, id\_company)

    VALUES

        (5, 1, 5);

    INSERT INTO tCompanyInvestorRelation

        (id, id\_investor, id\_company)

    VALUES

        (6, 6, 3);

    INSERT INTO tCompanyInvestorRelation

        (id, id\_investor, id\_company)

    VALUES

        (7, 6, 8);

    INSERT INTO tCompanyInvestorRelation

        (id, id\_investor, id\_company)

    VALUES

        (8, 6, 4);

    INSERT INTO tCompanyInvestorRelation

        (id, id\_investor, id\_company)

    VALUES

        (9, 5, 7);

    INSERT INTO tLocation

        (id, county, city, street, postal\_code)

    VALUES

        (0, 'Dambovita', 'Targoviste', 'Fericirii', 021345);

    INSERT INTO tLocation

        (id, county, city, street, postal\_code)

    VALUES

        (1, 'Covasna', 'Sfantu Gheorghe', 'Mantuirii', 054321);

    INSERT INTO tLocation

        (id, county, city, street, postal\_code)

    VALUES

        (2, 'Vrancea', 'Focsani', 'Bucuriei', 099772);

    INSERT INTO tLocation

        (id, county, city, street, postal\_code)

    VALUES

        (3, 'Dolj', 'Craiova', 'Rusaliilor', 072727);

    INSERT INTO tLocation

        (id, county, city, street, postal\_code)

    VALUES

        (4, 'Bihor', 'Oradea', 'Sperantei', 065432);

    INSERT INTO tLocation

        (id, county, city, street, postal\_code)

    VALUES

        (5, 'Gorj', 'Targu Jiu', 'Slabiciunii', 010199);

    INSERT INTO tLocation

        (id, county, city, street, postal\_code)

    VALUES

        (6, 'Ilfov', 'Bucuresti', 'V.V.Stanciu', 098765);

    INSERT INTO tLocation

        (id, county, city, street, postal\_code)

    VALUES

        (7, 'Olt', 'Slatina', 'Vulturului', 013370);

    INSERT INTO tLocation

        (id, county, city, street, postal\_code)

    VALUES

        (8, 'Alba', 'Alba Iulia', 'Stefan cel Mare', 087654);

    INSERT INTO tLocation

        (id, county, city, street, postal\_code)

    VALUES

        (9, 'Hunedoara', 'Deva', 'Vointei', 088877);

    INSERT INTO tJob

        (id, title, min\_salary, max\_salary)

    VALUES

        (0, 'Om de serviciu', 75265, 341382);

    INSERT INTO tJob

        (id, title, min\_salary, max\_salary)

    VALUES

        (1, 'Vanzator', 48492, 659990);

    INSERT INTO tJob

        (id, title, min\_salary, max\_salary)

    VALUES

        (2, 'HR', 59847, 479696);

    INSERT INTO tJob

        (id, title, min\_salary, max\_salary)

    VALUES

        (3, 'Programator', 61595, 436980);

    INSERT INTO tJob

        (id, title, min\_salary, max\_salary)

    VALUES

        (4, 'Bucatar', 75460, 128846);

    INSERT INTO tJob

        (id, title, min\_salary, max\_salary)

    VALUES

        (5, 'Securitate', 11166, 115100);

    INSERT INTO tJob

        (id, title, min\_salary, max\_salary)

    VALUES

        (6, 'Lucrator la depozit', 52197, 491538);

    INSERT INTO tJob

        (id, title, min\_salary, max\_salary)

    VALUES

        (7, 'Electrician', 65410, 829280);

    INSERT INTO tJob

        (id, title, min\_salary, max\_salary)

    VALUES

        (8, 'QA', 95510, 859832);

    INSERT INTO tJob

        (id, title, min\_salary, max\_salary)

    VALUES

        (9, 'Manager', 79757, 710392);

    INSERT INTO tJobHistory

        (id, id\_job, start\_date, end\_date)

    VALUES

        (0, 1, DATE '2016-09-25', DATE '2021-12-08');

    INSERT INTO tJobHistory

        (id, id\_job, start\_date, end\_date)

    VALUES

        (1, 1, DATE '2015-11-29', DATE '2020-03-21');

    INSERT INTO tJobHistory

        (id, id\_job, start\_date, end\_date)

    VALUES

        (2, 9, DATE '2015-05-25', DATE '2020-01-09');

    INSERT INTO tJobHistory

        (id, id\_job, start\_date, end\_date)

    VALUES

        (3, 6, DATE '2015-12-29', DATE '2019-06-09');

    INSERT INTO tJobHistory

        (id, id\_job, start\_date, end\_date)

    VALUES

        (4, 7, DATE '2016-04-05', DATE '2021-02-12');

    INSERT INTO tJobHistory

        (id, id\_job, start\_date, end\_date)

    VALUES

        (5, 3, DATE '2016-07-16', DATE '2021-11-05');

    INSERT INTO tJobHistory

        (id, id\_job, start\_date, end\_date)

    VALUES

        (6, 7, DATE '2014-07-30', DATE '2020-09-10');

    INSERT INTO tJobHistory

        (id, id\_job, start\_date, end\_date)

    VALUES

        (7, 7, DATE '2016-09-29', DATE '2020-09-16');

    INSERT INTO tJobHistory

        (id, id\_job, start\_date, end\_date)

    VALUES

        (8, 2, DATE '2014-10-19', DATE '2020-08-10');

    INSERT INTO tJobHistory

        (id, id\_job, start\_date, end\_date)

    VALUES

        (9, 6, DATE '2016-10-10', DATE '2021-10-02');

    INSERT INTO tStore

        (id, id\_location, type, id\_company)

    VALUES

        (0, 8, 'NONSTOP', 9);

    INSERT INTO tStore

        (id, id\_location, type, id\_company)

    VALUES

        (1, 8, 'NORMAL', 0);

    INSERT INTO tStore

        (id, id\_location, type, id\_company)

    VALUES

        (2, 5, 'NONSTOP', 2);

    INSERT INTO tStore

        (id, id\_location, type, id\_company)

    VALUES

        (3, 3, 'EXPRESS', 9);

    INSERT INTO tStore

        (id, id\_location, type, id\_company)

    VALUES

        (4, 9, 'EXPRESS', 0);

    INSERT INTO tStore

        (id, id\_location, type, id\_company)

    VALUES

        (5, 7, 'SUPERSTORE', 8);

    INSERT INTO tStore

        (id, id\_location, type, id\_company)

    VALUES

        (6, 7, 'NONSTOP', 8);

    INSERT INTO tStore

        (id, id\_location, type, id\_company)

    VALUES

        (7, 0, 'NORMAL', 1);

    INSERT INTO tStore

        (id, id\_location, type, id\_company)

    VALUES

        (8, 1, 'NONSTOP', 7);

    INSERT INTO tStore

        (id, id\_location, type, id\_company)

    VALUES

        (9, 9, 'EXPRESS', 4);

    INSERT INTO tSupplierContract

        (id, name, price, duration\_months, sign\_date, id\_store)

    VALUES

        (0, 'Ghiplo', 7672529, 90, DATE '2018-10-06', 1);

    INSERT INTO tSupplierContract

        (id, name, price, duration\_months, sign\_date, id\_store)

    VALUES

        (1, 'Morael', 146600, 47, DATE '2019-01-07', 3);

    INSERT INTO tSupplierContract

        (id, name, price, duration\_months, sign\_date, id\_store)

    VALUES

        (2, 'Orthos', 6576840, 8, DATE '2015-06-06', 8);

    INSERT INTO tSupplierContract

        (id, name, price, duration\_months, sign\_date, id\_store)

    VALUES

        (3, 'The Dollar Follow', 4345720, 3, DATE '2015-07-27', 6);

    INSERT INTO tSupplierContract

        (id, name, price, duration\_months, sign\_date, id\_store)

    VALUES

        (4, 'Iartur', 5732887, 81, DATE '2019-07-03', 1);

    INSERT INTO tSupplierContract

        (id, name, price, duration\_months, sign\_date, id\_store)

    VALUES

        (5, 'TORA', 969712, 66, DATE '2018-01-10', 6);

    INSERT INTO tSupplierContract

        (id, name, price, duration\_months, sign\_date, id\_store)

    VALUES

        (6, 'Saintsburey', 583649, 7, DATE '2021-05-25', 7);

    INSERT INTO tSupplierContract

        (id, name, price, duration\_months, sign\_date, id\_store)

    VALUES

        (7, 'Pesco', 539420, 9, DATE '2014-01-23', 8);

    INSERT INTO tSupplierContract

        (id, name, price, duration\_months, sign\_date, id\_store)

    VALUES

        (8, 'Golden Roads', 6668022, 9, DATE '2014-09-27', 3);

    INSERT INTO tSupplierContract

        (id, name, price, duration\_months, sign\_date, id\_store)

    VALUES

        (9, 'Cranda', 5964499, 2, DATE '2016-12-12', 3);

    INSERT INTO tEmployee

        (id, id\_store, id\_job\_history, salary, surname, name, id\_manager, email, phone\_number)

    VALUES

        (0, 1, 0, 47589, 'Ungureanu', 'Catalina', 9, 'Catalina.Ungureanu91@protonmail.com', '0702072221');

    INSERT INTO tEmployee

        (id, id\_store, id\_job\_history, salary, surname, name, id\_manager, email, phone\_number)

    VALUES

        (1, 5, 1, 76516, 'Ciobanu', 'Radu', 0, 'Radu.Ciobanu91@aol.com', '0734164410');

    INSERT INTO tEmployee

        (id, id\_store, id\_job\_history, salary, surname, name, id\_manager, email, phone\_number)

    VALUES

        (2, 3, 2, 6637, 'Mocanu', 'Georgiana', 2, 'Georgiana.Mocanu40@outlook.com', '0756388946');

    INSERT INTO tEmployee

        (id, id\_store, id\_job\_history, salary, surname, name, id\_manager, email, phone\_number)

    VALUES

        (3, 7, 3, 88855, 'Munteanu', 'Ada', 8, 'Ada.Munteanu61@yahoo.com', '0790433674');

    INSERT INTO tEmployee

        (id, id\_store, id\_job\_history, salary, surname, name, id\_manager, email, phone\_number)

    VALUES

        (4, 7, 4, 4221, 'Rusu', 'Laura', 6, 'Laura.Rusu93@protonmail.com', '0756315239');

    INSERT INTO tEmployee

        (id, id\_store, id\_job\_history, salary, surname, name, id\_manager, email, phone\_number)

    VALUES

        (5, 9, 5, 44551, 'Popescu', 'Stefania', 7, 'Stefania.Popescu85@aol.com', '0740152915');

    INSERT INTO tEmployee

        (id, id\_store, id\_job\_history, salary, surname, name, id\_manager, email, phone\_number)

    VALUES

        (6, 5, 6, 5604, 'Rusu', 'Paul', 3, 'Paul.Rusu4@gmail.com', '0734128041');

    INSERT INTO tEmployee

        (id, id\_store, id\_job\_history, salary, surname, name, id\_manager, email, phone\_number)

    VALUES

        (7, 8, 7, 9746, 'Stan', 'Ana', 2, 'Ana.Stan44@gmail.com', '0740196308');

    INSERT INTO tEmployee

        (id, id\_store, id\_job\_history, salary, surname, name, id\_manager, email, phone\_number)

    VALUES

        (8, 5, 8, 10800, 'Ungureanu', 'Laur', 9, 'Laur.Ungureanu8@protonmail.com', NULL);

    INSERT INTO tEmployee

        (id, id\_store, id\_job\_history, salary, surname, name, id\_manager, email, phone\_number)

    VALUES

        (9, 9, 9, 57982, 'Baciu', 'Paul', 1, 'Paul.Baciu25@gmail.com', '0756382891');

    INSERT INTO tProduct

        (id, name, price)

    VALUES

        (0, 'Paine', 2);

    INSERT INTO tProduct

        (id, name, price)

    VALUES

        (1, 'Tacamuri', 2);

    INSERT INTO tProduct

        (id, name, price)

    VALUES

        (2, 'Bauturi alcoolice', 764);

    INSERT INTO tProduct

        (id, name, price)

    VALUES

        (3, 'Dulciuri', 4);

    INSERT INTO tProduct

        (id, name, price)

    VALUES

        (4, 'Sanitare', 38);

    INSERT INTO tProduct

        (id, name, price)

    VALUES

        (5, 'Suc', 283);

    INSERT INTO tProduct

        (id, name, price)

    VALUES

        (6, 'Fruct', 492);

    INSERT INTO tProduct

        (id, name, price)

    VALUES

        (7, 'Plusuri', 35);

    INSERT INTO tProduct

        (id, name, price)

    VALUES

        (8, 'Carne', 5);

    INSERT INTO tProduct

        (id, name, price)

    VALUES

        (9, 'Apa', 80);

    INSERT INTO tClient

        (id, id\_store, type, first\_name, last\_name, email, phone\_number)

    VALUES

        (0, 4, 'Argint', 'George', 'Ciobanu', 'George.Ciobanu66@protonmail.com', '0702079588');

    INSERT INTO tClient

        (id, id\_store, type, first\_name, last\_name, email, phone\_number)

    VALUES

        (1, 1, 'Aur', 'Ioana', 'Ion', 'Ioana.Ion55@protonmail.com', '0702037446');

    INSERT INTO tClient

        (id, id\_store, type, first\_name, last\_name, email, phone\_number)

    VALUES

        (2, 2, 'Bronz', 'Vasile', 'Rotaru', 'Vasile.Rotaru54@gmail.com', '0734190879');

    INSERT INTO tClient

        (id, id\_store, type, first\_name, last\_name, email, phone\_number)

    VALUES

        (3, 0, 'Aur', 'George', 'Dumitru', 'George.Dumitru65@aol.com', '0702058146');

    INSERT INTO tClient

        (id, id\_store, type, first\_name, last\_name, email, phone\_number)

    VALUES

        (4, 2, 'Diamant', 'Maria', 'Munteanu', 'Maria.Munteanu8@yahoo.com', '0702005075');

    INSERT INTO tClient

        (id, id\_store, type, first\_name, last\_name, email, phone\_number)

    VALUES

        (5, 4, 'Bronz', 'Alexandru', 'Mocanu', 'Alexandru.Mocanu67@outlook.com', '0702038587');

    INSERT INTO tClient

        (id, id\_store, type, first\_name, last\_name, email, phone\_number)

    VALUES

        (6, 6, 'Bronz', 'George', 'Ionescu', 'George.Ionescu39@outlook.com', '0790440611');

    INSERT INTO tClient

        (id, id\_store, type, first\_name, last\_name, email, phone\_number)

    VALUES

        (7, 8, 'Platina', 'Stefania', 'Pop', 'Stefania.Pop51@gmail.com', '0756381460');

    INSERT INTO tClient

        (id, id\_store, type, first\_name, last\_name, email, phone\_number)

    VALUES

        (8, 5, 'Platina', 'Maria', 'Baciu', 'Maria.Baciu60@protonmail.com', '0790457601');

    INSERT INTO tClient

        (id, id\_store, type, first\_name, last\_name, email, phone\_number)

    VALUES

        (9, 8, 'Aur', 'Alexandru', 'Munteanu', 'Alexandru.Munteanu8@gmail.com', '0734109730');

    INSERT INTO tTransactionHistory

        (id, id\_client, sum, cr\_date)

    VALUES

        (0, 5, 76843, TIMESTAMP '2019-03-18 03:56:57');

    INSERT INTO tTransactionHistory

        (id, id\_client, sum, cr\_date)

    VALUES

        (1, 3, 4, TIMESTAMP '2017-01-18 06:06:09');

    INSERT INTO tTransactionHistory

        (id, id\_client, sum, cr\_date)

    VALUES

        (2, 4, 88830, TIMESTAMP '2019-09-08 12:12:11');

    INSERT INTO tTransactionHistory

        (id, id\_client, sum, cr\_date)

    VALUES

        (3, 1, 74, TIMESTAMP '2018-07-22 03:12:36');

    INSERT INTO tTransactionHistory

        (id, id\_client, sum, cr\_date)

    VALUES

        (4, 3, 8, TIMESTAMP '2017-07-29 19:52:55');

    INSERT INTO tTransactionHistory

        (id, id\_client, sum, cr\_date)

    VALUES

        (5, 2, 83, TIMESTAMP '2018-07-23 14:28:54');

    INSERT INTO tTransactionHistory

        (id, id\_client, sum, cr\_date)

    VALUES

        (6, 9, 680, TIMESTAMP '2019-12-14 10:49:25');

    INSERT INTO tTransactionHistory

        (id, id\_client, sum, cr\_date)

    VALUES

        (7, 5, 1062, TIMESTAMP '2019-04-19 08:52:44');

    INSERT INTO tTransactionHistory

        (id, id\_client, sum, cr\_date)

    VALUES

        (8, 8, 9, TIMESTAMP '2019-12-24 11:30:24');

    INSERT INTO tTransactionHistory

        (id, id\_client, sum, cr\_date)

    VALUES

        (9, 1, 3, TIMESTAMP '2018-07-19 06:44:19');

    INSERT INTO tProductStoreRelation

        (id, id\_store, id\_product)

    VALUES

        (0, 4, 8);

    INSERT INTO tProductStoreRelation

        (id, id\_store, id\_product)

    VALUES

        (1, 8, 7);

    INSERT INTO tProductStoreRelation

        (id, id\_store, id\_product)

    VALUES

        (2, 6, 9);

    INSERT INTO tProductStoreRelation

        (id, id\_store, id\_product)

    VALUES

        (3, 8, 8);

    INSERT INTO tProductStoreRelation

        (id, id\_store, id\_product)

    VALUES

        (4, 1, 7);

    INSERT INTO tProductStoreRelation

        (id, id\_store, id\_product)

    VALUES

        (5, 0, 0);

    INSERT INTO tProductStoreRelation

        (id, id\_store, id\_product)

    VALUES

        (6, 8, 6);

    INSERT INTO tProductStoreRelation

        (id, id\_store, id\_product)

    VALUES

        (7, 8, 5);

    INSERT INTO tProductStoreRelation

        (id, id\_store, id\_product)

    VALUES

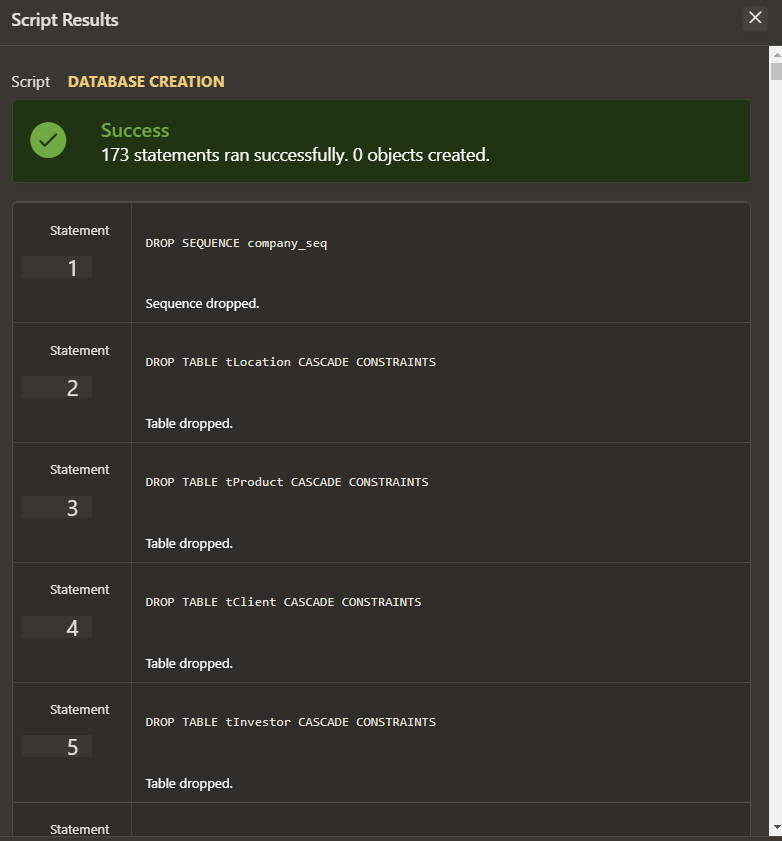
        (8, 9, 9);

    INSERT INTO tProductStoreRelation

        (id, id\_store, id\_product)

    VALUES

        (9, 6, 0);



REM   Script: EX6

REM   Procedura care ia o lista de salariati si calculeaza salariul total si returneaza salariul mediu.

DROP TYPE employee\_type FORCE;

DROP TYPE employee\_list\_type FORCE;

CREATE OR REPLACE TYPE employee\_type AS OBJECT (

  employee\_id INTEGER,

  name VARCHAR2(50),

  surname VARCHAR2(50),

  salary NUMBER);

/

CREATE OR REPLACE TYPE employee\_list\_type AS VARRAY(10) OF employee\_type;

/

DROP PROCEDURE calculate\_average\_salary;

CREATE PROCEDURE calculate\_average\_salary

AS

  v\_total\_salary NUMBER := 0;

  v\_average\_salary NUMBER;

  v\_employees employee\_list\_type;

BEGIN

  -- fill v\_employees

  SELECT employee\_type(id, name,surname, salary)

  BULK COLLECT INTO v\_employees

  FROM temployee;

  -- calculate total salary

  FOR i IN 1 .. v\_employees.COUNT

  LOOP

    v\_total\_salary := v\_total\_salary + v\_employees(i).salary;

  END LOOP;

  -- calculate average salary

  v\_average\_salary := v\_total\_salary / v\_employees.COUNT;

  -- print average salary

    DBMS\_OUTPUT.PUT\_LINE('The average salary is: ' || v\_average\_salary);

  RETURN;

END;

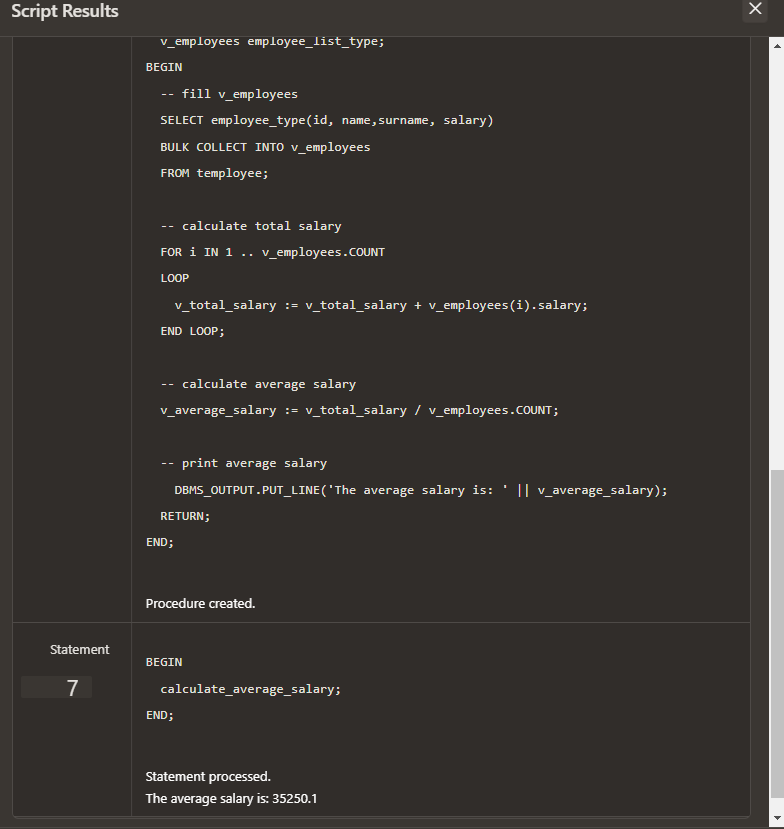
/

BEGIN

  calculate\_average\_salary;

END;

/



REM   Script: EX7

REM   Procedura care returneaza numele si salariul celor care au salariul mai mare decat parametrul acesteia.

DROP PROCEDURE get\_high\_paid\_sales\_employees;

CREATE OR REPLACE PROCEDURE get\_high\_paid\_sales\_employees (p\_salary IN NUMBER)

AS

  CURSOR employees\_cursor IS

    SELECT \* FROM temployee;

  CURSOR high\_paid\_cursor (c\_salary NUMBER) IS

    SELECT salary FROM temployee WHERE salary > c\_salary;

  l\_employee temployee%ROWTYPE;

  l\_salary NUMBER;

BEGIN

  OPEN employees\_cursor;

  LOOP

    FETCH employees\_cursor INTO l\_employee;

    EXIT WHEN employees\_cursor%NOTFOUND;

    OPEN high\_paid\_cursor(p\_salary);

    FETCH high\_paid\_cursor INTO l\_salary;

    CLOSE high\_paid\_cursor;

    IF l\_employee.salary > p\_salary THEN

      DBMS\_OUTPUT.PUT\_LINE(l\_employee.surname || ' ' || l\_employee.name || ' has a salary of ' || l\_employee.salary);

    END IF;

  END LOOP;

  CLOSE employees\_cursor;

END;

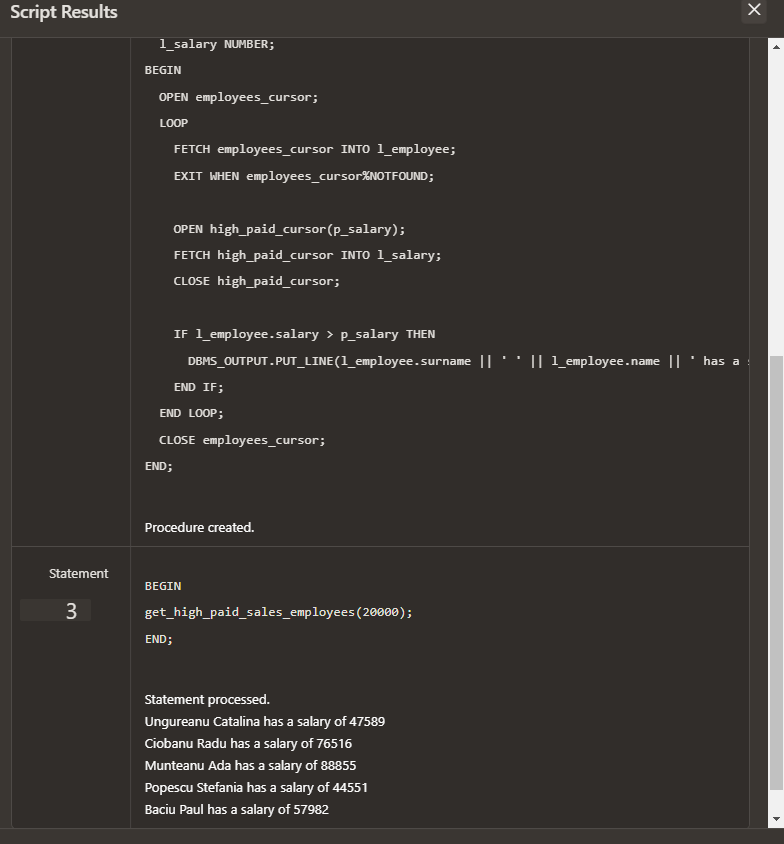
/

BEGIN

get\_high\_paid\_sales\_employees(20000);

END;

/



REM   Script: EX8

REM   Functie care calculeaza valoarea totala a pretului de baza a produselor care exista intr-un magazin folosind o comanda SQL cu 3 tabele . Totodata am definit 2 exceptii .

CREATE OR REPLACE FUNCTION get\_total\_product\_value\_by\_store\_id (store\_id IN INTEGER)

RETURN NUMBER

AS

  total\_value NUMBER;

  store\_not\_found EXCEPTION;

  no\_products\_sold EXCEPTION;

  cnt INTEGER;

BEGIN

  -- total value of products sold at the store

  SELECT SUM(p.price) INTO total\_value

  FROM tproduct p

  INNER JOIN tproductstorerelation r ON p.id = r.id\_product

  INNER JOIN tstore s ON r.id\_store = s.id

  WHERE s.id = store\_id;

  -- if id doesnt exists or if no products were sold, exception

  IF total\_value IS NULL THEN

    SELECT COUNT(\*) INTO cnt FROM tstore WHERE id = store\_id;

    IF cnt = 0 THEN

      RAISE store\_not\_found;

    ELSE

      RAISE no\_products\_sold;

    END IF;

  END IF;

  RETURN total\_value;

EXCEPTION

  WHEN store\_not\_found THEN

    RAISE\_APPLICATION\_ERROR(-20001, 'Store with ID ' || store\_id || ' not found');

  WHEN no\_products\_sold THEN

    RAISE\_APPLICATION\_ERROR(-20002, 'No products sold at store with ID ' || store\_id);

END;

/

BEGIN

  DBMS\_OUTPUT.PUT\_LINE(get\_total\_product\_value\_by\_store\_id(1));

END;

/

BEGIN

  DBMS\_OUTPUT.PUT\_LINE(get\_total\_product\_value\_by\_store\_id(2));

END;

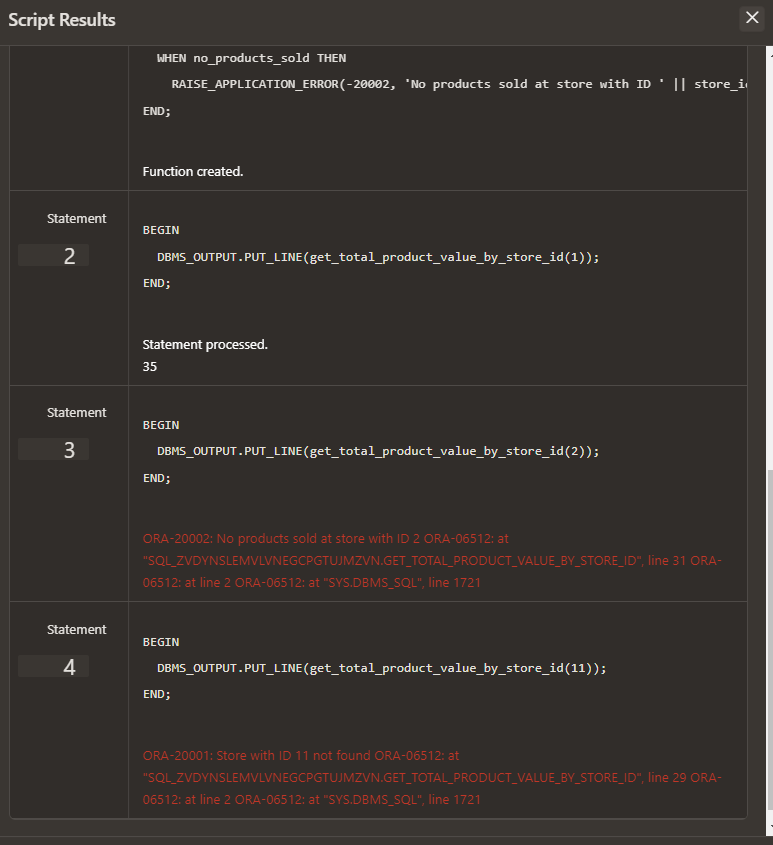
/

BEGIN

  DBMS\_OUTPUT.PUT\_LINE(get\_total\_product\_value\_by\_store\_id(11));

END;

/



REM   Script: EX9

REM   Procedura  care printeaza informatii despre un angajat avand ca parametru ID-ul acestuia (numele , emailul , numarul de telefon . titlul jobului al unui angajat , dar si numele CEO-ului si companiei pentru care lucreaza). Daca angajatul nu exista sau sunt gasiti mai multi -> exceptie.

CREATE OR REPLACE PROCEDURE get\_employee\_info (

  p\_employee\_id IN NUMBER,

  p\_employee\_name OUT VARCHAR2,

  p\_employee\_email OUT VARCHAR2,

  p\_employee\_phone OUT VARCHAR2,

  p\_job\_title OUT VARCHAR2,

  p\_company\_name OUT VARCHAR2,

  p\_company\_ceo OUT VARCHAR2

)

AS

BEGIN

  SELECT e.name, e.email, e.phone\_number, j.title, c.name, c.ceo

  INTO p\_employee\_name, p\_employee\_email, p\_employee\_phone, p\_job\_title, p\_company\_name, p\_company\_ceo

  FROM temployee e

  INNER JOIN tjobhistory jh ON e.id\_job\_history = jh.id

  INNER JOIN tjob j ON jh.id\_job = j.id

  INNER JOIN tstore s ON e.id\_store = s.id

  INNER JOIN tcompany c ON s.id\_company = c.id

  WHERE e.id = p\_employee\_id;

EXCEPTION

  WHEN NO\_DATA\_FOUND THEN

    p\_employee\_name := NULL;

    p\_employee\_email := NULL;

    p\_employee\_phone := NULL;

    p\_job\_title := NULL;

    p\_company\_name := NULL;

    p\_company\_ceo := NULL;

  WHEN TOO\_MANY\_ROWS THEN

    RAISE\_APPLICATION\_ERROR(-20001, 'Multiple employees found with the same ID');

END;

/

DECLARE

l\_name VARCHAR2(100);

l\_email VARCHAR2(100);

l\_phone VARCHAR2(100);

l\_job\_title VARCHAR2(100);

l\_company\_name VARCHAR2(100);

l\_company\_ceo VARCHAR2(100);

BEGIN

  FOR i in 0..10 LOOP

  get\_employee\_info(i, l\_name, l\_email, l\_phone, l\_job\_title, l\_company\_name, l\_company\_ceo);

  IF l\_name IS NULL THEN

    DBMS\_OUTPUT.PUT\_LINE('Employee not found'||CHR(10));

  ELSE

    DBMS\_OUTPUT.PUT\_LINE('Name: ' || l\_name);

    DBMS\_OUTPUT.PUT\_LINE('Email: ' || l\_email);

    DBMS\_OUTPUT.PUT\_LINE('Phone: ' || l\_phone);

    DBMS\_OUTPUT.PUT\_LINE('Job title: ' || l\_job\_title);

  DBMS\_OUTPUT.PUT\_LINE('Company name: ' || l\_company\_name);

  DBMS\_OUTPUT.PUT\_LINE('Company CEO: ' || l\_company\_ceo||CHR(10));

  END IF;

  END LOOP;

EXCEPTION

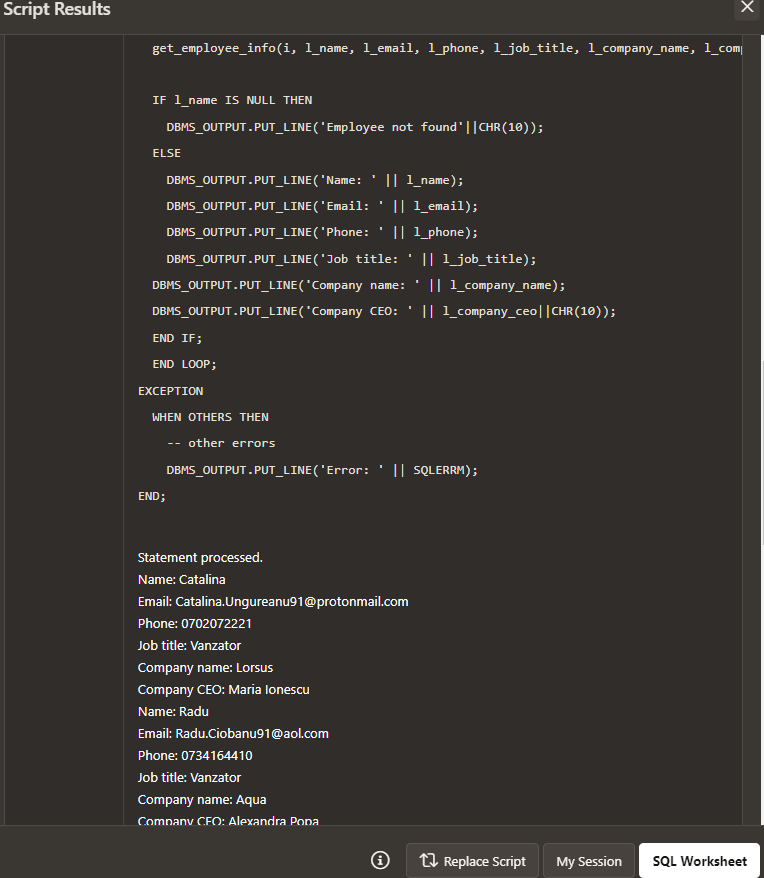
  WHEN OTHERS THEN

    -- other errors

    DBMS\_OUTPUT.PUT\_LINE('Error: ' || SQLERRM);

END;

/



REM   Script: EX10

REM   Trigger LMD la nivel de comanda , de asemenea declansat

CREATE OR REPLACE TRIGGER prevent\_salary\_update

  BEFORE UPDATE OF salary ON temployee

  FOR EACH ROW

BEGIN

  -- error if the salary is being modified

  RAISE\_APPLICATION\_ERROR(-20001, 'Cannot update salary');

END;

/

BEGIN

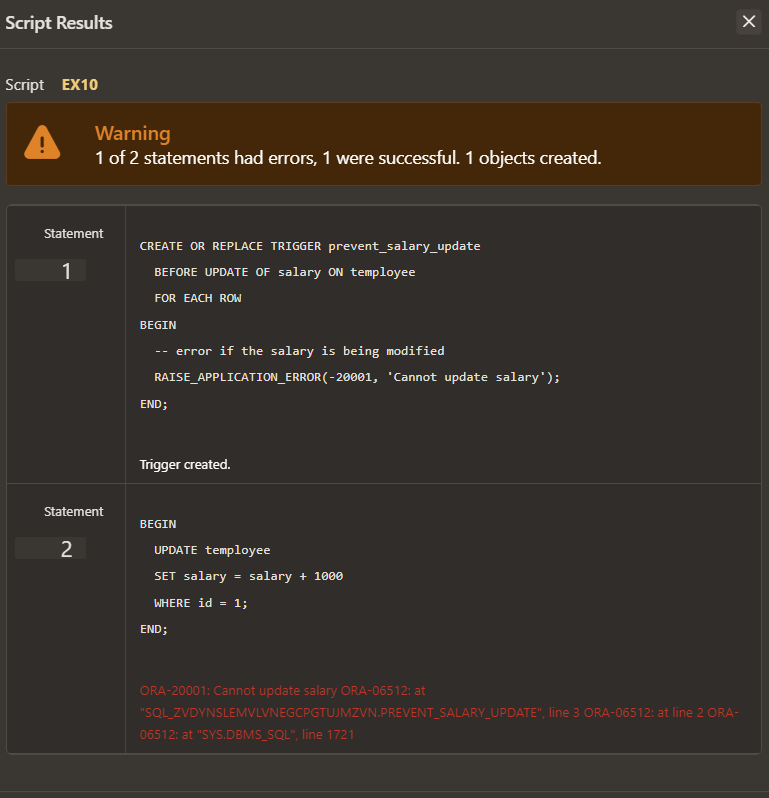
  UPDATE temployee

  SET salary = salary + 1000

  WHERE id = 1;

END;

/



REM   Script: EX11

REM   Trigger LMD la nivel de linie , declansat

CREATE OR REPLACE TRIGGER prevent\_name\_update

  BEFORE UPDATE ON temployee

  FOR EACH ROW

BEGIN

  -- Raise an error if the name is being modified

  IF :NEW.name <> :OLD.name THEN

    RAISE\_APPLICATION\_ERROR(-20001, 'Cannot update name');

  END IF;

END;

/

BEGIN

  UPDATE temployee

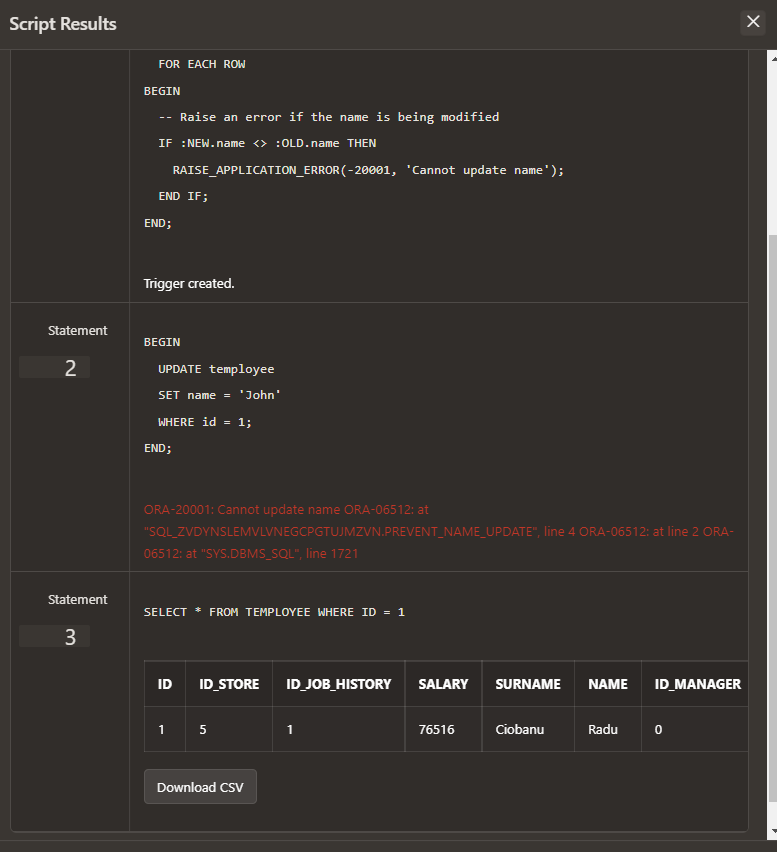
  SET name = 'John'

  WHERE id = 1;

END;

/

SELECT \* FROM TEMPLOYEE WHERE ID = 1;



REM   Script: EX12

REM   Trigger LDD , declansat

DROP TABLE log\_table;

CREATE TABLE log\_table (

  id          NUMBER(10)   PRIMARY KEY,

  operation   VARCHAR2(10) NOT NULL,

  name        VARCHAR2(50) NOT NULL,

  price       NUMBER(10,2) NOT NULL

);

CREATE OR REPLACE TRIGGER log\_product\_changes

  AFTER INSERT OR UPDATE OR DELETE ON tproduct

  FOR EACH ROW

BEGIN

  -- insert, update, or delete

  IF INSERTING THEN

    INSERT INTO log\_table (operation, id, name, price) VALUES ('INSERT', :NEW.id, :NEW.name, :NEW.price);

  ELSIF UPDATING THEN

    INSERT INTO log\_table (operation, id, name, price) VALUES ('UPDATE', :NEW.id, :NEW.name, :NEW.price);

  ELSIF DELETING THEN

    INSERT INTO log\_table (operation, id, name, price) VALUES ('DELETE', :OLD.id, :OLD.name, :OLD.price);

  END IF;

END;

/

INSERT INTO tproduct (id, name, price) VALUES (123, 'Widget', 10.99);

SELECT \* FROM log\_table;

SELECT \* FROM tproduct;

