

Proiect Baze de Date 2

344C3 Tudorache Bogdan-Mihai

Descrierea temei

Proiect la Baze de Date 2 2022-2023. Am ales sa proiectez o baza de date pentru o librerie.

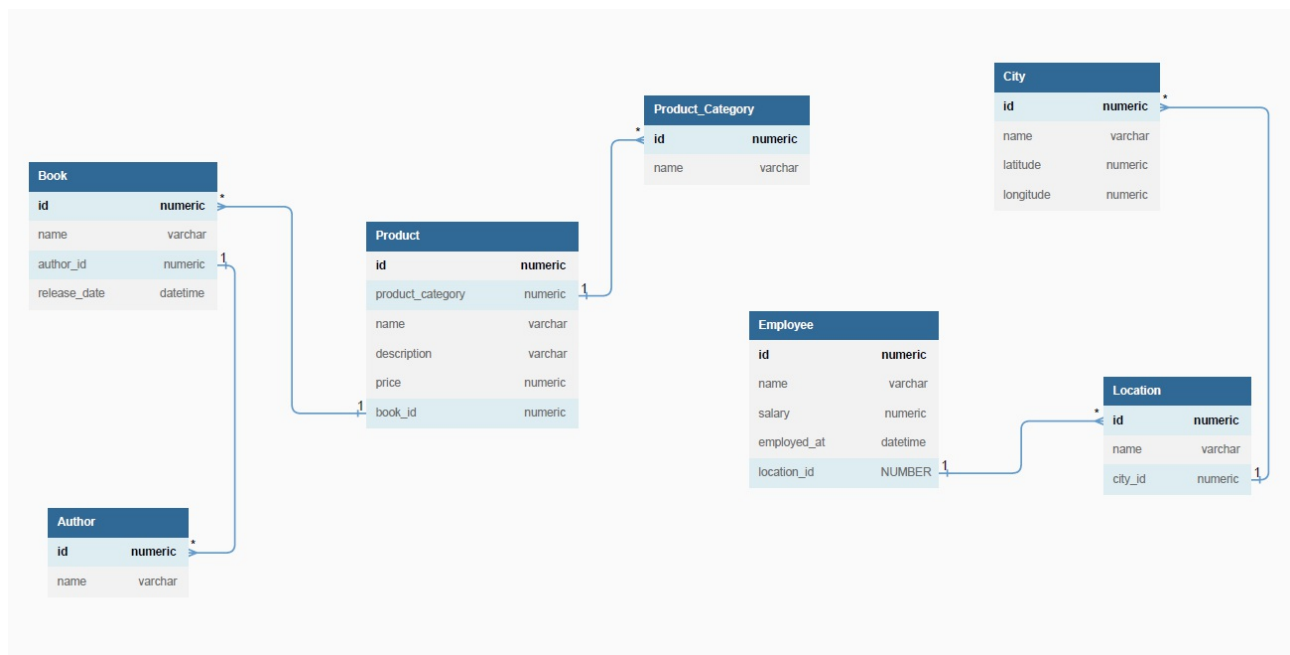
Descrierea bazei de date

Pentru proiectarea bazei de date pentru o librerie, am ales sa includ un minim de 7 tabele.

Tabelul principal este Location. Libraria are mai multe locatii de-a lungul tarii, asa ca exista mai multe intrari in tabela. Exista si o cheie straina catre un tabel City, pentru ca o locatie se afla neaparat intr-un oras. Tabelul Employee este pentru angajatii libreriei, iar fiecare angajat este asociat unei locatii anume.

Locatiile vand produse, iar acestea au asociate un tabel sub numele de Product. Product contine o cheie straina catre un tabel Product_Category, care reprezinta tipul produsului, dintr-un pool restrans de optiuni (Carte, Film, Decoratii etc...). In cazul in care Produsul este de tipul Carte, acesta are si o cheie straina catre un tabel Book, care contine informatii legate de aceea carte. Tabelul Book are o referinta straina catre un tabel Author, care are informatii despre autor.

Diagrama bazei de date



Structura tabelor

Structura tabelor este urmatoarea:

- City
 - id NUMBER(6) PRIMARY KEY NOT NULL
 - name VARCHAR2(25) NOT NULL
 - latitude REAL
 - longitude REAL
- Location
 - id NUMBER(6) PRIMARY KEY NOT NULL
 - name VARCHAR2(40) NOT NULL
 - city_id NUMBER(6)
- Employee
 - id NUMBER(6) PRIMARY KEY NOT NULL,
 - name VARCHAR2(40) NOT NULL

- salary NUMBER(10)
 - employed_at DATE
 - location_id NUMBER NOT NULL
- Product_Category
 - id NUMBER(6) PRIMARY KEY NOT NULL
 - name VARCHAR2(25) UNIQUE NOT NULL
- Author
 - id NUMBER(6) PRIMARY KEY NOT NULL
 - name VARCHAR2(40) NOT NULL
- Book
 - id NUMBER(6) PRIMARY KEY NOT NULL
 - name VARCHAR2(60) NOT NULL
 - author_id NUMBER(6) NOT NULL
 - release_date DATE NOT NULL
- Product
 - id NUMBER(10) PRIMARY KEY NOT NULL
 - product_category NUMBER(10) NOT NULL
 - name VARCHAR2(40) NOT NULL
 - description VARCHAR2(200)
 - price NUMBER(10) NOT NULL
 - book_id NUMBER(10)

Descrierea constrângerilor de integritate

În afara de constrângerile de cheie primare și de câmpuri not null descrise în structura de mai sus, mai există următoarele constrângeri de foreign key:

- Location conține o cheie străină (city_id) către City (id);
- Employee conține o cheie străină (location_id) către Location (id);
- Book conține o cheie străină (author_id) către Author (id);
- Product conține o cheie străină (product_category) către Product_Category(id);
- Product conține o cheie străină (book_id) către Book(id);

Descrierea procedurilor

Procedurile pe care le-am implementat modifică un parametru cursor de tip OUT printr-un query care accesează baza de date. Acestea extrag următoarele informații:

1. Locațiile din București și din orașul cu cea mai mică latitudine

```
CREATE OR REPLACE PROCEDURE report1(P_RC OUT SYS_REFCURSOR)
AS
BEGIN
    OPEN P_RC FOR SELECT *
    FROM City C
    JOIN Location L ON C.id = L.city_id
    WHERE C.name = 'Bucuresti' OR C.latitude = (SELECT MIN(latitude)
                                                FROM CITY);
END;
```

2. Numarul angajatilor din fiecare locatie care au salariu mai mare decat salariul mediu al angajatilor din Bucuresti

```

CREATE OR REPLACE PROCEDURE report2(P_RC OUT SYS_REFCURSOR)
AS
BEGIN
    OPEN P_RC FOR SELECT L.name, COUNT(*)
    FROM City C
    JOIN Location L ON C.id = L.city_id
    JOIN Employee E ON L.id = E.location_id
    WHERE E.salary > (SELECT AVG(salary)
                     FROM City C1
                     JOIN Location L1 ON C1.id = L1.city_id
                     JOIN Employee E1 ON L1.id = E1.location_id
                     WHERE C1.name = 'Bucuresti')

    GROUP BY L.name;
END;

```

3. Numele autorului cu cele mai multe carti

```

CREATE OR REPLACE PROCEDURE report3(P_RC OUT SYS_REFCURSOR)
AS
BEGIN
    OPEN P_RC FOR SELECT DISTINCT(A.name)
    FROM Author A
    JOIN Book B ON A.id = B.author_id
    JOIN Product P ON B.id = P.book_id
    WHERE A.id = (SELECT B1.author_id
                 FROM Book B1
                 GROUP BY B1.author_id
                 HAVING COUNT(*) = (SELECT MAX(COUNT(*))
                                    FROM Book B2
                                    GROUP BY B2.author_id));
END;

```

Conexiunea cu baza de date din Power BI

Am ales să folosesc Power BI pentru generarea rapoartelor. Conexiunea se face în felul următor: după ce aleg sursa de date (Oracle DB) și adaug totii parametri necesari (adresa db, nume, parola), pasesc următorul query inițial pentru a rula o procedură:

```

DECLARE P_RC SYS_REFCURSOR;

BEGIN
prodecure(P_RC);
DBMS_SQL.RETURN_RESULT(P_RC);
END;

```

Cursorul trebuie declarat pentru că alta metodă de a apela proceduri Oracle nu am găsit. Din aplicație procesul arată în felul următor:

×

Oracle database

Server

localhost:1521

Data Connectivity mode ⓘ

☒ Import
 ☐ DirectQuery

⌄ Advanced options

Command timeout in minutes (optional)

SQL statement (optional)

```

DECLARE P_RC SYS_REFCURSOR;

BEGIN
prodecure(P_RC);
DBMS_SQL.RETURN_RESULT(P_RC);
END;

```

☒ Include relationship columns

☒ Navigate using full hierarchy

OK

Cancel

Dupa care am generat rapoartele din interfata grafica.

Rapoarte

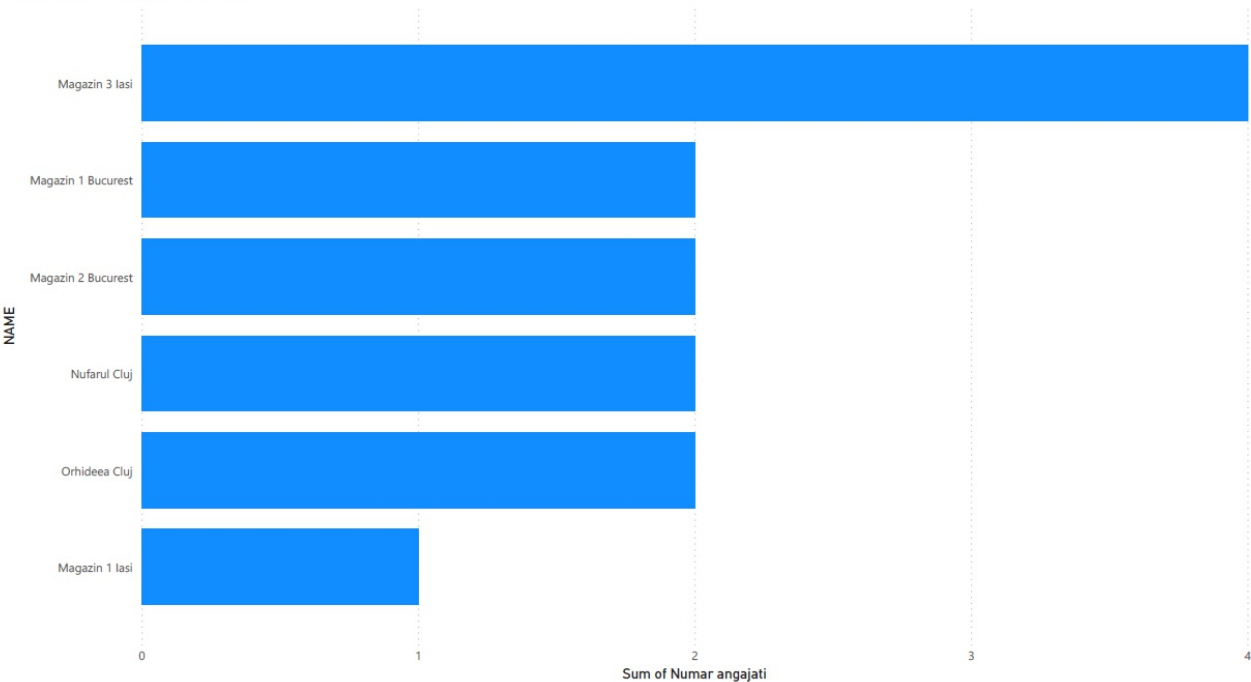
Graficele generate sunt urmatoarele:

Raport 1

Bucuresti	Magazin 1 Bucurest	26.08	44.53
Oras	Locatie	Longitudine	Latitudine
Bucuresti	Magazin 2 Bucurest	26.08	44.53
Oras	Locatie	Longitudine	Latitudine
Cluj	Nufarul Cluj	26.05	42.53
Oras	Locatie	Longitudine	Latitudine
Cluj	Orhideea Cluj	26.05	42.53
Oras	Locatie	Longitudine	Latitudine

Raport 2

Sum of Numar angajati by NAME



Raport 3

Numele autorului cu cele mai multe carti
Cormac McCarthy

Concluzii

N-am mai proiectat baze de date SQL folosind exclusiv query-uri native, si nici nu am rulat direct proceduri native, doar prin intermediul unui ORM, deci in concluzie mi s-a parut o experienta interesanta.