

Tema 4 - Laborator

Aelenei Alex

Cuprins

| | | |
|---|---------------|---|
| 1 | Introducere | 2 |
| 2 | Exercițiul 10 | 2 |

1 Introducere

Diagrama conceptuală și diagrama entitate-relație utilizate în această temă sunt mai jos. Datele de test folosite sunt cele utilizate în tema 1 și cele de la finalul acestui document.

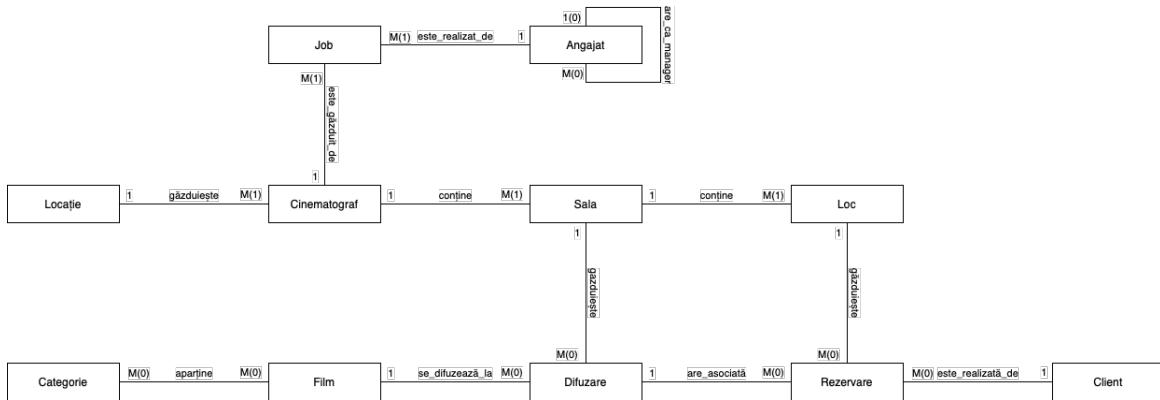


Figura 1: Diagrama conceptuală

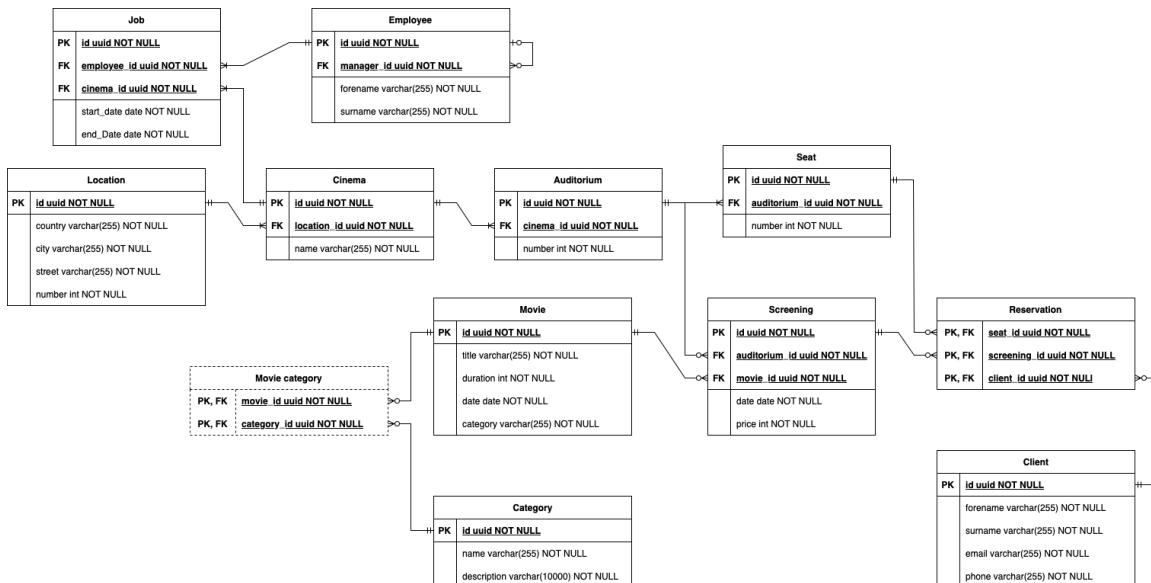


Figura 2: Diagrama entitate-relație

Valorile utilizate pentru a testa query-urile sunt cele citate în Tema 2. Aceste query-uri au fost rulate pe o instanță de Oracle Database 19, bazată pe imaginea oficială de la Oracle, dar cu un build pentru MacOS ARM local.

2 Exercițiul 10

Problema 10 din *Laborator3_PLSQL.pdf* spune: *Pentru fiecare dintre departamentele 10, 20, 30 și 40, obțineți numele precum și lista numelor angajaților care își desfășoară activitatea în cadrul acestora. Rezolvați problema folosind: a. cele trei tipuri de cursoare studiate; b. expresii cursor. Adaptând cerința la structura utilizată vom rezolva următoarea cerință: Pentru fiecare dintre filemele 1, 2 și 3, obțineți numele filmului și datele difuzărilor acestora.*

```

01 | SET SERVEROUTPUT ON
02 |
03 | DECLARE
04 |     MOVIE_TITLE      MOVIE.TITLE%TYPE;
05 |     MOVIE_ID        MOVIE.MOVIE_ID%TYPE;
06 |     MOVIE_SCREENING SCREENING%ROWTYPE;
07 |     DATES_STRING    VARCHAR(1000);
08 |     CURSOR MC IS (
09 |         SELECT
10 |             M.MOVIE_ID ,
11 |             M.TITLE
12 |         FROM
13 |             MOVIE M
14 |         WHERE
15 |             M.MOVIE_ID IN (1, 2, 3)
16 |     );
17 |     CURSOR SC(
18 |         SCREENING_MOVIE_ID MOVIE.MOVIE_ID%TYPE
19 |     ) IS (
20 |         SELECT
21 |             *
22 |         FROM
23 |             SCREENING S
24 |         WHERE
25 |             S.MOVIE_ID = SCREENING_MOVIE_ID
26 |     );
27 | BEGIN
28 |     OPEN MC;
29 |     LOOP
30 |         FETCH MC INTO MOVIE_ID , MOVIE_TITLE;
31 |         EXIT WHEN MC%NOTFOUND;
32 |         DATES_STRING := '';
33 |         OPEN SC(MOVIE_ID);
34 |         LOOP
35 |             FETCH SC INTO MOVIE_SCREENING;
36 |             EXIT WHEN SC%NOTFOUND;
37 |             DATES_STRING := DATES_STRING
38 |                 || TO_CHAR(MOVIE_SCREENING.DATE , 'dd/mm/yyyy')
39 |                 || ' ';
40 |         END LOOP;
41 |
42 |         CLOSE SC;
43 |         IF DATES_STRING = '' THEN
44 |             DBMS_OUTPUT.PUT_LINE('No screenings for the movie '
45 |                                 || MOVIE_TITLE
46 |                                 || '.');
47 |         ELSE
48 |             DBMS_OUTPUT.PUT_LINE('Screenings for the movie '
49 |                                 || MOVIE_TITLE
50 |                                 || ' are on the following dates:');
51 |             DBMS_OUTPUT.PUT_LINE(DATES_STRING);
52 |         END IF;
53 |     END LOOP;
54 |
55 |     CLOSE MC;
56 | END;

```

Listing 1: Query cursor clasic

```

01 | SET SERVEROUTPUT ON
02 |
03 | DECLARE
04 |   DATES_STRING VARCHAR(1000);
05 |   CURSOR MC IS (
06 |     SELECT
07 |       M.MOVIE_ID,
08 |       M.TITLE
09 |     FROM
10 |       MOVIE M
11 |     WHERE
12 |       M.MOVIE_ID IN (1, 2, 3)
13 |   );
14 |   CURSOR SC(
15 |     SCREENING_MOVIE_ID MOVIE.MOVIE_ID%TYPE
16 |   ) IS (
17 |     SELECT
18 |       *
19 |     FROM
20 |       SCREENING S
21 |     WHERE
22 |       S.MOVIE_ID = SCREENING_MOVIE_ID
23 |   );
24 | BEGIN
25 |   FOR MOVIE_ENTRY IN MC LOOP
26 |     DATES_STRING := '';
27 |     FOR SCREENING_ENTRY IN SC(MOVIE_ENTRY.MOVIE_ID) LOOP
28 |       DATES_STRING := DATES_STRING
29 |         || TO_CHAR(SCREENING_ENTRY.DATE, 'dd/mm/yyyy')
30 |         || ' ';
31 |     END LOOP;
32 |
33 |     IF DATES_STRING = '' THEN
34 |       DBMS_OUTPUT.PUT_LINE('No screenings for the movie '
35 |                           || MOVIE_ENTRY.TITLE
36 |                           || '.');
37 |     ELSE
38 |       DBMS_OUTPUT.PUT_LINE('Screenings for the movie '
39 |                           || MOVIE_ENTRY.TITLE
40 |                           || ' are on the following dates:');
41 |       DBMS_OUTPUT.PUT_LINE(DATES_STRING);
42 |     END IF;
43 |   END LOOP;
44 | END;

```

Listing 2: Query ciclu cursor

```

01 | SET SERVEROUTPUT ON
02 |
03 | DECLARE
04 |   DATES_STRING VARCHAR(1000);
05 | BEGIN
06 |   FOR MOVIE_ENTRY IN (
07 |     SELECT
08 |       M.MOVIE_ID,
09 |       M.TITLE
10 |     FROM
11 |       MOVIE M
12 |     WHERE
13 |       M.MOVIE_ID IN (1, 2, 3)
14 |   ) LOOP
15 |     DATES_STRING := '';
16 |     FOR SCREENING_ENTRY IN (
17 |       SELECT
18 |         *
19 |       FROM
20 |         SCREENING S
21 |       WHERE
22 |         S.MOVIE_ID = MOVIE_ENTRY.MOVIE_ID
23 |     ) LOOP
24 |       DATES_STRING := DATES_STRING
25 |                     || TO_CHAR(SCREENING_ENTRY.DATE, 'dd/mm/yyyy')
26 |                     || ' ';
27 |     END LOOP;
28 |
29 |     IF DATES_STRING = '' THEN
30 |       DBMS_OUTPUT.PUT_LINE('No screenings for the movie '
31 |                           || MOVIE_ENTRY.TITLE
32 |                           || '.');
33 |     ELSE
34 |       DBMS_OUTPUT.PUT_LINE('Screenings for the movie '
35 |                           || MOVIE_ENTRY.TITLE
36 |                           || ' are on the following dates:');
37 |       DBMS_OUTPUT.PUT_LINE(DATES_STRING);
38 |     END IF;
39 |   END LOOP;
40 | END;

```

Listing 3: Query ciclu cursor cu subcereri

```

01 | SET SERVEROUTPUT ON
02 |
03 | DECLARE
04 |     TYPE REFCURSOR IS
05 |         REF CURSOR;
06 |         MOVIE_ID      MOVIE.MOVIE_ID%TYPE;
07 |         MOVIE_TITLE   MOVIE.TITLE%TYPE;
08 |         MOVIE_SCREENING SCREENING%ROWTYPE;
09 |         DATES_STRING  VARCHAR(1000);
10 |         CURSOR MC IS (
11 |             SELECT
12 |                 M.MOVIE_ID ,
13 |                 M.TITLE ,
14 |                 CURSOR (
15 |                     SELECT
16 |                         *
17 |                     FROM
18 |                         SCREENING S
19 |                     WHERE
20 |                         S.MOVIE_ID = M.MOVIE_ID
21 |                 )
22 |             )
23 |             FROM
24 |             MOVIE M
25 |             WHERE
26 |                 M.MOVIE_ID IN (1, 2, 3)
27 | );
28 | SC          REFCURSOR;
29 | BEGIN
30 |     OPEN MC;
31 |     LOOP
32 |         FETCH MC INTO MOVIE_ID, MOVIE_TITLE, SC;
33 |         EXIT WHEN MC%NOTFOUND;
34 |         DATES_STRING := '';
35 |         LOOP
36 |             FETCH SC INTO MOVIE_SCREENING;
37 |             EXIT WHEN SC%NOTFOUND;
38 |             DATES_STRING := DATES_STRING
39 |                         || TO_CHAR(MOVIE_SCREENING.DATE, 'dd/mm/yyyy')
40 |                         || ' ';
41 |         END LOOP;
42 |         IF DATES_STRING = '' THEN
43 |             DBMS_OUTPUT.PUT_LINE('No screenings for the movie '
44 |                                 || MOVIE_TITLE
45 |                                 || '.');
46 |         ELSE
47 |             DBMS_OUTPUT.PUT_LINE('Screenings for the movie '
48 |                                 || MOVIE_TITLE
49 |                                 || ' are on the following dates:');
50 |             DBMS_OUTPUT.PUT_LINE(DATES_STRING);
51 |         END IF;
52 |     END LOOP;
53 |
54 |     CLOSE MC;
55 | END;

```

Listing 4: Query expresii cursor

The screenshot shows the Oracle SQL Developer interface on a Mac OS X desktop. The main window displays a PL/SQL procedure named 'Screenings'. The code uses a cursor to fetch data from the 'SCREENING' table and prints the results to the screen using DBMS_OUTPUT.PUT_LINE. The procedure handles cases where no screenings are found or multiple screenings are present for a movie on specific dates.

```

    PROCEDURE Screenings IS
        SC SCREENING%TYPE;
        MC MOVIE%TYPE;
        DATES_STRING VARCHAR2(100);
    BEGIN
        FOR MC IN MOVIE_ID%TYPE LOOP
            FETCH SC INTO MOVIE_SCREENING;
            EXIT WHEN MC%NOTFOUND;
            DATES_STRING := '';
            OPEN SC(MOVIE_ID);
            LOOP
                FETCH SC INTO MOVIE_SCREENING;
                EXIT WHEN SC%NOTFOUND;
                DATES_STRING := DATES_STRING
                    || TO_CHAR(MOVIE_SCREENING.DATE, 'dd/mm/yyyy')
                    || ',';
            END LOOP;
            CLOSE SC;
            IF DATES_STRING = '' THEN
                DBMS_OUTPUT.PUT_LINE('No screenings for the movie '
                    || MOVIE_TITLE
                    || ',');
            ELSE
                DBMS_OUTPUT.PUT_LINE('Screenings for the movie '
                    || MOVIE_TITLE
                    || ' are on the following dates:');
                DBMS_OUTPUT.PUT_LINE(DATES_STRING);
            END IF;
        END LOOP;
        CLOSE MC;
    END;

```

The 'Query Result' tab shows the output of the procedure, listing various screening dates for different movies like 'Palm Springs' and 'Aliens'.

Figura 3: Rezultat query cursor clasic

This screenshot shows the same Oracle SQL Developer environment, but the code has been modified to use a cursor loop instead of a traditional cursor loop. The logic remains the same, fetching data from the 'SCREENING' table and printing the results to the screen using DBMS_OUTPUT.PUT_LINE.

```

    PROCEDURE Screenings IS
        SC SCREENING%TYPE;
        MC MOVIE%TYPE;
        DATES_STRING VARCHAR2(100);
    BEGIN
        FOR MOVIE_ENTRY IN MC LOOP
            DATES_STRING := '';
            FOR SCREENING_ENTRY IN SC(MOVIE_ENTRY.MOVIE_ID) LOOP
                DATES_STRING := DATES_STRING
                    || TO_CHAR(SCREENING_ENTRY.DATE, 'dd/mm/yyyy')
                    || ',';
            END LOOP;
            IF DATES_STRING = '' THEN
                DBMS_OUTPUT.PUT_LINE('No screenings for the movie '
                    || MOVIE_ENTRY.TITLE
                    || ',');
            ELSE
                DBMS_OUTPUT.PUT_LINE('Screenings for the movie '
                    || MOVIE_ENTRY.TITLE
                    || ' are on the following dates:');
                DBMS_OUTPUT.PUT_LINE(DATES_STRING);
            END IF;
        END LOOP;
        CLOSE MC;
    END;

```

The 'Query Result' tab shows the output of the modified procedure, which is identical to the original output.

Figura 4: Rezultat query ciclu cursor

```

    DATES_STRING := '';
    FOR SCREENING_ENTRY IN (
        SELECT *
        FROM SCREENING_S
        WHERE S.MOVIE_ID = MOVIE_ENTRY.MOVIE_ID
    ) LOOP
        DATES_STRING := DATES_STRING
            || TO_CHAR(SCREENING_ENTRY.DATE, 'dd/mm/yyyy')
            || '|';
    END LOOP;

    IF DATES_STRING = '' THEN
        DBMS_OUTPUT.PUT_LINE('No screenings for the movie '
            || MOVIE_ENTRY.TITLE
            || '|');
    ELSE
        DBMS_OUTPUT.PUT_LINE('Screenings for the movie '
            || MOVIE_ENTRY.TITLE
            || '| are on the following dates:');
        DBMS_OUTPUT.PUT_LINE(DATES_STRING);
    END IF;
END LOOP;

```

Screenings for the movie Palm Springs are on the following dates:
20/09/2025 11/09/2025 15/09/2025 07/09/2025 09/09/2025 27/09/2025 27/09/2025 01/09/2025 27/09/2025 25/09/2025 24/09/2025 19/09/2025 29/09/2025
Screenings for the movie Alien Is Is available on the following dates:
21/09/2025 01/09/2025 21/09/2025 18/09/2025 10/09/2025 16/09/2025 28/09/2025 21/09/2025 13/09/2025 17/09/2025 13/09/2025 23/09/2025 08/09/2025 14/09/2025 23/09/2025
Screenings for the movie Aliens are on the following dates:
18/09/2025 09/09/2025 06/09/2025 05/09/2025 16/09/2025 03/09/2025 09/09/2025 29/09/2025 17/09/2025 13/09/2025 18/09/2025 21/09/2025 08/09/2025

PL/SQL procedure successfully completed.

Figura 5: Rezultat query ciclu cursor cu subcereri

```

    FETCH MC INTO MOVIE_ID, MOVIE_TITLE, SC;
    EXIT WHEN MC%NOTFOUND;
    DATES_STRING := '';
    LOOP
        FETCH SC INTO MOVIE_SCREENING;
        EXIT WHEN SC%NOTFOUND;
        DATES_STRING := DATES_STRING
            || TO_CHAR(MOVIE_SCREENING.DATE, 'dd/mm/yyyy')
            || '|';
    END LOOP;

    IF DATES_STRING = '' THEN
        DBMS_OUTPUT.PUT_LINE('No screenings for the movie '
            || MOVIE_TITLE
            || '|');
    ELSE
        DBMS_OUTPUT.PUT_LINE('Screenings for the movie '
            || MOVIE_TITLE
            || '| are on the following dates:');
        DBMS_OUTPUT.PUT_LINE(DATES_STRING);
    END IF;
END LOOP;
CLOSE MC;

```

Screenings for the movie Palm Springs are on the following dates:
20/09/2025 11/09/2025 15/09/2025 07/09/2025 09/09/2025 27/09/2025 27/09/2025 01/09/2025 27/09/2025 25/09/2025 24/09/2025 19/09/2025 29/09/2025
Screenings for the movie Alien Is Is available on the following dates:
21/09/2025 01/09/2025 21/09/2025 18/09/2025 10/09/2025 16/09/2025 28/09/2025 21/09/2025 13/09/2025 17/09/2025 13/09/2025 23/09/2025 08/09/2025 14/09/2025 23/09/2025
Screenings for the movie Aliens are on the following dates:
18/09/2025 09/09/2025 06/09/2025 05/09/2025 16/09/2025 03/09/2025 09/09/2025 29/09/2025 17/09/2025 13/09/2025 18/09/2025 21/09/2025 08/09/2025

PL/SQL procedure successfully completed.

Figura 6: Rezultat query expresii cursor