

# Management-ul unui magazin de muzică

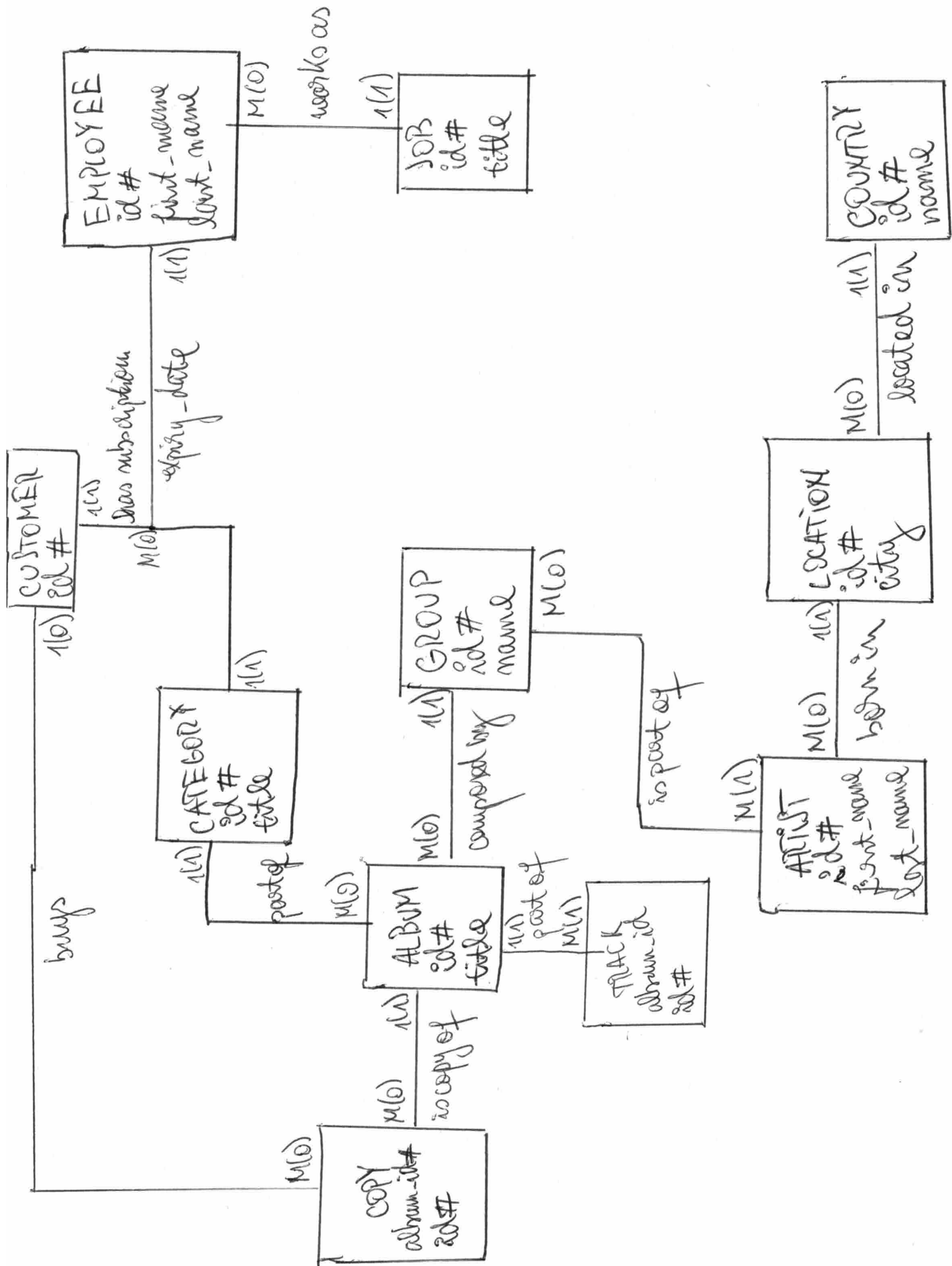
Nicula Ionuț-Cosmin

*mail:* `ionut.nicula@s.unibuc.ro`

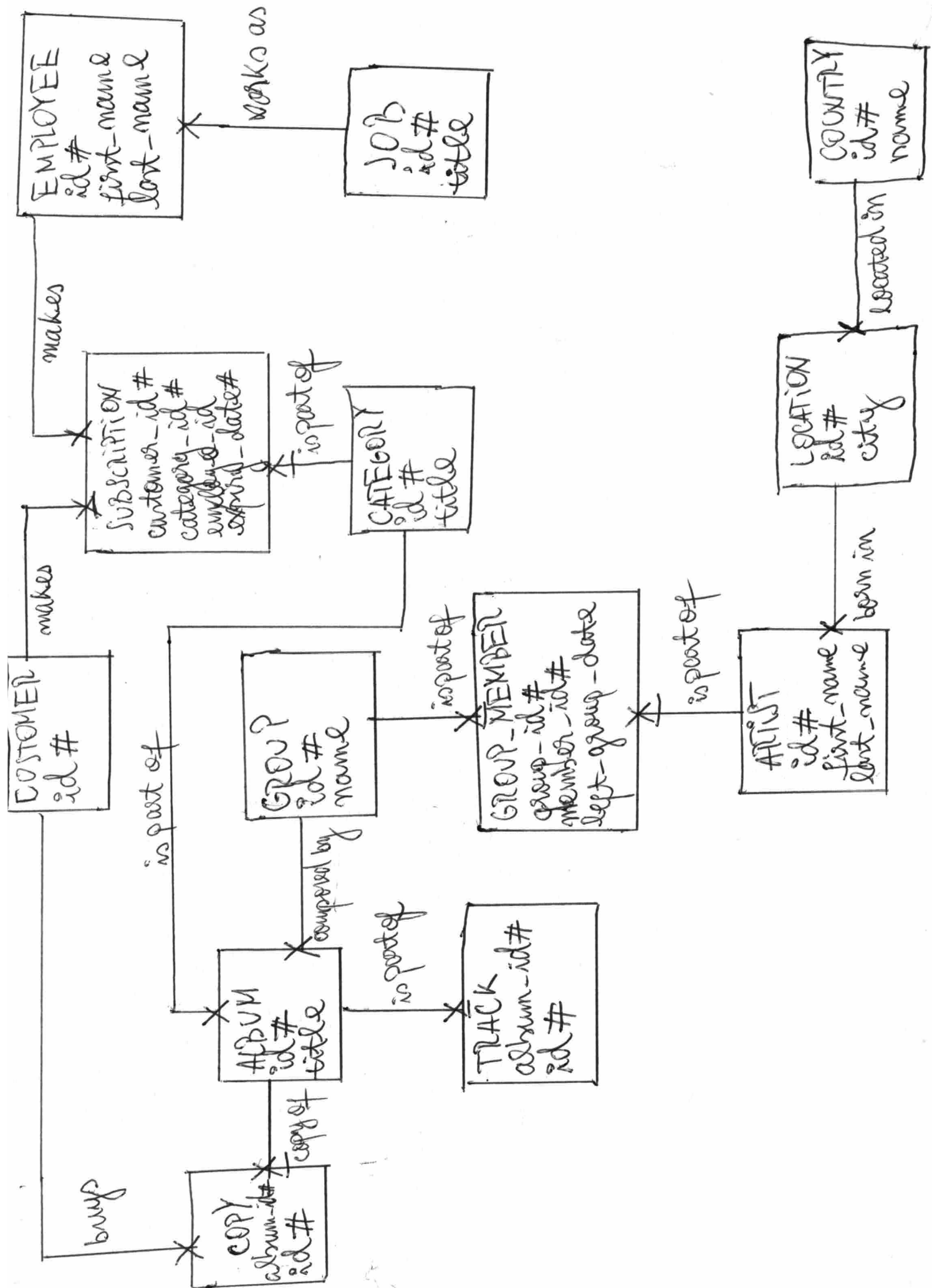
*surse:* `github.com/niculaionut/proiect-BD`

15 mai 2021

- 1 Descrierea modelului și a utilității acestuia
- 2 Constrângerile impuse asupra modelului
- 3 Descrierea entităților
- 4 Descrierea relațiilor
- 5 Descrierea atributelor
- 6 Diagrama entitate-relație (pagina următoare)



## 7 Diagrama conceptuală (pagina următoare)



## 8 Schemele relaționale

```
01 | COUNTRY (id#,
02 |         name)
03 |
04 | LOCATION (id#,
05 |         city,
06 |         country_id#)
07 |
08 | ARTIST (id#,
09 |        first_name,
10 |        last_name,
11 |        date_birth,
12 |        date_death,
13 |        location_id#)
14 |
15 | GROUP (id#,
16 |        name)
17 |
18 | GROUP_MEMBER (group_id#,
19 |              member_id#,
20 |              left_group_date)
21 |
22 | CATEGORY (id#,
23 |          title)
24 |
25 | ALBUM (id#,
26 |        title#,
27 |        release_date,
28 |        category_id#,
29 |        group_id#)
30 |
31 | TRACK (album_id#,
32 |        id#,
33 |        title,
34 |        length)
35 |
36 | JOB (id#,
37 |      title,
38 |      min_salary,
39 |      max_salary)
40 |
41 | EMPLOYEE (id#,
42 |          first_name,
43 |          last_name,
44 |          salary,
45 |          hire_date,
46 |          job_id#)
47 |
48 | CUSTOMER (id#,
49 |          first_name,
50 |          last_name,
51 |          email)
52 |
53 | COPY (album_id#,
54 |       id#,
55 |       serial_number,
56 |       price,
57 |       customer_id#)
58 |
59 | SUBSCRIPTION (customer_id#,
60 |              category_id#,
61 |              employee_id#,
```

## 9 Normalizarea până la forma normală 3

## 10 Crearea tabelor și inserarea de date coerente

screenshots go here

## 11 Cinci cereri SQL complexe

```
01 | --Pentru fiecare Membru afiseaza id-ul, numele complet si informatiile
02 | --despre fiecare track la care a contribuit. Track-ul trebuie sa apartina
03 | --unui album ce are cel putin o copie vanduta si un 'a' in titlu,
04 | --altfel nu se va afisa nimic.
05 |
06 | select art.id,
07 |       concat(concat(art.first_name, ' '), art.last_name),
08 |       t.*
09 | from artists art
10 | inner join group_members g_m
11 |       on art.id = g_m.member_id
12 | inner join groups g
13 |       on g_m.group_id = g.id
14 | inner join albums a
15 |       on g.id = a.group_id
16 | inner join tracks t
17 |       on a.id = t.album_id
18 | where (g_m.left_group_date is null or
19 |        g_m.left_group_date >= a.release_date) and
20 |        a.id in (select a2.id
21 |                from customers cust
22 |                inner join copies cop
23 |                  on cust.id = cop.customer_id
24 |                inner join albums a2
25 |                  on cop.album_id = a2.id
26 |                where lower(a2.title) like '%a%')
27 | order by art.id desc;
```

Listing 1: Operație *join* pe minimum 4 tabele, filtrare la nivel de linie, subcerere nesincronizată, ordonare, funcții pe șiruri de caractere

```

01 | --Afiseaza titlul si durata pentru fiecare album cu durata mai
02 | --mare decat 20(minute) si care este compus de un grup care are
03 | --in componenta cel putin un membru al carui nume incepe cu 'G'.
04 |
05 | select a.title,
06 |        sum(t.length)
07 | from albums a
08 | inner join tracks t
09 |        on a.id = t.album_id
10 | where exists(select art.last_name
11 |             from groups g
12 |             inner join group_members g_m
13 |                     on g.id = g_m.group_id
14 |             inner join artists art
15 |                     on g_m.member_id = art.id
16 |             where g.id = a.group_id and
17 |                    lower(art.last_name) like 'g%')
18 | group by a.id, a.title
19 | having sum(t.length) > 20;

```

Listing 2: Subcerere sincronizată, grupări, funcții grup, filtrare la nivel de grupuri, operații pe șiruri de caractere

```

01 | --Pentru fiecare artist al carui nume contine un 'e', afiseaza id-ul
02 | --si numele sau, ziua lunii in care s-a nascut, data decesului
03 | --sau alt mesaj corespunzator daca este in viata, varsta sa exprimata
04 | --in numar de luni. Se va afisa, in plus, 'DA/NU' daca artistul este/nu este
05 | --nascut in Anglia si 'DA/NU' daca numele orasului in care s-a nascut
06 | --contine/nu contine un 'u'.
07 |
08 | select art.id,
09 |        art.last_name,
10 |        to_char(art.date_birth, 'dd')           as "Ziua lunii nastere",
11 |        nvl(to_char(art.date_death),
12 |            'in viata')                         as "Decedat?",
13 |        trunc(months_between(nvl(art.date_death,
14 |                                sysdate),
15 |                                art.date_birth),
16 |            0)                                   as "Varsta in nr. luni",
17 |        decode(c.name, 'England', 'DA',
18 |              'NU')                               as "Nascut in Anglia?",
19 |        case
20 |            when lower(l.city) like '%u%'
21 |            then 'DA'
22 |            else 'NU'
23 |        end                                       as "Numele orasului contine 'u'"
24 | from artists art
25 | inner join locations l
26 |        on art.birth_location_id = l.id
27 | inner join countries c
28 |        on l.country_id = c.id
29 | where lower(art.last_name) like '%e%';

```

Listing 3: NVL, DECODE, CASE, funcții pe date calendaristice, funcții pe șiruri de caractere

```

01 | --Afiseaza titlul si durata pentru fiecare album cu durata mai mare
02 | --decat media duratelor tuturor albumelor.
03 |
04 | with avgduration(val)
05 |     as (select avg(sum(t.length))
06 |           from albums a
07 |           inner join tracks t
08 |               on a.id = t.album_id
09 |           group by a.id)
10 | select a.title,
11 |        sum(t.length)
12 | from avgduration avgdur, albums a
13 | inner join tracks t
14 |     on a.id = t.album_id
15 | group by a.id, a.title, avgdur.val
16 | having sum(t.length) > avgdur.val;

```

Listing 4: Clauza **WITH**

```

01 | --Pentru fiecare abonament care a expirat in anul 2019 sa se afiseze
02 | --categoria sa, numele clientului, numele vanzatorului si data expirarii.
03 |
04 | select cat.title,
05 |        c.last_name,
06 |        e.last_name,
07 |        s.expiry_date
08 | from subscriptions s
09 | inner join categories cat
10 |     on s.category_id = cat.id
11 | inner join customers c
12 |     on s.customer_id = c.id
13 | inner join employees e
14 |     on s.employee_id = e.id
15 | where to_char(expiry_date, 'yyyy') = '2019';

```

Listing 5: Funcții pe date calendaristice



## 12 Operații de actualizare/suprimare utilizând subcereri

```
01 | --Prelungeste pana la 1 ianuarie 2022 toate abonamentele care au
02 | --expirat in anul 2020 si care au fost facute de un angajat al
03 | --carui nume contine numar impar de litere.
04 |
05 | update subscriptions s
06 | set s.expiry_date = to_date('01/01/2022')
07 | where to_char(s.expiry_date, 'yyyy') = '2020' and
08 |       exists(select e.id
09 |              from employees e
10 |              where e.id = s.employee_id and
11 |                    mod(length(e.last_name), 2) = 1);
12 |
13 | --Se presupune ca in urma stingerii unor copii din baza de date,
14 | --au ramas clienti care nu au asociata nicio copie cumparata anterior.
15 | --Sterge acesti clienti din tabela Customers
16 |
17 | delete from customers cust
18 | where not exists(select c.id
19 |                  from copies c
20 |                  where c.customer_id is not null and
21 |                        c.customer_id = cust.id);
22 |
23 | --Mareste cu 10% salariul angajatilor ale caror prenume contin
24 | --litera 'm'. Se va tine cont de faptul ca salariul lor nu poate
25 | --sa depaseasca jobs.max_salary.
26 |
27 | update employees e
28 | set e.salary = decode(sign(e.salary + e.salary * (10 / 100) -
29 |                          (select j.max_salary
30 |                           from jobs j
31 |                           where e.job_id = j.id)),
32 |                      1, (select j.max_salary
33 |                          from jobs j
34 |                          where e.job_id = j.id),
35 |                      e.salary + e.salary * (10 / 100))
36 | where lower(e.first_name) like '%m%';
```

## 13 Crearea unei secvențe utilizate în inserarea unor înregistrări în tabele

```
01 | create sequence seq_artist_id
02 | minvalue 0
03 | start with 0
04 | increment by 1;
```

## 14 Cerința

## 15 Cerința