

# PROIECT BAZE DE DATE

---

1)

În acest proiect voi prezenta organizarea concertelor cu ajutorul bazelor de date. Această organizare este esențială din punct de vedere al gestionării tuturor rolurilor ce trebuie îndeplinite pentru susținerea concertelor, rezolvând eventualele probleme ce ar putea apărea pe parcursul organizării. De asemenea, bazele de date conferă claritate fiecărei attribute prezente în tabel. Tabela CONCERT reprezintă nucleul diagramei conceptuale. În organizarea acestuia trebuie să avem în vedere locul în care se va desfășura evenimentul. Astfel apare tabela VENUE ce este legată de CONCERT prin VENUE\_PRICE ce va reprezenta prețul închirierii locului respectiv pentru concerte diferite și pentru perioade diferite. În plus, organizarea concertului nu poate fi făcută fără ajutorul angajaților. Așa apare tabela EMPLOYEE legată de CONCERT prin tabela de legătură EMPLOYEE\_CONCERT. EMPLOYEE\_CONCERT va gestiona rolul fiecărui angajat și grupul din care face parte prin tabelele ROLE și GRUP. Tabela TICKET este legată de CONCERT prin tabela de legătură TICKET\_CONCERT. TICKET reprezintă biletele cumparate pentru fiecare concert în parte. Totodată, un asemenea eveniment nu poate fi organizat fără un moment artistic muzical, deci este necesară tabela ARTIST legată de CONCERT prin tabela de legătură CONCERT\_ARTIST\_PRICE ce va reprezenta pretul artistului pentru fiecare concert în parte, iar după cum știm un artist are nevoie de un impresar bun pentru a susține cât mai multe recitaluri, tabela răspunzătoare pentru gestionarea impresarilor fiind MANAGER. Un element crucial ce ține de organizarea concertului este sponsorizarea, sponsorii aparținând tabelii SPONSOR, tabelă legată de CONCERT prin SPONSORSHIP deoarece putem avea același sponsor la mai multe evenimente.

2)

-Tabela CONCERT va conține informații despre fiecare concert în parte.

-Este necesar un număr mare de bilete, astfel se va constitui numărul de participanți la concert (ce trebuie să fie cât mai mare), iar banii strânși din vânzarea билетelor vor fi intra în alcătuirea bugetului prevăzut de organizarea evenimentului.

-Numărul de angajați trebuie să fie de asemenea ridicat, iar fiecare angajat va face parte dintr-un grup și va avea un rol.

-La un concert poate cânta un singur artist, iar acesta va avea un preț diferit în funcție de evenimentul la care este chemat.

-Un artist poate avea un singur manager care necesită 5 ani de experiență în acest domeniu.

-Un concert poate fi într-un singur loc, iar locația va avea un preț în funcție de concert și perioadă.

-Concertul poate avea mai mulți sponsori, iar fiecare sponsor va contribui cu o sumă.

- Un angajat poate participa la mai multe concerte.
- Un sponsor poate sponsoriza mai multe concerte.
- Un artist poate cânta la mai multe concerte.
- Un manager poate lucra cu mai mulți artiști.
- Se pot organiza mai multe concerte într-o locație.

3)

Pentru modelul de date referitor la organizarea concertelor, structurile CONCERT, TICKET, VENUE\_PRICE, VENUE, SPONSOR, CONCERT\_ARTIST\_PRICE, ARTIST, MANAGER, EMPLOYEE, EMPLOYEE\_CONCERT, ROLE și GRUP reprezintă entități. Vom prezenta entitățile modelului de date, dând o descriere completă a fiecăreia. De asemenea, pentru fiecare entitate se va preciza cheia primară. Toate entitățile care vor fi prezentate sunt independente, cu excepția entităților dependente ROLE și GROUP.

-CONCERT = este tabela principală, în jurul căreia se orientează toți factori spre organizarea unui concert. Aceasta conține datele principale, cum ar fi numele evenimentului și ziua în care are loc. Cheia primară a entității este ID.

-TICKET = reprezintă biletul cumpărat de fiecare persoană în parte. Acesta va avea un preț diferit în funcție de concertul care are loc și în funcție de tipul persoanei ce îl deține (copil, student, adult). Cheia primară a entității este ID.

-VENUE\_PRICE = entitate dependentă de VENUE. Este o tabelă ce va afișa prețul închirierii unei locații în funcție de concert și în funcție de perioada în care are loc. Cheia primară este compusă aici din CONCERT\_ID și VENUE\_ID.

-VENUE = entitate ce conține informații utile despre locurile disponibile în vederea susținerii unui concert (adresa, oraș, etc.). Cheia primară a entității este ID.

-SPONSOR = persoană fizică sau juridică ce contribuie financiar sau în altă manieră la desfășurarea prezentării de modă. Cheia primară a acestei entități este ID.

-CONCERT\_ARTIST\_PRICE = entitate dependentă de ARTIST ce va deține suma cerută de un fiecare artist în parte pentru a susține un concert. Această sumă va fi diferi de la concert la concert. Cheia primară a entității este compusă din CONCERT\_ID și ARTIST\_ID.

-ARTIST = persoană fizică sau juridică selectată pentru a-și susține momentul artistic în cadrul unui concert. Cheia primară a entității este ID.

-MANAGER = persoană fizică sau juridică răspunzătoare pentru încasările artistului cu care este în colaborare. Cheia primară a entității este ID.

-EMPLOYEE = persoană fizică sau juridică ce ajută la organizarea unuia sau a mai multor concerte. Cheia primară a entității este ID.

-EMPLOYEE\_CONCERT = identifică fiecare angajat în parte și îl repartizează în funcție de rol și de grupul din care face parte. Cheia primară este compusă din CONCERT\_ID și EMPLOYEE\_ID.

-ROLE = entitate dependentă de EMPLOYEE\_CONCERT ce constituie rolul fiecărui angajat în parte. Cheia primară a entității este ID.

-GRUP = entitate dependentă de EMPLOYEE\_CONCERT ce constituie grupul din care fiecare angajat face parte. Cheia primară a entității este ID.

4)

Voi prezenta relațiile modelului de date, dând o descriere completă a fiecăreia. De fapt, denumirile acestor legături sunt sugestive, reflectând conținutul acestora și entitățile pe care le leagă. Pentru fiecare relație se va preciza cardinalitatea minimă și maximă.

-CONCERT\_are\_TICKET = relație care leagă entitățile CONCERT și TICKET, reflectând legătura dintre acestea (ce bilete are un concert). Relația are cardinalitatea minimă 1:0 și cardinalitatea maximă m:n.

-CONCERT\_are\_VENUE = relație care leagă entitățile CONCERT și VENUE, reflectând legătura dintre acestea (ce locație are un concert). Relația are cardinalitatea minimă 1:0 și cardinalitatea maximă m:n.

-VENUE\_PRICE\_are\_CONCERT = relație care leagă entitățile VENUE\_PRICE și CONCERT, reflectând legătura dintre acestea (care este prețul pentru concert). Relația are cardinalitatea minimă 1:1 și cardinalitatea maximă 1:n.

-VENUE\_are\_VENUE\_PRICE = relație care leagă entitățile VENUE și VENUE\_PRICE, reflectând legătura dintre acestea (ce preț are locul respectiv). Relația are cardinalitatea minimă 1:1 și cardinalitatea maximă m:n.

-SPONSOR\_finanțează\_CONCERT = relație care leagă entitățile SPONSOR și CONCERT, reflectând legătura dintre acestea (ce concert este finanțat de sponsor). Relația are cardinalitatea minimă 0:1 și cardinalitatea maximă m:n.

-ARTIST\_cântă\_la\_CONCERT = relație care leagă entitățile ARTIST și CONCERT, reflectând legătura dintre acestea (care artist cântă la concert). Relația are cardinalitatea minimă 0:1 și cardinalitatea maximă n:1.

-ARTIST\_are\_CONCERT\_ARTIST\_PRICE = relație care leagă entitățile ARTIST și CONCERT\_ARTIST\_PRICE, reflectând legătura dintre acestea (ce sumă cere un artist). Relația are cardinalitatea minimă 1:1 și cardinalitatea maximă m:n.

-CONCERT\_ARTIST\_PRICE\_are\_CONCERT = relație care leagă entitățile CONCERT\_ARTIST\_PRICE și CONCERT, reflectând legătura dintre acestea (care este prețul cerut de artist pentru un anumit concert). Relația are cardinalitatea minimă 1:1 și cardinalitatea maximă n:1.

-ARTIST\_are\_MANAGER = relație care leagă entitățile ARTIST și MANAGER, reflectând legătura dintre acestea (ce manager are un artist). Relația are cardinalitatea minimă 1:0 și cardinalitatea maximă 1:n.

-CONCERT\_are\_EMPLOYEE = relație care leagă entitățile CONCERT și, reflectând legătura dintre acestea (ce angajați lucrează la organizarea unui concert). Relația are cardinalitatea minimă 1:0 și cardinalitatea maximă m:n.

-EMPLOYEE\_are\_ROLE = relație care leagă entitățile EMPLOYEE și ROLE, reflectând legătura dintre acestea (care sunt rolurile pe care le poate avea un angajat). Relația are cardinalitatea minimă 0:1 și cardinalitatea maximă 1:n.

-EMPLOYEE\_are\_GRUP = relație care leagă entitățile EMPLOYEE și GRUP, reflectând legătura dintre acestea (care sunt grupurile din care angajații fac parte). Relația are cardinalitatea minimă 0:1 și cardinalitatea maximă m:n.

5)

Toate cheile primare și toate cheile străine vor fi diferite de null. Mai jos găsiți descrierea atributelor:

a)CONCERT:

CONCERT\_BUDGET = variabilă de tip întreg, de lungime maximă 10, care reprezintă bugetul concertului (concert\_budget>10000).

EVENT\_NAME = variabilă de tip caracter, de lungime maximă 20, care reprezintă numele evenimentului (event\_name is not null).

DATE = variabilă de tip data calendaristică, care reprezintă data în care concertul va avea loc.

b)TICKET\_CONCERT:

TICKET\_ID = variabilă de tip întreg, de lungime maximă 10, cheie primară, reprezintă id-ul unui bilet.

CONCERT\_ID = variabilă de tip întreg, de lungime maximă 10, cheie primară, reprezintă id-ul unui concert.

c)TICKET

TYPE = variabilă de tip caracter, de lungime maximă 20, reprezintă tipul biletului (type in ('adult','child','student')).

PRICE = variabilă de tip întreg, de lungime maxima 10, reprezintă prețul unui bilet (price>=10).

#### d)RENT\_PRICE

CONCERT\_ID = variabilă de tip întreg, de lungime maximă 10, cheie primară, reprezintă id-ul unui concert.

VENUE\_ID = variabilă de tip întreg, de lungime maximă 10, cheie primară, reprezintă id-ul unei locații.

PRICE = variabilă de tip întreg, de lungime maximă 10, reprezintă prețul unei locații pentru un concert (price>=200).

#### e)VENUE

NAME = variabilă de tip caracter, de lungime maximă 20, reprezintă numele unei locații (name is not null).

COUNTRY = variabilă de tip caracter, de lungime maximă 20, reprezintă numele țării în care se află locația.

CITY = variabilă de tip caracter, de lungime maximă 20, reprezintă numele orașului în care se află locația.

ADDRESS = variabilă de tip caracter, de lungime maximă 20, reprezintă adresa locației (address is not null).

#### f)SPONSORSHIP

CONCERT\_ID = variabilă de tip întreg, de lungime maximă 10, cheie primară, reprezintă id-ul unui concert.

SPONSOR\_ID = variabilă de tip întreg, de lungime maximă 10, cheie primară, reprezintă id-ul unui sponsor.

#### g)SPONSOR

NAME = variabilă de tip character, de lungime maximă 20, reprezintă numele unui sponsor (name is not null).

INVESTMENT = variabilă de tip întreg, de lungime maximă 10, reprezintă investiția făcută de un sponsor (investment>=100).

#### h)CONCERT\_ARTIST\_PRICE

CONCERT\_ID = variabilă de tip întreg, de lungime maximă 10, cheie primară, reprezintă id-ul unui concert.

ARTIST\_ID = variabilă de tip întreg, de lungime maximă 10, cheie primară, reprezintă id-ul unui artist.

PRICE = variabilă de tip întreg, de lungime maximă 10, ce reprezintă prețul cerut de artist pentru un concert (price>=500).

#### i)ARTIST

MANAGER\_ID = variabilă de tip întreg, de lungime maximă 10, cheie străină, reprezintă id-ul unui manager.

NAME = variabilă de tip caracter, de lungime maximă 20, reprezintă numele unui artist (name is not null).

MUSIC\_GENRE = variabilă de tip caracter, de lungime maximă 20, reprezintă stilul de muzică cântat de un artist (music\_genre in ('pop-rock','rock','blues','alternative','funk')).

#### j)MANAGER

FIRST\_NAME = variabilă de tip caracter, de lungime maximă 20, reprezintă prenumele unui manager (first\_name is not null).

LAST\_NAME = variabilă de tip caracter, de lungime maximă 20, reprezintă numele unui manager (last\_name is not null).

YEARS\_EXPERIENCE = variabilă de tip întreg, de lungime maximă 10, reprezintă anii de experiență acumulați de un manager în domeniul muzical (years\_experience>5).

EMAIL = variabilă de tip caracter, de lungime maximă 20, reprezintă email-ul unui manager(unique(email)).

#### k)EMPLOYEE\_CONCERT

EMPLOYEE\_ID = variabilă de tip întreg, de lungime maximă 10, cheie primară, reprezintă id-ul unui angajat.

CONCERT\_ID = variabilă de tip întreg, de lungime maximă 10, cheie primară, reprezintă id-ul unui concert.

ROLE\_ID = variabilă de tip întreg, de lungime maximă 10, cheie străină, reprezintă id-ul unui rol pe care îl poate avea un angajat.

GROUP\_ID = variabilă de tip întreg, de lungime maximă 10, cheie străină, reprezintă id-ul unui grup din care poate face parte un angajat.

#### l)EMPLOYEE

FIRST\_NAME = variabilă de tip caracter, de lungime maximă 20, reprezintă prenumele unui angajat (first\_name is not null).

LAST\_NAME = variabilă de tip caracter, de lungime maximă 20, reprezintă numele unui angajat (last\_name is not null).

SALARY = variabilă de tip întreg, de lungime maximă 10, reprezintă salariul unui angajat (salary>=20).

EMAIL = variabilă de tip caracter, de lungime maximă 20, reprezintă email-ul unui angajat(unique(email)).

#### m)ROLE

NAME = variabilă de tip caracter, de lungime maximă 20, reprezintă numele rolului pe care îl poate avea un angajat (name is not null).

DESCRIPTION = variabilă de tip caracter, de lungime maximă 50, reprezintă descrierea rolului pe care îl poate avea un angajat.

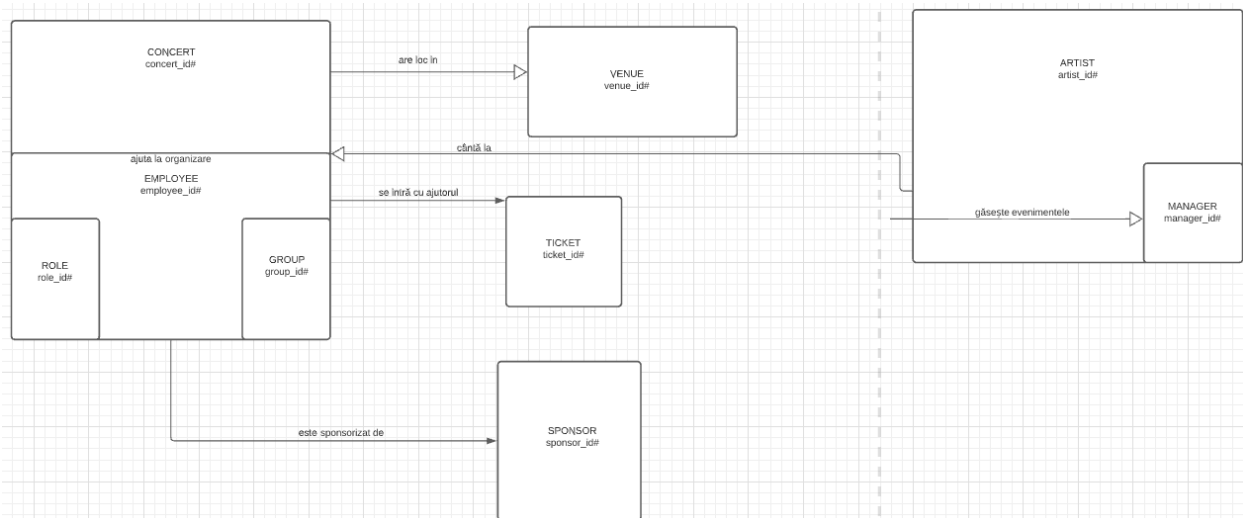
#### n)GRUP

NAME = variabilă de tip caracter, de lungime maximă 20, reprezintă numele grupului din care poate face parte un angajat (name is not null).

DESCRIPTION = variabilă de tip caracter, de lungime maximă 50, reprezintă descrierea grupului din care poate face parte un angajat.

6)

Mai jos găsiți diagrama E/R.



Mai jos găsiți diagrama conceptuală.



-CONCERT (ID#, CONCERT\_BUDGET, EVENT\_NAME, DATE)

-TICKET (ID#, TYPE, PRICE)

```
-RENT_PRICE(CONCERT_ID#, VENUE_ID#, PRICE)
```

-VENUE (ID#, NAME, COUNTRY, CITY, ADDRESS)

-SPONSORSHIP (CONCERT\_ID#, SPONSOR\_ID#)

-SPONSOR (ID#, NAME, INVESTMENT)

-CONCERT\_ARTIST\_PRICE (CONCERT\_ID#, ARTIST\_ID#, PRICE)

-ARTIST (ID#, MANAGER\_ID, NAME, MUSIC\_GENRE)

-MANAGER (ID#, FIRST\_NAME, LAST\_NAME, YEARS EXPERIENCE, EMAIL)

-EMPLOYEE CONCERT (EMPLOYEE ID#, CONCERT ID#, ROLE ID, GROUP ID)

-EMPLOYEE (ID#, FIRST\_NAME, LAST\_NAME, SALARY, EMAIL)



-ROLE (ID#, NAME, DESCRIPTION)

-GRUP (ID#, NAME, DESCRIPTION)

9)

Exemplu non-FN1

Prima formă normală exclude posibilitatea existenței grupurilor repetitive astfel încât fiecare câmp într-o baza de date să cuprindă numai o valoare atomică. Spre exemplu entitatea CONCERT ar fi avut următoarea schema relațională (ID#, CONCERT\_BUDGET, EVENT\_NAME, DATE, EMPLOYEES):

ID	CONCERT_BUDGET	EVENT_NAME	DATE	EMPLOYEES
1	15000	Wacken	08.11.2023	Schmidt, Muller, John, Fred

Acest tip de organizare nu exclude o posibilă aglomerare de informații pe coloana EMPLOYEES fapt ce ar crea ambiguitate la nivelul procesării datelor. Astfel vom aduce la prima formă normală (FN1), ceea ce presupune adăugarea unei linii separate pentru fiecare angajat în parte.

ID	CONCERT_BUDGET	EVENT_NAME	DATE	EMPLOYEES
1	15000	Wacken	08.11.2023	Schmidt
1	15000	Wacken	08.11.2023	Muller
1	15000	Wacken	08.11.2023	John
1	15000	Wacken	08.11.2023	Fred

Exemplu non-FN2

Cea de a doua formă normală(FN2) cere ca tabela să se afle în FN1 și toate elementele tabelului să fie dependente de cheia primară. Dacă unul sau mai multe elemente sunt dependente funcțional numai de o parte a cheii primare, atunci va fi nevoie de o repartizare pe tabele diferite.

ID	CONCERT_BUDGET	EVENT_NAME	DATE
1	15000	Wacken	08.11.2023

EMPLOYEE_ID	EMPLOYEE	CONCERT_ID
1	Schmidt	1
2	Muller	1
3	John	1
4	Fred	1

### Exemplu non-FN3

Toate attributele non-chei ale unei relații depind numai de chei candidate ale acelei relații. Toate attributele non-cheie sunt (trebuie sa fie) mutual independente.

Pornind de la tabelul de la FN2 vom mai adăuga două coloane ce vor reprezenta rolul angajatului la concertul respectiv și grupul din care face parte.

EMPLOYEE_ID	EMPLOYEE	CONCERT_ID	ROLE	GROUP
1	Schmidt	1	Tehnician	Alfa
2	Muller	1	Inginer	Gama
3	John	1	Inginer	Gama
4	Fred	1	Tehnician	Alfa

Schema finală normalizată FN3 va arăta astfel:

EMPLOYEE_ID	EMPLOYEE	CONCERT_ID	ROLE_ID
1	Schmidt	1	1
2	Muller	1	2
3	John	1	2
4	Fred	1	1

ROLE_ID	ROLE	GROUP
1	Tehnician	Alfa
2	Inginer	Gama

10)

Mai jos găsiți printscreen-urile cu tabelele create.

-CONCERT

	ID	CONCERT_BUDGET	EVENT_NAME	DATE
1	1	11000	Ragan	11-FEB-23
2	2	10500	Ant	03-APR-21
3	3	13500	Deck	17-JAN-30
4	4	11000	Wacken	23-AUG-22
5	5	20000	Vast	30-NOV-25
6	6	100000	Temper	29-DEC-27
7	7	14320	Castlevania	05-AUG-22
8	8	500000	Ant	01-JAN-24
9	9	20000	Ring	13-MAY-28
10	10	17000	Vast	20-DEC-21

-TICKET\_CONCERT

	TICKET_ID	CONCERT_ID
1	1	1
2	2	1
3	3	1
4	4	2
5	5	2
6	6	2
7	7	3
8	8	3
9	9	3
10	10	4
11	11	4
12	12	4
13	1	5
14	2	5
15	3	5
16	4	6
17	5	6
18	6	6
19	7	7
20	8	7
21	9	7
22	10	8
23	11	8
24	12	8
25	1	9
26	2	9
27	3	9
28	4	10
29	5	10
30	6	10

-TICKET

ID	TYPE	PRICE
1	1 adult	50
2	2 child	20
3	3 student	30
4	4 adult	100
5	5 child	50
6	6 student	70
7	7 adult	150
8	8 child	100
9	9 student	110
10	10 adult	200
11	11 child	150
12	12 student	155

-RENT\_PRICE

CONCERT_ID	VENUE_ID	PRICE
1	1	9 1000
2	2	1 2500
3	3	2 1000
4	4	5 700
5	5	7 925
6	6	4 1200
7	7	6 3000
8	8	6 1700
9	9	10 1350
10	10	3 880

-VENUE

ID	NAME	COUNTRY	CITY	ADDRESS
1	1 Alsace	Croatia	Zagreb	11 c
2	2 Loreenne	Greece	Samotraki	7 a
3	3 Cast	Serbia	Belgrade	4 k
4	4 Viren	Germany	Berlin	23 d
5	5 Kolos	Israel	Tel Aviv	15 b
6	6 Turuk	Turkey	Istanbul	19 f
7	7 Nepav	Greece	Athens	20 c
8	8 Antigua	Germany	Munchen	1 a
9	9 Bora	Russia	Moskow	2 b
10	10 Vanjt	Ukraine	Odessa	33 g

-SPONSORSHIP

	CONCERT_ID	SPONSOR_ID
1	1	10
2	1	9
3	2	1
4	2	7
5	3	4
6	3	5
7	3	3
8	4	2
9	4	10
10	5	7
11	6	6
12	6	4
13	7	6
14	8	2
15	8	9
16	8	3
17	9	1
18	9	4
19	9	5
20	10	8

-SPONSOR

	ID	NAME	INVESTMENT
1	1	Corvus	100
2	2	Tronte	900
3	3	Markus	1500
4	4	Octavian	125
5	5	Mannstein	100
6	6	Otto	750
7	7	Erwin	300
8	8	Heinz	1100
9	9	Calin	615
10	10	Valy	1500

-CONCERT\_ARTIST\_PRICE

	CONCERT_ID	ARTIST_ID	PRICE
1	1	1	4000
2	2	7	5000
3	3	5	3250
4	4	2	2700
5	5	3	7000
6	6	2	1300
7	7	10	6600
8	8	8	5000
9	9	4	2300
10	10	9	4950

-ARTIST

	ID	MANAGER_ID	NAME	MUSIC_GENRE
1	1	3	Ary	blues
2	2	1	Alex	rock
3	3	7	Brugner	pop-rock
4	4	2	Ervan	funk
5	5	9	Ingus	alternative
6	6	8	Toles	blues
7	7	10	Robert	rock
8	8	1	Vunk	alternative
9	9	4	Stamp	funk
10	10	5	Wong	pop-rock

-MANAGER

	ID	FIRST_NAME	LAST_NAME	YEARS_EXPERIENCE	EMAIL
1	1	Alex	Lars	12	alex@gmail.com
2	2	Bruks	Norton	6	nort@gmail.com
3	3	Peter	Muller	8	mull@gmail.com
4	4	Heidrich	Hoffmann	6	homan@gmail.com
5	5	Reinhardt	Baumer	15	baum@gmail.com
6	6	Jose	Maria	7	maria23@gmail.com
7	7	Alejandro	Miguel	9	migus@gmail.com
8	8	Marco	Polo	10	paus7@gmail.com
9	9	Cristoph	Waltz	8	swing@gmail.com
10	10	Renata	Carlovsky	11	sky33@gmail.com

-EMPLOYEE\_CONCERT

	EMPLOYEE_ID	CONCERT_ID	GRUP_ID	ROLE_ID
1	10	1	5	4
2	6	1	4	4
3	4	1	5	1
4	1	2	1	5
5	7	2	3	2
6	8	2	2	3
7	6	2	3	3
8	5	3	4	3
9	2	3	4	2
10	10	4	1	5
11	9	4	2	4
12	8	5	5	1
13	1	5	3	3
14	4	5	2	1
15	7	6	1	2
16	2	6	1	4
17	9	7	5	2
18	3	7	1	1
19	5	7	4	3
20	8	7	3	3
21	9	8	4	5
22	10	8	2	1
23	4	8	1	4
24	1	9	4	2
25	5	9	1	3
26	3	9	5	4
27	3	10	5	4
28	5	10	3	5
29	7	10	2	1
30	6	10	4	3

-EMPLOYEE



ID	FIRST_NAME	LAST_NAME	SALARY	EMAIL
1	1 Andrei	Bordeianu	130	ab@gmail.com
2	2 Cosmin	Nedelcu	300	cn@gmail.com
3	3 Hongos	Negyem	235	hn@gmail.com
4	4 Mirela	Voicu	115	mv@gmail.com
5	5 Alex	Pintiliasa	270	ap@gmail.com
6	6 Tudor	Stavar	130	ts@gmail.com
7	7 Robert	Moisei	300	rm@gmail.com
8	8 Dragos	Popovici	220	dp@gmail.com
9	9 Florin	Patrascu	170	fp@gmail.com
10	10 Valentin	Stirbu	420	vs@gmail.com

#### -ROLE

ID	NAME	DESCRIPTION
1	1 Marketing	Sells merchandise
2	2 Medical Assistant	Helps any injured participant
3	3 Web Designer	Makes an attractive website
4	4 Photographer	Photoshoot the funny moments
5	5 Construction Worker	Is setting up the concert place

#### -GRUP

ID	NAME	DESCRIPTION
1	1 Alpha	Young members
2	2 Beta	Dedicated members
3	3 Gamma	Attractive members
4	4 Sigma	Hardworking members
5	5 Omega	Valuable members

11)

a)

Oracle SQL Developer: C:\Users\the\_g\PROJECT\_BD2.sql

File Edit View Navigate Run Source Tools Window Help

Connections

- Oracle Connections
  - BD
  - PROJECT\_BD
  - PROJECTBD
- Oracle NetSQL Connections
- Database Schema Service Connections

SQL Worksheet: History

Worksheet: Query Builder

```
--e
--Se afisare numele evenimentului, orasul in care are loc , adresa si pretul biletelor care sunt
--de tipul student
select c.event_name, v.city, v.address, t.price from ticket t
join ticket_concert c on t.ticket_id = t.id
join concert c on c.concert_id = c.id
join seat_price sp on sp.concert_id = c.id
join venue v on v.id = sp.venue_id
where t.type = 'student'
order by c.event_name;
```

Script Output: Query Result

All Rows Fetched: 10 in 0.004 seconds

	EVENT_NAME	CITY	ADDRESS	PRICE
1	Ant	Zagreb	11 c	70
2	Ant	Istanbul	19 f	155
3	Castlevania	Istanbul	19 f	110
4	Deft	Sabotzaki	7 a	110
5	Pagan	Moscow	2 b	30
6	Ring	Odesa	33 g	30
7	Temper	Berlin	23 d	70
8	Vast	Belgrade	4 k	70
9	Vast	Athens	20 c	30
10	Wacken	Tei Aviv	15 b	155

Reports

- All Reports
- Analytic View Reports
- Data Dictionary Reports
- Data Modeler Reports
- OLAP Reports
- TimesTen Reports
- User Defined Reports

Click on an identifier with the Control key down to perform "Go to Declaration"

Type here to search

85°F Cloudy 3:14 PM 6/26/2021

b)

Oracle SQL Developer: C:\Users\the\_g\PROJECT\_BD2.sql

File Edit View Navigate Run Source Tools Window Help

Connections

- Oracle Connections
  - BD
  - PROJECT\_BD
  - PROJECTBD
- Oracle NetSQL Connections
- Database Schema Service Connections

SQL Worksheet: History

Worksheet: Query Builder

```
--b
--Se se afisare genul muzical si numarul de artisti
--care interpreteaza genul respectiv
select a.music_genre, count(a.id) as Number_of_artists from artist a
group by a.music_genre having count(a.id)>=2;
```

Script Output: Query Result

All Rows Fetched: 3 in 0.002 seconds

	MUSIC_GENRE	NUMBER_OF_ARTISTS
1	(null)	3
2	blues	2
3	rock	2

Reports

- All Reports
- Analytic View Reports
- Data Dictionary Reports
- Data Modeler Reports
- OLAP Reports
- TimesTen Reports
- User Defined Reports

Click on an identifier with the Control key down to perform "Go to Declaration"

Type here to search

85°F Cloudy 3:14 PM 6/26/2021

c)

Oracle SQL Developer: C:\Users\the\_g\PROJECT\_BD2.sql

File Edit View Navigate Run Source Team Tools Window Help

Connections

- Oracle Connections
  - BD
  - PROJECT\_BD
  - PROJECT\_BD2
  - Oracle NetSQL Connections
  - Database Schema Service Connections

SQL Worksheet: History

Worksheet Query Builder

```
--Sa se afiseze id-ul angajatilor, numele complet si id-ul rolurilor pe care le au
--ordonat crescator dupa nume
WITH employee_with AS (select e.id, e.first_name||' '||e.last_name AS Name, e.salary, e.email, ec.role_id from employee e , employee_concept ec where ec.employee_id = e.id)
select id, Name, role_id from employee_with
order by 2;
```

Script Output x Query Result x

All Rows Fetched: 30 in 0.004 seconds

ID	NAME	ROLE_ID
1	Alex Pintillasa	3
2	Alex Pintillasa	3
3	Alex Pintillasa	5
4	Alex Pintillasa	3
5	Andrei Bordeianu	5
6	Andrei Bordeianu	2
7	Andrei Bordeianu	5
8	Coman Medelcu	2
9	Coman Medelcu	4
10	Dragos Popovici	3
11	Dragos Popovici	1
12	Dragos Popovici	3
13	Florin Patrascu	5
14	Florin Patrascu	4
15	Florin Patrascu	2
16	Hongus Hegyen	1
17	Hongus Hegyen	4
18	Hongus Hegyen	4
19	Mirela Voicu	1
20	Mirela Voicu	4
21	Mirela Voicu	1
22	Robert Moisei	2
23	Robert Moisei	2
24	Robert Moisei	1
25	Tudor Stavar	3
26	Tudor Stavar	4
27	Tudor Stavar	3
28	Valentin Stirbu	5
29	Valentin Stirbu	4
30	Valentin Stirbu	1

Reports

- All Reports
- Analytic View Reports
- Data Dictionary Reports
- Data Modeler Reports
- OLAP Reports
- TimesTen Reports
- Line Defined Reports

Click on an identifier with the Control key down to perform 'Go to Declaration'

Type here to search

85°F Cloudy 3:14 PM 6/26/2021

d)

Oracle SQL Developer: C:\Users\the\_g\PROJECT\_BD2.sql

File Edit View Navigate Run Source Team Tools Window Help

Connections

- Oracle Connections
  - BD
  - PROJECT\_BD
  - PROJECT\_BD2
  - Oracle NetSQL Connections
  - Database Schema Service Connections

SQL Worksheet: History

Worksheet Query Builder

```
--Sa se afiseze data actuala si ora exacta
select TO_CHAR(sysdate, 'YYYY/MM/DD hh24:mi:ss') from dual;
```

Script Output x Query Result x

All Rows Fetched: 1 in 0.002 seconds

TO_CHAR(SYSDATE, 'YYYY/MM/DD HH24:MI:SS')
1 2021/06/26 15:14:55

Reports

- All Reports
- Analytic View Reports
- Data Dictionary Reports
- Data Modeler Reports
- OLAP Reports
- TimesTen Reports
- Line Defined Reports

Click on an identifier with the Control key down to perform 'Go to Declaration'

Type here to search

85°F Cloudy 3:15 PM 6/26/2021

e)

Oracle SQL Developer: C:\Users\the\_g\PROJECT\_BD2.sql

File Edit View Navigate Run Source Team Tools Window Help

Connections

- Oracle Connections
  - BD
  - PROJECT\_BD
  - PROJECTBD
- Oracle NetSQL Connections
- Database Schema Service Connections

SQL Worksheet: History

Worksheet: Query Builder

```
--Sa se afiseze toate informatiile din tabela CONCERT
--pentru concertele care vor avea loc de la anul
select * from concert
where trunc(event_date)>trunc(sysdate)+365;
```

Script Output: Query Result

All Rows Fetched: 8 in 0.003 seconds

ID	CONCERT_BUDGET	EVENT_NAME	EVENT_DATE
1	11000	Ragan	11-FEB-23
2	13600	Decca	17-MAR-20
3	11000	Wacken	23-AUG-22
4	20000	Vaat	30-NOV-25
5	100000	Tespey	29-DEC-27
6	14320	Castlevania	05-AUG-22
7	500000	Jact	01-OCT-24
8	20000	Bang	13-SEP-28

Reports

- All Reports
- Analytic View Reports
- Data Dictionary Reports
- Data Modeler Reports
- OLAP Reports
- TimesTen Reports
- User Defined Reports

Click on an identifier with the Control key down to perform "Go to Declaration"

Type here to search

85°F Cloudy 3:15 PM 6/26/2021

f)

Oracle SQL Developer: C:\Users\the\_g\PROJECT\_BD2.sql

File Edit View Navigate Run Source Team Tools Window Help

Connections

- Oracle Connections
  - BD
  - PROJECT\_BD
  - PROJECTBD
- Oracle NetSQL Connections
- Database Schema Service Connections

SQL Worksheet: History

Worksheet: Query Builder

```
--f
--Sa se afiseze suma salariilor de pe fiecare rol, id-ul rolurilor, salariul minim cat si cel maxim
--in functie de id-ul rolurilor
select sum(e.salary), ec.role_id, min(e.salary), max(e.salary) from employee e join employee_concert ec on e.id = ec.employee_id
group by ec.role_id;
```

Script Output: Query Result

All Rows Fetched: 5 in 0.004 seconds

SUM(E.SALARY)	ROLE_ID	MIN(E.SALARY)	MAX(E.SALARY)
1405	1	115	420
1200	2	130	300
1605	4	115	420
990	5	130	420
1640	3	130	270

Reports

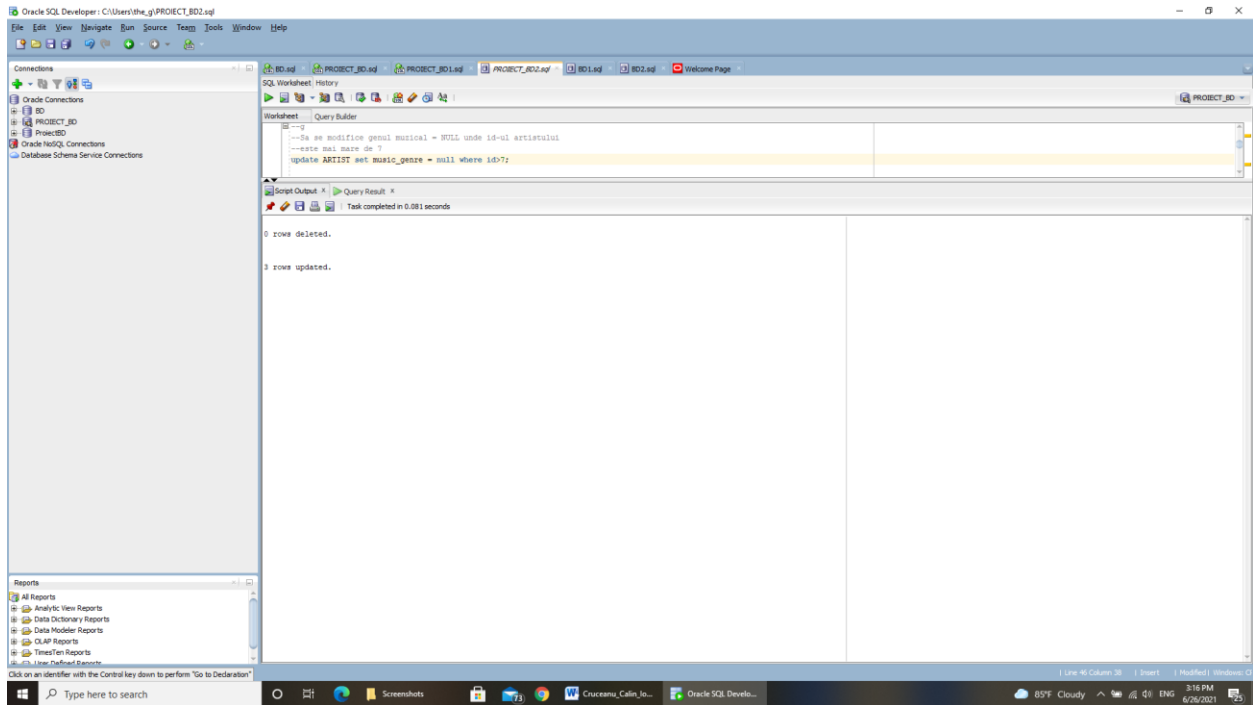
- All Reports
- Analytic View Reports
- Data Dictionary Reports
- Data Modeler Reports
- OLAP Reports
- TimesTen Reports
- User Defined Reports

Click on an identifier with the Control key down to perform "Go to Declaration"

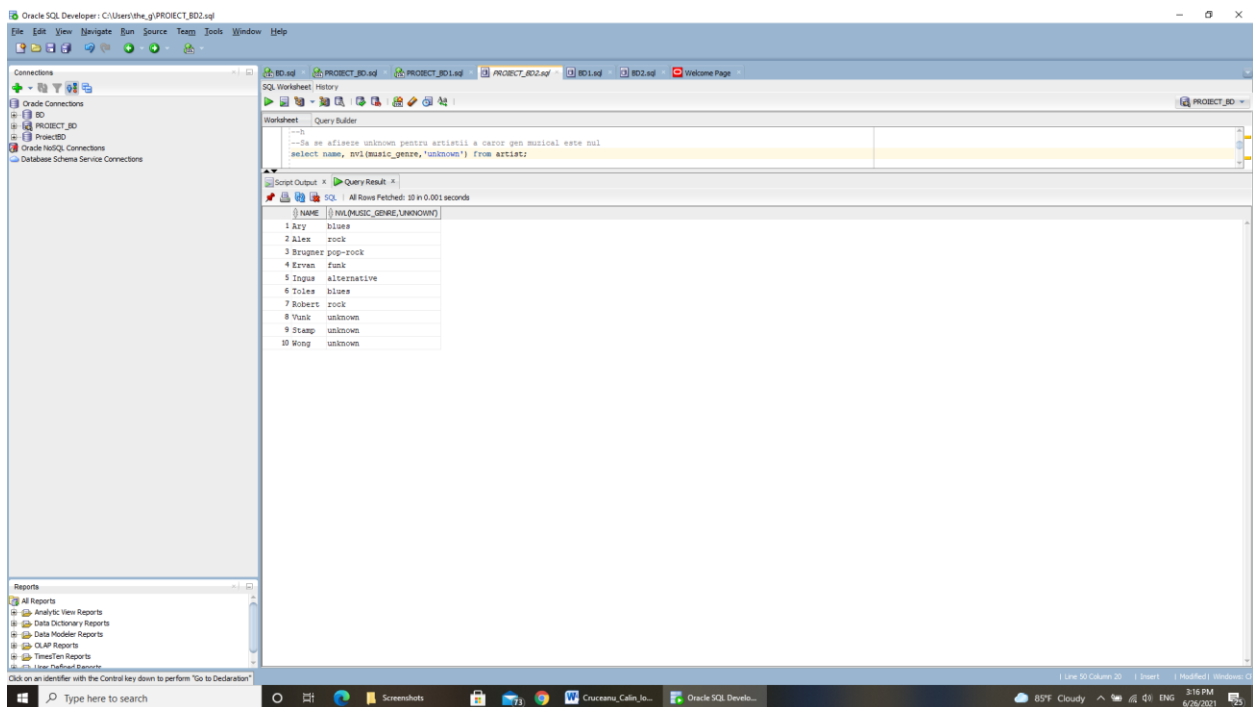
Type here to search

85°F Cloudy 3:15 PM 6/26/2021

g)



h)



i)

Oracle SQL Developer: C:\Users\the\_g\PROJECT\_BD2.sql

File Edit View Navigate Run Source Team Tools Window Help

Connections

- Oracle Connections
- BD
- PROJECT\_BD
- PROJECT\_BD2
- Oracle NetSQL Connections
- Database Schema Service Connections

SQL Worksheet: History

Worksheet: Query Builder

```
--Se se afiseaza yey pentru genul muzical rock,
--hura pentru funk, meh unde este nul si in rest '-' (varianta CASE)
select a.name, a.music_genre, (case
when a.music_genre = 'rock' then 'yey'
when a.music_genre = 'funk' then 'hura'
when a.music_genre is null then 'meh'
else '-'
end) as feelings
from artist a;
```

Script Output: Query Result

All Rows Fetched: 10 in 0.002 seconds

ID	NAME	MUSIC_GENRE	FEELINGS
1	Ary	blues	-
2	Alex	rock	yey
3	Brupner	pop-rock	-
4	Erven	funk	hura
5	Inguis	alternative	-
6	Toles	blues	-
7	Robert	rock	yey
8	Vank	(null)	meh
9	Scamp	(null)	meh
10	Wung	(null)	meh

Reports

- All Reports
- Analytic View Reports
- Data Dictionary Reports
- Data Modeler Reports
- OLAP Reports
- TimesTen Reports
- User Defined Reports

Click on an identifier with the Control key down to perform "Go to Declaration"

Type here to search

85°F Cloudy 3:16 PM 6/26/2021

j)

Oracle SQL Developer: C:\Users\the\_g\PROJECT\_BD2.sql

File Edit View Navigate Run Source Team Tools Window Help

Connections

- Oracle Connections
- BD
- PROJECT\_BD
- PROJECT\_BD2
- Oracle NetSQL Connections
- Database Schema Service Connections

SQL Worksheet: History

Worksheet: Query Builder

```
--Se se afiseaza numele complet, solul si grupul
--din care fac parte angajatii
select e.first_name||' '||e.last_name,
(select r.name from role r where r.id = ec.role_id) as role_name,
(select g.name from grup g where g.id = ec.grup_id) as group_name
from employee e join employee_concert ec on e.id = ec.employee_id;
```

Script Output: Query Result

All Rows Fetched: 30 in 0.003 seconds

ID	FIRST_NAME  LAST_NAME	ROLE_NAME	GROUP_NAME
1	Valentin Scirbu	Photographer	Omega
2	Tudor Stavar	Photographer	Sigma
3	Mirela Voicu	Marketing	Omega
4	Andrei Bordeianu	Construction Worker	Alpha
5	Robert Misiel	Medical Assistant	Gamma
6	Dragos Popovici	Web Designer	Beta
7	Tudor Stavar	Web Designer	Gamma
8	Alex Pintiliana	Web Designer	Sigma
9	Cosmin Medelciu	Medical Assistant	Sigma
10	Valentin Scirbu	Construction Worker	Alpha
11	Florin Patrascu	Photographer	Beta
12	Dragos Popovici	Marketing	Omega
13	Andrei Bordeianu	Web Designer	Gamma
14	Mirela Voicu	Marketing	Beta
15	Robert Misiel	Medical Assistant	Alpha
16	Cosmin Medelciu	Photographer	Alpha
17	Florin Patrascu	Medical Assistant	Omega
18	Songos Negyem	Marketing	Alpha
19	Alex Pintiliana	Web Designer	Sigma
20	Dragos Popovici	Web Designer	Gamma
21	Florin Patrascu	Construction Worker	Sigma
22	Valentin Scirbu	Marketing	Beta
23	Mirela Voicu	Photographer	Alpha
24	Andrei Bordeianu	Medical Assistant	Sigma
25	Alex Pintiliana	Web Designer	Alpha
26	Songos Negyem	Photographer	Omega
27	Songos Negyem	Photographer	Omega
28	Alex Pintiliana	Construction Worker	Gamma
29	Robert Misiel	Marketing	Beta
30	Tudor Stavar	Web Designer	Sigma

Reports

- All Reports
- Analytic View Reports
- Data Dictionary Reports
- Data Modeler Reports
- OLAP Reports
- TimesTen Reports
- User Defined Reports

Click on an identifier with the Control key down to perform "Go to Declaration"

Type here to search

85°F Cloudy 3:17 PM 6/26/2021

k)

Oracle SQL Developer: C:\Users\the\_g\PROJECT\_BD2.sql

File Edit View Navigate Run Source Tools Window Help

Connections

- Oracle Connections
  - BD
  - PROJECT\_BD
  - PROJECTBD
- Oracle NetSQL Connections
- Database Schema Service Connections

SQL Worksheet: History

Worksheet Query Builder

```
--Se se afiseze numele rolului cu id-ul 2 si numele grupului cu id-ul 1
select (select r.name from role r where r.id = 2),
(select g.name from grup g where g.id = 1) as group_name
from dual;
```

Script Output x Query Result x

All Rows Fetched: 1 in 0.001 seconds

DECODE(NAMEFROMBOLERWHEREID=2)	GROUP_NAME
Medical Assistant	Alpha

Reports

- All Reports
- Analytic View Reports
- Data Dictionary Reports
- Data Modeler Reports
- OLAP Reports
- TimesTen Reports
- Line Defined Reports

Click on an identifier with the Control key down to perform "Go to Declaration"

Type here to search

85°F Cloudy 3:17 PM 6/26/2021

1)

Oracle SQL Developer: C:\Users\the\_g\PROJECT\_BD2.sql

File Edit View Navigate Run Source Tools Window Help

Connections

- Oracle Connections
  - BD
  - PROJECT\_BD
  - PROJECTBD
- Oracle NetSQL Connections
- Database Schema Service Connections

SQL Worksheet: History

Worksheet Query Builder

```
--Se se afiseze yay pentru genul musical rock,
--baza pentru funk, meh unde este nul si in rest '-' (variantea DECODE)
select music_genre, decode(music_genre,
'rock', 'yay',
'funk', 'baza',
null, 'meh',
'-')
from artist;
```

Script Output x Query Result x

All Rows Fetched: 10 in 0.003 seconds

MUSIC_GENRE	DECODE(MUSIC_GENRE, 'ROCK', 'YAY', 'FUNK', 'BAZA', 'Meh', '-')
1 blues	-
2 rock	yay
3 pop-rock	-
4 funk	baza
5 alternative	-
6 blues	-
7 rock	yay
8 (null)	meh
9 (null)	meh
10 (null)	meh

Reports

- All Reports
- Analytic View Reports
- Data Dictionary Reports
- Data Modeler Reports
- OLAP Reports
- TimesTen Reports
- Line Defined Reports

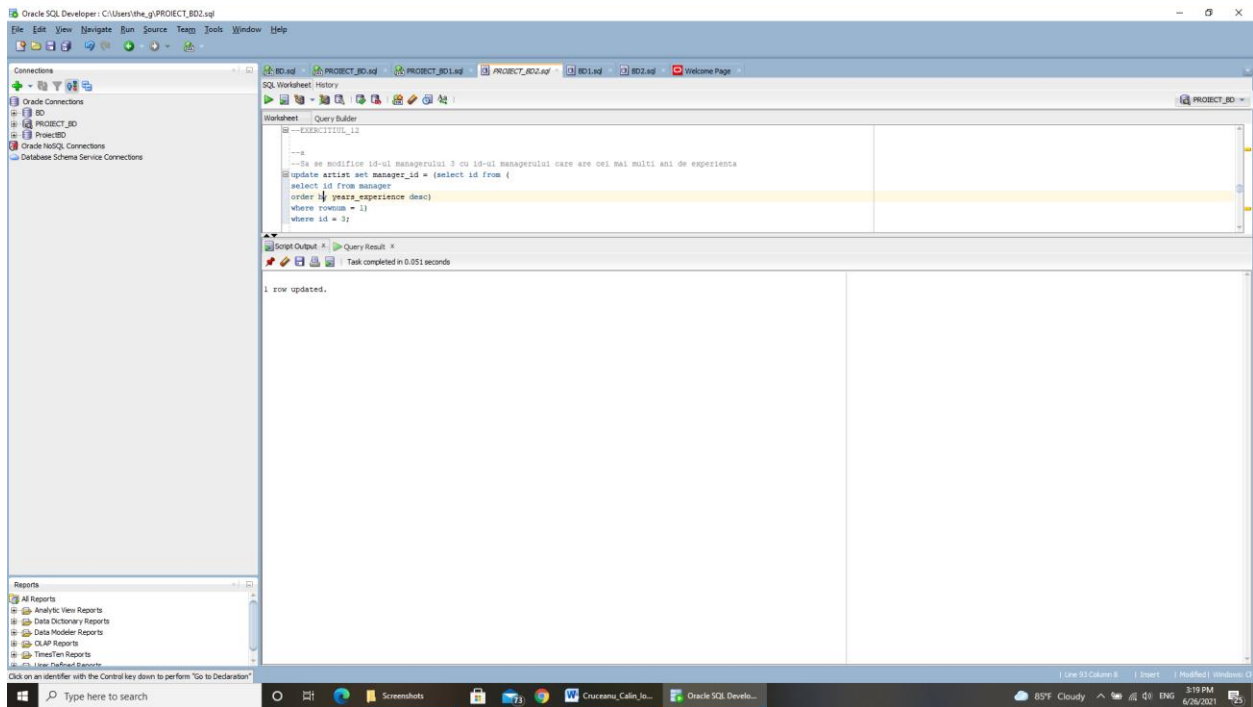
Click on an identifier with the Control key down to perform "Go to Declaration"

Type here to search

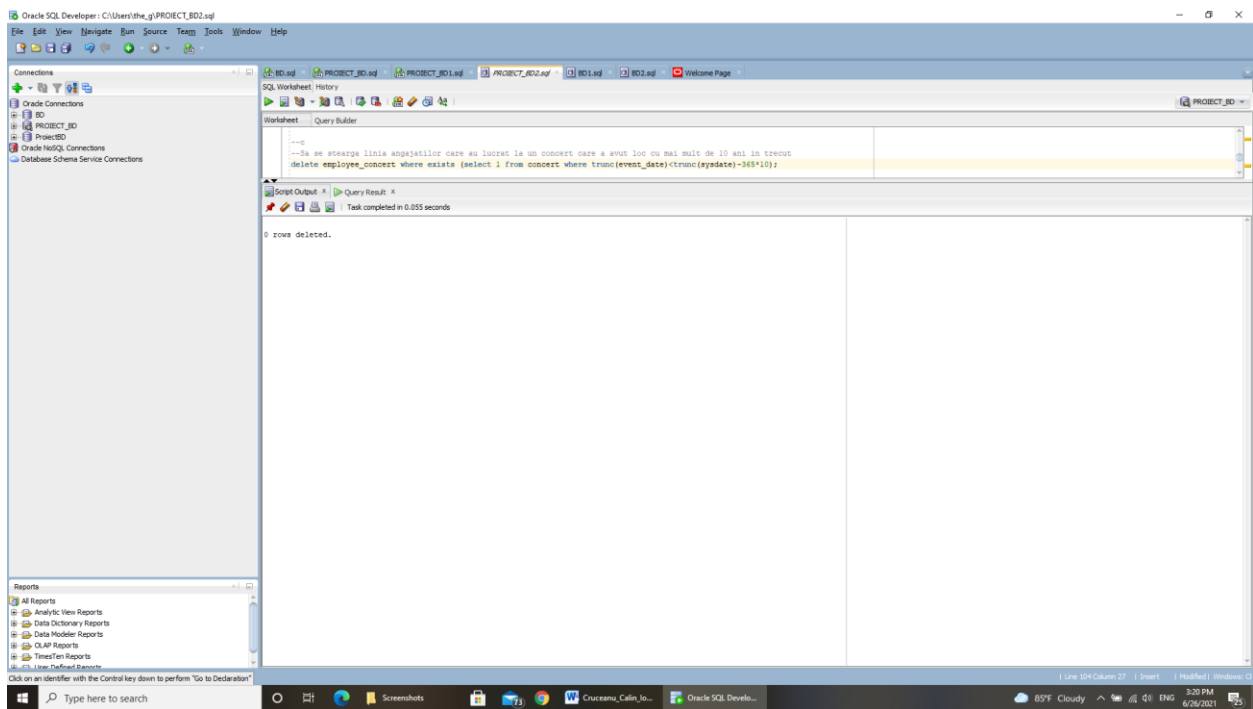
85°F Cloudy 3:17 PM 6/26/2021

12)

a)

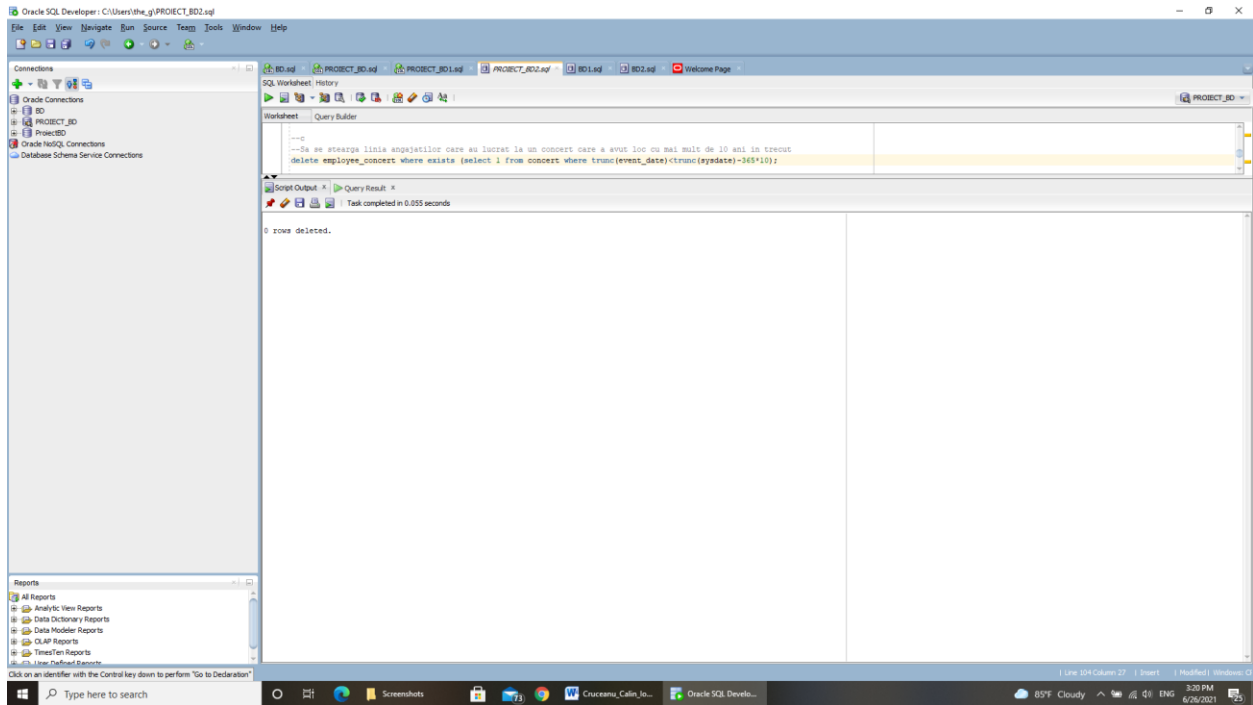


b)

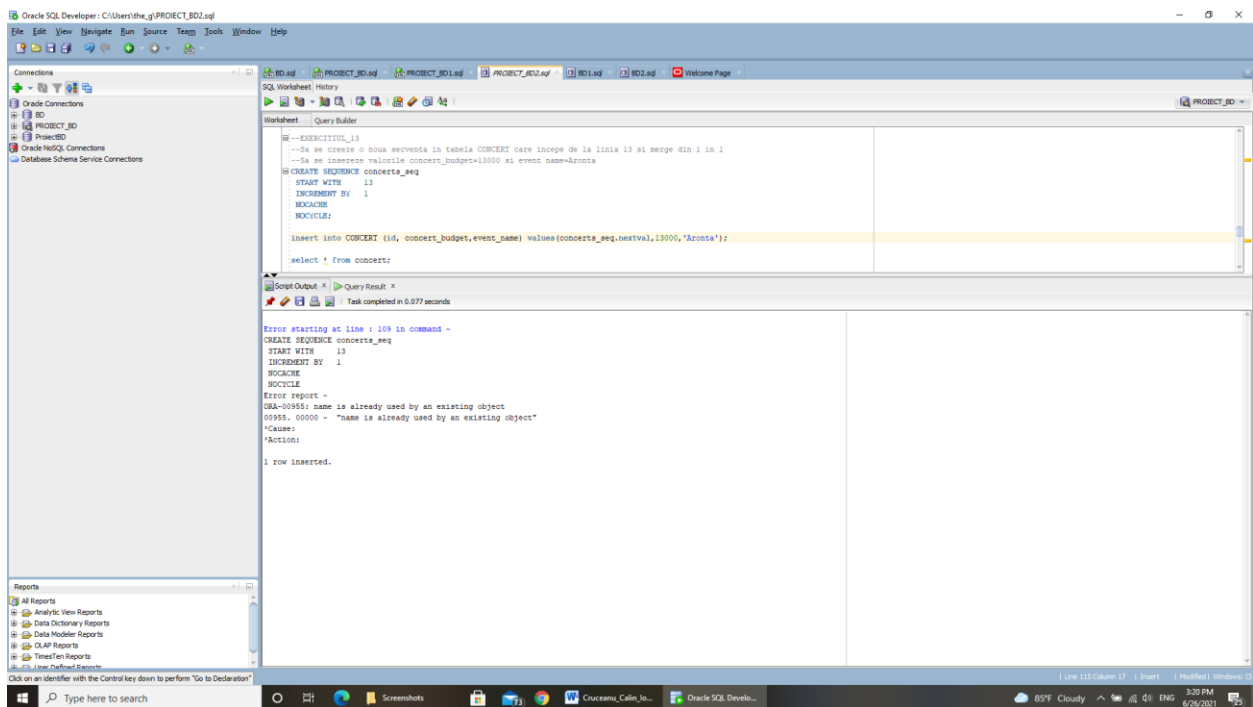


c)





13)



16)

a)

```

--EXERCITIUL_16

--a
--Sa se afiseze numele evenimentului, orasul in care are loc, adresa si pretul biletelor
--unde biletele sunt de tipul student cu ajutorul left outer join pe tabelele
--TICKET_CONCERT, CONCERT, RENT_PRICE, VENUE
--ordonat crescator in functie de numele evenimentului
select c.event_name, v.city, v.address, t.price from ticket t
left outer join ticket_concert tc on tc.ticket_id = t.id
left outer join concert c on tc.concert_id = c.id
left outer join rent_price rp on rp.concert_id = c.id
left outer join venue v on v.id = rp.venue_id
where t.type = 'student'
order by c.event_name;

```

Script Output x Query Result x


SQL | All Rows Fetched: 10 in 0.005 seconds

	EVENT_NAME	CITY	ADDRESS	PRICE
1	Ant	Zagreb	11 c	70
2	Ant	Istanbul	19 f	155
3	Castlevania	Istanbul	19 f	110
4	Deck	Samotraki	7 a	110
5	Ragan	Moskow	2 b	30
6	Ring	Odessa	33 g	30
7	Temper	Berlin	23 d	70
8	Vast	Belgrade	4 k	70
9	Vast	Athens	20 c	30
10	Wacken	Tel Aviv	15 b	155

b)

```
--b
--Sa se afiseze numele complet, salariul si salariul majorat cu 10% ale fiecarui angajat
select first_name||' '||last_name, salary, salary+salary/10 from employee;
```

Script Output x Query Result x


 SQL | All Rows Fetched: 10 in 0.002 seconds

	FIRST_NAME  ' '  LAST_NAME	SALARY	SALARY+SALARY/10
1	Andrei Bordeianu	130	143
2	Cosmin Nedelcu	300	330
3	Hongos Negyem	235	258.5
4	Mirela Voicu	115	126.5
5	Alex Pintiliasa	270	297
6	Tudor Stavar	130	143
7	Robert Moisei	300	330
8	Dragos Popovici	220	242
9	Florin Patrascu	170	187
10	Valentin Stirbu	420	462

c)

```
--c
--Sa se afiseze numele, investitia si investitia micorata cu 10% ale fiecarui sponsor
select name, investment, investment-investment/10 from sponsor;
```

Script Output x Query Result x

 SQL | All Rows Fetched: 10 in 0.002 seconds

	NAME	INVESTMENT	INVESTMENT-INVESTMENT/10
1	Corvus	100	90
2	Tronte	900	810
3	Markus	1500	1350
4	Octavian	125	112.5
5	Mannstein	100	90
6	Otto	750	675
7	Erwin	300	270