

Gestionarea unui lanț de magazine de jocuri video

Student: Anghel Fabian
Grupa 152

Cuprins

1.Descrierea modelului.....	3
2.Constrângeri	3
3.Entități	3
4.Relații	4
5.Atribute.....	6
6.Diagrama entitate-relație	10
7.Diagrama conceptuală	11
8. Scheme relaționale	12
9. Formele normale 1-3	13
10. Secvențe pentru inserare.....	15
11. Crearea tabelelor și inserarea datelor.....	16
12. Cereri.....	43
13. Actualizări și suprimări	46
14. Cereri complexe	50
15. Optimizarea unei cereri	52
16. Normalizările BCNF, FN4, FN5 și denormalizarea	55

1. Descrierea modelului

Modelul de date va gestiona informații legate de organizarea și gestionarea vânzărilor din cadrul unui lanț de magazine de jocuri video. Aceste magazine se află la o adresă în anumite orașe și anumite țări. De asemenea fiecare magazin are un stoc limitat de jocuri de care se ține evidența. Mai mulți angajați lucrează într-un magazin, iar unul dintre ei este manager în acea locație. Despre un angajat se rețin mai multe informații precum nume, salariu și date de contact.

Un joc, care are anumite specificații, are mai multe versiuni care depind de platforma pe care se află. De asemenea, un joc a fost făcut de anumiți dezvoltatori și publicat de editori. De asemenea el poate avea mai multe limbi disponibile.

Un client, despre care se rețin mai multe date de contact, poate da o comandă de mai multe jocuri de anumite versiuni. O comandă poate fi și online, fiind livrată la o anumită adresă. O comandă este procesată de un angajat.

2. Constrângeri

Modelul de date respectă anumite restricții de funcționare.

- Un magazin se află într-un singur oraș, care se află la rândul său într-o singură țară.
- Un magazin poate avea mai multe versiuni de jocuri în stoc, iar un anumit joc se poate afla în stoc în mai multe magazine.
- Un joc poate avea mai multe versiuni, iar o versiune se află pe o singură platformă.
- O platformă poate avea mai multe versiuni de jocuri.
- Un joc este făcut de unul sau mai mulți dezvoltatori, iar un dezvoltator poate face mai multe jocuri.
- Mai multe limbi sunt disponibile într-un singur joc, iar o limbă poate apărea în mai multe jocuri.
- Un client poate da mai multe comenzi, care poate conține mai multe versiuni de jocuri.
- Un angajat poate procesa mai multe comenzi, dar o comandă este procesată de un singur angajat.

3. Entități

Pentru modelul de date referitor la magazinele de jocuri video, structurile MAGAZIN, ORAȘ, ȚARĂ, STOC, JOC, JOC_ONLINE, PLATFORMĂ, VERSIUNE, DEZVOLTATOR, EDITOR, LIMBĂ, ANGAJAT, COMANDĂ, COMANDĂ_ONLINE, CLIENT.

Toate entitățile care vor fi prezentate sunt independente, cu excepția entităților dependente STOC, VERSIUNE și a subentităților JOC_ONLINE, COMANDĂ_ONLINE.

MAGAZIN = locație a lanțului de magazine de jocuri video ce conține informații legate de localizarea sa și de manager-ul său. Cheia primară a entității este id_magazin.

ORAȘ = localitate în cadrul căreia se pot afla mai multe magazine și se află într-o anumită țară. Cheia primară a entității este id_oraș.

ȚARĂ = stat în cadrul căreia se află mai multe orașe și, la rândul lor, mai multe magazine. Cheia primară a entității este id_țară.

STOC = entitate dependentă de MAGAZIN și VERSIUNE, reprezentând numărul de produse referitoare la o versiune a unui anumit joc din cadrul unui magazin. Cheia primară este compusă din id_magazin, id_joc și id_platformă.

JOC = joc video creat de anumiți developeri, de un anumit gen și tip care se poate juca pe mai multe platforme și în mai multe limbi. Cheia primară a entității este id_joc.

JOC_ONLINE = subentitate a entității JOC, ce conține informații despre activitatea și portabilitatea dintre platforme. Cheia primară a entității este id_joc.

PLATFORMĂ = mediul în care se pot rula jocurile, de diverse tipuri. Cheia primară a entității este id_platformă.

VERSIUNE = entitate dependentă de JOC și PLATFORMĂ, reprezentând o versiune a unui joc de pe o anumită platformă. Cheia primară este compusă din id_joc și id_platformă.

DEZVOLTATOR = companie specializată, care se ocupă cu crearea jocurilor video. Cheia primară a entității este id_dezvoltator.

EDITOR = companie specializată, care se ocupă cu publicarea jocurilor video. Cheia primară a entității este id_editor.

LIMBĂ = limbă în care sunt disponibile anumite jocuri. Cheia primară a entității este id_limbă.

ANGAJAT = persoană fizică responsabilă de vânzarea și de ordinea unui anumit magazin în care lucrează. Despre el se știu mai multe informații precum data de contact, nume, salariu și data angajării. Cheia primară a entității este id_angajat.

COMANDĂ = o comandă dată de un client ce conține mai multe jocuri. Cheia primară a entității este id_comandă.

COMANDĂ_ONLINE = subentitate a entității COMANDĂ, reprezentând o comandă dată prin intermediul online ce trebuie livrată la o anumită adresă. Cheia primară a entității este id_comandă.

CLIENT = persoană fizică care este un client al lanțului de magazine ce a făcut mai multe comenzi. Cheia primară a entității este id_client.

4. Relații

MAGAZIN_se_aflăîn_ORAȘ = relație dintre MAGAZIN și ORAȘ, reflectând legătura dintre acestea (în ce oraș se află un magazin). Relația are cardinalitatea minimă 1:1 (un oraș are cel puțin un magazin, un magazin se află într-un unic oraș) și cardinalitatea maximă n:1 (un oraș are mai multe magazine, un magazin se află într-un unic oraș).

ORAȘ_se_aflăîn_ȚARĂ = relație dintre ORAȘ și ȚARĂ, reflectând legătura dintre acestea (în ce țară se află un oraș). Relația are cardinalitatea minimă 1:1 (o țară are cel puțin un oraș, un oraș se află într-o unică țară) și cardinalitatea maximă n:1 (o țară are mai multe orașe, un oraș se află într-o unică țară).

MAGAZIN_are_STOC = relație dintre MAGAZIN și STOC, reflectând legătura dintre acestea (stocul de jocuri dintr-un anumit magazin). Relația are cardinalitatea minimă 1:0 (un magazin poate să nu aibă nimic în stoc, un stoc se află într-un magazin) și cardinalitatea maximă 1:n (un magazin are mai multe stocuri, un stoc se află într-un unic magazin).

VERSIUNE_se_aflăîn_STOC = relație dintre VERSIUNE și STOC, reflectând legătura dintre acestea (o versiune a unui joc se află în stocul unui magazin). Relația are cardinalitatea minimă 1:0 (o versiune poate să nu fie în stoc, un stoc are o singură versiune) și cardinalitatea maximă 1:n (o versiune poate fi în stoc în mai multe magazine, un stoc are o singură versiune de joc).

JOC_are_VERSIUNE = relație dintre JOC și VERSIUNE, reflectând legătura dintre acestea (un joc are o anumită versiune de pe o anumită platformă). Relația are cardinalitatea minimă 1:1 (un joc are cel puțin o versiune, o versiune este a unui unic joc) și cardinalitatea maximă n:1 (un joc poate avea mai multe versiuni, o versiune este a unui unic joc).

VERSIUNE_este_pe_PLATFORMĂ = relație dintre VERSIUNE și PLATFORMĂ, reflectând legătura dintre acestea (platforma unei versiuni a unui joc). Relația are cardinalitatea minimă 1:1 (o platformă are cel puțin o versiune, o versiune este pe o unică platformă) și cardinalitatea maximă n:1 (o platformă poate avea mai multe versiuni de jocuri, o versiune este pe o unică platformă).

JOC_creat_de_DEZVOLTATOR_publicat_de_EDITOR = relație de tip 3 ce leagă entitățile JOC, DEZVOLTATOR și EDITOR, reflectând cine a dezvoltat și cine a publicat un joc. Denumirea acestei relații va fi contribuie.

JOC_esteîn_LIMBĂ = relația dintre JOC și LIMBĂ, reflectând legătura dintre acestea (limbile disponibile ale unui joc). Relația are cardinalitatea minimă 1:1 (un joc este în cel puțin o limbă, o limbă apare în cel puțin un joc) și cardinalitatea maximă m:n (un joc poate fi în mai multe limbi, o limbă poate apărea în mai multe jocuri)

ANGAJAT_lucrează_la_MAGAZIN = relația dintre ANGAJAT și MAGAZIN, reflectând legătura dintre acestea (magazinul în care lucrează un angajat). Relația are cardinalitatea minimă 1:1 (un angajat lucrează într-un singur magazin, un magazin are cel puțin un angajat) și cardinalitatea maximă n:1 (un angajat lucrează într-un unic magazin, un magazin poate avea mai mulți angajați).

ANGAJAT_conduce_MAGAZIN = relația dintre ANGAJAT și MAGAZIN, reflectând legătura dintre acestea (un magazin are un manager). Relația are cardinalitatea minimă 0:1 (un angajat conduce un singur magazin, un magazin poate avea un singur manager) și cardinalitatea maximă 1:1.

CLIENT_face_COMANDĂ = relația dintre CLIENT și COMANDĂ, reflectând legătura dintre acestea (comenzile făcute de un anumit client). Relația are cardinalitatea minimă 1:0 (un client poate să nu facă nicio comandă, o comandă trebuie să fie făcută de un client) și cardinalitatea maximă 1:n (un client poate face mai multe comenzi, o comandă e făcută de un singur client).

COMANDĂ_conține_VERSIUNE = relația dintre COMANDĂ și VERSIUNE, reflectând legătura dintre acestea (versiunile ale jocurilor din cadrul unui comenzi). Relația are cardinalitatea minimă 0:1 (o comandă conține cel puțin o versiune, o versiune poate să nu fie în nicio comandă) și cardinalitatea maximă m:n (o comandă poate avea mai multe versiuni de jocuri, o versiune poate apărea în mai multe comenzi).

COMANDĂ_procesată_de_ANGAJAT = relația dintre COMANDĂ și ANGAJAT, reflectând legătura dintre acestea (comanda făcută de un client este procesată de un anumit angajat). Relația are cardinalitatea minimă 0:1 (o comandă este neapărat procesată de un angajat, un angajat poate să nu aibă comenzi) și cardinalitatea maximă n:1 (o comandă este făcută la un singur angajat, un angajat poate avea mai multe comenzi).

5. Atribute

Entitatea independentă MAGAZIN are ca atribute:

id_magazin = variabilă de tip întreg, de lungime maximă 3, care reprezintă id-ul unui magazin

nume_magazin = variabilă de tip caracter, de lungime maximă 30, care reprezintă numele magazinului

adresa = variabilă de tip caracter, de lungime maximă 100, care reprezintă adresa magazinului

id_oras = variabilă de tip întreg, de lungime maximă 3, care reprezintă id-ul orașului în care se află magazinul

id_manager = variabilă de tip întreg, de lungime maximă 4, care reprezintă id-ul managerului magazinului

Entitatea independentă ORAȘ are ca atribute:

id_oras = variabilă de tip întreg, de lungime maximă 3, care reprezintă id-ul unui oraș

nume_oras = variabilă de tip caracter, de lungime maximă 20, care reprezintă numele unui oraș

id_tara = variabilă de tip caracter, de lungime 2, care reprezintă id-ul țării în care se află orașul

Entitatea independentă ȚARĂ are ca atribute:

id_tara = variabilă de tip caracter, de lungime 2, care reprezintă id-ul unui țări

nume_tara = variabilă de tip caracter, de lungime maximă 20, care reprezintă nume țării

Entitatea dependentă STOC are ca atribute:

id_magazin = variabilă de tip întreg, de lungime maximă 3, care reprezintă id-ul unui magazin

id_joc = variabilă de tip întreg, de lungime maximă 6, care reprezintă id-ul unui joc

id_platforma = variabilă de tip întreg, de lungime maximă 2, care reprezintă id-ul unei platforme

nr_produce = variabilă de tip întreg, de lungime maximă 3, care reprezintă numărul de versiuni ale jocului disponibile în magazin

Entitatea independentă JOC are ca atribute:

id_joc = variabilă de tip întreg, de lungime maximă 6, care reprezintă id-ul unui joc

titlu = variabilă de tip caracter, de lungime maximă 30, care reprezintă numele unui joc

descriere = variabilă de tip caracter, de lungime maximă 300, care reprezintă o scurtă descriere a jocului

gen = variabilă de tip caracter, de lungime maximă 15, care reprezintă genul jocului

multiplayer = variabilă de tip întreg, luând valorile 0 sau 1, 1 reprezentând dacă jocul are și funcționalități multiplayer și 0 dacă nu

vârstă = variabilă de tip întreg, de lungime maximă 2, care reprezintă vârsta minimă pentru a juca jocul video

Subentitatea JOC ONLINE are ca atribut:

id_joc = variabilă de tip întreg, de lungime maximă 6, care reprezintă id-ul jocului online

nr_jucători = variabilă de tip întreg, de lungime maximă 8, care reprezintă activitatea online în medie din joc

crossplay = variabilă de tip întreg, luând valorile 0 sau 1, 1 reprezentând dacă jucătorii se pot juca împreună de pe platforme diferite și 0 dacă nu

cross_progression = variabilă de tip întreg, luând valorile 0 sau 1, 1 reprezentând dacă progresul unui jucător se salvează între platforme și 0 dacă nu

Entitatea independentă PLATFORMĂ are ca atribut:

id_platformă = variabilă de tip întreg, de lungime maximă 2, care reprezintă id-ul unei platforme

nume_platforma = variabilă de tip caracter, de lungime maximă 15, care reprezintă numele unei platforme

tip_platformă = variabilă de tip caracter, de lungime maximă 10, care reprezintă tipul platformei

Entitatea dependentă VERSIUNE are ca atribut:

id_joc = variabilă de tip întreg, de lungime maximă 6, care reprezintă id-ul jocului

id_platformă = variabilă de tip întreg, de lungime maximă 2, care reprezintă id-ul platformei

dimensiune = variabilă de tip real, care reprezintă dimensiunea de stocare a jocului în GB

data_lansare = variabilă de tip dată calendaristică, care reprezintă data în care s-a lansat versiunea jocului de pe o anumită platformă

preț = variabilă de tip real, care reprezintă prețul versiunii

Entitatea independentă DEZVOLTATOR are ca atribut:

id_dezvoltator = variabilă de tip întreg, de lungime maximă 3, care reprezintă id-ul dezvoltatorului

nume_dezvoltator = variabilă de tip caracter, de lungime maximă 15, care reprezintă numele dezvoltatorului

Entitatea independentă LIMBĂ are ca atribut:

id_limbă = variabilă de tip caracter, de lungime 2, care reprezintă id-ul limbii

nume_limba = variabilă de tip caracter, de lungime maximă 15, care reprezintă numele limbii

Entitatea independentă EDITOR are ca attribute:

id_editor = variabilă de tip întreg, de lungime maximă 3, care reprezintă id-ul editorului

nume_editor = variabilă de tip caracter, de lungime maximă 15 care reprezintă numele editorului

Entitatea independentă ANGAJAT are ca attribute:

id_angajat = variabilă de tip întreg, de lungime maximă 4, care reprezintă id-ul unui angajat

id_magazin = variabilă de tip întreg, de lungime maximă 3, care reprezintă id-ul magazinului în care lucrează un angajat

nume = variabilă de tip caracter, de lungime maximă 20, care reprezintă numele angajatului

prenume = variabilă de tip caracter, de lungime maximă 20, care reprezintă prenumele angajatului

email = variabilă de tip caracter, de lungime maximă 30, care reprezintă adresa de e-mail a angajatului

telefon = variabilă de tip caracter, de lungime maximă 11, care reprezintă numărul de telefon al angajatului

salariu = variabilă de tip întreg, de lungime maximă 5, care reprezintă salariul lunar al angajatului

data_angajare = variabilă de tip dată calendaristică, care reprezintă data în care a fost angajată persoana în magazin

Entitatea independentă CLIENT are ca attribute:

id_client = variabilă de tip întreg, de lungime maximă 7, care reprezintă id-ul clientului

nume = variabilă de tip caracter, de lungime maximă 20, care reprezintă numele clientului

prenume = variabilă de tip caracter, de lungime maximă 20, care reprezintă prenumele clientului

email = variabilă de tip caracter, de lungime maximă 30, care reprezintă adresa de e-mail a clientului

telefon = variabilă de tip caracter, de lungime maximă 11, care reprezintă numărul de telefon al clientului

Entitatea independentă COMANDĂ are ca attribute:

id_comandă = variabilă de tip întreg, de lungime maximă 10, care reprezintă id-ul comenzii

id_client = variabilă de tip întreg, de lungime maximă 7, care reprezintă id-ul clientului care a făcut comanda

id_angajat = variabilă de tip întreg, de lungime maximă 4, care reprezintă id-ul angajatului care a procesat comanda

dată_comandă = variabilă de tip dată calendaristică, care reprezintă data la care a fost efectuată comanda

metodă_plată = variabilă de tip caracter, de lungime maximă 10, care reprezintă modalitatea de plată a comenzii

Subentitatea COMANDĂ_ONLINE are ca atribute:

id_comandă = variabilă de tip întreg, de lungime maximă 10, care reprezintă id-ul comenzii online

adresă = variabilă de tip caracter, de lungime maximă 100, care reprezintă adresa la care e livrată comanda online

data_livrare = variabilă de tip dată calendaristică, care reprezintă data la care se livrează comanda

Relația JOC_esteîn_LIMBĂ are ca atribute:

id_joc = variabilă de tip întreg, de lungime maximă 6, care reprezintă id-ul jocului

id_limbă = variabilă de tip caracter, de lungime 2, care reprezintă id-ul limbii

dublare = variabilă de tip întreg, luând valorile 0 sau 1, 1 reprezentând faptul că jocul prezintă dublare în limba respectivă și 0 dacă nu

Relația COMANDĂ_conține_VERSIUNE are ca atribute:

id_joc = variabilă de tip întreg, de lungime maximă 6, care reprezintă id-ul jocului

id_platformă = variabilă de tip întreg, de lungime maximă 2, care reprezintă id-ul platformei

id_comandă = variabilă de tip întreg, de lungime maximă 10, care reprezintă id-ul comenzii

număr = variabilă de tip întreg, de lungime maximă 2, care reprezintă numărul de jocuri de același tip din comandă

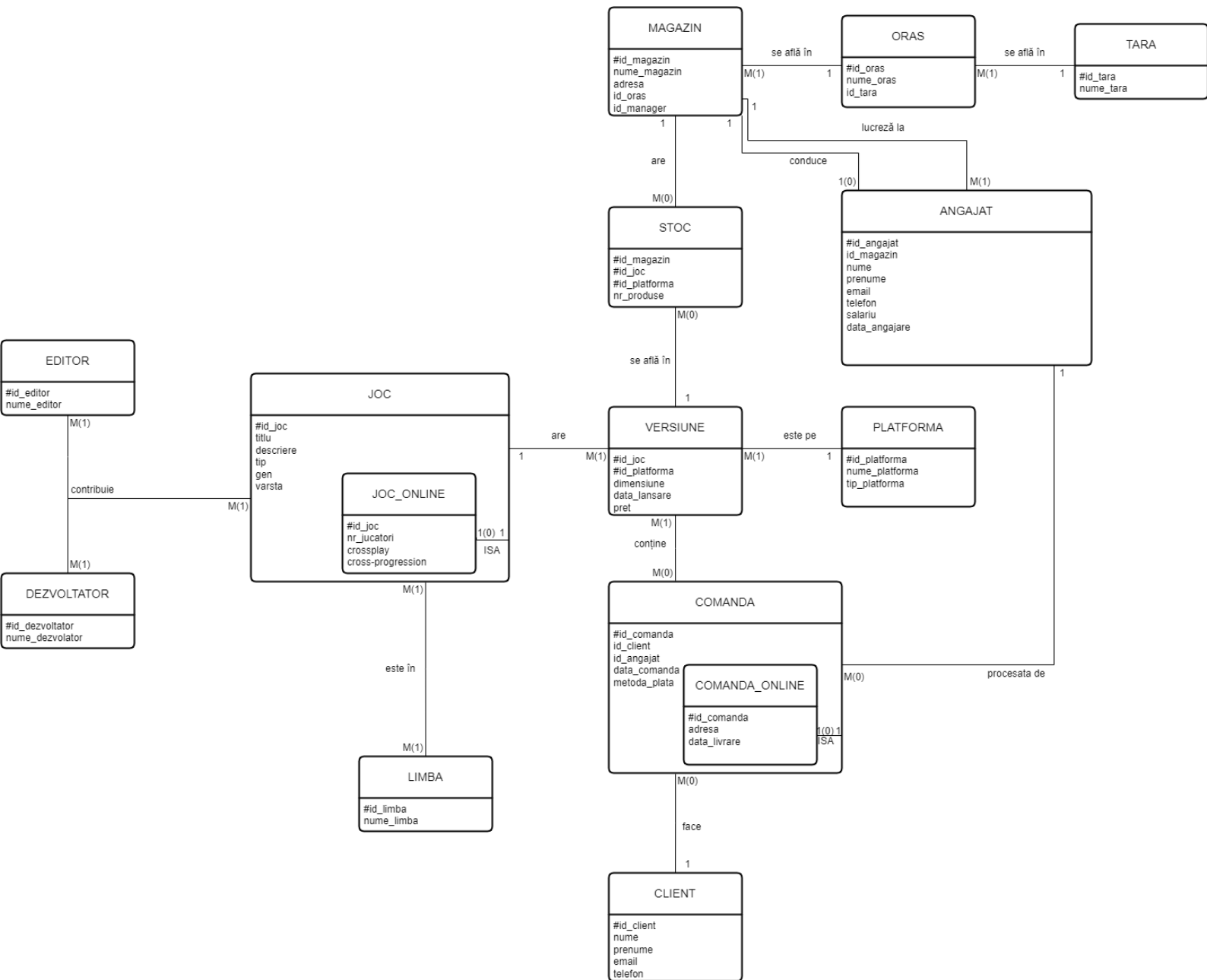
Relația JOC_creat_de_DEZVOLTATOR_publicat_de_EDITOR are ca atribute:

id_joc = variabilă de tip întreg, de lungime maximă 6, care reprezintă id-ul jocului

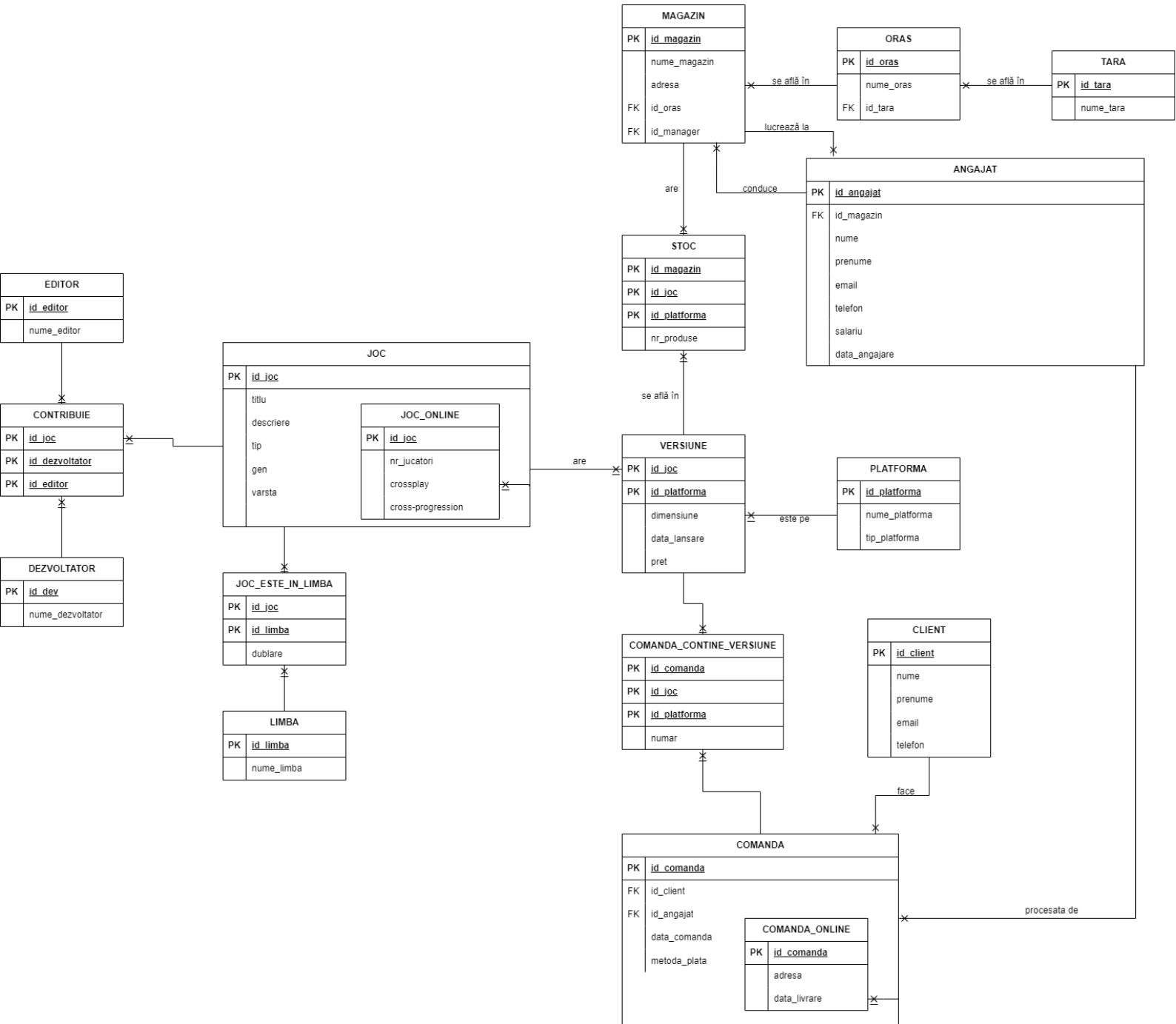
id_dezvoltator = variabilă de tip întreg, de lungime maximă 3, care reprezintă id-ul dezvoltatorului

id_editor = variabilă de tip întreg, de lungime maximă 3, care reprezintă id-ul editorului

6. Diagrama entitate-relație



7. Diagrama conceptuală



8. Scheme relațională

Schemele relaționale sunt:

TARA (id_tara#, nume_tara)

ORAS (id_oras#, nume_oras, id_tara)

MAGAZIN (id_magazin#, nume_magazin, adresa, id_oras, id_manager)

STOC (id_magazin#, id_joc#, id_platforma#, nr_produce)

JOC (id_joc#, titlu, descriere, gen, multiplayer, varsta)

JOC_ONLINE (id_joc#, nr_jucatori, crossplay, cross_progression)

PLATFORMA (id_platforma#, nume_platforma, tip_platforma)

VERSIUNE (id_joc#, id_platforma#, dimensiune, data_lansare, pret)

DEZVOLTATOR (id_dezvoltator#, nume_dezvoltator)

EDITOR (id_editor#, nume_editor)

CONTRIBUIE (id_joc#, id_dezvoltator#, id_editor#)

LIMBA (id_limba#, nume_limba)

JOC_IN_LIMBA (id_joc#, id_limba#, dublare)

ANGAJAT (id_angajat#, id_magazin, nume, prenume, email, telefon, salariu, data_angajare)

COMANDA (id_comanda#, id_client, id_angajat, data_comanda, metoda_plata)

COMANDA_ONLINE (id_comanda#, adresa, data_livrare)

COMANDA_CONTINE (id_joc#, id_platforma#, id_comanda#, numar)

CLIENT (id_client#, nume, prenume, email, telefon)

9. Formele normale 1-3

9.1 Forma normală 1

Proiectul se află în prima formă normală deoarece în fiecare relație unui atribut îi corespunde o valoare indivizibilă.

Dacă proiectul nu ar fi în FN1, atunci relația JOC_IN_LIMBA ar arăta astfel:

id_joc#	id_limba
40741	EN, JP, IT, SP
101000	EN, JP
104000	EN, JP, RO, IT, SP
107000	EN

Pentru fiecare intrare în tabel, pentru atributul id_limba sunt mai multe valori prezente.

Pentru ca tabelul să fie în FN1, atunci și atributul id_limba ar deveni cheie primară, tabelul arătând astfel:

id_joc#	id_limba
40741	EN
40741	JP
40741	IT
40741	SP
101000	EN
101000	JP
104000	EN
104000	JP
104000	RO
104000	IT
104000	SP
107000	EN

9.2 Forma normală 2

Proiectul se află în a doua formă normală deoarece se află în FN1, iar fiecare atribut care nu este cheie este dependent de întreaga cheie primară.

Dacă proiectul nu ar fi în FN2, relația VERSIUNE ar arăta astfel:

id_joc#	titlu	id_platforma#	dimensiune	data_lansare	pret
25200	Soul Hackers 2	1	20.5	26-AUG-22	60
25200	Soul Hackers 2	2	20.5	26-AUG-22	60
25200	Soul Hackers 2	4	20.5	26-AUG-22	60
100290	Nier:Automata	1	40.8	17-MAR-17	40
100290	Nier:Automata	2	38	23-FEB-17	40
100290	Nier:Automata	4	38	26-JUN-18	40
100290	Nier:Automata	7	15	26-JUN-18	60
100290	Nier:Automata	8	40.8	17-MAR-17	40
101000	Shin Megami Tensei V	7	16	12-NOV-21	60

Relația se află în FN1 deoarece fiecare atribut are asociat o singură valoare.

Atributele dimensiune, data_lansare si pret depind de întreaga cheie primară, și anume id_joc și id_platformă. În schimb se observă o dependență directă între id_joc și titlu, însemnând că titlu depinde doar de o parte a cheii primare, și anume doar de id_joc. Se aplică regula Casey-Dorobel și astfel o să avem un nou tabel joc care conține id_joc ca și cheie primară si titlul jocului.

Versiune:

id_joc#	id_platforma#	dimensiune	data_lansare	pret
25200	1	20.5	26-AUG-22	60
25200	2	20.5	26-AUG-22	60
25200	4	20.5	26-AUG-22	60
100290	1	40.8	17-MAR-17	40
100290	2	38	23-FEB-17	40
100290	4	38	26-JUN-18	40
100290	7	15	26-JUN-18	60
100290	8	40.8	17-MAR-17	40
101000	7	16	12-NOV-21	60

Joc:

id_joc#	titlu
25200	Soul Hackers 2
100290	Nier:Automata
101000	Shin Megami Tensei V

9.3 Forma normală 3

Proiectul se află în a treia formă normală deoarece se află în FN2, iar fiecare atribut care nu este cheie depinde direct de cheia primară.

Dacă proiectul nu ar fi în FN3, relația COMANDA ar arăta astfel:

id_comanda#	id_client	telefon	id_angajat	data_comanda	metoda_plata
1000000000	1000000	+40763192452	1001	11-MAY-23	cash
1037182341	1000000	+40763192452	1002	09-MAY-23	card
1074364682	1000001	+40763693212	1005	05-MAY-23	rate
1371823410	1000001	+40763693212	1018	07-MAY-23	cash
1148729364	1000003	(null)	1015	03-MAY-23	card

Relația se află în FN2 deoarece fiecare atribut care nu este cheie depinde de unica cheie primară id_comanda. În schimb se observă că atributul telefon depinde de id_client, care nu este cheie primară, rezultând că telefon nu depinde direct de id_comanda, deci relația nu se află în FN3. Se aplică regula Casey-Dorobel și se creează tabelul client care are ca atribute id_client și telefon, id_client fiind cheie primară.

Comanda:

id_comanda#	id_client	id_angajat	data_comanda	metoda_plata
1000000000	1000000	1001	11-MAY-23	cash
1037182341	1000000	1002	09-MAY-23	card
1074364682	1000001	1005	05-MAY-23	rate
1371823410	1000001	1018	07-MAY-23	cash
1148729364	1000003	1015	03-MAY-23	card

Client:

id_client#	telefon
1000000	+40763192452
1000001	+40763693212
1000003	(null)

10. Secvențe pentru inserare

CREATE SEQUENCE secventa_client

START WITH 1000000

INCREMENT BY 1;

CREATE SEQUENCE secventa_comanda

START WITH 1000000000

INCREMENT BY 37182341

MINVALUE 1000000000

```
CREATE SEQUENCE secventa_client
START WITH 1000000
INCREMENT BY 1;
```

```
CREATE SEQUENCE secventa_comanda
START WITH 1000000000
INCREMENT BY 37182341
MINVALUE 1000000000
MAXVALUE 9999999999
CYCLE;
```

Sequence SECVENTA_CLIENT created.

Sequence SECVENTA_COMANDA created.

11. Crearea tabelelor și inserarea datelor

```
CREATE TABLE tara(  
    id_tara varchar2(2) CONSTRAINT tara_pk PRIMARY KEY,  
    nume_tara varchar2(30) CONSTRAINT tara_nume_nn NOT NULL  
        CONSTRAINT tara_nume_unq UNIQUE  
);
```

```
CREATE TABLE oras(  
    id_oras number(3) CONSTRAINT oras_pk PRIMARY KEY,  
    nume_oras varchar2(20) CONSTRAINT oras_nume_nn NOT NULL,  
    id_tara varchar2(2) CONSTRAINT oras_fk REFERENCES tara(id_tara) ON DELETE SET  
NULL  
);
```

```
CREATE TABLE magazin(  
    id_magazin number(3) CONSTRAINT mag_pk PRIMARY KEY,  
    nume_magazin varchar2(30),  
    adresa varchar2(100),  
    id_oras CONSTRAINT mag_fk_oras REFERENCES oras(id_oras) ON DELETE SET NULL,  
    id_manager number(4)  
);
```

```
CREATE TABLE joc(  
    id_joc number(6) CONSTRAINT joc_pk PRIMARY KEY,  
    titlu varchar2(30) CONSTRAINT joc_titlu_nn NOT NULL,  
    descriere varchar2(300),  
    gen varchar2(15),  
    multiplayer number(1) CONSTRAINT joc_mp_chk CHECK(multiplayer = 0 or multiplayer = 1),  
    varsta number(2) DEFAULT 3  
);
```

```
CREATE TABLE joc_online(  
    id_joc number(6) CONSTRAINT joc_on_pk PRIMARY KEY
```



```

        CONSTRAINT joc_on_fk REFERENCES joc(id_joc) ON DELETE CASCADE,
nr_jucatori number(8) CONSTRAINT joc_on_juc_chk CHECK(nr_jucatori > 0),
crossplay number(1) CONSTRAINT joc_cplay_chk CHECK(crossplay = 0 or crossplay = 1),
cross_progression number(1) CONSTRAINT joc_cprog_chk CHECK(cross_progression = 0 or
cross_progression = 1)
);

```

```

CREATE TABLE platforma(
    id_platforma number(2) CONSTRAINT plat_pk PRIMARY KEY,
    nume_platforma varchar2(15) CONSTRAINT plat_nume_nn NOT NULL
        CONSTRAINT plat_nume_unq UNIQUE,
    tip_platforma varchar2(10) CONSTRAINT plat_tip_nn NOT NULL
);

```

```

CREATE TABLE versiune(
    id_joc number(6) CONSTRAINT vers_joc_fk REFERENCES joc(id_joc) ON DELETE
CASCADE,
    id_platforma number(2) CONSTRAINT vers_plat_fk REFERENCES platforma(id_platforma)
ON DELETE CASCADE,
    dimensiune number(4,2),
    data_lansare date,
    pret number(4,2) CONSTRAINT vers_pret_nn NOT NULL,
    CONSTRAINT vers_pk PRIMARY KEY(id_joc, id_platforma)
);

```

```

CREATE TABLE stoc(
    id_magazin number(3) CONSTRAINT stoc_mag_fk REFERENCES magazin(id_magazin) ON
DELETE CASCADE,
    id_joc number(6),
    id_platforma number(2),
    nr_produce number(3) CONSTRAINT stoc_nr_nn NOT NULL,
    CONSTRAINT stoc_pk PRIMARY KEY(id_magazin, id_joc, id_platforma),
    CONSTRAINT stoc_vers_fk FOREIGN KEY(id_joc, id_platforma) REFERENCES
versiune(id_joc, id_platforma) ON DELETE CASCADE

```

);

CREATE TABLE dezvoltator(

id_dezvoltator number(3) CONSTRAINT dezv_pk PRIMARY KEY,

nume_dezvoltator varchar2(15) CONSTRAINT dezv_nume_unq UNIQUE

);

CREATE TABLE editor(

id_editor number(3) CONSTRAINT editor_pk PRIMARY KEY,

nume_editor varchar2(15) CONSTRAINT editor_nume_unq UNIQUE

);

CREATE TABLE contribuie(

id_joc number(6) CONSTRAINT contr_joc_fk REFERENCES joc(id_joc) ON DELETE CASCADE,

id_dezvoltator number(3) CONSTRAINT contr_dezv_fk REFERENCES dezvoltator(id_dezvoltator) ON DELETE CASCADE,

id_editor number(3) CONSTRAINT contr_edit_fk REFERENCES editor(id_editor) ON DELETE CASCADE,

CONSTRAINT contr_pk PRIMARY KEY(id_joc, id_dezvoltator, id_editor)

);

CREATE TABLE limba(

id_limba varchar2(2) CONSTRAINT limba_pk PRIMARY KEY,

nume_limba varchar2(15) CONSTRAINT limba_nume_unq UNIQUE

);

CREATE TABLE joc_in_limba(

id_joc number(6) CONSTRAINT joc_lim_fk_joc REFERENCES joc(id_joc) ON DELETE CASCADE,

id_limba varchar2(2) CONSTRAINT joc_lim_fk_lim REFERENCES limba(id_limba) ON DELETE CASCADE,

dublare number(1) CONSTRAINT joc_lim_dub_chk CHECK(dublare = 0 or dublare = 1),

CONSTRAINT joc_lim_pk PRIMARY KEY(id_joc, id_limba)

);

CREATE TABLE angajat(

id_angajat number(4) CONSTRAINT ang_pk PRIMARY KEY,

id_magazin number(3) CONSTRAINT ang_fk REFERENCES magazin(id_magazin)

CONSTRAINT ang_mag_nn NOT NULL,

nume varchar2(20) CONSTRAINT ang_nume_nn NOT NULL,

prenume varchar2(20),

email varchar2(30),

telefon varchar2(12),

salariu number(5) CONSTRAINT ang_sal_chk CHECK (salariu > 500),

data_angajare date DEFAULT SYSDATE

);

CREATE TABLE client(

id_client number(7) CONSTRAINT clnt_pk PRIMARY KEY,

nume varchar2(20),

prenume varchar2(20),

email varchar2(30),

telefon varchar2(12)

);

CREATE TABLE comanda(

id_comanda number(10) CONSTRAINT comd_pk PRIMARY KEY,

id_client number(7) CONSTRAINT comd_clnt_fk REFERENCES client(id_client) ON DELETE SET NULL,

id_angajat number(4) CONSTRAINT comd_ang_fk REFERENCES angajat(id_angajat) ON DELETE SET NULL,

data_comanda date DEFAULT SYSDATE,

metoda_plata varchar2(10)

);

CREATE TABLE comanda_online(

```

        id_comanda number(10) CONSTRAINT comd_on_pk PRIMARY KEY
        CONSTRAINT comd_on_fk REFERENCES comanda(id_comanda) ON DELETE
CASCADE,
        adresa varchar2(100),
        data_livrare date
    );

```

```

CREATE TABLE comanda_contine(
    id_joc number(6),
    id_platforma number(2),
    id_comanda    number(10)    CONSTRAINT    comd_cont_fk_comd    REFERENCES
comanda(id_comanda) ON DELETE CASCADE,
    numar number(2) CONSTRAINT comd_cont_chk CHECK(numar > 0),
    CONSTRAINT comd_cont_fk_ver FOREIGN KEY(id_joc, id_platforma) REFERENCES
versiune(id_joc, id_platforma) ON DELETE CASCADE,
    CONSTRAINT comd_cont_pk PRIMARY KEY(id_joc, id_platforma, id_comanda)
);

```

```

ALTER TABLE magazin
ADD CONSTRAINT mag_fk_mng FOREIGN KEY(id_manager)
REFERENCES angajat(id_angajat);

```

Table TARA created.	Table STOC created.	Table CLIENT created.
Table ORAS created.	Table DEZVOLTATOR created.	Table COMANDA created.
Table MAGAZIN created.	Table EDITOR created.	Table COMANDA_ONLINE created.
Table JOC created.	Table CONTRIBUIE created.	Table COMANDA_CONTINE created.
Table JOC_ONLINE created.	Table LIMBA created.	Table MAGAZIN altered.
Table PLATFORMA created.	Table JOC_IN_LIMBA created.	
Table VERSIUNE created.	Table ANGAJAT created.	

```

CREATE TABLE tara(
    id_tara varchar2(2) CONSTRAINT tara_pk PRIMARY KEY,
    nume_tara varchar2(30) CONSTRAINT tara_nume_nn NOT NULL
                                CONSTRAINT tara_nume_unq UNIQUE
);

CREATE TABLE oras(
    id_oras number(3) CONSTRAINT oras_pk PRIMARY KEY,
    nume_oras varchar2(20) CONSTRAINT oras_nume_nn NOT NULL,
    id_tara varchar2(2) CONSTRAINT oras_fk REFERENCES tara(id_tara) ON DELETE SET NULL
);

CREATE TABLE magazin(
    id_magazin number(3) CONSTRAINT mag_pk PRIMARY KEY,
    nume_magazin varchar2(30),
    adresa varchar2(100),
    id_oras CONSTRAINT mag_fk_oras REFERENCES oras(id_oras) ON DELETE SET NULL,
    id_manager number(4)
);

CREATE TABLE joc(
    id_joc number(6) CONSTRAINT joc_pk PRIMARY KEY,
    titlu varchar2(30) CONSTRAINT joc_titlu_nn NOT NULL,
    descriere varchar2(300),
    gen varchar2(15),
    multiplayer number(1) CONSTRAINT joc_mp_chk CHECK(multiplayer = 0 or multiplayer = 1),
    varsta number(2) DEFAULT 3
);

CREATE TABLE joc_online(
    id_joc number(6) CONSTRAINT joc_on_pk PRIMARY KEY
                                CONSTRAINT joc_on_fk REFERENCES joc(id_joc) ON DELETE CASCADE,
    nr_jucatori number(8) CONSTRAINT joc_on_juc_chk CHECK(nr_jucatori > 0),
    crossplay number(1) CONSTRAINT joc_cplay_chk CHECK(crossplay = 0 or crossplay = 1),
    cross_progression number(1) CONSTRAINT joc_cprog_chk CHECK(cross_progression = 0 or cross_progression = 1)
);

CREATE TABLE platforma(
    id_platforma number(2) CONSTRAINT plat_pk PRIMARY KEY,
    nume_platforma varchar2(15) CONSTRAINT plat_nume_nn NOT NULL
                                CONSTRAINT plat_nume_unq UNIQUE,
    tip_platforma varchar2(10) CONSTRAINT plat_tip_nn NOT NULL
);

CREATE TABLE versiune(
    id_joc number(6) CONSTRAINT vers_joc_fk REFERENCES joc(id_joc) ON DELETE CASCADE,
    id_platforma number(2) CONSTRAINT vers_plat_fk REFERENCES platforma(id_platforma) ON DELETE CASCADE,
    dimensiune number(4,2),
    data_lansare date,
    pret number(4,2) CONSTRAINT vers_pret_nn NOT NULL,
    CONSTRAINT vers_pk PRIMARY KEY(id_joc, id_platforma)
);

```

```

CREATE TABLE stoc(
    id_magazin number(3) CONSTRAINT stoc_mag_fk REFERENCES magazin(id_magazin) ON DELETE CASCADE,
    id_joc number(6),
    id_platforma number(2),
    nr_produce number(3) CONSTRAINT stoc_nr_nn NOT NULL,
    CONSTRAINT stoc_pk PRIMARY KEY(id_magazin, id_joc, id_platforma),
    CONSTRAINT stoc_vers_fk FOREIGN KEY(id_joc, id_platforma) REFERENCES versiune(id_joc, id_platforma) ON DELETE CASCADE
);

CREATE TABLE dezvoltator(
    id_dezvoltator number(3) CONSTRAINT dezvolt_pk PRIMARY KEY,
    nume_dezvoltator varchar2(15) CONSTRAINT dezvolt_nume_unq UNIQUE
);

CREATE TABLE editor(
    id_editor number(3) CONSTRAINT editor_pk PRIMARY KEY,
    nume_editor varchar2(15) CONSTRAINT editor_nume_unq UNIQUE
);

CREATE TABLE contribuie(
    id_joc number(6) CONSTRAINT contr_joc_fk REFERENCES joc(id_joc) ON DELETE CASCADE,
    id_dezvoltator number(3) CONSTRAINT contr_dezv_fk REFERENCES dezvoltator(id_dezvoltator) ON DELETE CASCADE,
    id_editor number(3) CONSTRAINT contr_edit_fk REFERENCES editor(id_editor) ON DELETE CASCADE,
    CONSTRAINT contr_pk PRIMARY KEY(id_joc, id_dezvoltator, id_editor)
);

CREATE TABLE limba(
    id_limba varchar2(2) CONSTRAINT limba_pk PRIMARY KEY,
    nume_limba varchar2(15) CONSTRAINT limba_nume_unq UNIQUE
);

CREATE TABLE joc_in_limba(
    id_joc number(6) CONSTRAINT joc_lim_fk_joc REFERENCES joc(id_joc) ON DELETE CASCADE,
    id_limba varchar2(2) CONSTRAINT joc_lim_fk_lim REFERENCES limba(id_limba) ON DELETE CASCADE,
    dublare number(1) CONSTRAINT joc_lim_dub_chk CHECK(dublare = 0 or dublare = 1),
    CONSTRAINT joc_lim_pk PRIMARY KEY(id_joc, id_limba)
);

CREATE TABLE angajat(
    id_angajat number(4) CONSTRAINT ang_pk PRIMARY KEY,
    id_magazin number(3) CONSTRAINT ang_fk REFERENCES magazin(id_magazin)
        CONSTRAINT ang_mag_nn NOT NULL,
    nume varchar2(20) CONSTRAINT ang_nume_nn NOT NULL,
    prenume varchar2(20),
    email varchar2(30),
    telefon varchar2(12),
    salariu number(5) CONSTRAINT ang_sal_chk CHECK(salariu > 500),
    data_angajare date DEFAULT SYSDATE
);

CREATE TABLE client(
    id_client number(7) CONSTRAINT clnt_pk PRIMARY KEY,
    nume varchar2(20),
    prenume varchar2(20),
    email varchar2(30),
    telefon varchar2(12)
);

CREATE TABLE comanda(
    id_comanda number(10) CONSTRAINT comd_pk PRIMARY KEY,
    id_client number(7) CONSTRAINT comd_clnt_fk REFERENCES client(id_client) ON DELETE SET NULL,
    id_angajat number(4) CONSTRAINT comd_ang_fk REFERENCES angajat(id_angajat) ON DELETE SET NULL,
    data_comanda date DEFAULT SYSDATE,
    metoda_plata varchar2(10)
);

CREATE TABLE comanda_online(
    id_comanda number(10) CONSTRAINT comd_on_pk PRIMARY KEY
        CONSTRAINT comd_on_fk REFERENCES comanda(id_comanda) ON DELETE CASCADE,
    adresa varchar2(100),
    data_livrare date
);

CREATE TABLE comanda_contine(
    id_joc number(6),
    id_platforma number(2),
    id_comanda number(10) CONSTRAINT comd_cont_fk_comd REFERENCES comanda(id_comanda) ON DELETE CASCADE,
    numar number(2) CONSTRAINT comd_cont_chk CHECK(numar > 0),
    CONSTRAINT comd_cont_fk_ver FOREIGN KEY(id_joc, id_platforma) REFERENCES versiune(id_joc, id_platforma) ON DELETE CASCADE,
    CONSTRAINT comd_cont_pk PRIMARY KEY(id_joc, id_platforma, id_comanda)
);

ALTER TABLE magazin
ADD CONSTRAINT mag_fk_mng FOREIGN KEY(id_manager)
REFERENCES angajat(id_angajat);

```

```
INSERT ALL
INTO tara VALUES ('RO', 'Romania')
INTO tara VALUES ('NL', 'Olanda')
INTO tara VALUES ('GB', 'Marea Britanie')
INTO tara VALUES ('IT', 'Italia')
INTO tara VALUES ('HU', 'Ungaria')
SELECT * FROM DUAL;
```

```
INSERT ALL
INTO oras VALUES(11, 'Bucuresti', 'RO')
INTO oras VALUES(12, 'Galati', 'RO')
INTO oras VALUES(13, 'Cluj', 'RO')
INTO oras VALUES(14, 'Amsterdam', 'NL')
INTO oras VALUES(15, 'Brasov', 'RO')
INTO oras VALUES(16, 'Londra', 'GB')
INTO oras VALUES(17, 'Roma', 'IT')
INTO oras VALUES(18, 'Rotterdam', 'NL')
INTO oras VALUES(19, 'Budapesta', 'HU')
INTO oras VALUES(20, 'Milano', 'IT')
SELECT * FROM DUAL;
```

```
INSERT ALL
INTO magazin
VALUES(300, 'Gameshop AFI', 'Bulevardul General Paul Teodorescu 4', 11, NULL)
INTO magazin
VALUES(301, 'Gameshop Centru Vechi', 'Strada Lipscani 55, Bucure?ti', 11, NULL)
INTO magazin
VALUES(302, 'Gamecorner Tiglina', 'Strada Br?ilei nr. 163, 800309', 12, NULL)
INTO magazin
VALUES(303, 'Gamecorner Gorjului', 'Bulevardul Iuliu Maniu 67', 11, NULL)
INTO magazin
```

```

VALUES(304, 'Gameshop Cluj', 'Strada Alexandru Vaida Voevod 59, 400436', 13, NULL)
INTO magazin

VALUES(305, 'Gameshop Amsterdam', NULL, 14, NULL)
INTO magazin

VALUES(306, 'Megagame Brasov', 'Bulevardul 15 Noiembrie 78', 15, NULL)
INTO magazin

VALUES(307, 'Gamecorner Walworth', 'East St, London SE17 1EL', 16, NULL)
INTO magazin

VALUES(308, 'Gameshop Roma', NULL, 17, NULL)
INTO magazin

VALUES(309, 'Gameshop Rotterdam', 'Willem Ruyslaan 225, 3063 ER', 18, NULL)
INTO magazin

VALUES(310, 'Megashop Londra', '12 Walbrook, London EC4N 8AA', 16, NULL)
INTO magazin

VALUES(311, 'Gamecorner Budapesta 1', 'Maglódi út 14/B, 1106', 19, NULL)
INTO magazin

VALUES(312, 'Gamecorner Potcoava', 'Strada Brailei, Nr.17, Complex Comercial Potcoava de Aur',
12, NULL)
INTO magazin

VALUES(313, 'Gameshop Milano', 'Via Lorenteggio, 219, 20147 Milano MI', 20, NULL)
INTO magazin

VALUES(314, 'Gamecorner Budapesta 2', 'Könyves Kálmán krt. 12-14, 1097', 19, NULL)
SELECT * FROM DUAL;

INSERT ALL

INTO joc

VALUES(100290, 'Nier: Automata', 'Nier: Automata spune povestea androizilor 2B, 9S si A2 si
lupta lor pentru recapatarea distopiei controlate de masinarii puternice.', 'Action JRPG', 0, 18)
INTO joc

VALUES(40741, 'Persona 5 Royal', 'Scoate-ti masca si alatura-te Hotilor Fantoma si furturilor lor
magnifice, cum infiltreaza mintilor oamenilor corupti, facandu-i sa-si schimbe caile.', 'Turn-based
JRPG', 0, 16)
INTO joc

VALUES(40742, 'Persona 5 Strikers', NULL, 'Hack and Slash', 0, 16)

```


INTO joc

VALUES(460000, 'Rainbow Six: Siege', 'Rainbow Six: Siege este un shooter tactic de elita pe echipe, unde planificarile superioare si executiile trimfa.', 'Team Shooter', 1, 18)

INTO joc

VALUES(101000, 'Shin Megami Tensei V', NULL, 'Turn-based JRPG', 0, 16)

INTO joc

VALUES(102000, 'Counter Strike 2', NULL, 'Team Shooter', 1, 18)

INTO joc

VALUES(103000, 'Brawl Stars', NULL, 'Topdown Shooter', 1, 7)

INTO joc

VALUES(104000, 'Minecraft Bedrock', NULL, 'Sandbox', 1, 7)

INTO joc

VALUES(105000, 'Ghost Recon: Wildlands', NULL, 'Open World', 1, 18)

INTO joc

VALUES(106000, 'Cyberpunk 2077', NULL, 'Action RPG', 0, 18)

INTO joc

VALUES(107000, 'The Witcher 3', NULL, 'Open World', 0, 18)

INTO joc

VALUES(108000, 'Geometry Dash', NULL, 'Platformer', 0, 3)

INTO joc

VALUES(25200, 'Soul Hackers 2', NULL, 'Turn-based JRPG', 0, 16)

INTO joc

VALUES(109000, 'Roblox', NULL, 'Sandbox', 1, 3)

SELECT * FROM DUAL;

INSERT ALL

INTO joc_online VALUES(460000, 3800000, 1, 1)

INTO joc_online VALUES(102000, 10000000, 0, 0)

INTO joc_online VALUES(103000, 1000000, 1, 1)

INTO joc_online VALUES(104000, 30000000, 1, 0)

INTO joc_online VALUES(105000, 20000, 1, 1)

INTO joc_online VALUES(109000, 17800000, 1, 1)

SELECT * FROM DUAL;

```

INSERT ALL
INTO platforma VALUES(1, 'Windows', 'PC')
INTO platforma VALUES(2, 'Playstation 4', 'Consola')
INTO platforma VALUES(3, 'Playstation 5', 'Consola')
INTO platforma VALUES(4, 'Xbox One', 'Consola')
INTO platforma VALUES(5, 'Xbox Series X', 'Consola')
INTO platforma VALUES(6, 'Xbox Series S', 'Consola')
INTO platforma VALUES(7, 'Nintendo Switch', 'Handheld')
INTO platforma VALUES(8, 'Steam Deck', 'Handheld')
INTO platforma VALUES(9, 'Android', 'Telefon')
INTO platforma VALUES(10, 'IOS', 'Telefon')
INTO platforma VALUES(11, 'Stadia', 'Cloud')
INTO platforma VALUES(12, 'Linux', 'PC')
SELECT * FROM DUAL;

```

```

INSERT ALL
INTO versiune VALUES(100290, 1, 40.8, TO_DATE('17-03-2017', 'dd-mm-yyyy'), 40)
INTO versiune VALUES(100290, 2, 38, TO_DATE('23-02-2017', 'dd-mm-yyyy'), 40)
INTO versiune VALUES(100290, 4, 38, TO_DATE('26-06-2018', 'dd-mm-yyyy'), 40)
INTO versiune VALUES(100290, 7, 15, TO_DATE('06-10-2022', 'dd-mm-yyyy'), 60)
INTO versiune VALUES(100290, 8, 40.8, TO_DATE('17-03-2017', 'dd-mm-yyyy'), 40)
INTO versiune VALUES(40741, 1, 39.3, TO_DATE('21-10-2022', 'dd-mm-yyyy'), 60)
INTO versiune VALUES(40741, 2, 33.2, TO_DATE('31-03-2020', 'dd-mm-yyyy'), 40)
INTO versiune VALUES(40741, 3, 39.3, TO_DATE('21-10-2022', 'dd-mm-yyyy'), 60)
INTO versiune VALUES(40741, 4, 39.3, TO_DATE('21-10-2022', 'dd-mm-yyyy'), 60)
INTO versiune VALUES(40741, 5, 39.3, TO_DATE('21-10-2022', 'dd-mm-yyyy'), 60)
INTO versiune VALUES(40741, 6, 39.3, TO_DATE('21-10-2022', 'dd-mm-yyyy'), 60)
INTO versiune VALUES(40741, 7, 21.1, TO_DATE('21-10-2022', 'dd-mm-yyyy'), 60)
INTO versiune VALUES(40741, 8, 39.3, TO_DATE('21-10-2022', 'dd-mm-yyyy'), 60)
INTO versiune VALUES(40742, 1, 31.5, TO_DATE('23-02-2021', 'dd-mm-yyyy'), 40)
INTO versiune VALUES(40742, 2, 31.5, TO_DATE('23-02-2021', 'dd-mm-yyyy'), 40)
INTO versiune VALUES(40742, 3, 31.5, TO_DATE('23-02-2021', 'dd-mm-yyyy'), 40)

```

INTO versiune VALUES(40742, 7, 18.2, TO_DATE('23-02-2021', 'dd-mm-yyyy'), 40)
INTO versiune VALUES(460000, 1, 54.5, TO_DATE('01-12-2015', 'dd-mm-yyyy'), 20)
INTO versiune VALUES(460000, 2, 40.2, TO_DATE('01-12-2015', 'dd-mm-yyyy'), 20)
INTO versiune VALUES(460000, 4, 40.2, TO_DATE('01-12-2015', 'dd-mm-yyyy'), 20)
INTO versiune VALUES(460000, 3, 50.7, TO_DATE('01-12-2020', 'dd-mm-yyyy'), 30)
INTO versiune VALUES(460000, 5, 50.7, TO_DATE('01-12-2020', 'dd-mm-yyyy'), 30)
INTO versiune VALUES(460000, 6, 50.7, TO_DATE('01-12-2020', 'dd-mm-yyyy'), 30)
INTO versiune VALUES(460000, 11, 0, TO_DATE('30-6-2021', 'dd-mm-yyyy'), 8)
INTO versiune VALUES(101000, 7, 16, TO_DATE('12-11-2021', 'dd-mm-yyyy'), 60)
INTO versiune VALUES(102000, 1, 30.7, TO_DATE('21-08-2012', 'dd-mm-yyyy'), 10)
INTO versiune VALUES(102000, 8, 30.7, TO_DATE('21-08-2012', 'dd-mm-yyyy'), 10)
INTO versiune VALUES(102000, 12, 30.7, TO_DATE('23-09-2014', 'dd-mm-yyyy'), 10)
INTO versiune VALUES(103000, 9, 2.99, TO_DATE('12-12-2018', 'dd-mm-yyyy'), 0)
INTO versiune VALUES(103000, 10, 2.99, TO_DATE('12-12-2018', 'dd-mm-yyyy'), 0)
INTO versiune VALUES(104000, 1, 2, TO_DATE('29-07-2015', 'dd-mm-yyyy'), 20)
INTO versiune VALUES(104000, 2, 2, TO_DATE('29-07-2015', 'dd-mm-yyyy'), 20)
INTO versiune VALUES(104000, 4, 2, TO_DATE('29-07-2015', 'dd-mm-yyyy'), 20)
INTO versiune VALUES(104000, 7, 2, TO_DATE('29-07-2015', 'dd-mm-yyyy'), 20)
INTO versiune VALUES(104000, 9, 0.33, TO_DATE('07-10-2011', 'dd-mm-yyyy'), 8)
INTO versiune VALUES(104000, 10, 0.33, TO_DATE('17-11-2011', 'dd-mm-yyyy'), 8)
INTO versiune VALUES(105000, 1, 76.3, TO_DATE('07-03-2017', 'dd-mm-yyyy'), 50)
INTO versiune VALUES(105000, 2, 76.3, TO_DATE('07-03-2017', 'dd-mm-yyyy'), 50)
INTO versiune VALUES(105000, 4, 76.3, TO_DATE('07-03-2017', 'dd-mm-yyyy'), 50)
INTO versiune VALUES(105000, 11, 0, TO_DATE('07-03-2017', 'dd-mm-yyyy'), 50)
INTO versiune VALUES(106000, 1, 70, TO_DATE('10-12-2020', 'dd-mm-yyyy'), 60)
INTO versiune VALUES(106000, 3, 70, TO_DATE('10-12-2020', 'dd-mm-yyyy'), 60)
INTO versiune VALUES(106000, 5, 70, TO_DATE('10-12-2020', 'dd-mm-yyyy'), 60)
INTO versiune VALUES(106000, 6, 70, TO_DATE('10-12-2020', 'dd-mm-yyyy'), 60)
INTO versiune VALUES(107000, 1, 43.7, TO_DATE('19-05-2015', 'dd-mm-yyyy'), 30)
INTO versiune VALUES(107000, 2, 43.7, TO_DATE('19-05-2015', 'dd-mm-yyyy'), 30)
INTO versiune VALUES(107000, 4, 43.7, TO_DATE('19-05-2015', 'dd-mm-yyyy'), 30)
INTO versiune VALUES(107000, 3, 43.7, TO_DATE('14-12-2022', 'dd-mm-yyyy'), 30)

```

INTO versiune VALUES(107000, 5, 43.7, TO_DATE('14-12-2022','dd-mm-yyyy'), 30)
INTO versiune VALUES(107000, 6, 43.7, TO_DATE('14-12-2022','dd-mm-yyyy'), 30)
INTO versiune VALUES(107000, 7, 19.2, TO_DATE('15-10-2019','dd-mm-yyyy'), 40)
INTO versiune VALUES(108000, 1, 0.18, TO_DATE('22-12-2014','dd-mm-yyyy'), 4)
INTO versiune VALUES(108000, 9, 0.18, TO_DATE('13-08-2013','dd-mm-yyyy'), 4)
INTO versiune VALUES(108000, 10, 0.18, TO_DATE('13-08-2013','dd-mm-yyyy'), 4)
INTO versiune VALUES(108000, 12, 0.18, TO_DATE('22-12-2014','dd-mm-yyyy'), 4)
INTO versiune VALUES(25200, 1, 20.5, TO_DATE('26-08-2022','dd-mm-yyyy'), 60)
INTO versiune VALUES(25200, 2, 20.5, TO_DATE('26-08-2022','dd-mm-yyyy'), 60)
INTO versiune VALUES(25200, 4, 20.5, TO_DATE('26-08-2022','dd-mm-yyyy'), 60)
INTO versiune VALUES(109000, 1, 1, TO_DATE('01-09-2006','dd-mm-yyyy'), 0)
INTO versiune VALUES(109000, 9, 1, TO_DATE('16-07-2014','dd-mm-yyyy'), 0)
INTO versiune VALUES(109000, 10, 1, TO_DATE('11-12-2012','dd-mm-yyyy'), 0)
INTO versiune VALUES(109000, 12, 1, TO_DATE('01-09-2006','dd-mm-yyyy'), 0)
SELECT * FROM DUAL;

```

INSERT ALL

```

INTO stoc VALUES(300, 100290, 1, 20)
INTO stoc VALUES(300, 100290, 2, 15)
INTO stoc VALUES(300, 40741, 4, 4)
INTO stoc VALUES(300, 40741, 1, 30)
INTO stoc VALUES(300, 40741, 7, 10)
INTO stoc VALUES(300, 101000, 7, 3)
INTO stoc VALUES(300, 102000, 1, 300)
INTO stoc VALUES(300, 106000, 1, 98)
INTO stoc VALUES(300, 25200, 1, 134)
INTO stoc VALUES(301, 101000, 7, 201)
INTO stoc VALUES(301, 102000, 1, 153)
INTO stoc VALUES(301, 103000, 10, 40)
INTO stoc VALUES(301, 104000, 1, 329)
INTO stoc VALUES(301, 105000, 1, 104)
INTO stoc VALUES(301, 106000, 1, 30)

```

INTO stoc VALUES(301, 107000, 1, 309)
INTO stoc VALUES(301, 108000, 1, 980)
INTO stoc VALUES(301, 109000, 1, 189)
INTO stoc VALUES(302, 100290, 1, 4)
INTO stoc VALUES(302, 100290, 4, 5)
INTO stoc VALUES(302, 100290, 7, 1)
INTO stoc VALUES(302, 101000, 7, 1)
INTO stoc VALUES(302, 104000, 1, 35)
INTO stoc VALUES(302, 105000, 1, 16)
INTO stoc VALUES(304, 460000, 1, 20)
INTO stoc VALUES(304, 460000, 2, 20)
INTO stoc VALUES(304, 460000, 4, 20)
INTO stoc VALUES(304, 105000, 1, 30)
INTO stoc VALUES(304, 105000, 2, 30)
INTO stoc VALUES(304, 105000, 4, 30)
INTO stoc VALUES(305, 40741, 7, 178)
INTO stoc VALUES(305, 40742, 7, 341)
INTO stoc VALUES(305, 101000, 7, 250)
INTO stoc VALUES(305, 25200, 2, 999)
INTO stoc VALUES(306, 100290, 1, 20)
INTO stoc VALUES(306, 100290, 2, 15)
INTO stoc VALUES(306, 40741, 4, 4)
INTO stoc VALUES(306, 40741, 1, 30)
INTO stoc VALUES(306, 40741, 7, 10)
INTO stoc VALUES(306, 101000, 7, 3)
INTO stoc VALUES(306, 102000, 1, 300)
INTO stoc VALUES(306, 106000, 1, 98)
INTO stoc VALUES(306, 25200, 1, 134)
INTO stoc VALUES(306, 102000, 8, 153)
INTO stoc VALUES(306, 103000, 10, 40)
INTO stoc VALUES(306, 104000, 1, 329)
INTO stoc VALUES(306, 105000, 1, 104)

INTO stoc VALUES(306, 106000, 3, 30)
INTO stoc VALUES(306, 107000, 1, 309)
INTO stoc VALUES(306, 108000, 1, 980)
INTO stoc VALUES(306, 109000, 1, 189)
INTO stoc VALUES(307, 101000, 7, 539)
INTO stoc VALUES(308, 100290, 1, 12)
INTO stoc VALUES(308, 101000, 7, 13)
INTO stoc VALUES(308, 105000, 1, 15)
INTO stoc VALUES(308, 108000, 10, 10)
INTO stoc VALUES(310, 100290, 1, 20)
INTO stoc VALUES(310, 100290, 2, 15)
INTO stoc VALUES(310, 40741, 4, 4)
INTO stoc VALUES(310, 40741, 1, 30)
INTO stoc VALUES(310, 40741, 7, 10)
INTO stoc VALUES(310, 101000, 7, 3)
INTO stoc VALUES(310, 102000, 1, 300)
INTO stoc VALUES(310, 106000, 1, 98)
INTO stoc VALUES(310, 25200, 1, 134)
INTO stoc VALUES(310, 102000, 8, 153)
INTO stoc VALUES(310, 103000, 10, 40)
INTO stoc VALUES(310, 104000, 1, 329)
INTO stoc VALUES(310, 105000, 1, 104)
INTO stoc VALUES(310, 106000, 3, 30)
INTO stoc VALUES(310, 107000, 1, 309)
INTO stoc VALUES(310, 108000, 1, 980)
INTO stoc VALUES(310, 109000, 1, 189)
INTO stoc VALUES(310, 100290, 4, 12)
INTO stoc VALUES(310, 108000, 10, 10)
INTO stoc VALUES(312, 101000, 7, 5)
INTO stoc VALUES(312, 104000, 9, 5)
INTO stoc VALUES(312, 103000, 9, 5)
INTO stoc VALUES(312, 108000, 9, 5)

```
INTO stoc VALUES(313, 109000, 1, 10)
INTO stoc VALUES(313, 109000, 9, 10)
INTO stoc VALUES(313, 109000, 10, 10)
INTO stoc VALUES(313, 109000, 12, 10)
SELECT * FROM DUAL;
```

INSERT ALL

```
INTO dezvoltator VALUES(100, 'PlatinumGames')
INTO dezvoltator VALUES(110, 'Atlus')
INTO dezvoltator VALUES(210, 'P Studio')
INTO dezvoltator VALUES(120, 'Omega Force')
INTO dezvoltator VALUES(130, 'Ubisoft')
INTO dezvoltator VALUES(140, 'Valve')
INTO dezvoltator VALUES(150, 'Hidden Path')
INTO dezvoltator VALUES(160, 'Supercell')
INTO dezvoltator VALUES(170, 'CD Projekt Red')
INTO dezvoltator VALUES(180, 'RobTop')
INTO dezvoltator VALUES(190, 'Roblox Corp.')
INTO dezvoltator VALUES(200, 'Mojang')
SELECT * FROM DUAL;
```

INSERT ALL

```
INTO editor VALUES(100, 'Square Enix')
INTO editor VALUES(210, 'Atlus')
INTO editor VALUES(220, 'Sega')
INTO editor VALUES(300, 'Ubisoft')
INTO editor VALUES(400, 'Valve')
INTO editor VALUES(500, 'Supercell')
INTO editor VALUES(600, 'CD Projekt')
INTO editor VALUES(700, 'RobTop')
INTO editor VALUES(800, 'Roblox Corp.')
INTO editor VALUES(900, 'Mojang')
```

```
SELECT * FROM DUAL;
```

```
INSERT ALL
```

```
INTO contribuie VALUES(100290, 100, 100)
```

```
INTO contribuie VALUES(40741, 110, 210)
```

```
INTO contribuie VALUES(40741, 210, 210)
```

```
INTO contribuie VALUES(40741, 110, 220)
```

```
INTO contribuie VALUES(40741, 210, 220)
```

```
INTO contribuie VALUES(40742, 210, 210)
```

```
INTO contribuie VALUES(40742, 120, 210)
```

```
INTO contribuie VALUES(460000, 130, 300)
```

```
INTO contribuie VALUES(101000, 110, 210)
```

```
INTO contribuie VALUES(101000, 110, 220)
```

```
INTO contribuie VALUES(102000, 140, 400)
```

```
INTO contribuie VALUES(102000, 150, 400)
```

```
INTO contribuie VALUES(103000, 160, 500)
```

```
INTO contribuie VALUES(104000, 200, 900)
```

```
INTO contribuie VALUES(105000, 130, 300)
```

```
INTO contribuie VALUES(106000, 170, 600)
```

```
INTO contribuie VALUES(107000, 170, 600)
```

```
INTO contribuie VALUES(108000, 180, 700)
```

```
INTO contribuie VALUES(25200, 110, 210)
```

```
INTO contribuie VALUES(25200, 110, 220)
```

```
INTO contribuie VALUES(109000, 190, 800)
```

```
SELECT * FROM DUAL;
```

```
INSERT ALL
```

```
INTO limba VALUES('EN', 'Engleza')
```

```
INTO limba VALUES('JP', 'Japoneza')
```

```
INTO limba VALUES('RO', 'Romana')
```

```
INTO limba VALUES('IT', 'Italiana')
```

```
INTO limba VALUES('SP', 'Spaniola')
```



```
SELECT * FROM DUAL;
```

```
INSERT ALL
```

```
INTO joc_in_limba VALUES(100290, 'EN', 1)
INTO joc_in_limba VALUES(100290, 'JP', 1)
INTO joc_in_limba VALUES(40741, 'EN', 1)
INTO joc_in_limba VALUES(40741, 'JP', 1)
INTO joc_in_limba VALUES(40741, 'IT', 0)
INTO joc_in_limba VALUES(40741, 'SP', 0)
INTO joc_in_limba VALUES(40742, 'EN', 1)
INTO joc_in_limba VALUES(40742, 'JP', 1)
INTO joc_in_limba VALUES(101000, 'EN', 1)
INTO joc_in_limba VALUES(101000, 'JP', 1)
INTO joc_in_limba VALUES(25200, 'EN', 1)
INTO joc_in_limba VALUES(25200, 'JP', 1)
INTO joc_in_limba VALUES(460000, 'EN', 1)
INTO joc_in_limba VALUES(460000, 'JP', 0)
INTO joc_in_limba VALUES(460000, 'IT', 0)
INTO joc_in_limba VALUES(460000, 'SP', 0)
INTO joc_in_limba VALUES(102000, 'EN', 1)
INTO joc_in_limba VALUES(103000, 'EN', 1)
INTO joc_in_limba VALUES(104000, 'EN', 0)
INTO joc_in_limba VALUES(104000, 'JP', 0)
INTO joc_in_limba VALUES(104000, 'RO', 0)
INTO joc_in_limba VALUES(104000, 'SP', 0)
INTO joc_in_limba VALUES(104000, 'IT', 0)
INTO joc_in_limba VALUES(109000, 'EN', 0)
INTO joc_in_limba VALUES(109000, 'JP', 0)
INTO joc_in_limba VALUES(109000, 'RO', 0)
INTO joc_in_limba VALUES(109000, 'SP', 0)
INTO joc_in_limba VALUES(109000, 'IT', 0)
INTO joc_in_limba VALUES(105000, 'EN', 1)
```

```
INTO joc_in_limba VALUES(105000, 'IT', 1)
INTO joc_in_limba VALUES(106000, 'EN', 1)
INTO joc_in_limba VALUES(106000, 'SP', 1)
INTO joc_in_limba VALUES(107000, 'EN', 1)
INTO joc_in_limba VALUES(108000, 'EN', 0)
SELECT * FROM DUAL;
```

```
INSERT ALL
```

```
INTO angajat
```

```
VALUES(1000, 300, 'Florice', 'Andreea Renata', 'renata@gmail.com', '+40756701314', 501,
TO_DATE('11-09-2006','dd-mm-yyyy'))
```

```
INTO angajat
```

```
VALUES(1001, 300, 'Ionescu', 'Ion', 'the_ion@gmail.com', '+40756742812', 1700, TO_DATE('11-
09-2019','dd-mm-yyyy'))
```

```
INTO angajat
```

```
VALUES(1002, 301, 'Pinduchi', 'Andreea Corina', 'corina@gmail.com', '+40756319217', 2050,
TO_DATE('07-05-2022','dd-mm-yyyy'))
```

```
INTO angajat
```

```
VALUES(1003, 301, 'Vasiliu', 'Ana', 'ana@ana.com', '+40726791325', 2000, TO_DATE('11-05-
2022','dd-mm-yyyy'))
```

```
INTO angajat
```

```
VALUES(1004, 301, 'Tenea', 'Andrada', 'andrada@gmail.com', '+40759312203', 1970,
TO_DATE('13-05-2023','dd-mm-yyyy'))
```

```
INTO angajat
```

```
VALUES(1005, 302, 'Burlacu', 'Eduard Daniel', NULL, '+40756771315', 2300, TO_DATE('10-10-
2022','dd-mm-yyyy'))
```

```
INTO angajat
```

```
VALUES(1006, 303, 'Smadu', 'Andrei', 'smadu@andrei.ro', NULL, 2500, TO_DATE('12-02-
2023','dd-mm-yyyy'))
```

```
INTO angajat
```

```
VALUES(1007, 304, 'Coman', 'Andrei David', 'coman@coman.com', '+40770649949', 2400,
TO_DATE('11-11-2022','dd-mm-yyyy'))
```

```
INTO angajat(id_angajat, id_magazin, nume, prenume, email, telefon, salariu)
```

```
VALUES(1008, 305, 'Sefer', 'Emanuel Adrian', 'emanu@gamil.com', '+31756771315', 3000)
```

```
INTO angajat
```

```
VALUES(1009, 306, 'Adamita', 'Stefan David', NULL, '+40756781315', 1000, TO_DATE('23-09-2020','dd-mm-yyyy'))
```

```
INTO angajat
```

```
VALUES(1010, 306, 'Melinte', 'Florin', NULL, NULL, 3000, TO_DATE('19-07-2019','dd-mm-yyyy'))
```

```
INTO angajat
```

```
VALUES(1011, 306, 'Berechet', 'Tudor', 'budor@gmail.com', '+40756781315', 2000, sysdate)
```

```
INTO angajat
```

```
VALUES(1012, 307, 'Williams', 'Will', 'will@will.will', NULL, 1000, TO_DATE('15-04-1998','dd-mm-yyyy'))
```

```
INTO angajat
```

```
VALUES(1013, 308, 'Giovanna', 'Giorno', NULL, NULL, 10000, TO_DATE('16-04-2001','dd-mm-yyyy'))
```

```
INTO angajat(id_angajat, id_magazin, nume, salariu)
```

```
VALUES(1014, 309, 'Theodore', 700)
```

```
INTO angajat
```

```
VALUES(1015, 310, 'White', 'Sandra', 'london1@gameshop.com', '+44000111222', 632, TO_DATE('13-05-2023','dd-mm-yyyy'))
```

```
INTO angajat
```

```
VALUES(1016, 310, 'Black', 'Jonathan Joe', 'london2@gameshop.com', '+44333111222', 701, TO_DATE('12-05-2023','dd-mm-yyyy'))
```

```
INTO angajat
```

```
VALUES(1017, 311, 'Popescu', 'Popa', 'popa@gameshop.com', '+40700111222', 502, TO_DATE('13-05-2023','dd-mm-yyyy'))
```

```
INTO angajat
```

```
VALUES(1018, 312, 'Lovin', 'Raluca', 'raluca@gameshop.com', '+40710111222', 3000, TO_DATE('09-05-2023','dd-mm-yyyy'))
```

```
INTO angajat
```

```
VALUES(1019, 313, 'Petru', 'Mihai', 'mihai@gameshop.com', '+40700611222', 1200, TO_DATE('08-05-2023','dd-mm-yyyy'))
```

```
INTO angajat
```

```
VALUES(1020, 314, 'Popescu', 'Ana', NULL, '+40700111262', 1029, TO_DATE('13-05-2023','dd-mm-yyyy'))
```

```
SELECT * FROM DUAL;
```

```
UPDATE magazin SET id_manager = 1001 WHERE id_magazin = 300;
```

```
UPDATE magazin SET id_manager = 1002 WHERE id_magazin = 301;
UPDATE magazin SET id_manager = 1005 WHERE id_magazin = 302;
UPDATE magazin SET id_manager = 1006 WHERE id_magazin = 303;
UPDATE magazin SET id_manager = 1007 WHERE id_magazin = 304;
UPDATE magazin SET id_manager = 1008 WHERE id_magazin = 305;
UPDATE magazin SET id_manager = 1010 WHERE id_magazin = 306;
UPDATE magazin SET id_manager = 1013 WHERE id_magazin = 308;
UPDATE magazin SET id_manager = 1014 WHERE id_magazin = 309;
UPDATE magazin SET id_manager = 1015 WHERE id_magazin = 310;
UPDATE magazin SET id_manager = 1018 WHERE id_magazin = 312;
UPDATE magazin SET id_manager = 1019 WHERE id_magazin = 313;
```

```
INSERT INTO client
```

```
VALUES(secventa_client.nextval, 'Cazan', 'George', 'george@gmail.com', '+40763192452');
```

```
INSERT INTO client
```

```
VALUES(secventa_client.nextval, 'Onodi', 'Paul', 'paul@gmail.com', '+40763693212');
```

```
INSERT INTO client
```

```
VALUES(secventa_client.nextval, 'Jansen', 'Liam', NULL, '+31756821387');
```

```
INSERT INTO client
```

```
VALUES(secventa_client.nextval, 'Greyson', 'Henry', 'henry@yahoo.com', NULL);
```

```
INSERT INTO client
```

```
VALUES(secventa_client.nextval, 'Cocu', 'Elena', 'elena@cocu.com', '+40769695674');
```

```
INSERT INTO client
```

```
VALUES(secventa_client.nextval, 'De Lucca', 'Antonio', NULL, NULL);
```

```
INSERT INTO client
```

```
VALUES(secventa_client.nextval, 'Nagy', 'Zoltan', 'nagy@zoltan.com', '+40711112222');
```

```
INSERT INTO comanda
```

```
VALUES(secventa_comanda.nextval, 1000000, 1001, TO_DATE('11-05-2023','dd-mm-yyyy'),  
'cash');
```

```
INSERT INTO comanda
```

```
VALUES(secventa_comanda.nextval, 1000000, 1002, TO_DATE('09-05-2023','dd-mm-yyyy'),  
'card');
```

```

INSERT INTO comanda
VALUES(secventa_comanda.nextval, 1000001, 1005, TO_DATE('05-05-2023','dd-mm-yyyy'),
'rate');

INSERT INTO comanda
VALUES(secventa_comanda.nextval, 1000003, 1012, TO_DATE('02-05-2023','dd-mm-yyyy'),
'card');

INSERT INTO comanda
VALUES(secventa_comanda.nextval, 1000003, 1015, TO_DATE('03-05-2023','dd-mm-yyyy'),
'card');

INSERT INTO comanda
VALUES(secventa_comanda.nextval, 1000002, 1008, TO_DATE('13-05-2023','dd-mm-yyyy'),
'mixt');

INSERT INTO comanda
VALUES(secventa_comanda.nextval, 1000000, 1006, TO_DATE('13-05-2023','dd-mm-yyyy'),
'cash');

INSERT INTO comanda(id_comanda, id_client, id_angajat)
VALUES(secventa_comanda.nextval, 1000004, 1007);

INSERT INTO comanda
VALUES(secventa_comanda.nextval, 1000005, 1013, TO_DATE('15-05-2023','dd-mm-yyyy'),
'cash');

INSERT INTO comanda
VALUES(secventa_comanda.nextval, 1000006, 1020, TO_DATE('11-05-2023','dd-mm-yyyy'),
'card');

INSERT INTO comanda
VALUES(secventa_comanda.nextval, 1000001, 1018, TO_DATE('07-05-2023','dd-mm-yyyy'),
'cash');

INSERT INTO comanda
VALUES(secventa_comanda.nextval, 1000005, 1019, TO_DATE('04-05-2023','dd-mm-yyyy'),
'cash');

INSERT ALL
INTO comanda_online VALUES(1037182341, 'Str. Preciziei 24, Bucure?ti 062204',
TO_DATE('10-05-2023','dd-mm-yyyy'))

INTO comanda_online VALUES(1111547023, 'Wandsworth Bridge Rd., London SW6 2NY',
TO_DATE('05-05-2023','dd-mm-yyyy'))

INTO comanda_online VALUES(1185911705, 'Vijzelstraat 15, 1017 HD Amsterdam',
TO_DATE('14-05-2023','dd-mm-yyyy'))

```

```
INTO comanda_online VALUES(1409005751, 'Parcheggio pubblico, Via Borgo Palazzo, 217, 24125 Bergamo BG', TO_DATE('20-05-2023','dd-mm-yyyy'))
```

```
INTO comanda_online VALUES(1371823410, 'Strada General Ioan Dragalina 10, Gala?i 800402', TO_DATE('07-05-2023','dd-mm-yyyy'))
```

```
SELECT * FROM DUAL;
```

```
INSERT ALL
```

```
INTO comanda_contine VALUES(101000, 7, 1000000000, 2)
```

```
INTO comanda_contine VALUES(101000, 7, 1037182341, 1)
```

```
INTO comanda_contine VALUES(40741, 7, 1037182341, 1)
```

```
INTO comanda_contine VALUES(100290, 7, 1037182341, 1)
```

```
INTO comanda_contine VALUES(460000, 1, 1074364682, 1)
```

```
INTO comanda_contine VALUES(105000, 1, 1074364682, 1)
```

```
INTO comanda_contine VALUES(25200, 2, 1111547023, 5)
```

```
INTO comanda_contine VALUES(104000, 9, 1148729364, 12)
```

```
INTO comanda_contine VALUES(104000, 10, 1148729364, 3)
```

```
INTO comanda_contine VALUES(106000, 1, 1185911705, 1)
```

```
INTO comanda_contine VALUES(101000, 7, 1223094046, 1)
```

```
INTO comanda_contine VALUES(101000, 7, 1260276387, 1)
```

```
INTO comanda_contine VALUES(102000, 1, 1260276387, 1)
```

```
INTO comanda_contine VALUES(103000, 9, 1260276387, 1)
```

```
INTO comanda_contine VALUES(40741, 1, 1297458728, 1)
```

```
INTO comanda_contine VALUES(40742, 1, 1297458728, 1)
```

```
INTO comanda_contine VALUES(100290, 1, 1334641069, 1)
```

```
INTO comanda_contine VALUES(100290, 1, 1371823410, 2)
```

```
INTO comanda_contine VALUES(102000, 1, 1371823410, 2)
```

```
INTO comanda_contine VALUES(106000, 1, 1371823410, 2)
```

```
INTO comanda_contine VALUES(40741, 1, 1371823410, 2)
```

```
INTO comanda_contine VALUES(107000, 1, 1371823410, 2)
```

```
INTO comanda_contine VALUES(104000, 1, 1371823410, 2)
```

```
INTO comanda_contine VALUES(101000, 7, 1409005751, 20)
```

```
SELECT * FROM DUAL;
```

```
COMMIT;
```

<div><div></div><div>INSERT ALL</div></div> <div>INTO tara VALUES ('RO', 'Romania')</div> <div>INTO tara VALUES ('NL', 'Olanda')</div> <div>INTO tara VALUES ('GB', 'Marea Britanie')</div> <div>INTO tara VALUES ('IT', 'Italia')</div> <div>INTO tara VALUES ('HU', 'Ungaria')</div> <div>SELECT * FROM DUAL;</div>	
<div><div></div><div>INSERT ALL</div></div> <div>INTO oras VALUES(11, 'Bucuresti', 'RO')</div> <div>INTO oras VALUES(12, 'Galati', 'RO')</div> <div>INTO oras VALUES(13, 'Cluj', 'RO')</div> <div>INTO oras VALUES(14, 'Amsterdam', 'NL')</div> <div>INTO oras VALUES(15, 'Brasov', 'RO')</div> <div>INTO oras VALUES(16, 'Londra', 'GB')</div> <div>INTO oras VALUES(17, 'Roma', 'IT')</div> <div>INTO oras VALUES(18, 'Rotterdam', 'NL')</div> <div>INTO oras VALUES(19, 'Budapesta', 'HU')</div> <div>INTO oras VALUES(20, 'Milano', 'IT')</div> <div>SELECT * FROM DUAL;</div>	
<div><div></div><div>INSERT ALL</div></div> <div>INTO magazin</div> <div>VALUES(300, 'Gameshop API', 'Bulevardul General Paul Teodorescu 4', 11, NULL)</div> <div>INTO magazin</div> <div>VALUES(301, 'Gameshop Centru Vechi', 'Strada Lipscani 55, Bucure7ti', 11, NULL)</div> <div>INTO magazin</div> <div>VALUES(302, 'Gamecorner Tiglina', 'Strada Br7ilei nr. 163, 800309', 12, NULL)</div> <div>INTO magazin</div> <div>VALUES(303, 'Gamecorner Corjului', 'Bulevardul Iuliu Maniu 67', 11, NULL)</div> <div>INTO magazin</div> <div>VALUES(304, 'Gameshop Cluj', 'Strada Alexandru Vaida Voevod 59, 400436', 13, NULL)</div> <div>INTO magazin</div> <div>VALUES(305, 'Gameshop Amsterdam', NULL, 14, NULL)</div> <div>INTO magazin</div> <div>VALUES(306, 'Megagame Brasov', 'Bulevardul 15 Noiembrie 78', 15, NULL)</div> <div>INTO magazin</div> <div>VALUES(307, 'Gamecorner Walworth', 'East St, London SE17 1EL', 16, NULL)</div> <div>INTO magazin</div> <div>VALUES(308, 'Gameshop Roma', NULL, 17, NULL)</div> <div>INTO magazin</div> <div>VALUES(309, 'Gameshop Rotterdam', 'Willem Ruyslaan 225, 3063 ER', 18, NULL)</div> <div>INTO magazin</div> <div>VALUES(310, 'Megashop Londra', '12 Walbrook, London EC4N 8AA', 16, NULL)</div> <div>INTO magazin</div> <div>VALUES(311, 'Gamecorner Budapesta 1', 'Magl6di 6t 14/B, 1106', 19, NULL)</div> <div>INTO magazin</div> <div>VALUES(312, 'Gamecorner Potcoava', 'Strada Brailoi, Nr.17, Complex Comercial Potcoava de Aur', 12, NULL)</div> <div>INTO magazin</div> <div>VALUES(313, 'Gameshop Milano', 'Via Lorenteggio, 219, 20147 Milano MI', 20, NULL)</div> <div>INTO magazin</div> <div>VALUES(314, 'Gamecorner Budapesta 2', 'K6nyves K6lm6n krt. 12-14, 1097', 19, NULL)</div> <div>SELECT * FROM DUAL;</div>	
<div><div></div><div>INSERT ALL</div></div> <div>INTO joc</div> <div>VALUES(100290, 'Nier: Automata', 'Nier: Automata spune povestea androizilor 2B, 9S si A2 si lupta lor pentru recapatarea distopiei controlate de masinarii puternice.', 'Action JRPG', 0, 18)</div> <div>INTO joc</div> <div>VALUES(40741, 'Persona 5 Royal', 'Scoate-ti masca si alatura-te Hutilor Fantoma si furturilor lor magnifice, cum infiltreaza mintilor oamenilor corupti, facandu-i sa-si schimbe cale.', 'Turn-based JRPG', 0, 16)</div> <div>INTO joc</div> <div>VALUES(40742, 'Persona 5 Strikers', NULL, 'Hack and Slash', 0, 16)</div> <div>INTO joc</div> <div>VALUES(460000, 'Rainbow Six: Siege', 'Rainbow Six: Siege este un shooter tactic de elita pe echipe, unde planificarile superioare si executiile trimfa.', 'Team Shooter', 1, 18)</div> <div>INTO joc</div> <div>VALUES(101000, 'Shin Megami Tensei V', NULL, 'Turn-based JRPG', 0, 16)</div> <div>INTO joc</div> <div>VALUES(102000, 'Counter Strike 2', NULL, 'Team Shooter', 1, 18)</div> <div>INTO joc</div> <div>VALUES(103000, 'Brawl Stars', NULL, 'Topdown Shooter', 1, 7)</div> <div>INTO joc</div> <div>VALUES(104000, 'Minecraft Bedrock', NULL, 'Sandbox', 1, 7)</div> <div>INTO joc</div> <div>VALUES(105000, 'Ghost Recon: Wildlands', NULL, 'Open World', 1, 18)</div> <div>INTO joc</div> <div>VALUES(106000, 'Cyberpunk 2077', NULL, 'Action RPG', 0, 18)</div> <div>INTO joc</div> <div>VALUES(107000, 'The Witcher 3', NULL, 'Open World', 0, 18)</div> <div>INTO joc</div> <div>VALUES(108000, 'Geometry Dash', NULL, 'Platformer', 0, 3)</div> <div>INTO joc</div> <div>VALUES(25200, 'Soul Hackers 2', NULL, 'Turn-based JRPG', 0, 16)</div> <div>INTO joc</div>	

```

VALUES(25200, 'Soul Hackers 2', 'Soul', '1999-08-06', 0, 16)
INTO joe
VALUES(109000, 'Roblox', NULL, 'Sandbox', 1, 3)
SELECT * FROM DUAL;

INSERT ALL
INTO joe_online VALUES(460000, 3800000, 1, 1)
INTO joe_online VALUES(102000, 10000000, 0, 0)
INTO joe_online VALUES(103000, 1000000, 1, 1)
INTO joe_online VALUES(104000, 30000000, 1, 0)
INTO joe_online VALUES(105000, 20000, 1, 1)
INTO joe_online VALUES(109000, 17800000, 1, 1)
SELECT * FROM DUAL;

INSERT ALL
INTO platforma VALUES(1, 'Windows', 'PC')
INTO platforma VALUES(2, 'Playstation 4', 'Console')
INTO platforma VALUES(3, 'Playstation 5', 'Console')
INTO platforma VALUES(4, 'Xbox One', 'Console')
INTO platforma VALUES(5, 'Xbox Series X', 'Console')
INTO platforma VALUES(6, 'Xbox Series S', 'Console')
INTO platforma VALUES(7, 'Nintendo Switch', 'Handheld')
INTO platforma VALUES(8, 'Steam Deck', 'Handheld')
INTO platforma VALUES(9, 'Android', 'Telefon')
INTO platforma VALUES(10, 'IOS', 'Telefon')
INTO platforma VALUES(11, 'Stadia', 'Cloud')
INTO platforma VALUES(12, 'Linux', 'PC')
SELECT * FROM DUAL;

INSERT ALL
INTO versione VALUES(100290, 1, 40.8, TO_DATE('17-03-2017', 'dd-mm-yyyy'), 40)
INTO versione VALUES(100290, 2, 38, TO_DATE('23-02-2017', 'dd-mm-yyyy'), 40)
INTO versione VALUES(100290, 4, 38, TO_DATE('26-06-2018', 'dd-mm-yyyy'), 40)
INTO versione VALUES(100290, 7, 15, TO_DATE('06-10-2022', 'dd-mm-yyyy'), 60)
INTO versione VALUES(100290, 8, 40.8, TO_DATE('17-03-2017', 'dd-mm-yyyy'), 40)
INTO versione VALUES(40741, 1, 39.3, TO_DATE('21-10-2022', 'dd-mm-yyyy'), 60)
INTO versione VALUES(40741, 2, 33.2, TO_DATE('31-03-2020', 'dd-mm-yyyy'), 40)
INTO versione VALUES(40741, 3, 39.3, TO_DATE('21-10-2022', 'dd-mm-yyyy'), 60)
INTO versione VALUES(40741, 4, 39.3, TO_DATE('21-10-2022', 'dd-mm-yyyy'), 60)
INTO versione VALUES(40741, 5, 39.3, TO_DATE('21-10-2022', 'dd-mm-yyyy'), 60)
INTO versione VALUES(40741, 6, 39.3, TO_DATE('21-10-2022', 'dd-mm-yyyy'), 60)
INTO versione VALUES(40741, 7, 21.1, TO_DATE('21-10-2022', 'dd-mm-yyyy'), 60)
INTO versione VALUES(40741, 8, 39.3, TO_DATE('21-10-2022', 'dd-mm-yyyy'), 60)
INTO versione VALUES(40742, 1, 31.5, TO_DATE('23-02-2021', 'dd-mm-yyyy'), 40)
INTO versione VALUES(40742, 2, 31.5, TO_DATE('23-02-2021', 'dd-mm-yyyy'), 40)
INTO versione VALUES(40742, 3, 31.5, TO_DATE('23-02-2021', 'dd-mm-yyyy'), 40)
INTO versione VALUES(40742, 7, 18.2, TO_DATE('23-02-2021', 'dd-mm-yyyy'), 40)
INTO versione VALUES(460000, 1, 54.5, TO_DATE('01-12-2015', 'dd-mm-yyyy'), 20)
INTO versione VALUES(460000, 2, 40.2, TO_DATE('01-12-2015', 'dd-mm-yyyy'), 20)
INTO versione VALUES(460000, 4, 40.2, TO_DATE('01-12-2015', 'dd-mm-yyyy'), 20)
INTO versione VALUES(460000, 3, 50.7, TO_DATE('01-12-2020', 'dd-mm-yyyy'), 30)
INTO versione VALUES(460000, 5, 50.7, TO_DATE('01-12-2020', 'dd-mm-yyyy'), 30)
INTO versione VALUES(460000, 6, 50.7, TO_DATE('01-12-2020', 'dd-mm-yyyy'), 30)
INTO versione VALUES(460000, 11, 0, TO_DATE('30-6-2021', 'dd-mm-yyyy'), 8)
INTO versione VALUES(101000, 7, 16, TO_DATE('12-11-2021', 'dd-mm-yyyy'), 60)
INTO versione VALUES(102000, 1, 30.7, TO_DATE('21-08-2012', 'dd-mm-yyyy'), 10)
INTO versione VALUES(102000, 8, 30.7, TO_DATE('21-08-2012', 'dd-mm-yyyy'), 10)
INTO versione VALUES(102000, 12, 30.7, TO_DATE('23-09-2014', 'dd-mm-yyyy'), 10)
INTO versione VALUES(103000, 9, 2.99, TO_DATE('12-12-2018', 'dd-mm-yyyy'), 0)
INTO versione VALUES(103000, 10, 2.99, TO_DATE('12-12-2018', 'dd-mm-yyyy'), 0)
INTO versione VALUES(104000, 1, 2, TO_DATE('29-07-2015', 'dd-mm-yyyy'), 20)
INTO versione VALUES(104000, 2, 2, TO_DATE('29-07-2015', 'dd-mm-yyyy'), 20)
INTO versione VALUES(104000, 4, 2, TO_DATE('29-07-2015', 'dd-mm-yyyy'), 20)
INTO versione VALUES(104000, 7, 2, TO_DATE('29-07-2015', 'dd-mm-yyyy'), 20)
INTO versione VALUES(104000, 9, 0.33, TO_DATE('07-10-2011', 'dd-mm-yyyy'), 8)
INTO versione VALUES(104000, 10, 0.33, TO_DATE('17-11-2011', 'dd-mm-yyyy'), 8)
INTO versione VALUES(105000, 1, 76.3, TO_DATE('07-03-2017', 'dd-mm-yyyy'), 50)
INTO versione VALUES(105000, 2, 76.3, TO_DATE('07-03-2017', 'dd-mm-yyyy'), 50)
INTO versione VALUES(105000, 4, 76.3, TO_DATE('07-03-2017', 'dd-mm-yyyy'), 50)
INTO versione VALUES(105000, 11, 0, TO_DATE('07-03-2017', 'dd-mm-yyyy'), 50)
INTO versione VALUES(106000, 1, 70, TO_DATE('10-12-2020', 'dd-mm-yyyy'), 60)
INTO versione VALUES(106000, 3, 70, TO_DATE('10-12-2020', 'dd-mm-yyyy'), 60)
INTO versione VALUES(106000, 5, 70, TO_DATE('10-12-2020', 'dd-mm-yyyy'), 60)
INTO versione VALUES(106000, 6, 70, TO_DATE('10-12-2020', 'dd-mm-yyyy'), 60)
INTO versione VALUES(107000, 1, 43.7, TO_DATE('19-05-2015', 'dd-mm-yyyy'), 30)
INTO versione VALUES(107000, 2, 43.7, TO_DATE('19-05-2015', 'dd-mm-yyyy'), 30)
INTO versione VALUES(107000, 4, 43.7, TO_DATE('19-05-2015', 'dd-mm-yyyy'), 30)
INTO versione VALUES(107000, 3, 43.7, TO_DATE('14-12-2022', 'dd-mm-yyyy'), 30)
INTO versione VALUES(107000, 5, 43.7, TO_DATE('14-12-2022', 'dd-mm-yyyy'), 30)
INTO versione VALUES(107000, 6, 43.7, TO_DATE('14-12-2022', 'dd-mm-yyyy'), 30)
INTO versione VALUES(107000, 7, 19.2, TO_DATE('15-10-2019', 'dd-mm-yyyy'), 40)
INTO versione VALUES(108000, 1, 0.18, TO_DATE('22-12-2014', 'dd-mm-yyyy'), 4)
INTO versione VALUES(108000, 9, 0.18, TO_DATE('13-08-2013', 'dd-mm-yyyy'), 4)

```

```

INTO versione VALUES(108000, 9, 0.18, TO_DATE('13-08-2013', 'dd-mm-yyyy'), 4)
INTO versione VALUES(108000, 10, 0.18, TO_DATE('13-08-2013', 'dd-mm-yyyy'), 4)
INTO versione VALUES(108000, 12, 0.18, TO_DATE('22-12-2014', 'dd-mm-yyyy'), 4)
INTO versione VALUES(25200, 1, 20.5, TO_DATE('26-08-2022', 'dd-mm-yyyy'), 60)
INTO versione VALUES(25200, 2, 20.5, TO_DATE('26-08-2022', 'dd-mm-yyyy'), 60)
INTO versione VALUES(25200, 4, 20.5, TO_DATE('26-08-2022', 'dd-mm-yyyy'), 60)
INTO versione VALUES(109000, 1, 1, TO_DATE('01-09-2006', 'dd-mm-yyyy'), 0)
INTO versione VALUES(109000, 9, 1, TO_DATE('16-07-2014', 'dd-mm-yyyy'), 0)
INTO versione VALUES(109000, 10, 1, TO_DATE('11-12-2012', 'dd-mm-yyyy'), 0)
INTO versione VALUES(109000, 12, 1, TO_DATE('01-09-2006', 'dd-mm-yyyy'), 0)
SELECT * FROM DUAL;

INSERT ALL
INTO stoc VALUES(300, 100290, 1, 20)
INTO stoc VALUES(300, 100290, 2, 15)
INTO stoc VALUES(300, 40741, 4, 4)
INTO stoc VALUES(300, 40741, 1, 30)
INTO stoc VALUES(300, 40741, 7, 10)
INTO stoc VALUES(300, 101000, 7, 3)
INTO stoc VALUES(300, 102000, 1, 300)
INTO stoc VALUES(300, 106000, 1, 98)
INTO stoc VALUES(300, 25200, 1, 134)
INTO stoc VALUES(301, 101000, 7, 201)
INTO stoc VALUES(301, 102000, 1, 153)
INTO stoc VALUES(301, 103000, 10, 40)
INTO stoc VALUES(301, 104000, 1, 329)
INTO stoc VALUES(301, 105000, 1, 104)
INTO stoc VALUES(301, 106000, 1, 30)
INTO stoc VALUES(301, 107000, 1, 309)
INTO stoc VALUES(301, 108000, 1, 980)
INTO stoc VALUES(301, 109000, 1, 189)
INTO stoc VALUES(302, 100290, 1, 4)
INTO stoc VALUES(302, 100290, 4, 5)
INTO stoc VALUES(302, 100290, 7, 1)
INTO stoc VALUES(302, 101000, 7, 1)
INTO stoc VALUES(302, 104000, 1, 35)
INTO stoc VALUES(302, 105000, 1, 16)
INTO stoc VALUES(304, 460000, 1, 20)
INTO stoc VALUES(304, 460000, 2, 20)
INTO stoc VALUES(304, 460000, 4, 20)
INTO stoc VALUES(304, 105000, 1, 30)
INTO stoc VALUES(304, 105000, 2, 30)
INTO stoc VALUES(304, 105000, 4, 30)
INTO stoc VALUES(305, 40741, 7, 178)
INTO stoc VALUES(305, 40742, 7, 341)
INTO stoc VALUES(305, 101000, 7, 250)
INTO stoc VALUES(305, 25200, 2, 999)
INTO stoc VALUES(306, 100290, 1, 20)
INTO stoc VALUES(306, 100290, 2, 15)
INTO stoc VALUES(306, 40741, 4, 4)
INTO stoc VALUES(306, 40741, 1, 30)
INTO stoc VALUES(306, 40741, 7, 10)
INTO stoc VALUES(306, 101000, 7, 3)
INTO stoc VALUES(306, 102000, 1, 300)
INTO stoc VALUES(306, 106000, 1, 98)
INTO stoc VALUES(306, 25200, 1, 134)
INTO stoc VALUES(306, 102000, 8, 153)
INTO stoc VALUES(306, 103000, 10, 40)
INTO stoc VALUES(306, 104000, 1, 329)
INTO stoc VALUES(306, 105000, 1, 104)
INTO stoc VALUES(306, 106000, 3, 30)
INTO stoc VALUES(306, 107000, 1, 309)
INTO stoc VALUES(306, 108000, 1, 980)
INTO stoc VALUES(306, 109000, 1, 189)
INTO stoc VALUES(307, 101000, 7, 539)
INTO stoc VALUES(308, 100290, 1, 12)
INTO stoc VALUES(308, 101000, 7, 13)
INTO stoc VALUES(308, 105000, 1, 15)
INTO stoc VALUES(308, 108000, 10, 10)
INTO stoc VALUES(310, 100290, 1, 20)
INTO stoc VALUES(310, 100290, 2, 15)
INTO stoc VALUES(310, 40741, 4, 4)
INTO stoc VALUES(310, 40741, 1, 30)
INTO stoc VALUES(310, 40741, 7, 10)
INTO stoc VALUES(310, 101000, 7, 3)
INTO stoc VALUES(310, 102000, 1, 300)
INTO stoc VALUES(310, 106000, 1, 98)
INTO stoc VALUES(310, 25200, 1, 134)
INTO stoc VALUES(310, 102000, 8, 153)
INTO stoc VALUES(310, 103000, 10, 40)
INTO stoc VALUES(310, 104000, 1, 329)
INTO stoc VALUES(310, 105000, 1, 104)

```



```

INTO stoc VALUES(310, 106000, 3, 30)
INTO stoc VALUES(310, 107000, 1, 309)
INTO stoc VALUES(310, 108000, 1, 980)
INTO stoc VALUES(310, 109000, 1, 189)
INTO stoc VALUES(310, 100290, 4, 12)
INTO stoc VALUES(310, 108000, 10, 10)
INTO stoc VALUES(312, 101000, 7, 5)
INTO stoc VALUES(312, 104000, 9, 5)
INTO stoc VALUES(312, 103000, 9, 5)
INTO stoc VALUES(312, 108000, 9, 5)
INTO stoc VALUES(313, 109000, 1, 10)
INTO stoc VALUES(313, 109000, 9, 10)
INTO stoc VALUES(313, 109000, 10, 10)
INTO stoc VALUES(313, 109000, 12, 10)
SELECT * FROM DUAL;

) INSERT ALL
INTO dezvoltator VALUES(100, 'PlatinumGames')
INTO dezvoltator VALUES(110, 'Atlus')
INTO dezvoltator VALUES(210, 'P Studio')
INTO dezvoltator VALUES(120, 'Omega Force')
INTO dezvoltator VALUES(130, 'Ubisoft')
INTO dezvoltator VALUES(140, 'Valve')
INTO dezvoltator VALUES(150, 'Hidden Path')
INTO dezvoltator VALUES(160, 'Supercell')
INTO dezvoltator VALUES(170, 'CD Projekt Red')
INTO dezvoltator VALUES(180, 'RobTop')
INTO dezvoltator VALUES(190, 'Roblox Corp.')
INTO dezvoltator VALUES(200, 'Mojang')
SELECT * FROM DUAL;

) INSERT ALL
INTO editor VALUES(100, 'Square Enix')
INTO editor VALUES(210, 'Atlus')
INTO editor VALUES(220, 'Sega')
INTO editor VALUES(300, 'Ubisoft')
INTO editor VALUES(400, 'Valve')
INTO editor VALUES(500, 'Supercell')
INTO editor VALUES(600, 'CD Projekt')
INTO editor VALUES(700, 'RobTop')
INTO editor VALUES(800, 'Roblox Corp.')
INTO editor VALUES(900, 'Mojang')
SELECT * FROM DUAL;

) INSERT ALL
INTO contribuie VALUES(100290, 100, 100)
INTO contribuie VALUES(40741, 110, 210)
INTO contribuie VALUES(40741, 210, 210)
INTO contribuie VALUES(40741, 110, 220)
INTO contribuie VALUES(40741, 210, 220)
INTO contribuie VALUES(40742, 210, 210)
INTO contribuie VALUES(40742, 120, 210)
INTO contribuie VALUES(460000, 130, 300)
INTO contribuie VALUES(101000, 110, 210)
INTO contribuie VALUES(101000, 110, 220)
INTO contribuie VALUES(102000, 140, 400)
INTO contribuie VALUES(102000, 150, 400)
INTO contribuie VALUES(103000, 160, 500)
INTO contribuie VALUES(104000, 200, 900)
INTO contribuie VALUES(105000, 130, 300)
INTO contribuie VALUES(106000, 170, 600)
INTO contribuie VALUES(107000, 170, 600)
INTO contribuie VALUES(108000, 180, 700)
INTO contribuie VALUES(25200, 110, 210)
INTO contribuie VALUES(25200, 110, 220)
INTO contribuie VALUES(109000, 190, 800)
SELECT * FROM DUAL;

) INSERT ALL
INTO limba VALUES('EN', 'Engleza')
INTO limba VALUES('JP', 'Japoneza')
INTO limba VALUES('RO', 'Romana')
INTO limba VALUES('IT', 'Italiana')
INTO limba VALUES('SP', 'Spaniola')
SELECT * FROM DUAL;

) INSERT ALL
INTO joe_in_limba VALUES(100290, 'EN', 1)
INTO joe_in_limba VALUES(100290, 'JP', 1)
INTO joe_in_limba VALUES(40741, 'EN', 1)
INTO joe_in_limba VALUES(40741, 'JP', 1)
INTO joe_in_limba VALUES(40741, 'IT', 1)

```

```

INTO joe_in_limba VALUES(40741, 'SP', 0)
INTO joe_in_limba VALUES(40742, 'EN', 1)
INTO joe_in_limba VALUES(40742, 'JP', 1)
INTO joe_in_limba VALUES(101000, 'EN', 1)
INTO joe_in_limba VALUES(101000, 'JP', 1)
INTO joe_in_limba VALUES(25200, 'EN', 1)
INTO joe_in_limba VALUES(25200, 'JP', 1)
INTO joe_in_limba VALUES(460000, 'EN', 1)
INTO joe_in_limba VALUES(460000, 'JP', 0)
INTO joe_in_limba VALUES(460000, 'IT', 0)
INTO joe_in_limba VALUES(460000, 'SP', 0)
INTO joe_in_limba VALUES(102000, 'EN', 1)
INTO joe_in_limba VALUES(103000, 'EN', 1)
INTO joe_in_limba VALUES(104000, 'EN', 0)
INTO joe_in_limba VALUES(104000, 'JP', 0)
INTO joe_in_limba VALUES(104000, 'RO', 0)
INTO joe_in_limba VALUES(104000, 'SP', 0)
INTO joe_in_limba VALUES(104000, 'IT', 0)
INTO joe_in_limba VALUES(105000, 'EN', 1)
INTO joe_in_limba VALUES(105000, 'JP', 0)
INTO joe_in_limba VALUES(105000, 'RO', 0)
INTO joe_in_limba VALUES(105000, 'SP', 0)
INTO joe_in_limba VALUES(105000, 'IT', 0)
INTO joe_in_limba VALUES(105000, 'EN', 1)
INTO joe_in_limba VALUES(106000, 'EN', 1)
INTO joe_in_limba VALUES(106000, 'JP', 1)
INTO joe_in_limba VALUES(107000, 'EN', 1)
INTO joe_in_limba VALUES(108000, 'EN', 0)
SELECT * FROM DUAL;

) INSERT ALL
INTO angajat VALUES(1000, 300, 'Florinel', 'Andrea Renata', 'renata@gmail.com', '+40756701314', 501, TO_DATE('11-09-2006','dd-mm-yyyy'))
INTO angajat VALUES(1001, 300, 'Ionescu', 'Ion', 'the_ion@gmail.com', '+40756742812', 1700, TO_DATE('11-09-2019','dd-mm-yyyy'))
INTO angajat VALUES(1002, 301, 'Pinduch', 'Andrea Corina', 'corina@gmail.com', '+40756319217', 2050, TO_DATE('07-05-2022','dd-mm-yyyy'))
INTO angajat VALUES(1003, 301, 'Vasilu', 'Ana', 'ana@ana.com', '+40726791325', 2000, TO_DATE('11-05-2022','dd-mm-yyyy'))
INTO angajat VALUES(1004, 301, 'Teneu', 'Andrada', 'andrada@gmail.com', '+40759312203', 1970, TO_DATE('13-05-2023','dd-mm-yyyy'))
INTO angajat VALUES(1005, 302, 'Burlacu', 'Eduard Daniel', NULL, '+40756771315', 2300, TO_DATE('10-10-2022','dd-mm-yyyy'))
INTO angajat VALUES(1006, 303, 'Smadu', 'Andrei', 'smadu@andrei.ro', NULL, 2500, TO_DATE('12-02-2023','dd-mm-yyyy'))
INTO angajat VALUES(1007, 304, 'Coman', 'Andrei David', 'coman@coman.com', '+40770649949', 2400, TO_DATE('11-11-2022','dd-mm-yyyy'))
INTO angajat(id_angajat, id_magazin, nume, prenume, email, telefon, salariu)
VALUES(1008, 305, 'Sofici', 'Emanuel Adrian', 'emmanuel@gmail.com', '+31756771315', 3000)
INTO angajat VALUES(1009, 306, 'Adamita', 'Stefan David', NULL, '+40756781315', 1000, TO_DATE('23-09-2020','dd-mm-yyyy'))
INTO angajat VALUES(1010, 306, 'Melinte', 'Florin', NULL, NULL, 3000, TO_DATE('19-07-2019','dd-mm-yyyy'))
INTO angajat VALUES(1011, 306, 'Berechet', 'Tudor', 'tudor@gmail.com', '+40756781315', 2000, sysdate)
INTO angajat VALUES(1012, 307, 'Williams', 'Willi', 'williwill.will', NULL, 1000, TO_DATE('15-04-1998','dd-mm-yyyy'))
INTO angajat VALUES(1013, 308, 'Giovanna', 'Glorio', NULL, NULL, 10000, TO_DATE('16-04-2001','dd-mm-yyyy'))
INTO angajat(id_angajat, id_magazin, nume, salariu)
VALUES(1014, 309, 'Theodore', 700)
INTO angajat VALUES(1015, 310, 'White', 'Sandra', 'london1@gameshop.com', '+44000111222', 632, TO_DATE('13-05-2023','dd-mm-yyyy'))
INTO angajat VALUES(1016, 310, 'Black', 'Jonathan Joe', 'london2@gameshop.com', '+44333111222', 701, TO_DATE('12-05-2023','dd-mm-yyyy'))
INTO angajat VALUES(1017, 311, 'Popescu', 'Popa', 'popa@gameshop.com', '+40700111222', 502, TO_DATE('13-05-2023','dd-mm-yyyy'))
INTO angajat VALUES(1018, 312, 'Lovin', 'Raluca', 'raluca@gameshop.com', '+40710111222', 3000, TO_DATE('09-05-2023','dd-mm-yyyy'))
INTO angajat VALUES(1019, 313, 'Petru', 'Mihai', 'mihai@gameshop.com', '+40700611222', 1200, TO_DATE('08-05-2023','dd-mm-yyyy'))
INTO angajat VALUES(1020, 314, 'Popescu', 'Ana', NULL, '+40700111262', 1029, TO_DATE('13-05-2023','dd-mm-yyyy'))
SELECT * FROM DUAL;

UPDATE magazin SET id_manager = 1001 WHERE id_magazin = 300;
UPDATE magazin SET id_manager = 1002 WHERE id_magazin = 301;
UPDATE magazin SET id_manager = 1005 WHERE id_magazin = 302;
UPDATE magazin SET id_manager = 1006 WHERE id_magazin = 303;
UPDATE magazin SET id_manager = 1007 WHERE id_magazin = 304;
UPDATE magazin SET id_manager = 1008 WHERE id_magazin = 305;
UPDATE magazin SET id_manager = 1010 WHERE id_magazin = 306;
UPDATE magazin SET id_manager = 1013 WHERE id_magazin = 308;

) INSERT INTO client
VALUES(sectenta,client,numeal,'Cucu','George','george@gmail.com','+40761304527');
) INSERT INTO client
VALUES(sectenta,client,numeal,'Cucu','Paul','paul@gmail.com','+40763893127');
) INSERT INTO client
VALUES(sectenta,client,numeal,'Cucuta','Lina',NULL,'+317568213877');
) INSERT INTO client
VALUES(sectenta,client,numeal,'Cucuta','Henry','henry@pub.com',NULL);
) INSERT INTO client
VALUES(sectenta,client,numeal,'Cucu','Elena','elena@pub.com','+40769050747');
) INSERT INTO client
VALUES(sectenta,client,numeal,'Buc','Sofia','sofia@pub.com','+40711112222');
) INSERT INTO comanda
VALUES(sectenta,comanda,numeal,100000,1001,TO_DATE('11-05-2023','dd-mm-yyyy'),'cash');
) INSERT INTO comanda
VALUES(sectenta,comanda,numeal,100000,1002,TO_DATE('09-05-2023','dd-mm-yyyy'),'card');
) INSERT INTO comanda
VALUES(sectenta,comanda,numeal,100000,1005,TO_DATE('09-05-2023','dd-mm-yyyy'),'card');
) INSERT INTO comanda
VALUES(sectenta,comanda,numeal,100000,1012,TO_DATE('02-05-2023','dd-mm-yyyy'),'card');
) INSERT INTO comanda
VALUES(sectenta,comanda,numeal,100000,1015,TO_DATE('03-05-2023','dd-mm-yyyy'),'card');
) INSERT INTO comanda
VALUES(sectenta,comanda,numeal,100000,1008,TO_DATE('13-05-2023','dd-mm-yyyy'),'mixt');
) INSERT INTO comanda
VALUES(sectenta,comanda,numeal,100000,1006,TO_DATE('13-05-2023','dd-mm-yyyy'),'cash');
) INSERT INTO comanda(id_comanda, id_client, id_angajat)
VALUES(sectenta,comanda,numeal,100000,1007);
) INSERT INTO comanda
VALUES(sectenta,comanda,numeal,100000,1013,TO_DATE('13-05-2023','dd-mm-yyyy'),'cash');
) INSERT INTO comanda
VALUES(sectenta,comanda,numeal,100000,1020,TO_DATE('13-05-2023','dd-mm-yyyy'),'card');
) INSERT INTO comanda
VALUES(sectenta,comanda,numeal,100000,1018,TO_DATE('07-05-2023','dd-mm-yyyy'),'cash');
) INSERT INTO comanda
VALUES(sectenta,comanda,numeal,100000,1019,TO_DATE('04-05-2023','dd-mm-yyyy'),'cash');
) INSERT ALL
INTO comanda_online VALUES(103182341, 'Str. Pacalici 24, Bucuresti 062084', TO_DATE('10-05-2023','dd-mm-yyyy'))
INTO comanda_online VALUES(103182342, 'Str. Pacalici 24, Bucuresti 062084', TO_DATE('10-05-2023','dd-mm-yyyy'))
INTO comanda_online VALUES(103182343, 'Str. Pacalici 24, Bucuresti 062084', TO_DATE('10-05-2023','dd-mm-yyyy'))
INTO comanda_online VALUES(103182344, 'Str. Pacalici 24, Bucuresti 062084', TO_DATE('10-05-2023','dd-mm-yyyy'))
INTO comanda_online VALUES(103182345, 'Str. Pacalici 24, Bucuresti 062084', TO_DATE('10-05-2023','dd-mm-yyyy'))
SELECT * FROM DUAL;

) INSERT ALL
INTO comanda_continut VALUES(101000, 7, 1000000000, 2)
INTO comanda_continut VALUES(101000, 7, 103182341, 1)
INTO comanda_continut VALUES(40741, 7, 103182341, 1)
INTO comanda_continut VALUES(103182341, 7, 103182341, 1)
INTO comanda_continut VALUES(460000, 1, 107164682, 1)
INTO comanda_continut VALUES(105000, 1, 107164682, 1)
INTO comanda_continut VALUES(25200, 2, 111549221, 1)
INTO comanda_continut VALUES(104000, 9, 1148729164, 1)
INTO comanda_continut VALUES(104000, 16, 1148729164, 1)
INTO comanda_continut VALUES(104000, 1, 118511705, 1)
INTO comanda_continut VALUES(101000, 7, 1221846484, 1)
INTO comanda_continut VALUES(101000, 7, 1260783187, 1)
INTO comanda_continut VALUES(103000, 1, 1260783187, 1)
INTO comanda_continut VALUES(103000, 9, 1260783187, 1)
INTO comanda_continut VALUES(40741, 1, 129458728, 1)
INTO comanda_continut VALUES(40741, 1, 129458728, 1)
INTO comanda_continut VALUES(103000, 1, 133461169, 1)
INTO comanda_continut VALUES(103000, 1, 1371823410, 2)
INTO comanda_continut VALUES(103000, 1, 1371823410, 2)
INTO comanda_continut VALUES(103000, 1, 1371823410, 2)
INTO comanda_continut VALUES(40741, 1, 1371823410, 2)
INTO comanda_continut VALUES(107000, 1, 1371823410, 2)
INTO comanda_continut VALUES(104000, 1, 1371823410, 2)
INTO comanda_continut VALUES(101000, 7, 1496005751, 20)
SELECT * FROM DUAL;

COMMIT;

```

5 rows inserted.	1 row updated.	1 row inserted.	1 row inserted.
10 rows inserted.	1 row updated.	1 row inserted.	1 row inserted.
15 rows inserted.	1 row updated.	1 row inserted.	5 rows inserted.
14 rows inserted.	1 row updated.	1 row inserted.	24 rows inserted.
6 rows inserted.	1 row updated.	1 row inserted.	Commit complete.
12 rows inserted.	1 row updated.	1 row inserted.	
62 rows inserted.	1 row updated.	1 row inserted.	
83 rows inserted.	1 row updated.	1 row inserted.	
12 rows inserted.	1 row updated.	1 row inserted.	
10 rows inserted.	1 row updated.	1 row inserted.	
21 rows inserted.	1 row updated.	1 row inserted.	
5 rows inserted.	1 row updated.	1 row inserted.	
34 rows inserted.	1 row inserted.	1 row inserted.	
21 rows inserted.	1 row inserted.	1 row inserted.	
	1 row inserted.	1 row inserted.	

12. Cereri

Formulați în limbaj natural și implementați 5 cereri SQL complexe ce vor utiliza, în ansamblul lor, următoarele elemente:

- subcereri sincronizate în care intervin cel puțin 3 tabele
- subcereri nesincronizate în clauza FROM
- grupări de date cu subcereri nesincronizate în care intervin cel puțin 3 tabele, funcții grup, filtrare la nivel de grupuri (în cadrul aceleiași cereri)
- ordonări și utilizarea funcțiilor NVL și DECODE (în cadrul aceleiași cereri)
- utilizarea a cel puțin 2 funcții pe șiruri de caractere, 2 funcții pe date calendaristice, a cel puțin unei expresii CASE
- utilizarea a cel puțin 1 bloc de cerere (clauza WITH)

12.1 Să se afișeze jocul, descrierea sa ("Fara descriere" dacă nu are), "Da" sau "Nu" dacă jocul are multiplayer, luna și anul în care au fost lansate prima dată jocurile pentru jocurile lansate înainte de 2018 sortate după data lansării. (clauza WITH, NVL și DECODE în aceeași cerere, ordonare, subcerere nesincronizată în clauza FROM, 2 funcții pe date calendaristice, 1 funcție pe șir de caractere)

```
WITH lans_init AS (SELECT MIN(data_lansare) data_init, id_joc
```

```
FROM versiune
```

```
GROUP BY id_joc)
```

```
SELECT id_joc, titlu, NVL(descriere, 'Fara descriere') descriere, DECODE(multiplayer, 1, 'Da', 'Nu') multiplayer, TO_CHAR(data_init, 'mm-yyyy') data_lansare
```

```
FROM joc JOIN lans_init USING (id_joc)
```

```
WHERE TO_NUMBER(TO_CHAR(data_init, 'yyyy')) < 2018
```

```
ORDER BY data_init;
```

```
WITH lans_init AS (SELECT MIN(data_lansare) data_init, id_joc
FROM versiune
GROUP BY id_joc)
SELECT id_joc, titlu, NVL(descriere, 'Fara descriere') descriere, DECODE(multiplayer, 1, 'Da', 'Nu') multiplayer, TO_CHAR(data_init, 'mm-yyyy') data_lansare
FROM joc JOIN lans_init USING (id_joc)
WHERE TO_NUMBER(TO_CHAR(data_init, 'yyyy')) < 2018
ORDER BY data_init;
```

ID_JOC	TITLU	DESCRIERE	MULTIPLAYER	DATA_LANSARE
1	109000 Roblox	Fara descriere	Da	09-2006
2	104000 Minecraft Bedrock	Fara descriere	Da	10-2011
3	102000 Counter Strike 2	Fara descriere	Da	08-2012
4	108000 Geometry Dash	Fara descriere	Nu	08-2013
5	107000 The Witcher 3	Fara descriere	Nu	05-2015
6	460000 Rainbow Six: Siege	Rainbow Six: Siege este un shooter tactic de elita pe echipe, unde planificarile superioare si executiile trimfa.	Da	12-2015
7	100290 Nier: Automata	Nier: Automata spune povestea androizilor 2B, 9S si A2 si lupta lor pentru recapatarea distopiei controlate de masinarii puternice.	Nu	02-2017
8	105000 Ghost Recon: Wildlands	Fara descriere	Da	03-2017

12.2 Să se afișeze jocurile (id, titlu, gen) care sunt și pe playstation, iar acea versiune se afla în stoc într-un magazin. (subcerere sincronizată în care intervin cel puțin 3 tabele, 1 funcție pe șiruri de caractere)

```
SELECT id_joc, titlu, gen
FROM joc j
WHERE EXISTS(SELECT id_platforma FROM versiune JOIN platforma USING (id_platforma)
              JOIN stoc USING (id_joc, id_platforma)
              WHERE UPPER(ume_platforma) like '%PLAYSTATION%' AND id_joc = j.id_joc AND
nr_produce > 0 AND nr_produce IS NOT NULL);
```

```
SELECT id_joc, titlu, gen
FROM joc j
WHERE EXISTS(SELECT id_platforma FROM versiune JOIN platforma USING (id_platforma)
              JOIN stoc USING (id_joc, id_platforma)
              WHERE UPPER(ume_platforma) like '%PLAYSTATION%' AND id_joc = j.id_joc AND nr_produce > 0 AND nr_produce IS NOT NULL);
```

	ID_JOC	TITLU	GEN
1	100290	Nier: Automata	Action JRPG
2	460000	Rainbow Six: Siege	Team Shooter
3	105000	Ghost Recon: Wildlands	Open World
4	106000	Cyberpunk 2077	Action RPG
5	25200	Soul Hackers 2	Turn-based JRPG

12.3 Se cer orașele în care sunt cel puțin 2 magazine și media salariilor angajaților din ele pentru orașele care au media salariului peste media totală a tuturor orașelor. (grupări de date cu subcereri nesincronizate în care intervin cel puțin 3 tabele, funcții grup, filtrare la nivel de grupuri)

```
SELECT id_oras, nume_oras, ROUND(AVG(salariu)) Salariu_mediu
FROM oras JOIN magazin USING (id_oras)
JOIN angajat USING (id_magazin)
GROUP BY id_oras, nume_oras
HAVING AVG(salariu) > (SELECT AVG(AVG(salariu))
                       FROM oras JOIN magazin USING (id_oras)
                       JOIN angajat USING (id_magazin)
                       GROUP BY id_oras)
```

AND COUNT(DISTINCT id_magazin) >= 2;

```
SELECT id_oras, nume_oras, ROUND(AVG(salariu)) Salariu_mediu
FROM oras JOIN magazin USING (id_oras)
JOIN angajat USING (id_magazin)
GROUP BY id_oras, nume_oras
HAVING AVG(salariu) > (SELECT AVG(AVG(salariu))
                        FROM oras JOIN magazin USING (id_oras)
                        JOIN angajat USING (id_magazin)
                        GROUP BY id_oras)
AND COUNT(DISTINCT id_magazin) >= 2;
```

ID_ORAS	NUME_ORAS	SALARIU_MEDIU
1	12 Galati	2650

12.4 Să se afișeze pentru fiecare magazin (pe lângă id și nume): pentru magazinele din Romania numărul de angajati, pentru magazinele din Marea Britanie media salariilor angajaților, pentru magazinele din Olanda cel mai mare salariu, iar pentru restul magazinelor cel mai mic salariu. (expresia CASE, grupare, funcții grup)

```
SELECT id_magazin, nume_magazin,
CASE LOWER(nume_tara)
    WHEN 'romania' THEN COUNT(id_angajat)
    WHEN 'marea britanie' THEN ROUND(AVG(salariu))
    WHEN 'olanda' THEN MAX(salariu)
    ELSE MIN(salariu)
END "Raspuns"
FROM angajat JOIN magazin USING (id_magazin)
    JOIN oras USING (id_oras)
    JOIN tara USING (id_tara)
GROUP BY id_magazin, nume_magazin, nume_tara;
```

```
SELECT id_magazin, nume_magazin,
CASE LOWER(nume_tara)
    WHEN 'romania' THEN COUNT(id_angajat)
    WHEN 'marea britanie' THEN ROUND(AVG(salariu))
    WHEN 'olanda' THEN MAX(salariu)
    ELSE MIN(salariu)
END "Raspuns"
FROM angajat JOIN magazin USING (id_magazin)
    JOIN oras USING (id_oras)
    JOIN tara USING (id_tara)
GROUP BY id_magazin, nume_magazin, nume_tara;
```

ID_MAGAZIN	NUME_MAGAZIN	Raspuns
1	300 Gameshop AFI	2
2	301 Gameshop Centru Vechi	3
3	302 Gamecorner Tiglina	1
4	303 Gamecorner Gorjului	1
5	304 Gameshop Cluj	1
6	305 Gameshop Amsterdam	3000
7	306 Megagame Brasov	3
8	307 Gamecorner Walworth	1000
9	308 Gameshop Roma	10000
10	309 Gameshop Rotterdam	700
11	310 Megashop Londra	667
12	311 Gamecorner Budapesta 1	502
13	312 Gamecorner Potcoava	1
14	313 Gameshop Milano	1200
15	314 Gamecorner Budapesta 2	1029

12.5 Să se afișeze pentru fiecare client numărul total de jocuri cumparate și numărul de jocuri cumparate online. (grupare, funcții grup)

```
SELECT cl.id_client, nume, prenume, SUM(numar) total_cumparat, SUM(NVL2(co.id_comanda,
numar, 0)) cumparat_online
```

```
FROM client cl JOIN comanda c ON (cl.id_client = c.id_client)
```

```
JOIN comanda_contine cc ON (cc.id_comanda = c.id_comanda)
```

```
LEFT JOIN comanda_online co ON (c.id_comanda = co.id_comanda)
```

```
GROUP BY cl.id_client, nume, prenume;
```

```
SELECT cl.id_client, nume, prenume, SUM(numar) total_cumparat, SUM(NVL2(co.id_comanda, numar, 0)) cumparat_online
FROM client cl JOIN comanda c ON (cl.id_client = c.id_client)
JOIN comanda_contine cc ON (cc.id_comanda = c.id_comanda)
LEFT JOIN comanda_online co ON (c.id_comanda = co.id_comanda)
GROUP BY cl.id_client, nume, prenume;
```

	ID_CLIENT	NUME	PRENUME	TOTAL_CUMPARAT	CUMPARAT_ONLINE
1	1000000	Cazan	George	6	3
2	1000001	Onodi	Paul	14	12
3	1000003	Greyson	Henry	20	5
4	1000002	Jansen	Liam	1	1
5	1000004	Cocu	Elena	3	0
6	1000005	De Lucca	Antonio	22	20
7	1000006	Nagy	Zoltan	1	0

13. Actualizări și suprimări

Implementarea a 3 operații de actualizare și 3 de suprimare a datelor utilizând subcereri.

13.1 Să se scadă prețul cu 25% jocurilor publicate de Atlus.

```
UPDATE versiune
```

```
SET pret = pret * 0.75
```

```
WHERE id_joc IN (SELECT id_joc FROM joc
```

```
JOIN contribuie USING (id_joc)
```

```
JOIN editor USING (id_editor)
```

```
WHERE nume_editor = 'Atlus');
```

```
UPDATE versiune
SET pret = pret * 0.75
WHERE id_joc IN (SELECT id_joc FROM joc
JOIN contribuie USING (id_joc)
JOIN editor USING (id_editor)
WHERE nume_editor = 'Atlus');
```

ID_JOC	ID_PLATFORMA	TITLU	PRET
1	40741	1 Persona 5 Royal	60
2	40741	2 Persona 5 Royal	40
3	40741	3 Persona 5 Royal	60
4	40741	4 Persona 5 Royal	60
5	40741	5 Persona 5 Royal	60
6	40741	6 Persona 5 Royal	60
7	40741	7 Persona 5 Royal	60
8	40741	8 Persona 5 Royal	60
9	40742	1 Persona 5 Strikers	40
10	40742	2 Persona 5 Strikers	40
11	40742	3 Persona 5 Strikers	40
12	40742	7 Persona 5 Strikers	40
13	101000	7 Shin Megami Tensei V	60
14	25200	1 Soul Hackers 2	60
15	25200	2 Soul Hackers 2	60
16	25200	4 Soul Hackers 2	60

16 rows updated.

ID_JOC	ID_PLATFORMA	TITLU	PRET
1	40741	1 Persona 5 Royal	45
2	40741	2 Persona 5 Royal	30
3	40741	3 Persona 5 Royal	45
4	40741	4 Persona 5 Royal	45
5	40741	5 Persona 5 Royal	45
6	40741	6 Persona 5 Royal	45
7	40741	7 Persona 5 Royal	45
8	40741	8 Persona 5 Royal	45
9	40742	1 Persona 5 Strikers	30
10	40742	2 Persona 5 Strikers	30
11	40742	3 Persona 5 Strikers	30
12	40742	7 Persona 5 Strikers	30
13	101000	7 Shin Megami Tensei V	45
14	25200	1 Soul Hackers 2	45
15	25200	2 Soul Hackers 2	45
16	25200	4 Soul Hackers 2	45

13.2 Să se crească salariul cu 400 angajatilor din Romania angajati inainte de mai 2023.

UPDATE angajat

SET salariu = salariu + 400

WHERE id_magazin IN (SELECT id_magazin FROM magazin

JOIN oras USING (id_oras)

JOIN tara USING (id_tara)

WHERE nume_tara = 'Romania')

AND data_angajare < TO_DATE('MAY-2023','MON-YYYY');

```

UPDATE angajat
SET salariu = salariu + 400
WHERE id_magazin IN (SELECT id_magazin FROM magazin
                     JOIN oras USING (id_oras)
                     JOIN tara USING (id_tara)
                     WHERE nume_tara = 'Romania')
AND data_angajare < TO_DATE('MAY-2023','MON-YYYY');

```

ID_ANGAJAT	DATA_ANGAJARE	SALARIU
1	1000 11-SEP-06	501
2	1001 11-SEP-19	1700
3	1002 07-MAY-22	2050
4	1003 11-MAY-22	2000
5	1004 13-MAY-23	1970
6	1005 10-OCT-22	2300
7	1006 12-FEB-23	2500
8	1007 11-NOV-22	2400
9	1008 03-JUN-23	3000
10	1009 23-SEP-20	1000
11	1010 19-JUL-19	3000
12	1011 03-JUN-23	2000
13	1012 15-APR-98	1000
14	1013 16-APR-01	10000
15	1014 03-JUN-23	700
16	1015 13-MAY-23	632
17	1016 12-MAY-23	701
18	1017 13-MAY-23	502
19	1018 09-MAY-23	3000
20	1019 08-MAY-23	1200
21	1020 13-MAY-23	1029

ID_ANGAJAT	DATA_ANGAJARE	SALARIU
1	1000 11-SEP-06	901
2	1001 11-SEP-19	2100
3	1002 07-MAY-22	2450
4	1003 11-MAY-22	2400
5	1004 13-MAY-23	1970
6	1005 10-OCT-22	2700
7	1006 12-FEB-23	2900
8	1007 11-NOV-22	2800
9	1008 03-JUN-23	3000
10	1009 23-SEP-20	1400
11	1010 19-JUL-19	3400
12	1011 03-JUN-23	2000
13	1012 15-APR-98	1000
14	1013 16-APR-01	10000
15	1014 03-JUN-23	700
16	1015 13-MAY-23	632
17	1016 12-MAY-23	701
18	1017 13-MAY-23	502
19	1018 09-MAY-23	3000
20	1019 08-MAY-23	1200
21	1020 13-MAY-23	1029

9 rows updated.

13.3 Să se dubleze numărul jucătorilor din jocurile online ce se află pe PC.

UPDATE joc_online

SET nr_jucatori = nr_jucatori * 2

WHERE id_joc IN (SELECT id_joc FROM versiune

JOIN platforma USING (id_platforma)

WHERE tip_platforma = 'PC');

```
UPDATE joc_online
SET nr_jucatori = nr_jucatori * 2
WHERE id_joc IN (SELECT id_joc FROM versiune
                JOIN platforma USING (id_platforma)
                WHERE tip_platforma = 'PC');
```

ID_JOC	NR_JUCATORI
1	460000
2	102000
3	103000
4	104000
5	105000
6	109000

5 rows updated.

ID_JOC	NR_JUCATORI
1	760000
2	2000000
3	1000000
4	6000000
5	40000
6	3560000

13.4 Să se șteargă jocurile cu versiuni de sub 10GB.

DELETE FROM joc

WHERE id_joc IN (SELECT id_joc FROM versiune WHERE dimensiune < 10);

```
DELETE FROM joc
```

```
WHERE id_joc IN (SELECT id_joc FROM versiune WHERE dimensiune < 10);
```

ID_JOC	TITLU
1	100290 Nier: Automata
2	40741 Persona 5 Royal
3	40742 Persona 5 Strikers
4	460000 Rainbow Six: Siege
5	101000 Shin Megami Tensei V
6	102000 Counter Strike 2
7	103000 Brawl Stars
8	104000 Minecraft Bedrock
9	105000 Ghost Recon: Wildlands
10	106000 Cyberpunk 2077
11	107000 The Witcher 3
12	108000 Geometry Dash
13	25200 Soul Hackers 2
14	109000 Roblox

6 rows deleted.

ID_JOC	TITLU
1	100290 Nier: Automata
2	40741 Persona 5 Royal
3	40742 Persona 5 Strikers
4	101000 Shin Megami Tensei V
5	102000 Counter Strike 2
6	106000 Cyberpunk 2077
7	107000 The Witcher 3
8	25200 Soul Hackers 2

13.5 Să se ștergă platformele pentru care nu există jocuri în stoc.

DELETE FROM platforma p

WHERE NOT EXISTS (SELECT id_platforma FROM stoc WHERE p.id_platforma = id_platforma);

```
DELETE FROM platforma p
```

```
WHERE NOT EXISTS (SELECT id_platforma FROM stoc WHERE p.id_platforma = id_platforma);
```

ID_PLATFORMA	NUME_PLATFORMA	TIP_PLATFORMA
1	1 Windows	PC
2	2 Playstation 4	Consola
3	3 Playstation 5	Consola
4	4 Xbox One	Consola
5	5 Xbox Series X	Consola
6	6 Xbox Series S	Consola
7	7 Nintendo Switch	Handheld
8	8 Steam Deck	Handheld
9	9 Android	Telefon
10	10 IOS	Telefon
11	11 Stadia	Cloud
12	12 Linux	PC

3 rows deleted.

ID_PLATFORMA	NUME_PLATFORMA	TIP_PLATFORMA
1	1 Windows	PC
2	2 Playstation 4	Consola
3	3 Playstation 5	Consola
4	4 Xbox One	Consola
5	7 Nintendo Switch	Handheld
6	8 Steam Deck	Handheld
7	9 Android	Telefon
8	10 IOS	Telefon
9	12 Linux	PC

13.6 Să se ștergă clienții care nu au făcut nicio comandă online.

```
DELETE FROM client
```

```
WHERE id_client NOT IN (SELECT id_client
                        FROM comanda JOIN comanda_online USING(id_comanda));
```

	ID_CLIENT	NUME	PRENUME
1	1000000	Cazan	George
2	1000001	Onodi	Paul
3	1000002	Jansen	Liam
4	1000003	Greyson	Henry
5	1000004	Cocu	Elena
6	1000005	De Lucca	Antonio
7	1000006	Nagy	Zoltan

2 rows deleted.

	ID_CLIENT	NUME	PRENUME
1	1000000	Cazan	George
2	1000001	Onodi	Paul
3	1000002	Jansen	Liam
4	1000003	Greyson	Henry
5	1000005	De Lucca	Antonio

14. Cereri complexe

Formulați în limbaj natural și implementați în SQL: o cerere ce utilizează operația *outer-join* pe minimum 4 tabele, o cerere ce utilizează operația *division* și o cerere care implementează analiza top-*n*.

14.1 Ce stoc au jocurile în total de la fiecare dezvoltator în fiecare țară. (outer-join pe minimum 4 tabele)

```
SELECT nume_dezvoltator, nume_tara, SUM(NVL(nr_produce, 0)) numar
FROM dezvoltator d CROSS JOIN tara t
    LEFT JOIN oras o ON (o.id_tara = t.id_tara)
    LEFT JOIN magazin m ON (o.id_oras = m.id_oras)
    LEFT JOIN contribuie c ON (d.id_dezvoltator = c.id_dezvoltator)
    LEFT JOIN stoc s ON (c.id_joc = s.id_joc AND s.id_magazin = m.id_magazin)
GROUP BY nume_dezvoltator, nume_tara;
```

```
SELECT nume_dezvoltator, nume_tara, SUM(NVL(nr_produce, 0)) numar
FROM dezvoltator d CROSS JOIN tara t
    LEFT JOIN oras o ON (o.id_tara = t.id_tara)
    LEFT JOIN magazin m ON (o.id_oras = m.id_oras)
    LEFT JOIN contribuie c ON (d.id_dezvoltator = c.id_dezvoltator)
    LEFT JOIN stoc s ON (c.id_joc = s.id_joc AND s.id_magazin = m.id_magazin)
GROUP BY nume_dezvoltator, nume_tara;
```


14.3 Să se afișeze primele 5 jocuri după numărul total de vânzări. (analiză top-n)

```
SELECT id_joc, titlu, nr_vanzari
FROM (SELECT id_joc, titlu, SUM(numar) nr_vanzari
      FROM joc JOIN comanda_contine USING(id_joc)
      GROUP BY id_joc, titlu
      ORDER BY nr_vanzari DESC)
WHERE ROWNUM <= 5;
```

```
SELECT id_joc, titlu, nr_vanzari
FROM (SELECT id_joc, titlu, SUM(numar) nr_vanzari
      FROM joc JOIN comanda_contine USING(id_joc)
      GROUP BY id_joc, titlu
      ORDER BY nr_vanzari DESC)
WHERE ROWNUM <= 5;
```

	ID_JOC	TITLU	NR_VANZARI
1	101000	Shin Megami Tensei V	25
2	104000	Minecraft Bedrock	17
3	25200	Soul Hackers 2	5
4	40741	Persona 5 Royal	4
5	100290	Nier: Automata	4

15. Optimizarea unei cereri

Optimizarea unei cereri, aplicând regulile de optimizare ce derivă din proprietățile operatorilor algebrei relaționale. Cererea va fi exprimată prin expresie algebrică, arbore algebric și limbaj (SQL), atât anterior cât și ulterior optimizării.

Să se afișeze angajații care lucrează în București și care s-au angajat înainte de 2023.

Neoptimizat:

```
SELECT id_angajat, nume, prenume
FROM angajat JOIN magazin USING (id_magazin)
      JOIN oras USING (id_oras)
WHERE nume_oras = 'Bucuresti' AND TO_NUMBER(TO_CHAR(data_angajare, 'yyyy')) < 2023;
```

```

SELECT id_angajat, nume, prenume
FROM angajat JOIN magazin USING (id_magazin)
      JOIN oras USING (id_oras)
WHERE nume_oras = 'Bucuresti' AND TO_NUMBER(TO_CHAR(data_angajare, 'YYYY')) < 2023;

```

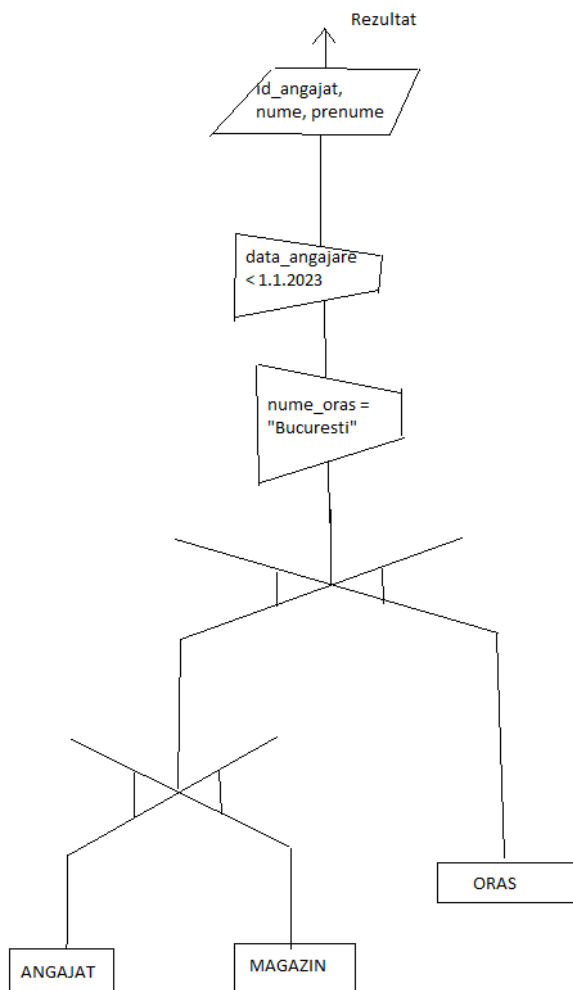
R1 = JOIN(ANGAJAT, MAGAZIN)

R2 = JOIN(R1, ORAS)

R3 = SELECT(R2, nume_oras = "Bucuresti")

R4 = SELECT(R3, data_angajare < 1.1.2023)

Rezultat = R5 = PROJECT(R4, id_angajat, nume, prenume)



Optimizările care trebuie făcute cererii constau în: executarea selecțiilor cât mai devreme posibil, înlocuirea produselor carteziene cu join-uri, executarea celor mai restrictive join-uri prima dată și executarea proiecțiilor la început pentru eliminarea atributelor nefolositoare.

Cererea ajunge astfel:

```
SELECT id_angajat, nume, prenume
```

```
FROM (SELECT id_oras FROM oras WHERE nume_oras = 'Bucuresti')
```

JOIN (SELECT id_magazin, id_oras FROM magazin) USING (id_oras)

JOIN (SELECT id_angajat, nume, prenume, id_magazin FROM angajat

WHERE TO_NUMBER(TO_CHAR(data_angajare, 'yyyy')) < 2023) USING (id_magazin);

```
SELECT id_angajat, nume, prenume
FROM (SELECT id_oras FROM oras WHERE nume_oras = 'Bucuresti')
JOIN (SELECT id_magazin, id_oras FROM magazin) USING (id_oras)
JOIN (SELECT id_angajat, nume, prenume, id_magazin FROM angajat
WHERE TO_NUMBER(TO_CHAR(data_angajare, 'yyyy')) < 2023) USING (id_magazin);
```

R1 = SELECT(ANAGAJAT, data_angajare < 1.1.2023)

R2 = PROJECT(R1, id_angajat, nume, prenume, id_magazin)

R3 = PROJECT(MAGAZIN, id_magazin, id_oras)

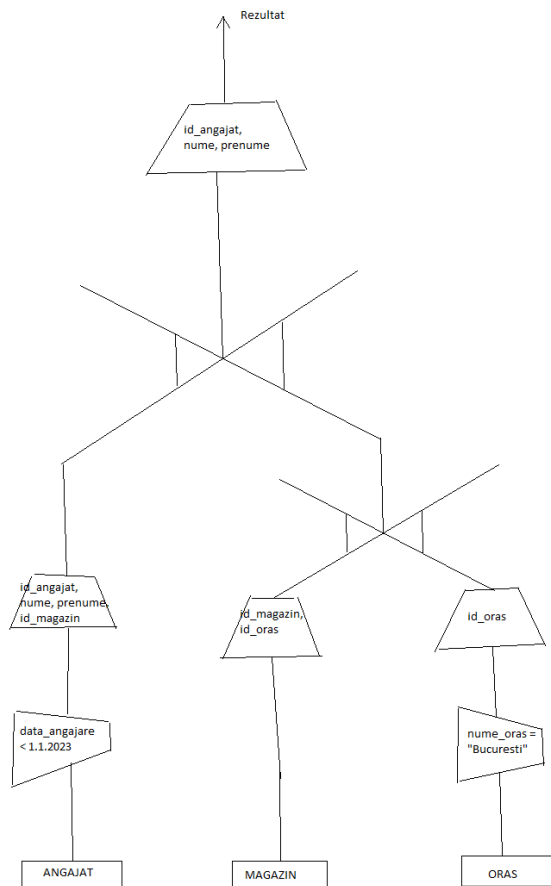
R4 = SELECT(ORAS, nume_oras = "Bucuresti")

R5 = PROJECT(R4, id_oras)

R6 = JOIN(R4, R5)

R7 = JOIN(R2, R6)

Rezultat = R8 = PROJECT(R7, id_angajat, nume, prenume)



Rezultatul amânduror cereri este:

	ID_ANGAJAT	NUME	PRENUME
1	1000	Florice1	Andreea Renata
2	1001	Ionescu	Ion
3	1002	Pinduchi	Andreea Corina
4	1003	Vasiliu	Ana

16. Normalizările BCNF, FN4, FN5 și denormalizarea

16a.1 Forma normală Boyce-Codd

O relație este în BCNF dacă și numai dacă fiecare determinant este o cheie candidat.

Fie relația VERISUNE(id_joc#, id_platforma#, dimensiune, data_lansare, pret):

id_joc#	id_platforma#	dimensiune	data_lansare	pret
100290	1	40.8	17-MAR-17	40
100290	7	15	06-OCT-22	60
40741	1	39.3	21-OCT-22	60
40741	2	33.2	31-MAR-20	40
40741	3	39.3	21-OCT-22	60
40741	5	30	21-OCT-22	60

Între atribute există dependențele:

{id_joc#, id_platforma#} -> {dimensiune, data_lansare, pret}

{data_lansare} -> {id_joc#}

Se aplică regula Casey-Delobel și se aduce relația în BCNF:

id_platforma#	dimensiune	data_lansare#	pret
1	40.8	17-MAR-17	40
7	15	06-OCT-22	60
1	39.3	21-OCT-22	60
2	33.2	31-MAR-20	40
3	39.3	21-OCT-22	60
5	30	21-OCT-22	60

VERSIUNE1 (id_joc, data_lansare#)

VERSIUNE2 (id_platforma#, data_lansare#, dimensiune, pret)

id_joc	data_lansare#
100290	17-MAR-17
100290	06-OCT-22
40741	21-OCT-22
40741	31-MAR-20

16a.2 Forma normală 4

FN4 elimină redundanțele datorate dependenței multiple.

Există relația CONTRIBUIE(id_joc#, id_dezvoltator#, id_editor#), care se află în BCNF:

id_joc#	id_dezvoltator#	id_editor#
40741	110	210
40741	110	220
40741	210	210
40741	210	220
40742	120	210
40742	210	210
101000	110	210
101000	110	220

Avem următoarele dependențe multiple:

ID_JOC -> ID_DEZVOLTATOR

ID_JOC -> ID_EDITOR

După aplicarea lui FN4 vom avea relațiile:

CONTRIBUIE1 (id_joc#, id_dezvoltator#)

CONTRIBUIE2 (id_joc#, id_editor#)

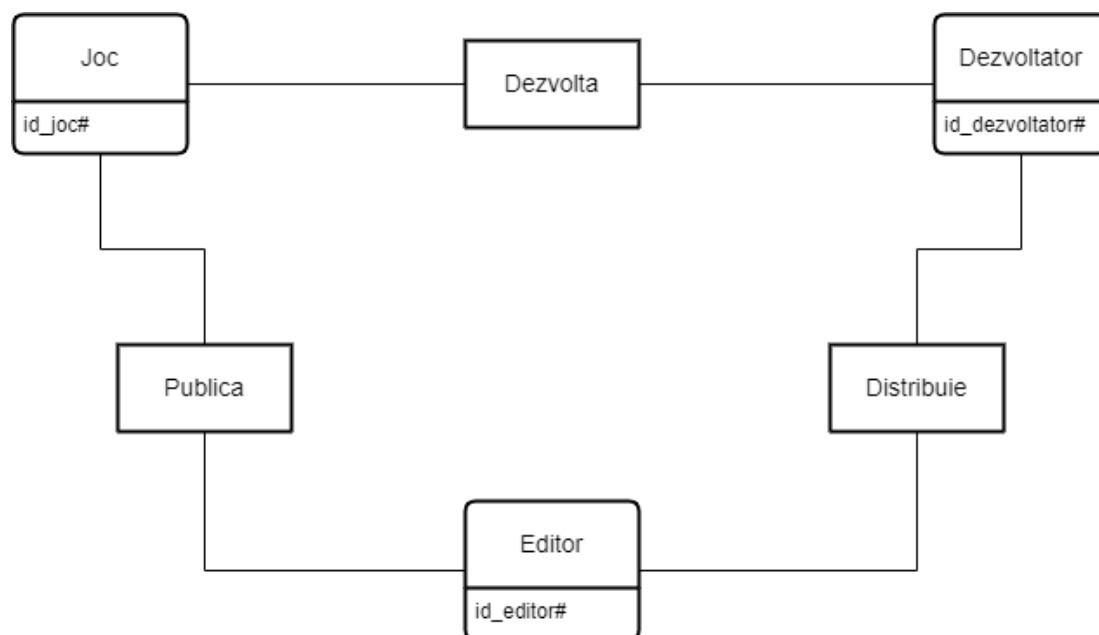
id_joc#	id_dezvoltator#
40741	110
40741	210
40742	120
40742	210
101000	110

id_joc#	id_editor#
40741	210
40741	220
40742	210
101000	210
101000	220

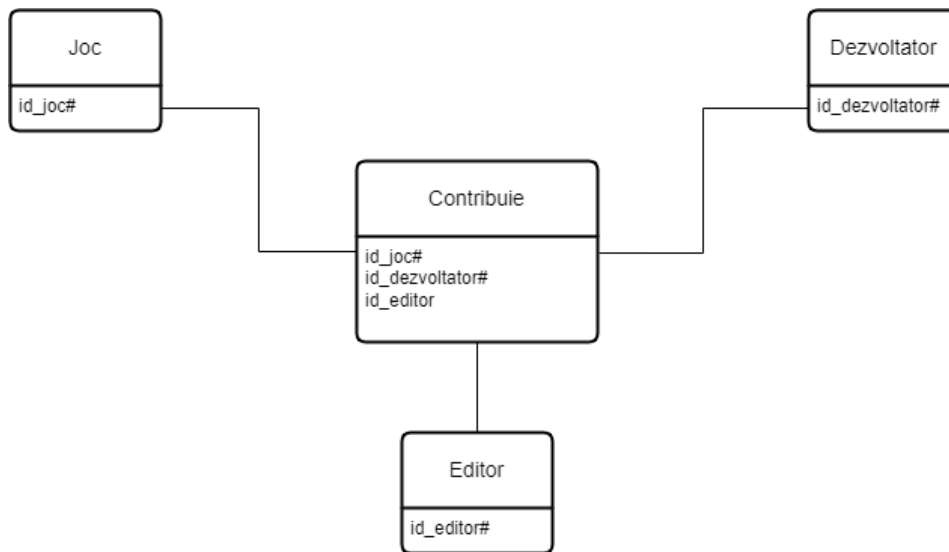
16a.3 Forma normală 5

FN5 își propune eliminarea redundanțelor care apar în relații m:n dependente, adică să nu existe dependențe ciclice.

Un exemplu de dependență ciclică este:



Relațiile DEZVOLTA, PUBLICA și DSTRIBUIE se transformă într-o relație de tip 3 astfel:



16b Denormalizarea

Rolul denormalizării este de a reduce numărul de join-uri efectuate, astfel micșorând timpul de execuție al query-urilor, iar în schimb, se mărește redundanța.

Un prim exemplu pentru aplicarea denormalizării este legat de exemplul de la BCNF, unde din relația VERSIUNE s-au creat VERSIUNE1 cu cheile primare id_platforma și data_lansare și VERSIUNE2 cu cheia primară data_lansare și atributul id_joc. Un avantaj al denormalizării este faptul că atributele data_lansare și id_joc sunt schimbate foarte rar, nefiind necesar crearea unui nou tabel. De asemenea, atributul id_joc este accesat des, denormalizarea făcând această accesare mai rapidă prin eliminarea unui join.

Un alt exemplu este cel de la FN4, în care relația CONTRIBUIE este împărțită în CONTRIBUIE1 și CONTRIBUIE2. Și în acest caz, atributele tabelor nu sunt actualizate des, dar în schimb datele au o rată a interogărilor foarte ridicată. Pentru a afla ce editori publică jocurile unui dezvoltator, inițial trebuie doar 2 join-uri, dar după aplicarea FN4, sunt necesare 4 join-uri. De asemenea, pentru a afla editorii și dezvoltatorii fiecărui joc, după denormalizare se aplică un join în minus. Deci denormalizarea face accesarea datelor mai rapidă.