1. Liceeni din România pot participa la mai multe tabere sponsorizate de companii. Taberele se desfășoară în intervale diverse, au o dată de început, o dată de sfârșit și o locație de desfășurare. Pe parcursul desfășurării taberei, liceeni sunt supravegheați de voluntari.

De asemenea, elevilor le sunt suportate cheltuielile de masa și cazare, opțional și cheltuielile legate de transport.

În cadrul taberei, participanții se pot înscrie la mai multe ateliere, unde se vor aprofunda/dezvolta anumite cunoștințe pe care elevul dorește să pună accent. Atelierele sunt organizate de către voluntari, iar predarea este asigurată de specialiști.

În funcție de numărul de participanți, liceeni sunt împărțiți pe grupe aproximativ egale.

De asemenea, în cadrul taberei au loc sesiuni speciale în care sunt chemați invitați de onoare.

La final, fiecare elev prezintă un proiect cu o tematică specifică unui atelier la care a luat parte. Proiectele se realizează individual.

În vederea realizării proiectului, elevul trebuie să își aleagă un unic coordonator.

1. - Un elev poate participa la o singura tabară.

- O companie poate sponsoriza una sau mai multe tabere.

- O tabără poate fi sponsorizată de una sau mai multe companii.

- Un voluntar poate superviza unul sau mai mulți elevi.

- Atelierele sunt organizate de cel puțin un voluntar.

- O tabără pune la dispoziție cel puțin un atelier.

- Un elev trebuie să se înscrie la cel puțin un atelier.

- Un voluntar poate organiza un singur atelier.

- Un elev are un singur coordonator.

-Un coordonator poate superviza mai multe proiecte sau niciunul.

- Toți voluntarii participă la toate taberele.

- Un licean aparține unei singure grupe.

- O tabără are cel puțin o sesiune specială.

- O sesiune specială are cel puțin un invitat de onoare.

**3.**

|  |  |  |
| --- | --- | --- |
| ENTITATE | CHEIE PRIMARA | OBSERVATII |
| Companie | id\_companie | Se întocmește un contract cu companiile care fac sponsorizările în schimbul promovarii imaginii lor prin plasare de produse, postări pe rețele de socializare. |
| Licean | id\_licean | Liceanul poate proveni atât din mediul rural cât și din mediul urban. De asemenea, poate fi atât din ciclul inferior (clasele IX-X), cât și superior (XI-XII). Liceanul trebuie să fie apt fizic pentru a participa la activități. |
| Tabara | id\_tabara | Taberele se desfășoară pe parcursul mai multor zile, iar în funcție de numărul de înscrieri poate exista un număr variat de participanți. |
| Voluntar | id\_voluntar | Voluntarii sunt aleși pe baza CV-ului depus, pot fi foști participanti ale taberelor trecute, sau pot fi tineri care au experiență în voluntariat. |
| Atelier | id\_atelier | Pe parcursul unei zile se desfășoară mai multe ateliere (eg. de matematică, fizica, chimie, public-speaking, teatru, etc) cu lucrări practice și proiecte în echipă. |
| Proiect | id\_proiect | Proiectele se realizează individual și o temă unică la nivel de atelier. |
| Specialist | id\_specialist | Specialiștii sunt selectați de elevi pentru îndrumare în ceea ce privește proiectul final. Specialiștii sunt fie olimpici, fie cadre didactice care susțin atelierele. |
| Grupa | id\_grupa | Liceeni sunt împartiti în grupe pentru a facilita lucrul in echipa. De obicei grupele au între 4 si 6 elevi. |
| Sesiune | id\_sesiune | Sesiunile speciale se desfășoară sub formatul de tip Q&A (întrebare și răspuns), se poartă discuții libere și se dezbat diverse subiecte. |
|  | | |

|  |  |  |
| --- | --- | --- |
| Invitat | id\_invitat | Invitații sunt persoane publice, antreprenori, speakeri motivationali, psihologi, etc. |

**4**.Descrierea relatiilor

|  |  |  |
| --- | --- | --- |
| RELATIE | CARDINALITATE | OBSERVATII |
| sponsorizeaza | Companie-Tabara  many-to-many | O companie poate sponsoriza mai multe tabere.  O tabără poate fi sponsorizată de mai multe companii. |
| participa | Tabara - Licean  one-to-many | Un licean poate participa la o singură tabără. La o tabără participă mai mulți liceeni. |
| supervizeaza | Voluntar – Licean  Many-to-many | Un voluntar poate superviza mai mulți liceeni sau niciunul. Un licean poate fi supervizat de mai multi voluntari. |
| realizeaza | Licean – Proiect  One-to-one | Un licean realizează un singur proiect.  Un proiect este realizat de un singur licean. |
| organizeaza | Atelier - Voluntar  One-to-many | Un voluntar poate organiza un singur atelier. Atelierele pot fi organizate de mai multi voluntari. |
| se inscrie | Licean – Atelier  Many-to-many | Un licean se poate înscrie la mai multe ateliere.  La un atelier se pot înscrie mai mulți liceeni. |
| coordoneaza | Specialist – Licean  One-to-many | Un specialist poate coordona mai mulți liceeni. Un licean este coordonat doar de un specialist. |
| sustine | Specialist-Atelier  One-to-one | Un specialist susține un singur atelier.  Un atelier este sustinut de un singur specialist. |
| contine | Tabără – Atelier  One – to -many | O tabără conține mai multe ateliere. Un atelier aparține de o singura tabără. |
| apartine | Grupa – Licean  One-to-many | Un licean face parte doar dintr-o grupă, iar o grupa contine mai mulți liceeni. |
| are loc | Tabără – Sesiune  One-to-many | O tabără are mai multe sesiuni, iar o sesiune apartine unei singure tabere. |
| invita | Sesiune- Invitat  One-to-many | La o sesiune pot fi chemați mai multi invitati, iar un invitat este chemat doar la o sesiune. |

**5.** Descrierea atributelor

Obs: Cheile primare ale tuturor entităților și relațiilor sunt de tip integer cu dimensiunea maximă de 5.

ENTITATE: Companie

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Atribut | Tip | Dimensiune/precizie | Valori posibile si valori default | Observatii, obligatoriu/optional |
| Nume | string | 30 |  | NOT NULL |
| An infiintare | integer | 4 |  |  |

ENTITATE: Tabara

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Atribut | Tip | Dimensiune/precizie | Valori posibile si valori default | Observatii, obligatoriu/optional |
| Data\_inceput | date |  |  | NOT NULL |
| Data\_final | date |  |  | NOT NULL |
| Locatie | string | 20 |  | NOT NULL |

ENTITATE: Licean

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Atribut | Tip | Dimensiune/precizie | Valori posibile si valori default | Observatii, obligatoriu/optional |
| Nume | string | 20 |  | NOT NULL |
| Prenume | string | 30 |  | NOT NULL |
| Clasa | integer | 2 | [9,12] |  |
| Liceu | string | 30 |  |  |
| Id\_tabara | integer | 5 |  | Reprezintă id-ul taberei la care participă liceanul. Trebuie să corespundă unei valori din tabelul TABARA (FK). |
| Id\_grupa | integer | 5 |  | Reprezintă id-ul grupei din care face parte. Trebuie să corespundă unei valori din tabelul GRUPA (FK). |
| Id\_specialist | integer | 5 |  | Reprezintă id-ul specialistului care îl coordonează. Trebuie să corespundă unei valori din tabelul SPECIALIST (FK). |
| Cheltuieli\_transport | integer | 3 |  |  |

ENTITATE: Voluntar

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Atribut | Tip | | Dimensiune/precizie | | Valori posibile si valori default | | Observatii, obligatoriu/optional |
| Nume | string | | 20 | |  | | Not null |
| Prenume | string | | 30 | |  | | Not null |
| Data\_nasterii | date | |  | |  | | Not null |
| Id\_atelier | | integer | | 5 | |  | Reprezintă id-ul atelierului pe care l-a organizat. Trebuie să corespundă unei valori din tabelul ATELIER (FK). |

ENTITATE: Atelier

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Atribut | Tip | Dimensiune/precizie | Valori posibile si valori default | Observatii, obligatoriu/optional |
| Nume | string | 20 |  | NOT NULL |
| Ore\_total | integer | 2 |  |  |
| Materie | string | 20 |  |  |
| Id\_tabara | integer | 5 |  | Reprezintă id-ul taberei din care face parte atelierul. Trebuie să corespundă unei valori din tabelul TABARA (FK). |
| Id\_specialist | integer | 5 |  | Reprezintă id-ul specialistului care sustine atelierul. Trebuie să corespundă unei valori din tabelul SPECIALIST (FK). |

ENTITATE: Proiect

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Atribut | Tip | Dimensiune/precizie | Valori posibile si valori default | Observatii, obligatoriu/optional |
| Titlu | string | 20 |  | NOT NULL |
| Materie | string | 20 |  |  |
| Id\_licean | integer | 5 |  | Reprezintă id-ul liceanului care realizeaza proiectul. Trebuie să corespundă unei valori din tabelul LICEAN (FK). |

ENTITATE: Specialist

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Atribut | Tip | Dimensiune/precizie | Valori posibile si valori default | Observatii, obligatoriu/optional |
| Nume | string | 20 |  | NOT NULL |
| Prenume | string | 30 |  | NOT NULL |
| Specializare | string | 20 |  | NOT NULL |
| Ani\_experienta | integer | 2 |  |  |
| Salariu | integer | 4 |  |  |

ENTITATE: Sesiune

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Atribut | Tip | Dimensiune/precizie | Valori posibile si valori default | Observatii, obligatoriu/optional |
| Data\_sesiune | date |  |  | NOT NULL |
| Durata | integer | 1 |  | Numărul de ore al unei sesiuni, NOT NULL |
| Id\_tabara | integer | 5 |  | Reprezintă id-ul taberei din care face parte sesiunea. Trebuie să corespundă unei valori din tabelul TABARA (FK). |

ENTITATE: Invitat

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Atribut | Tip | Dimensiune/precizie | Valori posibile si valori default | Observatii, obligatoriu/optional |
| Nume | string | 20 |  |  |
| Prenume | string | 30 |  |  |
| Ocupatie | string | 20 |  |  |
| Id\_sesiune | integer | 5 |  | Reprezintă id-ul sesiunii la care a fost invitat. Trebuie să corespundă unei valori din tabelul SESIUNE (FK). |

ENTITATE: Grupa

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Atribut | Tip | Dimensiune/precizie | Valori posibile si valori default | Observatii, obligatoriu/optional |
| Numar\_liceeni | integer | 1 |  | Numărul de liceeni dintr-o grupa, NOT NULL |

RELATIE: Sponsorizeaza

Cheie primară compusă formată din: id\_companie și id\_tabara.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Atribut | Tip | Dimensiune/precizie | Valori posibile si valori default | Observatii, obligatoriu/optional |
| Suma | integer | 6 |  | Suma oferită drept sponsorizare, NOT NULL |

RELATIE: Supervizeaza

Cheie primară compusă formată din: id\_licean și id\_voluntar.

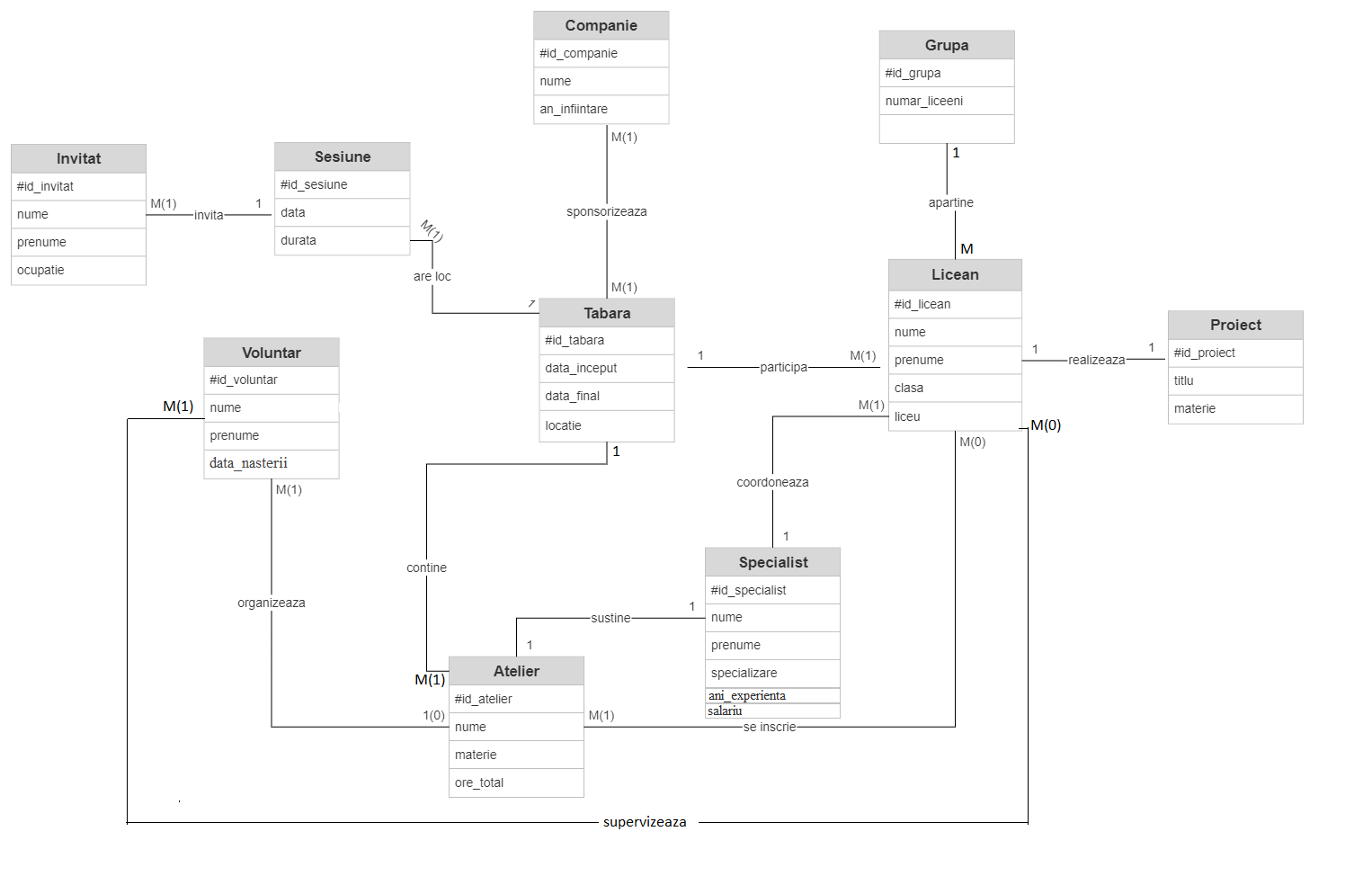
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Atribut | Tip | Dimensiune/precizie | Valori posibile si valori default | Observatii, obligatoriu/optional |
| Nr\_de\_ore | integer | 1 |  | Numarul de ore lucrate pe zi de voluntar, NOT NULL |

RELATIE: Se inscribe

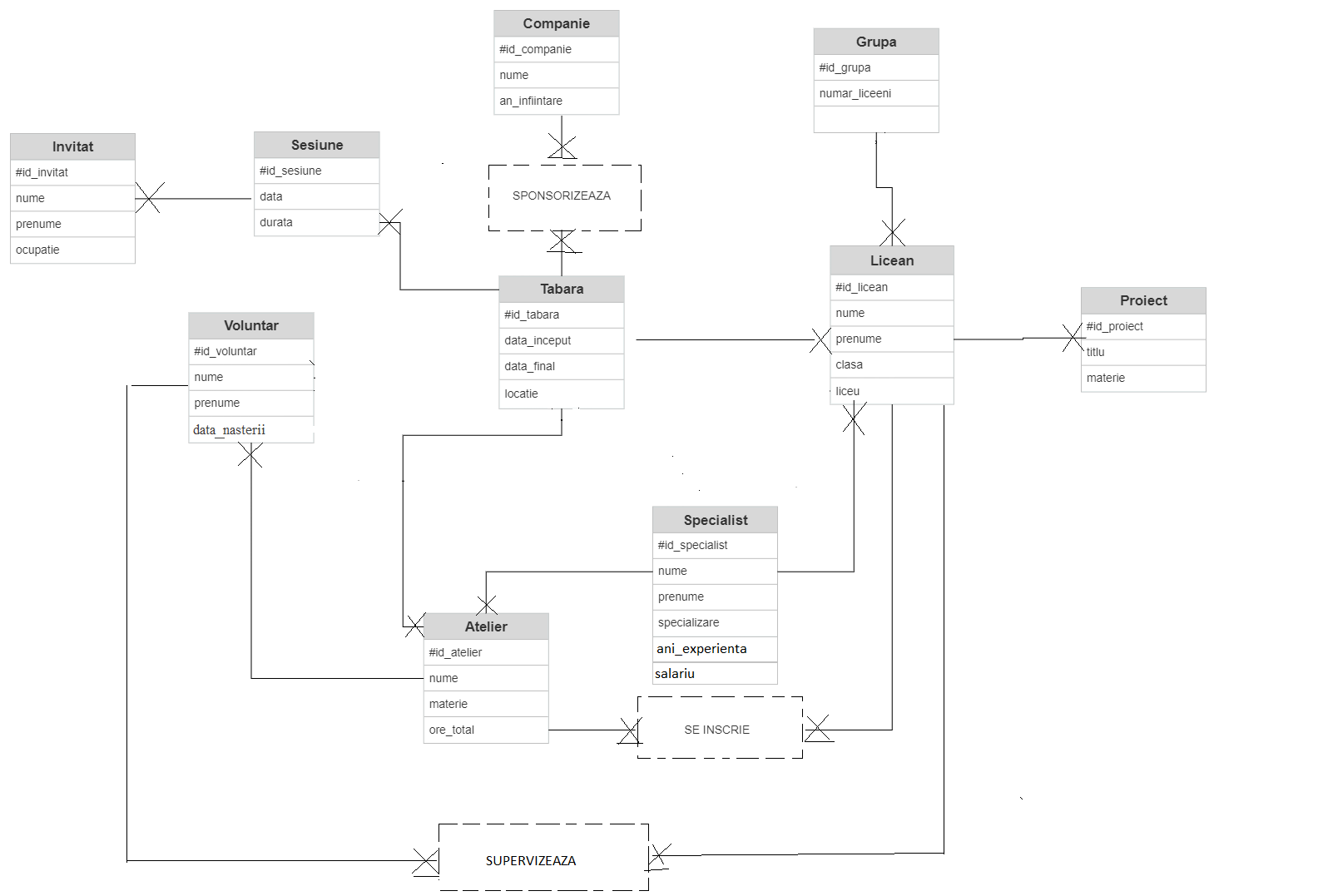
Cheie primară compusă formată din: id\_atelier și id\_licean.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Atribut | Tip | Dimensiune/precizie | Valori posibile si valori default | Observatii, obligatoriu/optional |
| data\_inscriere | data |  |  | NOT NULL |

**6.** Diagrama ERD



**7**. Diagrama conceptuală



**8.** Schemele relaționale

**COMPANIE**(*id\_companie#,* nume, an\_infiintare)

**SPONSORIZEAZA**(*id\_companie#,* *id\_tabara#,* suma)

**TABARA**(*id\_tabara#,* data\_inceput, data\_final, locatie)

**SESIUNE**(*id\_sesiune#,* data, durata, id\_tabara)

**INVITAT**(*id\_invitat#,* nume, prenume, ocupatie, id\_sesiune)

**LICEAN**(*id\_licean*#, nume, prenume, clasa, liceu, cheltuieli\_transport, id\_tabara, id\_grupa, id\_specialist)

**PROIECT**(*id\_proiect*#, titlu, materie, id\_licean)

**GRUPA**(*id\_grupa*#, numar\_liceeni)

**VOLUNTAR**(*id\_voluntar*#, nume, prenume, data\_nasterii, id\_atelier)

**ATELIER**(*id\_atelier*#, nume, materie, ore\_total, id\_tabara, id\_specialist)

**SPECIALIST**(*id\_specialist*#, nume, prenume, specializare, ani\_experienta, salariu)

**SUPERVIZEAZA**(*id\_voluntar*#, *id\_licean*#, nr\_de\_ore)

**SE INSCRIE**(*id\_licean#,* *id\_atelier*#, data\_inscriere)

**9**. Normalizarea

Forma normală 1 (FN1)

O relaţie se află în prima formă normală dacă fiecărui atribut care o compune îi corespunde o valoare indivizibilă. Concis, o relație se află în forma normală 1 dacă există un identificator unic.

Pentru a demonstra că relațiile avute în diagramă sunt în forma normală 1 se vor lua câteva exemple.

De exemplu, în relația COMPANIE cheia primară este id\_companie, reprezentând un identificator unic (fiecare companie are un id unic). Astfel, atributul id\_companie este o valoare indivizibilă, deci relația se află în forma normală 1, deoarece există un identificator unic pentru toate intrările din tabel.

De asemenea, relația SUPERVIZEAZĂ are o cheie primară compusă formată din id\_voluntar și id\_licean. Aceasta este identificator unic pentru entitate, astfel că atributele id\_voluntar și id\_licean formează împreună o valoare indivizibilă, deci relația se află în forma normală 1 (există identificator unic pentru toate intrările din tabel).

Pentru a exemplifica cum se aplică această normalizare se va crea un exemplu imaginar, mai precis se va presupune că pentru un invitat există mai multe ocupații (ocupație).

Cum arată inițial relația invitat nonFN1:

|  |  |
| --- | --- |
| id\_invitat# | ocupație |
| 1 | psiholog, antreprenor, filantrop |
| 2 | antreprenor |
| 3 | atlet, ambasador |

Relația prezentată mai sus nu este în FN1, deoarece atributului ocupație nu îi corespunde o valoare indivizibilă.

După aplicarea primei forme normale, relația va arăta astfel:

|  |  |
| --- | --- |
| id\_invitat# | ocupație |
| 1 | psiholog |
| 1 | antreprenor |
| 1 | filantrop |
| 2 | antreprenor |
| 3 | atlet |
| 3 | ambasador |

Se observă că după normalizare fiecărui atribut care o compune îi corespunde o valoare indivizibilă.

Forma normală 2 (FN2)

O relație se află în forma normală 2 dacă și numai dacă aceasta se află în prima formă normală și dacă fiecare atribut care nu face parte din cheie este dependent de întreaga cheie primară.

Dacă o relație are o cheie primară formată dintr-un singur atribut, atunci ea este automat în FN2, așa cum este și cazul relațiilor COMPANIE, TABĂRĂ, sau LICEAN, de exemplu.

Pentru a exemplifica această fomă normală se va arăta un exemplu care pornește de la o relație care inițial nu era în forma normală 2 și care se va transforma într-o relație din FN2.

Relația arăta inițial în felul următor:

SPONSORIZEAZĂ inițial

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Id\_companie# | Id\_tabara# | suma | Data\_început | Data\_final | locație |
| 1 | 1 | 15.000 | 05-11-2018 | 09-11-2018 | Constanța |
| 1 | 2 | 23.000 | 13-04-2019 | 18-04-2019 | București |
| 2 | 3 | 100.000 | 12-08-2019 | 18-08-2019 | Sibiu |
| 2 | 4 | 15.000 | 10-02-2020 | 14-02-2020 | Cluj-Napoca |
| 1 | 5 | 18.000 | 11-09-2020 | 18-09-2020 | București |

Relația de mai sus se află în FN1, deoarece există identificator unic pentru toate intrările din tabel.

Pentru a fi în FN2 trebuie ca fiecare atribut care nu este cheie să fie dependent de întrega cheie primară. În acest caz, atributele *data\_început, data\_final și locație* trebuie să depindă de întrega cheie primară **id\_companie#** și **id\_tabără#**, dar se poate observa faptul că acestea nu depind direct de toată cheia primară. Acest fapt se explică prin existența dependenței directe dintre **id\_tabără#,**  **data\_început, data\_final și locație**, ceea ce înseamnă că data\_început, data\_final și locație depind doar de o parte a cheii primare, anume de *id\_tabără*#. Astfel, relația prezentată mai sus nu se află în a doua formă normală.

Dependențele sunt:

{id\_tabără} -> {data\_început, data\_final, locație}

{id\_companie, id\_tabără} -> suma

Transformarea în FN2:

SPONSORIZEAZĂ

|  |  |  |
| --- | --- | --- |
| id\_companie# | Id\_tabara# | suma |
| 1 | 1 | 15.000 |
| 1 | 2 | 23.000 |
| 2 | 3 | 100.000 |
| 2 | 4 | 15.000 |
| 1 | 5 | 18.000 |

TABĂRĂ

|  |  |  |  |
| --- | --- | --- | --- |
| id\_tabara# | Data\_început | Data\_final | locație |
| 1 | 05-11-2018 | 09-11-2018 | Constanța |
| 2 | 13-04-2019 | 18-04-2019 | București |
| 3 | 12-08-2019 | 18-08-2019 | Sibiu |
| 4 | 10-02-2020 | 14-02-2020 | Cluj-Napoca |
| 5 | 11-09-2020 | 18-09-2020 | București |

Astfel, un exemplu de relație care se află în FN2 este SPONSORIZEAZĂ.

Se află în FN1 deoarece există identificator unic pentru toate intrările din tabel.

De asemenea, atributul *suma* care nu este în componența cheii primare este dependent de întreaga cheie primară formată din id\_companie și id\_tabără.

Forma normală 3 (FN3)

Pentru a exemplifica această formă normală, am ales realizarea unui exemplu fictiv pentru relația SPECIALIST (unde am presupus ca atributul salariu depinde de ani\_experiență, dar această regulă nu se aplică în modelul creat).

Fie date aceste date în SPECIALIST:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| id\_specialist# | nume | prenume | specializare | ani\_experiență | salariu |
| 10 | Pop | Matei | matematică | 12 | 3000 |
| 20 | Popa | Ion | biologie | 8 | 2500 |
| 30 | Munteanu | Diana | informatică | 15 | 3300 |
| 40 | Anton | Maria | engleză | 8 | 2500 |
| 50 | Bejan | Andrei | matematică | 14 | 3250 |

Această relație este atât în FN1 cât și în FN2, deoarece există indentificator unic pentru toate intrările din tabel (id\_specialist#), iar atributele care nu sunt în componența cheii primare sunt dependente de întreaga cheie primară.

Acest exemplu nu se află în FN3, deoarece atributul salariu depinde tranzitiv de cheia primară id\_specialist prin intermediul atributului ani\_experiență.

Relațiile de dependență sunt după cum urmează:

{Id\_specialist} –> {nume, prenume, specializare, ani\_experienta, salariu}

{Id\_specialist} –> {ani\_experienta} -> {salariu}

Transformarea arată în felul următor:

SPECIALIST\_FN3

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| id\_specialist# | nume | prenume | specializare | ani\_experiență |
| 10 | Pop | Matei | matematică | 12 |
| 20 | Popa | Ion | biologie | 8 |
| 30 | Munteanu | Diana | informatică | 15 |
| 40 | Anton | Maria | engleză | 8 |
| 50 | Bejan | Andrei | matematică | 14 |

ANI\_EXPERIENTA\_SALARIU

|  |  |
| --- | --- |
| ani\_experiență | salariu |
| 12 | 3000 |
| 8 | 2500 |
| 15 | 3300 |
| 8 | 2500 |
| 14 | 3250 |

**10+13.**

CREATE TABLE COMPANIE

(id\_companie NUMBER(5),

nume VARCHAR2(30),

an\_infiintare NUMBER(4),

CONSTRAINT primary\_key\_companie PRIMARY KEY (id\_companie));

ALTER TABLE COMPANIE

MODIFY nume NOT NULL;

CREATE TABLE TABARA

(id\_tabara NUMBER(5),

data\_inceput DATE CONSTRAINT data\_inceput\_nn NOT NULL,

data\_final DATE CONSTRAINT data\_final\_nn NOT NULL,

locatie VARCHAR2(20) CONSTRAINT locatie\_nn NOT NULL,

CONSTRAINT primary\_key\_tabara PRIMARY KEY (id\_tabara));

ALTER TABLE TABARA

ADD CONSTRAINT date\_tabara CHECK (data\_inceput < data\_final);

CREATE TABLE SPECIALIST

(id\_specialist NUMBER(5),

nume VARCHAR2(20) NOT NULL,

prenume VARCHAR2(30) NOT NULL,

specializare VARCHAR2(20) NOT NULL,

ani\_experienta NUMBER(2),

salariu NUMBER (4),

CONSTRAINT primary\_key\_specialist PRIMARY KEY (id\_specialist));

CREATE TABLE SESIUNE

(id\_sesiune NUMBER(5),

data\_sesiune DATE NOT NULL,

durata number(1) NOT NULL,

id\_tabara NUMBER(5),

CONSTRAINT primary\_key\_sesiune PRIMARY KEY (id\_sesiune));

ALTER TABLE SESIUNE

ADD CONSTRAINT id\_tabara\_fk FOREIGN KEY(id\_tabara) REFERENCES TABARA(id\_tabara);

CREATE TABLE INVITAT

(id\_invitat NUMBER(5),

nume VARCHAR2(20) NOT NULL,

prenume VARCHAR2(30) NOT NULL,

ocupatie VARCHAR2(20),

id\_sesiune NUMBER(5),

CONSTRAINT primaty\_key\_invitat PRIMARY KEY (id\_invitat));

ALTER TABLE INVITAT

ADD CONSTRAINT id\_sesiune\_fk FOREIGN KEY(id\_sesiune) REFERENCES SESIUNE(id\_sesiune);

CREATE TABLE GRUPA

(id\_grupa NUMBER(5),

numar\_liceeni NUMBER(1) NOT NULL,

CONSTRAINT primary\_key\_grupa PRIMARY KEY(id\_grupa));

CREATE TABLE ATELIER

(id\_atelier NUMBER(5),

nume VARCHAR2(20) NOT NULL,

ore\_total NUMBER(2),

materie VARCHAR2(20),

id\_tabara NUMBER(5),

id\_specialist NUMBER(5),

CONSTRAINT primaty\_key\_atelier PRIMARY KEY (id\_atelier));

ALTER TABLE ATELIER

ADD CONSTRAINT id\_tabara\_fk\_atelier FOREIGN KEY(id\_tabara) REFERENCES TABARA(id\_tabara);

ALTER TABLE ATELIER

ADD CONSTRAINT id\_specialist\_fk\_atelier FOREIGN KEY(id\_specialist) REFERENCES SPECIALIST(id\_specialist);

CREATE TABLE LICEAN

(id\_licean NUMBER(5),

nume VARCHAR2(20) NOT NULL,

prenume VARCHAR2(30) NOT NULL,

clasa NUMBER(2) CONSTRAINT clasa\_nr CHECK(clasa >= 9 and clasa <= 12),

liceu VARCHAR2(30),

id\_tabara NUMBER(5),

id\_grupa NUMBER(5),

id\_specialist NUMBER(5),

CONSTRAINT primary\_key\_licean PRIMARY KEY (id\_licean));

ALTER TABLE LICEAN

ADD CONSTRAINT id\_tabara\_fk\_licean FOREIGN KEY(id\_tabara) REFERENCES TABARA(id\_tabara);

ALTER TABLE LICEAN

ADD CONSTRAINT id\_grupa\_fk\_licean FOREIGN KEY(id\_grupa) REFERENCES GRUPA(id\_grupa);

ALTER TABLE LICEAN

ADD CONSTRAINT id\_specialist\_fk\_licean FOREIGN KEY(id\_specialist) REFERENCES SPECIALIST(id\_specialist);

CREATE TABLE VOLUNTAR

(id\_voluntar NUMBER(5),

nume VARCHAR2(20) NOT NULL,

prenume VARCHAR2(30) NOT NULL,

data\_nasterii DATE,

id\_atelier NUMBER(5),

CONSTRAINT primary\_key\_voluntar PRIMARY KEY (id\_voluntar));

ALTER TABLE VOLUNTAR

ADD CONSTRAINT id\_atelier\_fk\_voluntar FOREIGN KEY(id\_atelier) REFERENCES ATELIER(id\_atelier);

CREATE TABLE PROIECT

(id\_proiect NUMBER (5),

titlu VARCHAR2(20) NOT NULL,

materie VARCHAR2(20),

id\_licean NUMBER(5),

CONSTRAINT primary\_key\_proiect PRIMARY KEY (id\_proiect));

ALTER TABLE PROIECT

ADD CONSTRAINT id\_licean\_fk\_proiect FOREIGN KEY (id\_licean) REFERENCES LICEAN(id\_licean);

CREATE TABLE SPONSORIZEAZA

(id\_companie NUMBER(5),

id\_tabara NUMBER(5),

suma NUMBER (6),

CONSTRAINT primary\_key\_sponsorizeaza PRIMARY KEY (id\_companie, id\_tabara));

ALTER TABLE SPONSORIZEAZA

ADD CONSTRAINT id\_companie\_fk\_sponsor FOREIGN KEY(id\_companie) REFERENCES COMPANIE(id\_companie);

ALTER TABLE SPONSORIZEAZA

ADD CONSTRAINT id\_tabara\_fk\_sponsor FOREIGN KEY(id\_tabara) REFERENCES TABARA(id\_tabara);

ALTER TABLE SPONSORIZEAZA

MODIFY suma NOT NULL;

CREATE TABLE SUPERVIZEAZA

(id\_voluntar NUMBER(5),

id\_licean NUMBER(5),

nr\_de\_ore NUMBER(1) not null,

CONSTRAINT primary\_key\_supervizeaza PRIMARY KEY (id\_voluntar, id\_licean),

CONSTRAINT id\_voluntar\_fk\_super FOREIGN KEY (id\_voluntar) REFERENCES VOLUNTAR(id\_voluntar),

CONSTRAINT id\_licean\_fk\_super FOREIGN KEY(id\_licean) REFERENCES LICEAN(id\_licean));

CREATE TABLE SE\_INSCRIE

(id\_licean NUMBER(5),

id\_atelier NUMBER(5),

data\_inscriere DATE NOT NULL,

CONSTRAINT primary\_key\_inscrie PRIMARY KEY(id\_licean, id\_atelier),

CONSTRAINT id\_licean\_fk\_inscrie FOREIGN KEY (id\_licean) REFERENCES LICEAN(id\_licean),

CONSTRAINT id\_atelier\_fk\_inscrie FOREIGN Key(id\_atelier) REFERENCES ATELIER(id\_atelier));

CREATE SEQUENCE SEQ\_GRUPA3

INCREMENT BY 10

START WITH 10

MAXVALUE 1000

NOCYCLE;

--EXERCITIU 13 CU SECVENTE

INSERT INTO GRUPA

VALUES(SEQ\_GRUPA3.NEXTVAL, 3);

INSERT INTO GRUPA

VALUES(SEQ\_GRUPA3.NEXTVAL, 5);

INSERT INTO GRUPA

VALUES(SEQ\_GRUPA3.NEXTVAL, 5);

INSERT INTO GRUPA

VALUES(SEQ\_GRUPA3.NEXTVAL, 4);

INSERT INTO GRUPA

VALUES(SEQ\_GRUPA3.NEXTVAL, 5);

CREATE SEQUENCE SEQ\_TABARA3

INCREMENT BY 10

START WITH 10

MAXVALUE 10000

NOCYCLE;

INSERT INTO TABARA

VALUES(SEQ\_TABARA3.NEXTVAL, TO\_DATE('05-11-2018', 'DD-MM-YYYY'), TO\_DATE('09-11-2018', 'DD-MM-YYYY'), 'ARAD');

INSERT INTO TABARA

VALUES(SEQ\_TABARA3.NEXTVAL, TO\_DATE('13-04-2019', 'DD-MM-YYYY'), TO\_DATE('18-04-2019', 'DD-MM-YYYY'), 'BUCURESTI');

INSERT INTO TABARA

VALUES(SEQ\_TABARA3.NEXTVAL, TO\_DATE('12-08-2019', 'DD-MM-YYYY'), TO\_DATE('18-08-2019', 'DD-MM-YYYY'), 'SIBIU');

INSERT INTO TABARA

VALUES(SEQ\_TABARA3.NEXTVAL, TO\_DATE('10-02-2020', 'DD-MM-YYYY'), TO\_DATE('14-02-2020', 'DD-MM-YYYY'), 'CLUJ-NAPOCA');

INSERT INTO TABARA

VALUES(SEQ\_TABARA3.NEXTVAL, TO\_DATE('11-09-2020', 'DD-MM-YYYY'), TO\_DATE('18-09-2020', 'DD-MM-YYYY'), 'BUCURESTI');

INSERT INTO TABARA

VALUES(SEQ\_TABARA3.NEXTVAL, TO\_DATE('04-03-2018', 'DD-MM-YYYY'), TO\_DATE('09-03-2018', 'DD-MM-YYYY'), 'TIMISOARA');

INSERT INTO TABARA

VALUES(SEQ\_TABARA3.NEXTVAL, TO\_DATE('05-08-2019', 'DD-MM-YYYY'), TO\_DATE('09-08-2019', 'DD-MM-YYYY'), 'IASI');

INSERT INTO TABARA

VALUES(SEQ\_TABARA3.NEXTVAL, TO\_DATE('05-11-2017', 'DD-MM-YYYY'), TO\_DATE('09-11-2017', 'DD-MM-YYYY'), 'GALATI');

INSERT INTO TABARA

VALUES(SEQ\_TABARA3.NEXTVAL, TO\_DATE('02-06-2017', 'DD-MM-YYYY'), TO\_DATE('08-06-2017', 'DD-MM-YYYY'), 'IASI');

SELECT \* FROM TABARA;

--REVENIRE EXERCITIU 10

INSERT INTO COMPANIE

VALUES(10, 'AQUACARPATICA.SRL', 2002);

INSERT INTO COMPANIE

VALUES(20, 'SUPERBET.SRL', 2004);

INSERT INTO COMPANIE

VALUES(30, 'EUROSTAR STUDIOS.SRL', 2002);

INSERT INTO COMPANIE

VALUES(40, 'KAUFLAND.SRL', 1999);

INSERT INTO COMPANIE

VALUES(50, 'FLOWERS DESIGN.SRL', 2009);

INSERT INTO SPECIALIST

VALUES(10, 'Pop', 'Alexandra', 'biologie moleculara', 12, 1800);

INSERT INTO SPECIALIST

VALUES(20, 'Pop', 'Matei', 'matematica', 12, 3000);

INSERT INTO SPECIALIST

VALUES(30, 'Popa', 'Ion', 'biologie', 8, 2500);

INSERT INTO SPECIALIST

VALUES(40, 'Munteanu', 'Diana', 'informatica', 15, 3300);

INSERT INTO SPECIALIST

VALUES(50, 'Anton', 'Maria', 'engleza', 8, 2000);

INSERT INTO SPECIALIST

VALUES(60, 'Bejan', 'Andrei', 'matematica', 14, 3250);

INSERT INTO ATELIER

VALUES (10, 'Bazele biologiei', 12, 'biologie', 30, 30);

INSERT INTO ATELIER

VALUES (20, 'Literatura engleza', 12, 'engleza', 20, 50);

INSERT INTO ATELIER

VALUES (30, 'Matematica avansata', 12, 'matematica', 30, 60);

INSERT INTO ATELIER

VALUES (40, 'Algoritmi de baza', 12, 'informatica', 40, 40);

INSERT INTO ATELIER

VALUES (50, 'Bazele biologiei II', 10, 'biologie', 30, 10);

INSERT INTO ATELIER

VALUES (60, 'Biologie moleculara', 12, 'biologie moleculara', 50, 10);

INSERT INTO ATELIER

VALUES (70, 'Structuri de date', 14, 'informatica', 60, 40);

INSERT INTO ATELIER

VALUES (80, 'Engleza avansata', 9, 'engleza', 70, 50);

INSERT INTO ATELIER

VALUES (90, 'Fizica aplicata', 12, 'fizica', 80, 60);

INSERT INTO ATELIER

VALUES (100, 'Fizica si mate', 14, 'fizica', 90, 20);

INSERT INTO ATELIER

VALUES (110, 'Informatica', 15, 'informatica', 100, 40);

INSERT INTO VOLUNTAR

VALUES(10, 'Popescu', 'Clara', TO\_DATE('26-11-2002', 'dd-mm-yyyy'), 30);

INSERT INTO VOLUNTAR

VALUES(20, 'Marin', 'Andreea', TO\_DATE('14-09-2000', 'dd-mm-yyyy'), 40);

INSERT INTO VOLUNTAR

VALUES(30, 'Mircea', 'Ana', TO\_DATE('06-04-2001', 'dd-mm-yyyy'), 50);

INSERT INTO VOLUNTAR

VALUES(40, 'Marinescu', 'Larisa', TO\_DATE('24-01-2002', 'dd-mm-yyyy'), 60);

INSERT INTO VOLUNTAR

VALUES(50, 'Popescu', 'Clara', TO\_DATE('26-11-2002', 'dd-mm-yyyy'), 10);

INSERT INTO VOLUNTAR

VALUES(60, 'Popescu', 'Clara', TO\_DATE('26-11-2002', 'dd-mm-yyyy'), 20);

UPDATE VOLUNTAR

SET NUME = 'Ciurea' , PRENUME = 'Bianca'

WHERE id\_voluntar = 50;

UPDATE VOLUNTAR

SET NUME = 'Bogdan' , PRENUME = 'Ioana'

WHERE id\_voluntar = 60;

UPDATE VOLUNTAR

SET data\_nasterii = TO\_DATE('02-08-2001', 'dd-mm-yyyy')

WHERE id\_voluntar = 50;

UPDATE VOLUNTAR

SET data\_nasterii = TO\_DATE('03-09-2001', 'dd-mm-yyyy')

WHERE id\_voluntar = 60;

INSERT INTO VOLUNTAR

VALUES(70, 'Baciu', 'Otilia', TO\_DATE('22-04-2000', 'dd-mm-yyyy'),30);

INSERT INTO VOLUNTAR

VALUES(80, 'Vladici', 'Sara', TO\_DATE('02-04-2000', 'dd-mm-yyyy'),10);

INSERT INTO VOLUNTAR

VALUES(90, 'Bob', 'Alice', TO\_DATE('21-06-2000', 'dd-mm-yyyy'), 30);

INSERT INTO VOLUNTAR

VALUES(100, 'Bran', 'Alexandru', TO\_DATE('12-09-2001', 'dd-mm-yyyy'), 20);

INSERT INTO VOLUNTAR

VALUES(110, 'Costea', 'Mihail', TO\_DATE('20-12-2000', 'dd-mm-yyyy'), 60);

INSERT INTO VOLUNTAR

VALUES(120, 'Corolescu', 'Catalin', TO\_DATE('29-01-2002', 'dd-mm-yyyy'), 60);

INSERT INTO VOLUNTAR

VALUES(130, 'Mihai', 'Violeta', TO\_DATE('28-08-2000', 'dd-mm-yyyy'), 50);

INSERT INTO VOLUNTAR

VALUES(140, 'Manea', 'Tudor', TO\_DATE('17-05-2000', 'dd-mm-yyyy'), 30);

INSERT INTO VOLUNTAR

VALUES(150, 'Sorescu', 'Teodor', TO\_DATE('13-10-2001', 'dd-mm-yyyy'), 20);

INSERT INTO SESIUNE

VALUES(10, TO\_DATE('06-11-2018', 'DD-MM-YYYY'), 3, 20);

INSERT INTO SESIUNE

VALUES(20, TO\_DATE('15-04-2019', 'DD-MM-YYYY'), 2, 30);

INSERT INTO SESIUNE

VALUES(30, TO\_DATE('13-08-2019', 'DD-MM-YYYY'), 3, 40);

INSERT INTO SESIUNE

VALUES(40, TO\_DATE('13-02-2020', 'DD-MM-YYYY'), 3, 50);

INSERT INTO SESIUNE

VALUES(50, TO\_DATE('17-09-2020', 'DD-MM-YYYY'), 3, 60);

INSERT INTO SESIUNE

VALUES(60, TO\_DATE('04-03-2018', 'DD-MM-YYYY'), 3, 70);

INSERT INTO SESIUNE

VALUES(70, TO\_DATE('07-08-2019', 'DD-MM-YYYY'), 3, 80);

INSERT INTO SESIUNE

VALUES(80, TO\_DATE('06-11-2017', 'DD-MM-YYYY'), 3, 90);

INSERT INTO SESIUNE

VALUES(90, TO\_DATE('06-06-2017', 'DD-MM-YYYY'), 3, 100);

INSERT INTO INVITAT

VALUES(10, 'Anghel', 'Sorana', 'antreprenor', 10);

INSERT INTO INVITAT

VALUES(20, 'Angelescu', 'Bogdan', 'atlet', 20);

INSERT INTO INVITAT

VALUES(30, 'Milu', 'Alexandru', 'atlet', 30);

INSERT INTO INVITAT

VALUES(40, 'Barbu', 'Simona', 'psiholog', 40);

INSERT INTO INVITAT

VALUES(50, 'Antonescu', 'Alina', 'politician', 50);

INSERT INTO INVITAT

VALUES(60, 'Mircescu', 'Vlad', 'antreprenor', 60);

INSERT INTO INVITAT

VALUES(70, 'Manea', 'Bogdan', 'antreprenor', 70);

INSERT INTO INVITAT

VALUES(80, 'Begu', 'Briana', 'psiholog', 80);

INSERT INTO INVITAT

VALUES(90, 'Carstea', 'Marian', 'antreprenor', 90);

INSERT INTO INVITAT

VALUES(100, 'Deaconu', 'Razvan', 'filantrop', 70);

INSERT INTO LICEAN

VALUES(10, 'Popa', 'Alexandra', 10, 'Colegiul National Unirea', 20, 20, 10);

INSERT INTO LICEAN

VALUES(20, 'Minea', 'Alin', 9, 'CN Mihai Eminescu', 30, 30, 20);

INSERT INTO LICEAN

VALUES(30, 'Pelea', 'Mihail', 12, 'CN Ion Luca Caragiale', 40, 20, 10);

INSERT INTO LICEAN

VALUES(40, 'Manea', 'Sandra', 10, 'CN A.I. Cuza', 50, 30, 40);

INSERT INTO LICEAN

VALUES(50, 'Banciu', 'Maria', 11, 'CN Tehnologic Buzau', 60, 40, 50);

INSERT INTO LICEAN

VALUES(60, 'Vasiliu', 'Catalin', 10, 'CN Pedagogic Zalau', 70, 40, 50);

INSERT INTO LICEAN

VALUES(70, 'Neacsu', 'Teodora', 12, 'Liceul Teoretic Nr.1 Vaslui', 80, 20, 60);

INSERT INTO LICEAN

VALUES(80, 'Velea', 'Miruna', 9, 'CN Marin Preda', 90, 30, 10);

INSERT INTO LICEAN

VALUES(90, 'Popescu', 'Lavinia', 10, 'Colegiul National Unirea', 100, 20, 60);

INSERT INTO LICEAN

VALUES(100, 'Nicolae', 'Daria', 9, 'CN Constantin Noica', 20, 20, 20);

INSERT INTO LICEAN

VALUES(110, 'Nania', 'Alexandra', 10, 'CN Tehnologic Nr.2 Braila', 20, 40, 10);

INSERT INTO LICEAN

VALUES(120, 'Popa', 'Mircea', 11, 'Liceul Teoretic Nr.4 Bucuresti', 20, 50, 10);

INSERT INTO LICEAN

VALUES(130, 'Barbu', 'Laurentiu', 12, 'CN Pedagogic Alexandria', 20, 50, 30);

INSERT INTO LICEAN

VALUES(140, 'Chircu', 'Alex', 12, 'CN Stefan Cel Mare', 20, 50, 30);

INSERT INTO LICEAN

VALUES(150, 'Minea', 'Valeria', 12, 'CN Marin Preda', 80, 40, 30);

INSERT INTO LICEAN

VALUES(160, 'Badea', 'Marina', 11, 'Liceul Teoretic Nr.2 Braila', 80, 40, 20);

INSERT INTO LICEAN

VALUES(170, 'Badescu', 'Gina', 9, 'CN Vasile Alecsandri', 100, 50, 50);

INSERT INTO LICEAN

VALUES(180, 'Horea', 'Iulia', 10, 'CN Stefan Cel Mare', 90, 20, 40);

INSERT INTO LICEAN

VALUES(190, 'Bordei', 'Isabela', 10, 'CN Al. I. Cuza', 60, 50, 50);

INSERT INTO LICEAN

VALUES(200, 'Patrascu', 'Andra', 12, 'Liceul Tehnologic Nr.2 Iasi', 20, 50, 20);

INSERT INTO PROIECT

VALUES(10, 'Fizica de azi', 'fizica', 100);

INSERT INTO PROIECT

VALUES(20, 'Mate aplicata', 'matematica', 30);

INSERT INTO PROIECT

VALUES(30, 'Biologie aprofundata', 'biologie moleculara', 40);

INSERT INTO PROIECT

VALUES(40, 'Probleme informatica', 'informatica', 50);

INSERT INTO PROIECT

VALUES(50, 'Literatura engleza', 'engleza', 60);

INSERT INTO PROIECT

VALUES(60, 'Aplicatii', 'matematica', 10);

INSERT INTO PROIECT

VALUES(70, 'Fizica moderna', 'fizica', 20);

INSERT INTO PROIECT

VALUES(80, 'Sub microscop', 'biologie moleculara', 70);

INSERT INTO PROIECT

VALUES(90, 'Arta cuvintelor', 'engleza', 80);

INSERT INTO PROIECT

VALUES(100, 'Fizica altfel', 'fizica', 110);

INSERT INTO PROIECT

VALUES(110, 'Studiul molecular', 'biologie moleculara', 90);

INSERT INTO PROIECT

VALUES(120,'Probleme reale', 'matematica', 120);

INSERT INTO PROIECT

VALUES(130, 'O noua abordare', 'informatica', 130);

INSERT INTO PROIECT

VALUES(140, 'Aplicatii utile', 'informatica', 140);

INSERT INTO PROIECT

VALUES(150, 'Teoria informatiei', 'informatica', 150);

INSERT INTO PROIECT

VALUES(160, 'English Tenses', 'engleza', 160);

INSERT INTO PROIECT

VALUES(170, 'US History', 'engleza', 170);

INSERT INTO PROIECT

VALUES(180, 'Multimea nr reale', 'matematica', 180);

INSERT INTO PROIECT

VALUES(190, 'Regnul animal', 'biologie', 190);

INSERT INTO PROIECT

VALUES(200, 'Legile lui Newton', 'fizica', 200);

INSERT INTO SPONSORIZEAZA

VALUES(20,20, 15000);

INSERT INTO SPONSORIZEAZA

VALUES(10,20, 12000);

INSERT INTO SPONSORIZEAZA

VALUES(20,30, 15000);

INSERT INTO SPONSORIZEAZA

VALUES(20,70, 20000);

INSERT INTO SPONSORIZEAZA

VALUES(30,20, 5000);

INSERT INTO SPONSORIZEAZA

VALUES(40,40, 15000);

INSERT INTO SPONSORIZEAZA

VALUES(50,50, 70000);

INSERT INTO SPONSORIZEAZA

VALUES(20,60, 15000);

INSERT INTO SPONSORIZEAZA

VALUES(10,70, 65000);

INSERT INTO SPONSORIZEAZA

VALUES(30,80, 24500);

INSERT INTO SPONSORIZEAZA

VALUES(20,90, 15000);

INSERT INTO SPONSORIZEAZA

VALUES(30,100, 100000);

INSERT INTO SPONSORIZEAZA

VALUES(40,100, 1800);

INSERT INTO SPONSORIZEAZA

VALUES(10,90, 29000);

INSERT INTO SPONSORIZEAZA

VALUES(40,80, 17000);

SELECT \* FROM SPONSORIZEAZA;

INSERT INTO SUPERVIZEAZA

VALUES(10, 20, 8);

INSERT INTO SUPERVIZEAZA

VALUES(20,10, 3);

INSERT INTO SUPERVIZEAZA

VALUES(30,10, 2);

INSERT INTO SUPERVIZEAZA

VALUES(40,30, 5);

INSERT INTO SUPERVIZEAZA

VALUES(50,40, 5);

INSERT INTO SUPERVIZEAZA

VALUES(60,10, 1);

INSERT INTO SUPERVIZEAZA

VALUES(70,50, 4);

INSERT INTO SUPERVIZEAZA

VALUES(70,60, 4);

INSERT INTO SUPERVIZEAZA

VALUES(40,20, 3);

INSERT INTO SUPERVIZEAZA

VALUES(30,30, 5);

INSERT INTO SUPERVIZEAZA

VALUES(60,90, 3);

INSERT INTO SUPERVIZEAZA

VALUES(20,100, 3);

INSERT INTO SUPERVIZEAZA

VALUES(70,120, 5);

INSERT INTO SUPERVIZEAZA

VALUES(10,110, 6);

INSERT INTO SUPERVIZEAZA

VALUES(20,130, 3);

INSERT INTO SUPERVIZEAZA

VALUES(40,140, 7);

INSERT INTO SUPERVIZEAZA

VALUES(50,70, 3);

INSERT INTO SUPERVIZEAZA

VALUES(20,80, 3);

INSERT INTO SUPERVIZEAZA

VALUES(70,130, 5);

INSERT INTO SUPERVIZEAZA

VALUES(20,140, 3);

INSERT INTO SUPERVIZEAZA

VALUES(20,150, 2);

INSERT INTO SUPERVIZEAZA

VALUES(40,160, 3);

INSERT INTO SUPERVIZEAZA

VALUES(10,170, 2);

INSERT INTO SUPERVIZEAZA

VALUES(50,180, 4);

INSERT INTO SUPERVIZEAZA

VALUES(10,190, 3);

INSERT INTO SUPERVIZEAZA

VALUES(20,200, 1);

INSERT INTO SUPERVIZEAZA

VALUES(20,20, 1);

INSERT INTO SUPERVIZEAZA

VALUES(40,10, 4);

INSERT INTO SUPERVIZEAZA

VALUES(20,30, 1);

INSERT INTO SE\_INSCRIE

VALUES(10, 10, TO\_DATE('12-03-2018', 'DD-MM-YYYY'));

INSERT INTO SE\_INSCRIE

VALUES(20, 10, TO\_DATE('13-03-2018', 'DD-MM-YYYY'));

INSERT INTO SE\_INSCRIE

VALUES(30, 20, TO\_DATE('12-09-2018', 'DD-MM-YYYY'));

INSERT INTO SE\_INSCRIE

VALUES(40, 30, TO\_DATE('02-03-2019', 'DD-MM-YYYY'));

INSERT INTO SE\_INSCRIE

VALUES(50, 40, TO\_DATE('12-11-2019', 'DD-MM-YYYY'));

INSERT INTO SE\_INSCRIE

VALUES(60, 50, TO\_DATE('12-03-2020', 'DD-MM-YYYY'));

INSERT INTO SE\_INSCRIE

VALUES(70, 60, TO\_DATE('04-06-2020', 'DD-MM-YYYY'));

INSERT INTO SE\_INSCRIE

VALUES(80, 10, TO\_DATE('12-03-2018', 'DD-MM-YYYY'));

INSERT INTO SE\_INSCRIE

VALUES(90, 30, TO\_DATE('01-04-2019', 'DD-MM-YYYY'));

INSERT INTO SE\_INSCRIE

VALUES(100, 40, TO\_DATE('30-11-2019', 'DD-MM-YYYY'));

INSERT INTO SE\_INSCRIE

VALUES(110, 50, TO\_DATE('11-06-2020', 'DD-MM-YYYY'));

INSERT INTO SE\_INSCRIE

VALUES(120, 60, TO\_DATE('17-06-2020', 'DD-MM-YYYY'));

INSERT INTO SE\_INSCRIE

VALUES(130, 20, TO\_DATE('09-09-2018', 'DD-MM-YYYY'));

INSERT INTO SE\_INSCRIE

VALUES(140, 20, TO\_DATE('17-09-2018', 'DD-MM-YYYY'));

INSERT INTO SE\_INSCRIE

VALUES(140, 10, TO\_DATE('10-03-2018', 'DD-MM-YYYY'));

INSERT INTO SE\_INSCRIE

VALUES(110, 40, TO\_DATE('12-11-2019', 'DD-MM-YYYY'));

INSERT INTO SE\_INSCRIE

VALUES(90, 50, TO\_DATE('12-06-2020', 'DD-MM-YYYY'));

INSERT INTO SE\_INSCRIE

VALUES(150, 50, TO\_DATE('12-06-2020', 'DD-MM-YYYY'));

INSERT INTO SE\_INSCRIE

VALUES(160, 10, TO\_DATE('11-03-2018', 'DD-MM-YYYY'));

INSERT INTO SE\_INSCRIE

VALUES(170, 40, TO\_DATE('12-11-2019', 'DD-MM-YYYY'));

INSERT INTO SE\_INSCRIE

VALUES(180, 20, TO\_DATE('16-09-2018', 'DD-MM-YYYY'));

INSERT INTO SE\_INSCRIE

VALUES(190, 60, TO\_DATE('12-06-2020', 'DD-MM-YYYY'));

INSERT INTO SE\_INSCRIE

VALUES(200, 60, TO\_DATE('10-06-2020', 'DD-MM-YYYY'));

INSERT INTO SE\_INSCRIE

VALUES(10, 20, TO\_DATE('10-09-2018', 'DD-MM-YYYY'));

INSERT INTO SE\_INSCRIE

VALUES(20, 20, TO\_DATE('11-09-2018', 'DD-MM-YYYY'));

SELECT \* FROM se\_inscrie;

ALTER TABLE LICEAN

ADD Cheltuieli\_transport NUMBER(3);

UPDATE LICEAN

SET CHELTUIELI\_TRANSPORT = 50

WHERE (ID\_LICEAN = 10 OR ID\_LICEAN = 20);

UPDATE LICEAN

SET CHELTUIELI\_TRANSPORT = 97

WHERE (ID\_LICEAN = 40 OR ID\_LICEAN = 120);

UPDATE LICEAN

SET CHELTUIELI\_TRANSPORT = 70

WHERE (ID\_LICEAN = 60 OR ID\_LICEAN = 70 OR ID\_LICEAN = 150);

UPDATE LICEAN

SET CHELTUIELI\_TRANSPORT = 110

WHERE (ID\_LICEAN = 200);

UPDATE LICEAN

SET CHELTUIELI\_TRANSPORT = 120

WHERE (ID\_LICEAN = 100);

UPDATE LICEAN

SET CHELTUIELI\_TRANSPORT = 100

WHERE (ID\_LICEAN = 50);

UPDATE LICEAN

SET CHELTUIELI\_TRANSPORT = 125

WHERE (ID\_LICEAN = 30);

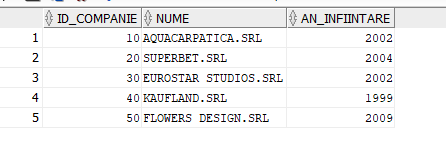
UPDATE LICEAN

SET CHELTUIELI\_TRANSPORT = 30

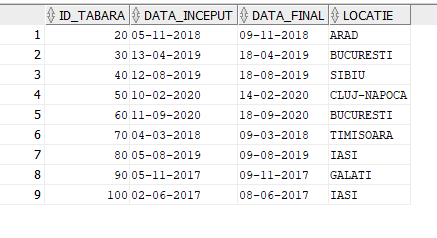
WHERE (ID\_LICEAN = 190);

commit;

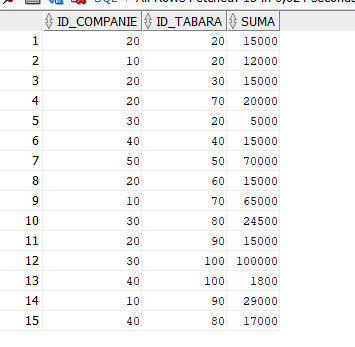
select \* from companie:



Select \* from tabara:



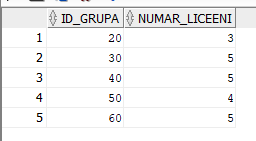
Select \* from sponsorizeaza:



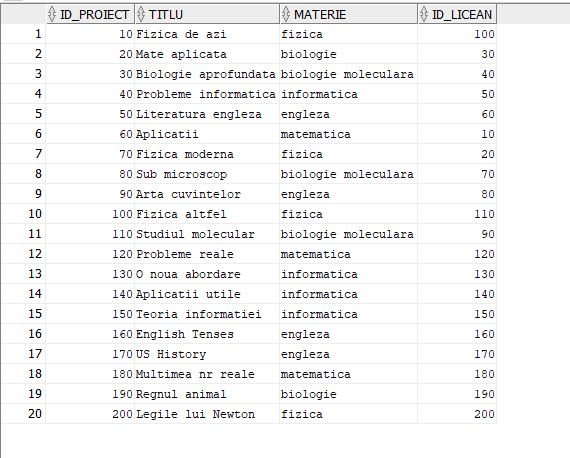
select \* from licean:



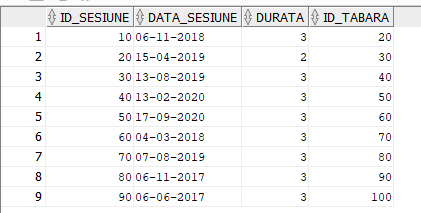
Select \* from grupa:



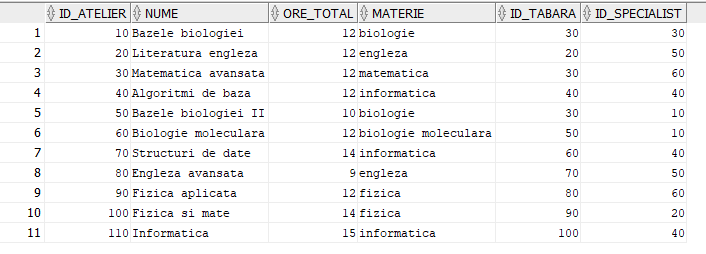
Select \* from proiect:



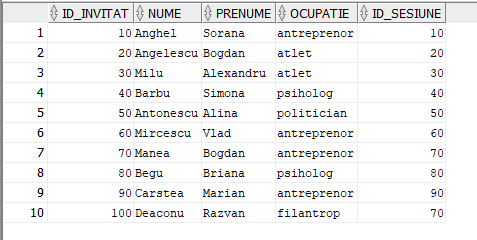
Select \* from sesiune:



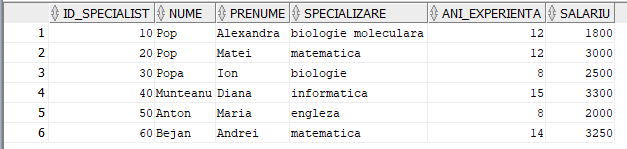
Select \* from atelier;



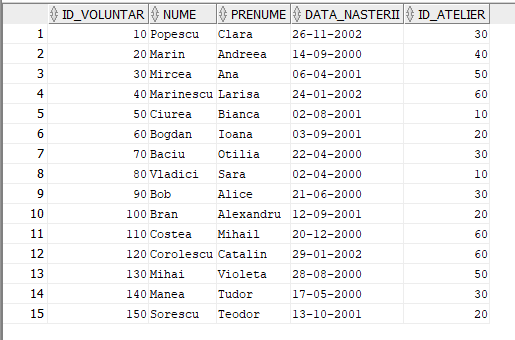
Select \* from invitat;



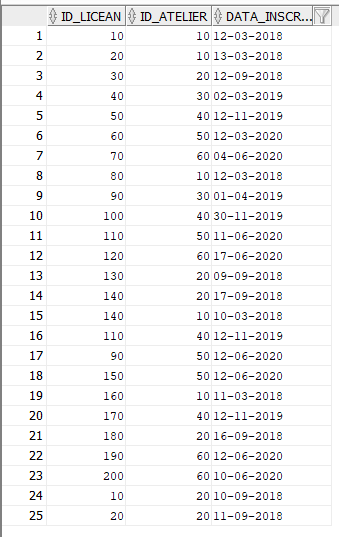
Select \* from specialist;



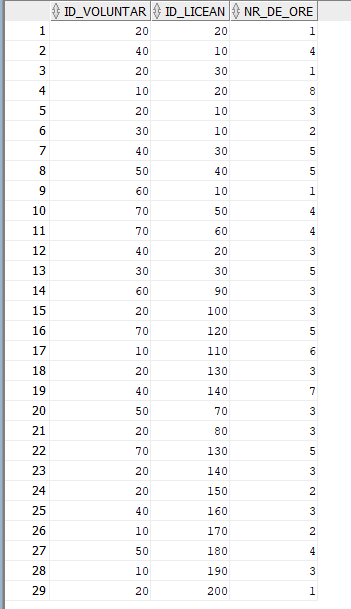
Select \* from voluntar;



Select \* from se\_inscrie:



Select \* from supervizeaza;



**13.** Următoarele linii de cod se regăsesc în exercițiul 10, pentru realizarea inserărilor:

CREATE SEQUENCE SEQ\_GRUPA3

INCREMENT BY 10

START WITH 10

MAXVALUE 1000

NOCYCLE;

--EXERCITIU 13 CU SECVENTE

INSERT INTO GRUPA

VALUES(SEQ\_GRUPA3.NEXTVAL, 3);

INSERT INTO GRUPA

VALUES(SEQ\_GRUPA3.NEXTVAL, 5);

INSERT INTO GRUPA

VALUES(SEQ\_GRUPA3.NEXTVAL, 5);

INSERT INTO GRUPA

VALUES(SEQ\_GRUPA3.NEXTVAL, 4);

INSERT INTO GRUPA

VALUES(SEQ\_GRUPA3.NEXTVAL, 5);

CREATE SEQUENCE SEQ\_TABARA3

INCREMENT BY 10

START WITH 10

MAXVALUE 10000

NOCYCLE;

INSERT INTO TABARA

VALUES(SEQ\_TABARA3.NEXTVAL, TO\_DATE('05-11-2018', 'DD-MM-YYYY'), TO\_DATE('09-11-2018', 'DD-MM-YYYY'), 'ARAD');

INSERT INTO TABARA

VALUES(SEQ\_TABARA3.NEXTVAL, TO\_DATE('13-04-2019', 'DD-MM-YYYY'), TO\_DATE('18-04-2019', 'DD-MM-YYYY'), 'BUCURESTI');

INSERT INTO TABARA

VALUES(SEQ\_TABARA3.NEXTVAL, TO\_DATE('12-08-2019', 'DD-MM-YYYY'), TO\_DATE('18-08-2019', 'DD-MM-YYYY'), 'SIBIU');

INSERT INTO TABARA

VALUES(SEQ\_TABARA3.NEXTVAL, TO\_DATE('10-02-2020', 'DD-MM-YYYY'), TO\_DATE('14-02-2020', 'DD-MM-YYYY'), 'CLUJ-NAPOCA');

INSERT INTO TABARA

VALUES(SEQ\_TABARA3.NEXTVAL, TO\_DATE('11-09-2020', 'DD-MM-YYYY'), TO\_DATE('18-09-2020', 'DD-MM-YYYY'), 'BUCURESTI');

INSERT INTO TABARA

VALUES(SEQ\_TABARA3.NEXTVAL, TO\_DATE('04-03-2018', 'DD-MM-YYYY'), TO\_DATE('09-03-2018', 'DD-MM-YYYY'), 'TIMISOARA');

INSERT INTO TABARA

VALUES(SEQ\_TABARA3.NEXTVAL, TO\_DATE('05-08-2019', 'DD-MM-YYYY'), TO\_DATE('09-08-2019', 'DD-MM-YYYY'), 'IASI');

INSERT INTO TABARA

VALUES(SEQ\_TABARA3.NEXTVAL, TO\_DATE('05-11-2017', 'DD-MM-YYYY'), TO\_DATE('09-11-2017', 'DD-MM-YYYY'), 'GALATI');

INSERT INTO TABARA

VALUES(SEQ\_TABARA3.NEXTVAL, TO\_DATE('02-06-2017', 'DD-MM-YYYY'), TO\_DATE('08-06-2017', 'DD-MM-YYYY'), 'IASI');

**11.**

11.1

--Să se afiseze id-ul, numele, prenumele, cheltuielile de transport asigurate, numele si prenumele

--specialistului care îi îndrumă pentru proiect, titlul proiectului realizat precum si locatia taberei

--la care au participat pentru liceeni

--din ciclul superior(clasele 11 si 12) care sunt îndrumasi de un specialist al cărui nume

--începe cu litera P sau are litera D în componenta prenumelui. De asemenea, să se afiseze datele

--cerute doar pentru liceeni care au participat la tabere din ultimii 5 ani fată de data curentă.

--Coloanele se vor numi corespunzător, numele si prenumele se vor concatena atât pt licean cât si pentru

--specialist, iar datele vor fi afi?ate după numele liceanului.

select lic.id\_licean as "Id\_licean", concat(lic.nume, ' ')||lic.prenume as "Nume Licean",

nvl(cheltuieli\_transport, 0) "Cheltuieli acoperite", concat(spec.nume, ' ')||spec.prenume as "Nume Specialist",

pr.titlu as "Titlu Proiect", tab.locatie as "Locatie Tabara"

from licean lic join specialist spec on (lic.id\_specialist = spec.id\_specialist)

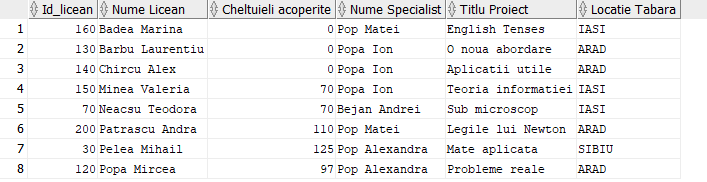
join proiect pr on (lic.id\_licean = pr.id\_licean)

join tabara tab on (lic.id\_tabara = tab.id\_tabara)

where(lic.clasa >= 11) and (spec.nume like 'P%' or UPPER(spec.prenume) like '%D%') and

months\_between(sysdate, tab.data\_final) < 60

order by "Nume Licean";



--Am folosit operatia JOIN pe 4 tabele, filtrare la nivel de linii, ordonare după nume,

--3 functii pentru siruri de caractere (UPPER, CONCAT, LIKE)

--2 functii pentru date calendaristice(sysdate si months\_between)

--am folosit NVL

11.2

--Pentru taberele în care transportul maxim asigurat este mai mare sau egal cu 70

--si transportul minim mai mic sau egal cu 50

--să se afiseze fiecare companie care a facut o sponsorizare, suma oferită, dar si suma minimă si maximă oferită drept decont pentru transport.

--Nu se tine cont de valorile null.

select id\_companie, comp.nume, id\_tabara, locatie, suma, MAX(cheltuieli\_transport), min(cheltuieli\_transport)

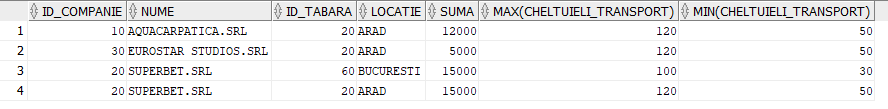
from licean join tabara using (id\_tabara)

join sponsorizeaza using(id\_tabara)

join companie comp using(id\_companie)

group by id\_companie, id\_tabara, suma, locatie, comp.nume

having MAX(cheltuieli\_transport) >= 70 and min(cheltuieli\_transport) <= 50;



--Am folosit functii grup(max, min), filtrare la nivel de grupuri si grupări de date.

11.3

--Pentru fiecare tabara, sa se afiseze suma totala a sponsorizarilor primite de la companii, precum si id-ul taberei.

--Să se adauge o nouă coloană numită Balant Sponsorizari care să contină, dupa caz, mesajele:

--Dacă totalul de sponsorizari este cel putin 70.000: 'AU FOST ATRASE SPONSORIZARI MARI'

--Dacă totalul este sub 70.000, dar peste 30.000: 'SPONSORIZARILE AU FOST MEDII'

--Altfel: 'SPONSORIZARILE SUNT IN VALOARE MICA'

with sponsorizari as(select id\_tabara, sum(suma) TOTAL

from sponsorizeaza

group by id\_tabara

order by TOTAL desc)

select id\_tabara, TOTAL,

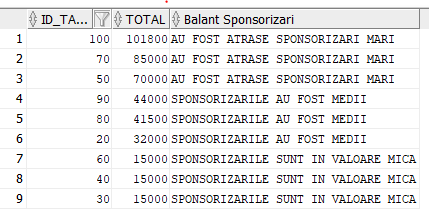
case when TOTAL >= 70000 then 'AU FOST ATRASE SPONSORIZARI MARI'

when TOTAL < 70000 and TOTAL >= 30000 then 'SPONSORIZARILE AU FOST MEDII'

else 'SPONSORIZARILE SUNT IN VALOARE MICA'

END "Balant Sponsorizari"

from sponsorizari;



-- am folosit WITH, group by, ordonare descrescatoare, CASE

11.4

--Sa se afiseze id-ul, numele și prenumele voluntarilor nascuti in anul 2000

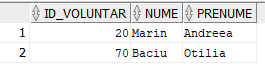
--care supervizează elevi al caror nume incepe cu aceeasi litera ca numele lor

select v.id\_voluntar, v.nume, v.prenume

from voluntar v

where v.id\_voluntar in (select id\_voluntar from supervizeaza where id\_licean in

(select id\_licean from licean lic where substr(v.nume,1,1) = substr(lic.nume,1,1))) and decode(to\_char(v.data\_nasterii, 'yyyy'), 2000, 'adev') = 'adev';



--am folosit subcereri, decode, functia substr

11.5

--Sa se afiseze id-ul, numele si prenumele voluntarilor care au organizat atelierele la care

--a participat liceanul cu id-ul 110, in ordine alfabetica.

select id\_voluntar, nume, prenume

from voluntar

where id\_atelier in (

select id\_atelier from atelier where id\_atelier in (

select id\_atelier from se\_inscrie where id\_licean in (

select id\_licean from se\_inscrie where id\_licean = 110)))

order by nume;



--Am folosit subcereri nesinronizate pe 4 tabele si order by

**12.**

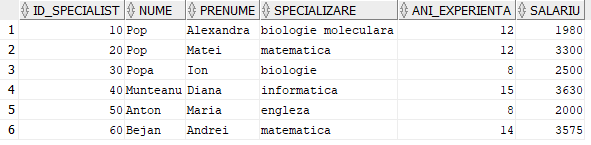
12.1

--Sa se creasca salariul cu 10% specialistilor care au peste 10 ani de experienta.

UPDATE SPECIALIST

SET SALARIU = SALARIU \* 1.1

where(salariu in (select salariu from specialist where ani\_experienta >= 10));



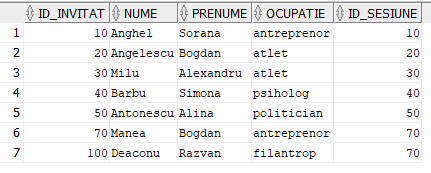
rollback;

12.2

--Sa se stearga din sistem invitatii care au participat la tabere care s-au terminat cu mai mult de 4 ani in urma

delete from invitat

where (id\_sesiune in (select id\_sesiune from sesiune where id\_tabara in (select id\_tabara from tabara where months\_between(sysdate, data\_final) > 48)));



rollback;

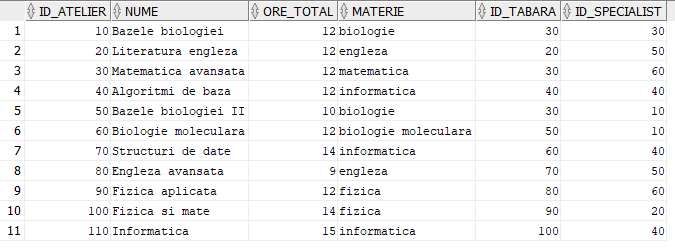
12.3

--Pentru atelierele tinute de specialisti in bioloigie, sa se adauge 2 ore in plus la totalul de ore

UPDATE ATELIER

SET ORE\_TOTAL = ORE\_TOTAL + 2

WHERE ID\_SPECIALIST IN (SELECT ID\_SPECIALIST FROM SPECIALIST WHERE lower(SPECIALIZARE) LIKE '%biologie%');



rollback;

**14.**

Creați o vizualizare care să conțină id-ul liceanului, numele, prenumele, clasa, liceul, precum și titlul proiectului pe care îl realizează și materia pe care se bazează.

CREATE OR REPLACE VIEW Licean\_si\_Proiect

as select lic.id\_licean, lic.nume, lic.prenume, lic.clasa, lic.liceu, pr.titlu "Titlu Proiect", pr.materie from licean lic join proiect pr

on lic.id\_licean = pr.id\_licean;

Operația de UPDATE următoare îmi permite să modific câmpurile dintr-un singur tabel (proiect).

--operatie MDL permisa:

UPDATE Licean\_si\_Proiect

SET materie = 'biologie' where nume = 'Pelea';

rollback;

--operatie MDL nepermisă:

UPDATE Licean\_si\_Proiect

SET liceu = 'Liceul Teoretic Nr.2 Vaslui', materie = 'fizica'

where id\_licean = 20;

rollback;

Acest UPDATE nu este permis deoarece modifică câmpuri din două tabele, ceea ce nu este permis.

**15.**

create unique index Cautare\_Nume\_Clasa on Licean(nume, clasa);

Cererea: Să se afișeze liceeni care sunt în ciclul de învățământ superior (cls 11 și 12) și al căror nume începe cu litera P.

EXPLAIN PLAN FOR

select \* from licean

where nume like 'P%' and clasa >= 11;

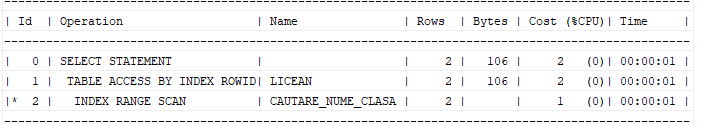
SELECT

PLAN\_TABLE\_OUTPUT

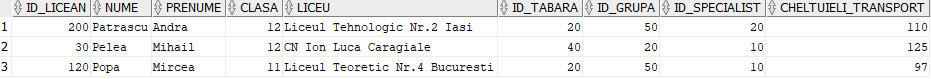
FROM

TABLE(DBMS\_XPLAN.DISPLAY());

Aici vedem că s-a folosit index-ul creat pentru căutare:



Ce s-a afișat:



**16.**

Primul DIVISION:

--Să se afi?eze id-ul, numele, prenumele liceenilor care s-au înscris la acelea?i

--ateliere ca liceanul cu id-ul 140.

select s.id\_licean, nume, prenume

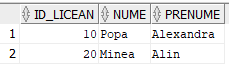
from se\_inscrie s join licean lic on (s.id\_licean = lic.id\_licean)

where id\_atelier in (select id\_atelier from se\_inscrie where id\_licean = 140)

and s.id\_licean != 140

group by s.id\_licean, nume, prenume

having count(\*) = (select count(id\_atelier) from se\_inscrie where id\_licean = 140);



--OUTER JOIN

--Să se afișeze numele și prenumele tuturor liceenilor, precum și id-ul si locatia taberei unde au participat,

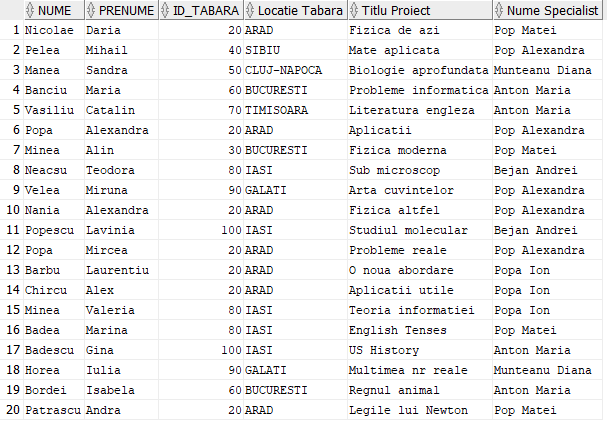
--titlul proiectului pe care l-au realizat, numele și prenumele specialistului care i-a îndrumat.

select lic.nume, lic.prenume,t.id\_tabara, t.locatie "Locatie Tabara", pr.titlu "Titlu Proiect", sp.nume||' ' ||sp.prenume "Nume Specialist" from licean lic

full outer join proiect pr on (lic.id\_licean = pr.id\_licean)

full outer join specialist sp on (lic.id\_specialist = sp.id\_specialist)

full outer join tabara t on (lic.id\_tabara = t.id\_tabara);



Al doilea division:

--Să se afi?eze id-ul, numele, prenumele voluntarilor care supervizează aceia?i liceeni pe care îi supervizează

--?i voluntarul cu id-ul 30 (10,30)

select s.id\_voluntar, nume, prenume

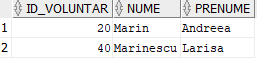
from supervizeaza s join voluntar v on (s.id\_voluntar = v.id\_voluntar)

where id\_licean in (select id\_licean from supervizeaza where id\_voluntar = 30)

and s.id\_voluntar != 30

group by s.id\_voluntar, nume, prenume

having count(\*) = (select count(id\_licean) from supervizeaza where id\_voluntar = 30);



\*Inca un divison care foloseste not exists:

--Sa se afiseze liceeni care au avut proiecte la engleza

select \*

from licean l

where not exists(

(select id\_licean

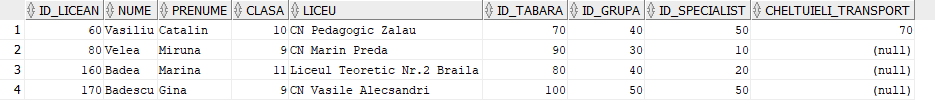
from proiect

where l.id\_licean = id\_licean)

minus(select id\_licean

from proiect

where materie = 'engleza'));



**18.**

O relație se află în forma normală mai sus menționată dacă și numai dacă fiecare determinant este o cheie candidat.

Pentru a exemplifica o normalizare de tipul BCNF se va crea un exemplu fictiv în care se va presupune că atributul *suma* din relația SPONSORIZEAZA este unic și determină functional atributul *id\_companie* care face parte din cheia primară (acest lucru nu se întâmplă totuși în modelul creat).

SPONSORIZEAZA\_AUX

|  |  |  |
| --- | --- | --- |
| Id\_companie# | Id\_tabara# | suma |
| 50 | 10 | 15.000 |
| 10 | 20 | 23.000 |
| 30 | 30 | 100.000 |
| 40 | 40 | 17.000 |
| 20 | 50 | 18.000 |

Astfel, între atribute există dependențele:

{id\_companie#, id\_tabara#} -> {suma} si

{suma} -> {id\_companie};

Se poate observa în acest caz că suma este o cheie candidat (deoarece este unică). De asemenea, cheia id\_companie depinde de atributul suma.

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

1. SPONSORIZEAZA\_AUX\_1 (id\_companie, suma)
2. SPONSORIZEAZA\_AUX\_2 (id\_tabara#, suma)

SPONSORIZEAZA\_AUX\_1

|  |  |
| --- | --- |
| Id\_companie | Suma |
| 50 | 15.000 |
| 10 | 23.000 |
| 30 | 100.000 |
| 40 | 17.000 |
| 20 | 18.000 |

SPONSORIZEAZA\_AUX\_2

|  |  |
| --- | --- |
| Id\_tabara# | Suma |
| 10 | 15.000 |
| 20 | 23.000 |
| 30 | 100.000 |
| 40 | 17.000 |
| 50 | 18.000 |

Forma normală 4

O relație se află în a FN4 dacă și numai dacă se află în forma normală Boyce-Codd și nu conține relații m:n independente.

Pentru a exemplifica această relație se va crea un exemplu imaginar: un licean este îndrumat de mai mulți specialiști și realizează mai multe proiecte. Dacă se dorește introducerea unui nou sportiv va trebui să se introducă și specialistul și proiectul.

|  |  |  |
| --- | --- | --- |
| Id\_licean# | Id\_specialist# | Id\_proiect# |
| 10 | 20 | 10 |
| 10 | 20 | 40 |
| 110 | 30 | 50 |
| 110 | 50 | 50 |
| 20 | 60 | 60 |
| 20 | 60 | 100 |

Astfel, relația este următoarea:

Licean\_nonFN4(id\_licean#, id\_specialist#, id\_proiect#)

Relația este în BCNF.

Se identifică următoarele dependențe multiple:

id\_licean -> -> id\_specialist

id\_licean -> -> id\_proiect

Relațiile rezultate înaplicate în urma FN4:

Licean\_1(id\_licean#, id\_specialist#)

Licean\_2(id\_licean#, id\_specialist#)

Licean\_1

|  |  |
| --- | --- |
| Id\_licean# | Id\_specialist# |
| 10 | 20 |
| 110 | 30 |
| 110 | 50 |
| 20 | 60 |

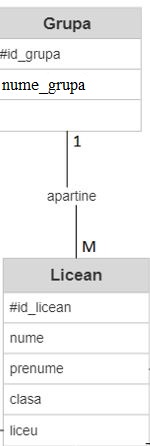
Licean\_2

|  |  |
| --- | --- |
| Id\_licean# | Id\_proiect# |
| 10 | 10 |
| 10 | 40 |
| 110 | 50 |
| 20 | 60 |
| 20 | 100 |

b. Denormalizarea este necesară pentru a reduce numărul de join-uri dintre tabele pentru rezolvarea unei interogări.

Pentru exemplificarea denormalizării am ales entitățile licean și grupa. Relația dintre LICEAN și GRUPA este de M:1.

Cum arătau inițial entitățile înainte de denormalizare:



In urma denormalizării, atributul grupa va fi plasat in tabelul Licean, pentru că în acest fel se elimină o operație de join. În acest mod diagrama este mai eficientă , iar timpul interogărilor a fost îmbunătățit.

Cum arată tabelul dupa denormalizare:

