

Modelarea unei rețele sociale, interogari

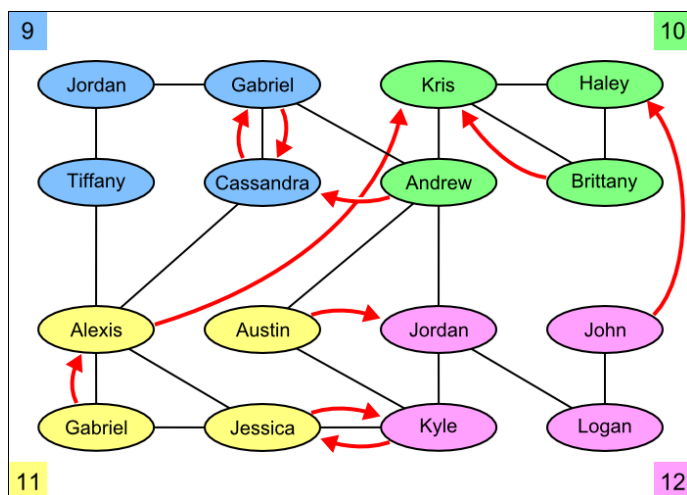
Schema:

```
Elev(ID int, nume VARCHAR2(10), clasa int)
Prieten(ID1 int, ID2 int)
Place(ID1 int, ID2 int)
```

Relația Prieten este mutuală: dacă perechea A,B există, înseamnă nu doar că A este prieten cu B dar și că B este prieten cu A.

Relația Place nu este mutuală: Dacă perechea A,B există înseamnă că A îl place pe B dar reciproca poate să nu fie adevărată.

Setul de date e reprezentat de graful de mai jos. Muchiile negre corespund relației Prieten, arcele roșii corespund relației Place.



Interogări de bază, operatori pe mulțimi

1. Afișați prietenii lui Gabriel.

R: Alexis, Andrew, Cassandra, Jessica, Jordan

2. Pentru fiecare elev care place pe un altul cu 2 sau mai mult de 2 clase mai puțin decât a sa, afișați numele și clasa sa, și numele și clasa elevului pe care îl place.

R:

John	12	Haley	10
------	----	-------	----

3. Pentru fiecare pereche de elevi care se plac reciproc, returnați numele și clasa ambilor. Fiecare pereche să apară o singură dată.

R:

Cassandra	9	Gabriel	9
Jessica	11	Kyle	12

4. Pentru fiecare situație în care elevul A îl place pe B, dar elevul B îl place pe un alt elev C, returnați numele și clasa pentru A,B,C

R:

Andrew	10	Cassandra	9	Gabriel	9
Gabriel	11	Alexis	11	Kris	10

5. Găsiți toți elevii care nu apar în tabela Place (nu place și nu e plăcut) și returnați numele și clasa lor. Sortați după clasă apoi după nume.

R:

Jordan	9
Tiffany	9
Logan	12

Subinterogări

6. Pentru fiecare situație în care A îl place pe B dar B nu place pe nimeni, returnați numele și clasa lui A și B.

R:

John	12	Haley	10
Alexis	11	Kris	10
Brittany	10	Kris	10
Austin	11	Jordan	12

7. Găsiți numele și clasa pentru elevii care au prieteni doar în clasa lor. Returnați rezultatul sortat după clasă apoi după nume.

R:

Jordan	9
Brittany	10
Haley	10
Kris	10
Gabriel	11
John	12
Logan	12

8. Găsiți numele și clasa elevilor care au doar prieteni în alte clase.

R: Austin 11

9. Pentru fiecare elev A care îl place pe B iar cei doi nu sunt prieteni, găsiți dacă au un prieten C în comun. Pentru aceste situații returnați numele și clasa elevilor (A,B,C).

R:

Andrew	10	Cassandra	9	Gabriel	9
Austin	11	Jordan	12	Andrew	10
Austin	11	Jordan	12	Kyle	12

Agregare

10. Găsiți diferența dintre numărul numărul de elevi din școală și numărul de nume diferite.

R: 2

11. Care este numrul mediu de prieteni pe student?

R: 2.5

12. Găsiți numărul de elevi care sunt prieteni cu Cassandra sau sunt prieteni ai prietenilor Cassandrei. Nu o numărați și pe Cassandra chiar dacă tehnic este prietenă cu prietenii Cassandrei.

R: 7

13. Găsiți elevii și clasa tuturor elevilor care sunt plăcuți de mai mult de un elev.

R:

Cassandra	9
Kris	10

14. Găsiți numele și clasa studentului/ilor cu cel mai mare număr de prieteni.

R:

Alexis	11
Andrew	10

Modificarea datelor

Răspunsurile sunt obținute în urma execuției tuturor comenzilor de modificare anterioare.

Reîncărcați scriptul de creare pentru a reveni la starea inițială.

15. Este sfârșitul anului școlar. Incrementați clasa tuturor elevilor cu o unitate. Pentru a verifica modificarea realizată, găsiți numele elevilor care au clasa > 12.

R: Jordan, John, Kyle, Logan

16. Eliminați elevii care au terminat liceul (clasa>12). Pentru a verifica modificarea determinați numărul de elevi care au rămas în tabela Elev.

R: 12

17. Deși relația de prietenie poate continua, ea nu poate fi modelată în tabelele noastre pentru elevi care nu mai sunt. Eliminați toate uplele din Prieten și toate uplele din Place care fac referire la elevi ce nu mai există. Pentru verificare găsiți numele elevilor pe care îi place Austin sau care sunt prieteni cu Austin.

R: Andrew

18. Este vacanță. Drept rezultat cercul social al elevilor se extinde. Pentru toate cazurile în care A este prieten cu B și B este prieten cu C, adăugați relația de prietenie între A și C. Nu duplicați prietenii existente și nu inserați prietenia unui elev cu el însuși. Pentru verificare, returnați numele prietenilor lui Jordan (din clasa a 10-a la momentul current).

R: Tiffany, Gabriel, Alexis, Cassandra, Andrew

Script creare:

```
drop table Elev;
drop table Prieten;
drop table Place;

create table Elev(ID int, nume VARCHAR2(10), clasa int);
create table Prieten(ID1 int, ID2 int);
create table Place(ID1 int, ID2 int);

insert into Elev values (1510, 'Jordan', 9);
insert into Elev values (1689, 'Gabriel', 9);
insert into Elev values (1381, 'Tiffany', 9);
insert into Elev values (1709, 'Cassandra', 9);
insert into Elev values (1101, 'Haley', 10);
insert into Elev values (1782, 'Andrew', 10);
insert into Elev values (1468, 'Kris', 10);
insert into Elev values (1641, 'Brittany', 10);
insert into Elev values (1247, 'Alexis', 11);
insert into Elev values (1316, 'Austin', 11);
insert into Elev values (1911, 'Gabriel', 11);
insert into Elev values (1501, 'Jessica', 11);
insert into Elev values (1304, 'Jordan', 12);
insert into Elev values (1025, 'John', 12);
insert into Elev values (1934, 'Kyle', 12);
insert into Elev values (1661, 'Logan', 12);
```

```
insert into Prieten values (1510, 1381);
```

```
insert into Prieten values (1510, 1689);
insert into Prieten values (1689, 1709);
insert into Prieten values (1381, 1247);
insert into Prieten values (1709, 1247);
insert into Prieten values (1689, 1782);
insert into Prieten values (1782, 1468);
insert into Prieten values (1782, 1316);
insert into Prieten values (1782, 1304);
insert into Prieten values (1468, 1101);
insert into Prieten values (1468, 1641);
insert into Prieten values (1101, 1641);
insert into Prieten values (1247, 1911);
insert into Prieten values (1247, 1501);
insert into Prieten values (1911, 1501);
insert into Prieten values (1501, 1934);
insert into Prieten values (1316, 1934);
insert into Prieten values (1934, 1304);
insert into Prieten values (1304, 1661);
insert into Prieten values (1661, 1025);
insert into Prieten select ID2, ID1 from Prieten;
```

```
insert into Place values(1689, 1709);
insert into Place values(1709, 1689);
insert into Place values(1782, 1709);
insert into Place values(1911, 1247);
insert into Place values(1247, 1468);
insert into Place values(1641, 1468);
insert into Place values(1316, 1304);
insert into Place values(1501, 1934);
insert into Place values(1934, 1501);
insert into Place values(1025, 1101);
```