Aprob

șeful Catedrei MI

\_\_\_\_\_\_\_\_\_ M. Petic

„\_\_\_” \_\_\_\_\_\_ 2022

Proba I pentru evaluarea curentă la unitatea de curs „Practica SGBD”

Fie este dată baza de date cu următoarele tabele:

PERSOANE (idPersoana, Numele, Vârsta)

RUDE (idPersoana1, idPersoana2)

AMICI (idPersoana1, idPersoana2)

1. Să se proiecteze o interogare care ar determina numele și vârsta persoanelor cu cel mai mare număr de rude.
2. Să se proiecteze o interogare care ar returna numele persoanelor care sunt amici. Perechile de prieteni nu trebuie să se repete.
3. Să se proiecteze o interogare care ar determina rudele persoanelor care sunt mai mici ca ei. Să se afișeze perechi de persoane fără repetări.
4. Să se elaboreze o procedură care pentru o persoană ar determina lista membrilor familiei sale. Dacă A este membru al familiei X și B este membru al familiei X, atunci B trebuie să apară în lista membrilor familiei lui A. Ca parametru este dat numele persoanei.
5. Să se elaboreze o procedură care pentru o persoană ar determina numărul de prieteni. Ca parametru este dat numele persoanei.
6. Să se elaboreze o funcție care returnează True sau False dacă două persoane sunt prieteni. Numele persoanelor sunt date ca parametru.
7. Să se creeze o vedere care ar conține lista persoanelor ce au un membru de familie, dar nu au nici un prieten. Este posibil de a adăuga date în tabele prin intermediul vederii? Este posibil de a elimina date din tabele prin intermediul vederii? Este posibil de a modifica date în tabele prin intermediul vederii? Dați exemple de operații de actualizare prin intermediul vederilor, dacă este posibil.
8. Dați câte un exemplu de vedere care ar permite de a adăuga date în tabelul PERSOANE, a doua vedere ar permite modificarea, și a treia ar permite eliminarea datelor din tabelul menționat. Toate trei vederi trebuie să conțină interogări diferite.

Titularul disciplinei Corina Negara

1) Să se proiecteze o interogare care ar determina numele și vârsta persoanelor cu cel mai mare număr de rude.

SELECT persoane.Numele, persoane.Virsta, COUNT(persoane.idPersoana) AS rude  
FROM persoane  
INNER JOIN rude r ON persoane.idPersoana = r.idpersoana1  
OR persoane.idPersoana = r.idpersoana2  
GROUP BY persoane.idPersoana  
HAVING rude = (SELECT COUNT(persoane.idPersoana) AS rude  
FROM persoane  
INNER JOIN rude r ON persoane.idPersoana = r.idpersoana1  
OR persoane.idPersoana = r.idpersoana2  
GROUP BY persoane.idPersoana  
LIMIT 1)

2) Să se proiecteze o interogare care ar returna numele persoanelor care sunt amici. Perechile de prieteni nu trebuie să se repete.

SELECT p1.Numele, p2.Numele  
FROM amici a  
INNER JOIN persoane p1 ON p1.idPersoana = a.idpersoana1  
INNER JOIN persoane p2 ON p2.idPersoana = a.idpersoana2

3) Să se proiecteze o interogare care ar determina rudele persoanelor care sunt mai mici ca ei. Să se afișeze perechi de persoane fără repetări.

SELECT P1.NUMELE, P1.Virsta, P2.NUMELE, P2.Virsta  
FROM PERSOANE AS P1 INNER JOIN RUDE ON P1.IDPERSOANA = RUDE.IDPERSOANA1  
INNER JOIN PERSOANE AS P2 ON P2.IDPERSOANA = RUDE.IDPERSOANA2  
WHERE P1.VIRSTA<>P2.VIRSTA

4)Să se elaboreze o procedură care pentru o persoană ar determina lista membrilor familiei sale. Dacă A este membru al familiei X și B este membru al familiei X, atunci B trebuie să apară în lista membrilor familiei lui A. Ca parametru este dat numele persoanei.

create procedure pr1()  
begin  
select distinct rude.idpersoana1 AS A,  
rude.idpersoana2 AS X,  
if ( r.idpersoana1 <> rude.idpersoana1 AND r.idpersoana1 <> rude.idpersoana2, r.idpersoana1,  
if ( r.idpersoana2 <> rude.idpersoana1 AND r.idpersoana2 <> rude.idpersoana2, r.idpersoana2, NULL )) AS B  
from rude  
inner join rude r on r.idpersoana1 = rude.idpersoana2  
having B is not null;  
end;  
call pr1();

5) Să se elaboreze o procedură care pentru o persoană ar determina numărul de prieteni. Ca parametru este dat numele persoanei

delimiter $$  
CREATE PROCEDURE proc5(nume\_pers VARCHAR(20))  
BEGIN  
SELECT COUNT(persoane.idPersoana)  
FROM persoane  
INNER JOIN amici ON persoane.idPersoana = amici.idpersoana1  
OR persoane.idPersoana = amici.idpersoana2  
WHERE persoane.Numele = nume\_pers  
GROUP BY persoane.idPersoana;  
END; $$  
delimiter ;  
CALL proc5("Elvi");

6) Să se elaboreze o funcție care returnează True sau False dacă două persoane sunt prieteni. Numele persoanelor sunt date ca parametru

DROP FUNCTION if EXISTS func6;  
delimiter $$  
CREATE FUNCTION func6(nume1 VARCHAR(40),nume2 VARCHAR(40))  
RETURNS VARCHAR(5)  
DETERMINISTIC  
BEGIN  
if EXISTS (SELECT \*  
FROM persoane  
INNER JOIN amici ON idpersoana = idpersoana1  
OR idpersoana=idpersoana2  
WHERE numele = nume1 OR numele = nume2 )  
then  
RETURN "TRUE";  
ELSE  
RETURN "FALSE";  
END if;  
END $$  
delimiter ;  
  
SELECT func6('Elvira','Kris');  
\*/