**MINISTERUL EDUCAŢIEI ȘI CERCETĂRII AL REPUBLICII MOLDOVA**

**Universitatea Tehnică a Moldovei**

**Facultatea Calculatoare, Informatică şi Microelectronică**

**Departamentul Ingineria Software și Automatică**

**Programul de studii: Tehnologia informației**

RAPORT

# LUCRARE DE LABORATOR NR. 8

# la Baze de Date

**Tema: Interogări cu joncțiuni și cu operatori din teoria mulțimilor**

A efectuat:

st. gr. TI-211 Popa Cătălin

A verificat: Olga Grosu

UTM, Chișinău 2023

**Interogări cu joncțiuni externe(LEFT, RIGHT, FULL).**

**LEFT JOIN**

--left join

--a fost realizata interogarea, in care se foloseste left join, se vor afisa datele din prima tabela si cele care corespund conditiei din tabela 2

select a.Id\_Student,

a.Id\_Disciplina,

a.Nota,

b.Nume\_Profesor,

b.Prenume\_Profesor,

b.Adresa\_Postala\_Profesor

from univer.studenti\_reusita a

left join univer.profesori b on a.Id\_Profesor = b.Id\_Profesor;

Comanda LEFT JOIN returnează toate rândurile din tabelul din stânga, indiferent dacă condiția de joncțiune nu se potrivește cu înregistrările din al doilea tabel, din dreapta.

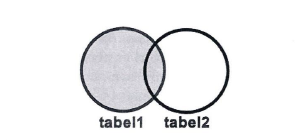


Figura 1 – LEFT JOIN.

Mai jos este afișat răspunsul după realizarea left join.



Figura 2 – Left join.

**RIGHT JOIN**

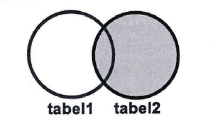


Figura 3 – Right join.

RIGHT JOIN returneză toate rândurile din al doilea tabel, chiar dacă nu coincid pe câmpurile de joncțiune cu rânduri din primul tabel.

--opusul la left join

--right join

select a.Id\_Student,

a.Id\_Disciplina,

a.Nota,

b.Nume\_Profesor,

b.Prenume\_Profesor,

b.Adresa\_Postala\_Profesor

from univer.studenti\_reusita a

right join univer.profesori b on a.Id\_Profesor = b.Id\_Profesor;

În figura 2, este afișat răspunsul interogării cu right join.

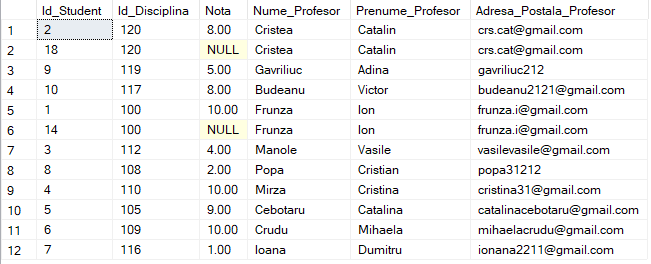


Figura 4 – Right join.

**FULL JOIN**

Funcționeză ca o uniune a două mulțimi. Returneză toate rândurile din al doilea tabel, choar dacă nu coincid pe câmpurile de joncțiune din primul tabel

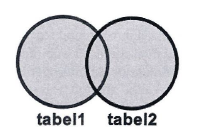


Figura 5 – FULL JOIN.

--va afisa toate datele din abele tabele

--full join

select a.Id\_Disciplina, b.Disciplina, b.Nr\_ore\_plan\_disciplina from univer.studenti\_reusita a

full join univer.discipline b on a.Id\_Disciplina = b.Id\_Disciplina;

În figura 6 , este afișarea după folosirea full join.

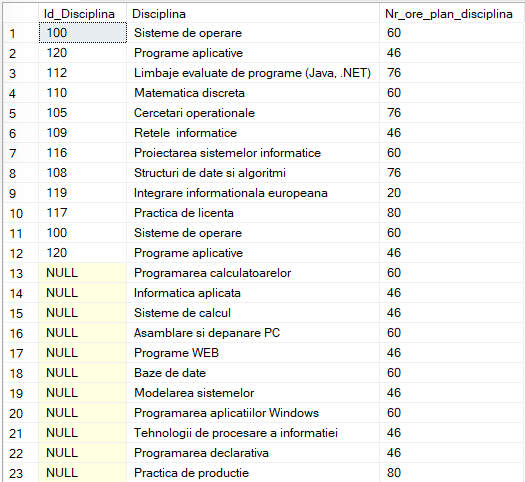


Figura 6 – Full join.

**Interogare cu joncțiune internă**

Joncțiunea INNER JOIN întoarce intersecția a două seturi de înregistrări.

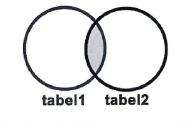


Figura 7 – INNER JOIN.

--iner join este cel mai des folosit, el face elgatura dintre doua tabele in baza unui camp anumit

--inner join

select a.Id\_Disciplina, b.Disciplina, b.Nr\_ore\_plan\_disciplina from univer.studenti\_reusita a,

univer.discipline b

where a.Id\_Disciplina = b.Id\_Disciplina;

Inner join returnează toate rândurile din ambele tabele, care coincid pe câmpurile de joncțiune.

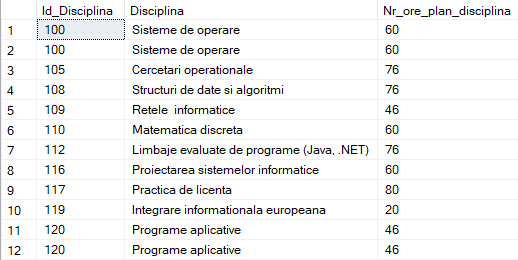


Figura 8 – Inner join.

**Produsul cartezian**

--cross join

--fiecare rand din primul tude cu toate randurile din tabelul al doilea

select \* from univer.discipline cross join univer.studenti\_reusita

În figura 9, este afișat rezultatul la crros join, unde are loc combinarea la fiecare rand din primul tabel cu toate randurile din al doilea tabel.

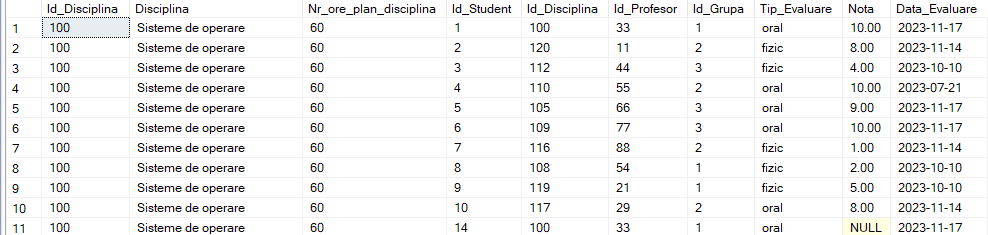


Figura 9 – CRROS JOIN.

**UNION**

UNION exclude automat rândurile duplicate din rezultat.

--union

--uneste doua rezultate de select, unde este nevoie sa fie acelasi numar de coloane cu aceleasi denumiri si elimina dublicatele

select a.\* from univer.studenti\_reusita a

where a.Nota > 5 and a.Data\_Evaluare = ('2023-11-17')

union

select a.\* from univer.studenti\_reusita a

where a.Nota > 5 and a.Data\_Evaluare = ('2023-11-14')

În figura 10, este afișat rezultatul la union.

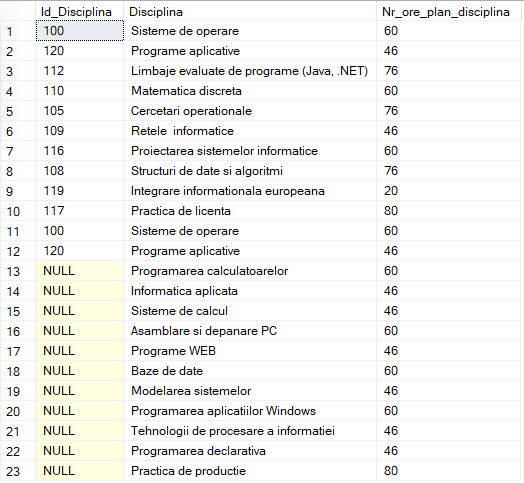


Figura 10 – Union.

**UNION ALL**

UNION ALL lucrează la fel ca și UNION, doar că rândurile dublicate nu sânt eliminate.

În figura 6, am realizar union all.

--functioneaza la fel ca UNION simplu doar ca nu exclude dublicatele

--union all

select a.\* from univer.studenti\_reusita a

where a.Nota > 5 and a.Data\_Evaluare > ('2023-11-10')

union all

select a.\* from univer.studenti\_reusita a

where a.Nota > 5 and a.Data\_Evaluare > ('2023-11-14');

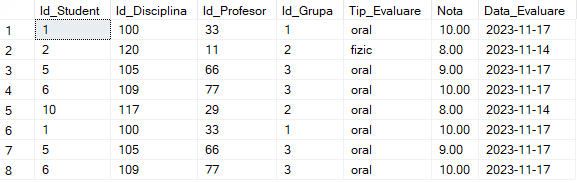


Figura 11 – Union All.

**INTERSECT**

Returnează rândurile găsite atât în rezultatul primei instrucțiuni SELECT, cât și al celei de-a doua instrucțiuni SELECT. Dublicate nu sunt afișate.

--returneaza datele care se intersecteaza din primul select cu al doilea

--INTERSECT

select b.Id\_Grupa

from univer.studenti\_reusita b

where b.Tip\_Evaluare like 'fizic'

INTERSECT

select b.Id\_Grupa

from univer.studenti\_reusita b

where b.Tip\_Evaluare like 'oral';

În figura 7 am realizat INTERSECT.



Figura 12 – INTERSECT.

**EXCEPT**

Returneză rândurile care apar ca rezultat al primei instrucțiuni SELECT și nu sunt găsite ca urmare a celei de-a doua instrucțiuni SELECT. Dublicate nu sunt afișate.

--returneaza datele din primul select, care nu se gasesc in selectul 2

--EXCEPT

select a.\* from univer.discipline a where a.Nr\_ore\_plan\_disciplina in(60, 76)

EXCEPT

select a.\* from univer.discipline a where a.Disciplina like '%Pro%';

În figura 8 am realizat EXCEPT.

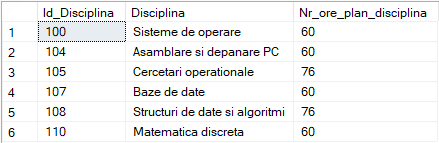


Figura 13 – EXCEPT.

**Concluzie**

În lucrarea dată am exersat în SQL interogări în care am folosit toate tipurile de joncțiuni, realizarea produsului cartezian, interogări cu UNION, UNION ALL, INTERSECT și EXCEPT. Aceste ne pot ajuta în calculi de statistici sau în alte situații când avem de calculate o cantitate mare de date. De asemenea, ele ușurează lucrul cu datele și munca fizică.