**Lab 8 : Administrarea viziunilor si a expresiilor-tabel**

**Task1: Sa se creeze doua viziuni in baza interogarilor formulate in doua exercitii indicate din capitolul Prima viziune sa fie construita in Editorul de interogari, iar a doua, utilizand View Designer.**

**Editor**

create view view1 as

select distinct s\_s.Nume\_Student, s\_s.Prenume\_Student

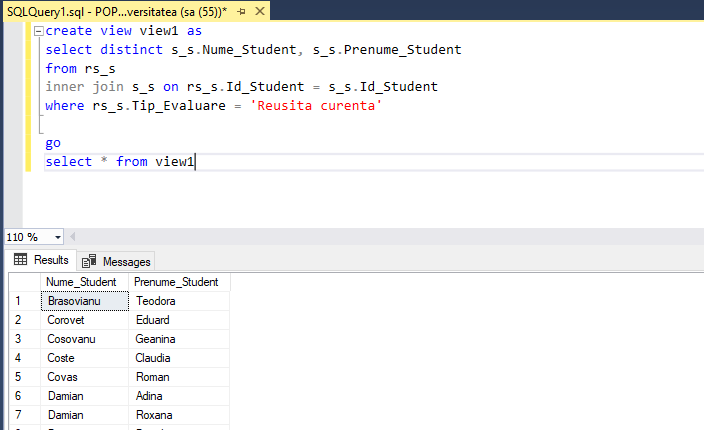
from rs\_s

inner join s\_s on rs\_s.Id\_Student = s\_s.Id\_Student

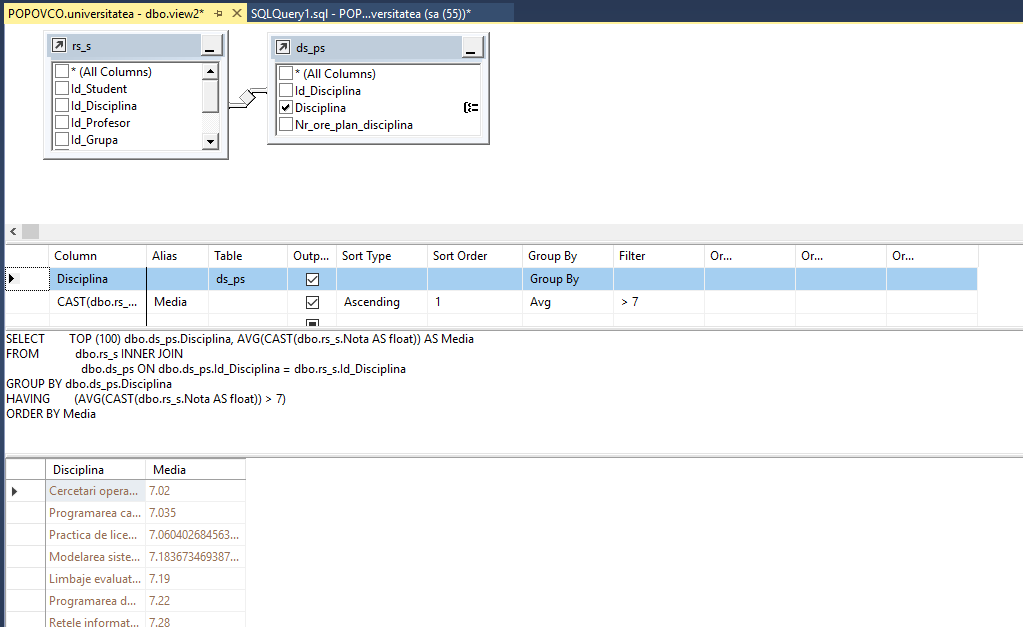
where rs\_s.Tip\_Evaluare = 'Reusita curenta'

go

select \* from view1



**Designer**



**Task2: Sa se scrie cate un exemplu de instructiuni INSERT, UPDATE, DELETE asupra viziunilor create. Sa se adauge comentariile respective referitoare la rezultatele executarii acestor instructiuni.**

create view view3 as

select Id\_Student,Nume\_Student, Prenume\_Student

from s\_s

insert into view3

values (999999,'Nume test', 'Prenume test')

select \* from view3



update view3

set Nume\_Student = 'Nume test2'

where Id\_Student = 999999

select \* from view3



delete from view3 where Nume\_Student = 'Nume test2'

select \* from view3



**Task3: Sa se scrie instructiunile SQL care ar modifica viziunile create (in exercitiul 1) in asa fel, incat sa nu fie posibila modificarea sau stergerea tabelelor pe care acestea sunt definite si viziunile sa nu accepte operatiuni DML, daca conditiile clauzei WHERE nu sunt satisfacute.**

alter view view1 with schemabinding as

select distinct studenti.Nume\_Student, studenti.Prenume\_Student

from studenti.studenti\_reusita

inner join studenti.studenti on studenti.studenti\_reusita.Id\_Student = studenti.studenti.Id\_Student

where studenti.studenti\_reusita.Tip\_Evaluare = 'Reusita curenta'

with check option;

alter view view2 with schemabinding as

select plan\_studii.discipline.Disciplina, AVG(cast(studenti.studenti\_reusita.Nota as float)) as Media

from studenti.studenti\_reusita

inner join plan\_studii.discipline on studenti.studenti\_reusita.Id\_Disciplina = plan\_studii.discipline.Id\_Disciplina

group by plan\_studii.discipline.Disciplina

having AVG(cast(studenti.studenti\_reusita.Nota as float)) > 7

with check option;

**Task4: Sa se scrie instructiunile de testare a proprietatilor noi definite**

alter table studenti.studenti drop column Nume\_Student

insert into view1

values('Tester','Tester')

alter table plan\_studii.discipline drop column Disciplina

insert into view2

values('Testing', 1)

**Task5: Sa se rescrie 2 interogari formulate in exercitiile din capitolul 4, in asa fel incat interogarile imbricate sa fie redate sub forma expresiilor CTE.**

with interogare1 (Id\_Student) as(

select s\_s.Id\_Student

from s\_s

where Nume\_Student = 'Orian'

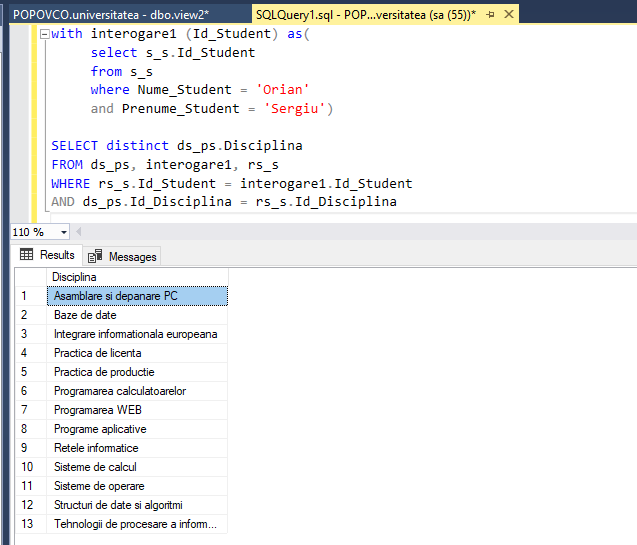
and Prenume\_Student = 'Sergiu')

select distinct ds\_ps.Disciplina

from ds\_ps, interogare1, rs\_s

where rs\_s.Id\_Student = interogare1.Id\_Student

and ds\_ps.Id\_Disciplina = rs\_s.Id\_Disciplina



--39 Gasiti denumirile disciplinelor la care nu au sustinut examenul, in medie, peste 5% de studenti.

with interogarea39 (Disciplina) as

(select distinct ds\_ps.Disciplina

from rs\_s

inner join ds\_ps on rs\_s.Id\_Disciplina = ds\_ps.Id\_Disciplina

inner join s\_s on rs\_s.Id\_Student = s\_s.ID\_Student

where rs\_s.Tip\_Evaluare = 'Examen'

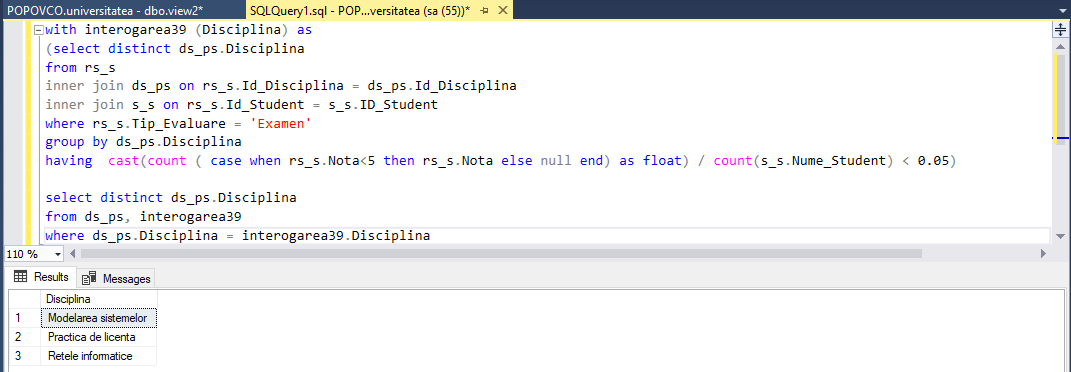
group by ds\_ps.Disciplina

having cast(count ( case when rs\_s.Nota<5 then rs\_s.Nota else null end) as float) / count(s\_s.Nume\_Student) < 0.05)

select distinct ds\_ps.Disciplina

from ds\_ps, interogarea39

where ds\_ps.Disciplina = interogarea39.Disciplina



**Task6:**

**6.1: Se considera un graf orientat, si fie se doreste parcursa calea de la nodul id = 3 la nodul unde id = 0. Sa se faca reprezentarea grafului orientat in forma de expresie-tabel recursiv.**

[4]

|

v

[5] -> [0] <- [1] <- [2]

^

|

[3]

**6.2: Sa se observe instructiunea de dupa UNION ALL a membrului recursiv, precum si partea de pana la UNION ALL reprezentata de membrul-ancora.**

create table task6 (

number1 int primary key,

number2 int);

insert into task6

values

(5,0), (4,2), (3,2), (1,0), (2,1), (0, null);

select \* from task6;

with task6CTE AS (

select number1 , number2 from task6

where number1 = 3 and number2 = 2

union all

select task6.number1, task6.number2 from task6

inner join task6CTE

on task6.number1 = task6CTE.number2

)

SELECT \* from task6CTE

