

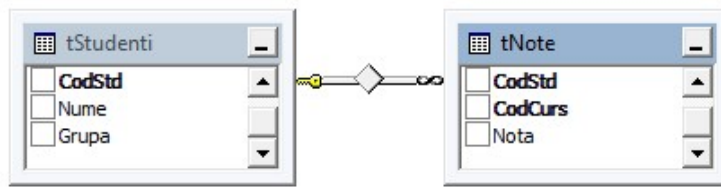
Subinterogari

1. Obiectivul lucrării:
Formarea și dezvoltarea abilităților de interogare a bazelor de date.
2. Breviar teoretic cu exerciții și probleme rezolvate

Subinterogarea este o comandă SELECT care apare într-o clauză a altei comenzi select. O subinterogare poate conține la rândul său alte subinterogari.

Subinterogările pot fi plasate în clauzele WHERE, HAVING, FROM sau SELECT și trebuie să fie scrise între paranteze rotunde.

Pentru exemplificarea lucrului cu subinterogări, considerăm următorul model relațional:



Scriptul următor crează și populează cu date tabelele aplicației.

```
create database dbStudenti
go

use dbStudenti

create table tStudenti
( CodStd char(10) primary key,
  Nume varchar(25) not null,
  Grupa int
)
create table tNote
( CodStd char(10) foreign key references tStudenti(CodStd),
  CodCurs char(10),
  Nota int check(nota between 1 and 10),
  primary key(CodStd,CodCurs)
)

insert into tStudenti
values('s1','student1',1),
      ('s2','student2',1),
      ('s3','student3',1),
      ('s4','student4',2),
```

```

('s5','student5',2),
('s6','student6',2)

```

```

insert into tNote
values ('s1','c1',8), ('s1','c2',8), ('s2','c3',6), ('s2','c1',7),
      ('s3','c1',4), ('s3','c2',6), ('s3','c3',8),
      ('s4','c2',8), ('s4','c1',5), ('s5','c1',7), ('s5','c2',4)

```

```
select * from tStudenti
```

CodStd	Nume	Grupa
s1	student1	1
s2	student2	1
s3	student3	1
s4	student4	2
s5	student5	2
s6	student6	2

```
select * from tNote
```

CodStd	CodCurs	Nota
s1	c1	8
s1	c2	9
s2	c1	7
s2	c3	6
s3	c1	4
s3	c2	6
s3	c3	9
s4	c1	5
s4	c2	8
s5	c1	7
s5	c2	4

Exemple de subinterogări utilizate în clauzele *where* și *having*:

Să se afișeze *nume student, cod curs, nota* pentru studenții care au obținut nota cea mai mare

```

select Nume,CodCurs,Nota from tStudenti inner join tNote
on tStudenti.CodStd=tNote.CodStd
where Nota=(select max(Nota) from tNote)

```

Nume	CodCurs	Nota
student1	c2	9
student3	c3	9

Sa se obțină colegii de grupă ai studentului cu CodStd='s3':

```

select Nume,Grupa from tStudenti
where Grupa= (select Grupa from tStudenti where CodStd='s3')

```

Nume	Grupa
student1	1
student2	1
student3	1

Să se afișeze cursurile pentru care media notelor studenților este mai mare decât media notelor obținute la cursul cu CodCurs='c2'

```
select CodCurs, avg(convert(numeric(4,2),Nota)) as media
from tNote
group by CodCurs
having avg(convert(numeric(4,2),Nota)) >
      (select avg(convert(numeric(4,2),Nota))
       from tNote
       where CodCurs='c2'
      )
```

CodCurs	media
c3	7.500000

Subinterogarea din exemplele anterioare sunt caracterizate faptul că întorc un rezultat care este transmis subinterogării de nivel superior. Rezultatul furnizat de o astfel de subinterogare poate fi prelucrat în interogarea de nivel superior prin utilizarea oricărui operator de comparare =, >, <, <> sau !=, >= sau !<, <= sau !=.

Dacă subinterogarea întoarce mai multe rezultate putem folosi operatorii logici *IN*, *EXISTS*, *ALL*, *ANY*.

Operatorul **IN**

expresie **in** (set de valori)

returnează adevărat dacă *expresie* se aparține setului de valori specificat între paranteze.

Operatorul EXISTS

exists (subinterogare)

returnează adevărat dacă *subinterogare* întoarce cel puțin un rezultat.

Operatorii **in** și **exists** pot fi folosiți și în varianta negată **not in** și **not exists**

Operatorul **ALL** – condiția este adevărată dacă este verificată pentru toate elementele listei returnate de subinterogare.

Operatorul **ANY** sau **SOME** – condiția este adevărată dacă este verificată pentru cel puțin una din valorile returnate de subinterogare.

>Any este echivalentă cu *mai mare decât minimul din listă*

<Any este echivalentă cu *mai mic decât maximul din listă*

Operatorul **IN** poate fi înlocuit cu **=ANY**.

Să se determine studenții care au obținut note mai mici decât 5

```
select * from tStudenti
```

```
where CodStd in
( select distinct codStd from tNote
  where Nota<5 )
```

CodStd	Nume	Grupa
s3	student3	1
s5	student5	2

Să se afișeze studenții care nu s-au prezentat la examen la disciplina cu codul 'c3'

```
Select codStd, nume from tStudenti
where codStd not in (select codStd from tNote
                    where codCurs='c3')
```

codStd	nume
s1	student1
s4	student4
s5	student5
s6	student6

Să se afișeze studenții care nu s-au prezentat la niciun examen:

```
Select CodStd, nume, grupa from tStudenti
where CodStd <> all(select distinct CodStd
from tNote)
```

CodStd	nume	grupa
s6	student6	2

sau

```
Select CodStd, nume, grupa from tStudenti
where CodStd NOT IN(select distinct CodStd
from tNote)
```

CodStd	nume	grupa
s6	student6	2

Caracteristica comună a tuturor subinterogărilor din exemplele precedente este dată de faptul că valoarea returnată de subinterogare nu depinde în niciun fel de valorile produse de instrucțiunile SELECT exterioare ei. Din această cauză o astfel de subinterogare se numește *subinterogare fără sincronizare (necorelată)* sau *subinterogare îmbricată(încuibărită)*. În astfel de situații subinterogarea este evaluată independent de interogarea principală.

SQL poate să trateze și subinterogari care fac referire la coloanele tabelului interogării principale, astfel, este posibilă sincronizarea subinterogării cu interogarea principală.

Prelucrarea în acest caz este mai complexă, deoarece interogarea principală trebuie să evalueze subinterogarea pentru fiecare rând furnizat de interogarea principală.

Subinterogarea *corelată (cu sincronizare)* este caracterizată de faptul că face referire la una sau mai multe coloane ale interogării exterioare. Evaluarea subinterogării corelate se face pentru fiecare rând rezultat al interogării exterioare care o conține.

Exemplu:

Să se obțină lista studenților care au obținut note mai mari decât media notelor obținute la același curs:

```
select nume, CodCurs, Nota
from
    tStudenti as A
    inner join
    tNote as B
    on A.CodStd=B.CodStd
where Nota>= (select AVG(Nota*1.0) from tNote
              where CodCurs=B.CodCurs)
```

nume	CodCurs	Nota
student1	c1	8
student2	c1	7
student5	c1	7
student4	c2	8
student1	c2	9
student3	c3	9

Să se determine studenții care au obținut cel puțin o notă mai mare decât 8.

```
select Distinct * from tStudenti
    where exists(select * from tNote where codStd=tStudenti.codStd and
    nota>8)
```

CodStd	Nume	Grupa
s1	student1	1
s3	student3	1

Să se determine studenții care s-au prezentat la examen la cursul 'c3' :

```
select CodStd, Nume, Grupa
from tStudenti as A
where exists(select * from tNote where CodStd=A.CodStd and
              CodCurs='c3')
```

CodStd	Nume	Grupa
s2	student2	1
s3	student3	1

Obs. Enunțul precedent se rezolva mai simplu și mai eficient utilizand o subinterogare imbricata

```
select Distinct * from tStudenti
where codStd in (select codStd from tNote where nota>8)
```

CodStd	Nume	Grupa
s1	student1	1
s3	student3	1

Să se determine studenții care au obținut cea mai mare notă:

```
select A.CodStd, Nume,CodCurs,Nota from tStudenti as A inner join
tNote B
on A.CodStd=B.CodStd
where not exists(select * from tNote where Nota>B.Nota)
```

CodStd	Nume	CodCurs	Nota
s1	student1	c2	9
s3	student3	c3	9

Exemplu de subinterogare utilizată în clauza *select*:

Să se determine nota maximă obținută la examene de către fiecare student

```
select CodStd, Nume, (select max(nota) from tNote
where codStd=A.codStd ) as [Nota maxima]
from tStudenti as A
```

CodStd	Nume	Nota maxima
s1	student1	9
s2	student2	7
s3	student3	9
s4	student4	8
s5	student5	7
s6	student6	NULL

Exemplu de subinterogare utilizată în clauza *from*:

Să se determine, la nivel de grupă, maximul dintre mediile notelor obținute de studenți:

```
select Grupa,MAX(Media) as [Medie maxima]
from
tStudenti as A
inner join
(select CodStd,AVG(nota*1.0) as Media from tNote group by CodStd)
as B
```

```
        on A.CodStd=B.CodStd  
group by Grupa
```

Grupa	Medie maxima
--------------	---------------------

1	8.500000
---	----------

2	6.500000
---	----------

Sa se determine media mediilor studentilor la nivel de grupa

```
select grupa, avg(media) as [Media grupei]  
from  
(select A.CodStd,nume,grupa,avg(nota*1.0) as Media  
from tStudenti as A inner join tNote as B on A.CodStd=B.CodStd  
group by A.CodStd,nume,grupa) as A  
group by grupa
```