



TECHNICAL UNIVERSITY OF MOLDOVA

30.11.2018

DB Laboratory 9

Submitted To:

Maria Cojanu

Asst. Univ.

Computer Science

Department

Submitted By :

Sezgin Erol

Group FAF-161

Semester 1

Chisinau 2018

General purpose:

Learn about SQL Query Language

Tasks:

- Answer Questions at the end of Chapter 9;
- Solve ex. 1 - 8 at the end of Chapter 9.

Task Realization:

In Figure 1 we can see the two stored Procedures. Basically they do the same as queries from Lab4 but now they are stored in memory and can be accessed any time

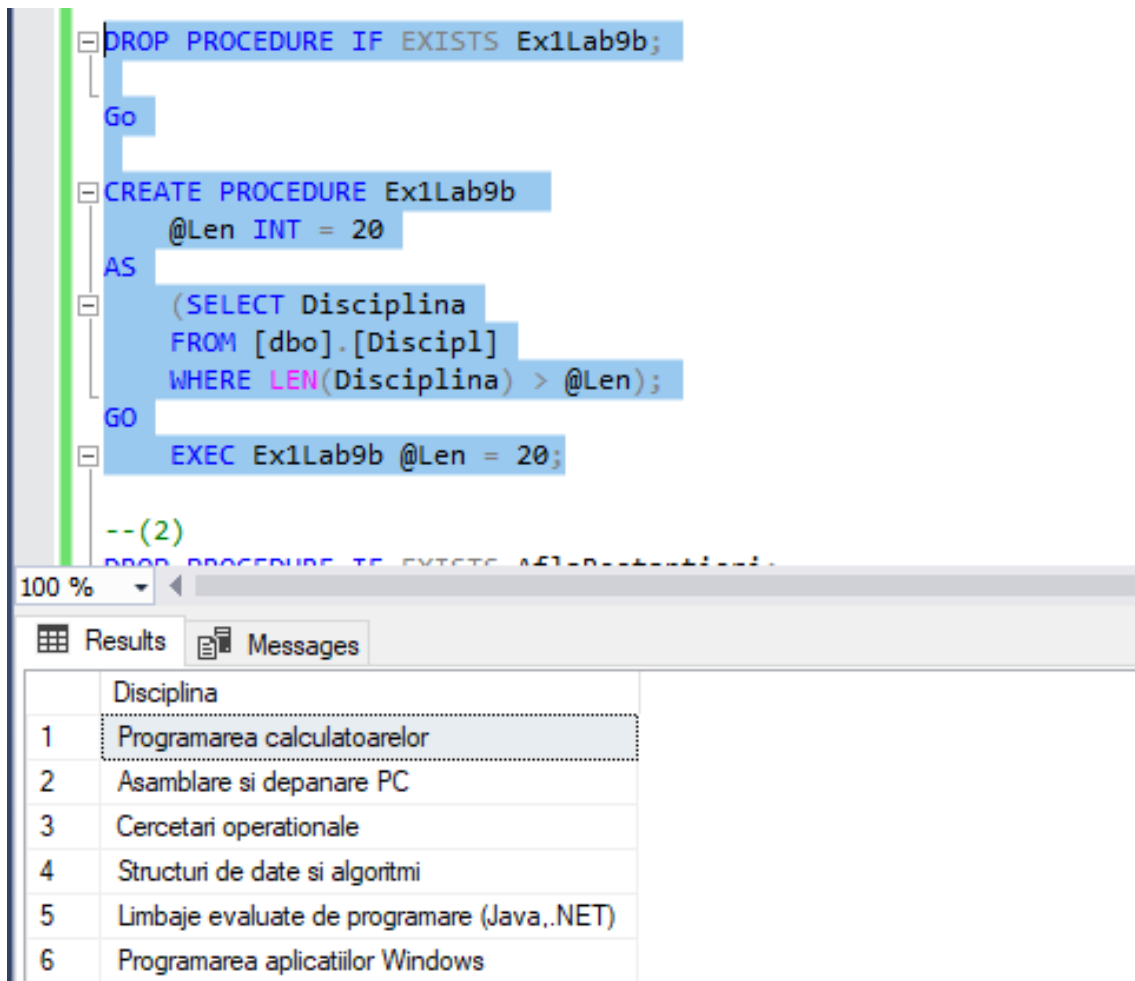


Figure 1: Ex1 Lab9

In figure 2 we can observe Stored procedure which doesn't have parameters but possess one output value, the total number of student who didnt pass any exam.

```
--(2)
DROP PROCEDURE IF EXISTS AflaRestantieri;
GO
CREATE PROCEDURE AflaRestantieri
    @restantieri INT OUTPUT
AS
    SELECT @restantieri=COUNT(DISTINCT Id_Student)
    FROM [dbo].[SReusita]
    WHERE Nota < 5 OR Nota IS NULL
GO
DECLARE @Result INT=0;
EXEC AflaRestantieri @restantieri = @Result output;
SELECT @Result as [NumarRestantieri];
```

100 %

Results Messages

	NumarRestantieri
1	56

Figure 2: Ex2 Lab9

In figure 3 i created 2 functions which return same result as queries from Lab4.

```
--(6)
DROP FUNCTION IF EXISTS Ex6Lab9;
GO
CREATE FUNCTION Ex6Lab9(@street VARCHAR(20))
RETURNS TABLE
AS
RETURN
(
    (select Nume_Student as last_name, Prenume_Student as first_name from [dbo].[Stdnt]
     where Adresa_Postala_Student like '%' + @street + '%')
    UNION
    (select Nume_Profesor as last_name, Prenume_Profesor as first_name from Prof
     where Adresa_Postala_Profesor like '%' + @street + '%')
)
GO
SELECT * FROM Ex6Lab9('31 August');
```

100 %

Results Messages

	last_name	first_name
1	Dascal	Florina
2	Goia	Ariana
3	Hanea	Marius
4	Pop	Irina

Figure 3: Query ex 6

In figure 4 we can observe user function which calculates student's age.

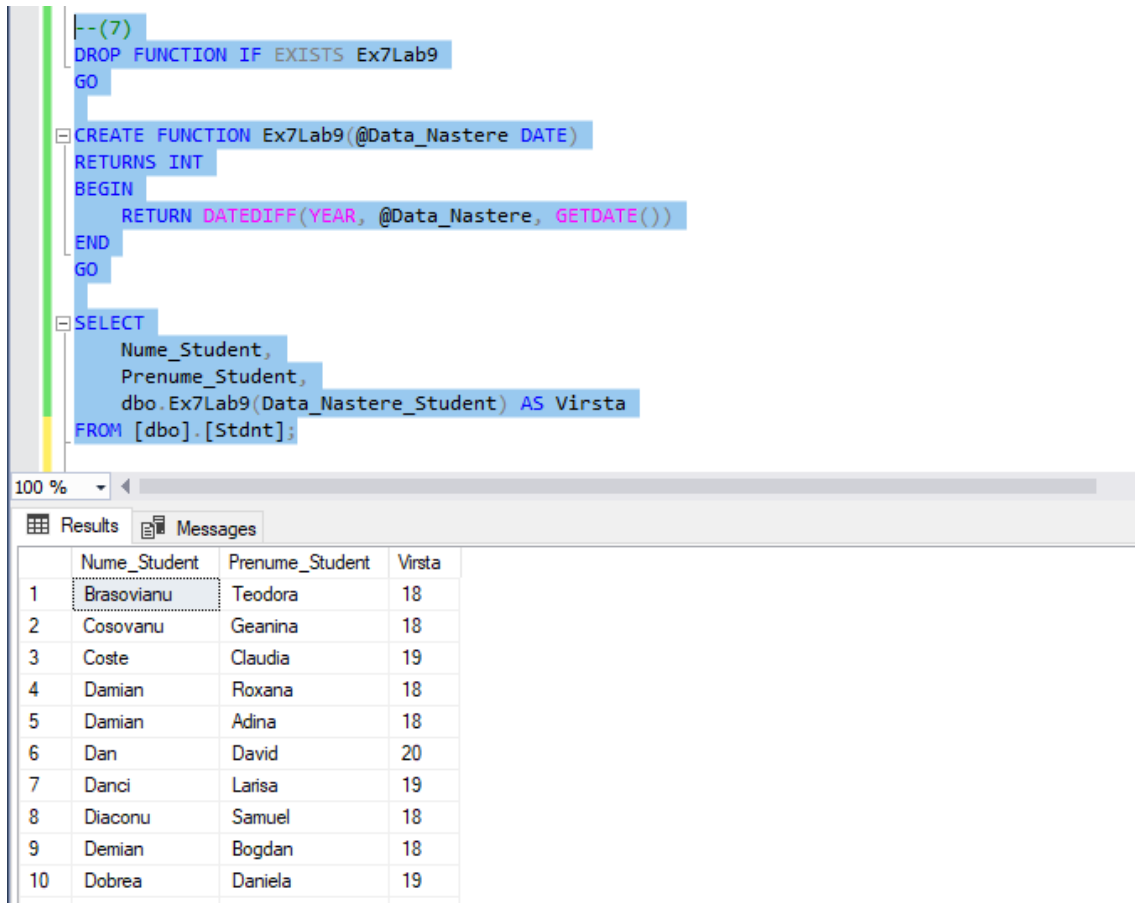


Figure 4: Query ex 7

In figure 5 we can observe user function which takes as input parameters first and last name of certain student and show his performance in study.

```

--(8)
Use universitatea
Go
drop function if exists Ex8Lab9
Go
create function Ex8Lab9(@Nume_Prenume_Student as varchar(255))
returns table
as
return
select concat(Nume_Student, ' ', Prenume_Student) Nume_Prenume, Disciplina, Nota, Data_Evaluare
from [dbo].[Stdnt] st, [dbo].[Discipl] d, [dbo].[SReusita] st_r
where st.Id_Student = st_r.Id_Student and st_r.Id_Disciplina = d.Id_Disciplina
and @Nume_Prenume_Student = concat(Nume_Student, ' ', Prenume_Student)
go
select * from Ex8Lab9('Demian Bogdan')
--(9)

```

Nume_Prenume	Disciplina	Nota	Data_Evaluare
Demian Bogdan	Cercetari operationale	7	2018-01-25
Demian Bogdan	Cercetari operationale	7	2018-12-03
Demian Bogdan	Cercetari operationale	9	2017-10-06
Demian Bogdan	Cercetari operationale	9	2017-12-09
Demian Bogdan	Baze de date	8	2018-01-15
Demian Bogdan	Baze de date	8	2018-12-03
Demian Bogdan	Baze de date	8	2017-10-10
Demian Bogdan	Baze de date	8	2017-12-15
Demian Bogdan	Structuri de date si ...	4	2018-01-09
Demian Bogdan	Structuri de date si ...	4	2018-12-03

Figure 5: Query ex 8

In figure 6 we can observe user function which takes as input parameters the from and shows the best student or the worst one from this group.

```
Drop function if exists Ex9Lab9
Go
create function Ex9Lab9(@CodGrupa varchar(6), @Is_Good varchar(20))
returns @result table (Grupa varchar(6), Nume_Prenume varchar(50), Nota_Medie decimal(4, 2), Is_Good varchar(20))
with encryption
as
begin
if(@Is_Good = 'sarguincios')
Insert @result
Select top 1 Cod_Grupa, concat(Nume_Student, ' ', Prenume_Student) as Nume_Prenume, convert(decimal(5,2),round(Avg(Nota+0.0),3))
from grupe g, studenti.studenti st, studenti.studenti_reusita st_R
where g.Id_Grupa = st_R.Id_Grupa and st_R.Id_Student = st.Id_Student
and Cod_Grupa = @CodGrupa
Group by Cod_Grupa, concat(Nume_Student, ' ', Prenume_Student)
order by Nota_Medie desc
else
if(@Is_Good = 'slab')
Insert @result
Select top 1 Cod_Grupa, concat(Nume_Student, ' ', Prenume_Student) as Nume_Prenume, convert(decimal(5,2),round(Avg(Nota+0.0),3))
from grupe g, studenti.studenti st, studenti.studenti_reusita st_R
where g.Id_Grupa = st_R.Id_Grupa and st_R.Id_Student = st.Id_Student
and Cod_Grupa = @CodGrupa
Group by Cod_Grupa, concat(Nume_Student, ' ', Prenume_Student)
order by Nota_Medie asc
```

100 %

Results Messages

	Grupa	Nume_Prenume	Nota_Medie	Is_Good
1	CIB171	Coste Claudia	6.25	slab

	Grupa	Nume_Prenume	Nota_Medie	Is_Good
1	T1171	Luca Alex	8.64	sarguincios

Figure 6: Query ex 9

Conclusion

During This lab work i find out how to Make stored procedures and functions. this is extremely important in real life programming because it makes programmer not to lose his time and write similar queries a lot of times. [1]

References

- [1] SQL Server Management Studio 2017, Tutorials for Lab 9
- [2] MSSQL Official Documentation <https://docs.microsoft.com/en-us/sql/t-sql/language-elements/try-catch-transact-sql?view=sql-server-2017>