**Лабораторна робота №1**

**Тема:** Адміністрування БД

***Мета роботи:*** навчитися досліджувати БД за допомогою T-SQL.

Хід роботи

**Виконання:**

1. **Вивчаємо сервера.**

-- Імена сервера і примірника

Select @@SERVERNAME as [Server\Instance];

-- версія SQL Server

Select @@VERSION as SQLServerVersion;

-- примірник SQL Server

Select @@ServiceName AS ServiceInstance;

-- Поточна БД (БД, в контексті якої виконується запит)

Select DB\_NAME() AS CurrentDB\_Name;

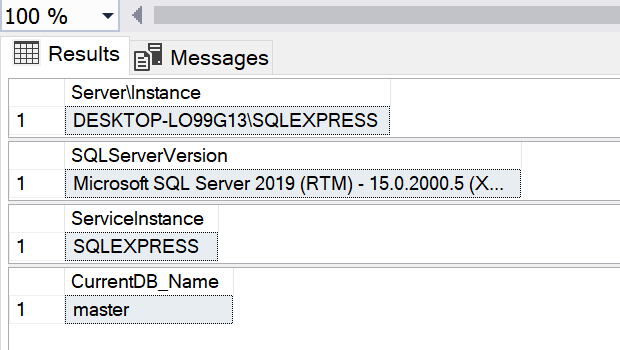


Рис. 1.1 Результат виконання запиту.

SELECT @@Servername AS ServerName ,

create\_date AS ServerStarted ,

DATEDIFF(s, create\_date, GETDATE()) / 86400.0 AS DaysRunning ,

DATEDIFF(s, create\_date, GETDATE()) AS SecondsRunnig

FROM sys.databases

WHERE name = 'tempdb';

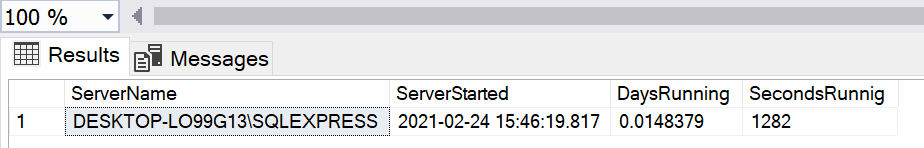


Рис. 1.2 Результат виконання запиту.

EXEC sp\_helpserver;

--OR

EXEC sp\_linkedservers;

--OR

SELECT @@SERVERNAME AS Server ,

Server\_Id AS LinkedServerID ,

name AS LinkedServer ,

Product ,

Provider ,

Data\_Source ,

Modify\_Date

FROM sys.servers

ORDER BY name;

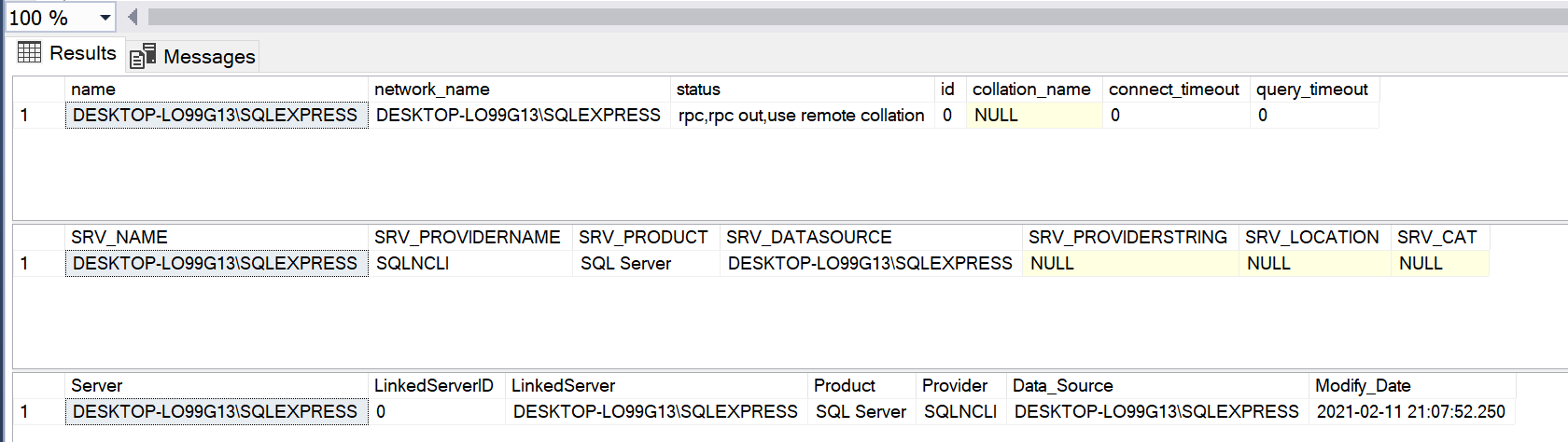


Рис. 1.3 Результат виконання запиту.

EXEC sp\_helpdb;

--OR

EXEC sp\_Databases;

--OR

SELECT @@SERVERNAME AS Server ,

name AS DBName ,

recovery\_model\_Desc AS RecoveryModel ,

Compatibility\_level AS CompatiblityLevel ,

create\_date ,

state\_desc

FROM sys.databases

ORDER BY Name;

--OR

SELECT @@SERVERNAME AS Server ,

d.name AS DBName ,

create\_date ,

compatibility\_level ,

m.physical\_name AS FileName

FROM sys.databases d

JOIN sys.master\_files m ON d.database\_id = m.database\_id

WHERE m.[type] = 0 -- data files only

ORDER BY d.name;

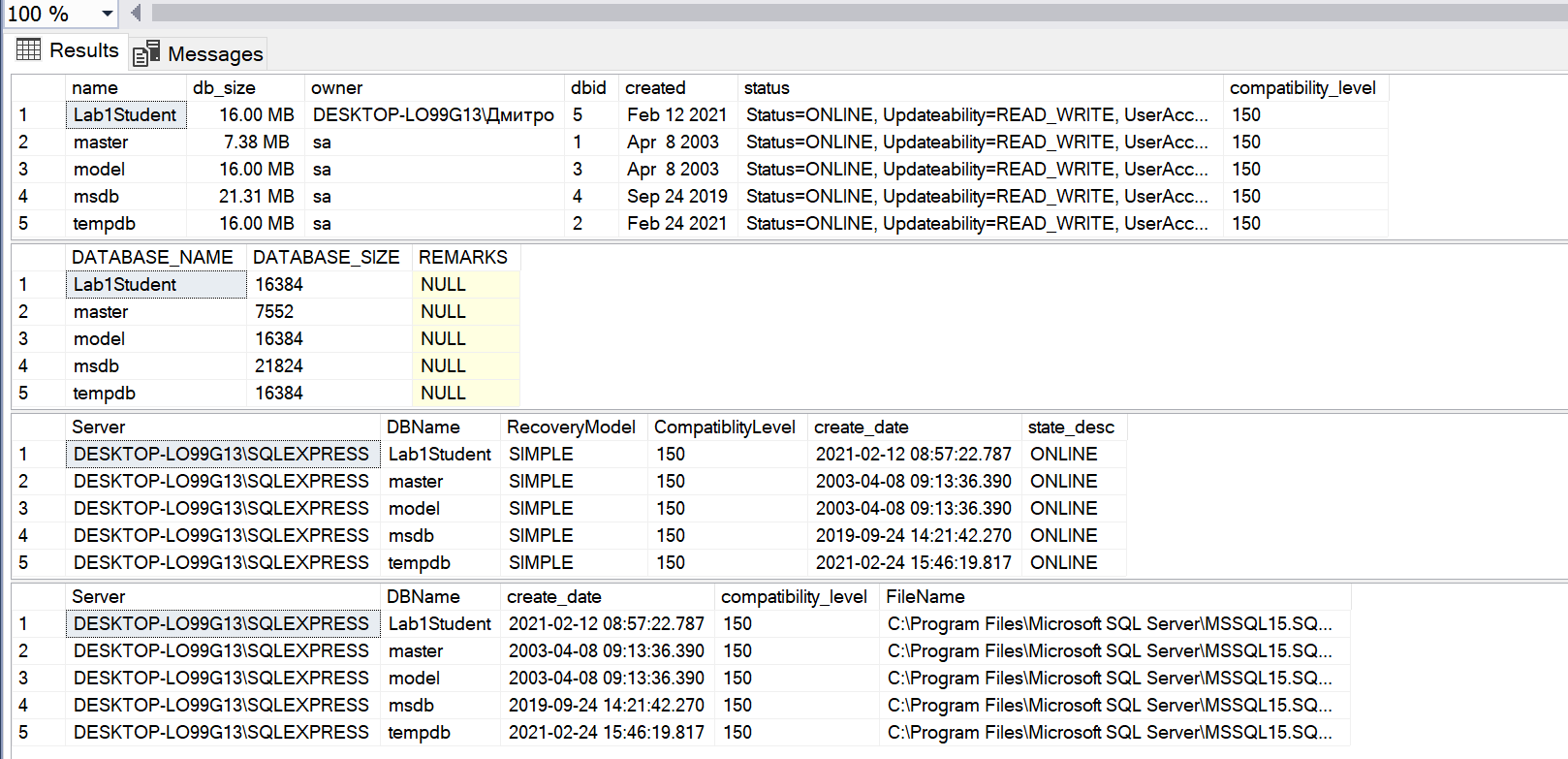


Рис. 1.4 Результат виконання запиту.

**2. Вивчаємо бази даних.**

USE MyDatabase;

SELECT \*

FROM sys.objects

WHERE type = 'U';

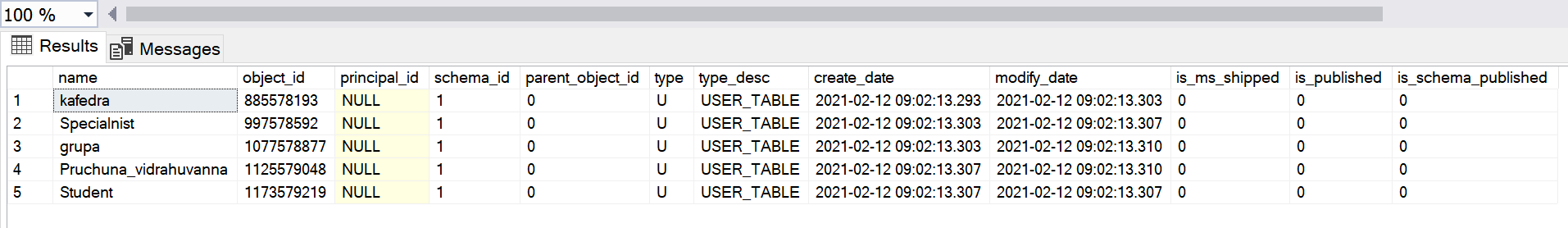


Рис. 1.5 Результат виконання запиту.

EXEC sp\_Helpfile;

--OR

SELECT @@Servername AS Server ,

DB\_NAME() AS DB\_Name ,

File\_id ,

Type\_desc ,

Name ,

LEFT(Physical\_Name, 1) AS Drive ,

Physical\_Name ,

RIGHT(physical\_name, 3) AS Ext ,

Size ,

Growth

FROM sys.database\_files

ORDER BY File\_id;

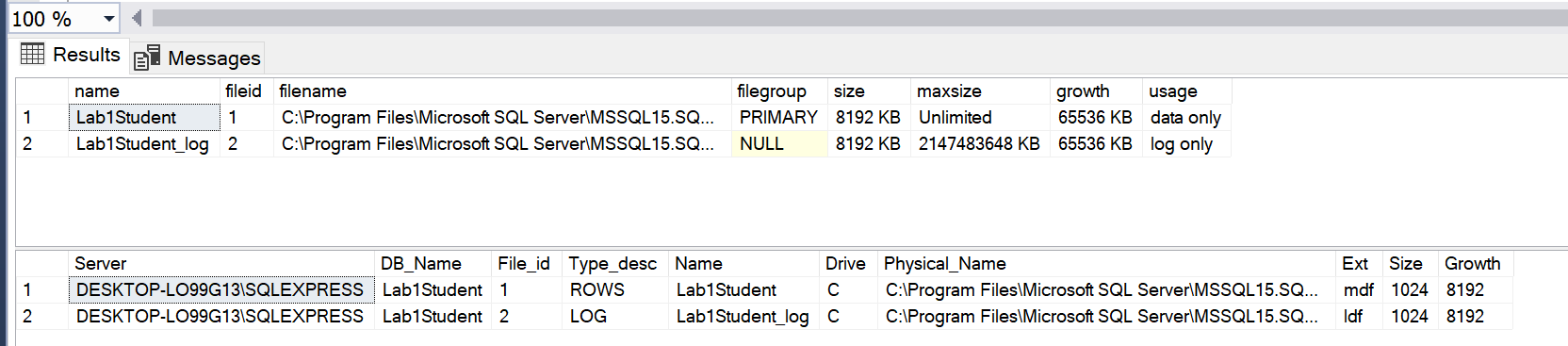


Рис. 1.6 Результат виконання запиту.

**3. Досліджуємо таблиці.**

EXEC sp\_tables; -- Пам'ятайте, що цей метод поверне і таблиці, і відображення

--OR

SELECT @@Servername AS ServerName ,

TABLE\_CATALOG ,

TABLE\_SCHEMA ,

TABLE\_NAME

FROM INFORMATION\_SCHEMA.TABLES

WHERE TABLE\_TYPE = 'BASE TABLE'

ORDER BY TABLE\_NAME ;

--OR

SELECT @@Servername AS ServerName ,

DB\_NAME() AS DBName ,

o.name AS 'TableName' ,

o.[Type] ,

o.create\_date

FROM sys.objects o

WHERE o.Type = 'U' -- User table

ORDER BY o.name;

--OR

SELECT @@Servername AS ServerName ,

DB\_NAME() AS DBName ,

t.Name AS TableName,

t.[Type],

t.create\_date

FROM sys.tables t

ORDER BY t.Name;

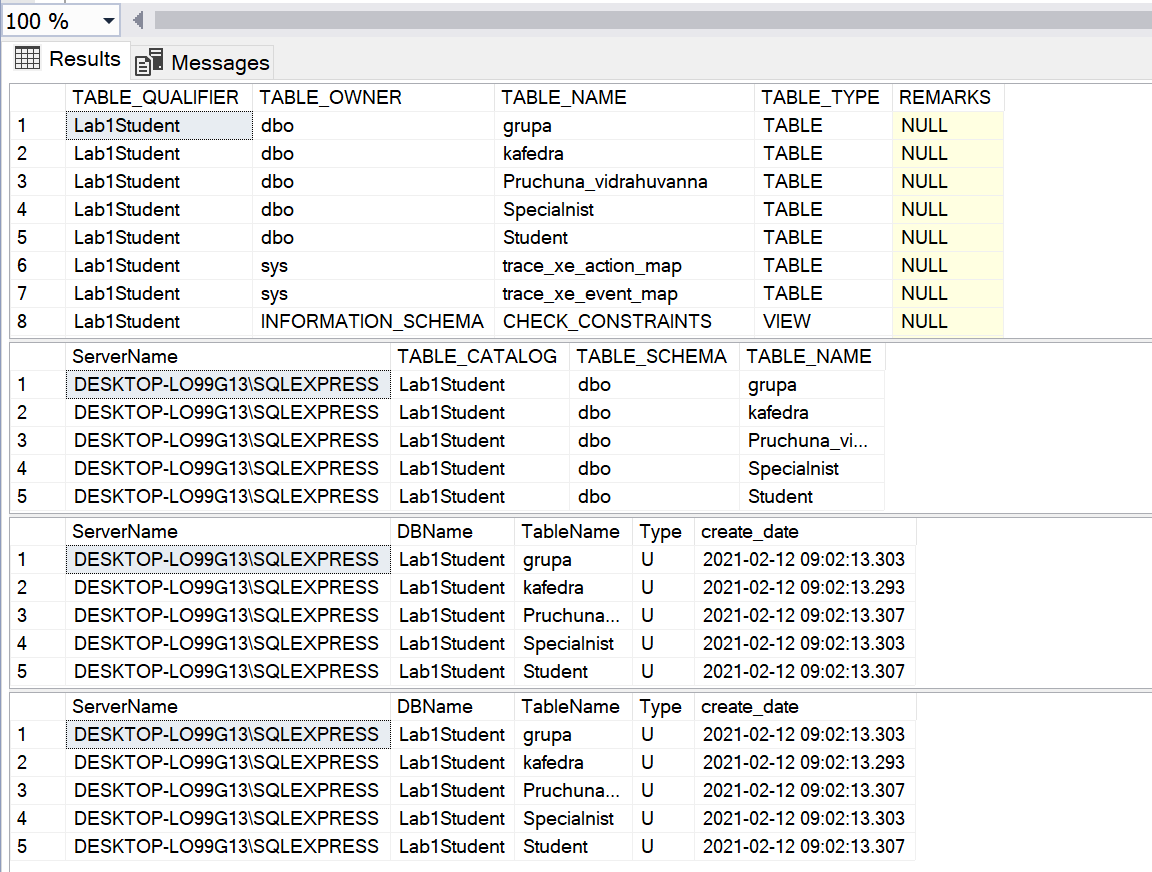


Рис. 1.7 Результат виконання запиту.

SELECT 'Select ''' + DB\_NAME() + '.' + SCHEMA\_NAME(SCHEMA\_ID) + '.'

+ LEFT(o.name, 128) + ''' as DBName, count(\*) as Count From ' + SCHEMA\_NAME(SCHEMA\_ID) + '.' + o.name

+ ';' AS ' Script generator to get counts for all tables'

FROM sys.objects o

WHERE o.[type] = 'U'

ORDER BY o.name;

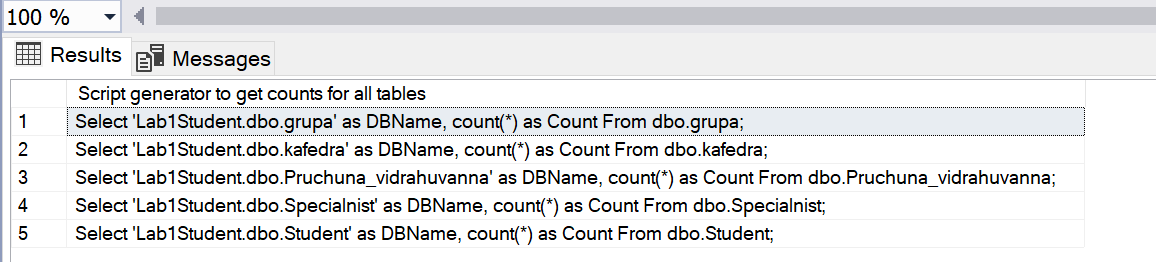


Рис. 1.8 Результат виконання запиту.

SELECT @@ServerName AS ServerName ,

DB\_NAME() AS DBName ,

OBJECT\_NAME(ddius.object\_id) AS TableName ,

SUM(ddius.user\_seeks + ddius.user\_scans + ddius.user\_lookups)

AS Reads ,

SUM(ddius.user\_updates) AS Writes ,

SUM(ddius.user\_seeks + ddius.user\_scans + ddius.user\_lookups

+ ddius.user\_updates) AS [Reads&Writes] ,

( SELECT DATEDIFF(s, create\_date, GETDATE()) / 86400.0

FROM master.sys.databases

WHERE name = 'tempdb'

) AS SampleDays ,

( SELECT DATEDIFF(s, create\_date, GETDATE()) AS SecoundsRunnig

FROM master.sys.databases

WHERE name = 'tempdb'

) AS SampleSeconds

FROM sys.dm\_db\_index\_usage\_stats ddius

INNER JOIN sys.indexes i ON ddius.object\_id = i.object\_id

AND i.index\_id = ddius.index\_id

WHERE OBJECTPROPERTY(ddius.object\_id, 'IsUserTable') = 1

AND ddius.database\_id = DB\_ID()

GROUP BY OBJECT\_NAME(ddius.object\_id)

ORDER BY [Reads&Writes] DESC;

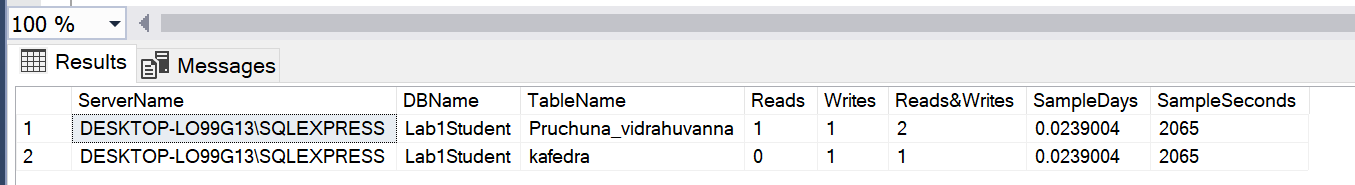


Рис. 1.9 Результат виконання запиту.

DECLARE DBNameCursor CURSOR

FOR

SELECT Name

FROM sys.databases

WHERE Name NOT IN ( 'master', 'model', 'msdb', 'tempdb',

'distribution' )

ORDER BY Name;

DECLARE @DBName NVARCHAR(128)

DECLARE @cmd VARCHAR(4000)

IF OBJECT\_ID(N'tempdb..TempResults') IS NOT NULL

BEGIN

DROP TABLE tempdb..TempResults

END

CREATE TABLE tempdb..TempResults

(

ServerName NVARCHAR(128) ,

DBName NVARCHAR(128) ,

TableName NVARCHAR(128) ,

Reads INT ,

Writes INT ,

ReadsWrites INT ,

SampleDays DECIMAL(18, 8) ,

SampleSeconds INT

)

OPEN DBNameCursor

FETCH NEXT FROM DBNameCursor INTO @DBName

WHILE @@fetch\_status = 0

BEGIN

SELECT @cmd = 'Use ' + @DBName + '; '

SELECT @cmd = @cmd + ' Insert Into tempdb..TempResults

SELECT @@ServerName AS ServerName,

DB\_NAME() AS DBName,

object\_name(ddius.object\_id) AS TableName ,

SUM(ddius.user\_seeks

+ ddius.user\_scans

+ ddius.user\_lookups) AS Reads,

SUM(ddius.user\_updates) as Writes,

SUM(ddius.user\_seeks

+ ddius.user\_scans

+ ddius.user\_lookups

+ ddius.user\_updates) as ReadsWrites,

(SELECT datediff(s,create\_date, GETDATE()) / 86400.0

FROM sys.databases WHERE name = ''tempdb'') AS SampleDays,

(SELECT datediff(s,create\_date, GETDATE())

FROM sys.databases WHERE name = ''tempdb'') as SampleSeconds

FROM sys.dm\_db\_index\_usage\_stats ddius

INNER JOIN sys.indexes i

ON ddius.object\_id = i.object\_id

AND i.index\_id = ddius.index\_id

WHERE objectproperty(ddius.object\_id,''IsUserTable'') = 1 --True

AND ddius.database\_id = db\_id()

GROUP BY object\_name(ddius.object\_id)

ORDER BY ReadsWrites DESC;'

EXECUTE (@cmd)

FETCH NEXT FROM DBNameCursor INTO @DBName

END

CLOSE DBNameCursor

DEALLOCATE DBNameCursor

SELECT \*

FROM tempdb..TempResults

ORDER BY DBName ,

TableName;

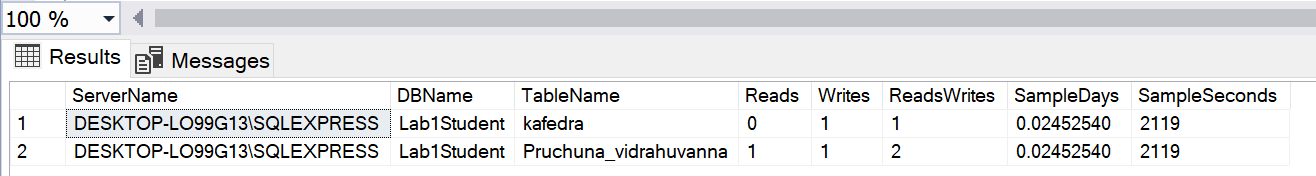


Рис. 1.10 Результат виконання запиту.

**4. Досліджуємо відображення.**

SELECT @@Servername AS ServerName ,

DB\_NAME() AS DBName ,

o.name AS ViewName ,

o.[Type] ,

o.create\_date

FROM sys.objects o

WHERE o.[Type] = 'V' -- View

ORDER BY o.NAME

--OR

SELECT @@Servername AS ServerName ,

DB\_NAME() AS DBName ,

Name AS ViewName ,

create\_date

FROM sys.Views

ORDER BY Name

--OR

SELECT @@Servername AS ServerName ,

TABLE\_CATALOG ,

TABLE\_SCHEMA ,

TABLE\_NAME ,

TABLE\_TYPE

FROM INFORMATION\_SCHEMA.TABLES

WHERE TABLE\_TYPE = 'VIEW'

ORDER BY TABLE\_NAME

--OR

-- CREATE VIEW Code

SELECT @@Servername AS ServerName ,

DB\_NAME() AS DB\_Name ,

o.name AS 'ViewName' ,

o.Type ,

o.create\_date ,

sm.[DEFINITION] AS 'View script'

FROM sys.objects o

INNER JOIN sys.sql\_modules sm ON o.object\_id = sm.OBJECT\_ID

WHERE o.Type = 'V' -- View

ORDER BY o.NAME;

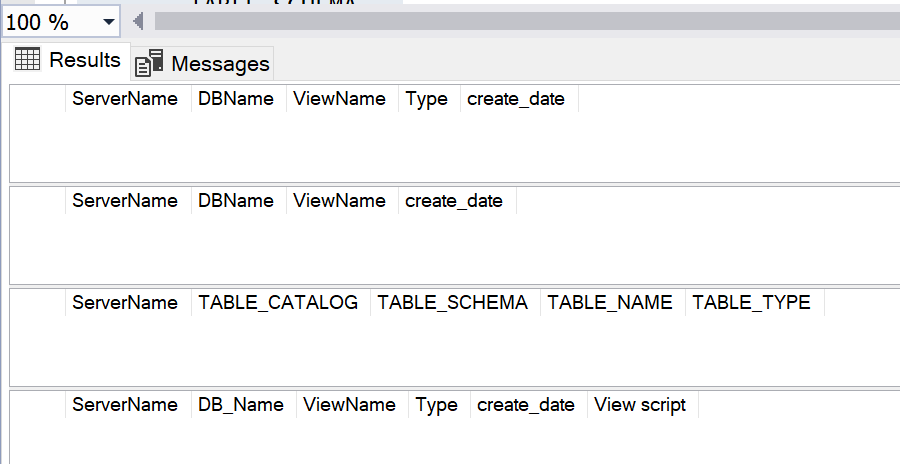


Рис. 1.11 Результат виконання запиту.

**5. Збережені процедури.**

SELECT @@Servername AS ServerName ,

DB\_NAME() AS DBName ,

o.name AS StoredProcedureName ,

o.[Type] ,

o.create\_date

FROM sys.objects o

WHERE o.[Type] = 'P' -- Stored Procedures

ORDER BY o.name

--OR

-- Додаткова інформація про ЗП

SELECT @@Servername AS ServerName ,

DB\_NAME() AS DB\_Name ,

o.name AS 'ViewName' ,

o.[type] ,

o.Create\_date ,

sm.[definition] AS 'Stored Procedure script'

FROM sys.objects o

INNER JOIN sys.sql\_modules sm ON o.object\_id = sm.object\_id

WHERE o.[type] = 'P' -- Stored Procedures

-- AND sm.[definition] LIKE '%insert%'

-- AND sm.[definition] LIKE '%update%'

-- AND sm.[definition] LIKE '%delete%'

-- AND sm.[definition] LIKE '%tablename%'

ORDER BY o.name;

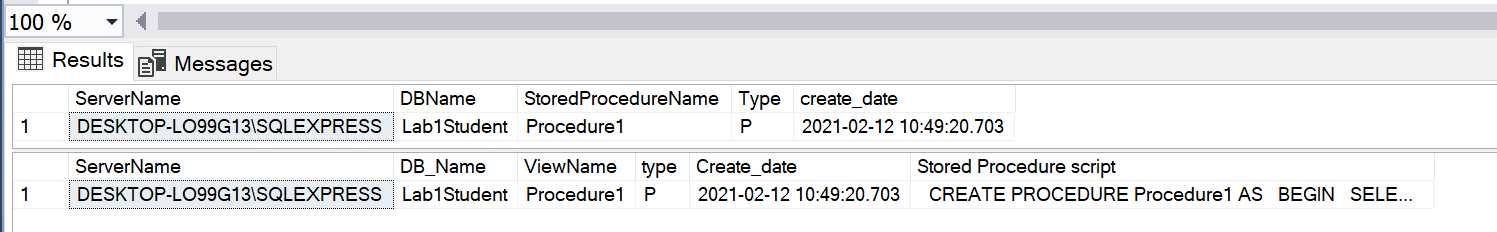


Рис. 1.12 Результат виконання запиту.

1. **Функції**

SELECT @@Servername AS ServerName ,

DB\_NAME() AS DB\_Name ,

o.name AS 'Functions' ,

o.[Type] ,

o.create\_date

FROM sys.objects o

WHERE o.Type = 'FN' -- Function

ORDER BY o.NAME;

--OR

-- Додаткова інформація про функції

SELECT @@Servername AS ServerName ,

DB\_NAME() AS DB\_Name ,

o.name AS 'FunctionName' ,

o.[type] ,

o.create\_date ,

sm.[DEFINITION] AS 'Function script'

FROM sys.objects o

INNER JOIN sys.sql\_modules sm ON o.object\_id = sm.OBJECT\_ID

WHERE o.[Type] = 'FN' -- Function

ORDER BY o.NAME;

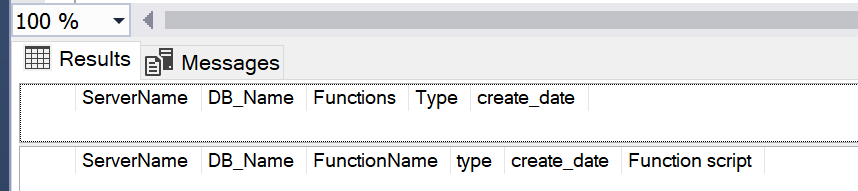


Рис. 1.13 Результат виконання запиту.

1. **Тригери.**

SELECT @@Servername AS ServerName ,

DB\_NAME() AS DBName ,

parent.name AS TableName ,

o.name AS TriggerName ,

o.[Type] ,

o.create\_date

FROM sys.objects o

INNER JOIN sys.objects parent ON o.parent\_object\_id = parent.object\_id

WHERE o.Type = 'TR' -- Triggers

ORDER BY parent.name ,

o.NAME

--OR

SELECT @@Servername AS ServerName ,

DB\_NAME() AS DB\_Name ,

Parent\_id ,

name AS TriggerName ,

create\_date

FROM sys.triggers

WHERE parent\_class = 1

ORDER BY name;

--OR

-- Додаткова інформація по тригерах

SELECT @@Servername AS ServerName ,

DB\_NAME() AS DB\_Name ,

OBJECT\_NAME(Parent\_object\_id) AS TableName ,

o.name AS 'TriggerName' ,

o.Type ,

o.create\_date ,

sm.[DEFINITION] AS 'Trigger script'

FROM sys.objects o

INNER JOIN sys.sql\_modules sm ON o.object\_id = sm.OBJECT\_ID

WHERE o.Type = 'TR' -- Triggers

ORDER BY o.NAME;

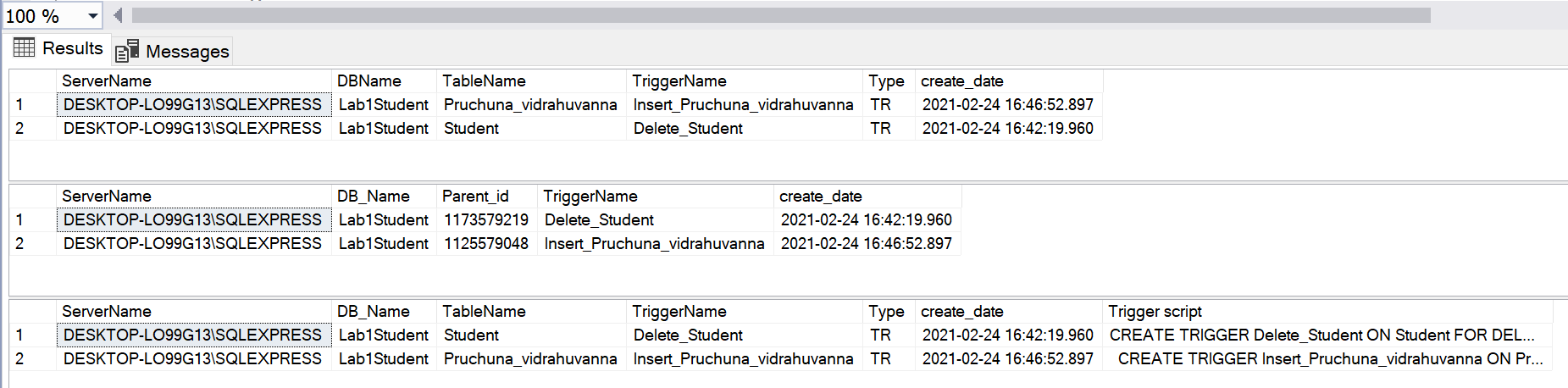


Рис. 1.14 Результат виконання запиту.

## CHECK-обмеження.

SELECT @@Servername AS ServerName ,

DB\_NAME() AS DBName ,

parent.name AS 'TableName' ,

o.name AS 'Constraints' ,

o.[Type] ,

o.create\_date

FROM sys.objects o

INNER JOIN sys.objects parent

ON o.parent\_object\_id = parent.object\_id

WHERE o.Type = 'C' -- Check Constraints

ORDER BY parent.name ,

o.name

--OR

--CHECK constriant definitions

SELECT @@Servername AS ServerName ,

DB\_NAME() AS DBName ,

OBJECT\_SCHEMA\_NAME(parent\_object\_id) AS SchemaName ,

OBJECT\_NAME(parent\_object\_id) AS TableName ,

parent\_column\_id AS Column\_NBR ,

Name AS CheckConstraintName ,

type ,

type\_desc ,

create\_date ,

OBJECT\_DEFINITION(object\_id) AS CheckConstraintDefinition

FROM sys.Check\_constraints

ORDER BY TableName ,

SchemaName ,

Column\_NBR

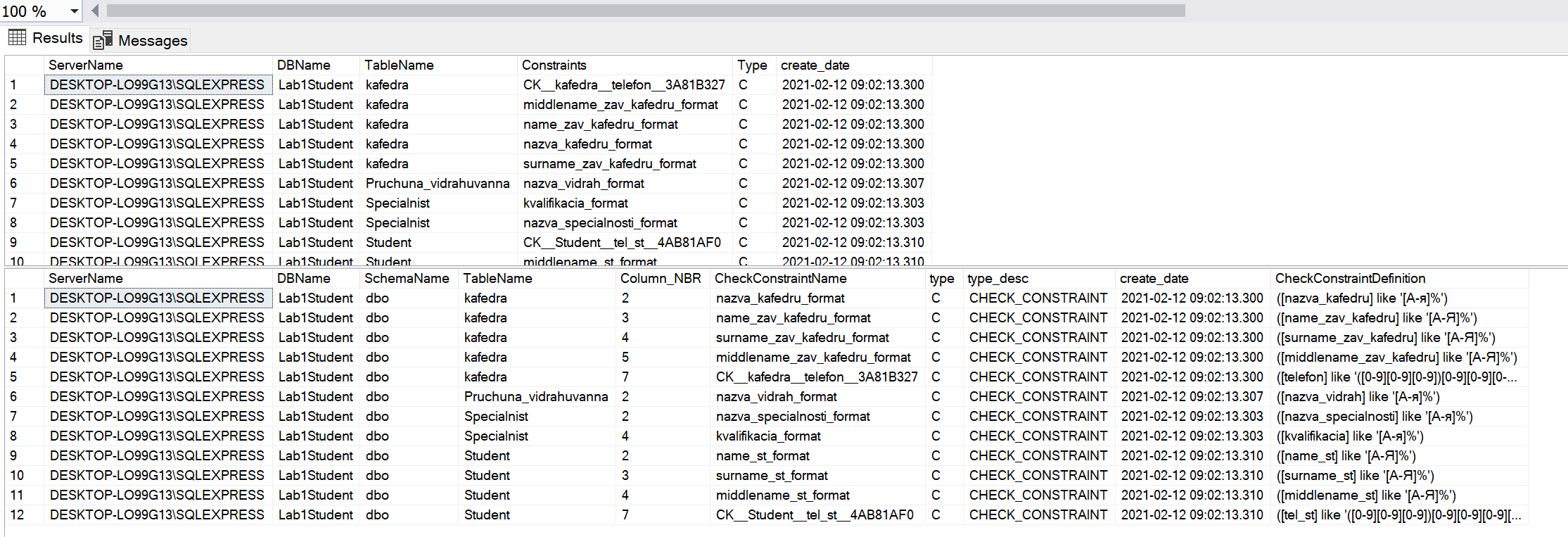


Рис. 1.15 Результат виконання запиту.

## Модель даних.

SELECT @@Servername AS Server ,

DB\_NAME() AS DBName ,

isc.Table\_Name AS TableName ,

isc.Table\_Schema AS SchemaName ,

Ordinal\_Position AS Ord ,

Column\_Name ,

Data\_Type ,

Numeric\_Precision AS Prec ,

Numeric\_Scale AS Scale ,

Character\_Maximum\_Length AS LEN , -- -1 means MAX like Varchar(MAX)

Is\_Nullable ,

Column\_Default ,

Table\_Type

FROM INFORMATION\_SCHEMA.COLUMNS isc

INNER JOIN information\_schema.tables ist

ON isc.table\_name = ist.table\_name

-- WHERE Table\_Type = 'BASE TABLE' -- 'Base Table' or 'View'

ORDER BY DBName ,

TableName ,

SchemaName ,

Ordinal\_position;

-- Назви стовпців та кількість повторів

-- Використовується для пошуку одноіменних стовпців з різними типами даних

SELECT @@Servername AS Server ,

DB\_NAME() AS DBName ,

Column\_Name ,

Data\_Type ,

Numeric\_Precision AS Prec ,

Numeric\_Scale AS Scale ,

Character\_Maximum\_Length ,

COUNT(\*) AS Count

FROM information\_schema.columns isc

INNER JOIN information\_schema.tables ist

ON isc.table\_name = ist.table\_name

WHERE Table\_type = 'BASE TABLE'

GROUP BY Column\_Name ,

Data\_Type ,

Numeric\_Precision ,

Numeric\_Scale ,

Character\_Maximum\_Length;

--Інформація по типам даних, що використовуються

SELECT @@Servername AS ServerName ,

DB\_NAME() AS DBName ,

Data\_Type ,

Numeric\_Precision AS Prec ,

Numeric\_Scale AS Scale ,

Character\_Maximum\_Length AS [Length] ,

COUNT(\*) AS COUNT

FROM information\_schema.columns isc

INNER JOIN information\_schema.tables ist

ON isc.table\_name = ist.table\_name

WHERE Table\_type = 'BASE TABLE'

GROUP BY Data\_Type ,

Numeric\_Precision ,

Numeric\_Scale ,

Character\_Maximum\_Length

ORDER BY Data\_Type ,

Numeric\_Precision ,

Numeric\_Scale ,

Character\_Maximum\_Length

-- Пам'ятайте, що індекси по цих таблиць не можуть бути перебудовані в режимі "online"

SELECT @@Servername AS ServerName ,

DB\_NAME() AS DBName ,

isc.Table\_Name ,

Ordinal\_Position AS Ord ,

Column\_Name ,

Data\_Type AS BLOB\_Data\_Type ,

Numeric\_Precision AS Prec ,

Numeric\_Scale AS Scale ,

Character\_Maximum\_Length AS [Length]

FROM information\_schema.columns isc

INNER JOIN information\_schema.tables ist

ON isc.table\_name = ist.table\_name

WHERE Table\_type = 'BASE TABLE'

AND ( Data\_Type IN ( 'text', 'ntext', 'image', 'XML' )

OR ( Data\_Type IN ( 'varchar', 'nvarchar', 'varbinary' )

AND Character\_Maximum\_Length = -1

)

) -- varchar(max), nvarchar(max), varbinary(max)

ORDER BY isc.Table\_Name ,

Ordinal\_position;

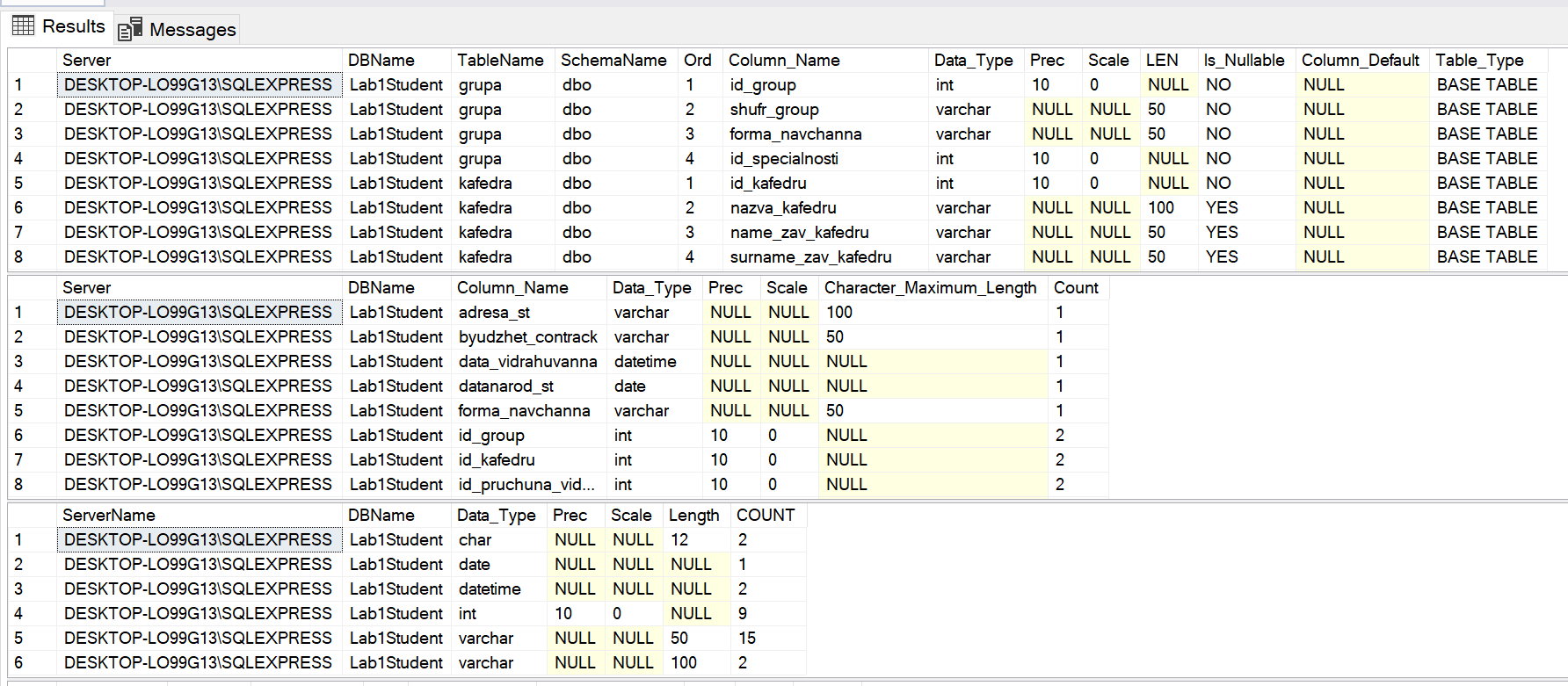


Рис. 1.16 Результат виконання запиту.

SELECT @@Servername AS ServerName ,

DB\_NAME() AS DBName ,

OBJECT\_SCHEMA\_NAME(object\_id) AS SchemaName ,

OBJECT\_NAME(object\_id) AS Tablename ,

Column\_id ,

Name AS Computed\_Column ,

[Definition] ,

is\_persisted

FROM sys.computed\_columns

ORDER BY SchemaName ,

Tablename ,

[Definition];

--Or

-- Computed Columns

SELECT @@Servername AS ServerName ,

DB\_NAME() AS DBName ,

OBJECT\_SCHEMA\_NAME(t.object\_id) AS SchemaName,

t.Name AS TableName ,

c.Column\_ID AS Ord ,

c.Name AS Computed\_Column

FROM sys.Tables t

INNER JOIN sys.Columns c ON t.object\_id = c.object\_id

WHERE is\_computed = 1

ORDER BY t.Name ,

SchemaName ,

c.Column\_ID

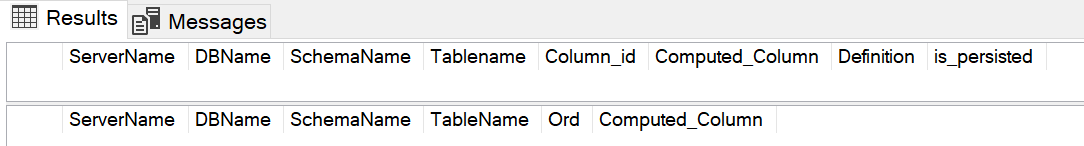


Рис. 1.17 Результат виконання запиту.

**Висновок**: в ході виконання лабороторної роботи, навчився досліджувати БД за допомогою T-SQL.