Міністерство освіти і науки України

Національний технічний університет України

“Київський політехнічний інститут”

Кафедра АСОІУ

**ЗВІТ**

про виконання комп’ютерного практикуму № 3

з дисципліни

“ Програмні додатки з використанням баз даних”

по темі

«Ієрархічні запити»

Варіант 6

|  |  |  |
| --- | --- | --- |
| Прийняв: |  | Виконав: |
| Cт.вик. Клименко О. М. |  | студент 3-го курсу  гр. ІП-51 ФІОТ  Зарічковий Олександр Анатолійович |

Київ – 2017

**ЗМІСТ:**

[1 Завдання 3](#_Toc495230724)

[2 Виконання Завдання 4](#_Toc495230725)

# Завдання

Завдання 1:

Для того, щоб можна було попрактикуватися в виконанні ієрархічних запитів в базу даних згідно варіанту необхідно додати таблицю та cтворити primary key та 2 foreign keys. Зразки структури додаткової таблиці:

6 варіант. Таблиця «Слухачі-куратори» має поля:

* id запису,
* код слухача («нащадок»),
* код слухача-куратора («предок»).

Завдання 2:

Для своєї предметної області написати наступні ієрархічні запити: 1) Вивести список всіх «нащадків» вказаного «предка». 2) Вивести список всіх «предків» вказаного «нащадка». 3) Вивести список, другий полем якого є «рівень» ( аналог псевдостовпчика level в connect by). 4) ( 2 запити ) Змінити дані в доданій таблиці так, щоб утворився цикл. Написати запит, що видає помилку при зациклюванні. Змінити цей запит так, щоб помилки не було. 5) Для всіх «нащадків» (це перше поле: Іванов ) вивести список «предків» через «/», де останнім в ланцюгу є цей «нащадок» ( це друге поле: Іваненко/Іванченко/Іванчук/Іванов)

**Преметна область**: Навчання з охорони праці

# Виконання Завдання

1. Завдання по написанню запитів
   1. Вивести список всіх «нащадків» вказаного «предка».  
      *Призначення*: Для заданого куратора рекурсивно вивести всіх його слухачів.

*Текст запиту*:

CREATE PROCEDURE CURATORS\_LITENERS

@curator\_id INT

AS

WITH sub(listener\_id) AS

(SELECT ListnerId

FROM [dbo].[Curators]

WHERE CuratorsId = @curator\_id

UNION ALL

SELECT [C].ListnerId

FROM sub

INNER JOIN [Curators] AS [C] ON

[C].[CuratorsId] = [sub].[listener\_id]

)

SELECT distinct listener\_id

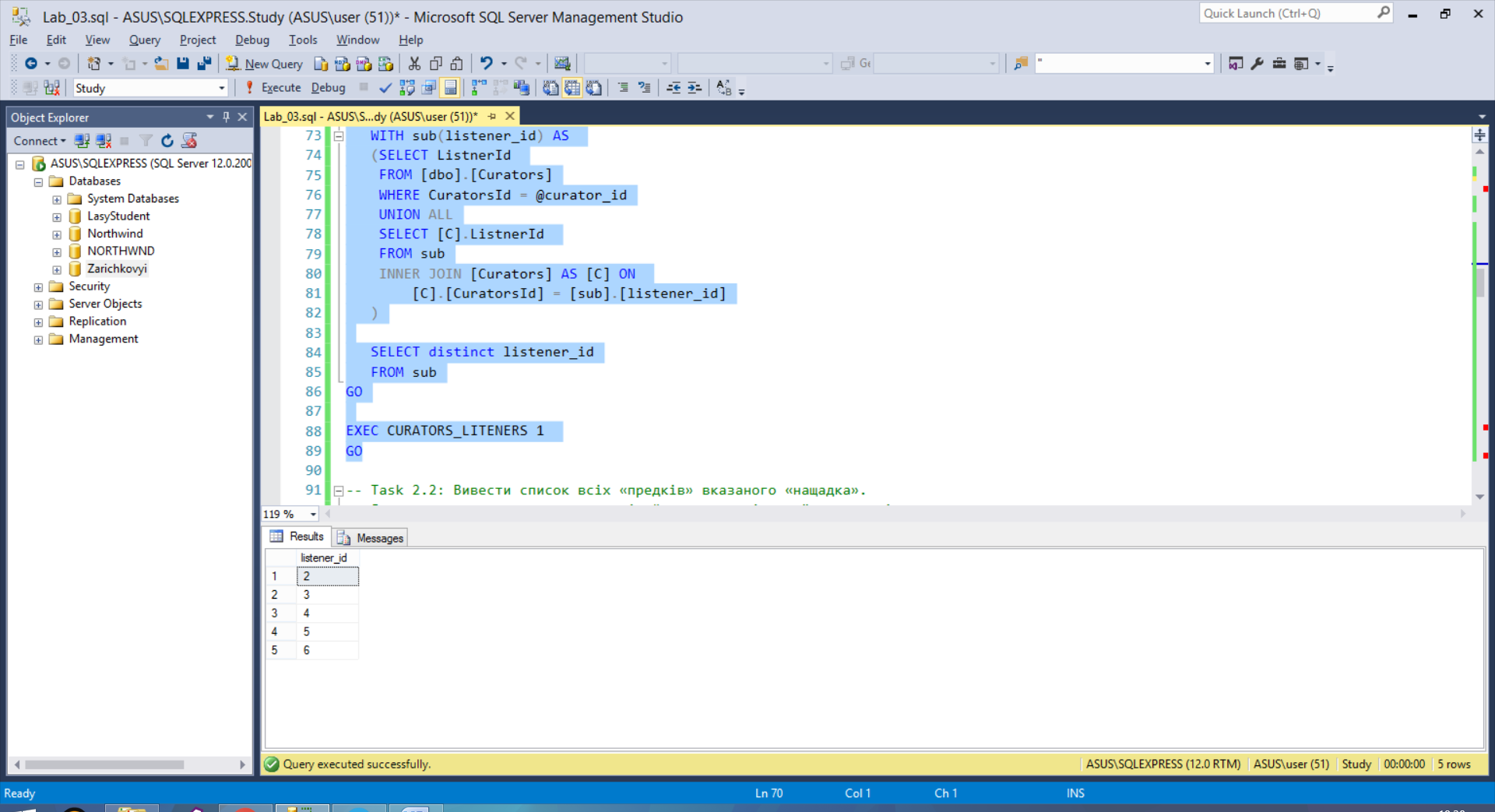
FROM sub

GO

EXEC CURATORS\_LITENERS 1

GO

*Результат виконання з використання аналітичних функцій*:



* 1. Вивести список всіх «предків» вказаного «нащадка»:

*Призначення*: Для заданого слухача рекурсивно вивести усіх його кураторів та їх кураторів.

*Текст запиту*:

CREATE PROCEDURE LITENERS\_CURATORS

@listner\_id INT

AS

WITH sub(curator\_id) AS

(SELECT CuratorsId

FROM [dbo].[Curators]

WHERE [ListnerId] = @listner\_id

UNION ALL

SELECT [C].CuratorsId

FROM sub

INNER JOIN [Curators] AS [C] ON

[C].[ListnerId] = [sub].[curator\_id]

)

SELECT distinct curator\_id

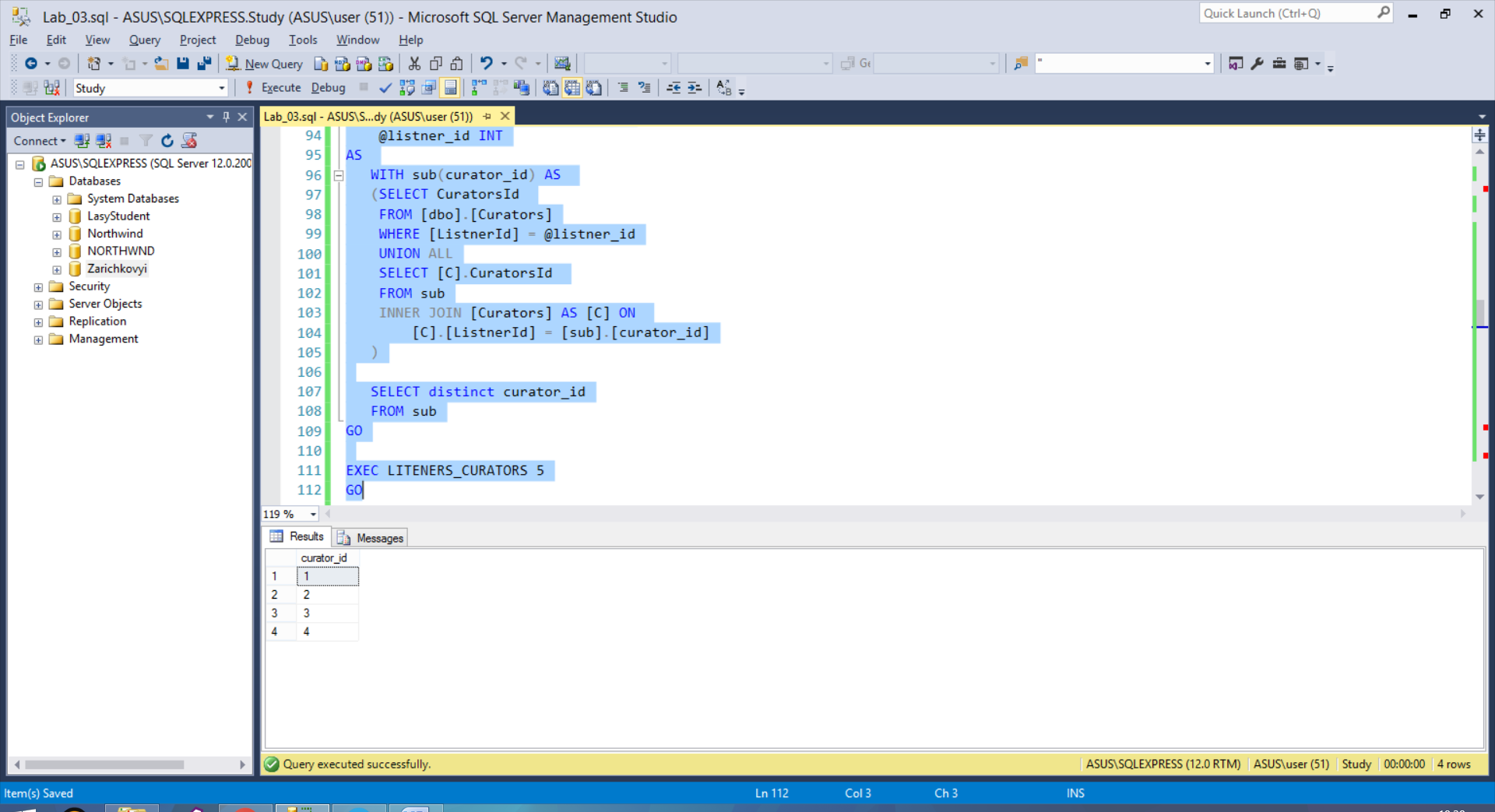
FROM sub

GO

EXEC LITENERS\_CURATORS 5

GO

*Результат виконання*:



* 1. Вивести список, другий полем якого є «рівень» ( аналог псевдо стовпчика level в connect by).

*Призначення*: Для заданого куратора вивести усіх його слухачів та рівень слухача в ієрархії слухачів

*Текст запиту з використанням аналітичних функцій*:

CREATE PROCEDURE LISTNERS\_LEVEL

@curator\_id INT

AS

WITH sub(listener\_id, listner\_level) AS

(SELECT ListnerId, 1

FROM [dbo].[Curators]

WHERE CuratorsId = @curator\_id

UNION ALL

SELECT [C].ListnerId , [sub].[listner\_level] + 1

FROM sub

INNER JOIN [Curators] AS [C] ON

[C].[CuratorsId] = [sub].[listener\_id]

)

SELECT listener\_id, MIN(listner\_level) as level

FROM sub

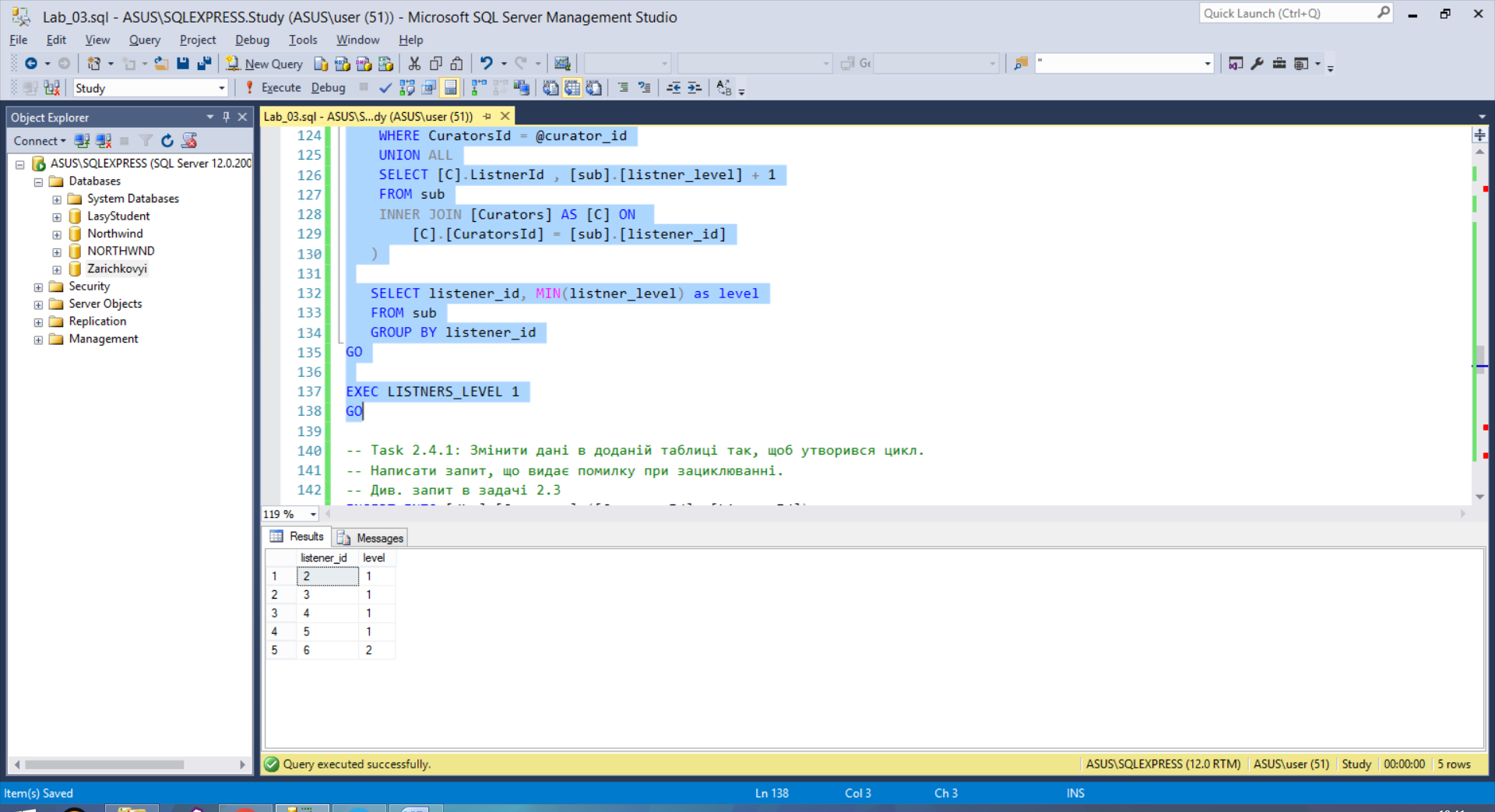
GROUP BY listener\_id

GO

EXEC LISTNERS\_LEVEL 1

GO

*Результат виконання*:



* 1. ( 2 запити ) Змінити дані в доданій таблиці так, щоб утворився цикл. Написати запит, що видає помилку при зациклюванні. Змінити цей запит так, щоб помилки не було.

*Призначення*: Для кожного слухача та кожного придбаного курсу підрахувати сердні розходи на курси (хронологічно середнє курсів придбаних починаючи з поточного курсу).

*Текст запиту з циклом*:

INSERT INTO [dbo].[Curators] ([CuratorsId], [ListnerId])

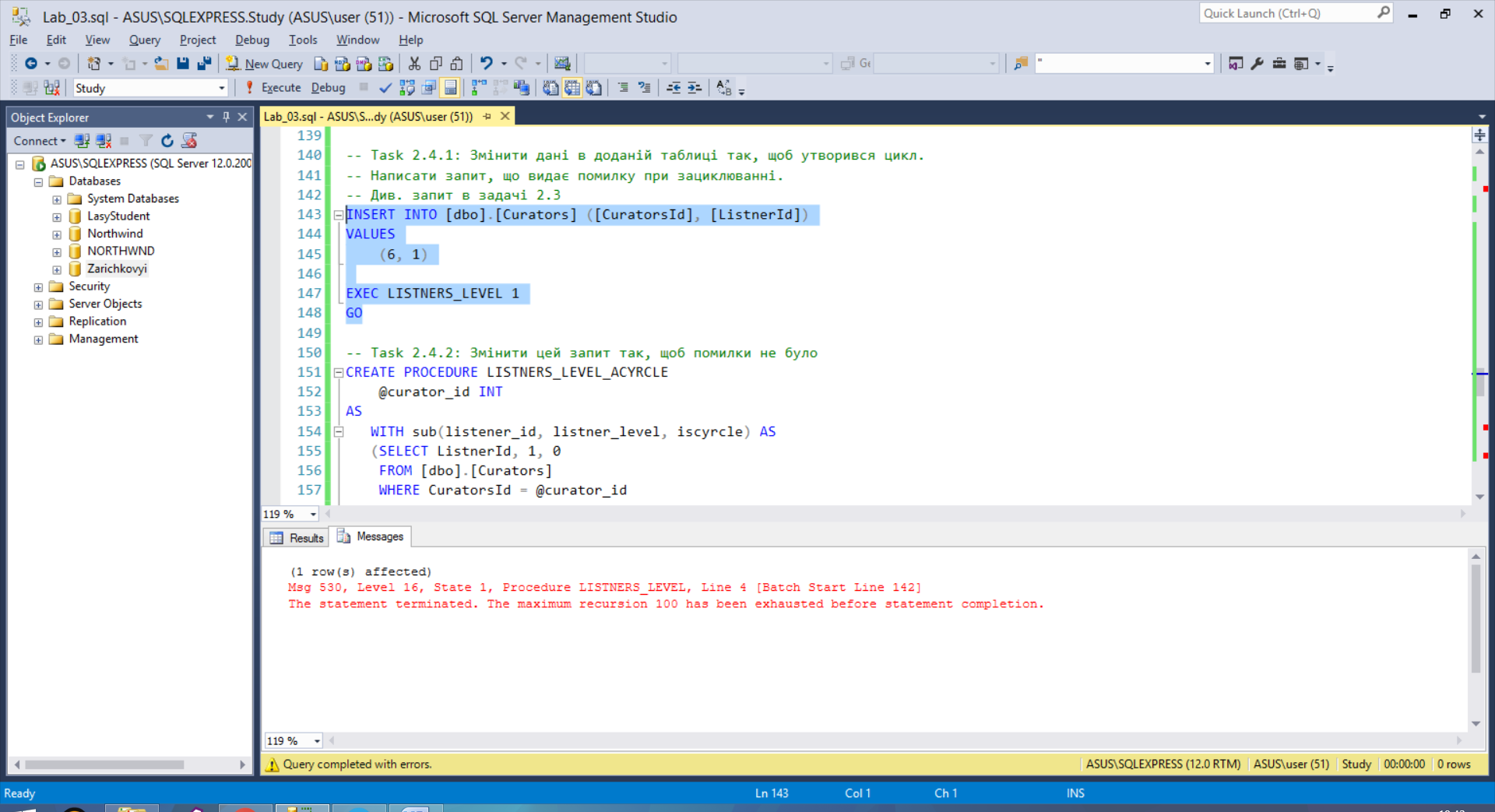
VALUES

(6, 1)

EXEC LISTNERS\_LEVEL 1

GO

*Результат виконання з використання аналітичних функцій*:



*Текст запиту з обробкою циклу*:

CREATE PROCEDURE LISTNERS\_LEVEL\_ACYRCLE

@curator\_id INT

AS

WITH sub(listener\_id, listner\_level, iscyrcle) AS

(SELECT ListnerId, 1, 0

FROM [dbo].[Curators]

WHERE CuratorsId = @curator\_id

UNION ALL

SELECT [C].ListnerId , [sub].[listner\_level] + 1,

(CASE WHEN sub.[listener\_id] = [C].[CuratorsId] THEN 1 ELSE 0 END) as iscycle

FROM sub

INNER JOIN [Curators] AS [C] ON

[C].[CuratorsId] = [sub].[listener\_id]

WHERE [sub].iscyrcle = 0

)

SELECT listener\_id, MIN(listner\_level) as level

FROM sub

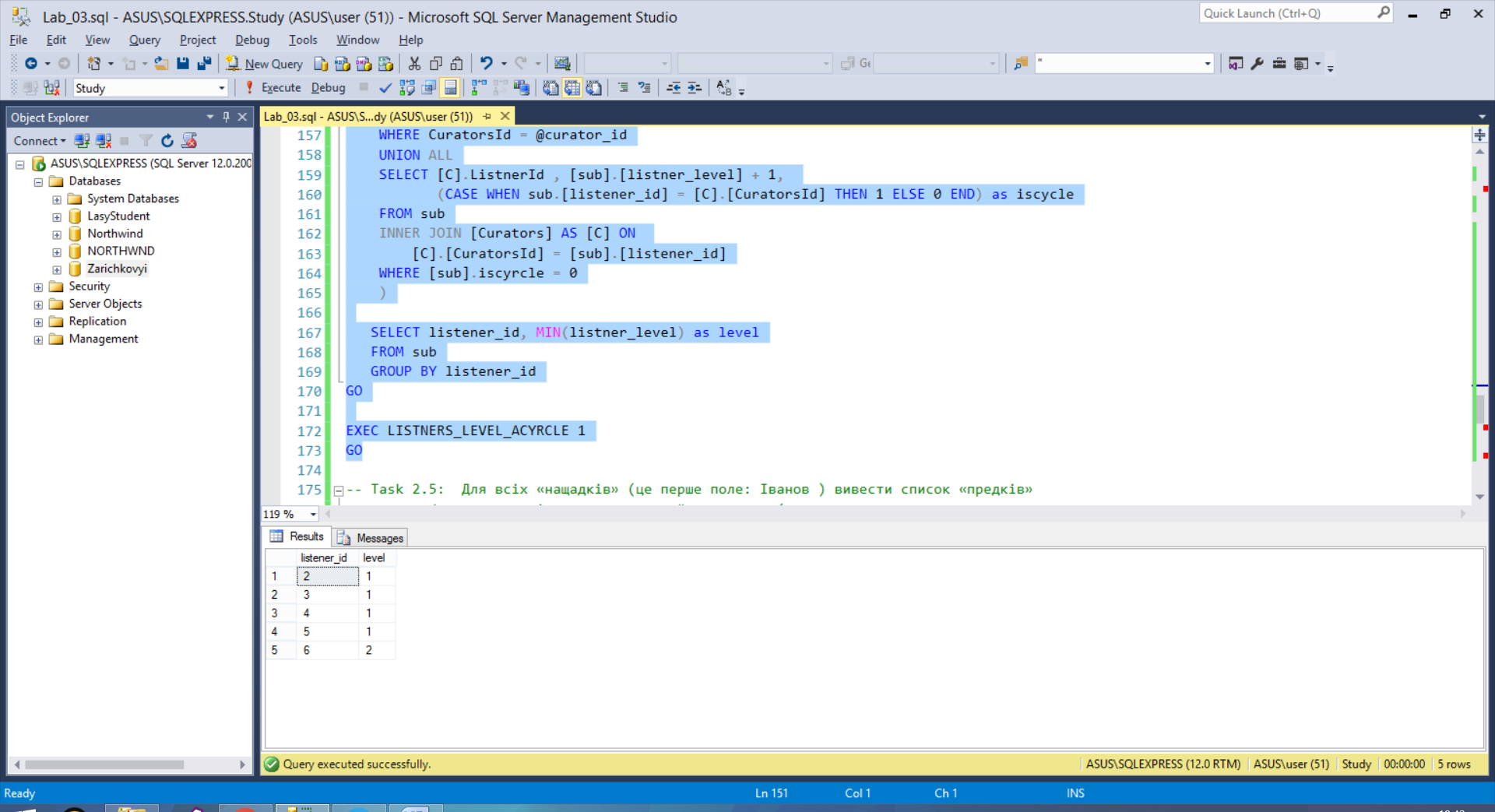
GROUP BY listener\_id

GO

EXEC LISTNERS\_LEVEL\_ACYRCLE 1

GO

*Результат виконання*:



* 1. Для всіх «нащадків» (це перше поле: Іванов ) вивести список «предків» через «/», де останнім в ланцюгу є цей «нащадок» ( це друге поле: Іваненко/Іванченко/Іванчук/Іванов):

*Призначення*: Вивести ієрахію слухач-куратор.

*Текст запиту з використанням аналітичних функцій*:

CREATE PROCEDURE LISTNER\_LEVEL

AS

WITH sub(listener\_id, lister\_path, iscyrcle) AS

(SELECT [CuratorsId], CAST([CuratorsId] AS NVARCHAR(256)), 0

FROM [dbo].[Curators]

UNION ALL

SELECT [C].ListnerId ,

CAST(CONCAT(sub.lister\_path, '/', CAST([C].ListnerId AS NVARCHAR(256))) AS NVARCHAR(256)),

(CASE WHEN sub.[listener\_id] = [C].[CuratorsId] THEN 1 ELSE 0 END) as iscycle

FROM sub

INNER JOIN [Curators] AS [C] ON

[C].[CuratorsId] = [sub].[listener\_id]

WHERE [sub].iscyrcle = 0

)

SELECT DISTINCT listener\_id, lister\_path

FROM sub

GO

EXEC LISTNER\_LEVEL

GO

*Результат виконання*:

