Міністерство освіти і науки України

Національний технічний університет України

“Київський політехнічний інститут”

Кафедра АСОІУ

**ЗВІТ**

про виконання комп’ютерного практикуму № 2

з дисципліни

“ Програмні додатки з використанням баз даних”

по темі

«Ієрархічні запити.»

Варіант 1

|  |  |  |
| --- | --- | --- |
| Прийняв: |  | Виконав: |
| Cт.вик. Клименко О. М. |  | студент 3-го курсу  гр. ІП-51 ФІОТ  Булатов Дмитро Єгорович |

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:

1 варіант.

Таблиця «Режиссери-наставники» має поля:

id запису,

код режиссера («нащадок»),

код режиссера-наставника («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)

# Виконання Завдання

Завдання 1

додати таблицю та cтворити primary key та 2 foreign keys. Зразки структури додаткової таблиці:

1 варіант.

Таблиця «Режиссери-наставники» має поля:

id запису,

код режиссера («нащадок»),

код режиссера-наставника («предок»).

CREATE DATABASE LR3

GO

USE LR3

GO

CREATE TABLE Directors(

ID int PRIMARY KEY

)

CREATE TABLE Director\_mentors(

ID int PRIMARY KEY,

mentor\_id int FOREIGN KEY REFERENCES Directors(ID), -- наставник

mentee\_id int FOREIGN KEY REFERENCES Directors(ID), -- ученик

)

-- task 1

INSERT INTO Directors VALUES

(1, 'Gendalf'),

(2, 'Sauron'),

(3, 'R2D2'),

(4, 'Leonardo Di Caprio'),

(5, 'Luke Skywalker'),

(6, 'Filipp J. Fry'),

(7, 'Van Gogh'),

(8, 'Obi Wan Kenobi'),

(9, 'Mukha'),

(10, 'Superman')

INSERT INTO Director\_mentors VALUES

(1, NULL, 3 ),

(2, NULL, 5 ),

(3, 1 , 6 ),

(4, 1 , 9 ),

(5, 2 , NULL),

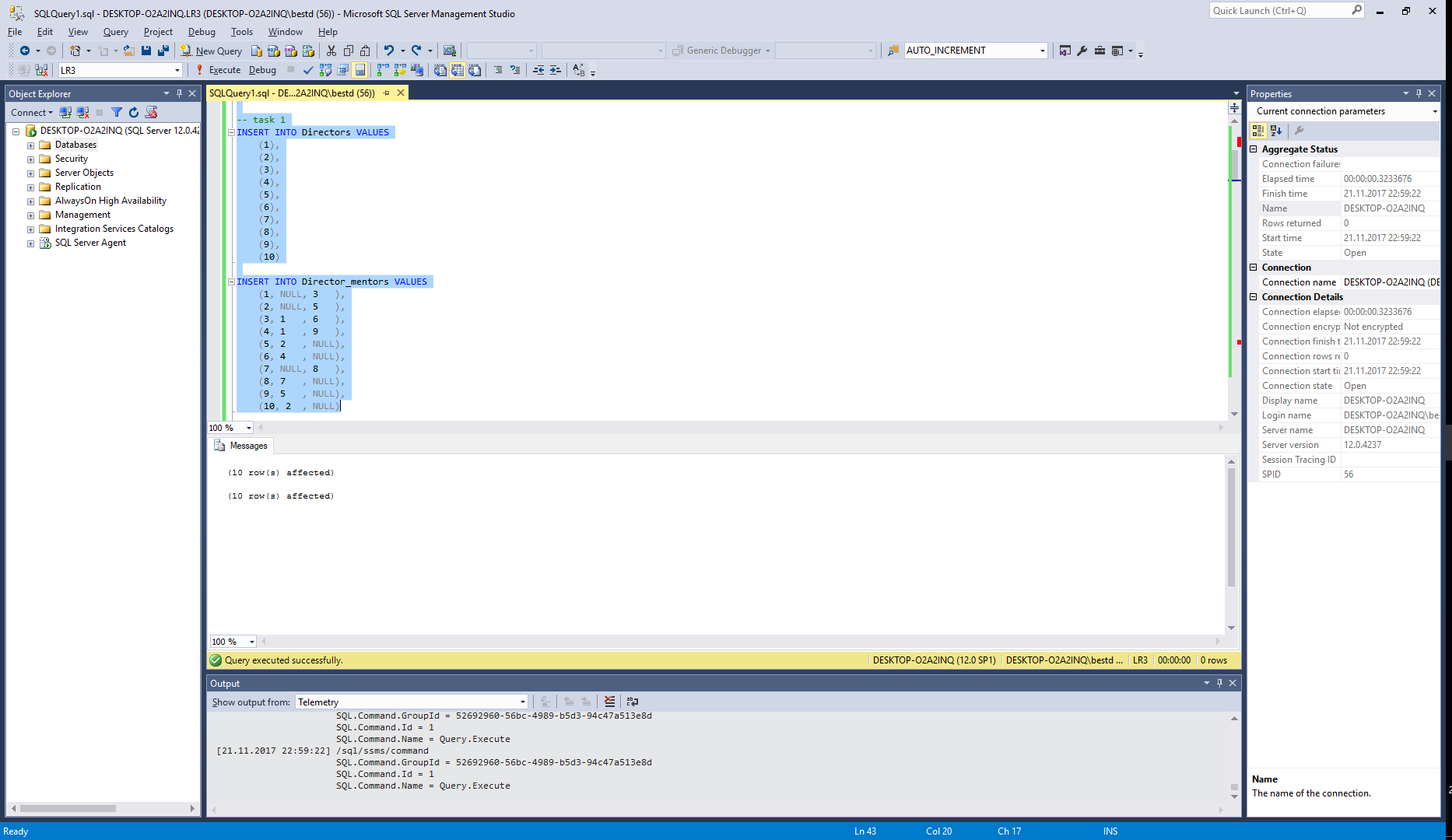
(6, 4 , NULL),

(7, NULL, 8 ),

(8, 7 , NULL),

(9, 5 , NULL),

(10, 2 , NULL)

*Результат виконання програми*: 

Завдання 2

1. Вивести список всіх «нащадків» вказаного «предка».

GO

CREATE PROCEDURE listMentees

@ID INT

AS

WITH subquery(mentee\_ids) AS

(

SELECT ID

FROM Director\_mentors

WHERE mentor\_id = @ID

UNION ALL

SELECT T.ID

FROM subquery

INNER JOIN Director\_mentors AS T ON

T.mentor\_id = subquery.mentee\_ids

)

SELECT distinct mentee\_ids, Directors.Name

FROM subquery

JOIN Directors ON Directors.ID = mentee\_ids

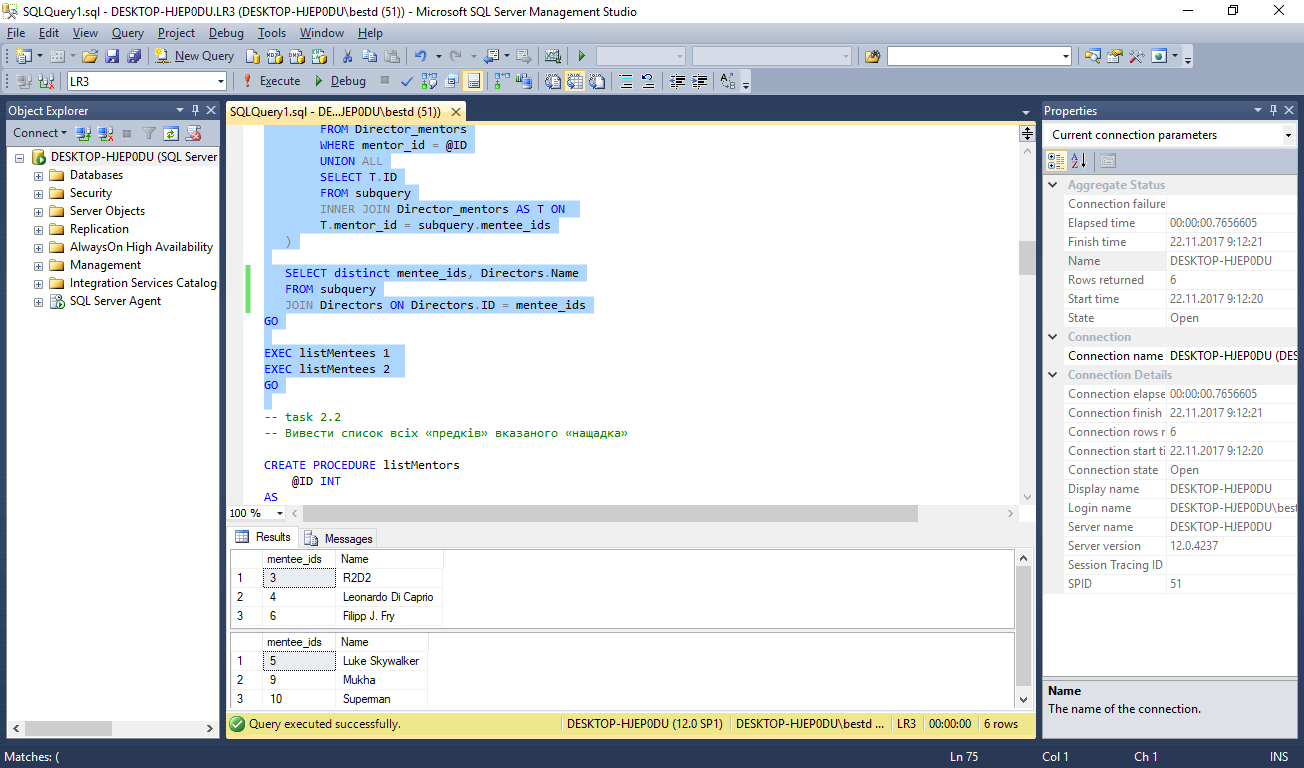
GO

EXEC listMentees 1

EXEC listMentees 2

GO

*Результат виконання програми*:



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

CREATE PROCEDURE listMentors

@ID INT

AS

WITH subquery(mentor\_ids) AS

(

SELECT mentor\_id

FROM Director\_mentors

WHERE ID = @ID

UNION ALL

SELECT T.mentor\_id

FROM subquery

INNER JOIN Director\_mentors AS T ON

T.ID = subquery.mentor\_ids

WHERE T.mentor\_id IS NOT NULL

)

SELECT distinct mentor\_ids, Directors.Name

FROM subquery

JOIN Directors ON Directors.ID = mentor\_ids

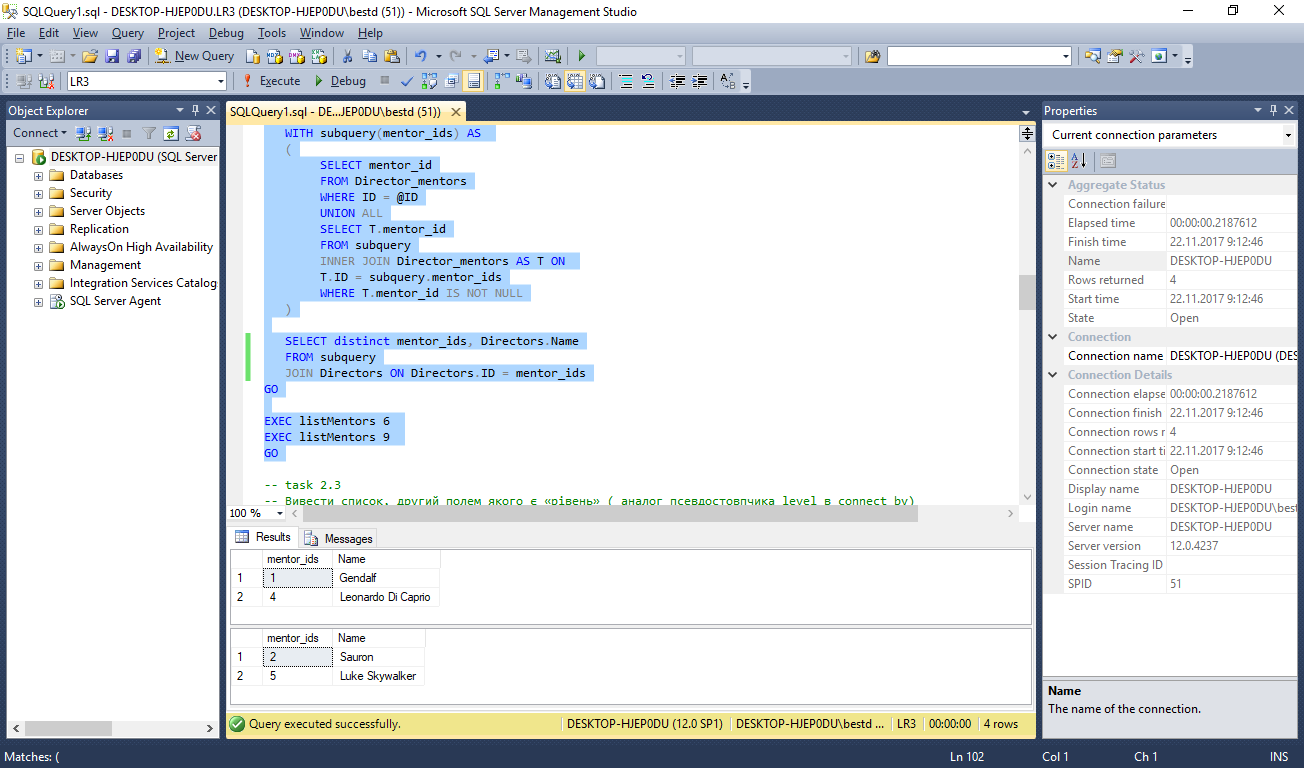
GO

EXEC listMentors 6

EXEC listMentors 9

GO

*Результат виконання програми*:



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

CREATE PROCEDURE listMenteesWithLevel

@ID INT

AS

WITH subquery(mentee\_ids,lvl) AS

(

SELECT ID, 1

FROM Director\_mentors

WHERE mentor\_id = @ID

UNION ALL

SELECT T.ID, lvl+1

FROM subquery

INNER JOIN Director\_mentors AS T ON

T.mentor\_id = subquery.mentee\_ids

)

SELECT distinct mentee\_ids, lvl, Directors.Name

FROM subquery

JOIN Directors ON Directors.ID = mentee\_ids

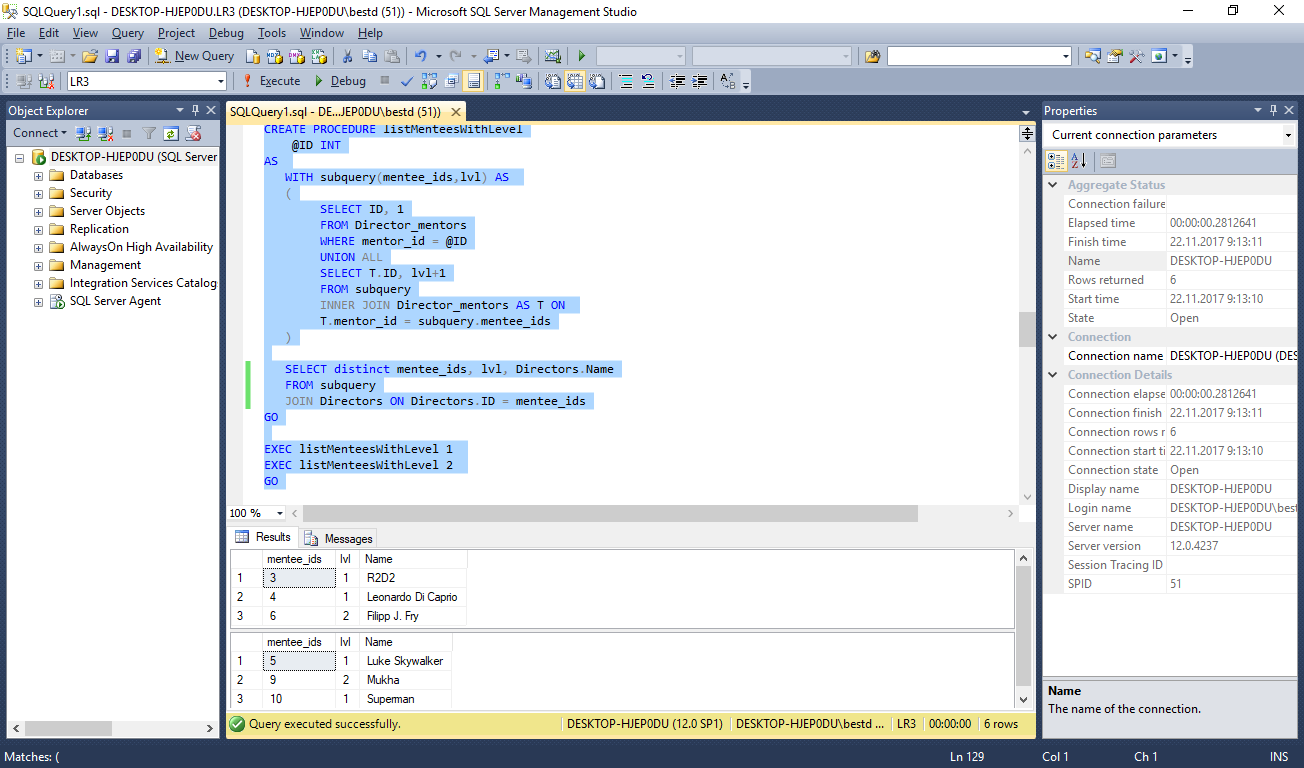
GO

EXEC listMenteesWithLevel 1

EXEC listMenteesWithLevel 2

GO

*Результат виконання програми*:



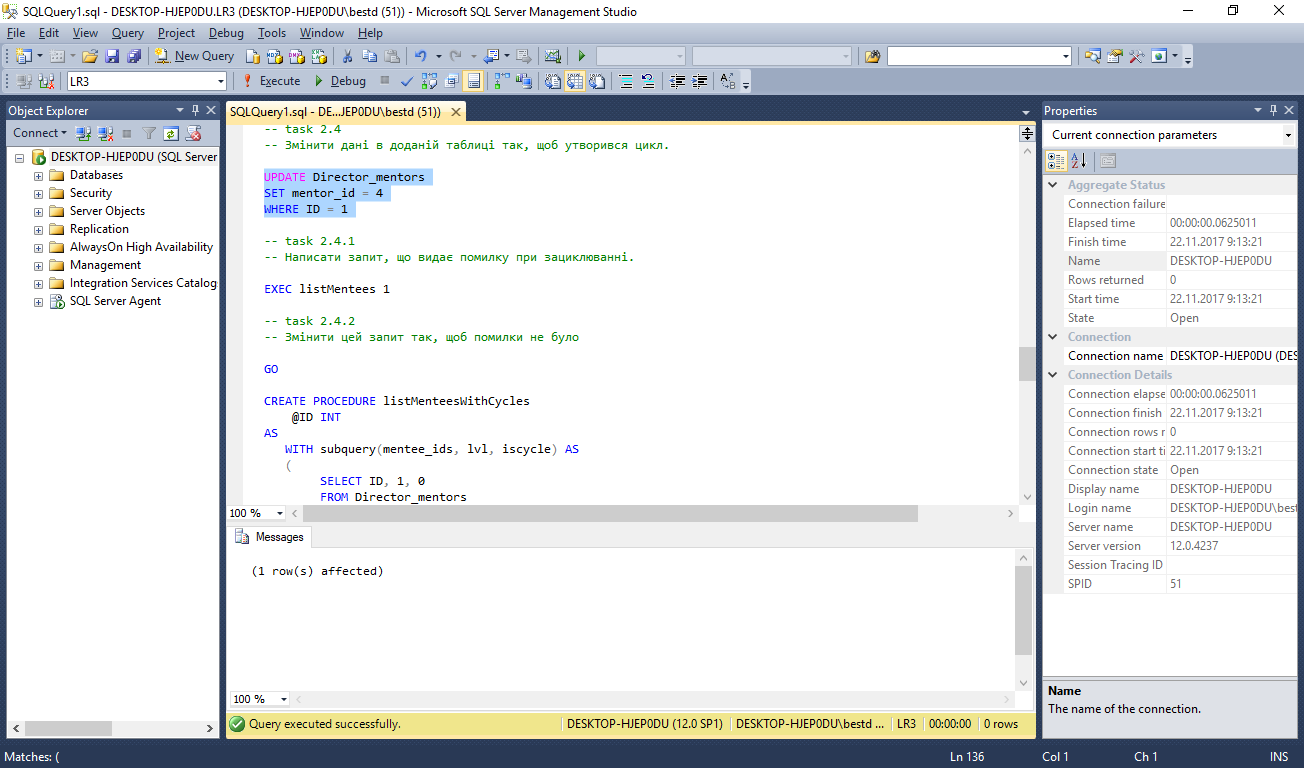
4) Змінити дані в доданій таблиці так, щоб утворився цикл.

UPDATE Director\_mentors

SET mentor\_id = 4

WHERE ID = 1

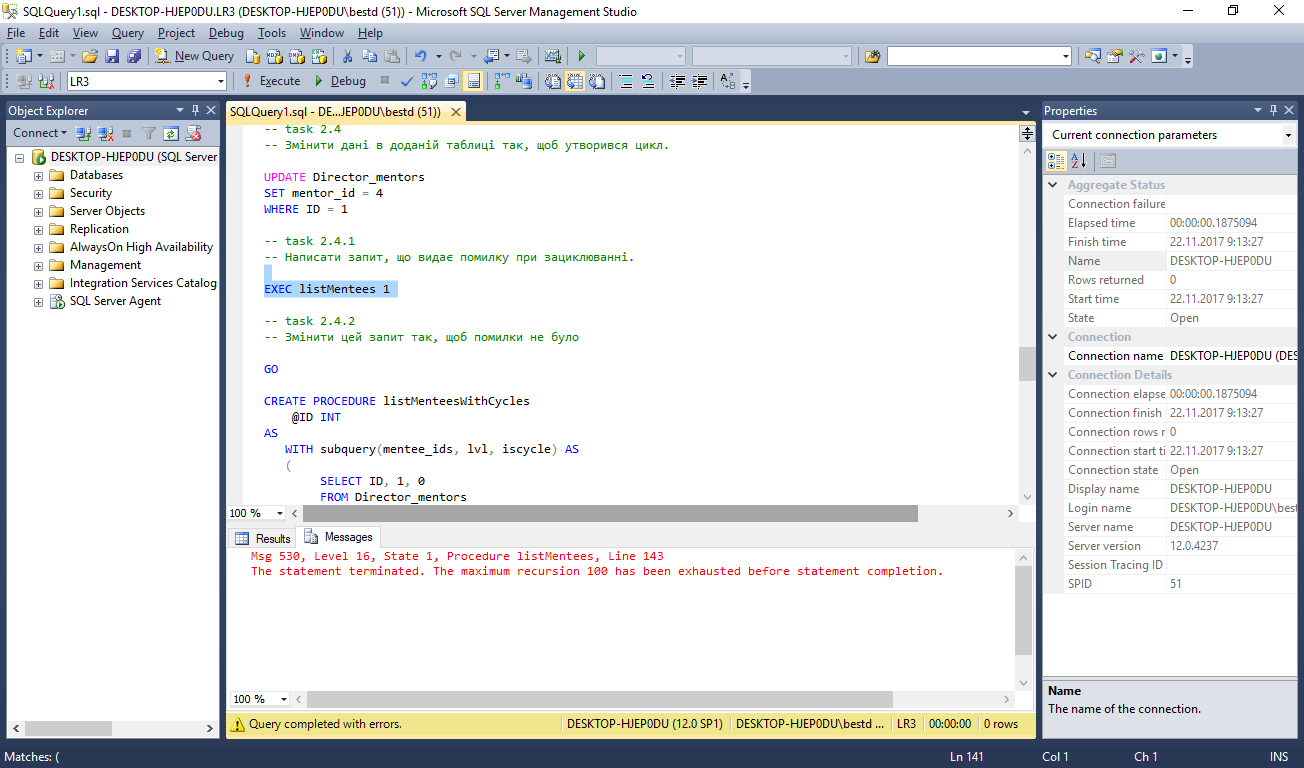
*Результат виконання програми*:



4.1) Написати запит, що видає помилку при зациклюванні.

EXEC listMentees 1

*Результат виконання програми*:



4.2) Змінити цей запит так, щоб помилки не було

GO

CREATE PROCEDURE listMenteesWithCycles

@ID INT

AS

WITH subquery(mentee\_ids, lvl, iscycle) AS

(

SELECT ID, 1, 0

FROM Director\_mentors

WHERE mentor\_id = @ID

UNION ALL

SELECT T.ID, lvl+1, (CASE WHEN T.mentor\_id = subquery.mentee\_ids THEN 1 ELSE 0 END)

FROM subquery

INNER JOIN Director\_mentors AS T ON

T.mentor\_id = subquery.mentee\_ids

WHERE iscycle = 0

)

SELECT distinct mentee\_ids, lvl, Directors.Name

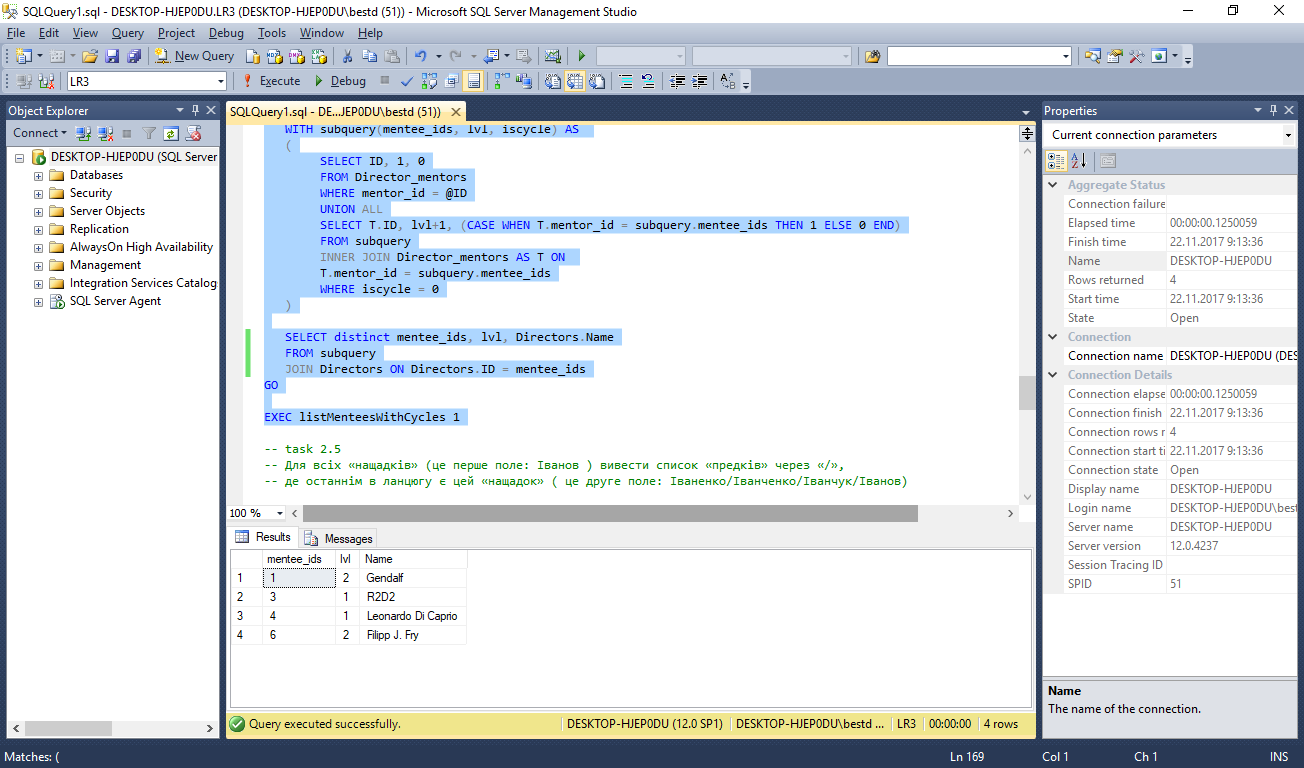
FROM subquery

JOIN Directors ON Directors.ID = mentee\_ids

GO

EXEC listMenteesWithCycles 1

*Результат виконання програми*:



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)

GO

CREATE PROCEDURE listMenteesWithHistory

@ID INT

AS

WITH subquery(mentee\_ids, history, iscycle) AS

(

SELECT Director\_mentors.ID,

CAST(CONCAT((SELECT Directors.Name FROM Directors WHERE ID = @ID),'\',CAST(Directors.Name AS nvarchar(256))) AS nvarchar(256)), 0

FROM Director\_mentors

JOIN Directors ON Directors.ID = Director\_mentors.ID

WHERE mentor\_id = @ID

UNION ALL

SELECT

T.ID,

CAST(CONCAT(subquery.history,'\',CAST(Directors.Name AS nvarchar(256))) AS nvarchar(256)),

(CASE WHEN T.mentor\_id = subquery.mentee\_ids THEN 1 ELSE 0 END)

FROM subquery

INNER JOIN Director\_mentors AS T ON

T.mentor\_id = subquery.mentee\_ids

JOIN Directors ON Directors.ID = T.ID

WHERE iscycle = 0

)

SELECT mentee\_ids, history, Directors.Name

FROM subquery

JOIN Directors ON Directors.ID = mentee\_ids

GO

EXEC listMenteesWithHistory 1

*Результат виконання програми*:

