|  |  |
| --- | --- |
| **лого для документов 2022** | ***Федеральное агентство по рыболовству***  ***Федеральное государственное бюджетное образовательное***  ***учреждение высшего образования***  ***«Астраханский государственный технический университет»***  **Система менеджмента качества в области образования, воспитания, науки и инноваций**  **ООО «ДКС РУС» по международному стандарту ISO 9001:2015** |

ИНСТИТУТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И КОММУНИКАЦИЙ

**КАФЕДРА АВТОМАТИЗИРОВАННЫХ СИСТЕМ ОБРАБОТКИ ИНФОРМАЦИИ И УПРАВЛЕНИЯ**

**БАЗЫ ДАННЫХ**

(*наименование дисциплины*)

**ОТЧЕТ**

о выполнении заданий к лабораторной работе № 16

**Пользовательские функции**

|  |  |
| --- | --- |
|  | Выполнила:  студент гр. ДИНРБ-31 |
|  | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Кузургалиев Р.А. |
|  | «\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_2023 г. |
|  | Максимальное количеству баллов \_\_\_\_\_  ЗАЩИЩЕНО:  Получено баллов \_\_\_\_\_\_ |
|  | Преподаватель: \_\_\_\_\_\_\_\_ Мамлеева А.Р. |
|  | «\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_2023 г. |

**Астрахань – 2023**

**Лабораторная работа №16 «Пользовательские функции»:**

1. Цель работы: разобраться в устройстве языка управления данных DML, а также в написании функций.

**ЗАДАНИЕ К ЛАБОРАТОРНОЙ РАБОТЕ №15**

Схема базы данных состоит из четырех отношений:

**Автор** (id\_authors int, fio\_authors varchar(20))

**Издательство** (id\_publishers int, name varchar(20))

**Книга** (id\_books int, name varchar(50), year\_out varchar(4))

**Сводная** (id\_out int, id\_books int, id\_authors int, id\_publishers int)

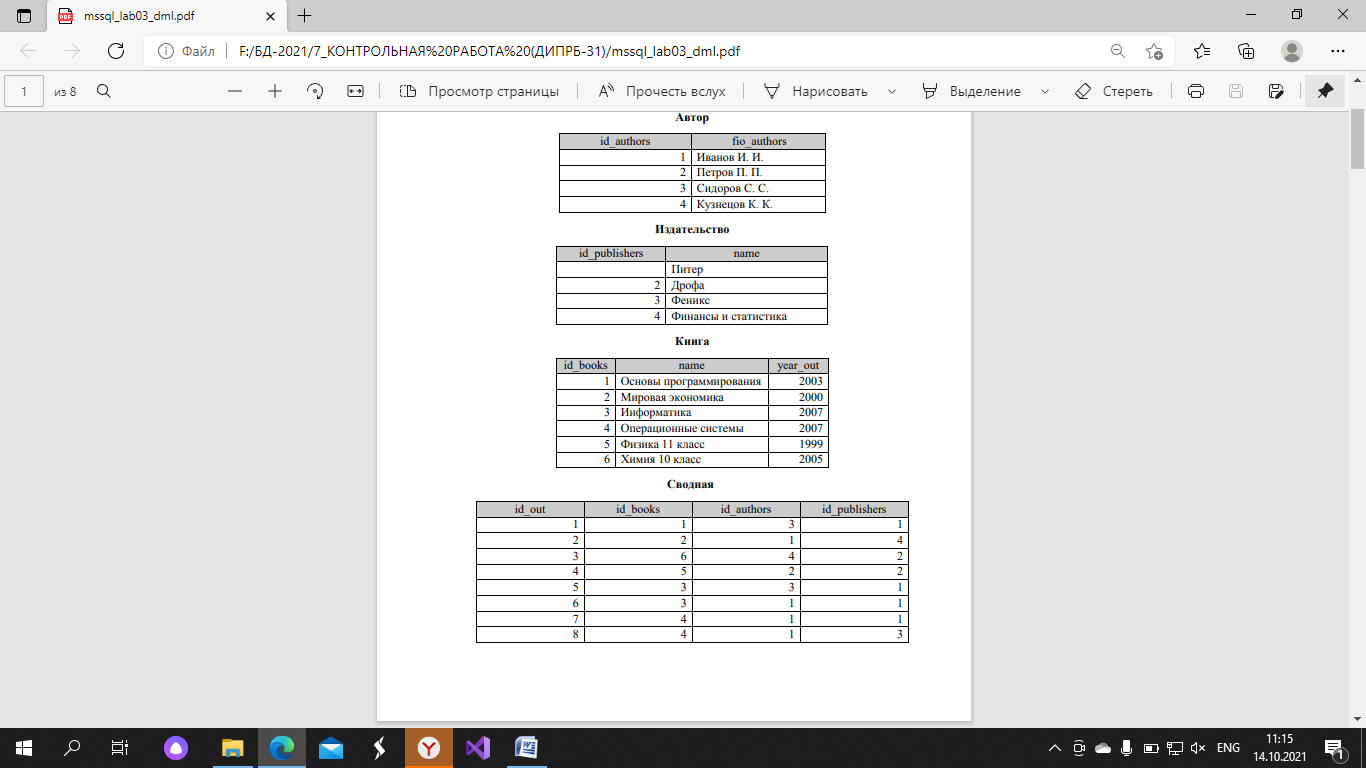


Схема базы данных представлена на рис. 1.

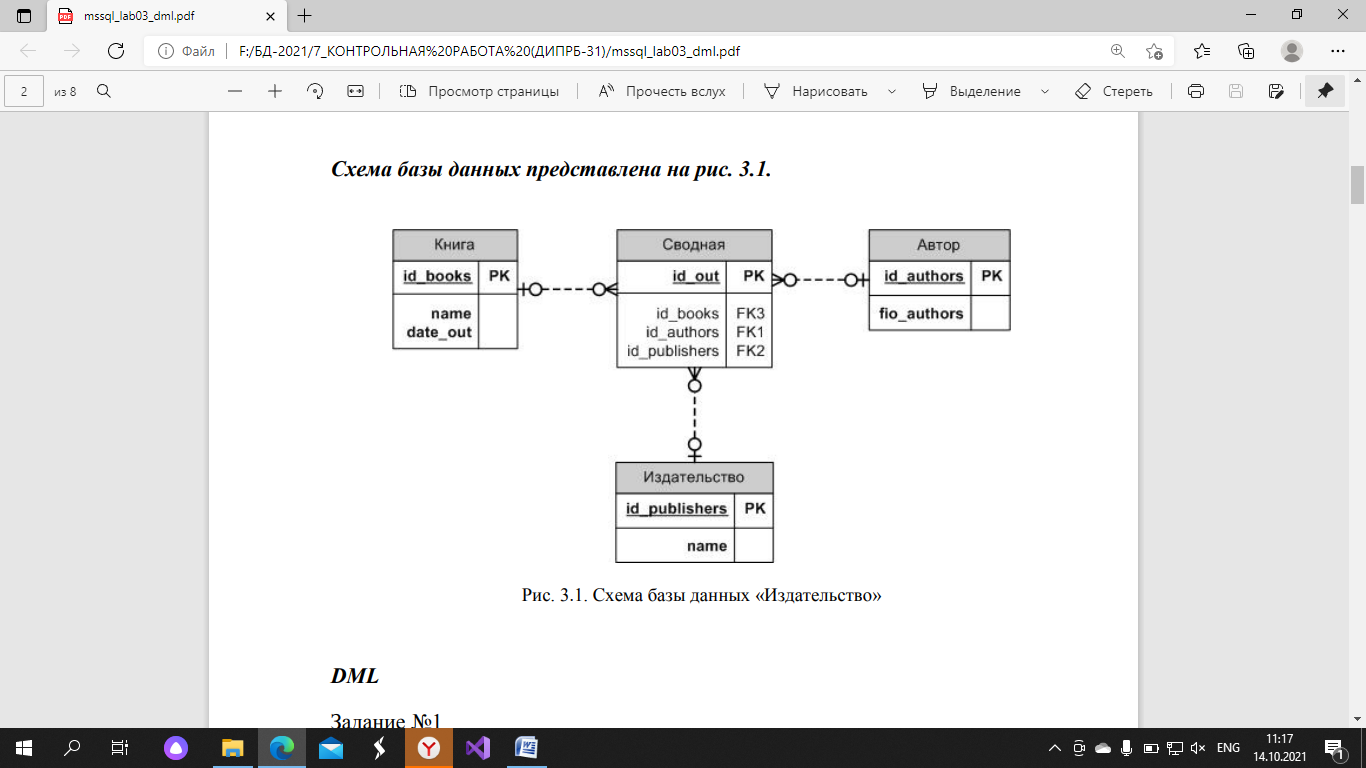


Рисунок 1 - Схема базы данных «Издательство»

Задание 1. Создать функцию, принимающую в качестве параметра фамилию автора и возвращающую количество написанных им книг. Если автор не написал ни одной книги, функция должна возвратить 0. Если автора не существует в базе данных, функция должна возвратить -1

CREATE FUNCTION count\_books\_by\_author

(

@author\_lastname varchar(20)

)

RETURNS int

AS

BEGIN

DECLARE @author\_id int

-- Получаем id автора по его фамилии

SELECT @author\_id = id\_authors

FROM Автор

WHERE fio\_authors = @author\_lastname

-- Если автор не найден, возвращаем -1

IF (@author\_id IS NULL)

RETURN -1

-- Иначе, считаем количество книг, написанных автором

DECLARE @book\_count int

SELECT @book\_count = COUNT(\*)

FROM Сводная

WHERE id\_authors = @author\_id

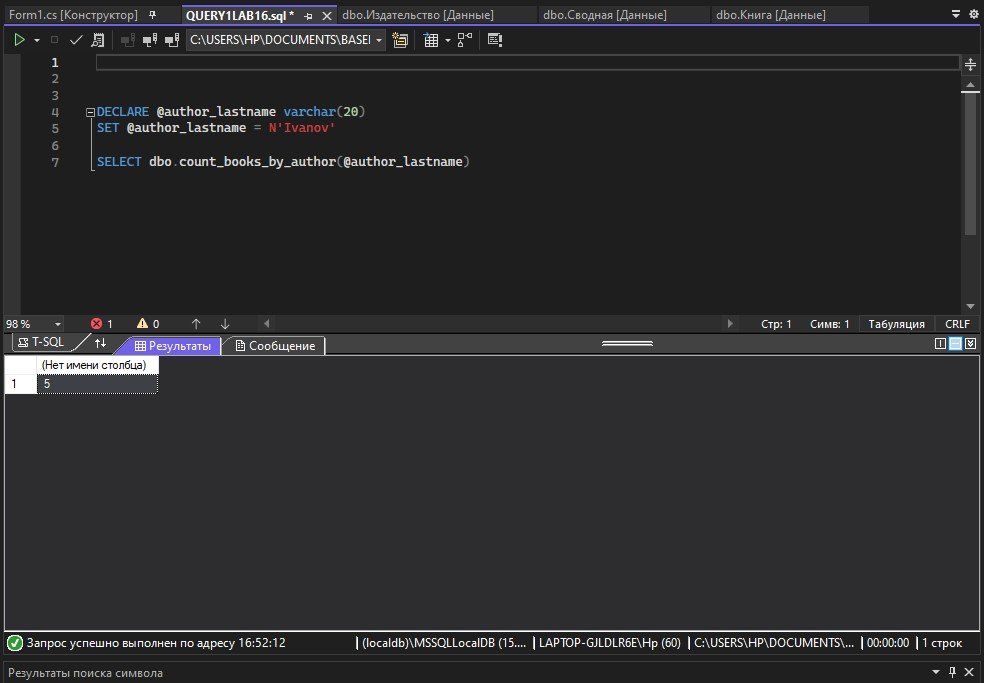
RETURN @book\_count

END

DECLARE @author\_lastname varchar(20)

SET @author\_lastname = N'Ivanov'

SELECT dbo.count\_books\_by\_author(@author\_lastname)



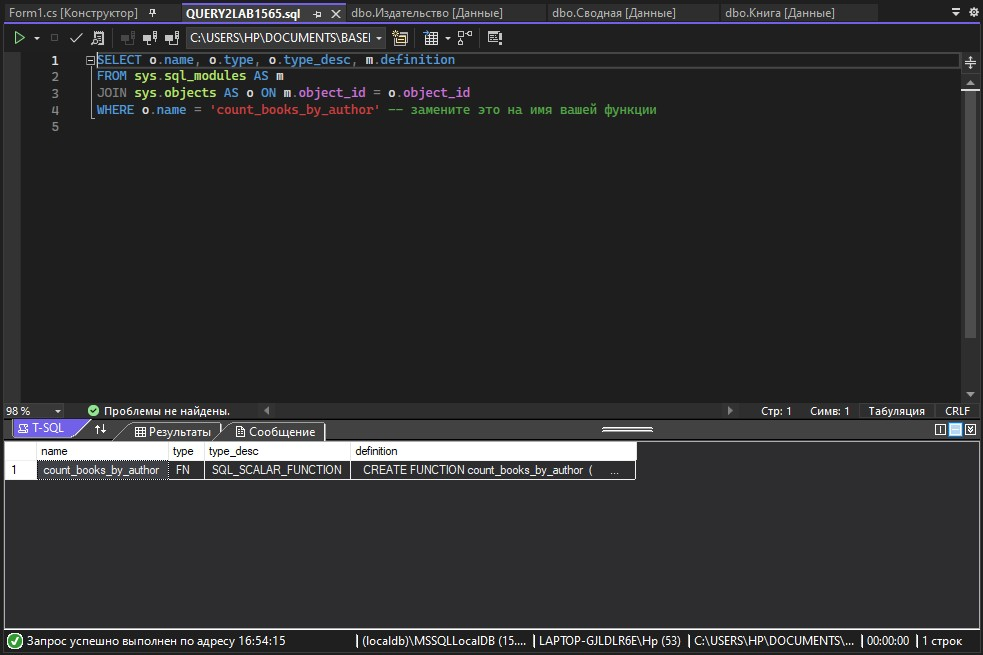
Задание 2: Просмотрите информацию о созданной вами функции, выбрав соответствующие поля из системных таблиц:

SELECT o.name, o.type, o.type\_desc, m.definition

FROM sys.sql\_modules AS m

JOIN sys.objects AS o ON m.object\_id = o.object\_id

WHERE o.name = 'count\_books\_by\_author' -- замените это на имя вашей функции



Задание 3: Измените созданную вами функцию при помощи инструкции ALTER FUNCTION (синтаксис которой аналогичен синтаксису CREATE FUNCTION), таким образом, чтобы еѐ определение было зашифрованным. Повторно просмотрите информацию о созданной функции.

ALTER FUNCTION count\_books\_by\_author

(

@author\_lastname varchar(20)

)

RETURNS int

WITH ENCRYPTION

AS

BEGIN

DECLARE @author\_id int

-- Получаем id автора по его фамилии

SELECT @author\_id = id\_authors

FROM Автор

WHERE fio\_authors = @author\_lastname

-- Если автор не найден, возвращаем -1

IF (@author\_id IS NULL)

RETURN -1

-- Иначе, считаем количество книг, написанных автором

DECLARE @book\_count int

SELECT @book\_count = COUNT(\*)

FROM Сводная

WHERE id\_authors = @author\_id

RETURN @book\_count

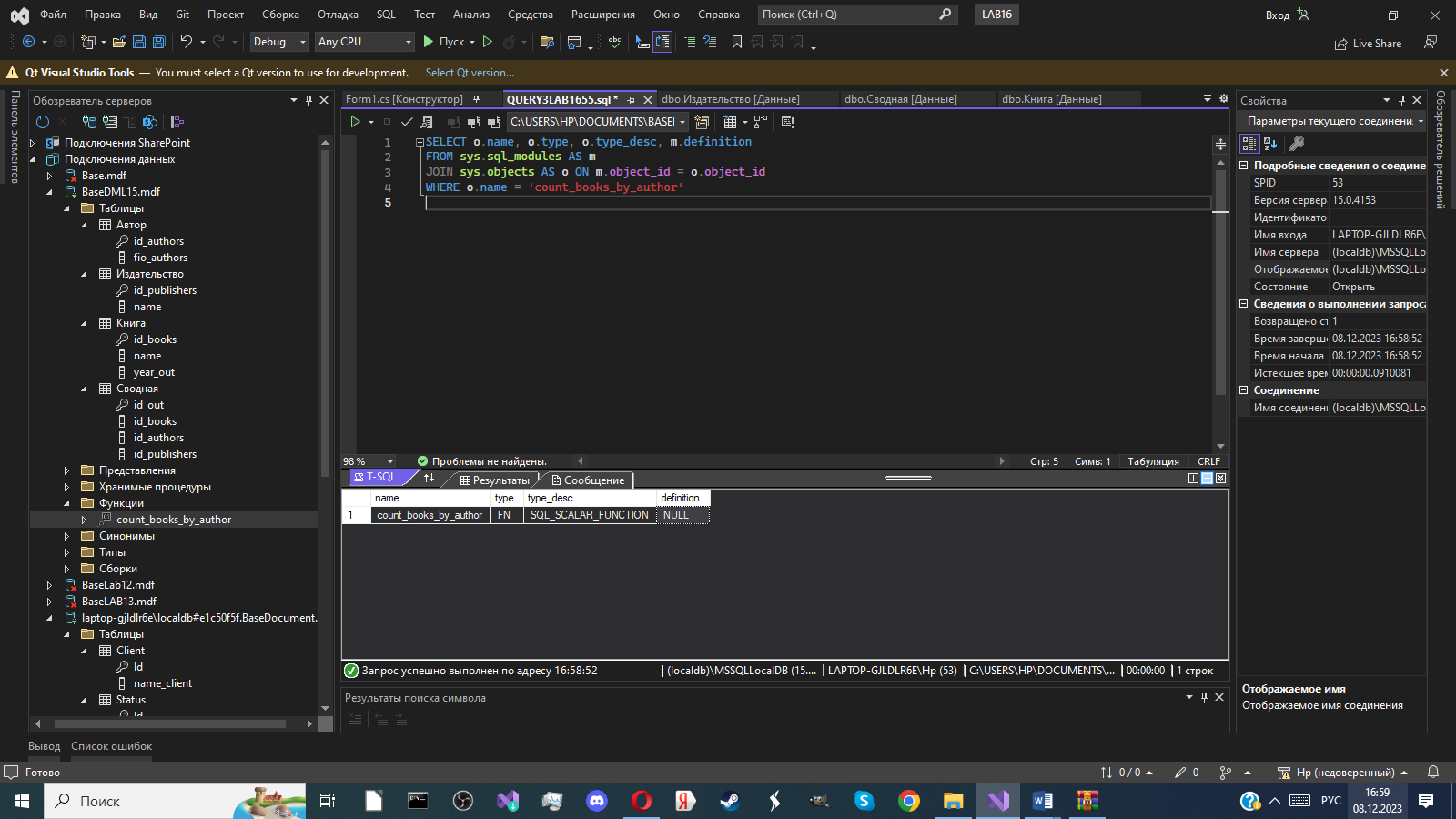
END

SELECT o.name, o.type, o.type\_desc, m.definition

FROM sys.sql\_modules AS m

JOIN sys.objects AS o ON m.object\_id = o.object\_id

WHERE o.name = 'count\_books\_by\_author'



Задание 4: Удалите созданную вами функцию, используя следующую конструкцию:

DROP FUNCTION count\_books\_by\_author

Задание 5: Создайте **inline-функцию**, принимающую параметром название издательства и возвращающую табличное значение, содержащее книги, изданные в данном издательстве (***Название книги***, ***Фамилии авторов*** через запятую; ***Год издания книги***).

Синтаксис создания ***inline-функции***, возвращающей табличное значение, следующий:

CREATE FUNCTION GetBooksByPublisher (@publisherName varchar(20))

RETURNS TABLE

AS

RETURN (

SELECT

b.name AS 'Название книги',

STRING\_AGG(a.fio\_authors, ', ') AS 'Фамилии авторов',

b.year\_out AS 'Год издания книги'

FROM

Издательство AS p

JOIN

Сводная AS s ON p.id\_publishers = s.id\_publishers

JOIN

Книга AS b ON s.id\_books = b.id\_books

JOIN

Автор AS a ON s.id\_authors = a.id\_authors

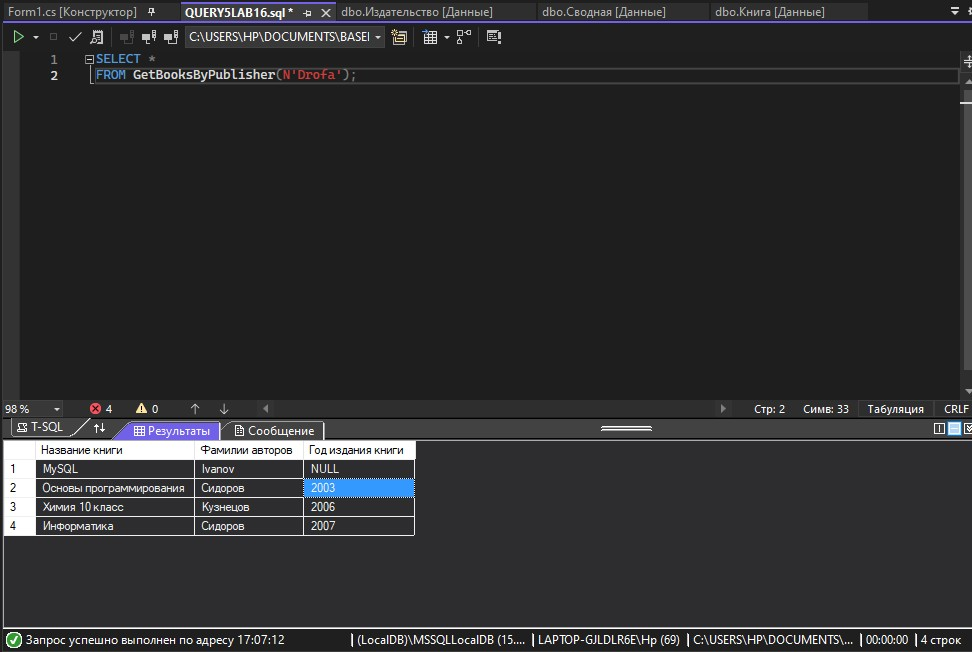
WHERE

p.name = @publisherName

GROUP BY

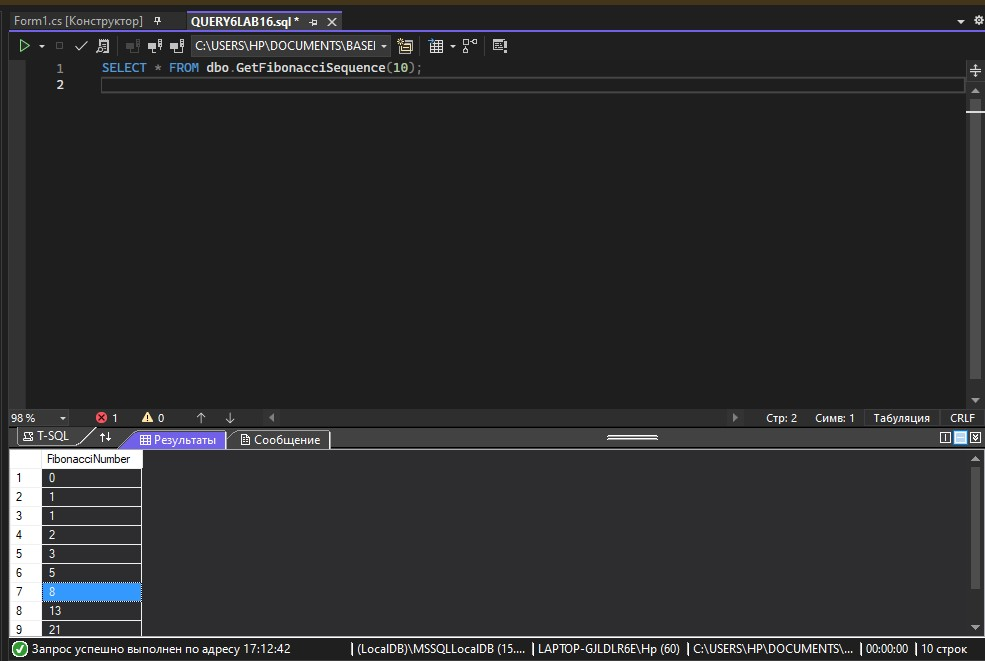
b.name, b.year\_out

)



Задание 6: Создайте функцию, возвращающую табличное значение, принимающую параметром число возвращаемых записей (RecordCount), и возвращающую RecordCount первых чисел из последовательности Фибоначчи.

Синтаксис создания функции, возвращающей табличное значение, следующий:



**CREATE FUNCTION dbo.GetFibonacciSequence(@RecordCount INT)**

**RETURNS @FibonacciTable TABLE (FibonacciNumber INT)**

**AS**

**BEGIN**

**DECLARE @a INT = 0;**

**DECLARE @b INT = 1;**

**DECLARE @count INT = 0;**

**WHILE @count < @RecordCount**

**BEGIN**

**INSERT INTO @FibonacciTable (FibonacciNumber) VALUES (@a);**

**SET @b = @a + @b;**

**SET @a = @b - @a;**

**SET @count = @count + 1;**

**END**

**RETURN;**

**END;**

SELECT \* FROM dbo.GetFibonacciSequence(10);