МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Казанский национальный исследовательский технический университет

им. А.Н. Туполева – КАИ»

Институт компьютерных технологий и защиты информации

Отделение СПО ИКТЗИ (Колледж информационных технологий)

ЛАБОРАТОРНАЯ РАБОТА №3

по дисциплине

Информационная безопасность

Тема: «Реализация доступа пользователей к базе данных»

Работу выполнил

Студент гр.4238

Гилязов К.Р

Принял

Преподаватель Кожевников К.Д.

Казань 2024

**Цель** – Научиться выполнять резервное копирования и восстановление базы данных из резервной копии.

**Шаг 1.**

С помощью SQL Server Management Studio подключаемся к используемому экземпляру SQL Server.

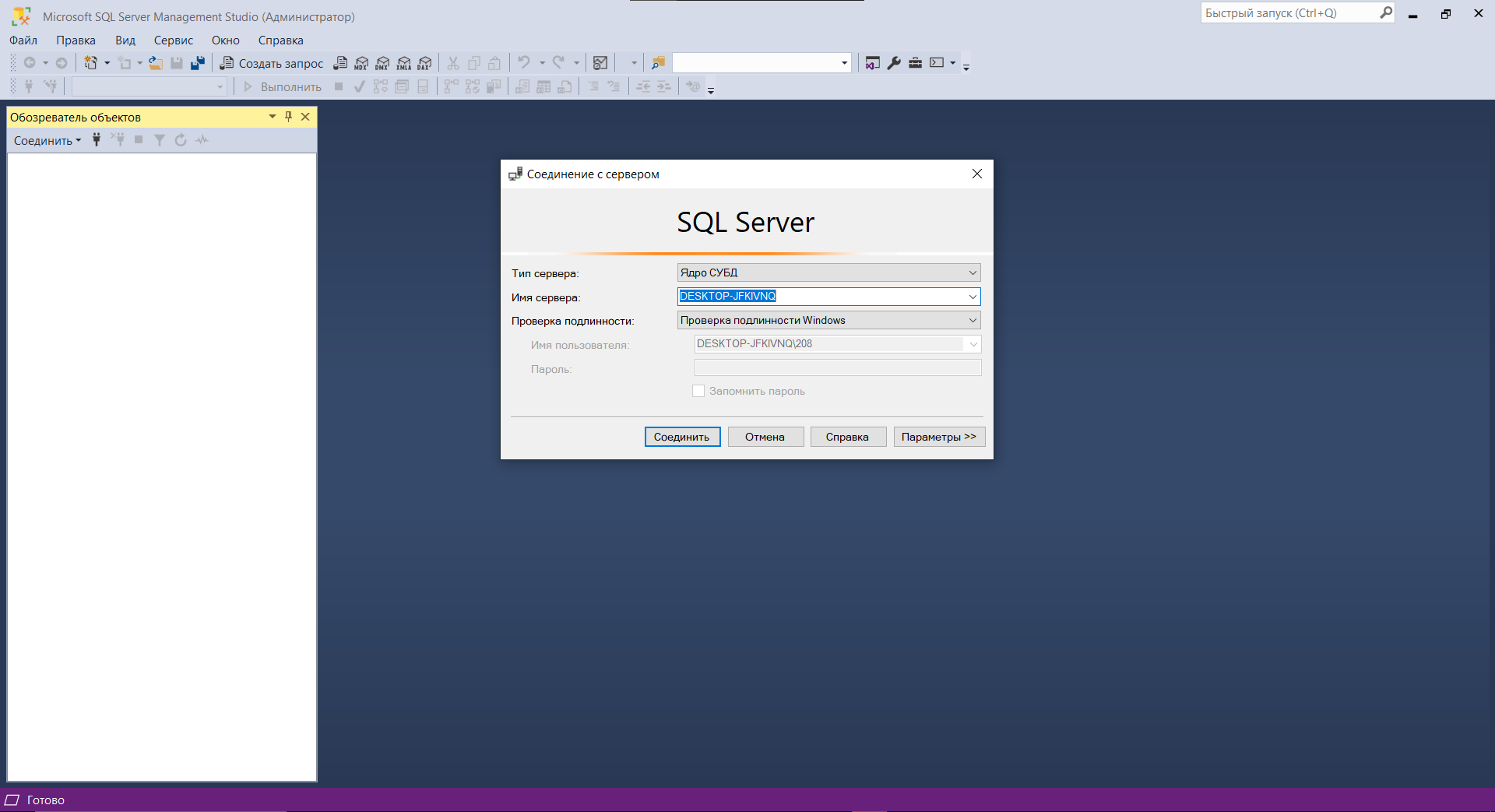
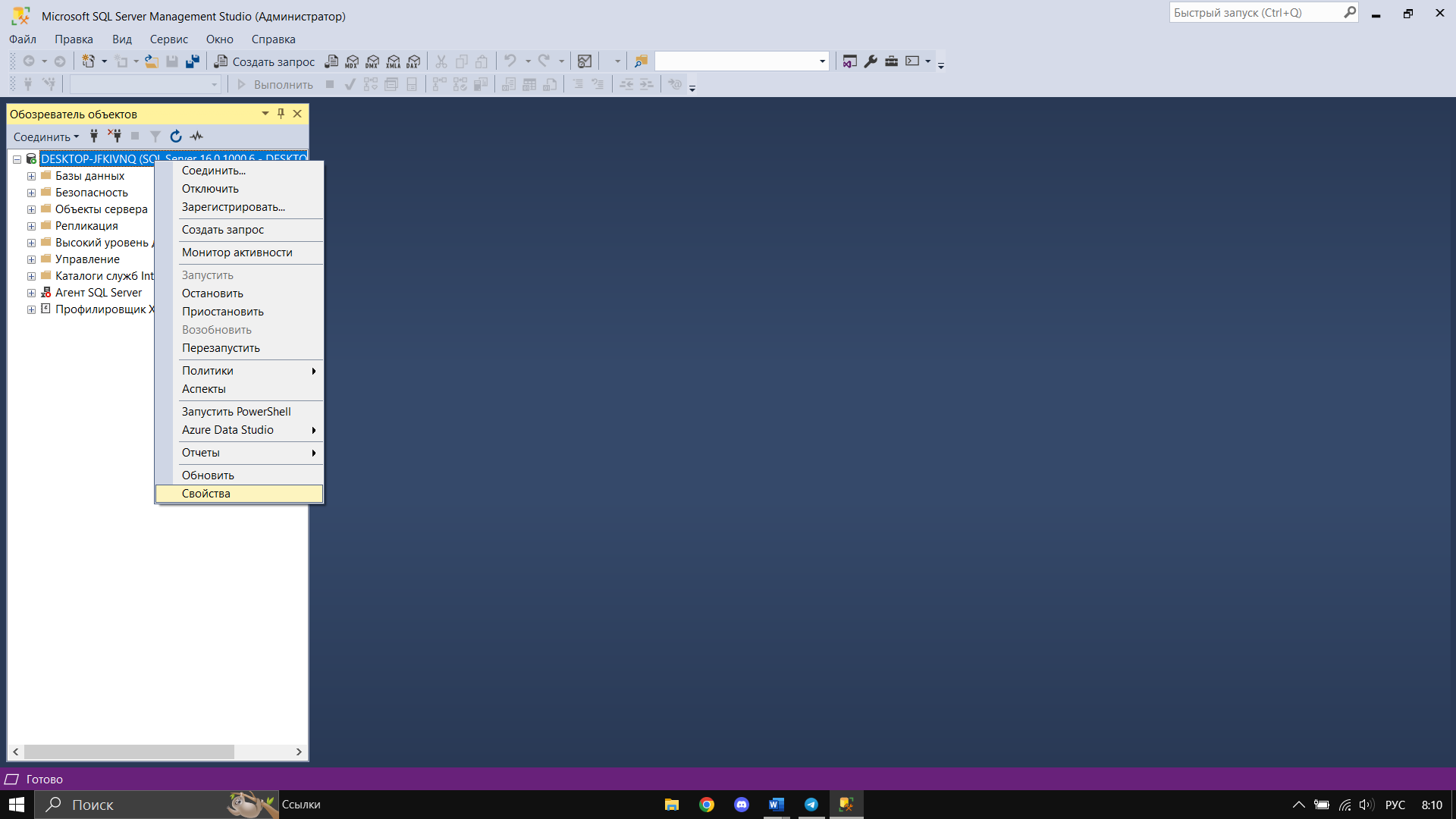


Рисунок 1. -Соединение с сервером.

Заходим в свойства



Проверяем какие настройки у нас установлены, в графе “Серверная проверка подлинности”. Нужно изменить на проверку подлинности SQL Server и Windows

Рисунок 2. – Свойства.

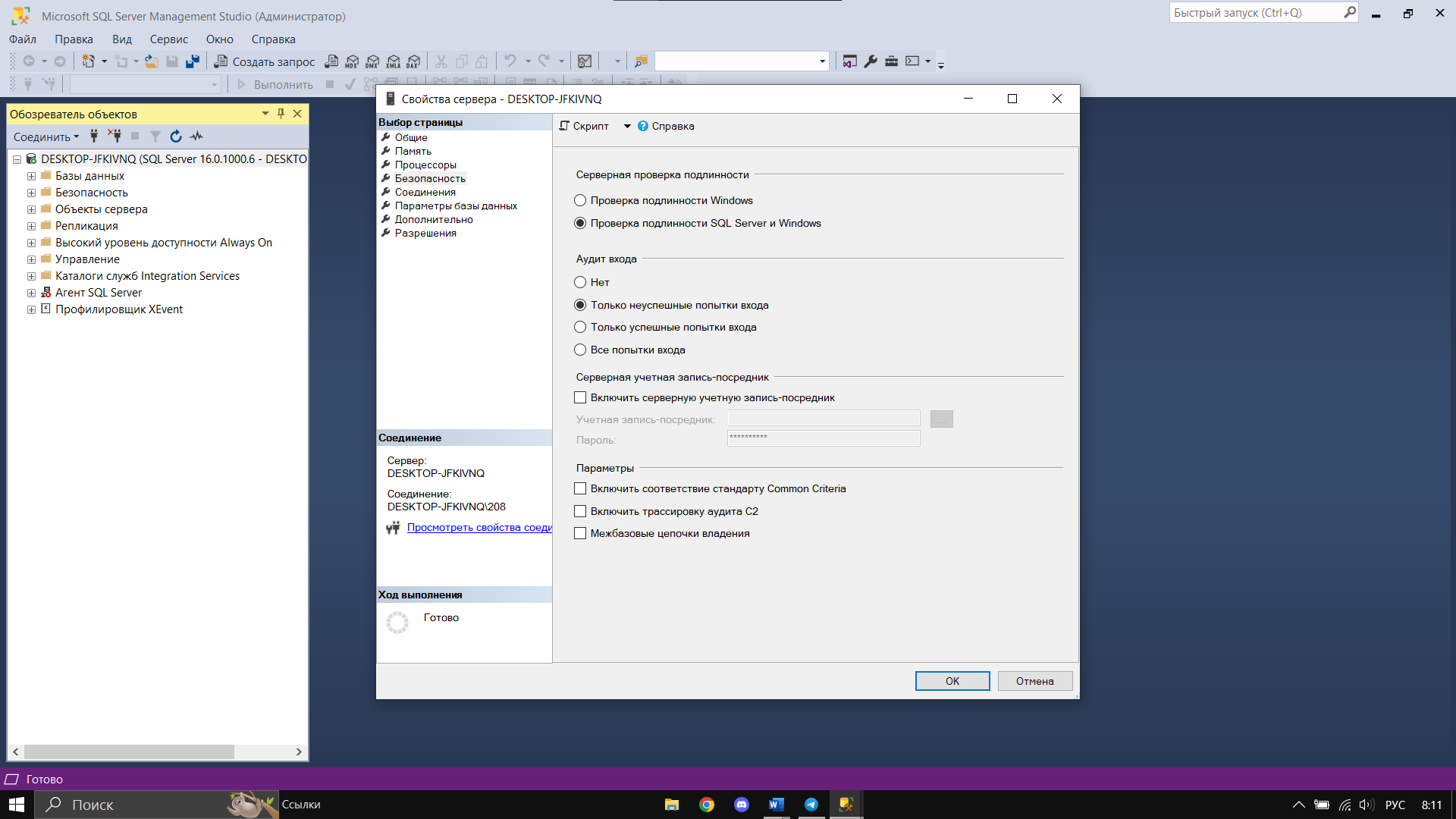


Рисунок 3. – Проверка подлинности.

**Шаг 2.**

В окне Object Explorer найти и развернуть дерево объектов, чтобы увидеть различные папки и объекты, связанные с сервером SQL.

Для этого выполните следующие действия:

Найти папку "Security" (Безопасность) в дереве объектов.

В папке "Security" развернуть подпапку "Logins" (Учетные записи).

Найти используемую учетную запись в списке учетных записей (logins).

Щелкнуть правой кнопкой мыши на учетной записи и выбрать опцию "Properties" (Свойства).

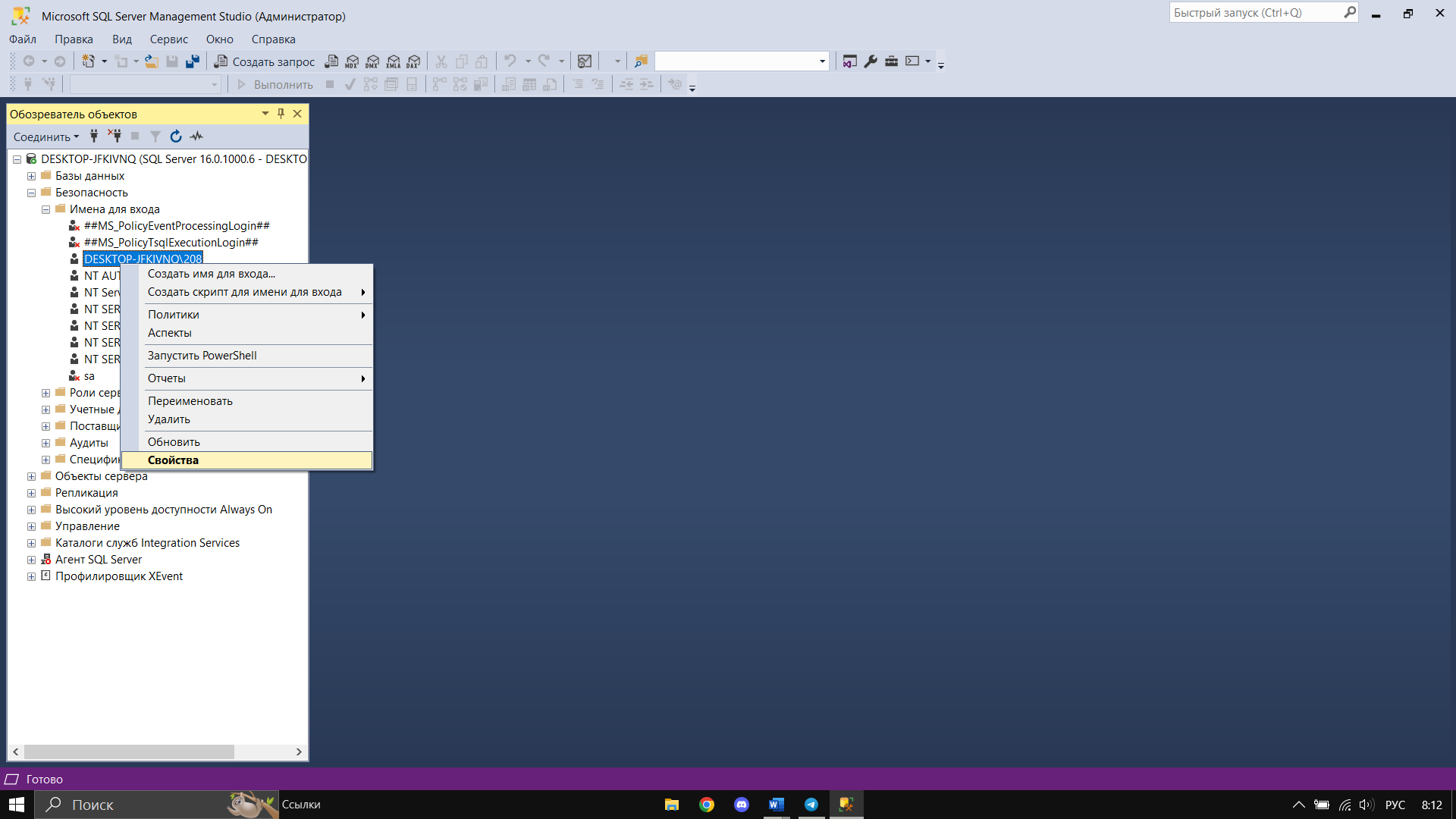


Рисунок 4. – Свойства.

В открывшемся окне свойств учетной записи мы сможем увидеть информацию о настройках и разрешениях вашей учетной записи.

Перейдем на вкладку "Server Roles" (Роли сервера), чтобы увидеть, на выполнение каких серверных ролей авторизована учетная запись.

Здесь можем увидеть список доступных серверных ролей (например, sysadmin, dbcreator, public и т.д.) и проверить, к каким из них относится учетная запись.

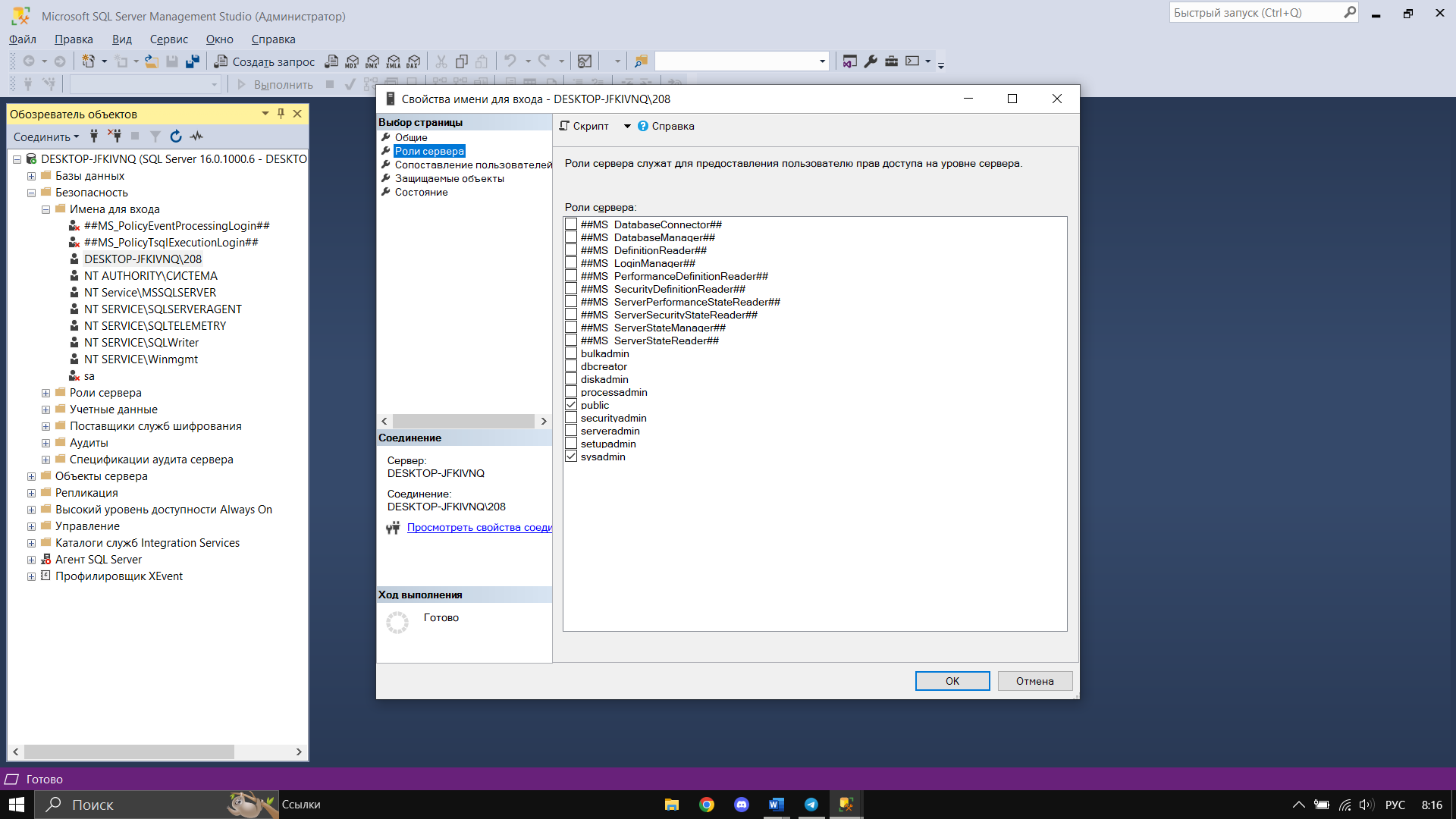


Рисунок 5. – Проверка роли сервера.

**Шаг 3.**

Найти в окне Object Explorer ваш сервер SQL и раскрыть его для просмотра баз данных.

Развернуть папку "Databases" (Базы данных), чтобы увидеть список всех баз данных, существующих на вашем сервере.

Нужно найти базы данных, в которых учетная запись сопоставлены пользователи.

Для этого выполним следующие действия:

Развернуть папку "Security" внутри выбранной базы данных.

Под папкой "Security" найдити папку "Users" (Пользователи). Здесь увидим список пользователей, сопоставленных с данной базой данных.

Найти учетную запись среди пользователей базы данных и проверить, на выполнение каких ролей она авторизована в рамках этой базы данных.

Чтобы узнать, на выполнение каких ролей пользователь авторизован в базе данных:

Выбрать пользователя в списке.

Щелкнуть правой кнопкой мыши на пользователе и выберать опцию "Properties" (Свойства).



Рисунок 6. - Свойства.

Перейдем на вкладку "Membership" (Членство), чтобы увидеть список ролей баз данных, к которым пользователь относится.

Ознакомимся с информацией о ролях (например, db\_owner, db\_datareader, db\_datawriter и т.д.), к которым пользователь авторизован в данной базе данных.

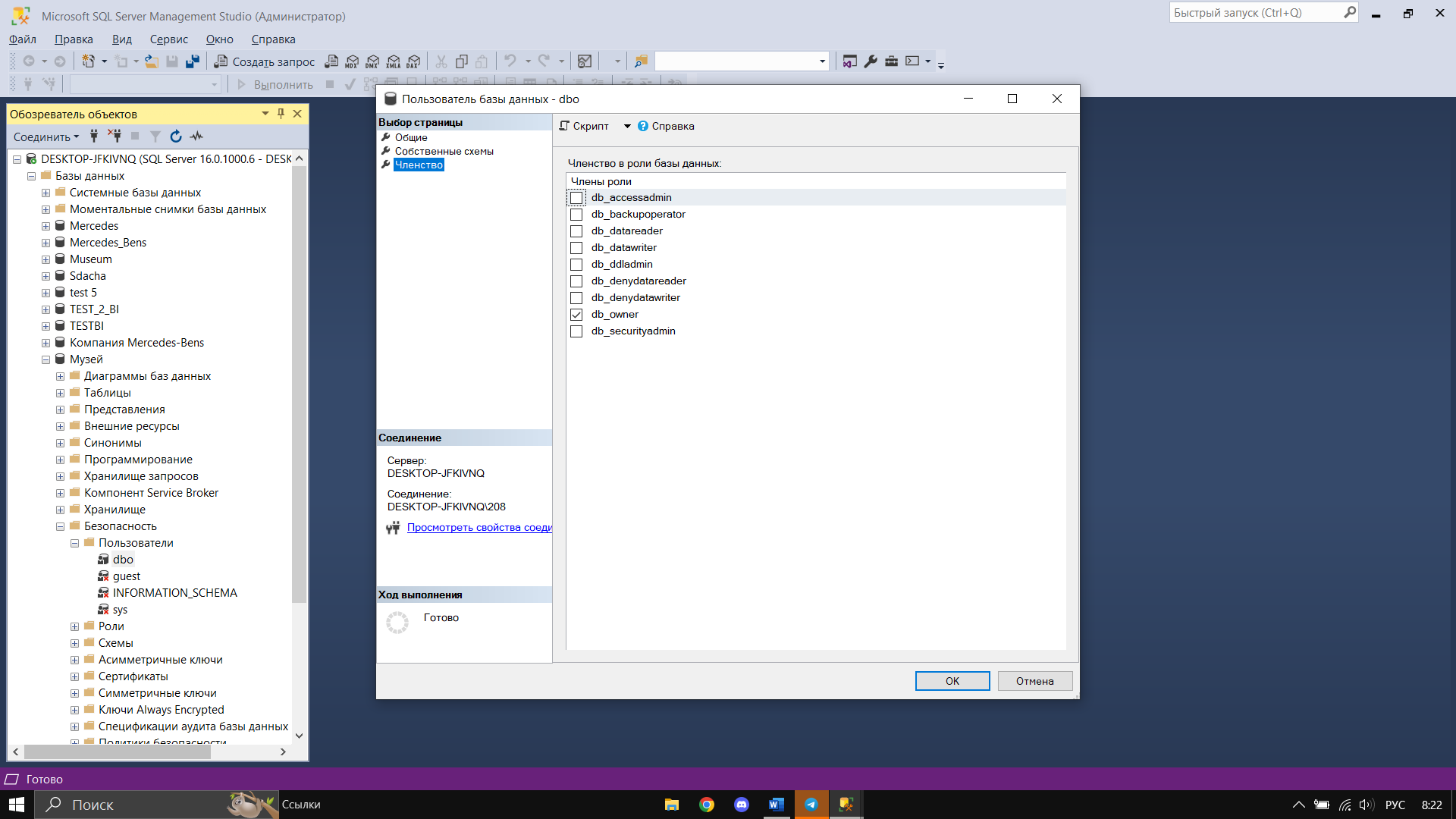


Рисунок 7. – Членство.

**Шаг 4.**

В среде Management Studio создадим новую базу данных. Откроем список пользователей и ролей. Убедимся, что учетная запись, под которой вы работаете, сопоставлена пользователю dbo, авторизованному на роль db owner.

Создание новой базы данных:

Щелкним правой кнопкой мыши на папке "Databases" (Базы данных) в Object Explorer.

Выберем опцию "New Database" (Новая база данных).

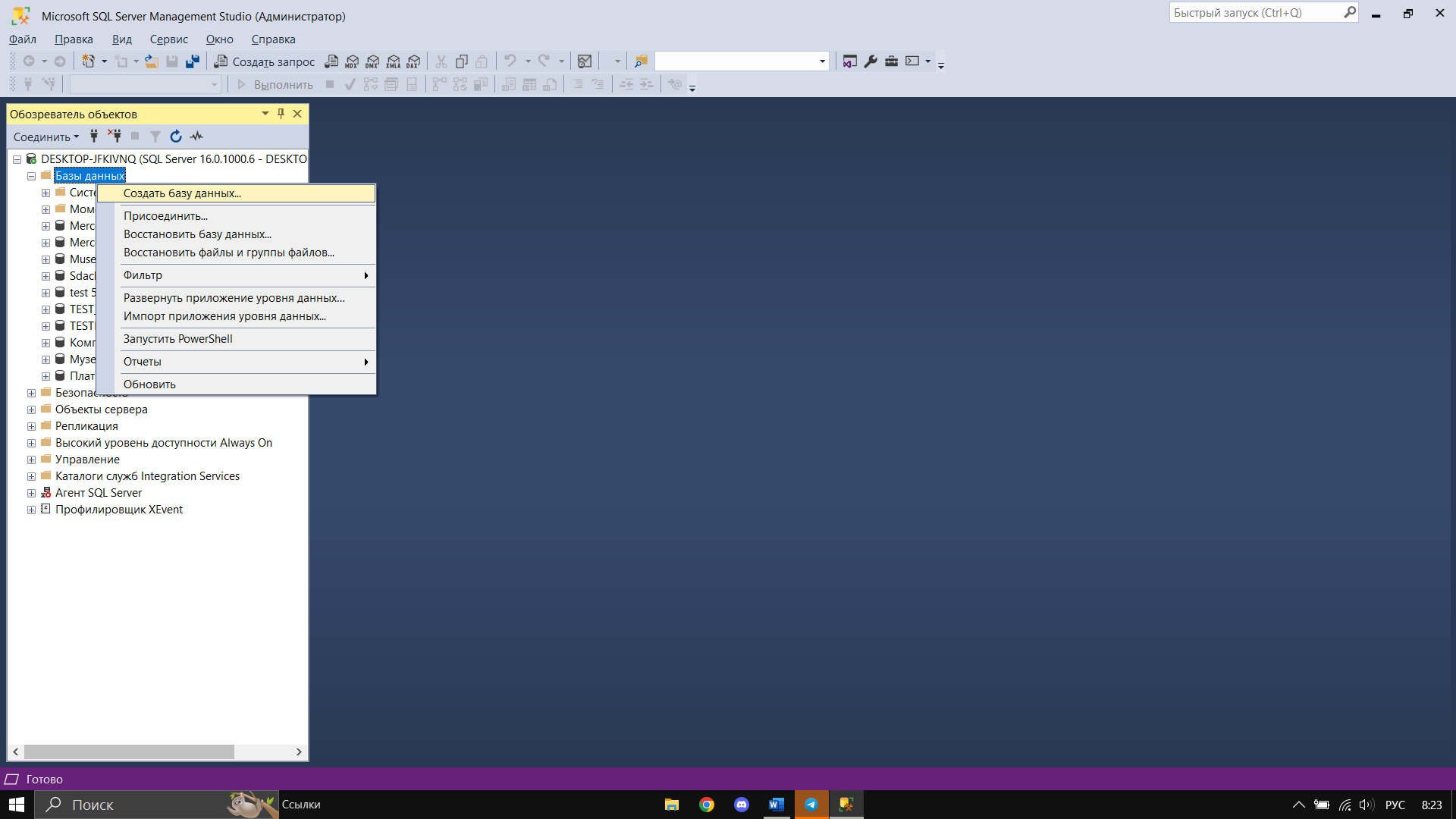


Рисунок 8. – Создание БД.

В появившемся диалоговом окне введем имя для новой базы.

Нажимаем "OK", чтобы создать новую базу данных.

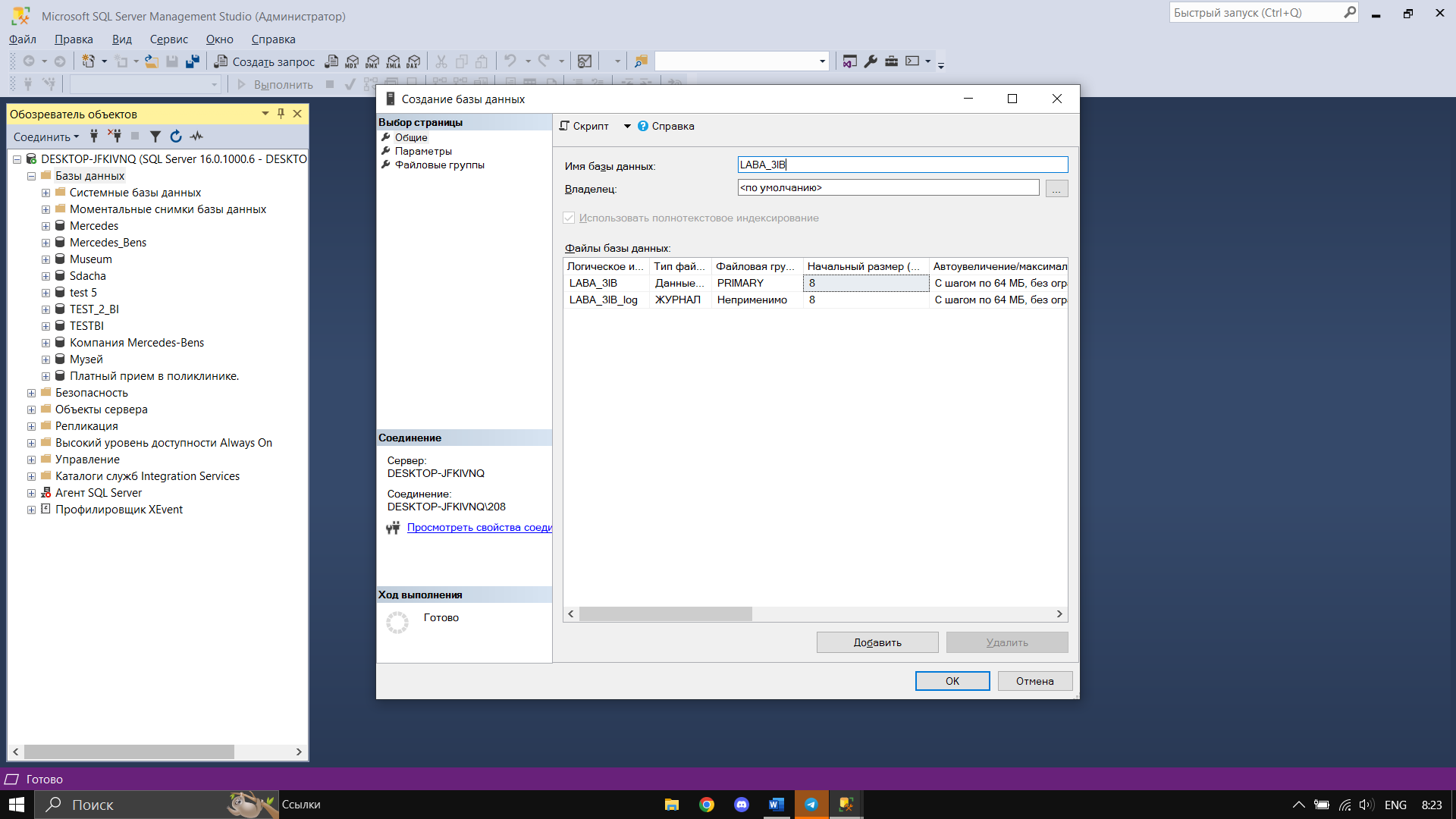


Рисунок 9. – Имя БД.

Проверка списка пользователей и ролей:

Развернем созданную новую базу данных в списке баз данных.

Найдем папку "Security" внутри новой базы данных и развернем ее.

Посмотрим список пользователей, сопоставленных с новой базой данных.

Найдем пользователя "dbo" (Database Owner) среди списка пользователей.

Убедитесь, что ваша учетная запись сопоставлена пользователю dbo, авторизованному на роль db owner:

Щелкните правой кнопкой мыши на пользователе "dbo" и выберите опцию "Properties" (Свойства).

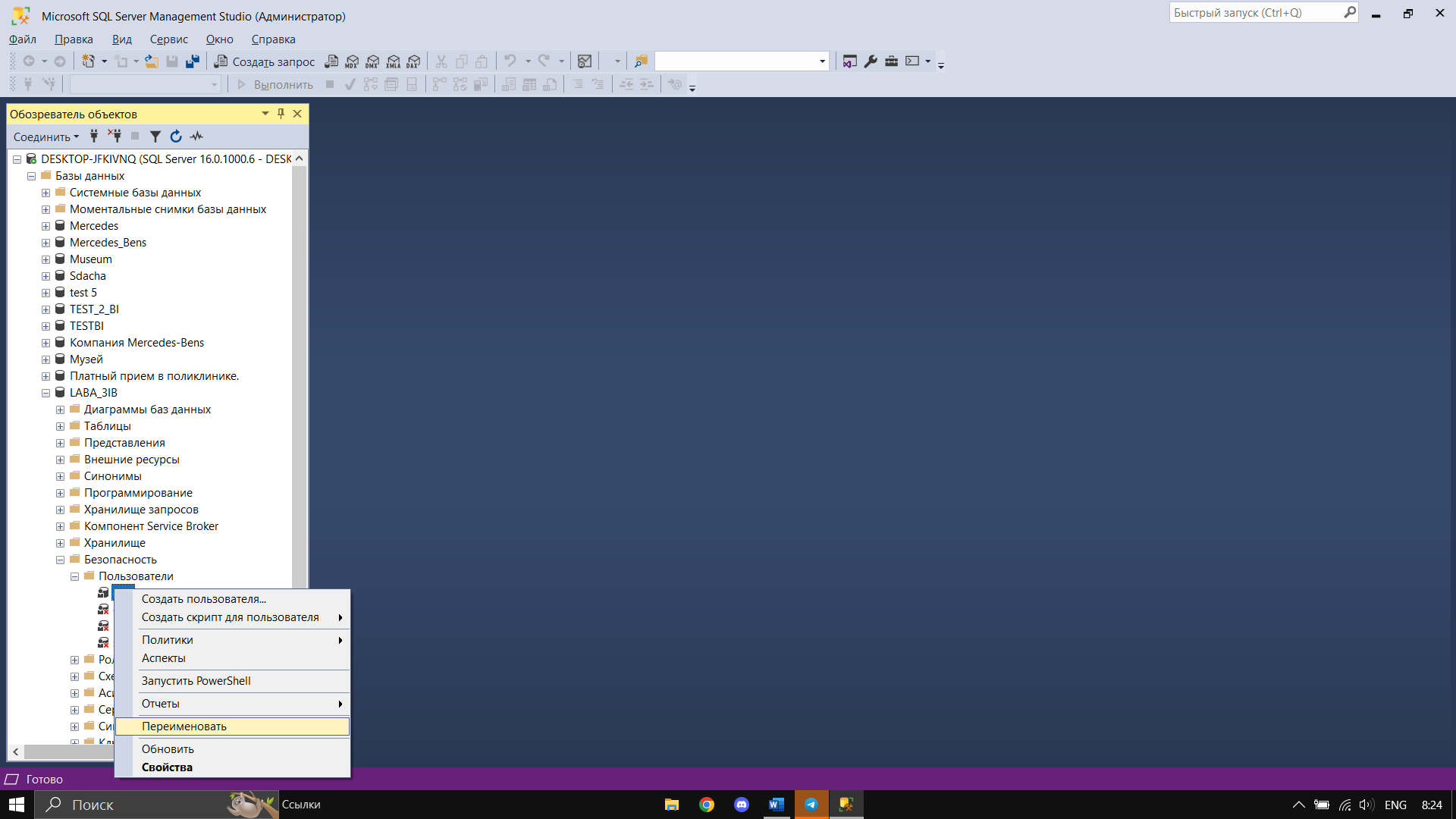


Рисунок 10. Свойства dbo.

Перейдите на вкладку "Membership" (Членство) в окне свойств пользователя "dbo".

Убедимся, что роль "db\_owner" присутствует в списке ролей, к которым этот пользователь авторизован.

Если роль "db\_owner" отсутствует, то сделаем пользователю "dbo" членом роли "db\_owner".

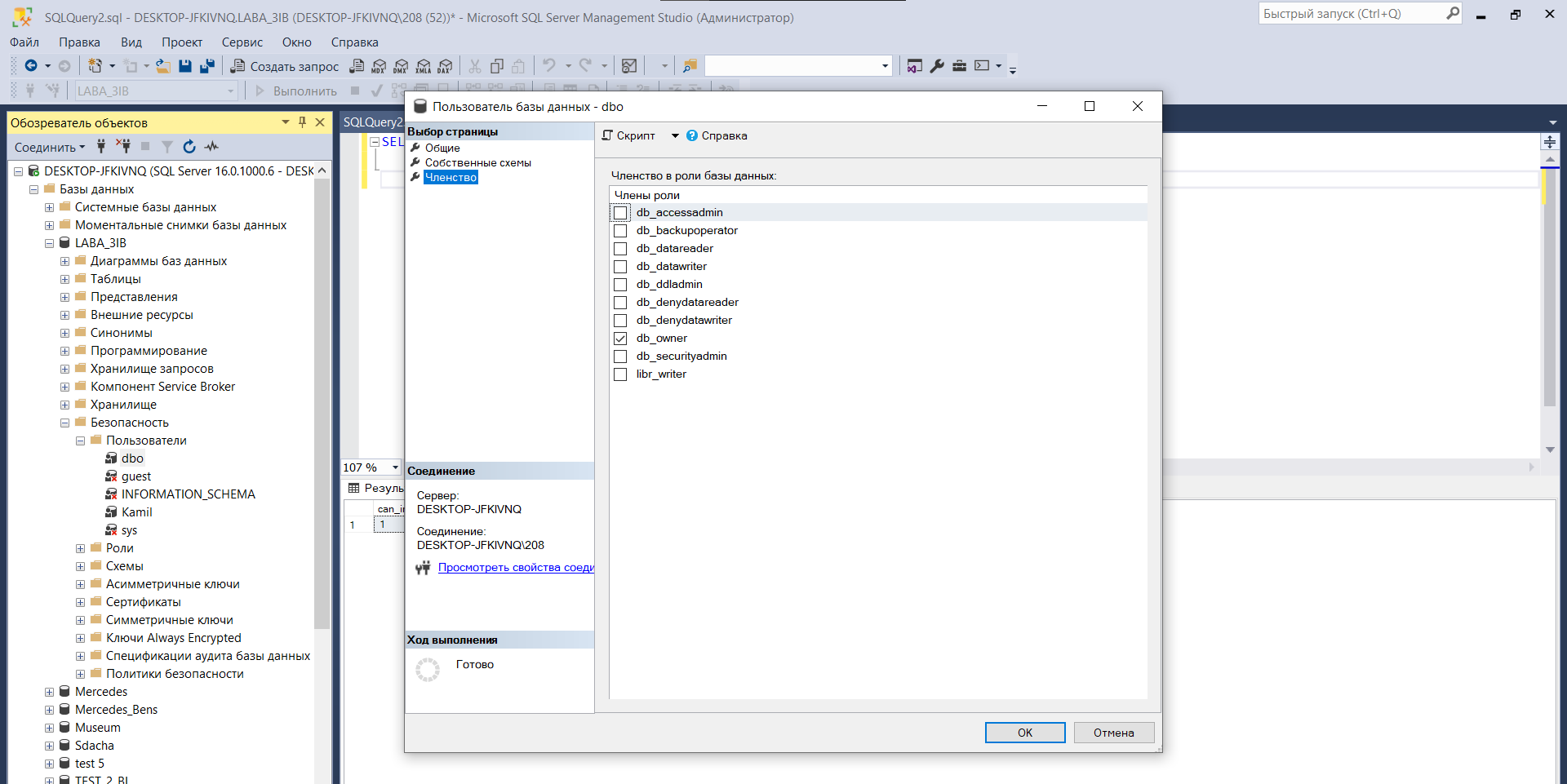


Рисунок 11. – Членство.

**Шаг 5.**

Используя приведенный ниже скрипт, создадим в базе данных таблицы

use [LABA\_3IB]

GO

CREATE TABLE dbo.Book (

book\_id int IDENTITY (1, 1) primary key,

Title varchar(50) NOT NULL,

Author varchar(50),

Publisher varchar(50),

[Year] smallint

)

GO

CREATE TABLE dbo.Status (

Status\_id int IDENTITY (1, 1) primary key,

Status\_name varchar(50) NOT NULL

)

GO

CREATE SCHEMA libr

GO

CREATE TABLE libr.Book\_in\_lib (

lib\_id int primary key,

book\_id int references dbo.Book,

status\_id int references dbo.[Status]

)

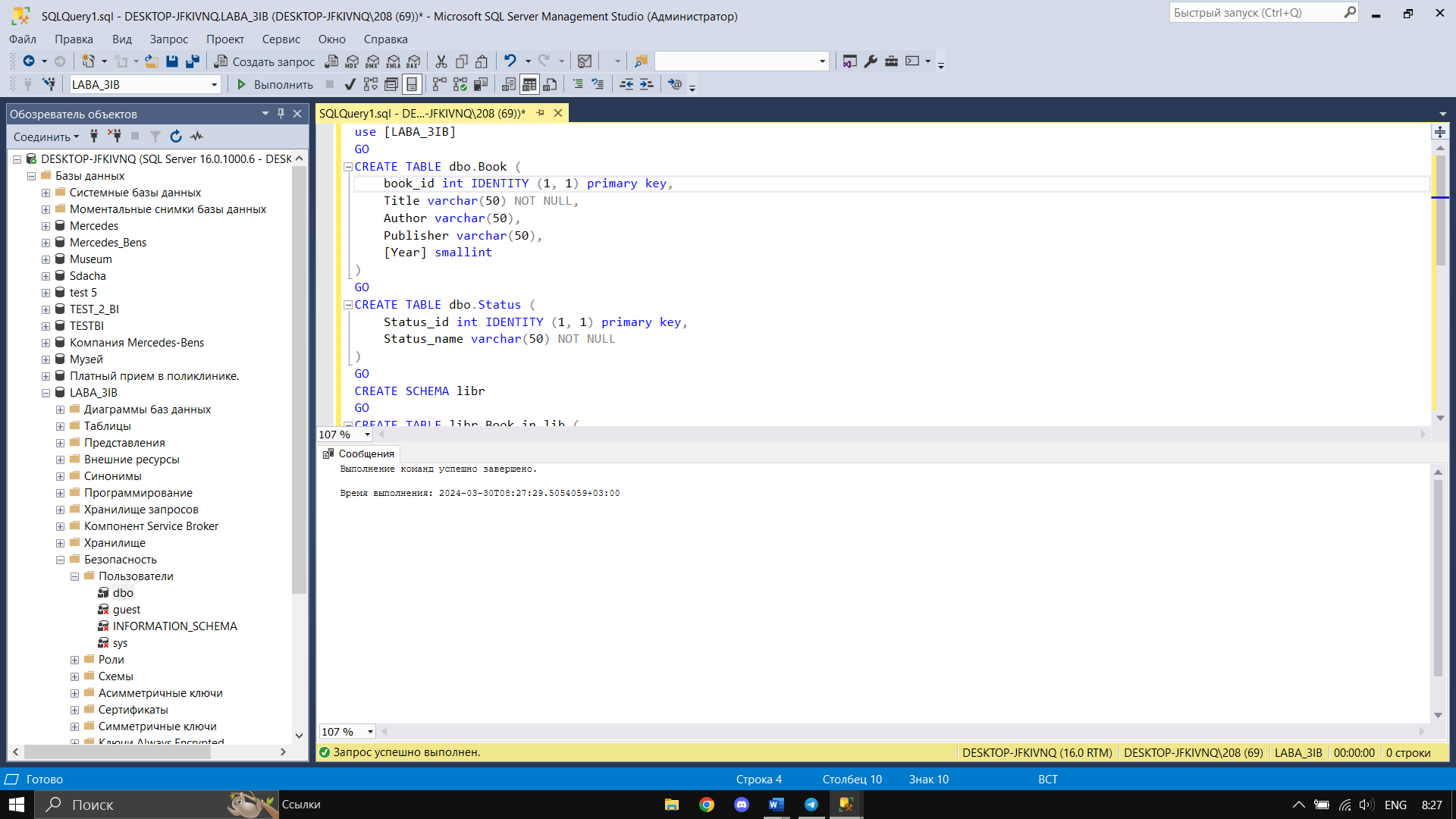


Рисунок 12. – Создание таблиц.

После успешного выполнения должны быть созданы таблицы: "Book", "Status", "Book\_in\_lib" и схема "libr".

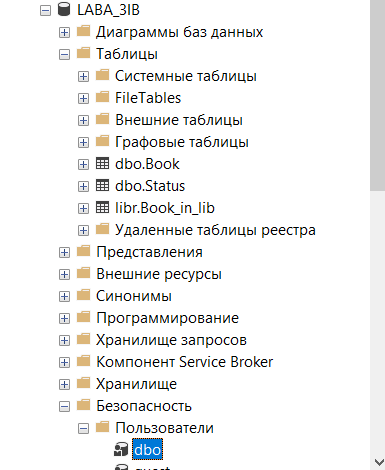


Рисунок 13. – Проверка.

**Шаг 6.**

Создание пользователя, выбор схемы dbo, добавление в роль db\_datareader:

Для создания пользователя, выбора схемы по умолчанию и добавления в роль db\_datareader, нужно выполнить следующий скрипт.

USE [LABA\_3IB];

-- Создание логина с указанием пароля

CREATE LOGIN Kamil WITH PASSWORD = '1111';

-- Создание пользователя и назначение схемы dbo

CREATE USER Kamil FOR LOGIN Kamil WITH DEFAULT\_SCHEMA = dbo;

-- Добавление пользователя в роль db\_datareader

EXEC sp\_addrolemember 'db\_datareader', ' Kamil ';

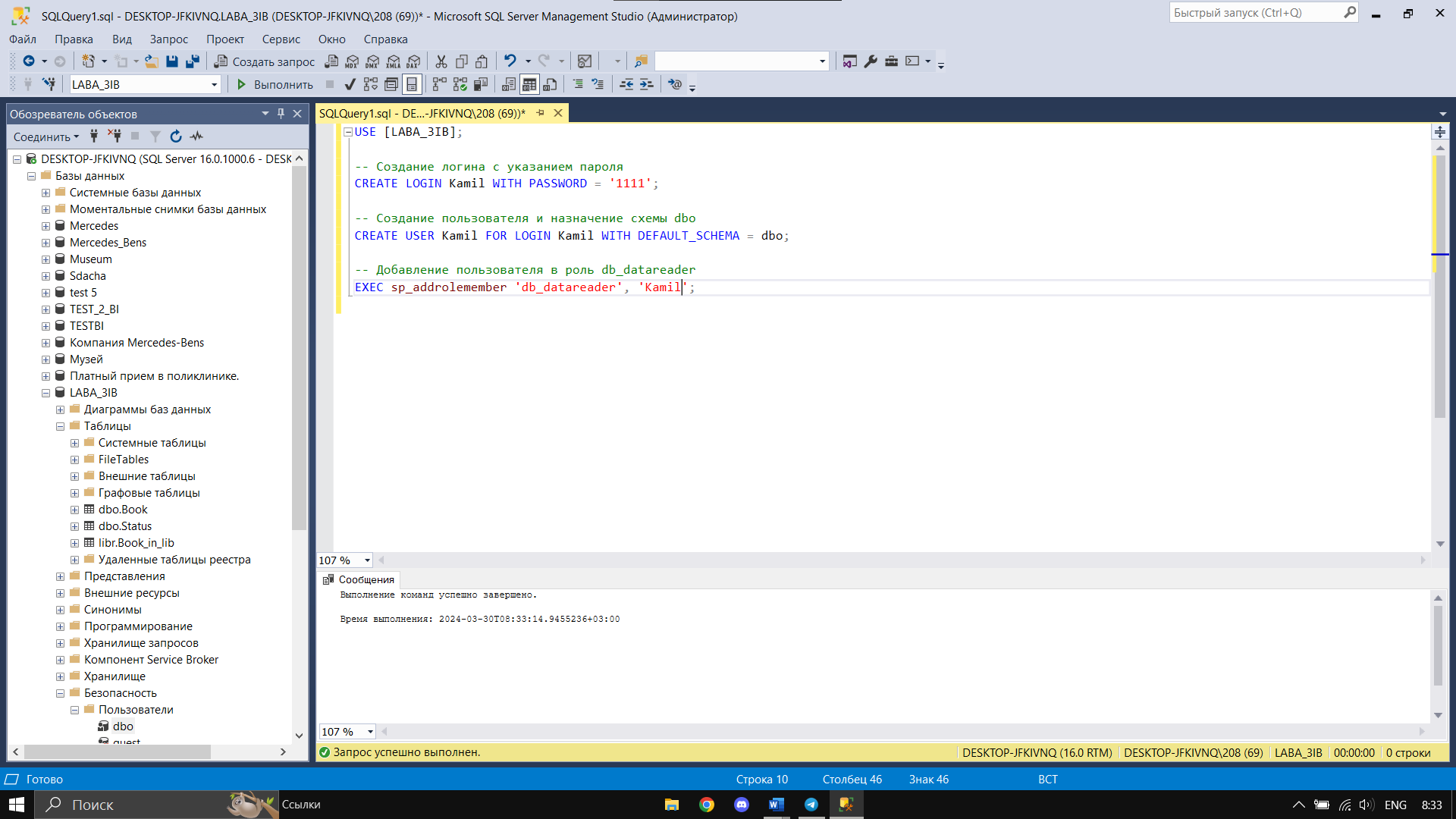


Рисунок 14. – Скрипт.

Тестирование не прошло проверку

**Шаг 7.**

Создание новой роли и назначение разрешений:

Подключимся к серверу базы данных:

Используя административные учетные данные для подключения к серверу баз данных.

Создаем новую роль уровня базы данных: Открываем новый запрос T-SQL и введем следующий запрос:

CREATE ROLE libr\_writer;

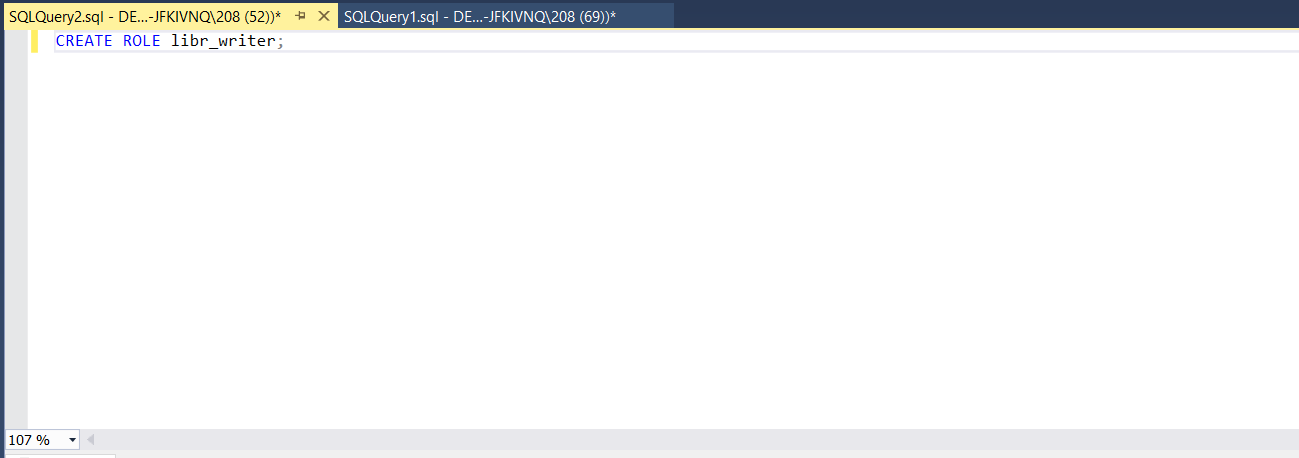


Рисунок 15. – Новая роль.

Этот запрос создаст новую роль с именем "libr\_writer".

Назначаем разрешения роли для схемы "libr":

Введем следующий запрос, чтобы предоставить разрешения роли "libr\_writer" на выполнение операций INSERT, UPDATE и DELETE для объектов в схеме "libr":

GRANT INSERT, UPDATE, DELETE ON SCHEMA::libr TO libr\_writer

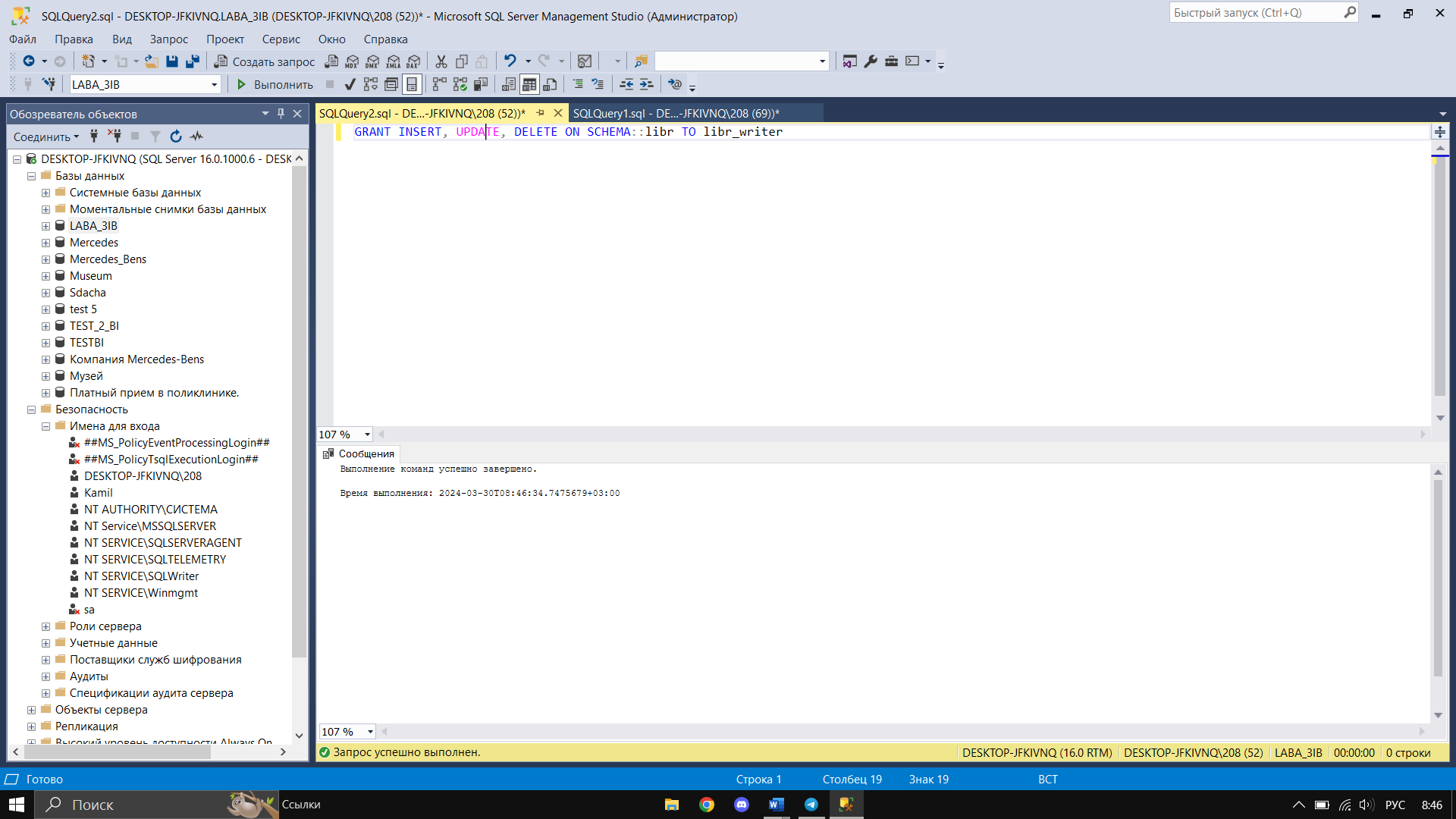


Рисунок 16. – Разращение роли.

EXEC sp\_addrolemember 'libr\_writer', 'Kamil';

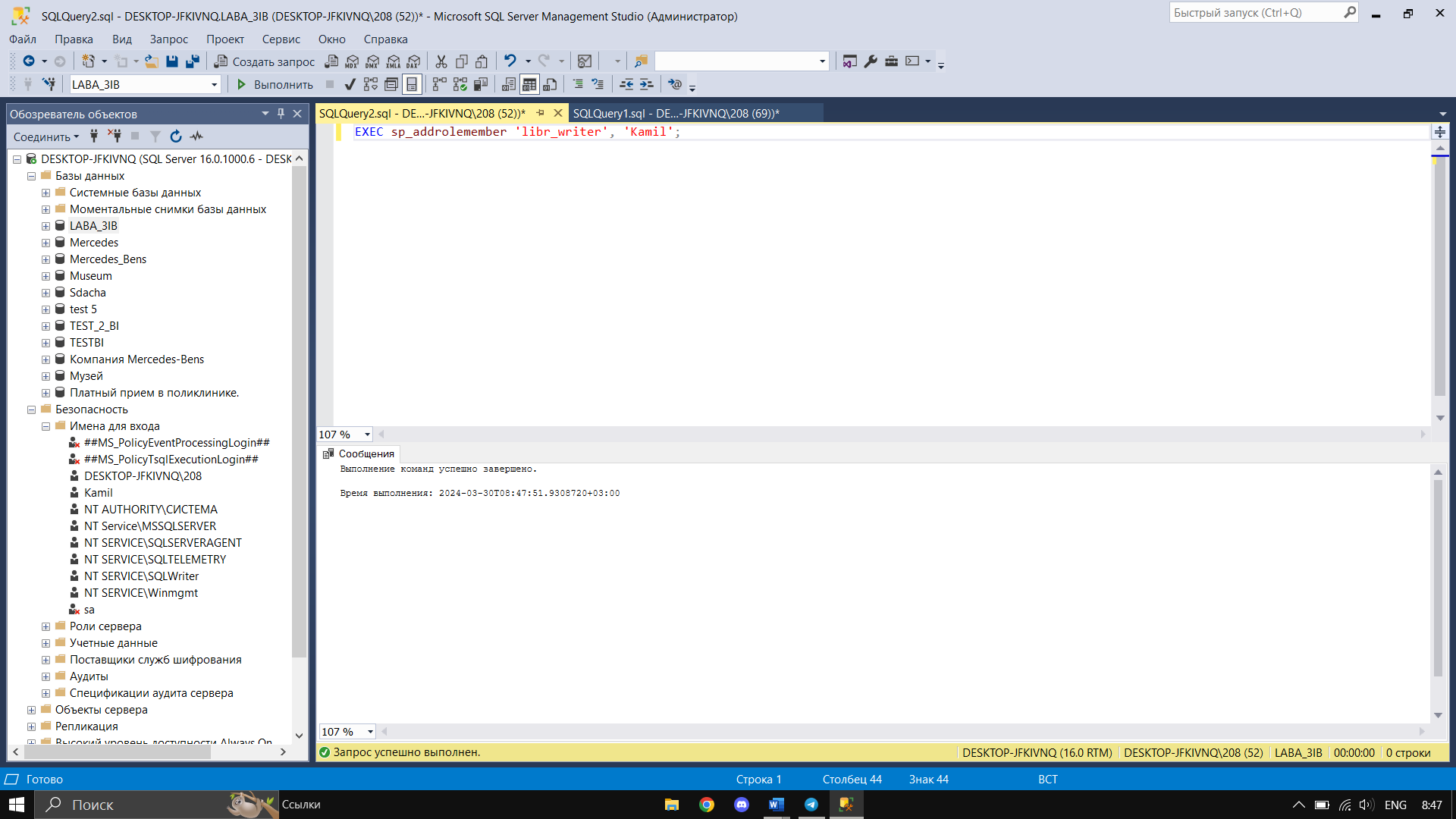


Рисунок 17. – Назначение роли.

Выполним операции INSERT, UPDATE и DELETE для данных в таблице "Book\_in\_lib", принадлежащей схеме "libr".

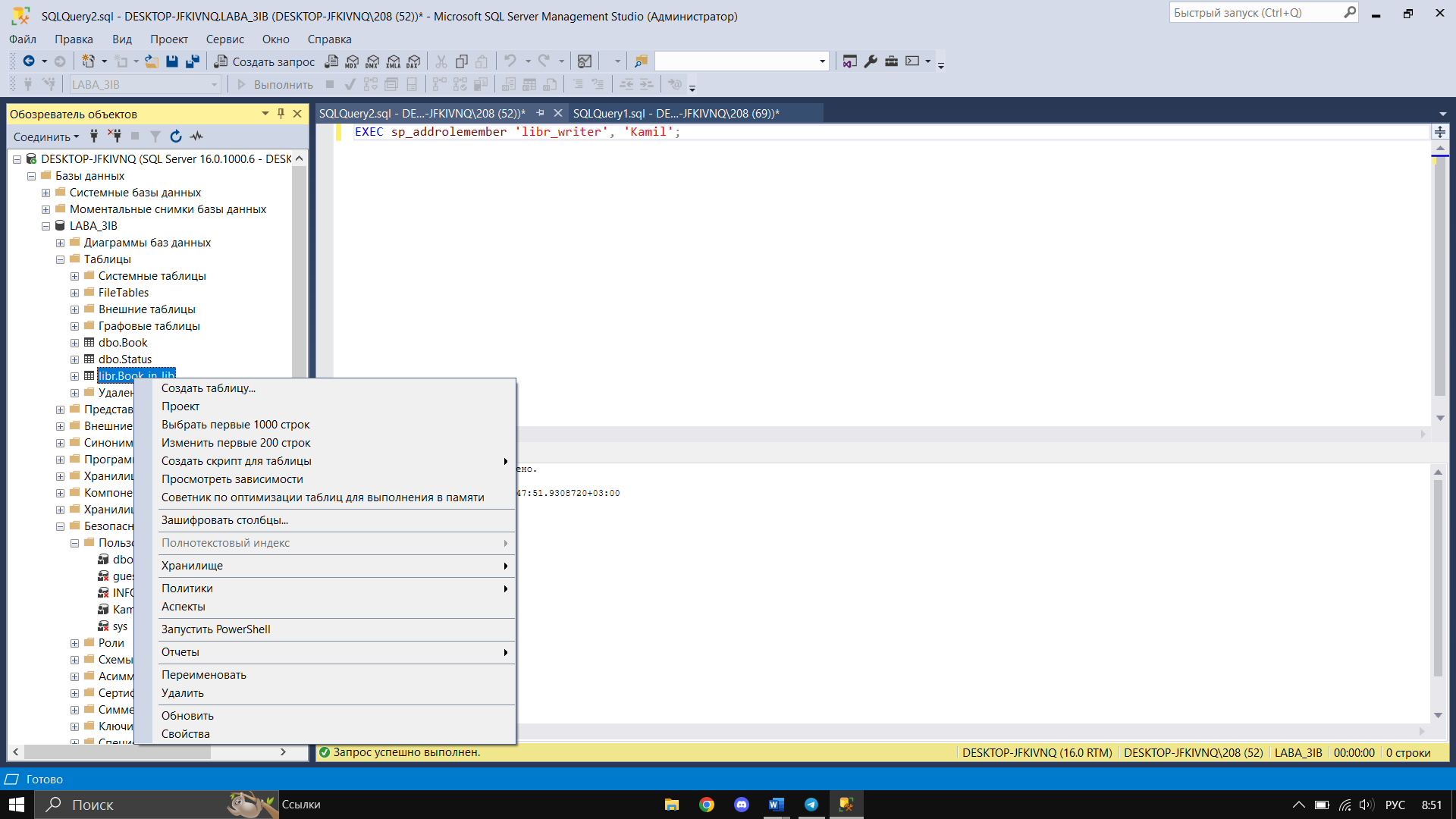


Рисунок 18. – Свойства.

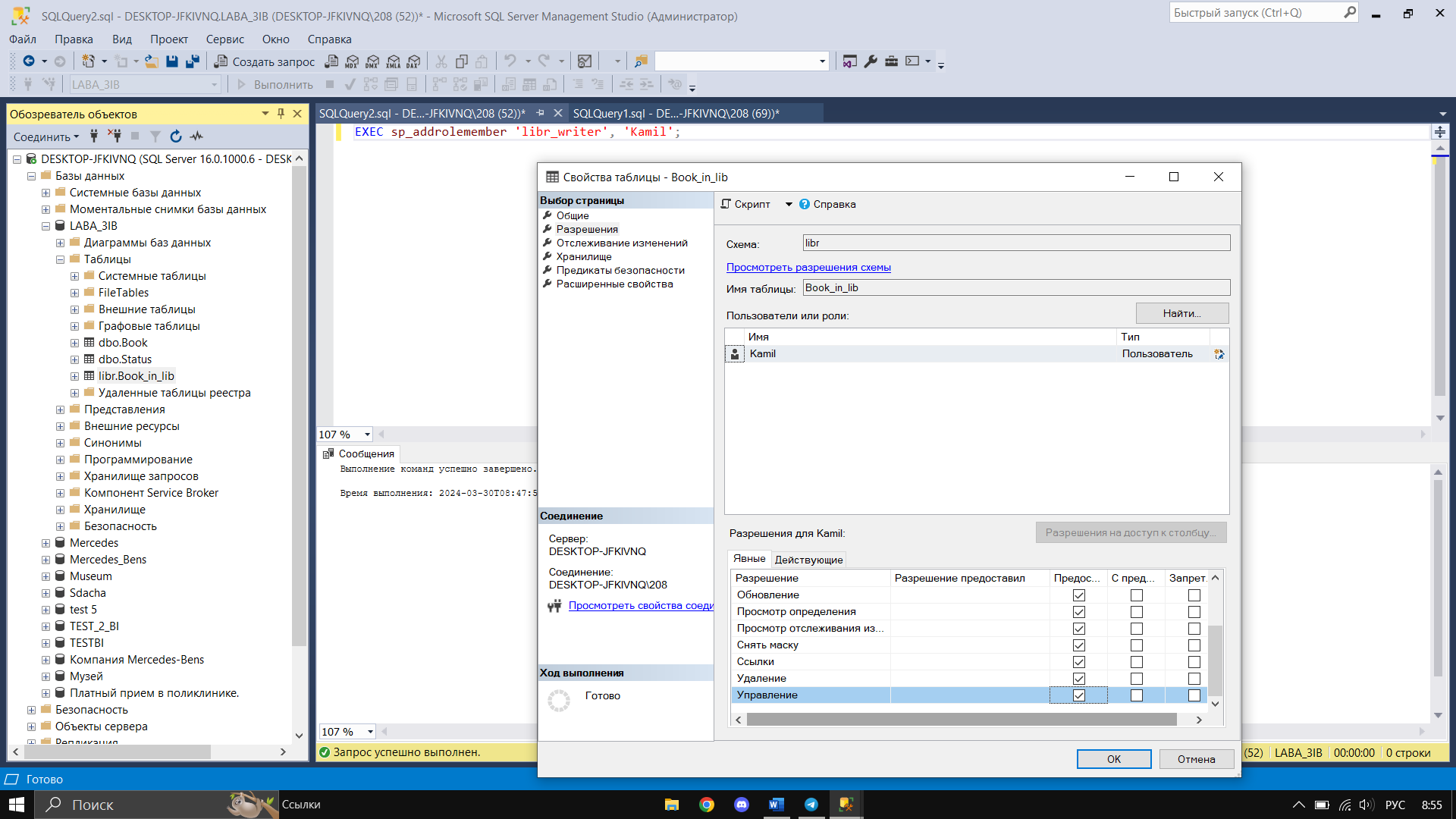


Рисунок 19. – Настройка разращений.

**Шаг 8.**

Выполним запрос на предоставление прав:

В открытом запросе введем следующий запрос, чтобы предоставить пользователю "Kamil" право на обновление столбца "Title" в таблице "Book":

GRANT UPDATE ON dbo.Book(Title) TO Kamil;

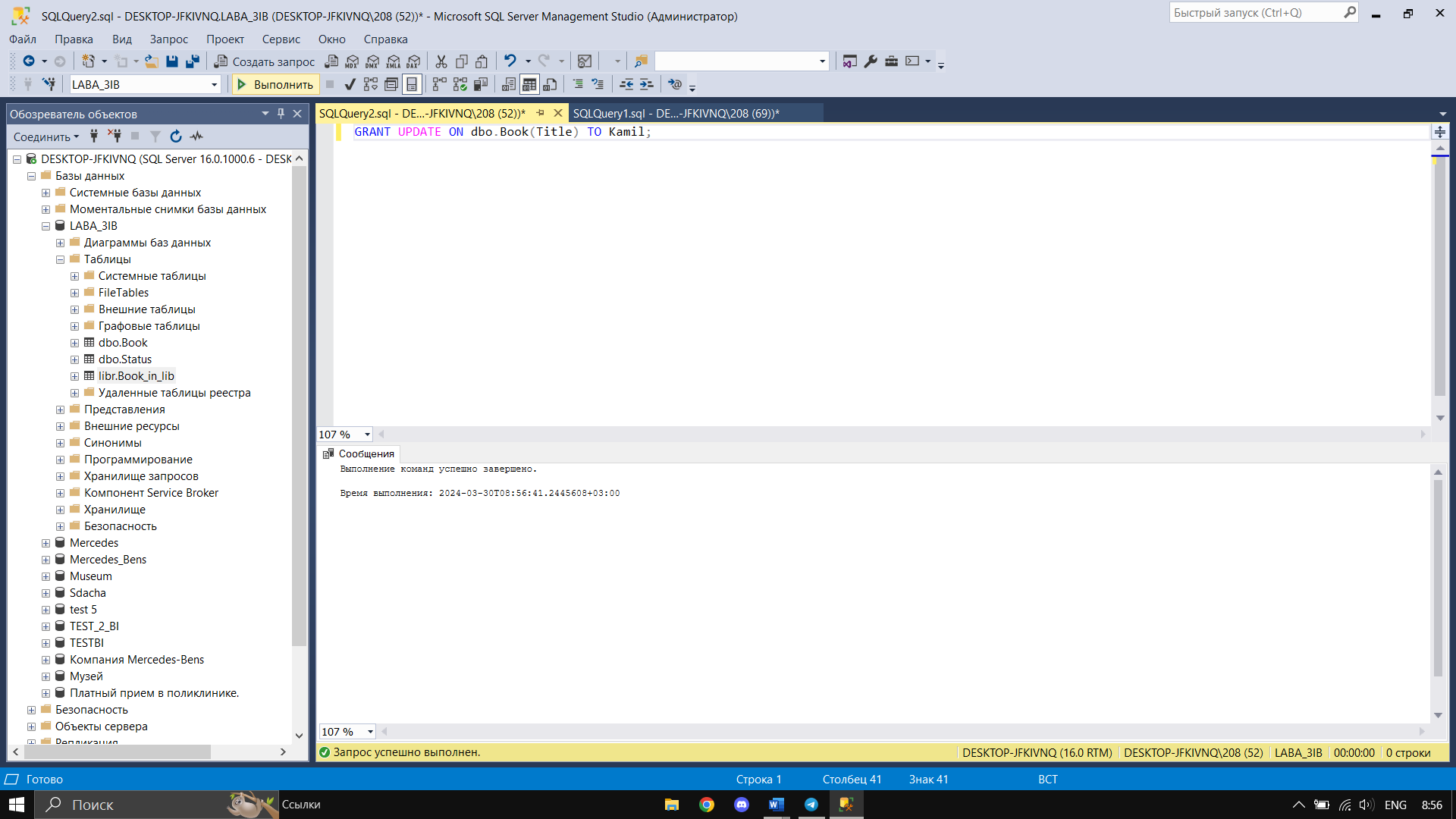


Рисунок 20. – Представление право.

После выполнения запроса попробуем выполнить операцию обновления значения в столбце "Title" таблицы "Book".

UPDATE dbo.Book SET Title = 'Новое название' WHERE book\_id = 1;

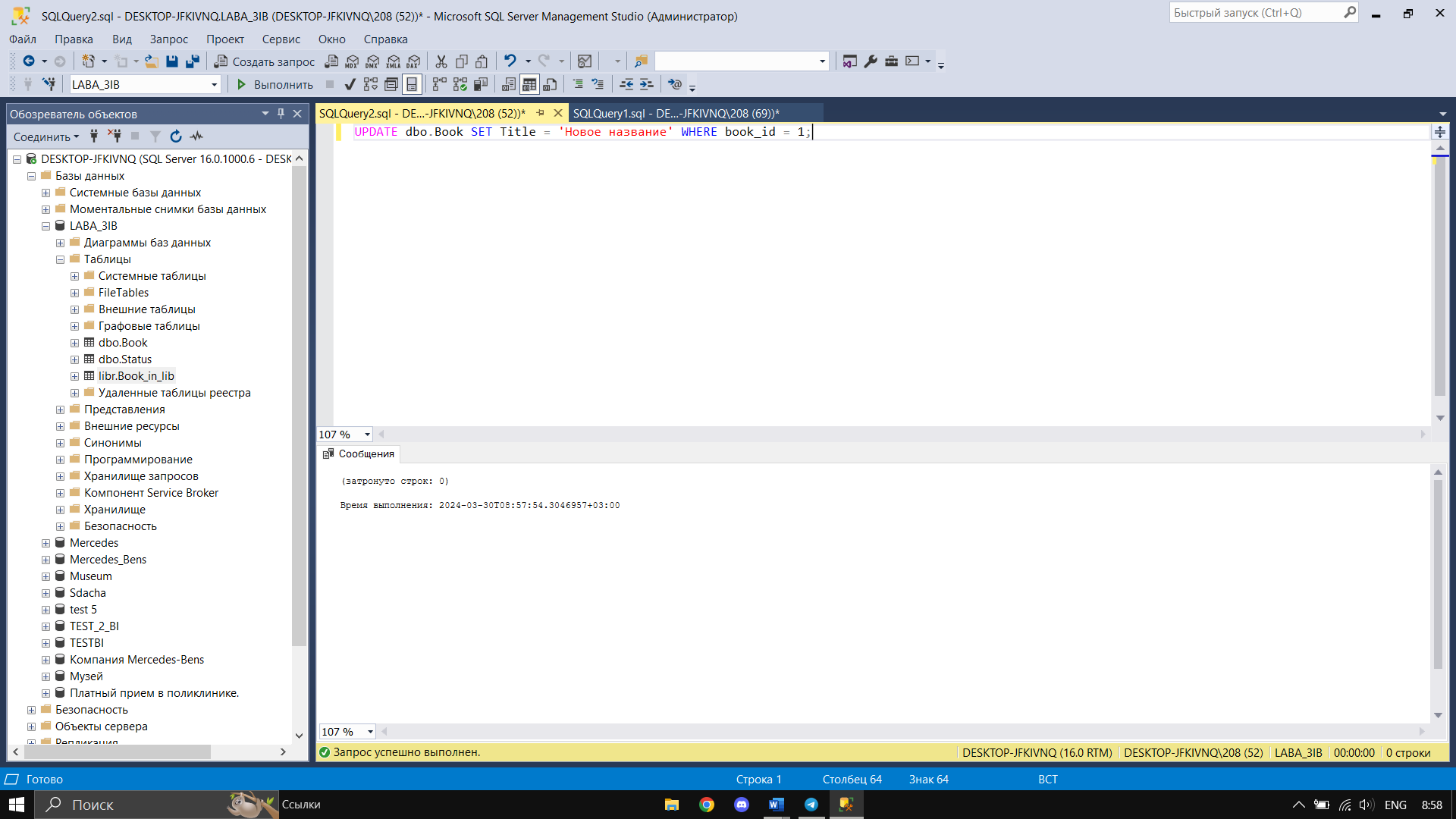


Рисунок 21. – Обновление столбцов.

Проверим результат:

Если запрос обновления выполнится успешно без ошибок доступа, это означает, что пользователь "Kamil" успешно получил право на изменение столбца "Title".

**Шаг 9**

Создание представления и предоставление прав на изменение и добавление записей:

Изучение формата оператора CREATE VIEW:

Первым делом, изучим формат оператора CREATE VIEW и особенности его использования. Обратим внимание на синтаксис и дополнительные параметры, такие как параметры безопасности.

Создание представления:

В открытом запросе выполним оператор CREATE VIEW для создания представления. Выберите из таблицы "Book" только те книги, которые были изданы не ранее 2000 года. Пример создания представления:

CREATE VIEW dbo.NewBooks AS

SELECT \*

FROM dbo.Book

WHERE [Year] >= 2000;

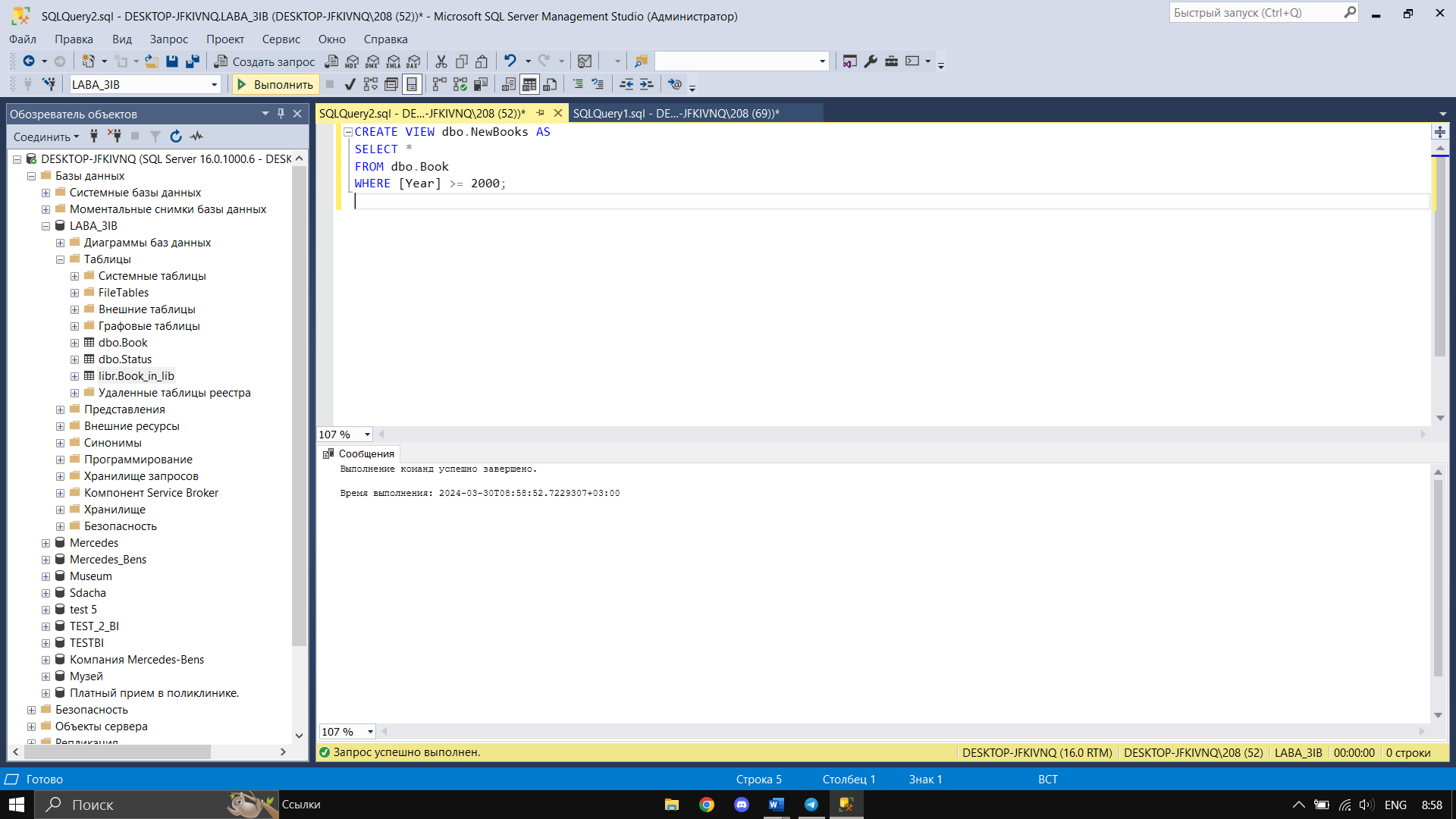


Рисунок 22. – Создание представления.

Предоставление прав на изменение и добавление записей:

Теперь предоставляем пользователю "Kamil" права на изменение и добавление записей в представление "NewBooks", но только для книг, изданных не ранее 2000 года.

GRANT INSERT, UPDATE ON dbo.NewBooks TO Kamil;

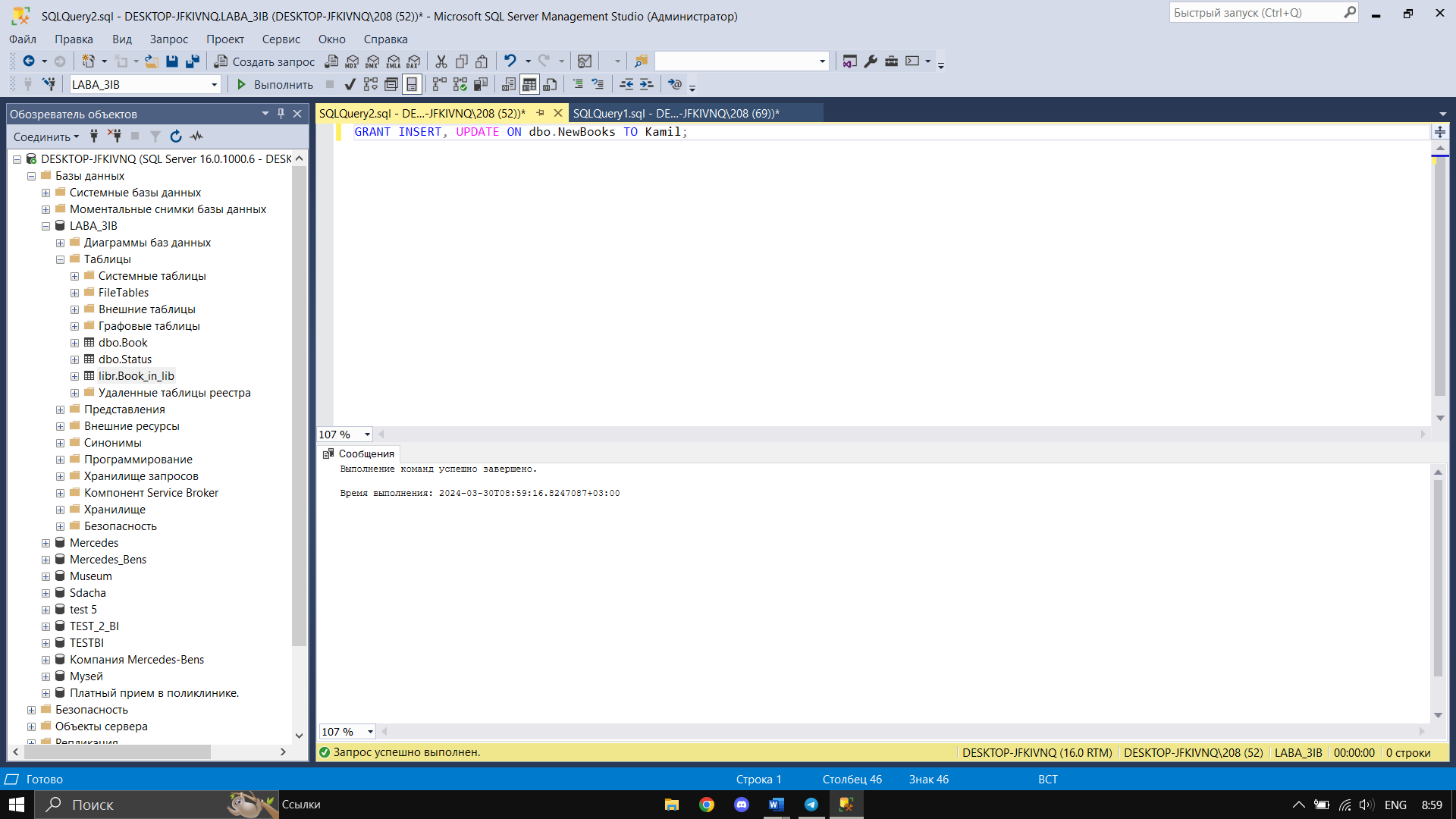


Рисунок 23. – Права на изменение записей.

Проверка прав пользователя:

Убедимся, что пользователь "Kamil" имеет права на изменение и добавление записей в представление "NewBooks". Это можно проверить, выполнив следующий запрос:

SELECT HAS\_PERMS\_BY\_NAME('dbo.NewBooks', 'OBJECT', 'INSERT') AS can\_insert,

HAS\_PERMS\_BY\_NAME('dbo.NewBooks', 'OBJECT', 'UPDATE') AS can\_update;

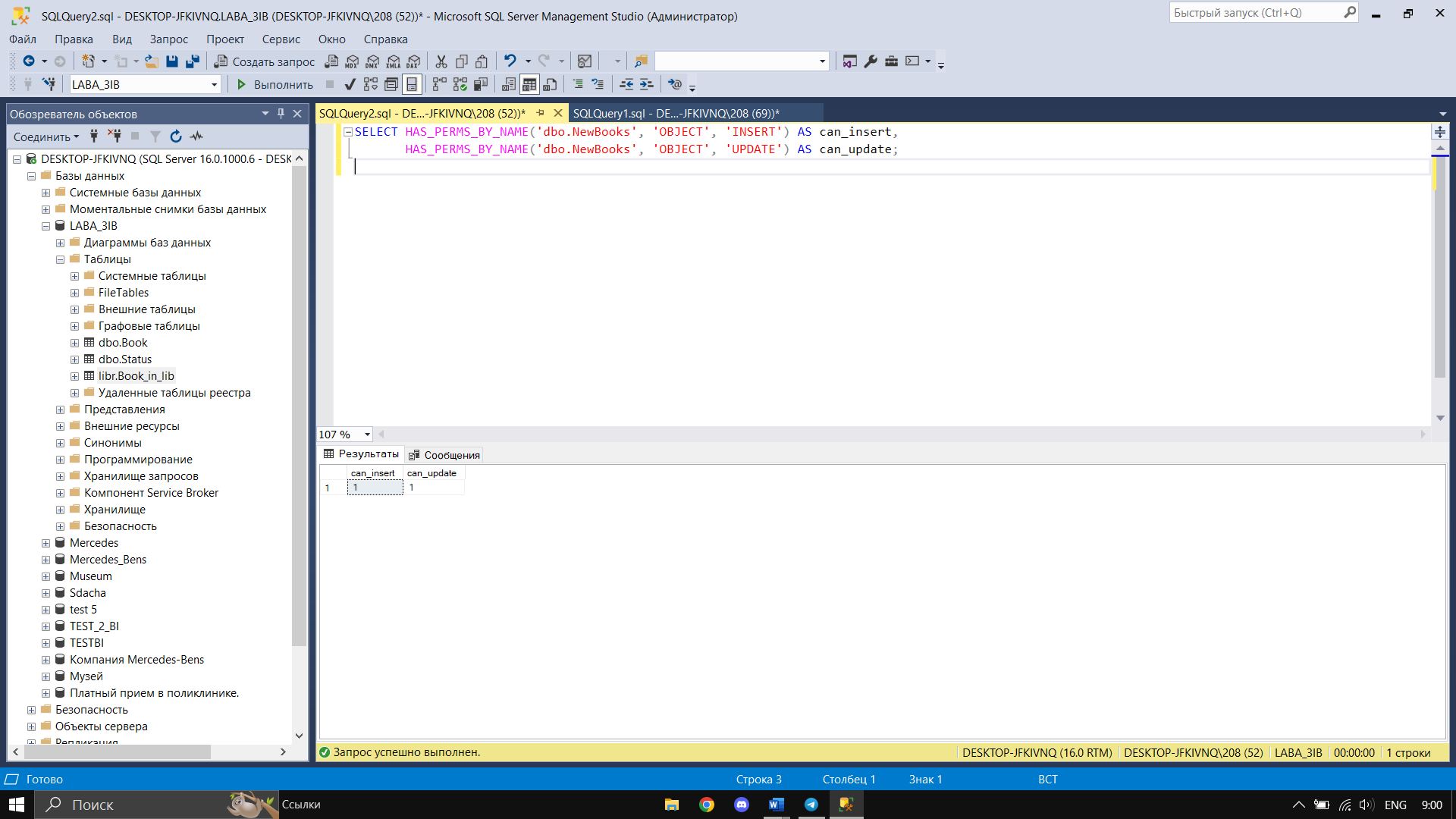


Рисунок 24. – Проверка прав пользователя.

Если для обеих операций выводится значение "1", это означает, что пользователь "Kamil" имеет соответствующие права.

Вывод: Научился выполнять резервное копирование и восстановление баз данных из резервной копии.