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. 4235

Якимова Е. А.

Преподаватель

Кожевников К. Д.

Казань 2024

**Цель**

Научиться реализовывать доступ пользователей к базе данных.

**Результат выполнения работы**

**Шаг 1**

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

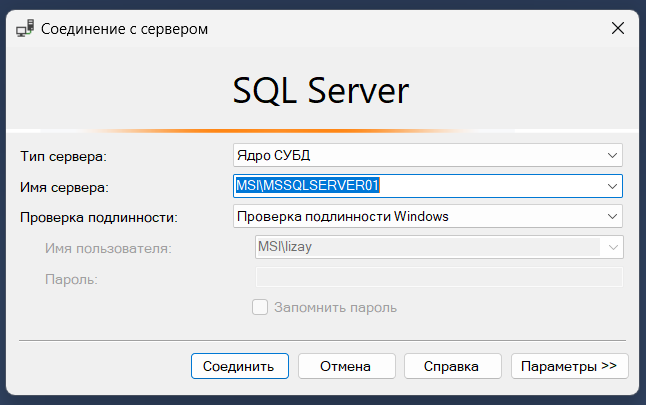


Рисунок 1. – Соединение с сервером.

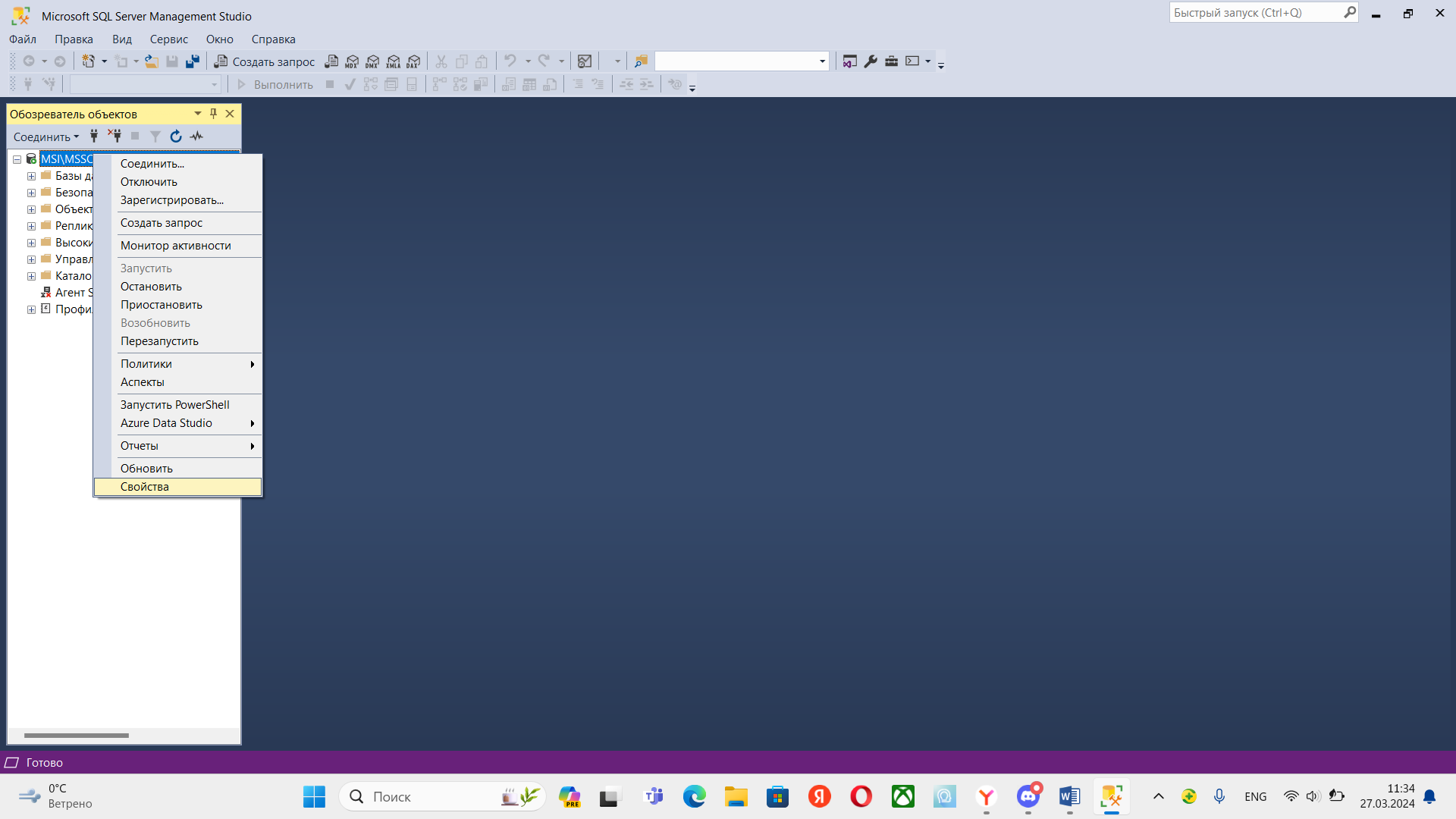
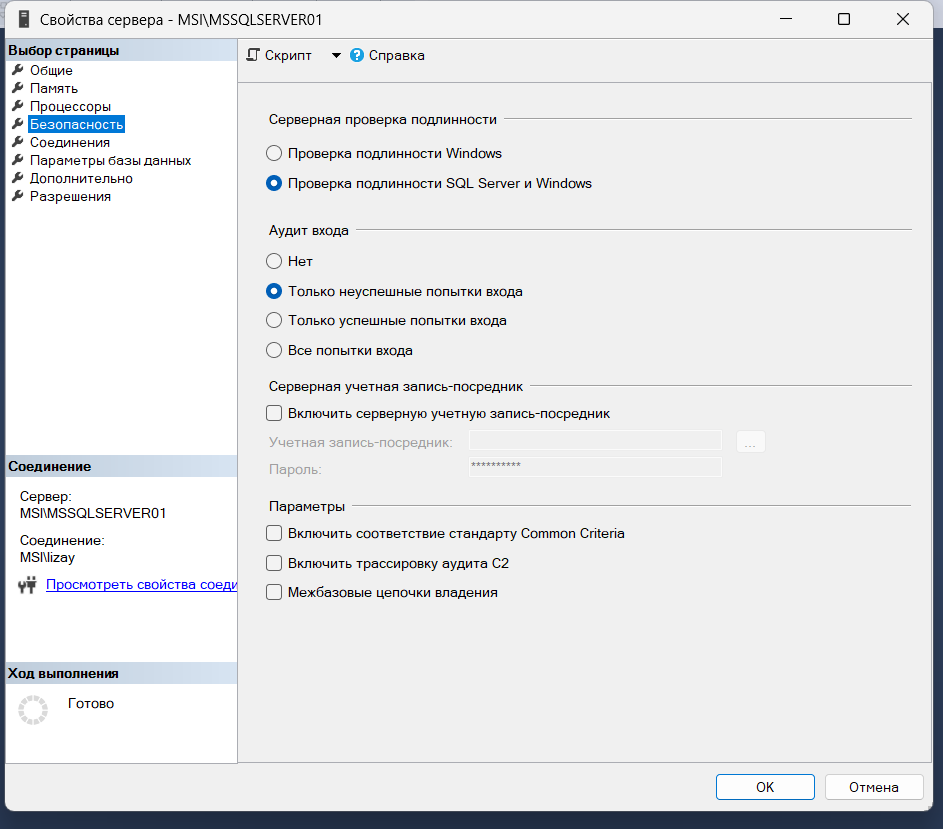


Рисунок 2. – Свойства.

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

Рисунок 3. – Проверка подлинности.

**Шаг 2**

В окне Object Explorer находим и развертываем дерево объектов. Находим папку "Security" (Безопасность) в дереве объектов. В папке "Security" разворачиваем подпапку "Logins" (Учетные записи). Находим свою учётную запись (lizay), щелкаем правой кнопкой мыши на учетную запись и выбираем опцию "Properties" (Свойства).

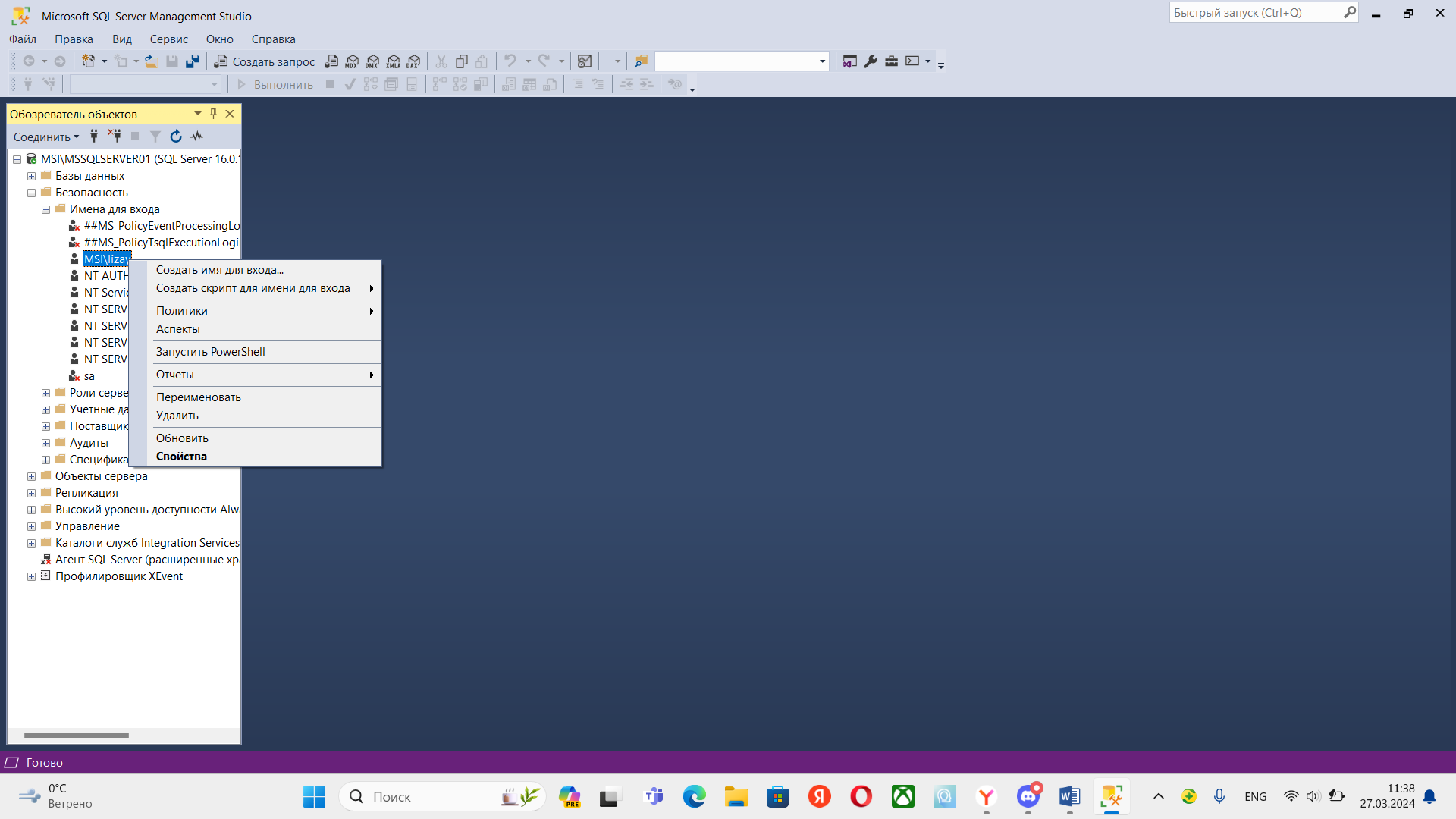


Рисунок 4. - Свойства

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

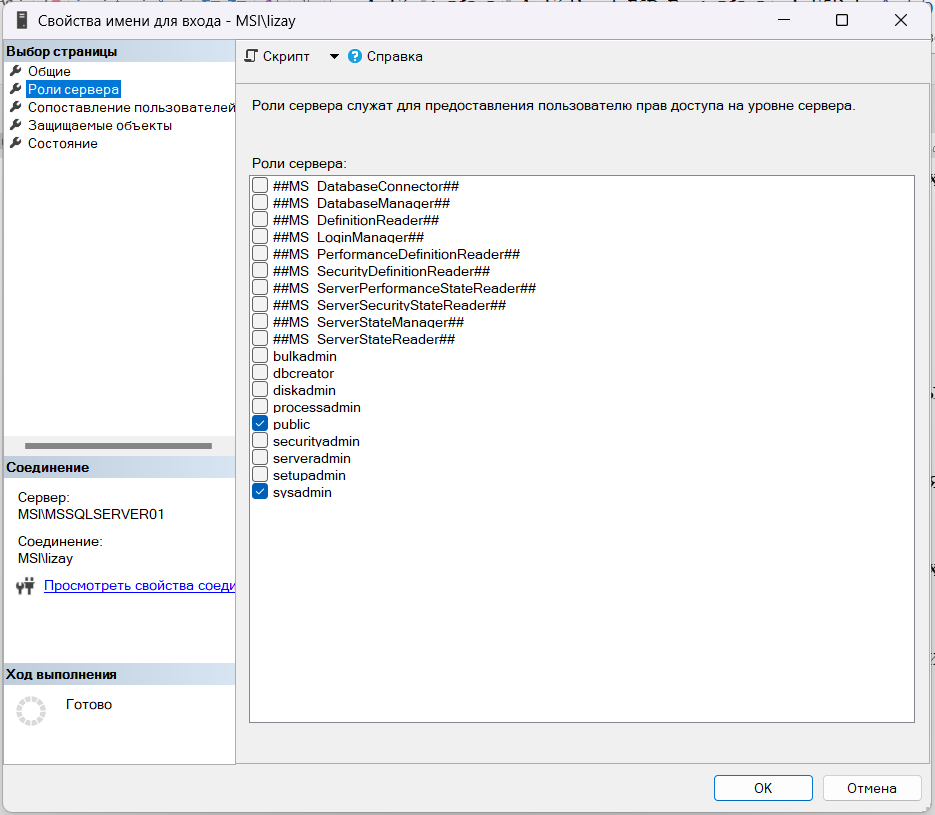
****

Рисунок 5. – Проверка роли сервера.

**Шаг 3**

Находим в окне Object Explorer наш сервер SQL и раскрываем его для просмотра баз данных, разворачиваем папку "Databases" (Базы данных). Находим нужную нам БД, разворачиваем внутри БД папку "Security". Находим папку "Users" (Пользователи), в ней нашу учетную запись и выбираем пользователя, "Properties" (Свойства). В результате узнаем, на выполнение каких ролей мы авторизованы в БД.

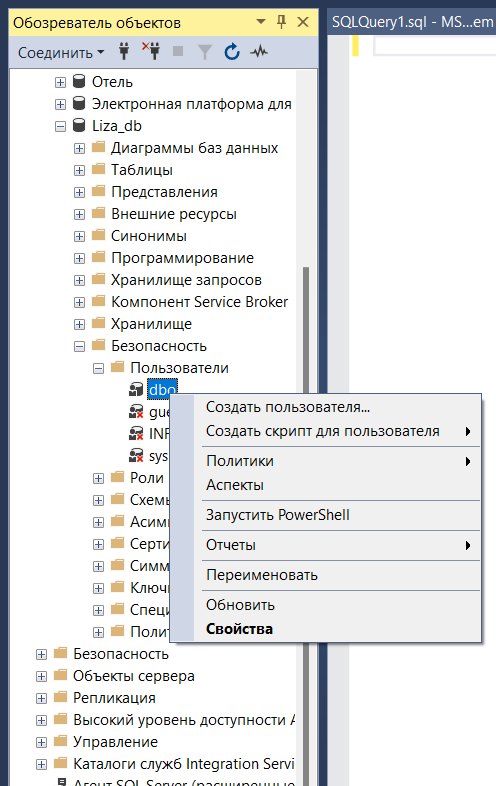


Рисунок 6. – Свойства.

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

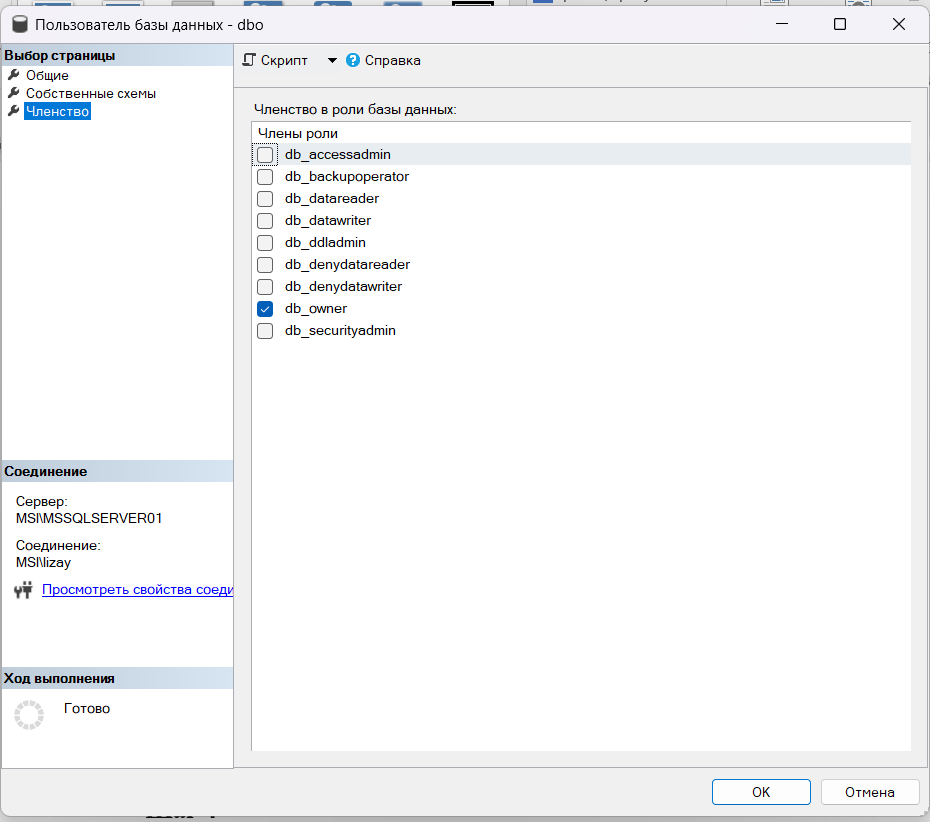


Рисунок 7. – Членство.

**Шаг 4**

В среде Management Studio создаём новую базу данных.

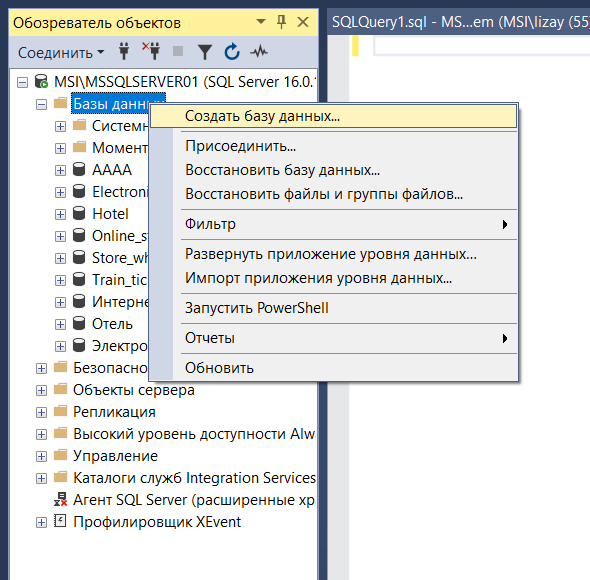
****

Рисунок 8. – Создание БД.

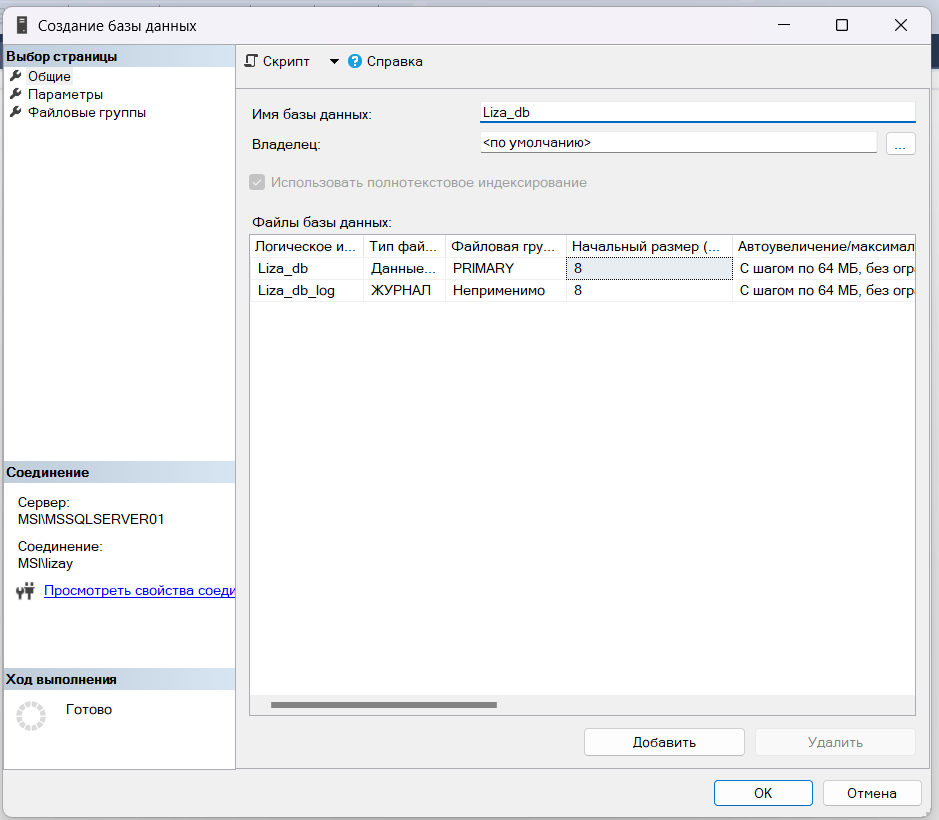
****

Рисунок 9. – Имя БД.

Проверяем список пользователей и ролей: находим в созданной нами БД папку "Security" и разворачиваем ее.

Посмотрите список пользователей, сопоставленных с вашей новой базой данных. Находим пользователя "dbo" (Database Owner) и переходим в "Properties" (Свойства).

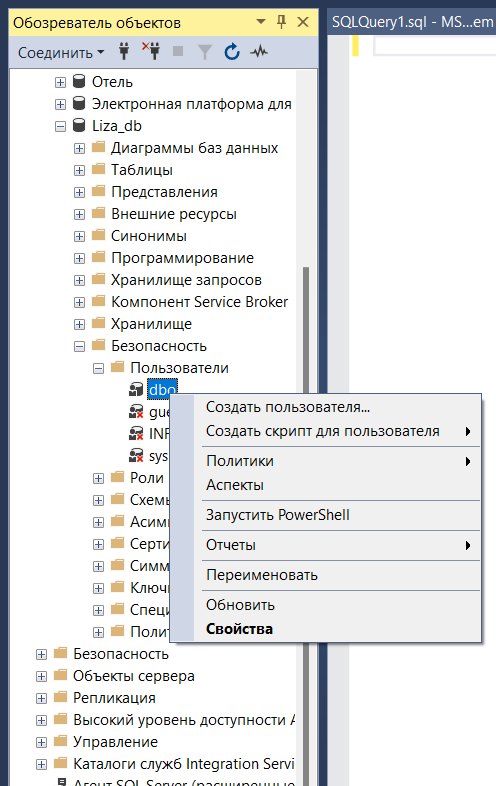


Рисунок 10. – Свойства dbo.

Переходим на вкладку "Membership" (Членство) в окне свойств пользователя "dbo". Убеждаемся, что роль "db\_owner" присутствует в списке ролей, к которым наш пользователь авторизован.

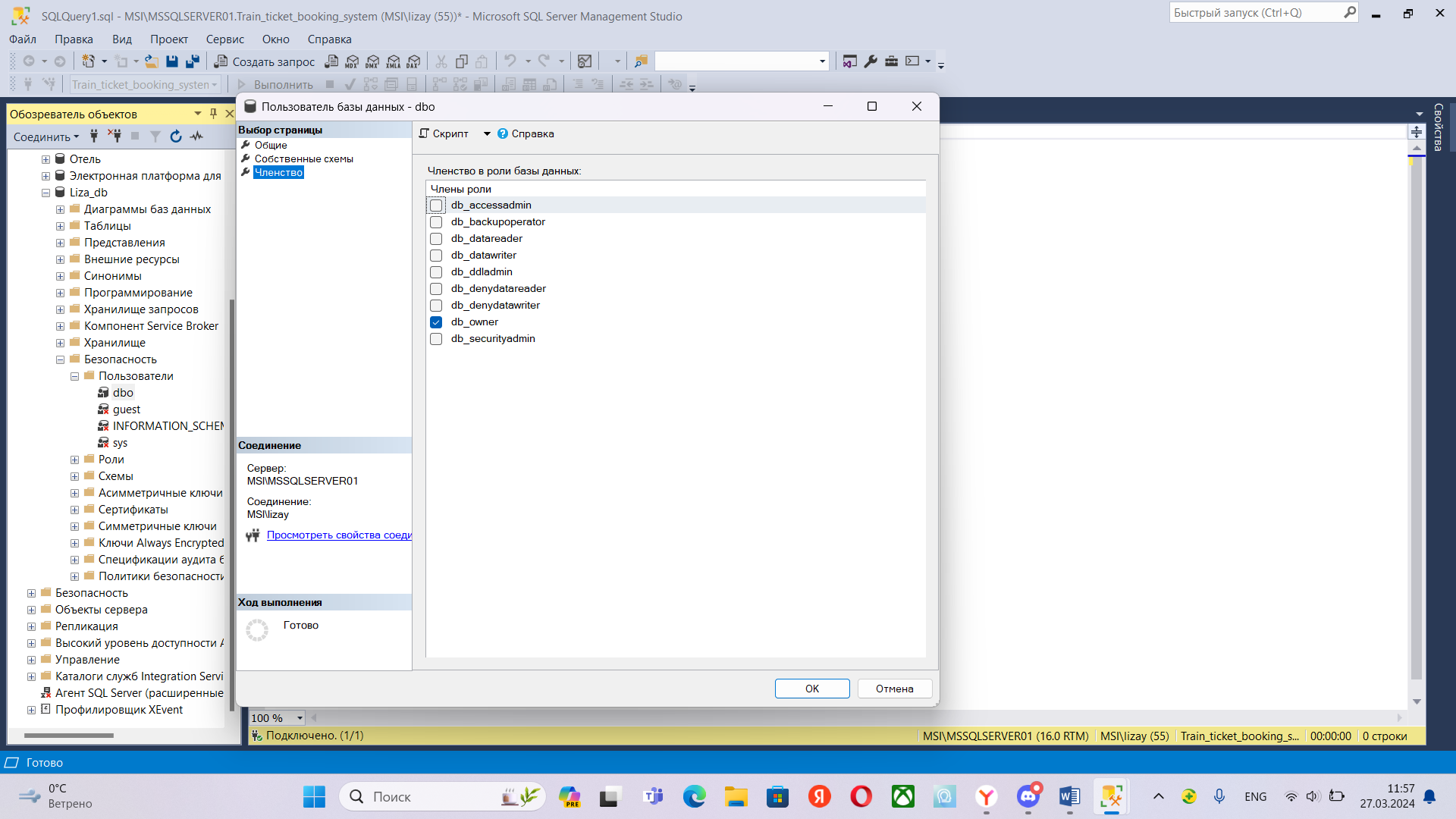
****

Рисунок 11. – Членство.

**Шаг 5**

Создаём в базе данных таблицы.

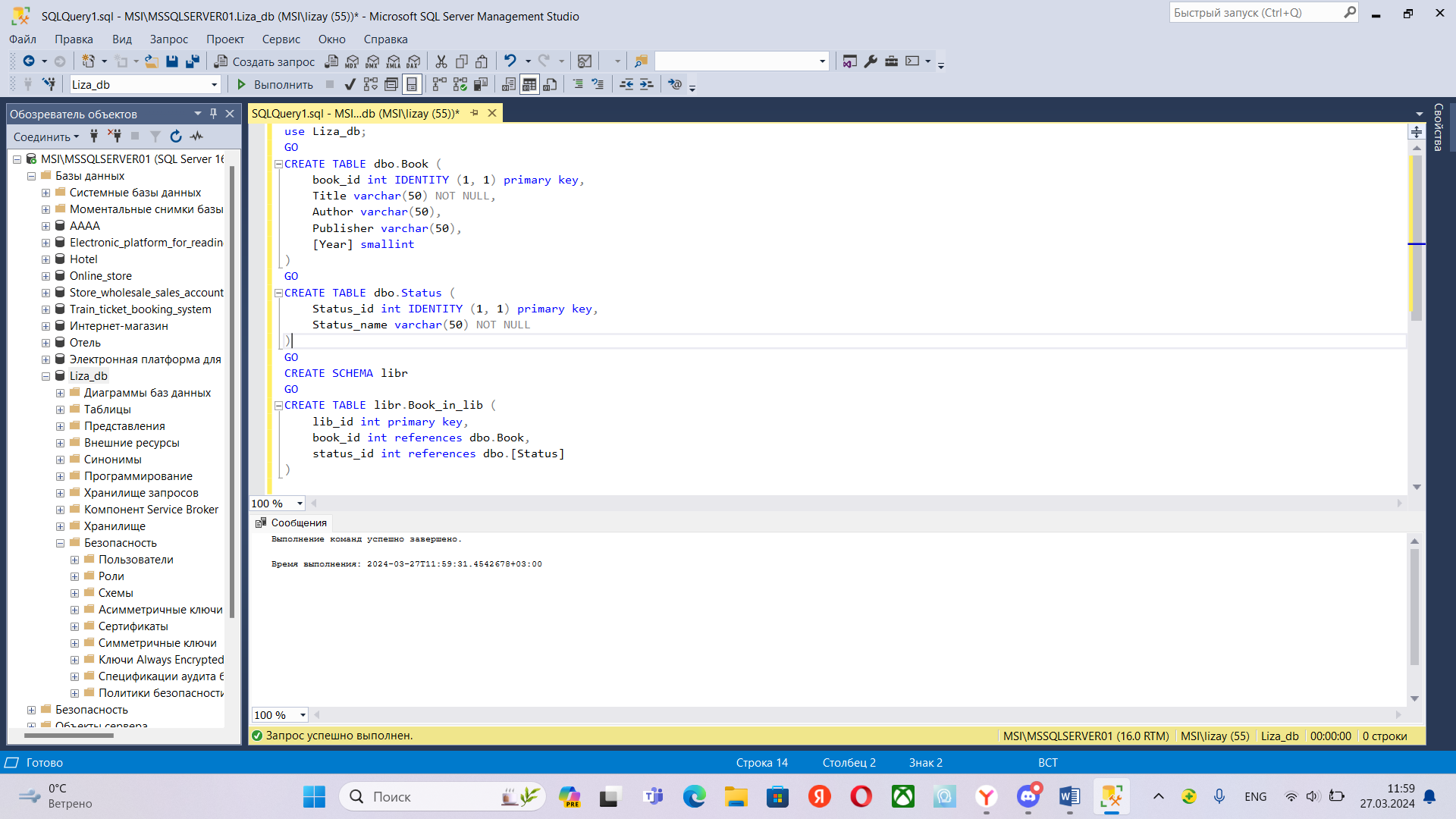


Рисунок 12. – Создание таблиц.

**Листинг:**

use Liza\_db;

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]

)

Проверяем, что таблицы создались.

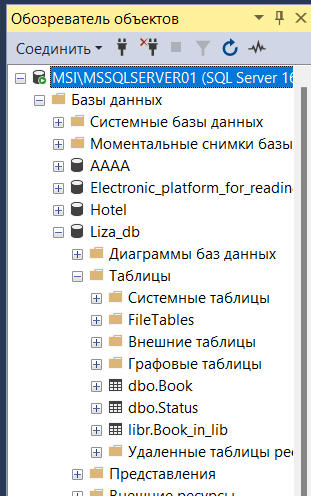


Рисунок 13. – Проверка.

**Шаг 6.**

Создаём пользователя, выбираем схему dbo, добавляем в роль db\_datareader:

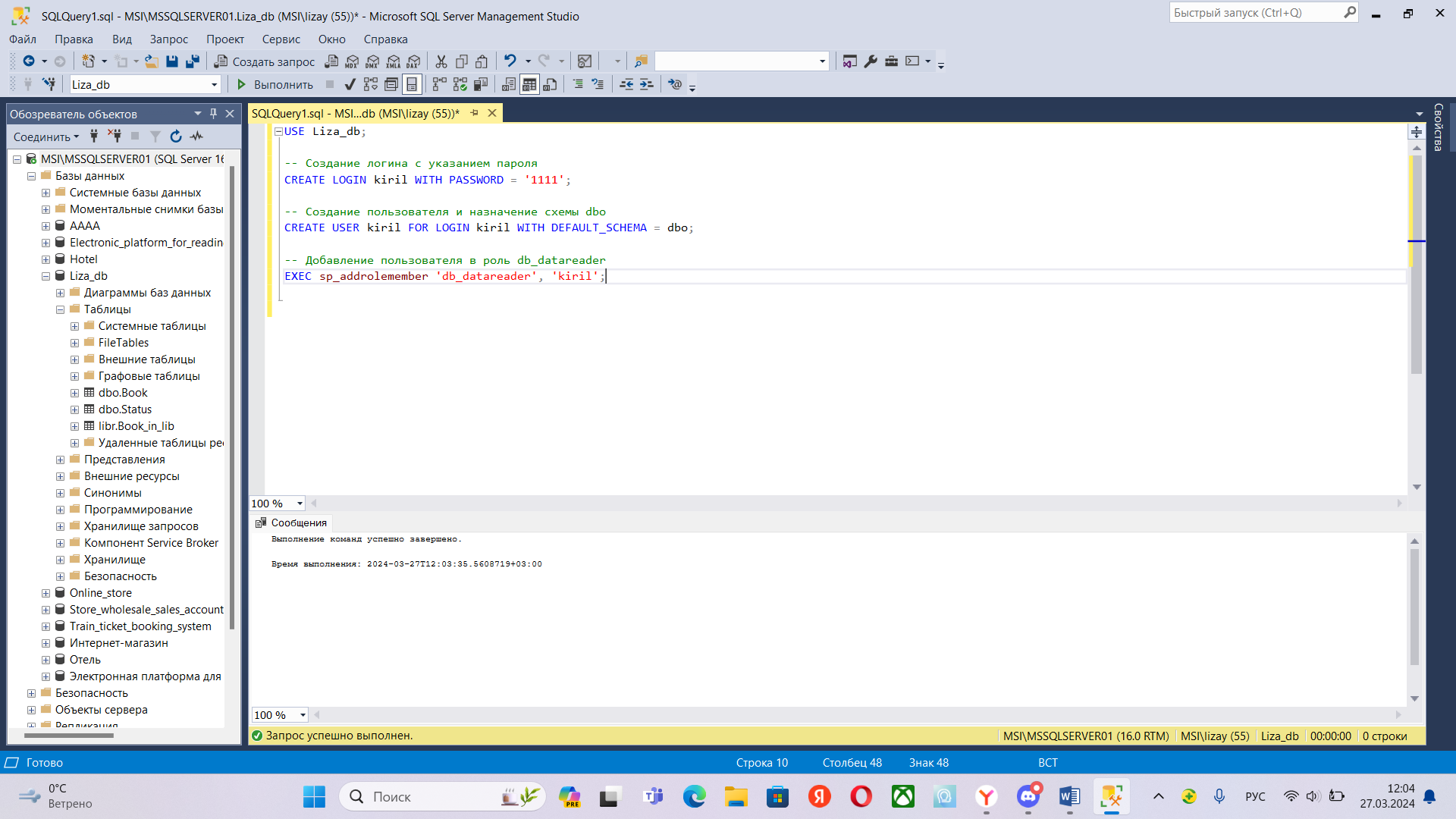
****

Рисунок 14. – Скрипт.

**Листинг:**

USE Liza\_db;

-- Создание логина с указанием пароля

CREATE LOGIN kiril WITH PASSWORD = '1111';

-- Создание пользователя и назначение схемы dbo

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

-- Добавление пользователя в роль db\_datareader

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

Тестирование:

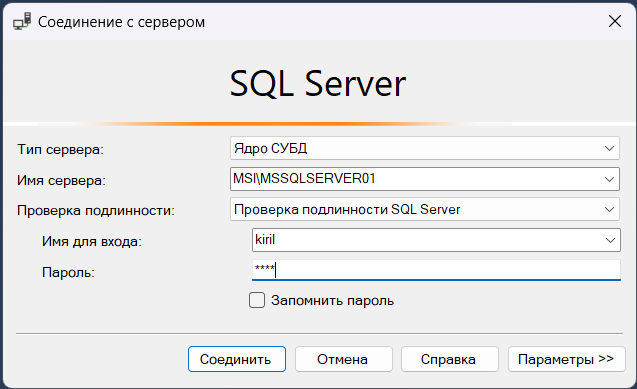


Рисунок 15 – Соединение с сервером под новым именем.

Убеждаемся, что новый пользователь имеет доступ только на чтение данных из всех таблиц базы данных.

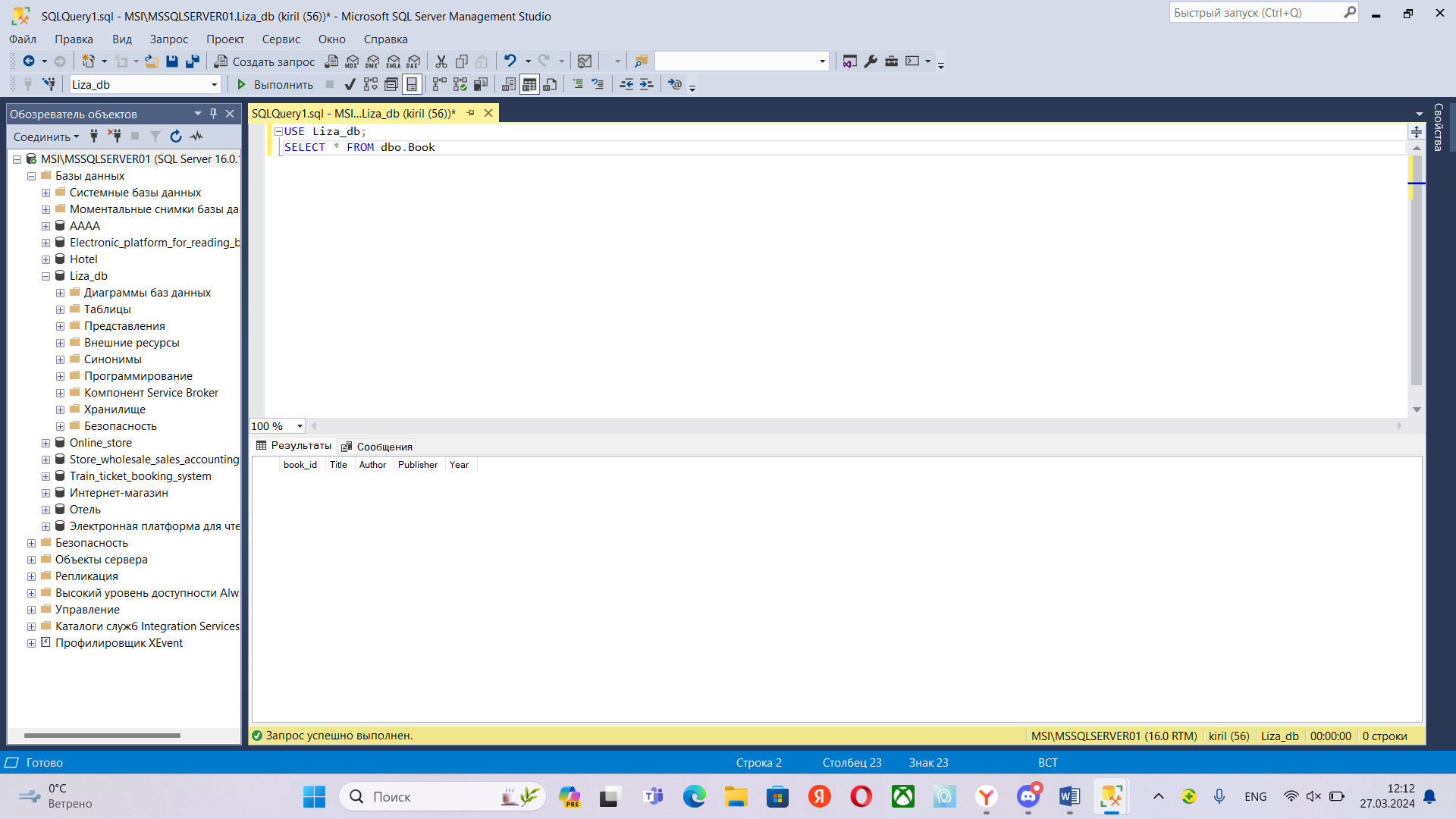
****

Рисунок 16. – Проверка со стороны пользователя.

**Шаг 7.**

Создаём новые роли и назначаем разрешения:

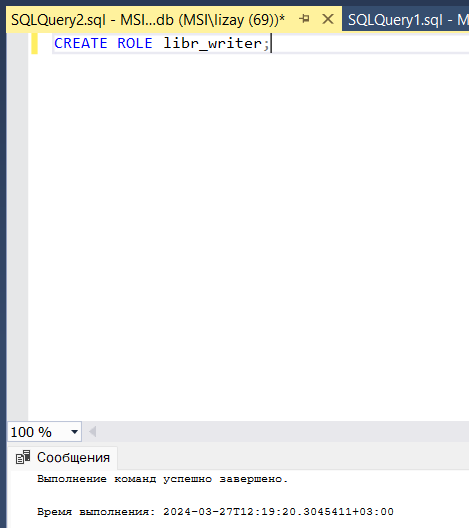


Рисунок 17. – Новая роль.

**Листинг:**

CREATE ROLE libr\_writer;

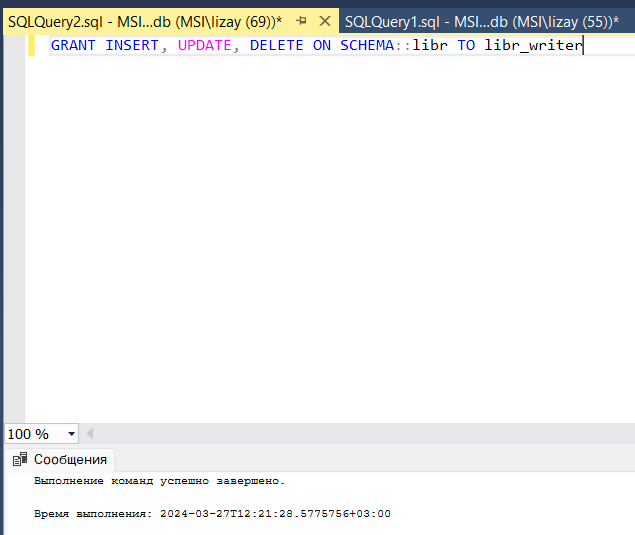


Рисунок 18. – Разращение роли.

**Листинг:**

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

Добавляем пользователя "kiril" в роль "libr\_writer":

Этот запрос назначает пользователя "kiril" в роль "libr\_writer", что дает ему возможность изменять данные в объектах схемы "libr".

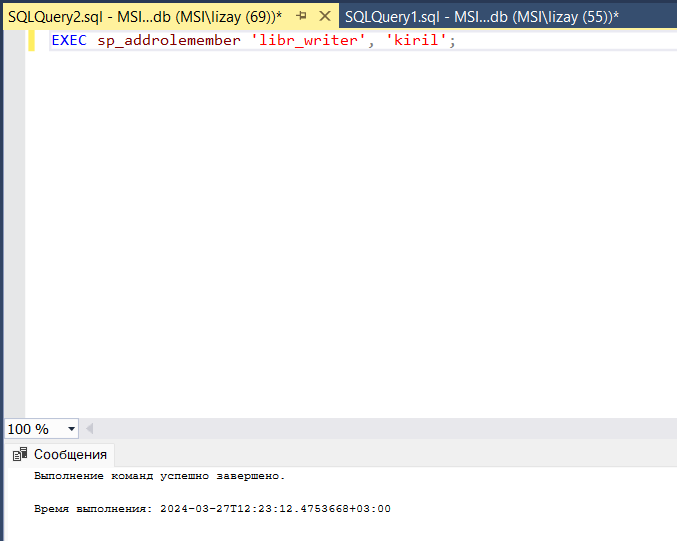


Рисунок 19. – Назначение роли.

**Листинг:**

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

Проверяем доступ пользователя к таблице: выполняем операции INSERT, UPDATE и DELETE для данных в таблице "Book\_in\_lib", принадлежащей схеме "libr".

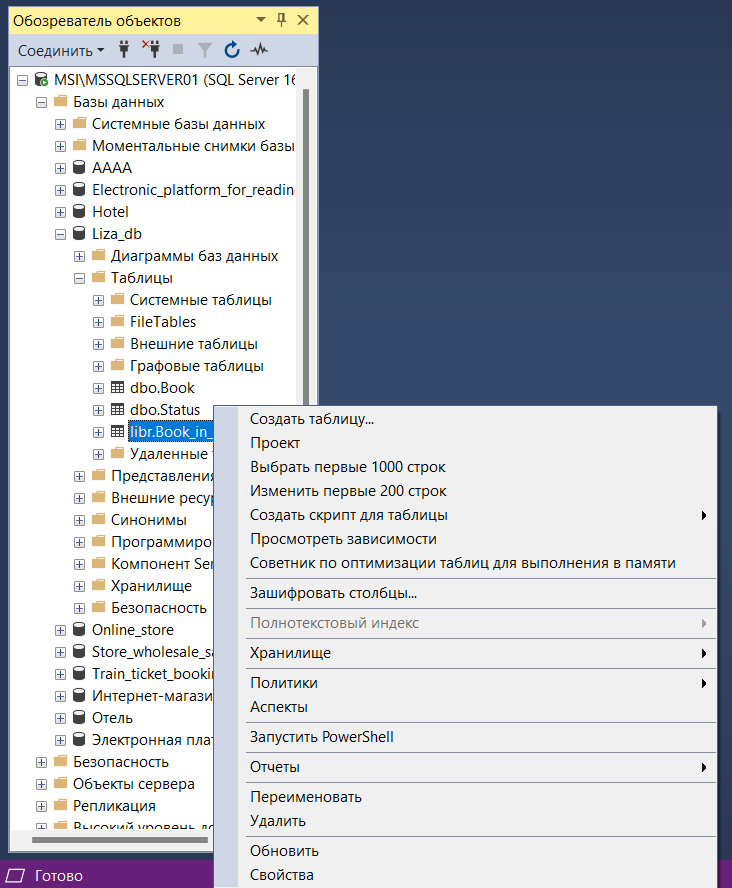


Рисунок 20. – Свойства.

Назначаем разрешения для пользователя “kiril”.

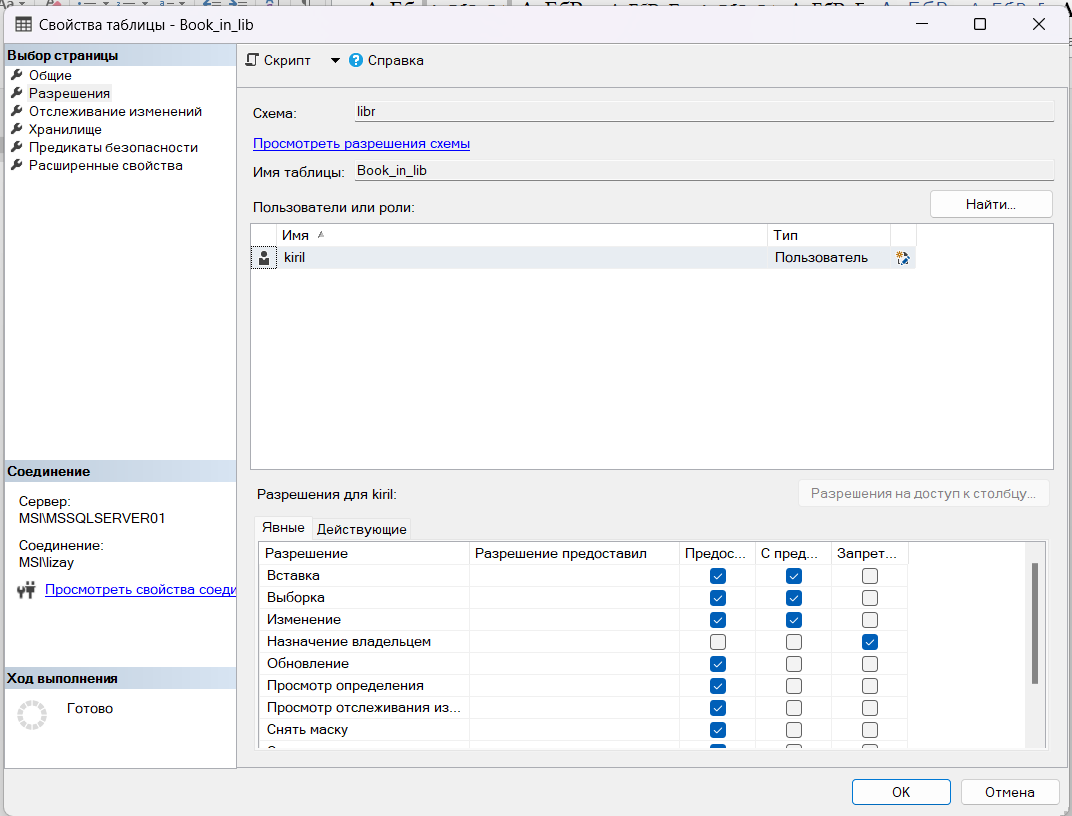


Рисунок 21. – Настройка разращений.

**Шаг 8.**

Предоставляем пользователю права на изменение отдельных столбцов.

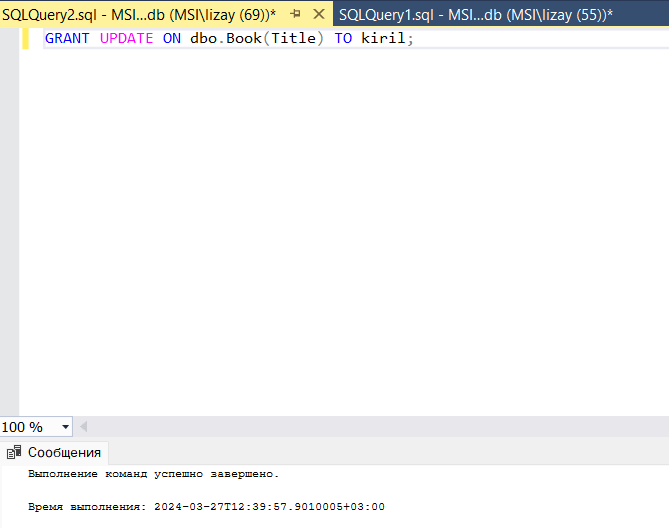


Рисунок 22. – Предоставление прав.

**Листинг:**

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

Выполняем операцию обновления значения в столбце "Title" таблицы "Book".

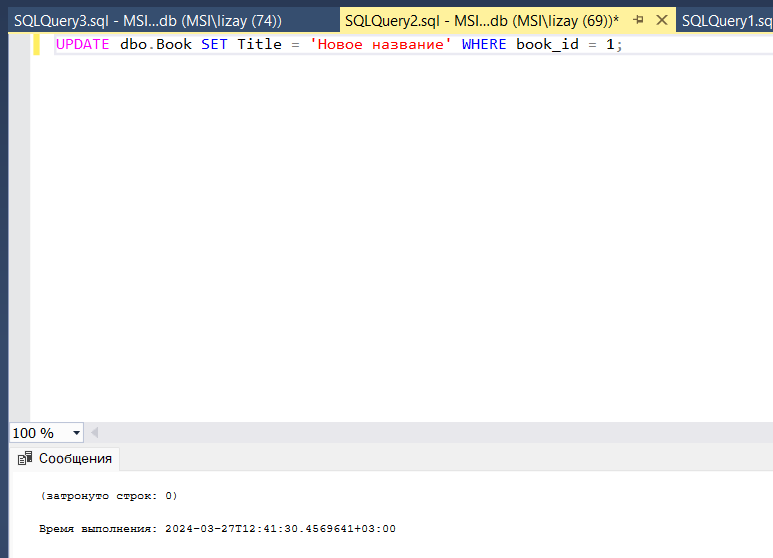


Рисунок 23. – Обновление столбцов.

**Листинг:**

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

**Шаг 9.**

Создаём представления и предоставляем права на изменение и добавление записей.

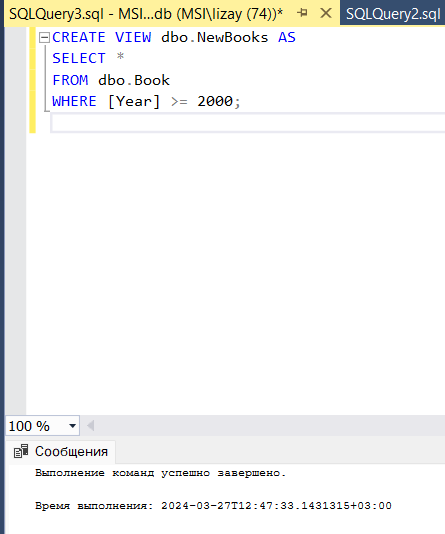


Рисунок 24. - Создания представления.

**Листинг:**

CREATE VIEW dbo.NewBooks AS

SELECT \*

FROM dbo.Book

WHERE [Year] >= 2000;

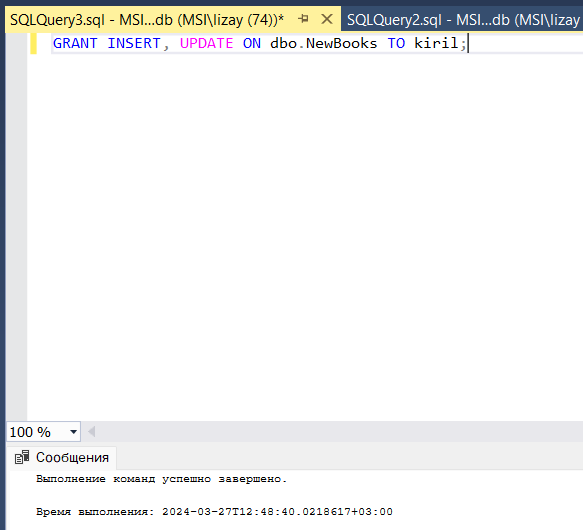


Рисунок 25. – Права на изменение записей.

**Листинг:**

GRANT INSERT, UPDATE ON dbo.NewBooks TO kiril;

Проверяем права пользователя "kiril".

Для обеих операций выводится значение "1", что значит: пользователь "kiril" имеет соответствующие права.

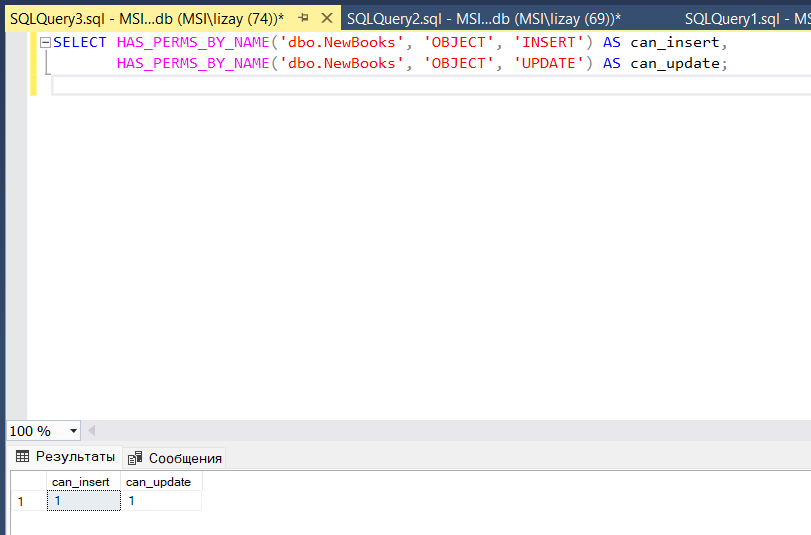


Рисунок 26. – Проверка прав пользователя.

**Листинг:**

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

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

**Вывод:**

Я научилась предоставлять пользователям доступ к БД.