

**МИНОБРНАУКИ РОССИИ**  
**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ**  
**ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**  
**«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)**  
**Кафедра Систем автоматизированного проектирования**

**ОТЧЕТ**  
**по лабораторной работе №4**  
**по дисциплине «Базы данных»**  
**Тема: Модификация данных**

Студенты гр. 2308

---

---

---

Попов Н.А.

Бебия Р.А.

Чиков А.А.

Преподаватель

---

Горяинов С.В.

Санкт-Петербург

2024

## Цель работы

Научиться выполнять команды вставки, удаления и обновления данных. В лабораторной работе используется база данных Library.

## Выполнение работы

### Упражнение 1 – применение команды INSERT

В этом задании необходимо использовать команду INSERT для добавления строк в таблицы БД Library. После выполнения вставки данных надо написать запрос для проверки сделанных изменений

Запрос 1: Вставка строк в таблицу item.

USE library

```
INSERT INTO item (isbn, Title_no, Cover, Loanable, Translation)
VALUES (10001, 8, 'HARDBACK', 'Y', 'ENGLISH'),
(10101, 8, 'SOFTBACK', 'Y', 'ENGLISH')
SELECT * FROM item WHERE isbn = 10001 or isbn = 10101
```

Результат выполнения запроса представлен на Рисунке 1.

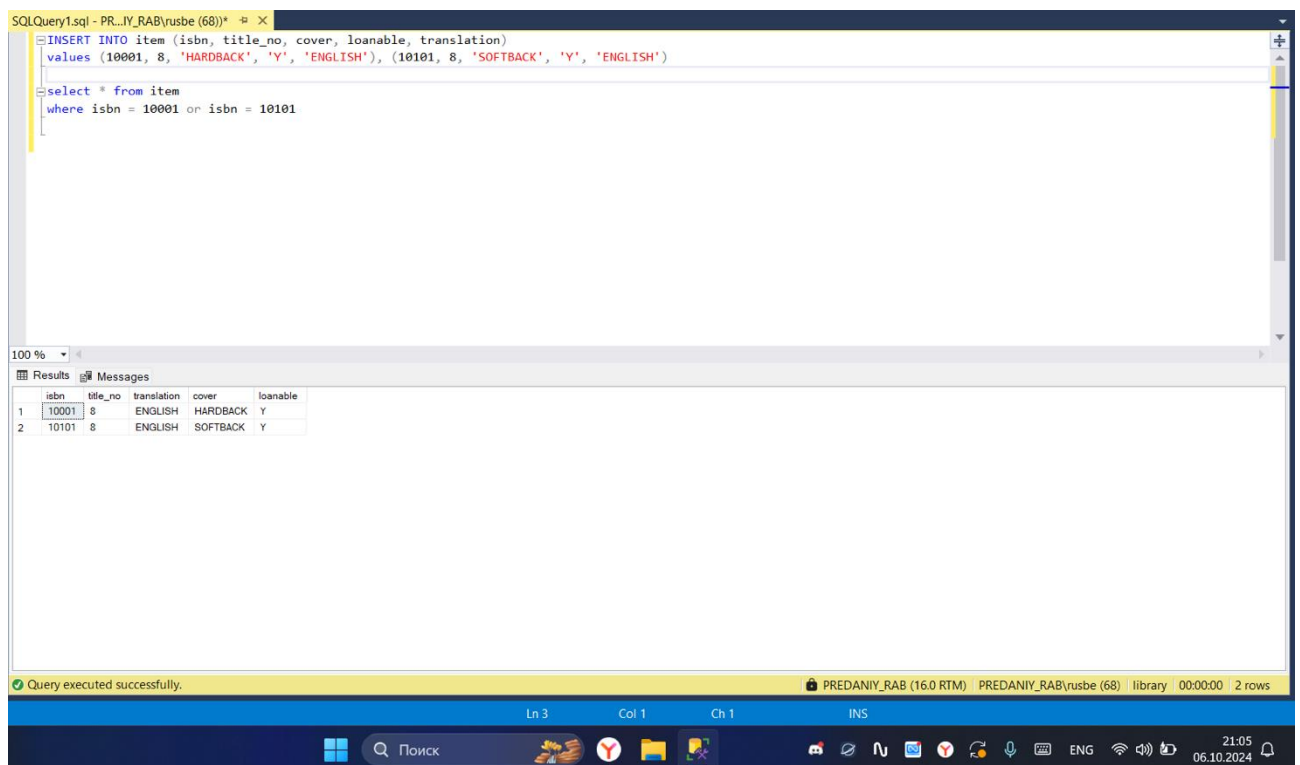


Рисунок 1 – Результат выполнения запроса

Запрос 2: Вставка строк в таблицу copy.

```
INSERT INTO copy (isbn, copy_no, title_no, on_loan)
VALUES (10001, 1, 8, 'N')
SELECT * FROM copy WHERE isbn = 10001
```

Результат выполнения запроса представлен на Рисунке 2.

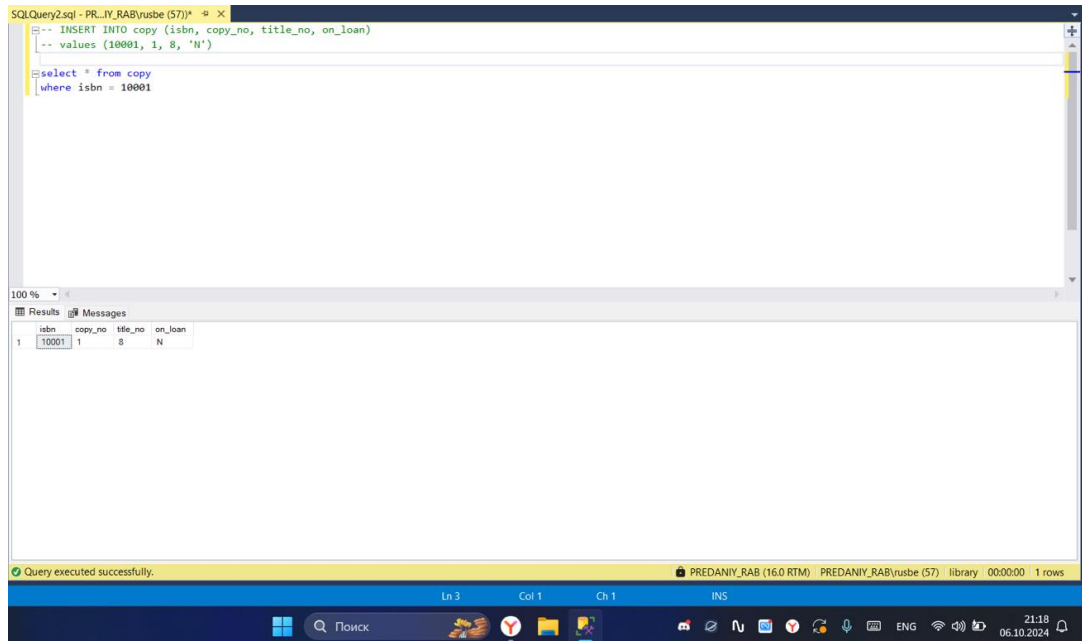


Рисунок 2 – Результат выполнения запроса

Запрос 3: Возвращение одного из добавленных элементов.

```
SELECT translation FROM item WHERE isbn = 10001
```

Результат выполнения запроса представлен на Рисунке 3.

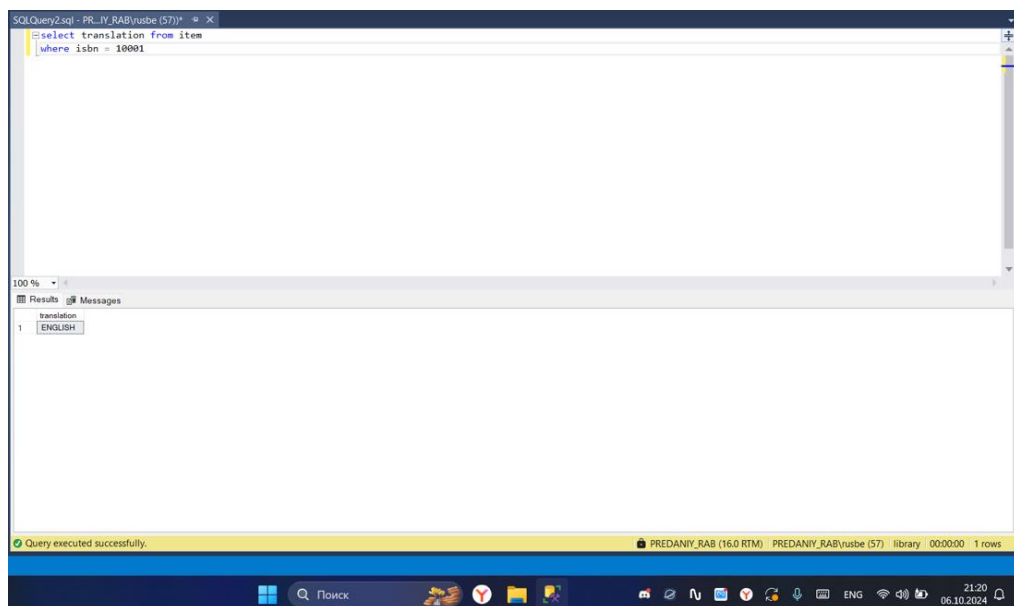


Рисунок 3 – Результат выполнения запроса

## Упражнение 2 – использование команды INSERT с ключевым словом DEFAULT

Запрос 1: Определение столбцов, для которых разрешено значение null.

`sp_help title`

Результат выполнения запроса представлен на Рисунке 4.

Name	Owner	Type	Created_datetime
title	dbo	user table	2006-03-18 21:32:36.297

Column_name	Type	Computed	Length	Prec	Scale	Nullable	TrimTrailingBlanks	FixedLenNullInSource	Collation
title_no	int	no	4	10	0	no	(n/a)	(n/a)	NULL
title	longstring	no	63			no	no	no	Cyrillic_General_CI_AS
author	normstring	no	31			no	no	no	Cyrillic_General_CI_AS
synopsis	text	no	16			yes	(n/a)	(n/a)	Cyrillic_General_CI_AS

Identity	Seed	Increment	Not For Replication
title_no	1	1	0

RowGUIDCol
No rowguidcol column defined.

Data_located_on_filegroup
PRIMARY

index_name	index_description	index_keys
title_idnt	nonclustered, unique, primary key located on PRI...	title_no

constraint_type	constraint_name	delete_action	update_action	status_enabled	status_for_replication	constraint_keys
PRIMARY KEY (non-clustered)	title_idnt	(n/a)	(n/a)	(n/a)	(n/a)	title_no

Table is referenced by foreign key
library.dbo.copy: copy_title_link
library.dbo.item: item_title_link
library.dbo.loan: loan_title_link
library.dbo.loanhist: loanhist_title...

Query executed successfully.

Рисунок 4 – Результат выполнения запроса

Запрос 2: Вставка значений в таблицу title.

`INSERT INTO title (title, author, synopsis)`

`VALUES ('The Art of Lawn Tennis', 'William T. Tilden', DEFAULT)`

`SELECT * FROM title WHERE title = 'The Art of Lawn Tennis'`

Результат выполнения запроса представлен на Рисунке 5.

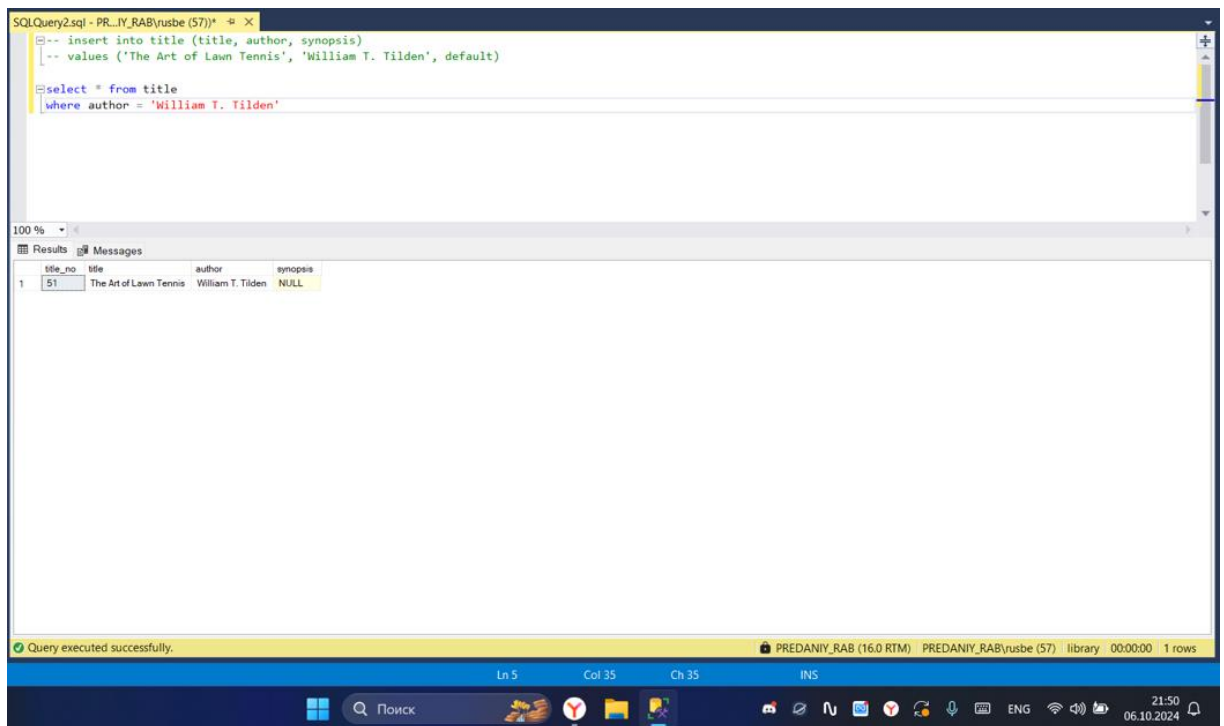


Рисунок 5 – Результат выполнения запроса

Запрос 3: Определение последнего использованного значения IDENTITY.  
 SELECT SCOPE\_IDENTITY() as LastIdentity

Результат выполнения запроса представлен на Рисунке 6.

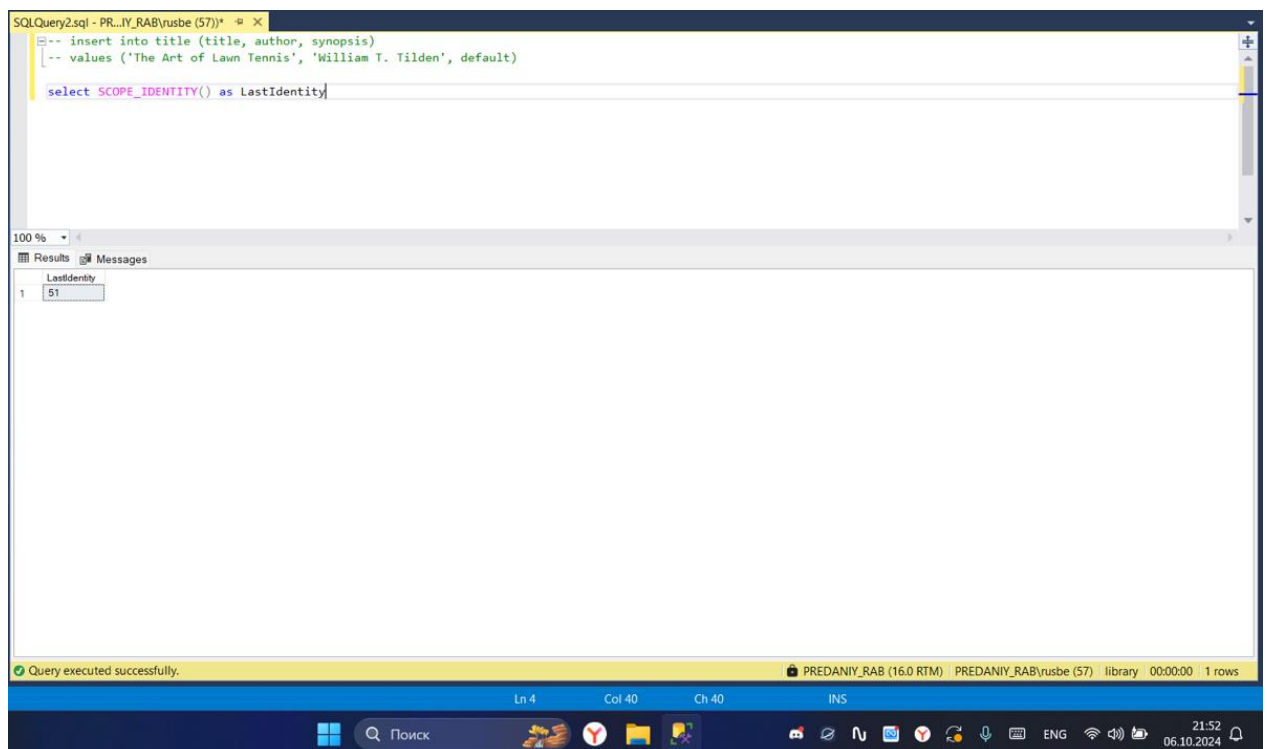


Рисунок 6 – Результат выполнения запроса

Запрос 4: Получение последней вставленной записи в таблице title.

```
SELECT * FROM title  
WHERE title_no = 51
```

Результат выполнения запроса представлен на Рисунке 7.

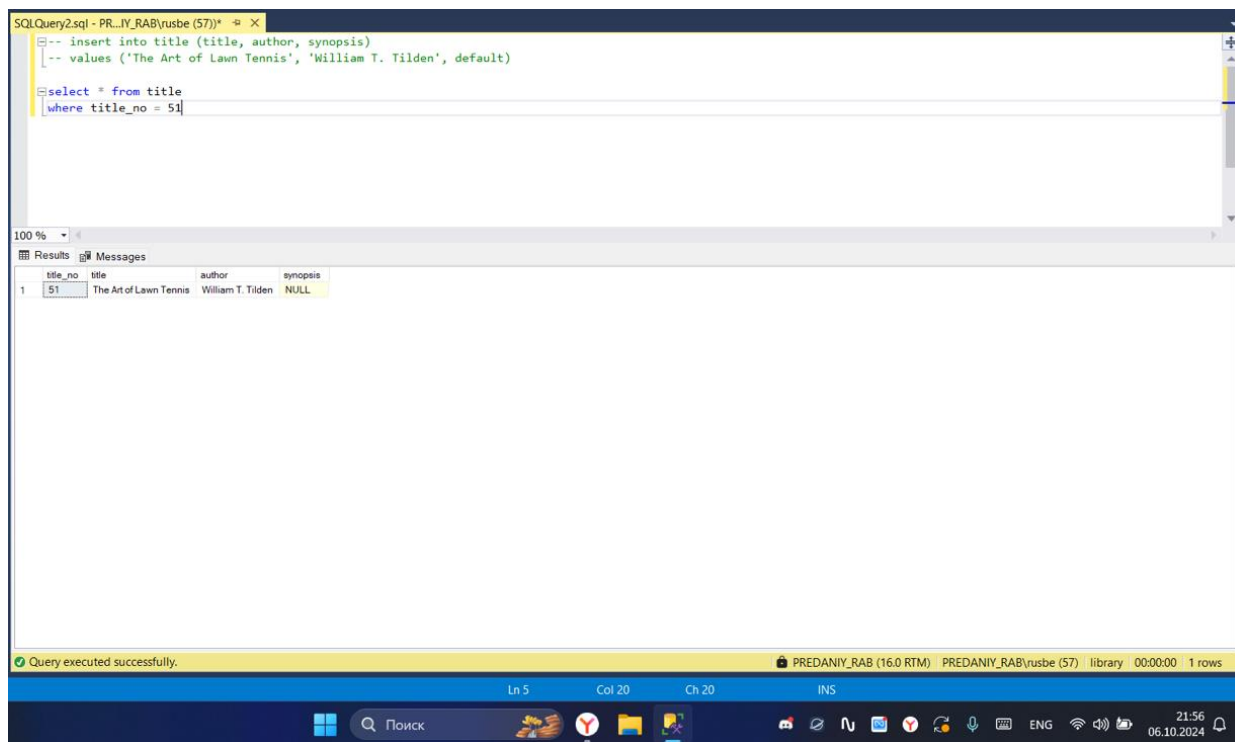


Рисунок 7 – Результат выполнения запроса

Запрос 5: Добавление новых записей в таблицу title.

```
INSERT INTO title (title, author)  
VALUES ('Riders of the Purple Sage', 'Zane Grey')  
SELECT * FROM title  
WHERE title_no = 52
```

Результат выполнения запроса представлен на Рисунке 8.

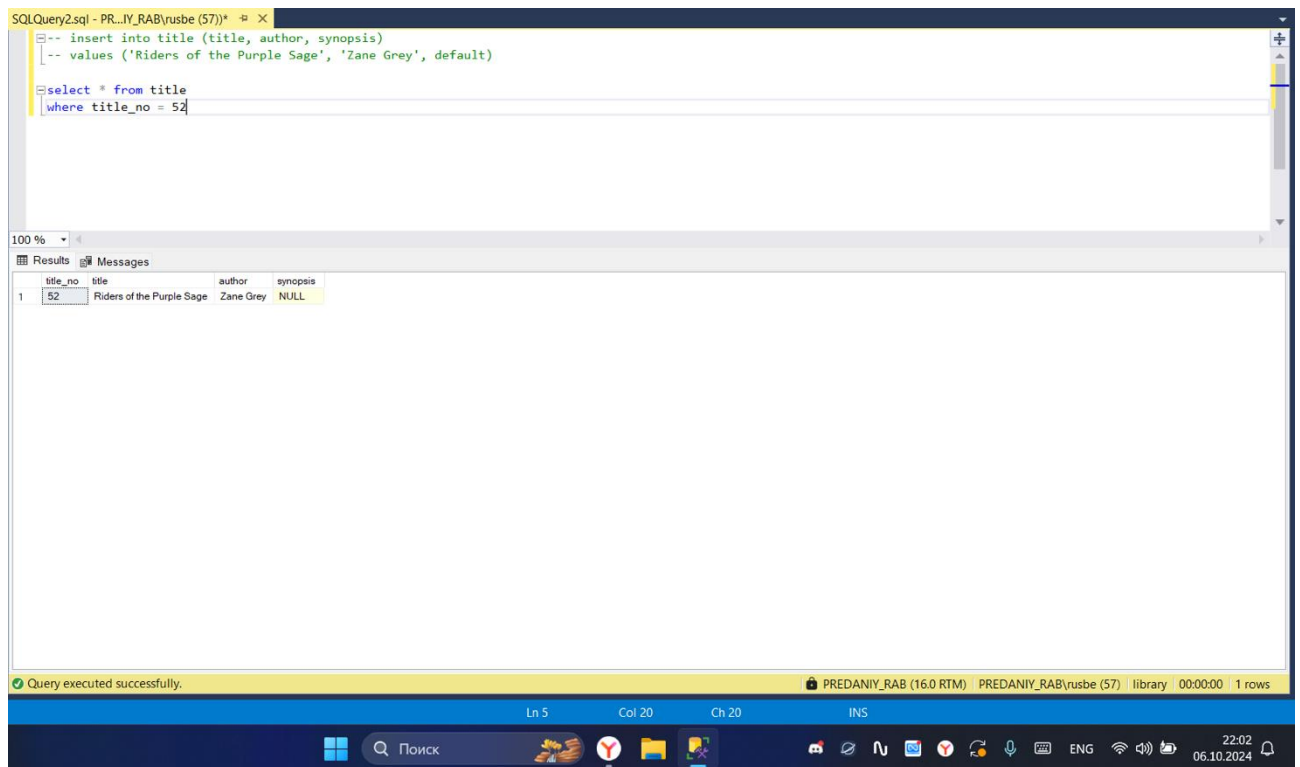


Рисунок 8 – Результат выполнения запроса

### Упражнение 3 – использование команды INSERT с ключевыми словами DEFAULT VALUES

Запрос 1: Создание новой таблицы sample1 и вставка записи со значениями по умолчанию.

```
INSERT INTO sample1 DEFAULT VALUES
```

```
SELECT * FROM sample1
```

Результат выполнения запроса представлен на Рисунке 9.

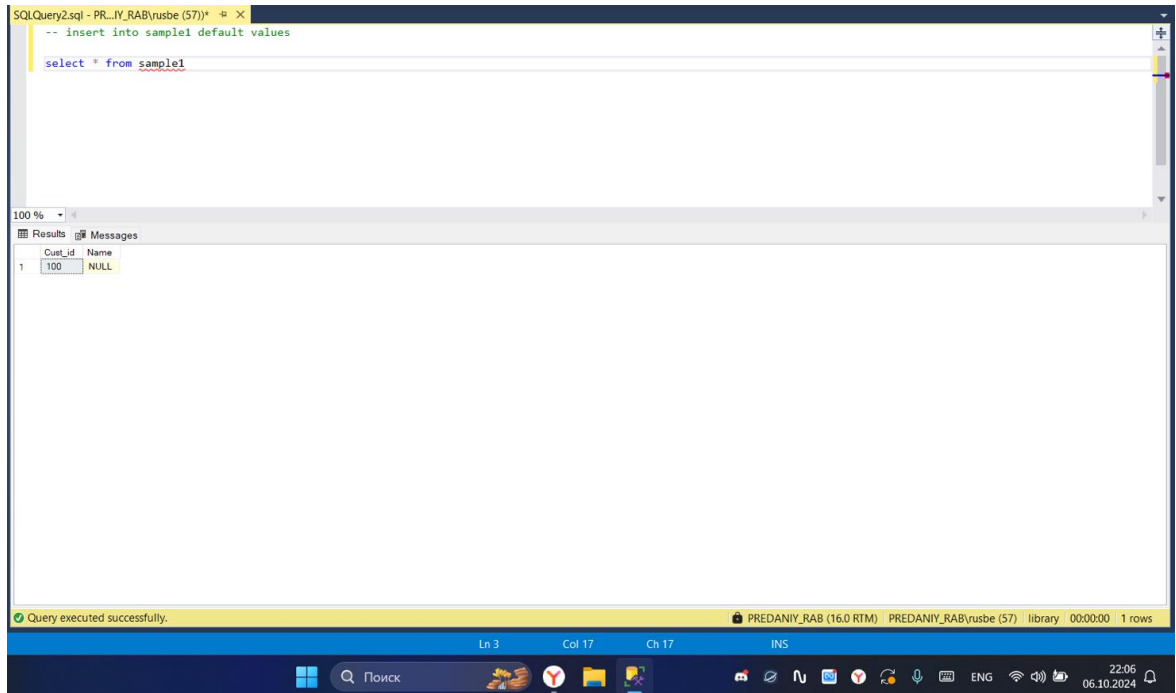


Рисунок 9 – Результат выполнения запроса

#### Упражнение 4 – использование команды DELETE

Запрос 1: Возвращение определенной записи из таблицы item.

**SELECT \* FROM item WHERE isbn=10101**

Результат выполнения запроса представлен на Рисунке 10.

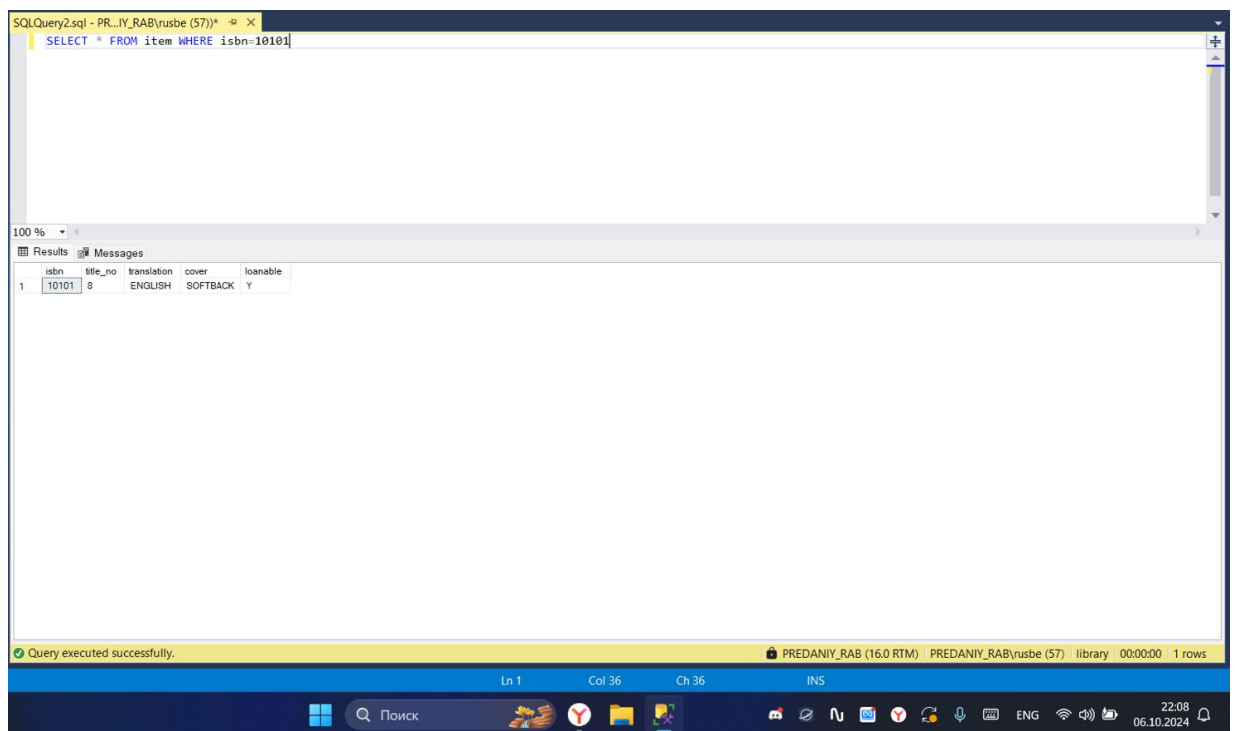


Рисунок 10 – Результат выполнения запроса



Запрос 2: Удаление указанной записи.

```
DELETE FROM item WHERE isbn=10101
```

Результат выполнения запроса представлен на Рисунке 11.

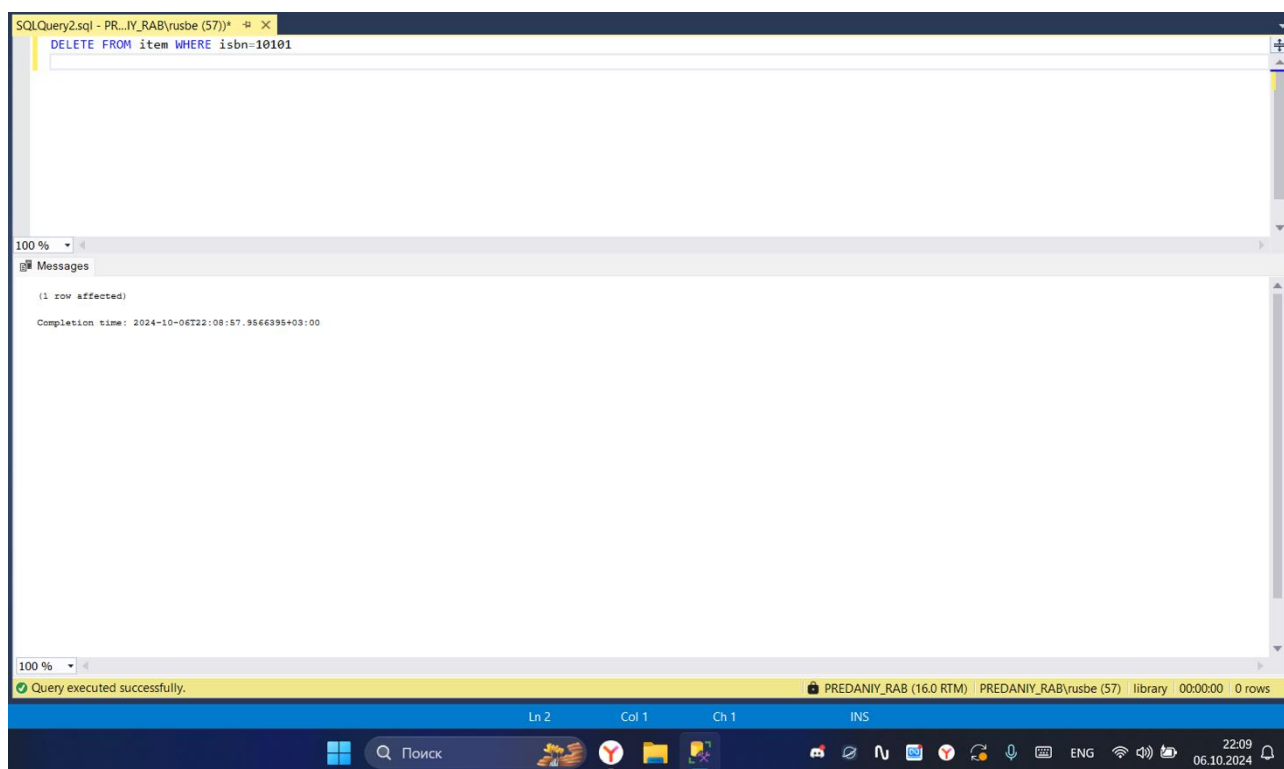


Рисунок 11 – Результат выполнения запроса

## Упражнение 5 – использование команды UPDATE

Запрос 1: Возвращение определенной записи, которую следует обновить.

```
SELECT * FROM member  
WHERE member_no=507
```

Результат выполнения запроса представлен на Рисунке 12.

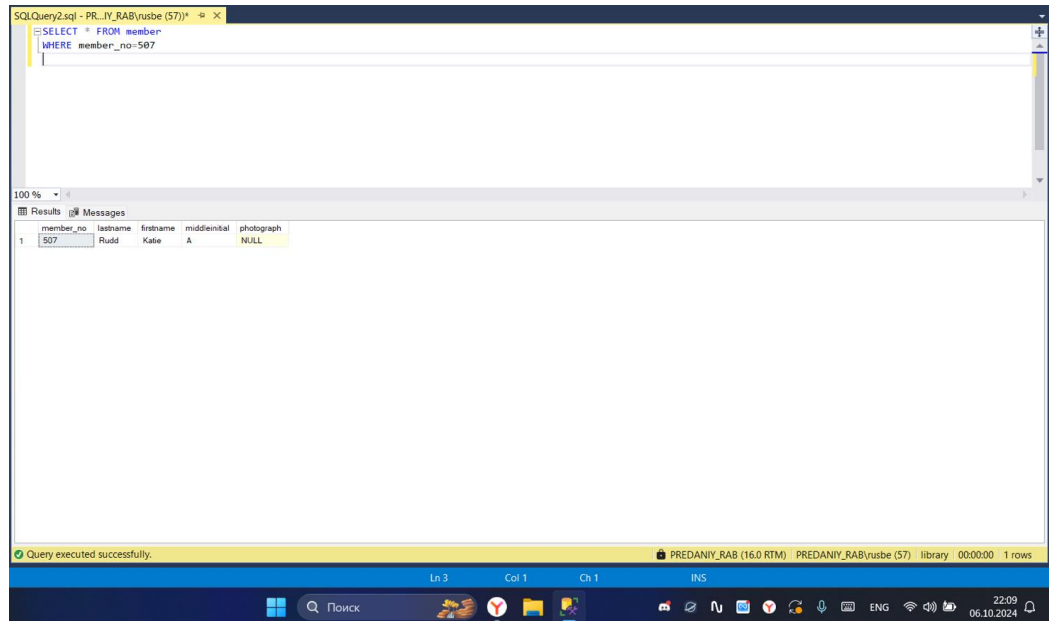


Рисунок 12 – Результат выполнения запроса

Запрос 2: Обновление записи.

```
UPDATE member
SET lastname = 'Smith'
WHERE member_no = 507
SELECT * FROM member
WHERE member_no = 507
```

Результат выполнения запроса представлен на Рисунке 13.

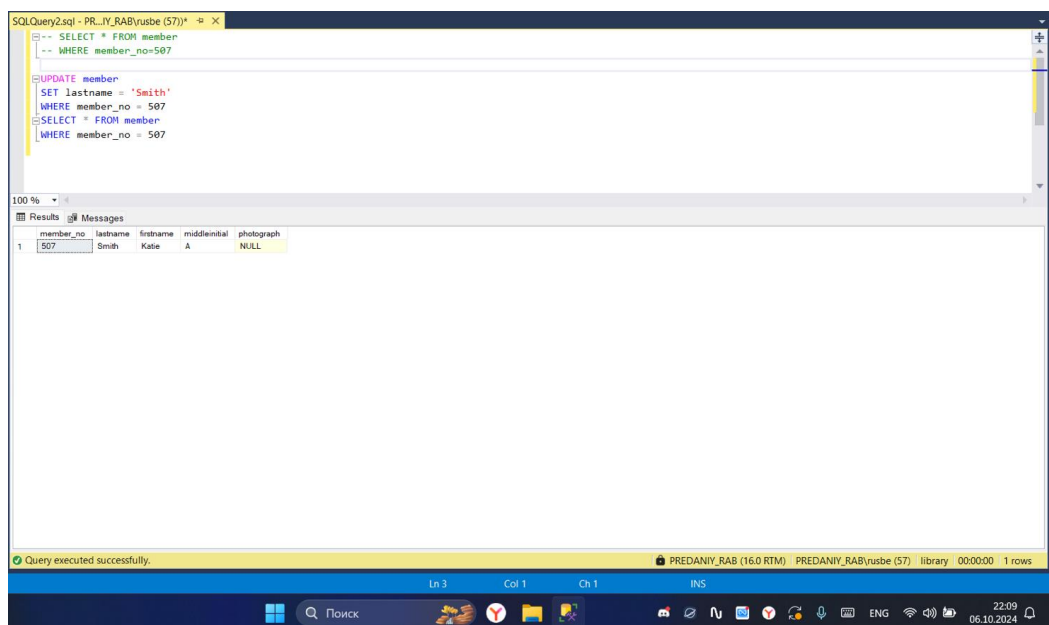


Рисунок 13 – Результат выполнения запроса

## Упражнение 6 – изменение данных на основе информации других таблиц

Запрос 1: Добавление нового юного читателя в базу данных.

Результат выполнения запроса представлен на Рисунке 14.

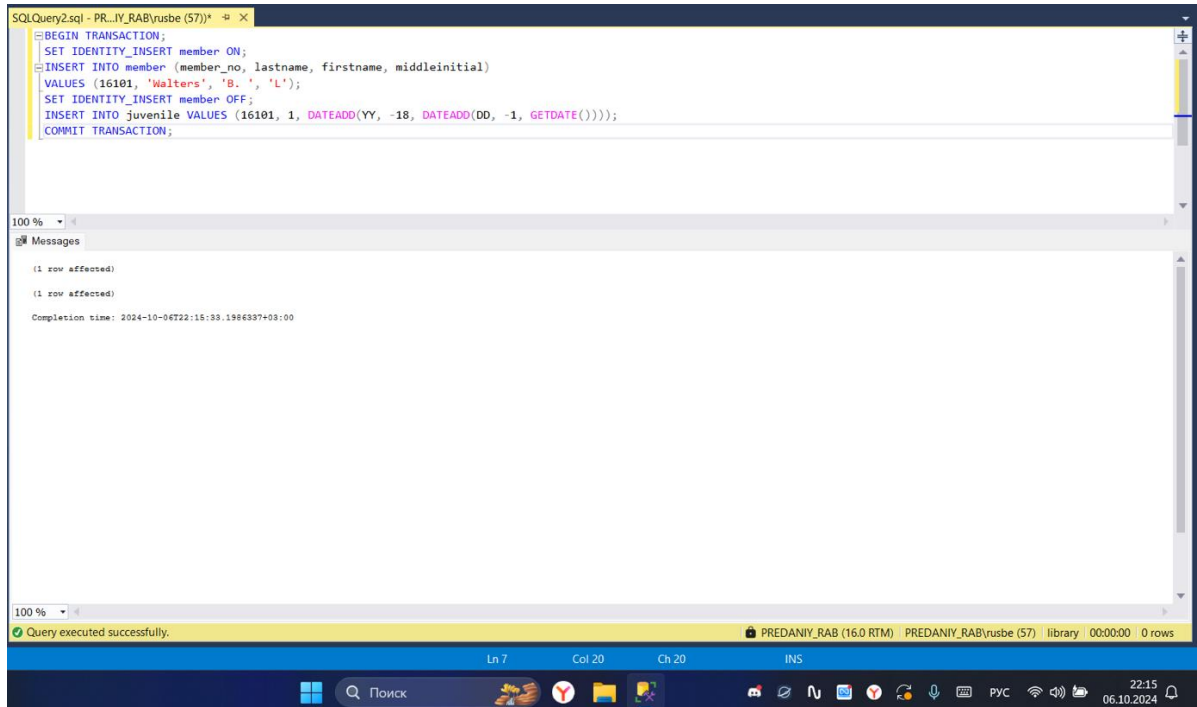


Рисунок 14 – Результат выполнения запроса

Запрос 2: Получение записей, которые должны быть перенесены из таблицы juvenile в таблицу adult.

```
SELECT juvenile.member_no, adult.street, adult.city, adult.state,
adult.zip, adult.phone_no, DATEADD (YY, 1, GETDATE())
FROM juvenile
JOIN adult ON juvenile.adult_member_no = adult.member_no
WHERE DATEADD (YY, 1, juvenile.birth_date) < GETDATE()
```

Результат выполнения запроса представлен на Рисунке 15.

SQLQuery2.sql - PR...IV\_RAB(rusbe (57))

```

SELECT juvenile.member_no, adult.street, adult.city, adult.state, adult.zip, adult.phone_no, DATEADD(YY, 1, GETDATE())
FROM juvenile
JOIN adult ON juvenile.adult_member_no = adult.member_no
WHERE DATEADD(YY, 1, juvenile.birth_date) < GETDATE();

```

Results Messages

	member_no	street	city	state	zip	phone_no	(No column name)
1	2	Bowery Estates	Montgomery	AL	36100	NULL	2025-10-06 22:16:30.480
2	7126	Bowery Estates	Montgomery	AL	36100	NULL	2025-10-06 22:16:30.480
3	8524	Bowery Estates	Montgomery	AL	36100	NULL	2025-10-06 22:16:30.480
4	9506	Bowery Estates	Montgomery	AL	36100	NULL	2025-10-06 22:16:30.480
5	16101	Bowery Estates	Montgomery	AL	36100	NULL	2025-10-06 22:16:30.480
6	9508	Dogwood Drive	Sacramento	CA	94203	NULL	2025-10-06 22:16:30.480
7	8526	Dogwood Drive	Sacramento	CA	94203	NULL	2025-10-06 22:16:30.480
8	7128	Dogwood Drive	Sacramento	CA	94203	NULL	2025-10-06 22:16:30.480
9	4	Dogwood Drive	Sacramento	CA	94203	NULL	2025-10-06 22:16:30.480
10	6	Fir Street	Washington	DC	20510-0001	NULL	2025-10-06 22:16:30.480
11	7130	Fir Street	Washington	DC	20510-0001	NULL	2025-10-06 22:16:30.480
12	8528	Fir Street	Washington	DC	20510-0001	NULL	2025-10-06 22:16:30.480
13	9510	Fir Street	Washington	DC	20510-0001	NULL	2025-10-06 22:16:30.480
14	9512	The Highlands	Atlanta	GA	30026	NULL	2025-10-06 22:16:30.480
15	8530	The Highlands	Atlanta	GA	30026	NULL	2025-10-06 22:16:30.480
16	7132	The Highlands	Atlanta	GA	30026	NULL	2025-10-06 22:16:30.480
17	8	The Highlands	Atlanta	GA	30026	NULL	2025-10-06 22:16:30.480
18	10	James Road	Springfield	IL	62700	NULL	2025-10-06 22:16:30.480
19	7134	James Road	Springfield	IL	62700	NULL	2025-10-06 22:16:30.480
20	8532	James Road	Springfield	IL	62700	NULL	2025-10-06 22:16:30.480
21	9514	James Road	Springfield	IL	62700	NULL	2025-10-06 22:16:30.480
22	9516	Larch Mont Road	Boston	MA	02138-7190	NULL	2025-10-06 22:16:30.480
23	8534	Larch Mont Road	Boston	MA	02138-7190	NULL	2025-10-06 22:16:30.480
24	7126	Larch Mont Road	Boston	MA	02138-7190	NULL	2025-10-06 22:16:30.480

Query executed successfully.

Ln 5 Col 1 Ch 1 INS

Поиск 22:16 06.10.2024

Рисунок 15 – Результат выполнения запроса

Запрос 3: Вставка записей в таблицу juvenile из таблицы adult.

```

INSERT INTO adult (member_no, street, city, state, zip, phone_no,
expr_date)

```

```

SELECT j.member_no, a.street, a.city, a.state, a.zip, a.phone_no,
GETDATE()

```

```

FROM juvenile AS j

```

```

JOIN adult AS a ON j.adult_member_no = a.member_no

```

```

WHERE DATEADD(YY, 18, j.birth_date) < GETDATE()

```

```

SELECT * FROM adult WHERE member_no = 16101

```

Результат выполнения запроса представлен на Рисунке 16.

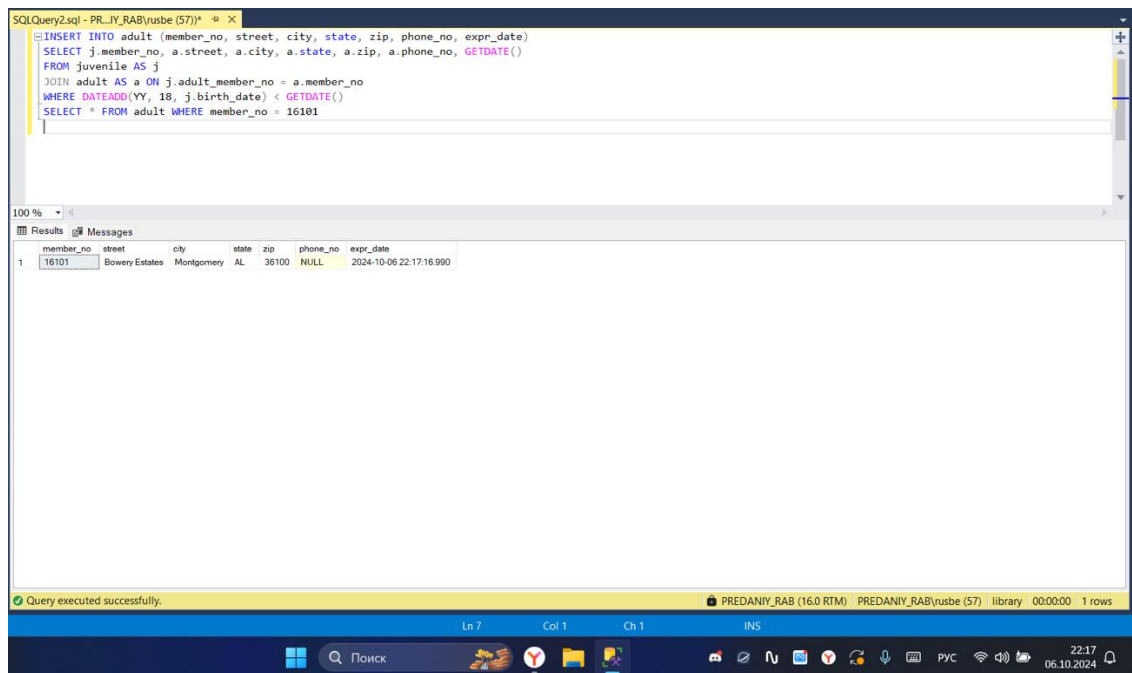


Рисунок 16 – Результат выполнения запроса

Запрос 4: Определение подлежащих удалению записей из таблицы juvenile.

```

SELECT juvenile.memeber_no
FROM juvenile
JOIN adult AS a ON juvenile.member_no = adult.member_no

```

Результат выполнения запроса представлен на Рисунке 17.

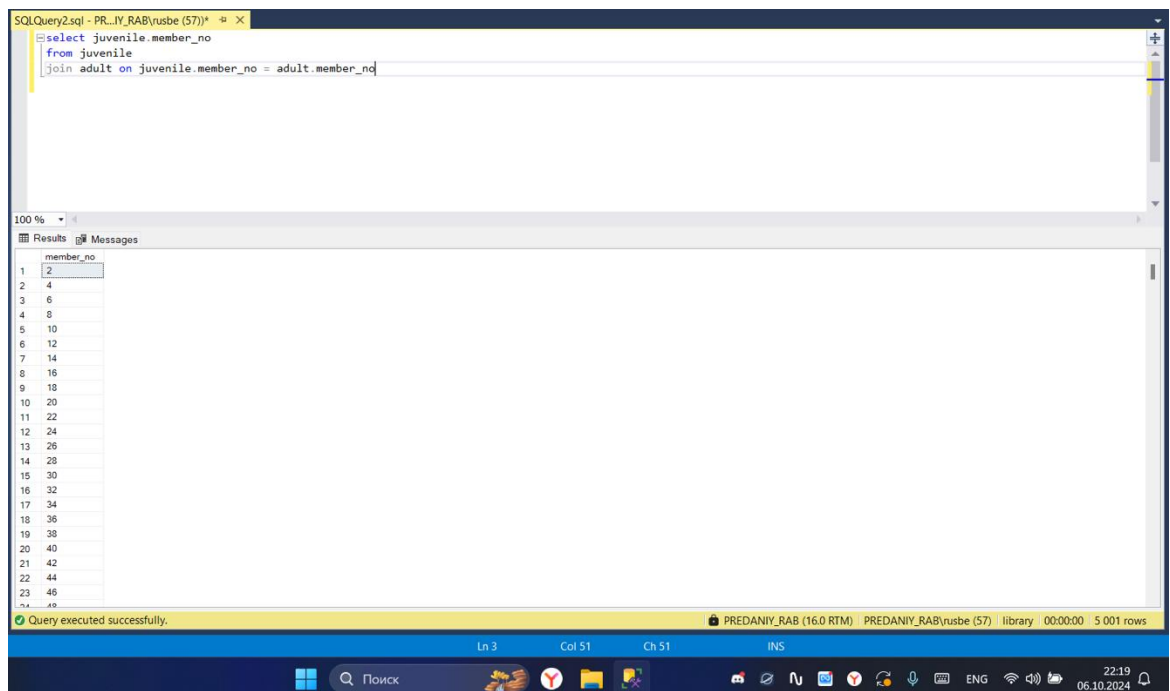


Рисунок 17 – Результат выполнения запроса

Запрос 5: Удаление записей из таблицы juvenile.

```
DELETE FROM juvenile
WHERE member_no IN (
    SELECT j.member_no
    FROM juvenile AS j
    JOIN adult AS a ON j.member_no = a.member_no
)
SELECT * FROM juvenile WHERE member_no = 16101
```

Результат выполнения запроса представлен на Рисунке 18.

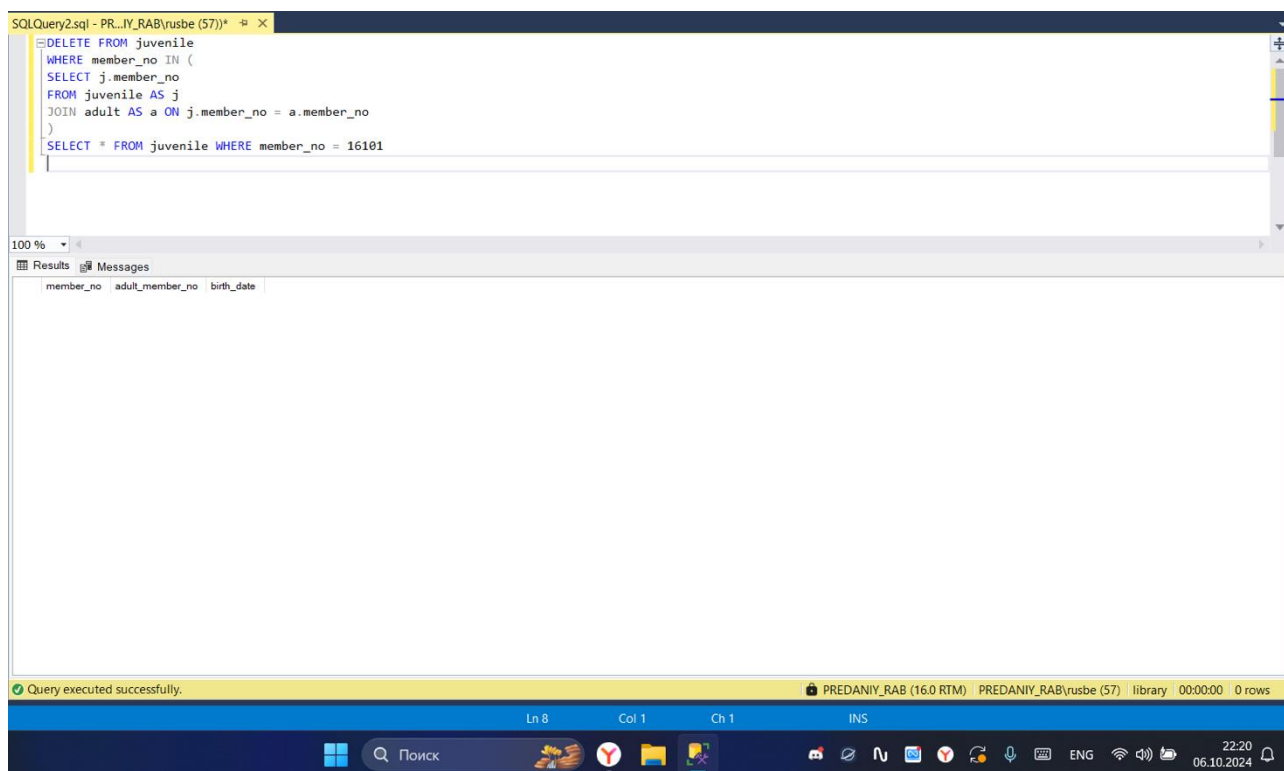


Рисунок 18 – Результат выполнения запроса

## Выводы

В процессе лабораторной работы были рассмотрены и применены команды для работы с данными: вставки, удаления и обновления записей в базе данных. На практике была использована команда INSERT для добавления строк в таблицы базы данных Library, включая варианты с ключевым словом DEFAULT для полей, допускающих null или имеющих значение по умолчанию. Также применялась команда INSERT с использованием DEFAULT VALUES,

позволяющая не указывать имена полей. Команда DELETE использовалась для удаления определенных записей, а UPDATE – для их изменения. Кроме того, было изучено использование команды SET IDENTITY INSERT, которая позволяет задавать пользовательские значения для полей с атрибутом IDENTITY. Все эти операции способствуют эффективному управлению данными в базе, обеспечивая корректную вставку, обновление, удаление и обработку информации.