Министерство науки и высшего образования Российской Федерации
федеральное государственное автономное образовательное учреждение
высшего образования
«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»

Отчет

по лабораторной работе 2

«Запросы на выборку и модификацию данных,

представления и индексы в PostgreSQL»

по дисциплине «**Проектирование и реализация баз данных**»

Вариант 3

Автор: Мосин З. И.

Факультет: Инфокоммуникационных технологий (ИКТ)

Группа: К32402

Преподаватель: Говорова М. М.

Дата: 10.08.2023

Санкт-Петербург 2022

Цель работы: овладеть практическими навыками создания представлений и запросов на выборку данных к базе данных PostgreSQL, использования подзапросов при модификации данных и индексов.

Программное обеспечение: СУБД PostgreSQL 1X, pgAdmin 4

Практическое задание:

Вариант 3. БД «Библиотека»

1. Вывести список читателей, имеющих на руках книги, переведенные с английского языка, изданные позднее 2000 года.
2. Вывести список читателей, не вернувших в срок книги и имеющих на руках более десяти книг.
3. Найти количество читателей, не вернувших в срок книги и имеющих на руках более десяти книг.
4. Вывести список книг, которые находятся в библиотеке в единственном экземпляре.
5. Подсчитать количество читателей, которые не обращались в библиотеку в течение года.
6. Подсчитать количество читателей библиотеки по уровню образования.
7. Вывести список книг по программированию на C#, экземпляры которых отсутствуют в библиотеке, и которые должны быть возвращены не позднее, чем через 3 дня.

Выполнение:

Задание 1. Создайте запросы.

Запрос №1 - Вывести список читателей, имеющих на руках книги, переведенные с английского языка, изданные позднее 2000 года.

SELECT full_name

FROM "library"."reader"

LEFT JOIN "library"."reader_card" t

ON ("reader".reader_id = t.reader_id)

LEFT JOIN "library"."subscription" a

ON (t.card_id = a.card_id)

LEFT JOIN "library"."book_copy"

ON ("book_copy".copy_id = a.copy_id)

LEFT JOIN "library"."edition"

ON ("edition".edition_id = "book_copy".edition_id)

LEFT JOIN "library"."book_info"

ON ("book_info".book_id = "edition".book_id)

WHERE "edition".edition_date >'2000-12-31'::date

AND "book_info".original_language = 'английский'

AND "edition".edition_language != "book_info".original_language

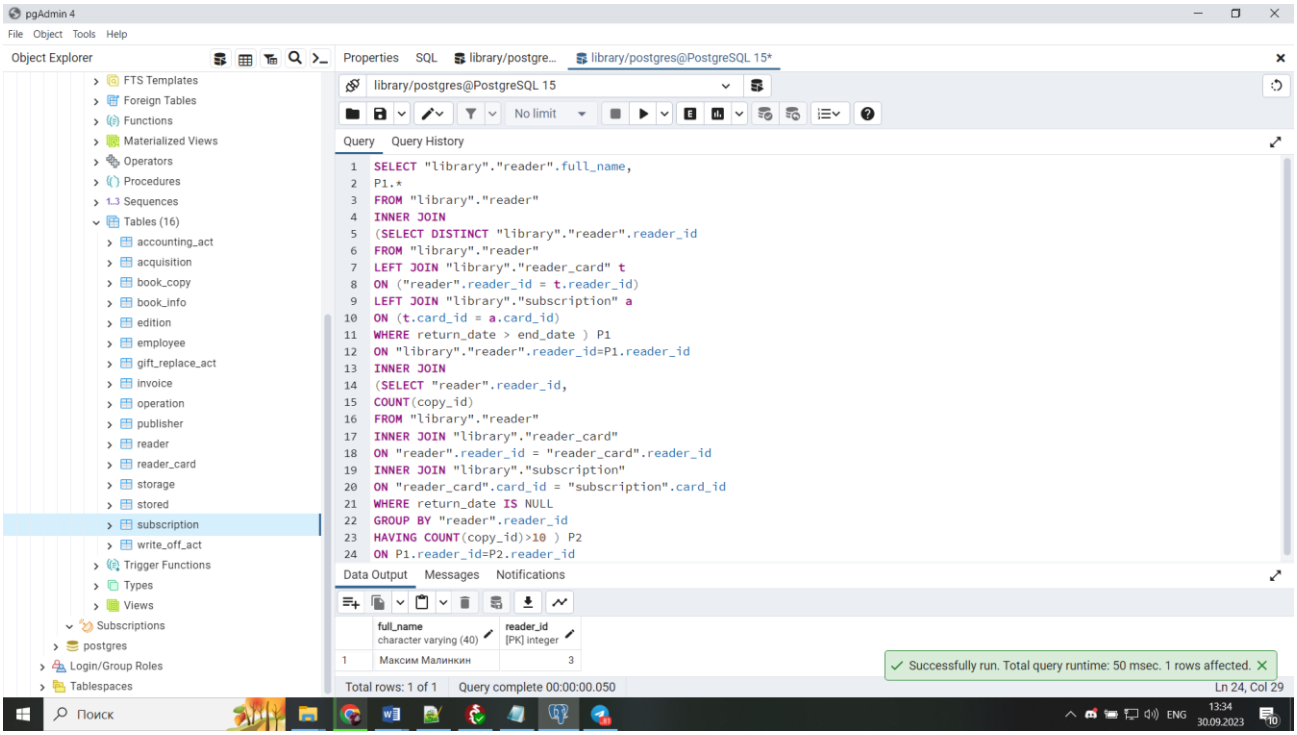
```
1  SELECT full_name
2  FROM "library"."reader"
3  LEFT JOIN "library"."reader_card" t
4  ON ("reader".reader_id = t.reader_id)
5  LEFT JOIN "library"."subscription" a
6  ON (t.card_id = a.card_id)
7  LEFT JOIN "library"."book_copy"
8  ON ("book_copy".copy_id = a.copy_id)
9  LEFT JOIN "library"."edition"
10 ON ("edition".edition_id = "book_copy".edition_id)
11 LEFT JOIN "library"."book_info"
12 ON ("book_info".book_id = "edition".book_id)
13 WHERE "edition".edition_date >'2000-12-31'::date
14 AND "book_info".original_language = 'английский'
15 AND "edition".edition_language != "book_info".original_language
```

Data Output | Messages | Notifications

	full_name character varying (40)
1	Максим Малинкин

Запрос №2 - Вывести список читателей, не вернувших в срок книги и имеющих на руках более десяти книг

```
SELECT "library"."reader".full_name,
P1.*
FROM "library"."reader"
INNER JOIN
(SELECT DISTINCT "library"."reader".reader_id
FROM "library"."reader"
LEFT JOIN "library"."reader_card" t
ON ("reader".reader_id = t.reader_id)
LEFT JOIN "library"."subscription" a
ON (t.card_id = a.card_id)
WHERE return_date > end_date ) P1
ON "library"."reader".reader_id=P1.reader_id
INNER JOIN
(SELECT "reader".reader_id,
COUNT(book_copy_id)
FROM "library"."reader"
INNER JOIN "library"."reader_card"
ON "reader".reader_id = "reader_card".reader_id
INNER JOIN "library"."subscription"
ON "reader_card".card_id = "subscription".card_id
WHERE return_date IS NULL
GROUP BY "reader".reader_id
HAVING COUNT(book_copy_id)>10 ) P2
ON P1.reader_id=P2.reader_id
```



Запрос №3 - *Найти количество читателей, не вернувших в срок книги и имеющих на руках более десяти книг.*

```
SELECT COUNT(DISTINCT "library"."reader".reader_id) AS count_debtors
FROM "library"."reader"
INNER JOIN(
    SELECT DISTINCT "library"."reader".reader_id
    FROM "library"."reader"
    LEFT JOIN "library"."reader_card" t
    ON ("library"."reader".reader_id = t.reader_id)
    LEFT JOIN "library"."subscription" a
    ON (t.card_id = a.card_id)
    WHERE return_date > end_date
) P1
ON "library"."reader".reader_id = P1.reader_id
INNER JOIN
(
    SELECT "reader".reader_id
    FROM "library"."reader"
    INNER JOIN "library"."reader_card"
```

ON "reader".reader_id = "reader_card".reader_id

INNER JOIN "library"."subscription"

ON "reader_card".card_id = "subscription".card_id

WHERE return_date IS NULL

GROUP BY "reader".reader_id

HAVING COUNT(copy_id) > 10) P2

ON P1.reader_id = P2.reader_id;

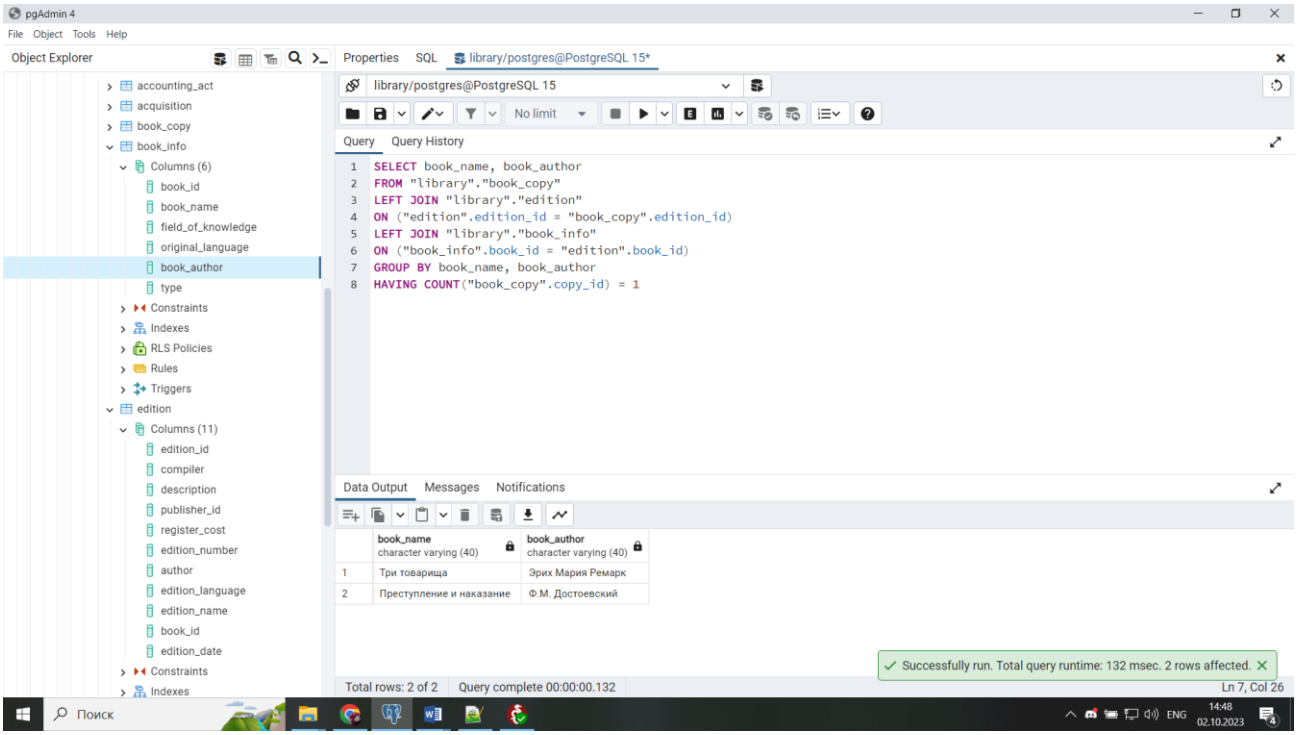
```sql
SELECT COUNT(DISTINCT "library"."reader".reader_id) AS count_debtors
FROM "library"."reader"
INNER JOIN
(
    SELECT DISTINCT "library"."reader".reader_id
    FROM "library"."reader"
    LEFT JOIN "library"."reader_card" t
    ON ("library"."reader".reader_id = t.reader_id)
    LEFT JOIN "library"."subscription" a
    ON (t.card_id = a.card_id)
    WHERE return_date > end_date
) P1
ON "library"."reader".reader_id = P1.reader_id
INNER JOIN
(
    SELECT "reader".reader_id
    FROM "library"."reader"
    INNER JOIN "library"."reader_card"
    ON "reader".reader_id = "reader_card".reader_id
    INNER JOIN "library"."subscription"
```

Data Output | Messages | Notifications

	count_debtors bigint
1	1

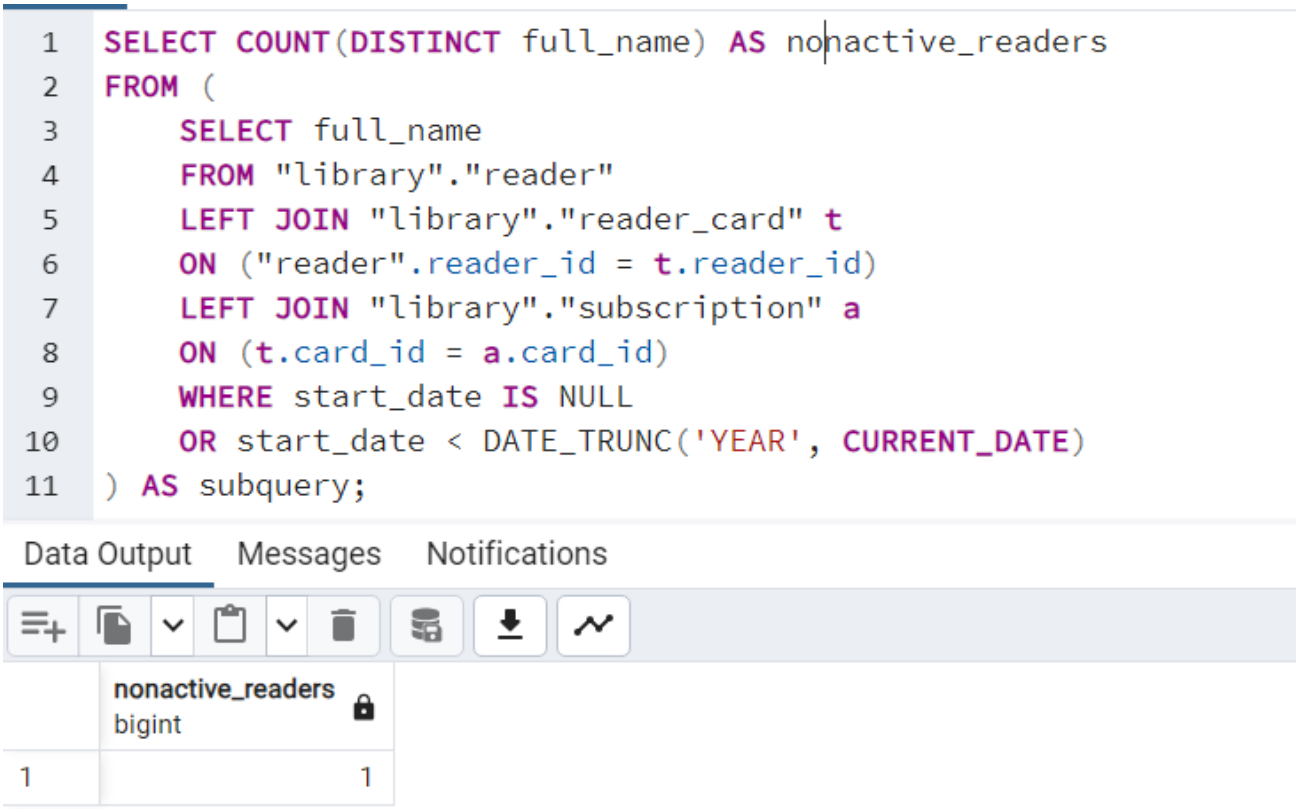
Запрос №4 - *Вывести список книг, которые находятся в библиотеке в единственном экземпляре.*

```
SELECT book_name, book_author
FROM "library"."book_book_copy"
LEFT JOIN "library"."edition"
ON ("edition".edition_id = "book_book_copy".edition_id)
LEFT JOIN "library"."book_info"
ON ("book_info".book_id = "edition".book_id)
GROUP BY book_name, book_author
HAVING COUNT("book_book_copy".book_copy_id) = 1
```



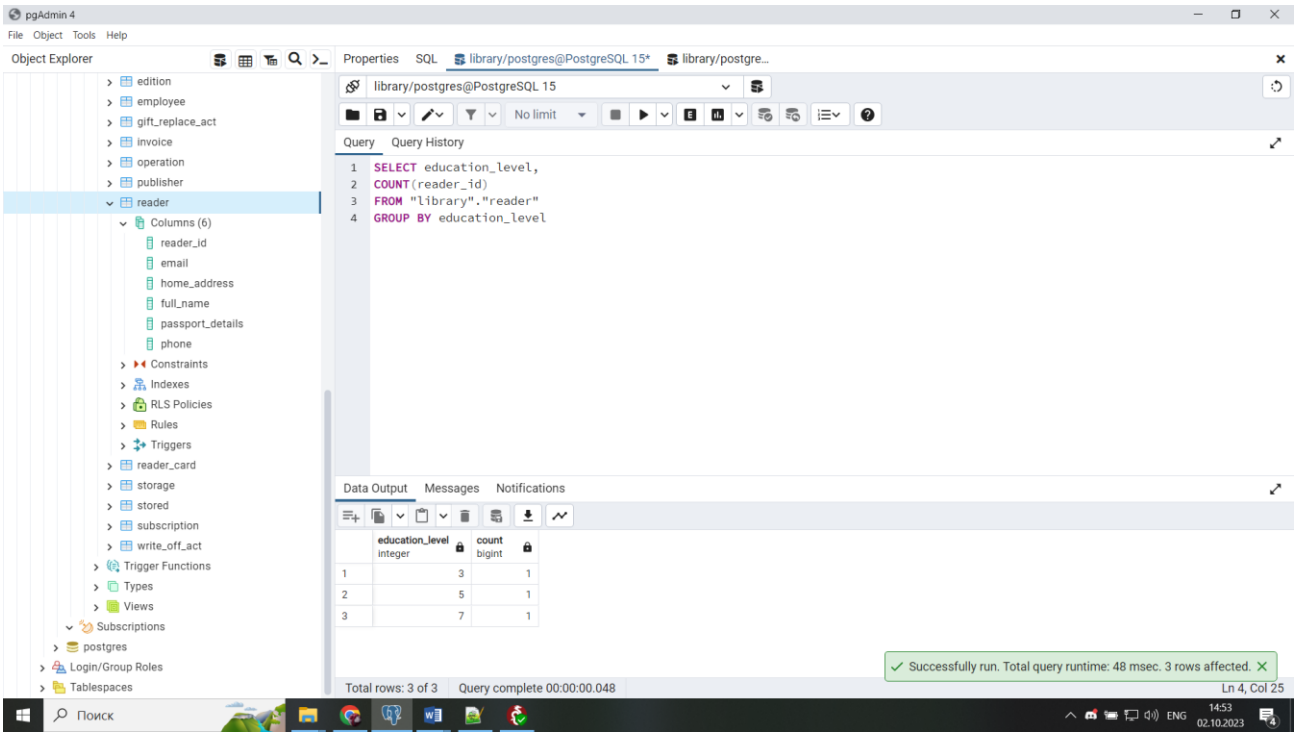
Запрос №5 - *Подсчитать количество читателей, которые не обращались в библиотеку в течение года.*

```
SELECT COUNT(DISTINCT full_name) AS nonactive_readers
FROM (
    SELECT full_name
    FROM "library"."reader"
    LEFT JOIN "library"."reader_card" t
    ON ("reader".reader_id = t.reader_id)
    LEFT JOIN "library"."subscription" a
    ON (t.card_id = a.card_id)
    WHERE start_date IS NULL
    OR start_date < DATE_TRUNC('YEAR', CURRENT_DATE)
) AS subquery;
```



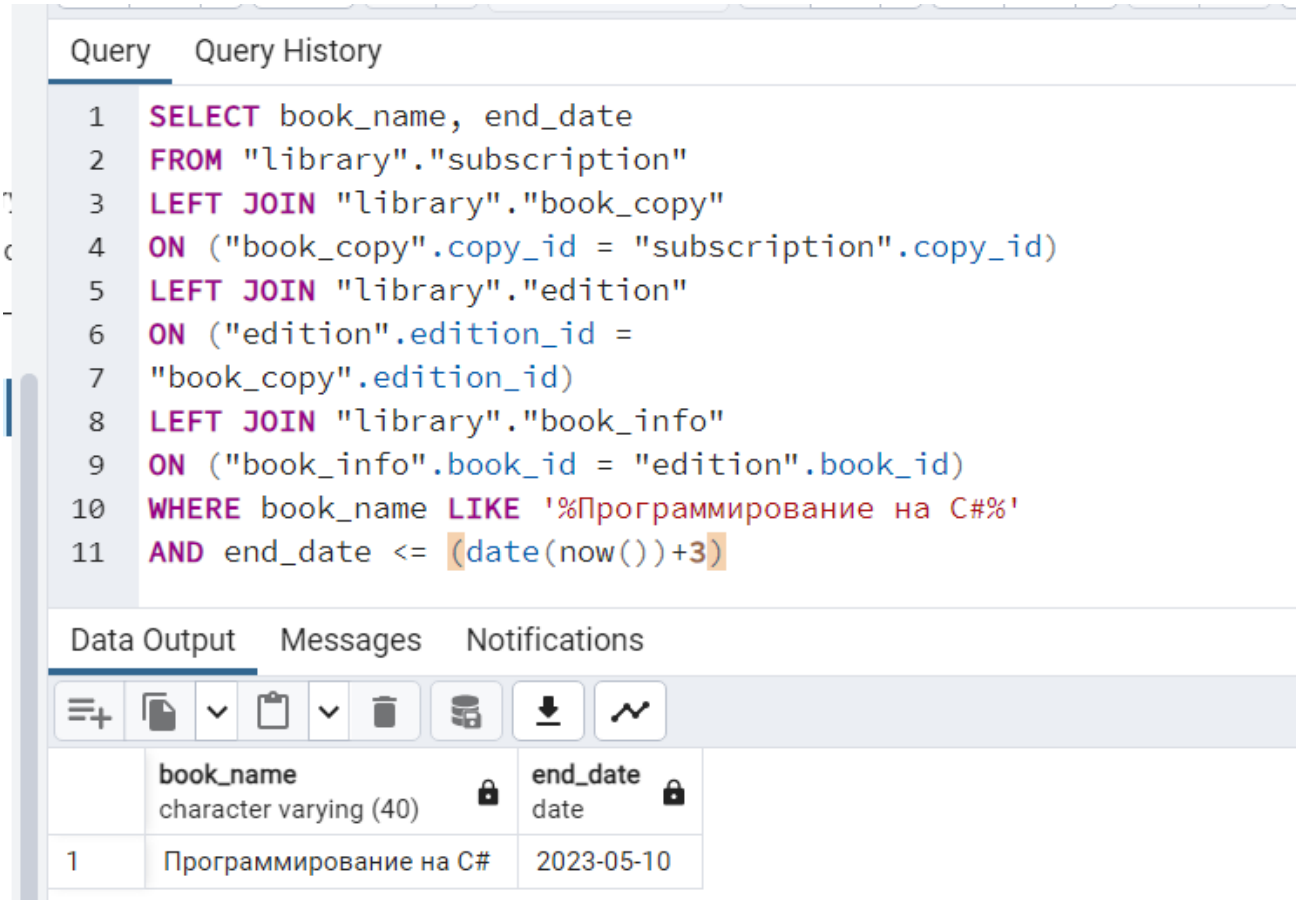
Запрос №6 - Подсчитать количество читателей библиотеки по уровню образования.

```
SELECT education_level,
COUNT(reader_id)
FROM "library"."reader"
GROUP BY education_level
```



Запрос №7 - *Вывести список книг по программированию на C#, экземпляры которых отсутствуют в библиотеке, и которые должны быть возвращены не позднее, чем через 3 дня.*

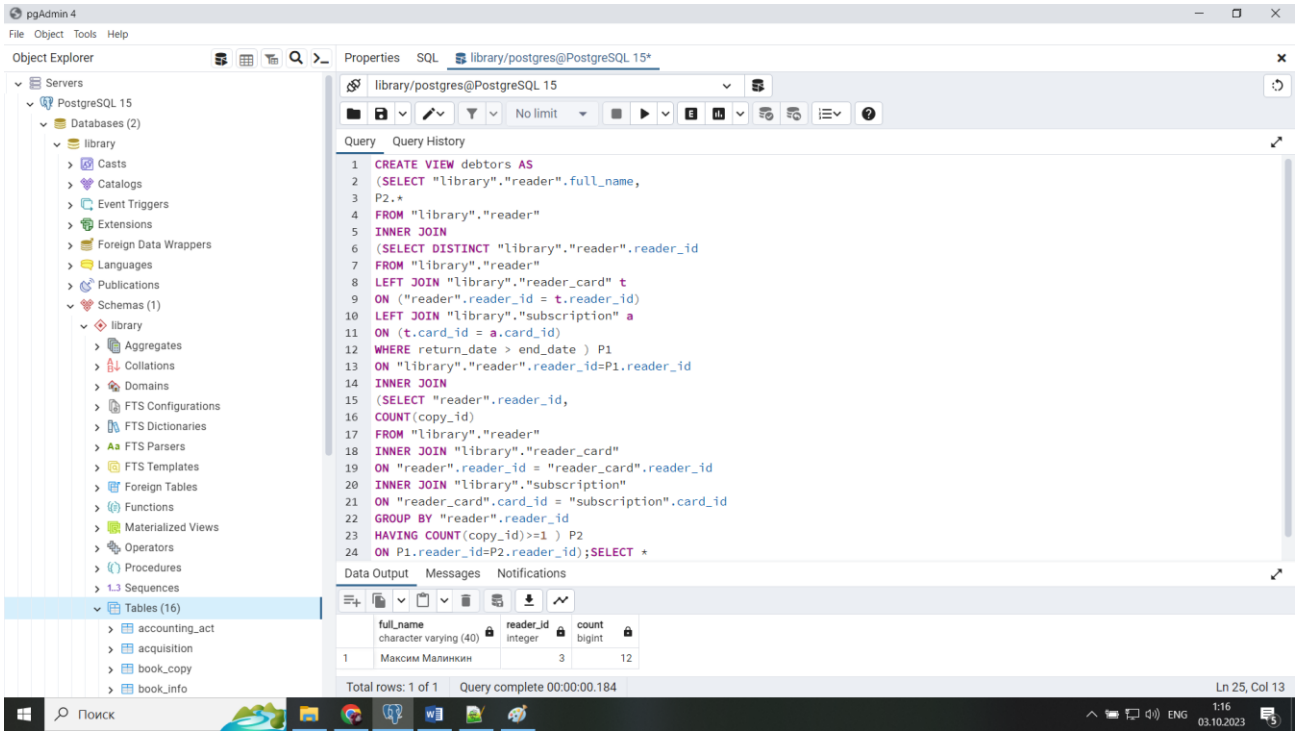
```
SELECT book_name, end_date
FROM "library"."subscription"
LEFT JOIN "library"."book_copy"
ON ("book_copy".copy_id = "subscription".copy_id)
LEFT JOIN "library"."edition"
ON ("edition".edition_id =
"book_copy".edition_id)
LEFT JOIN "library"."book_info"
ON ("book_info".book_id = "edition".book_id)
WHERE book_name LIKE '%Программирование на C#%'
AND end_date <= (date(now())+3)
```



Задание 2. Создать представления для администрации библиотеки.

Представление №1 - сведения о должниках.

```
CREATE VIEW debtors AS
(SELECT "library"."reader".full_name,
P2.*
FROM "library"."reader"
INNER JOIN
(SELECT DISTINCT "library"."reader".reader_id
FROM "library"."reader"
LEFT JOIN "library"."reader_card" t
ON ("reader".reader_id = t.reader_id)
LEFT JOIN "library"."subscription" a
ON (t.card_id = a.card_id)
WHERE return_date > end_date ) P1
ON "library"."reader".reader_id=P1.reader_id
INNER JOIN
(SELECT "reader".reader_id,
COUNT(copy_id)
FROM "library"."reader"
INNER JOIN "library"."reader_card"
ON "reader".reader_id = "reader_card".reader_id
INNER JOIN "library"."subscription"
ON "reader_card".card_id = "subscription".card_id
GROUP BY "reader".reader_id
HAVING COUNT(copy_id)>=1 ) P2
ON P1.reader_id=P2.reader_id);SELECT *
FROM debtors
```


pgAdmin 4
File Object Tools Help
Object Explorer
Servers
PostgreSQL 15
Databases (2)
library
Casts
Catalogs
Event Triggers
Extensions
Foreign Data Wrappers
Languages
Publications
Schemas (1)
library
Aggregates
Collations
Domains
FTS Configurations
FTS Dictionaries
FTS Parsers
FTS Templates
Foreign Tables
Functions
Materialized Views
Operators
Procedures
Sequences
Tables (16)
accounting_act
acquisition
book_copy
book_info
Properties SQL library/postgres@PostgreSQL 15*
library/postgres@PostgreSQL 15
No limit
Query Query History
1 CREATE VIEW debtors AS
2 (SELECT "library"."reader".full_name,
3 P2.*
4 FROM "library"."reader"
5 INNER JOIN
6 (SELECT DISTINCT "library"."reader".reader_id
7 FROM "library"."reader"
8 LEFT JOIN "library"."reader_card" t
9 ON ("reader".reader_id = t.reader_id)
10 LEFT JOIN "library"."subscription" a
11 ON (t.card_id = a.card_id)
12 WHERE return_date > end_date) P1
13 ON "library"."reader".reader_id=P1.reader_id
14 INNER JOIN
15 (SELECT "reader".reader_id,
16 COUNT(copy_id)
17 FROM "library"."reader"
18 INNER JOIN "library"."reader_card"
19 ON "reader".reader_id = "reader_card".reader_id
20 INNER JOIN "library"."subscription"
21 ON "reader_card".card_id = "subscription".card_id
22 GROUP BY "reader".reader_id
23 HAVING COUNT(copy_id)>=1) P2
24 ON P1.reader_id=P2.reader_id);SELECT *
Data Output Messages Notifications
full_name character varying (40)
reader_id integer
count bigint
1 Максим Малинкин 3 12
Total rows: 1 of 1
Query complete 00:00:00.184
Ln 25, Col 13
Поиск
ENG
1:16
03.10.2023

Представление №2 - сведения о наиболее популярных книгах (все экземпляры находятся на руках у читателей).

```
CREATE VIEW popular_books AS
SELECT
   bi.book_id,
   bi.book_name
FROM
   "library"."book_info" bi
WHERE
   bi.book_id NOT IN (
      SELECT e.book_id FROM "library"."edition" e
      LEFT JOIN "library"."book_copy" c
      ON e.edition_id = c.edition_id
      LEFT JOIN "library"."subscription" s
      ON c.copy_id = s.copy_id
      WHERE s.return_date IS NULL
   );
```

```
1  CREATE VIEW popular_books AS
2  SELECT
3      bi.book_id,
4      bi.book_name
5  FROM
6      "library"."book_info" bi
7  WHERE
8      bi.book_id NOT IN (
9          SELECT e.book_id FROM "library"."edition" e
10         LEFT JOIN "library"."book_copy" c
11         ON e.edition_id = c.edition_id
12         LEFT JOIN "library"."subscription" s
13         ON c.copy_id = s.copy_id
14         WHERE s.return_date IS NULL
15     );
```

Data Output | Messages | Notifications

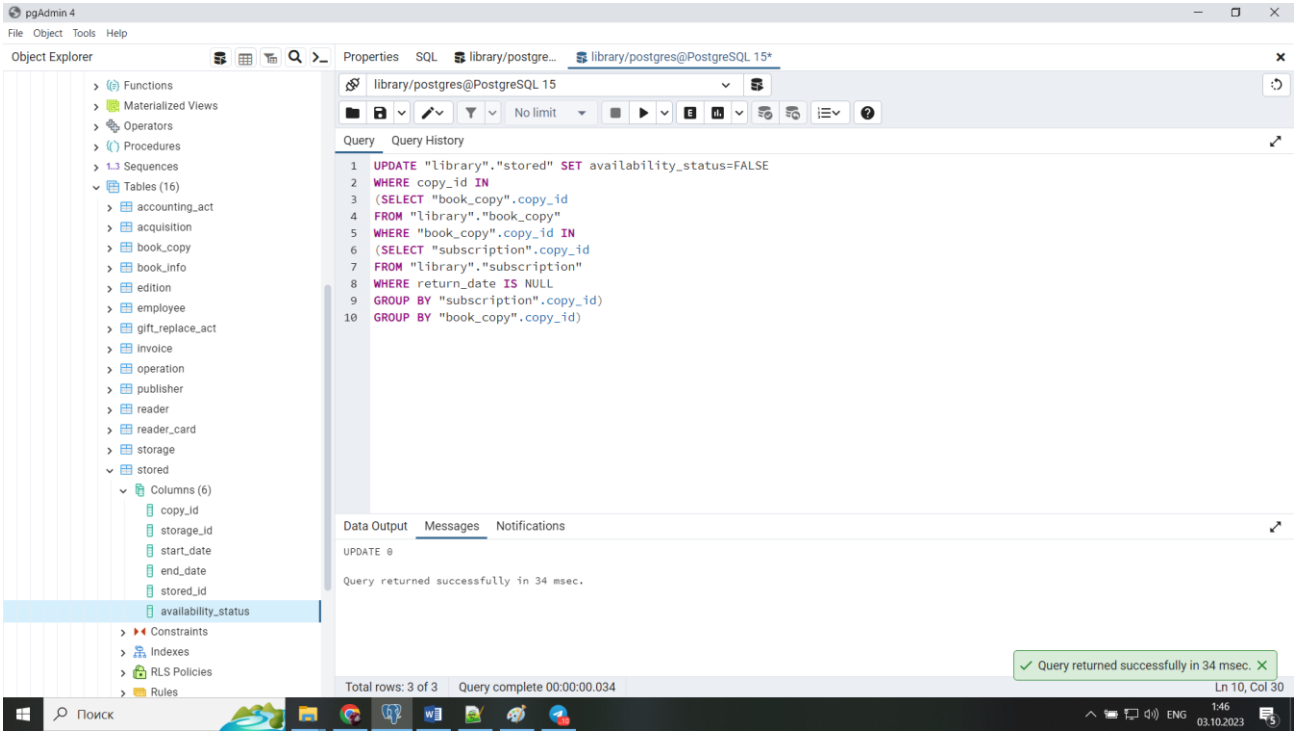
	book_id integer	book_name character varying (40)
1	1	Преступление и наказание
2	8	Над пропастью во ржи

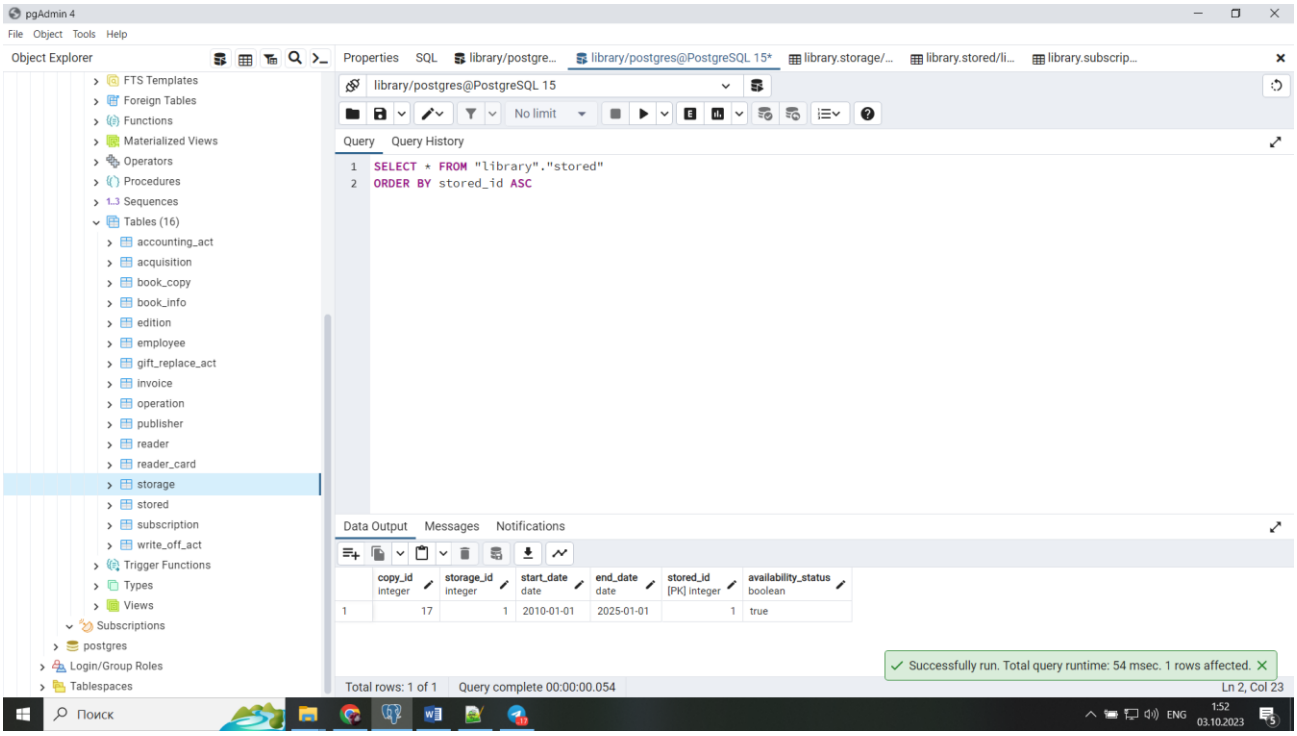
Задание 3. Запросы на модификацию данных

Запрос 1. Update с подзапросом

Запрос: изменить статус на занято у тех книг, которые находятся на руках.

```
UPDATE "library"."stored" SET availability_status=FALSE
WHERE copy_id IN
(SELECT "book_copy".copy_id
FROM "library"."book_copy"
WHERE "book_copy".copy_id IN
(SELECT "subscription".copy_id
FROM "library"."subscription"
WHERE return_date IS NULL
GROUP BY "subscription".copy_id)
GROUP BY "book_copy".copy_id)
```

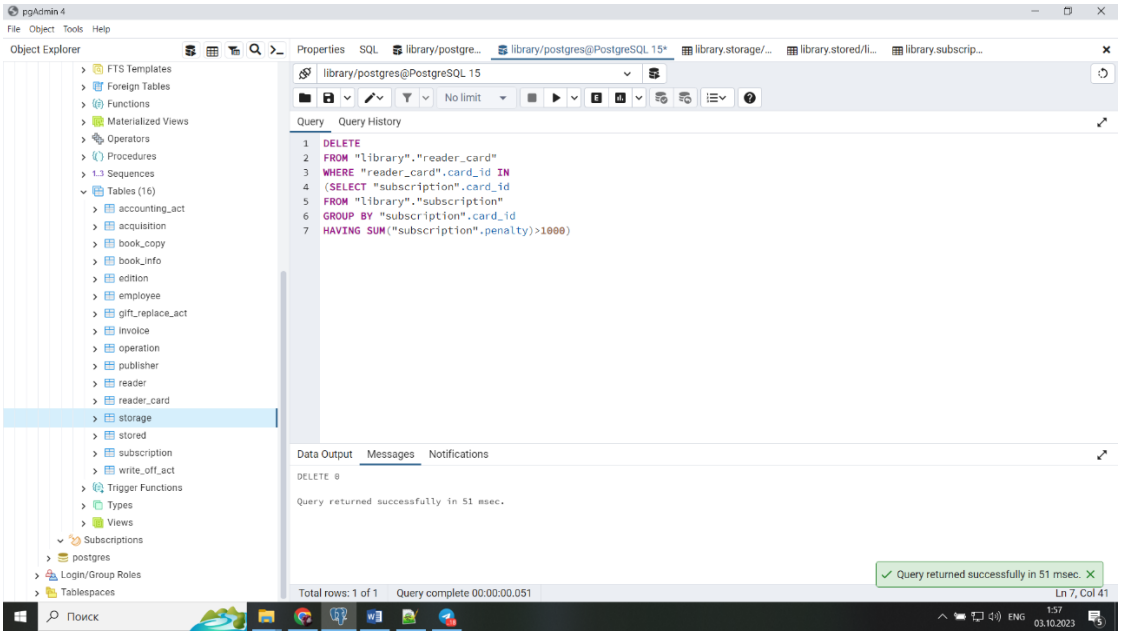


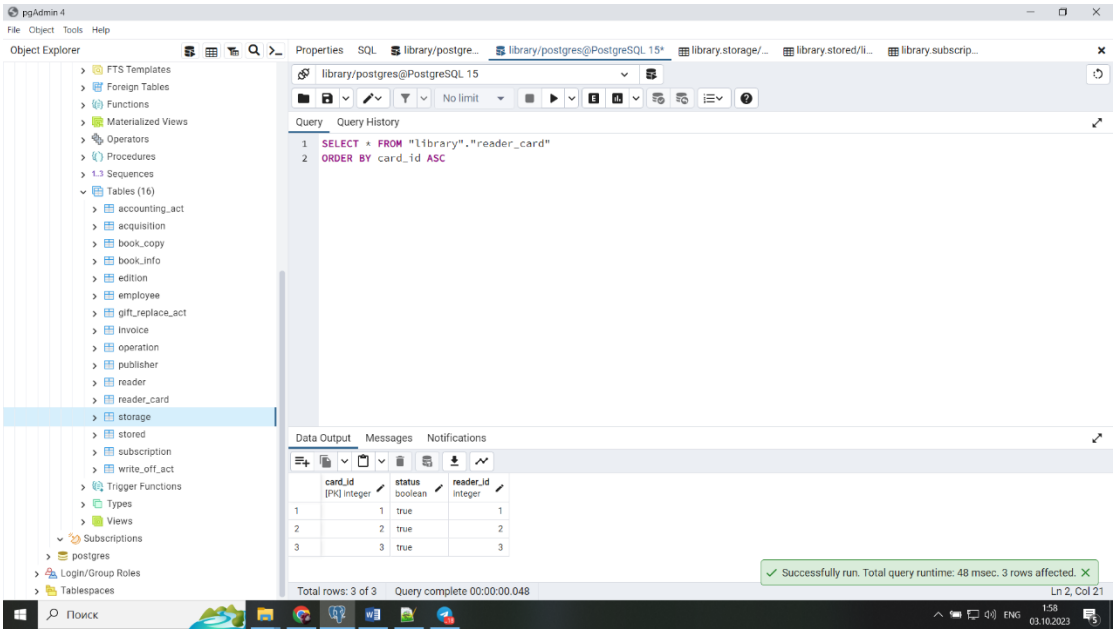

pgAdmin 4
File Object Tools Help
Object Explorer
Properties SQL library/postgre... library/postgres@PostgreSQL 15* library.storage/... library.stored/li... library.subscrip...
FTS Templates
Foreign Tables
Functions
Materialized Views
Operators
Procedures
Sequences
Tables (16)
accounting_act
acquisition
book_copy
book_info
edition
employee
gift_replace_act
invoice
operation
publisher
reader
reader_card
storage
stored
subscription
write_off_act
Trigger Functions
Types
Views
Subscriptions
postgres
Login/Group Roles
Tablespaces
library/postgres@PostgreSQL 15
No limit
Query Query History
1 SELECT * FROM "library"."stored"
2 ORDER BY stored_id ASC
Data Output Messages Notifications
copy_id integer
storage_id integer
start_date date
end_date date
stored_id [PK] integer
availability_status boolean
1 17 1 2010-01-01 2025-01-01 1 true
Successfully run. Total query runtime: 54 msec. 1 rows affected.
Total rows: 1 of 1
Query complete 00:00:00.054
Ln 2, Col 23
Поиск
ENG
1:52
03.10.2023

Запрос 2. Delete с подзапросом

Запрос: удалить карту читателя, который суммарно получил штрафов больше 1000р.

```
DELETE
FROM "library"."reader_card"
WHERE "reader_card".card_id IN
(SELECT "subscription".card_id
FROM "library"."subscription"
GROUP BY "subscription".card_id
HAVING SUM("subscription".penalty)>1000)
```

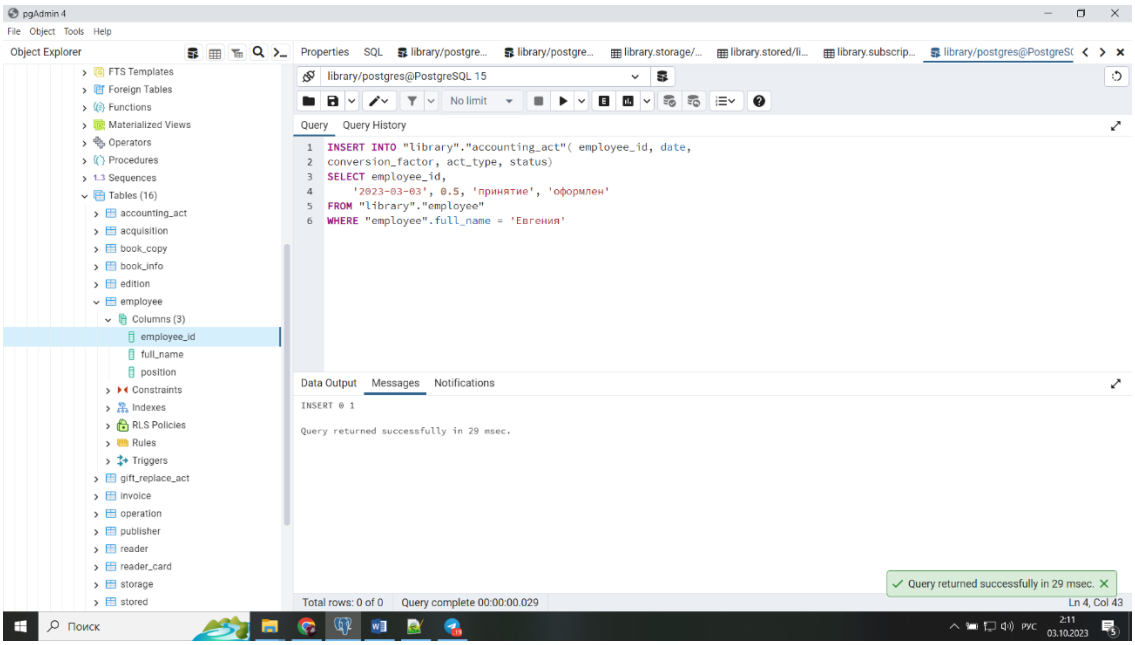


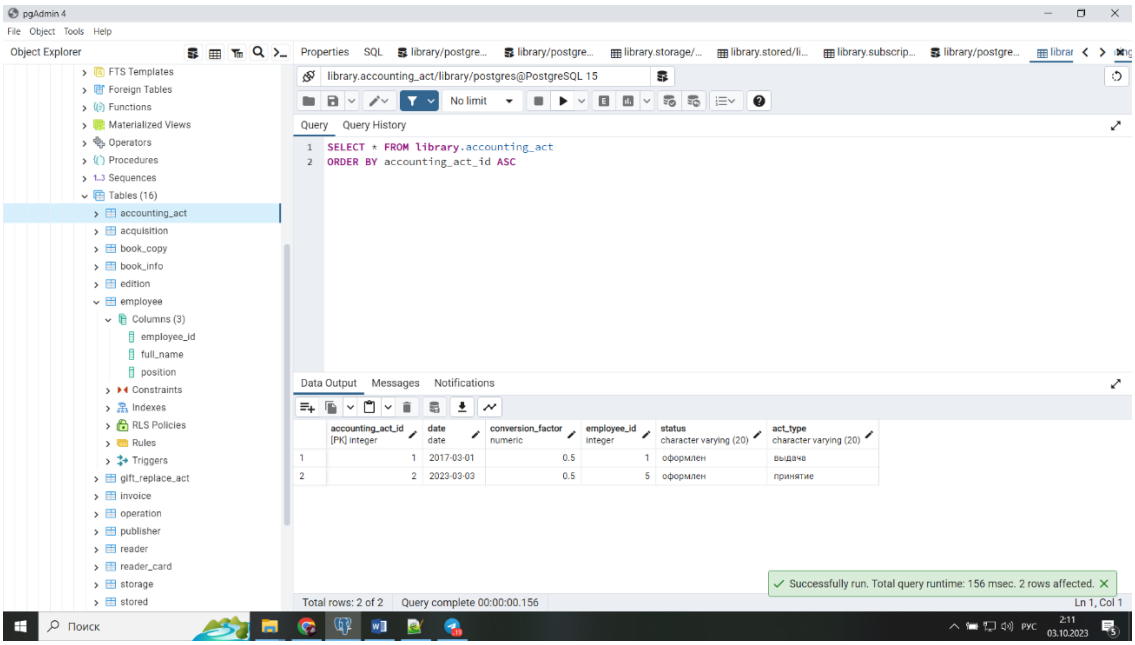


Запрос 3. Insert с подзапросом

Запрос: добавить новый акт принятия книги, оформленный библиотекарем Евгенией сегодня.

```
INSERT INTO "library"."accounting_act"( employee_id, date,
conversion_factor, type, status)
SELECT employee_id,
'2023-03-10', 0.002, 'принятие', 'оформлен'
FROM "library"."employee"
WHERE "employee".full_name = 'Евгения'
```

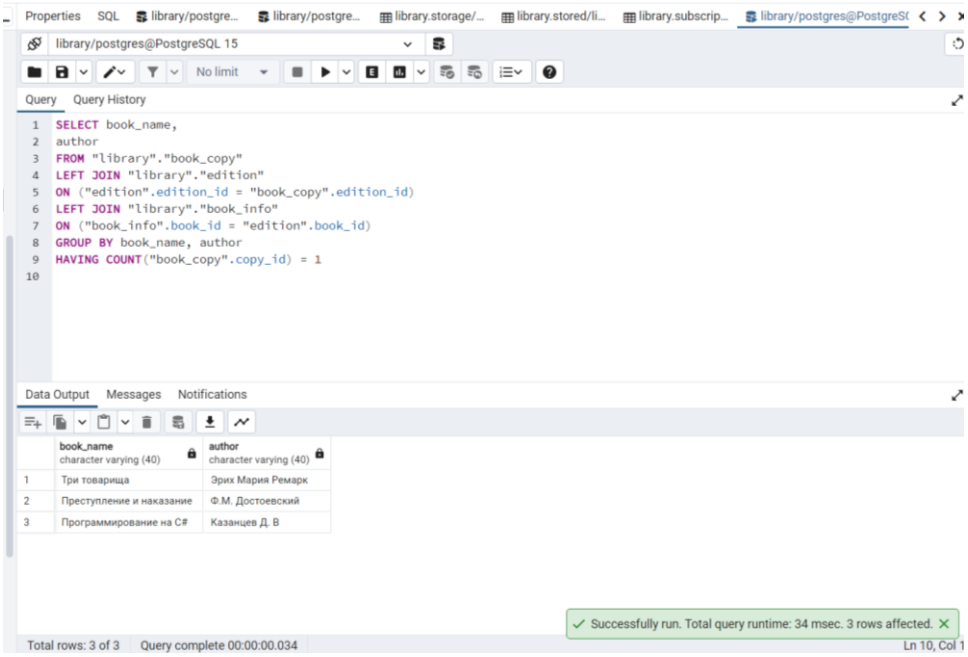




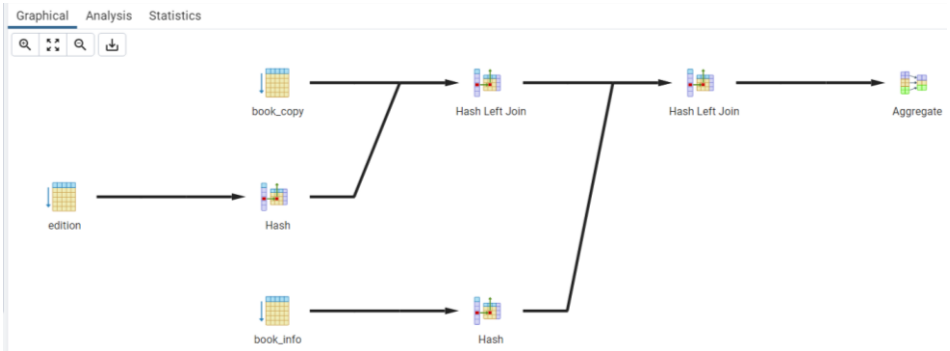
Запрос 4. Индексы

1)Запрос без индекса.

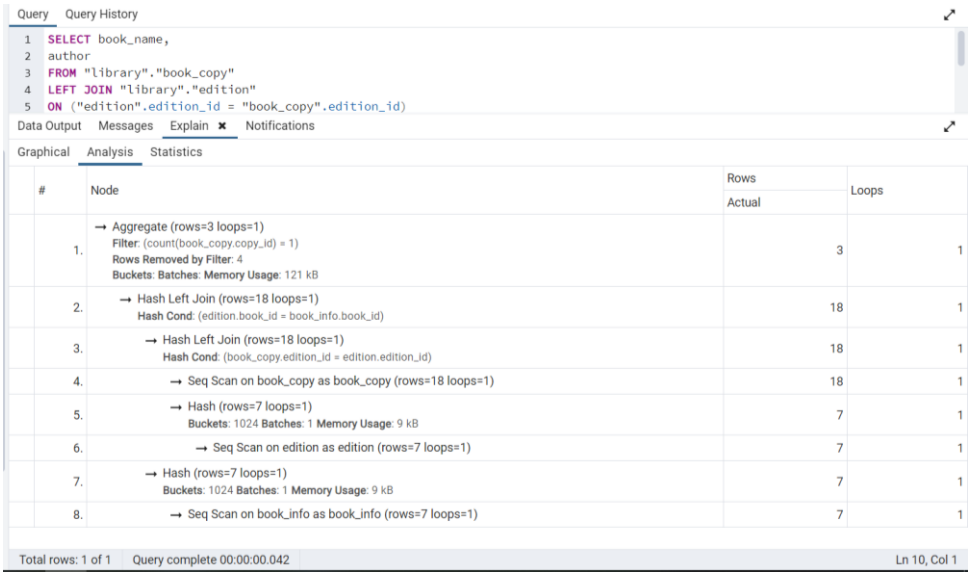
```
SELECT book_name,
author
FROM "library"."book_copy"
LEFT JOIN "library"."edition"
ON ("edition".edition_id = "book_copy".edition_id)
LEFT JOIN "library"."book_info"
ON ("book_info".book_id = "edition".book_id)
GROUP BY book_name, author
HAVING COUNT("book_copy".copy_id) = 1
```



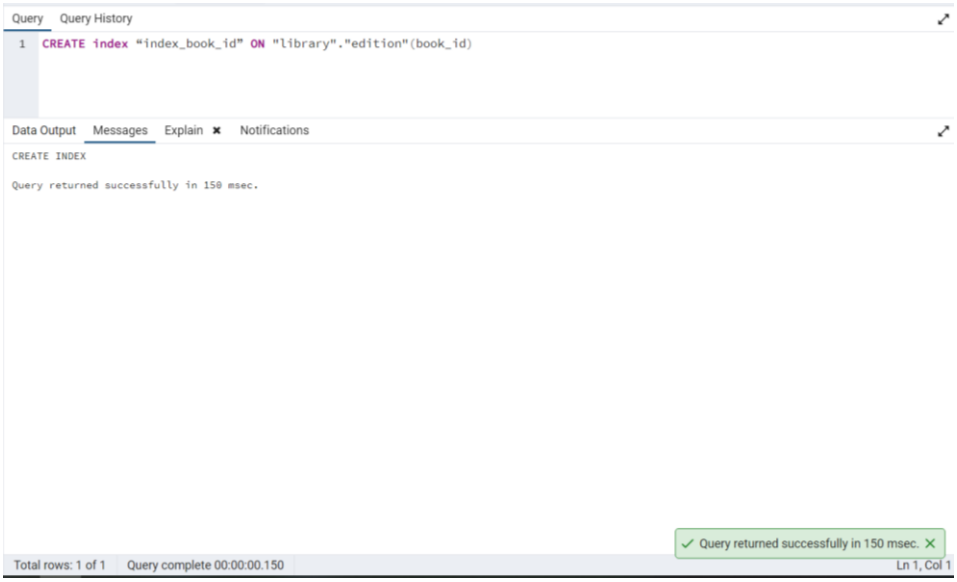
План запроса в графическом виде:



План запроса в аналитическом виде:

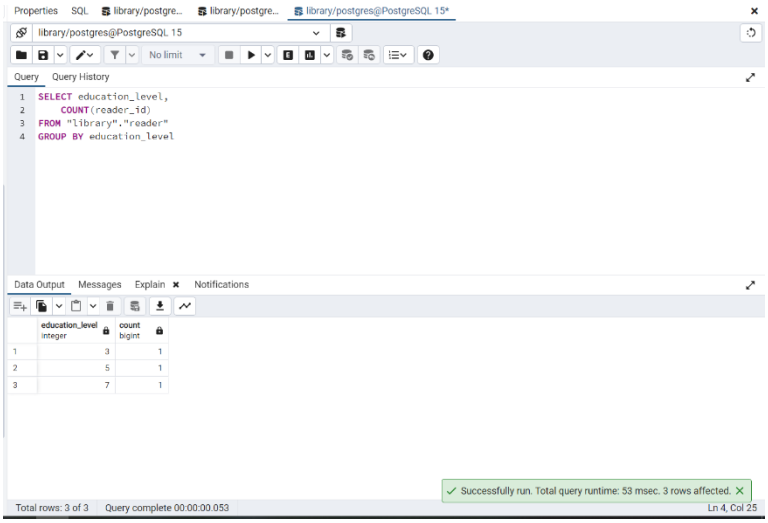


Создание индекса:

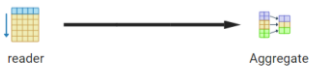


2)Запрос с индексом 2

```
SELECT education_level,
        COUNT(reader_id)
FROM "library"."reader"
GROUP BY education_level
```



План запроса:



Время выполнения:

Выводы:

В процессе выполнения лабораторной работы были созданы запросы для извлечения данных из базы данных PostgreSQL в соответствии с индивидуальными заданиями части 2 и 3. Также были разработаны три запроса для изменения данных (INSERT, UPDATE, DELETE), которые включали использование подзапросов. Мы также провели анализ графических представлений запросов с помощью EXPLAIN. Мы успешно выполнили все этапы, указанные в практическом задании, и приобрели опыт работы с представлениями, индексами и

выполнением запросов на изменение данных с использованием подзапросов. Особенно стоит отметить, что использование индексов в больших запросах значительно сокращает время их выполнения.