Министерство науки и высшего образования Российской Федерации

**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ**

**ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ «ЛЭТИ»**

**ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)**

197376, Санкт-Петербург, ул. проф. Попова, 5.

Факультет компьютерных технологий и информатики

Кафедра вычислительной техники

«ЗАЧТЕНО»

\_\_\_\_\_\_\_\_\_\_ О.А. Жирнова

“\_\_” \_\_\_\_\_\_\_\_\_\_\_\_ 2021 г.

ОТЧЁТ  
по дисциплине «Базы данных»

**Лабораторная работа № 5  
«Работа с вложенными запросами»**

|  |  |
| --- | --- |
| Студент группы 9308 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | Н.В. Яловега |

Санкт Петербург 2021

*Цель работы:* научиться писать и применять вложенные запросы.

*Используемая база данных (БД):* Library.

**Порядок выполнения**

*Упражнение 1 –* **использование вложенных запросов как производных таблиц.**

Запрос 1. Запрос, формирующий производную таблицу, возвращает столбец juvenile.adult\_member\_no и количество подростков для каждого взрослого читателя библиотеки, имеющего более трех детей записанных в библиотеку. Список раздела SELECT основного запроса включает поля adult\_member\_no и No\_Of\_Children из производного запроса и поле expr\_date из таблицы adult. Результат на рисунке 1

SELECT adult.member\_no, children.child\_num, expr\_date

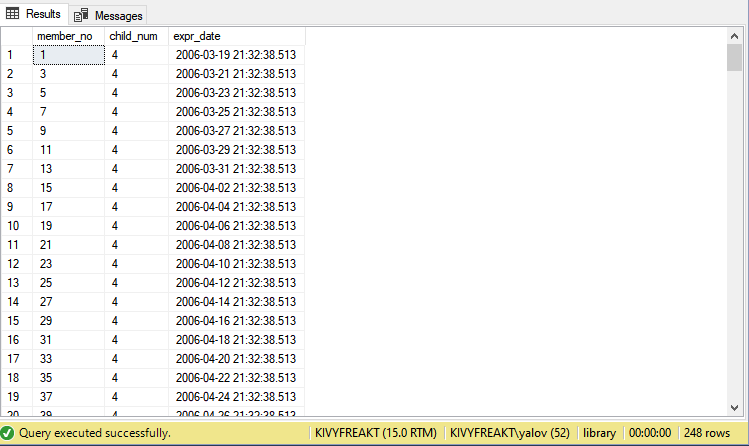
FROM (SELECT adult\_member\_no, COUNT(member\_no) as child\_num

FROM juvenile

GROUP BY adult\_member\_no

HAVING COUNT(member\_no) > 3) AS children

INNER JOIN adult on adult.member\_no = children.adult\_member\_no

Рисунок 1

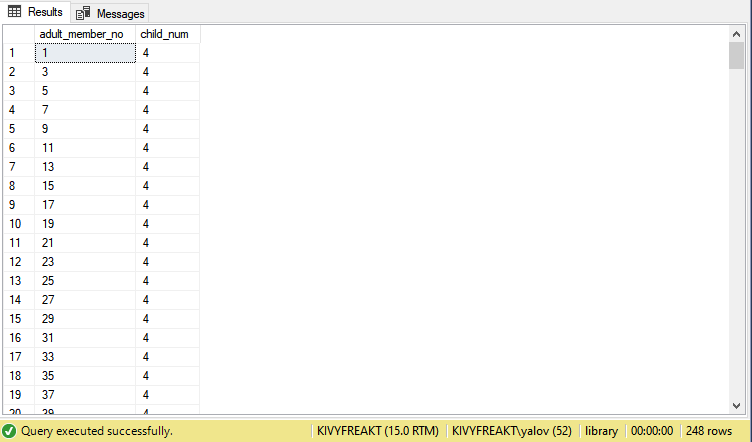
Запрос 2. Проверка выполнения запроса №1. Результат на рисунке 2.

SELECT adult\_member\_no, COUNT(member\_no) as child\_num

FROM juvenile

GROUP BY adult\_member\_no

HAVING COUNT(member\_no) > 3

Рисунок 2

Запрос 3. Создание запроса производной таблицей в виде двух разных запросов. Результат на рисунке 3.

WITH MemberInfo AS

(

SELECT juvenile.adult\_member\_no, COUNT(juvenile.member\_no) as

child\_num, adult.expr\_date

FROM juvenile INNER JOIN adult ON adult.member\_no =

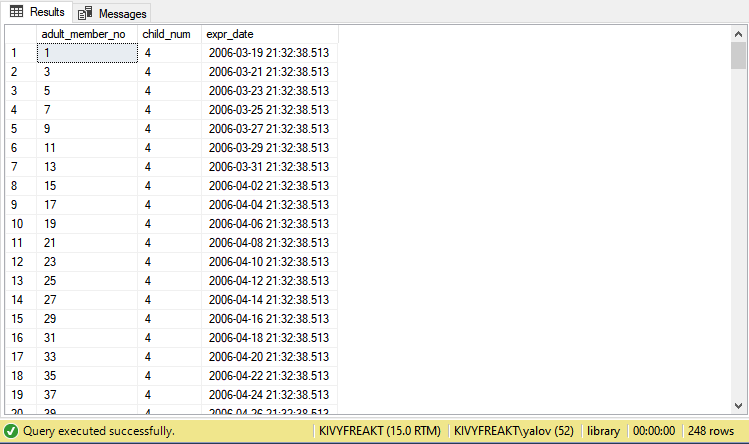
juvenile.adult\_member\_no

GROUP BY juvenile.adult\_member\_no, adult.expr\_date

HAVING COUNT(juvenile.member\_no) > 3

)

SELECT \* FROM MemberInfo

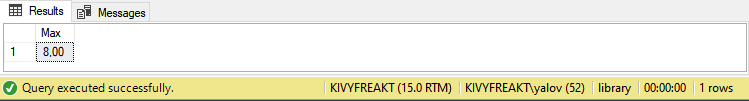
Рисунок 3

*Упражнение 2 –* **использование подзапросов как выражений**

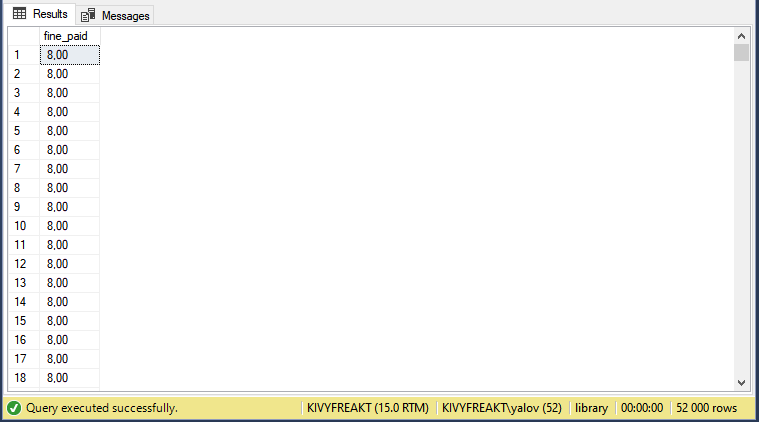
Запрос 1. Возвращающение значения полей member.firstname, member.lastname, loanhist.isbn и loanhist.fine\_paid для читателей библиотеки, заплативших максимальных штраф за все книги и который возвращает максимальное значение поля loanhist.fine\_paid. Результат на рисунке 4.

SELECT MAX(loanhist.fine\_paid) as Max FROM loanhist

Запрос 2. Проверка результата. Результат на рисунке 5.

Рисунок 4

SELECT fine\_paid FROM loanhist ORDER BY fine\_paid DESC

Рисунок 5

Запрос 3. Соединение таблицы member и loanhist и возвращение значения firstname, lastname, isbn, в которых fine\_paid имеет максимальное значение для всех строк, с ключевым словом DISTICT. Результат на рисунке 6.

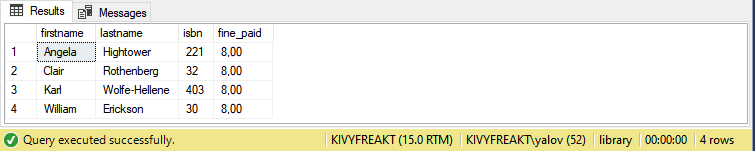
SELECT DISTINCT member.firstname, member.lastname, loanhist.isbn,

loanhist.fine\_paid

FROM loanhist INNER JOIN member on member.member\_no =

loanhist.member\_no

WHERE fine\_paid = (SELECT MAX(loanhist.fine\_paid) FROM loanhist)

Рисунок 6

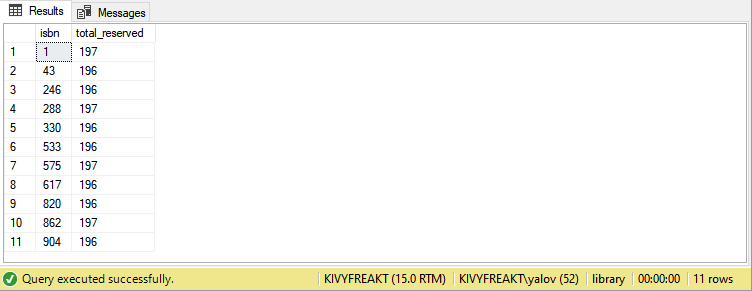
Запрос 4. Запрос, возвращающий номера isbn книг из таблицы reservation для книг, у которых более 50 копий или менее 5. Результат на рисунке 7

SELECT isbn, COUNT(\*) as total\_reserved

FROM reservation

GROUP BY isbn

HAVING COUNT(\*) > 50 or COUNT(\*) < 5

Рисунок 7

Запрос 5. запрос, который возвращает поля title\_no, title, isbn и Total Reserved, где количество копий книг должно быть менее 5 и должно использоваться ключевое слово IN. Результат на рисунке 8.

SELECT DISTINCT t.title\_no, t.title, c.isbn, reserved.total\_reserved

FROM copy c

INNER JOIN title t ON c.title\_no = t.title\_no

INNER JOIN (SELECT isbn, COUNT(\*) as total\_reserved

FROM reservation

GROUP BY isbn

HAVING COUNT(\*) > 50 or COUNT(\*) < 5) as reserved ON reserved.isbn =

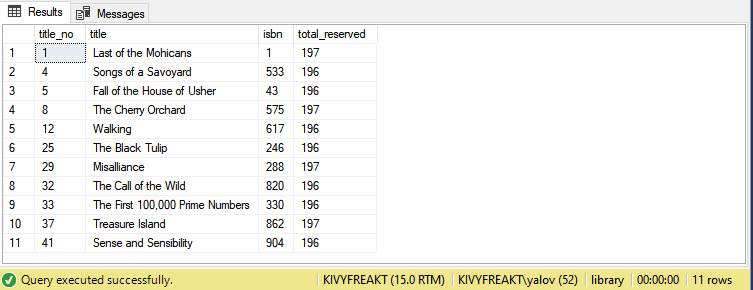
c.isbn

WHERE c.isbn IN (SELECT isbn

FROM reservation

GROUP BY isbn

HAVING COUNT(\*) > 50 or COUNT(\*) < 5)

Рисунок 8

*Упражнение 3 –* **использование коррелирующих подзапросов**

Запрос 1. Запрос, показывающий читателей со штрафом более 5. Результат на рисунке 9.

SELECT fines.member\_no, lastname, fines.total\_fine

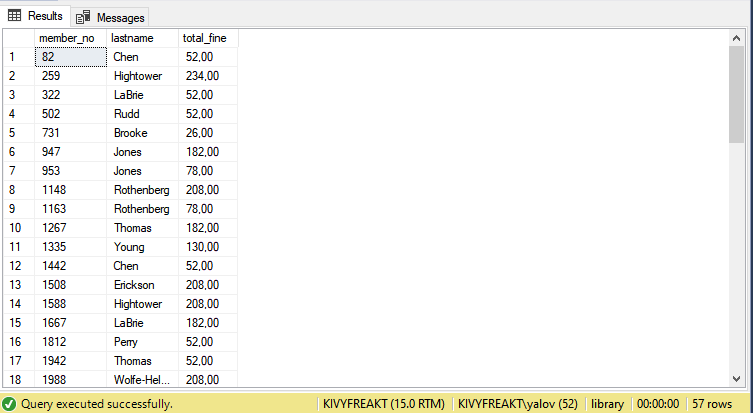
FROM member as m

INNER JOIN (SELECT member\_no, SUM(fine\_assessed) as total\_fine

FROM loanhist

GROUP BY member\_no) as fines ON fines.member\_no = m.member\_no

WHERE fines.total\_fine > 5

Рисунок 9

**Выводы**

Получены навыки в написании и применении вложенных запросов, использования подзапросов как выражений, подзапросов как части условия поиска, подзапросов с несколькими значениями, коррелированных подзапросов. Для создания производной таблицы использовалась команда WITH … AS.

**Список использованных источников**

1. Горячев А. В., Новакова Н. Е. Распределенные базы данных. Мет. указания к лаб. работам., СПб. Изд-во СПбГЭТУ «ЛЭТИ», 2008

2. Горячев А.В, Новакова Н.Е. Особенности разработки и администрирования приложений баз данных: учеб. пособие. СПб.: Изд-во СПбГЭТУ «ЛЭТИ», 2016. 68 с.

3. Дейт К. Введение в системы баз данных. : Пер. с англ. – 6-е изд. -К.:Диалектика, 1998.