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 | С. А.Дубенков |
|-----------------------|------------------|
| Orygoni ipyinibi oooo | О. 7 ч.ду обинов |

Санкт Петербург 2021

Цель работы: научиться писать и применять вложенные запросы [лит 1]. *Используемая база данных (БД):* Library.

Порядок выполнения

Упражнение 1 — использование вложенных запросов как производных таблиц

1. Выполнен запрос, использующий производные таблицы. Запрос, формирующий производную таблицу, возвращает столбец juvenile.adult_member_no и количество подростков для каждого взрослого читателя библиотеки, имеющего более трех детей (подростков) записанных в библиотеку (являющихся ее читателями). Список раздела SELECT основного запроса включает поля adult_member_no и No_Of_Children из производного запроса и поле expr_date из таблицы adult. Выполнен запрос, использующий производную таблицу:

```
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 Pesyльтат выполнения запроса показан на рисунке 1
```

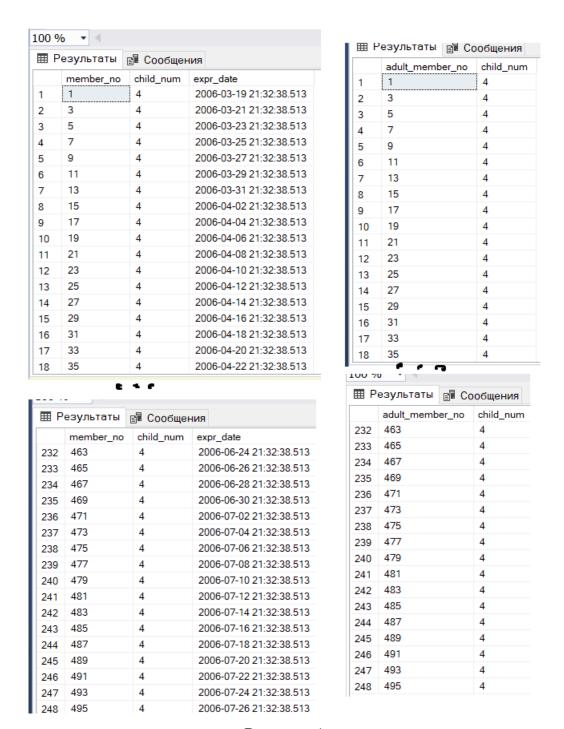


Рисунок 1

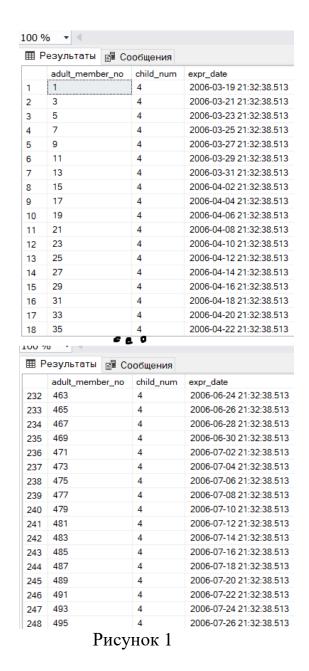
Проверка успешного формирования производной таблицы при помощи запроса

```
SELECT adult_member_no, COUNT(member_no) as child_num
FROM juvenile
GROUP BY adult_member_no
HAVING COUNT(member_no) > 3
```

2. Создание запроса производной таблице в виде двух разных запросов. Предыдущий запрос был переписан так, чтобы он выглядел как два разных запроса, показывающие работу с производными таблицами. Написан

запрос, который возвращает поле adult_member_no, вычисляет количество детей для каждого взрослого читателя и возвращает те записи, в которых детей в таблице juvenile больше 3. Напишем еще один запрос, который возвращает значение поля expr_date таблицы adult. Перепишем запрос так, чтобы он использовал JOIN:

```
WITH info 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 info
Результат выполнения запроса показан на рисунке 2
```



Упражнение 2 – использование подзапросов как выражений

1. Использован подзапрос с одним значением (single-value). Был написан запрос, возвращающий значения полей member.firstname, member.lastname, loanhist.isbn и loanhist.fine paid для читателей библиотеки, заплативших максимальных штраф за все книги.

Написан запрос, который возвращает максимальное значение поля loanhist.fine_paid

SELECT MAX(loanhist.fine paid) FROM loanhist

Результат выполнения запроса показан на рисунке 3

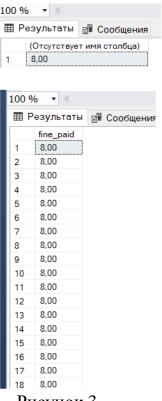


Рисунок 3

Проверка успешного возвращения максимального значения поля была выполнена при помощи запроса

```
SELECT fine_paid
FROM loanhist
ORDER BY fine paid DESC
```

- 2. Использование подзапроса как части условия поиска:
 - 2.1 Написан запрос, который соединяет таблицы member и loanhist и возвращает значения firstname, lastname, isbn и fine_paid для всех строк.
 - 2.2 Использование запроса из шага 1 предыдущего задания как критерий выбора в предложении WHERE так, чтобы возвращались только те записи, в которых штраф
 - 2.3 Включение ключевого слова DISTINCT.

```
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)

Результат выполнения запроса показан на рисунке 4

| 100 % 🔻 | | | | | | |
|--|-----------|---------------|------|-----------|--|--|
| Ш Результаты Ш Сообщения | | | | | | |
| | firstname | lastname | isbn | fine_paid | | |
| 1 | Angela | Hightower | 221 | 8,00 | | |
| 2 | Clair | Rothenberg | 32 | 8,00 | | |
| 3 | Karl | Wolfe-Hellene | 403 | 8,00 | | |
| 4 | William | Erickson | 30 | 8.00 | | |

Рисунок 4

3. Использование запросов для создания списка значений. Написан запрос к таблицам title, loan и reservation, возвращающий значения 4 полей: title_no, title, isbn и Total Reserved. Поле Total Reserved представляет собой количество резервных экземпляров для каждой книги. Отображены те записи, которых в резерве или более 50, или менее 5. Написан запрос, возвращающий номера isbn из таблицы reservations для книг, у которых более 50 копий, или менее 5.

100.06 - 4

```
SELECT isbn, COUNT(*) as total_reserved FROM reservation GROUP BY isbn HAVING COUNT(*) > 50 or COUNT(*) < 5 Результат выполнения запроса показан на рисунке 5
```

| 100 % ▼ | | | | | | |
|---------|-------|---------|----------|--|--|--|
| ⊞F | езуль | г Сооби | | | | |
| | isbn | total_ | reserved | | | |
| 1 | 1 | 197 | | | | |
| 2 | 43 | 196 | | | | |
| 3 | 246 | 196 | | | | |
| 4 | 288 | 197 | | | | |
| 5 | 330 | 196 | | | | |
| 6 | 533 | 196 | | | | |
| 7 | 575 | 197 | | | | |
| 8 | 617 | 196 | | | | |
| 9 | 820 | 196 | | | | |
| 10 | 862 | 197 | | | | |
| 11 | 904 | 196 | | | | |
| | | | | | | |

Рисунок 5

- 4. Использование подзапроса с несколькими значениями. Написан внешний запрос, который возвращает поля title_no, title, isbn и Total Reserved, где Total Reserved это число записей для каждой группы остальных полей:
- 4.1 Ограничено число записей, задав условие на количество копий книг, которое должно быть менее 5

4.2 Использовано ключевое слово IN, как часть предложения WHERE для списка, созданного на шаге 1 предыдущего задания

```
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)
Pesyльтат выполнения запроса показан на рисунке 6
```

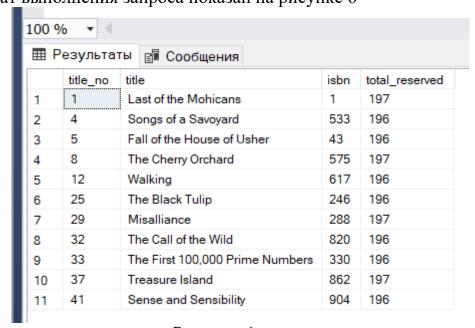


Рисунок 6

Упражнение 3 – использование коррелированных подзапросов

В этом задании нужно создать запрос, использующий коррелированный подзапрос для вычисления значений, основанных на данных из внешнего запроса, и использующий это значения как часть условия сравнения. Необходимо отобразить список читателей, имеющих сумму штрафов, превышающую 5 у. е.

1. Написан внешний запрос, возвращающий значения полей member_no и lastname. Псевдоним для таблицы member использован.

- 2. Написан внутренний запрос, вычисляющий общий штраф для каждого читателя. Для этого используем псевдоним для таблицы loanhist
- 3. Связаны поля member.member_no из внешнего запроса с полем loanhist.member no внутреннего подзапроса.
- 4. Использован оператор сравнения в предикате where

```
SELECT fines.member_no, lastname, fines.total_fine
FROM member as mem
INNER JOIN (SELECT member_no, SUM(fine_assessed) as to-
tal_fine
FROM loanhist
GROUP BY member_no) as fines ON fines.member_no =
mem.member_no
WHERE fines.total_fine > 5
```

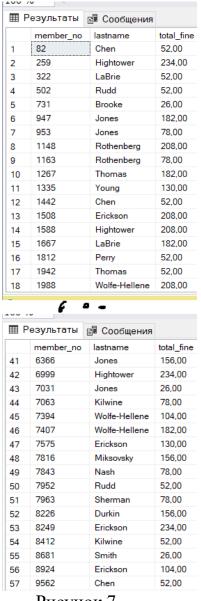


Рисунок 7

Выводы: в ходе выполнения данной лабораторной работы была достигнута ее цель — были получены навыки в написании и применении вложенных запросов. Также, были получены навыки 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

Методические указания к лабораторным работам / Сост.: А. В. Горячев, Н.Е. Новакова. СПб.: Изд-во СПБГЭТУ «ЛЭТИ», 2008. 32 с.