

МИНОБРНАУКИ РОССИИ
Санкт-Петербургский государственный
электротехнический университет
«ЛЭТИ» им. В.И. Ульянова (Ленина)
Кафедра систем автоматизированного проектирования (САПР)

отчет
по лабораторной работе № 5
по дисциплине «Базы данных»
Тема: «Работа с вложенными запросами»

Студенты гр. 3311

Преподаватель

Аршин А. Д

Баймухамедов Р. Р.

Пасечный Л. В.

Новакова Н. Е.

Санкт-Петербург

2025

Цель работы

Научиться писать и применять вложенные запросы. В лабораторной работе используется БД Library.

Упражнение 1

Выполним запрос, использующий производные таблицы. Запрос, формирующий производную таблицу, будет возвращать столбец `juveline.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 будет включать поля `adult_member_no` и `No_Of_Children` из производного запроса и поле `expr_date` из таблицы `adult`. Выполним запросы, использующие производные таблицы, а также в виде двух запросов и общие табличные выражения

Запрос

```
SELECT jv.adult_member_no, No_Of_Children, ad.expr_date
FROM adult AS ad
JOIN
(
SELECT adult_member_no, COUNT(member_no) AS 'No_Of_Children'
FROM juvenile
GROUP BY adult_member_no
HAVING COUNT(member_no)>3
) AS jv
ON ad.member_no = jv.adult_member_no
```

Результат

adult_member_no	No_Of_Children	expr_date
1	4	2006-03-19 21:32:38.513
3	4	2006-03-21 21:32:38.513
5	4	2006-03-23 21:32:38.513

7	4	2006-03-25 21:32:38.513
9	4	2006-03-27 21:32:38.513...
...

(Затронуто строк: 248)

Время выполнения: 2025-10-08T20:30:27.1595373+03:00

Запрос

```
SELECT ad.member_no, jv.count_child, expr_date
FROM (
    SELECT adult_member_no, COUNT(member_no) AS count_child
    FROM juvenile
    GROUP BY adult_member_no
    HAVING COUNT(member_no)>3
) AS jv
JOIN adult AS ad
ON ad.member_no = jv.adult_member_no
```

Результат

member_no	count_child	expr_date
1	4	2006-03-19 21:32:38.513
3	4	2006-03-21 21:32:38.513
5	4	2006-03-23 21:32:38.513
7	4	2006-03-25 21:32:38.513
9	4	2006-03-27 21:32:38.513
...

(Затронуто строк: 248)

Время выполнения: 2025-10-13T14:58:36.6624610+03:00

Запрос

```
WITH adult_w_kids AS(
```

```

SELECT jv.adult_member_no, COUNT(jv.member_no) AS
'count_of_child', ad.expr_date
FROM adult AS ad
JOIN
juvenile AS jv
ON ad.member_no = jv.adult_member_no
GROUP BY jv.adult_member_no, ad.expr_date
HAVING COUNT(jv.member_no)>3
)

```

```

SELECT adult_member_no, count_of_child, expr_date
FROM adult_w_kids

```

Результат

adult_member_no	count_of_child	expr_date
1	4	2006-03-19 21:32:38.513
3	4	2006-03-21 21:32:38.513
5	4	2006-03-23 21:32:38.513
7	4	2006-03-25 21:32:38.513
9	4	2006-03-27 21:32:38.513
...

(Затронуто строк: 248)

Время выполнения: 2025-10-08T21:19:07.4869212+03:00

Упражнение 2

Использование подзапроса как части условия поиска

Напишем запрос, который соединяет таблицы member и loanhist и возвращает значения firstname, lastname, isbn и fine_paid для всех строк. Используем этот запрос как критерий выбора в предложении WHERE, так чтобы возвращались

только те записи, в которых штраф имеет максимальное значение. Включим ключевое слово DISTINCT.

Запрос

```
WITH who_what_paid AS (  
    SELECT mb.firstname, mb.lastname, lh.isbn, lh.fine_paid  
    FROM  
    member AS mb  
    JOIN  
    loanhist AS lh  
    ON mb.member_no = lh.member_no  
)  
  
SELECT DISTINCT firstname, lastname, isbn, fine_paid  
FROM who_what_paid  
WHERE fine_paid = (  
    SELECT MAX(fine_paid)  
    FROM who_what_paid  
)
```

Результат

firstname	lastname	isbn	fine_paid
Angela	Hightower	221	8,00
Clair	Rothenberg	32	8,00
Karl	Wolfe-Hellene	403	8,00
William	Erickson	30	8,00

(Затронуто строк: 4)

Время выполнения: 2025-10-08T21:11:57.8384799+03:00

Теперь напишем запрос к таблице 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. Нужно отобразить те записи, которых в резерве или более 50, или менее 5.

Запрос

```
SELECT tt.title_no, tt.title, ln.isbn, COUNT(*) AS 'Total_Reserved'
FROM title AS tt
JOIN loan AS ln
ON tt.title_no = ln.title_no
WHERE ln.isbn IN (
    SELECT isbn
    FROM reservation
    GROUP BY isbn
    HAVING COUNT(*)>50 OR COUNT(*)<5
)
GROUP BY tt.title_no, tt.title, ln.isbn
```

Результат

title_no	title	isbn	Total_Reserved
1	Last of the Mohicans	1	2
5	Fall of the House of Usher	43	2
25	The Black Tulip	246	2
29	Misalliance	288	2
33	The First 100,000 Prime Numbers	330	2
...

(Затронуто строк: 11)

Время выполнения: 2025-10-08T21:59:49.7759635+03:00

Упражнение 3

Используем коррелированные подзапросы. В этом задании создадим запрос, использующий коррелированный подзапрос для вычисления значений, основанных на данных из внешнего запроса, и использующий эти значения

как часть условия сравнения. Необходимо отобразить список читателей, имеющих сумму штрафов, превышающую 5 у.е.

Запрос

```
SELECT mb.member_no, mb.lastname
FROM member AS mb
WHERE (
    SELECT SUM(fine_assessed) AS 'total_fine'
    FROM loanhist AS lh
    WHERE lh.member_no = mb.member_no
) > 5
```

Результат

member_no	lastname
82	Chen
259	Hightower
322	LaBrie
502	Rudd
731	Brooke
...	...

Внимание! Значение NULL исключено в агрегатных или других операциях SET.

(Затронуто строк: 57)

Время выполнения: 2025-10-08T22:21:39.5139467+03:00

Вывод

В рамках лабораторной работы закрепились базовые принципы использования вложенных запросов в SQL. На практике рассмотрены и применены разные типы подзапросов:

1. Производные таблицы. Освоено построение временных наборов данных и их соединение через 'JOIN', что позволяет структурировать сложные запросы и повышать их читаемость.

2. Подзапросы-выражения. Изучено применение скалярных подзапросов для получения единственного значения и последующей фильтрации данных в условиях `WHERE`.
3. Подзапросы со списками значений. Отработано использование оператора `IN` для работы с наборами ключей, формируемых внутренними выборками.