

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

ОТЧЕТ
по лабораторной работе № 5
по дисциплине "Базы данных"
Тема: "Работа с вложенными запросами"

Студент гр. 2311

Завьялов Н.С.

Князев И.М.

Преподаватель

Новакова Н.Е.

Санкт-Петербург

2024

Цель работы: Научиться писать и применять вложенные запросы

Используемая база данных: Library

Упражнение 1 – использование вложенных запросов как производных таблиц.

Запрос 1. Использование производной таблицы в виде единого запроса:

```
SELECT  SubQuery.adult_member_no,
        SubQuery.No_Of_Children,
        adu.expr_date
FROM
    (SELECT juv.adult_member_no,
            COUNT(*) AS No_Of_Children
    FROM library.dbo.juvenile AS juv
    GROUP BY juv.adult_member_no
    HAVING COUNT(*) > 3) AS SubQuery
JOIN library.dbo.adult AS adu
    ON adu.member_no = SubQuery.adult_member_no;
```

Результаты выполнения запроса

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
...		
495	4	2006-07-26 21:32:38.513

(248 rows affected)

Completion time: 2024-10-02T17:39:32.0996817+03:00

Запрос 2. Извлечение таблицы juvenile значения поля adult_member_no и количество значений поля member_no для каждого значения adult_member_no

таких, что их количество будет более 3 и извлечение всех данных из полученной таблицы :

```
WITH test AS
  (SELECT juvenile.adult_member_no,
          COUNT(juvenile.member_no) AS No_Of_Children,
          adult.expr_date
   FROM juvenile
   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 test;
```

Результаты выполнения запроса

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
...		
495	4	2006-07-26 21:32:38.513

(248 rows affected)

Completion time: 2024-10-02T17:39:32.0996817+03:00

Упражнение 2 – использование подзапросов как выражений.

Запрос 1. Извлечение максимального значения поля fine_paid из таблицы loanhist :

```
SELECT MAX(loanhist.fine_paid)
FROM loanhist;
```

Результаты выполнения запроса

8,00

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

(1 row affected)

Completion time: 2024-10-15T01:09:16.4288220+03:00

Запрос 2. извлечение уникальных полей firstname и lastname из таблицы member, полей isbn и fine_paid из таблицы loanhist, где значение поля loanhist равно максимальному значению поля fine_paid. :

```
SELECT DISTINCT member.firstname,
                 member.lastname,
                 loanhist.isbn,
                 loanhist.fine_paid
FROM loanhist
JOIN member ON member.member_no = loanhist.member_no
WHERE fine_paid =
      (SELECT MAX(loanhist.fine_paid)
       FROM loanhist);
```

Результаты выполнения запроса

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

(4 rows affected)

Completion time: 2024-10-15T01:14:24.3147275+03:00

Запрос 3. извлечение значений поля isbn из таблицы reservation и количества значений остальных полей таблицы reservation как поля total_reserved, для которых количество значений остальных полей больше 50 или меньше 50.

Результат сгруппирован по полю isbn. :

```
SELECT isbn,  
        COUNT(*) AS total_reserved  
FROM reservation  
GROUP BY isbn  
HAVING COUNT(*) > 50 OR COUNT(*) < 5;
```

Результаты выполнения запроса

isbn	total_reserved
1	197
43	196
...	
862	197
904	196

(11 rows affected)

Completion time: 2024-10-15T01:28:43.9196892+03:00

Запрос 4. возвращающий поля title_no, title, isbn и Total Reserved, где Total Reserved – это число записей для каждой группы остальных полей, количество копий книг должно быть меньше 5. :

```

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)

```

Результаты выполнения запроса

title_no	title	isbn	total_reserved
----------	-------	------	----------------

1	Last of the Mohicans		
1	197		
...			
41	Sense and Sensibility		
904	196		

(11 rows affected)

Completion time: 2024-10-15T01:35:31.0444670+03:00

Упражнение 3 – использование коррелированных подзапросов.

Запрос 1. использование коррелированного подзапроса для извлечение полей member_no и lastname из таблицы member и поля fine, которое содержит значение суммы поля fine_assessed для каждого member_no из таблицы loanhist, где данное значение больше 5. :

```
SELECT m.member_no,  
       m.lastname,  
       f.fine  
FROM member AS m  
JOIN  
    (SELECT member_no,  
             SUM(fine_assessed) AS fine  
     FROM loanhist  
     GROUP BY member_no) AS f ON f.member_no = m.member_no  
WHERE f.fine > 5;
```

Результаты выполнения запроса

member_no	lastname	fine
82	Chen	52,00
259	Hightower	234,00
...		
8924	Erickson	104,00
9562	Chen	52,00

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

(57 rows affected)

Completion time: 2024-10-15T01:39:34.9506734+03:00

Вывод

В данной лабораторной работе было проведено знакомство с вложенными запросами.

Вложенный запрос (подзапрос) — это запрос, который используется внутри SQL инструкции или внутри другого вложенного запроса. Подзапрос может быть использован везде, где разрешены выражения (в командах SELECT, опциях FROM, WHERE). Выполняется сначала вложенный, затем внешний. Внешний запрос — инструкция, в которой используется вложенный запрос.

В упражнении №1 вложенные запросы использовались как производные таблицы. Производная таблица — это табличное выражение при использовании опции FROM внешнего (основного) запроса.

В упражнении №2 подзапросы использовались как выражения. В этом случае подзапрос встроен в предложение WHERE. Подзапрос используется для возврата данных, которые будут использоваться в основном запросе в качестве условия для дальнейшего ограничения данных, подлежащих извлечению.

В упражнении №3 использовались коррелированные подзапросы. Результаты коррелированных запросов зависят от строк, которые выбираются в основном запросе, т.к. такие подзапросы ссылаются на значения столбцов внешнего запроса.