

ЛАБОРАТОРНАЯ РАБОТА №8

Тема: Раздел FROM предложения SELECT. Внутренние запросы в качестве источника данных.

Цель: Пробрести навыки использования различных видов соединений при создании SQL запросов.

В данной лабораторной работе используются база данных HR, созданная в рамках первой лабораторной работы. Также используется таблица TB_ELEKTROSTAL_2018, которая создана и наполнена данными в процессе выполнения лаб. работы №.1.

Дополнительно используется удаленная база данных проекта stackoverflow.com. Структура этой базы данных описана в методических указаниях к выполнению лабораторной работы № 3. Напомним, адрес ресурса <http://data.stackexchange.com/stackoverflow/query/new>.

Подзапросы обычно содержатся в разделах **where** или **having** SQL оператора **select** или в списке выбора этого оператора. С помощью подзапросов можно проводить дальнейший отбор данных из результатов других запросов. Оператор, содержащий подзапрос, может обрабатывать строки некоторой таблицы, основываясь на результатах вычисления списка выбора подзапроса, который в свою очередь может ссылаться на эту же таблицу как внешний запрос, или обращаться к другой таблице. В языке SQL подзапросы допускаются почти везде, где допускаются выражения, если подзапрос возвращает одно значение.

Операторы выбора, содержащие один или несколько подзапросов, называются также составными запросами или составными операторами выбора. Возможность включения одного оператора выбора внутрь другого является одной из причин, по которой язык SQL называется “структурированным” (Structured Query Language).

SQL оператор, который включает подзапросы, называемые также внутренними запросами, можно иногда заменить соединением. Есть вопросы,

которые можно сформулировать только с помощью подзапросов. Некоторые люди предпочитают всегда использовать подзапросы, поскольку находят их легкими для понимания. Другие стремятся их избегать всегда, когда это возможно. Разработчик может выбрать сам удобный для него способ.

Такие запросы обычно используются для получения данных из двух и более таблиц. Они нужны чтобы данные из разных таблиц можно было соотнести и по зависимости осуществить выборку.

Существуют два основных типа подзапросов:

- Подзапросы, которым предшествует немодифицированная операция сравнения и которые возвращают единственное значение, называются подзапросами-выражениями (скалярными подзапросами).
- Подзапросы, которые возвращают список значений и которым предшествует ключевое слово `in` (принадлежит) или операция сравнения, модифицированная кванторами `any` (некоторый) или `all` (все), а также подзапросы, проверяющие существование с помощью квантора `exists` (существует), называются квантифицированными предикатными подзапросами.

Подзапросы любого из этих типов могут быть либо коррелированными (повторяющимися), либо некоррелированными.

- Некоррелированный подзапрос может вычисляться как независимый запрос. Иначе говоря, результаты подзапроса подставляются в основной оператор (или внешний запрос). Это не значит, что SQL-сервер именно так выполняет операторы с подзапросами. Некорреляционные подзапросы могут быть заменены соединением и будут выполняться как соединения SQL-сервером.
- Коррелированные подзапросы не могут выполняться как независимые запросы, поскольку они могут обращаться к данным, находящимся в столбцах таблицы, указанной в разделе **from** внешнего запроса. Коррелированные подзапросы детально обсуждаются в конце этой главы.

Подзапрос можно использовать в разделе FROM оператора SELECT. При таком использовании их называют inline представлениями.

Рассмотрим пример, использующий БД OE. С этой БД можно работать на сайте livesql.oracle.com. Процесс регистрации на сайте описан в методических указаниях к выполнению лабораторной работы №1.

```
SELECT a.cust_first_name||' '||a.cust_last_name
      , subquery1.total_summa
FROM OE.customers,
      (SELECT customer_id
        , SUM(orders.order_total) AS total_summa
        FROM OE.orders
        GROUP BY customer_id) subquery1
WHERE
      subquery1.customer_id = a.customer_id;
```

В этом примере мы создали подзапрос в предложении FROM следующим образом:

```
(SELECT customer_id
      , SUM(orders.order_total) AS total_summa
FROM OE.orders
GROUP BY customer_id) subquery1
```

Этот подзапрос с псевдонимом subquery1. Subquery1 используется для ссылки на этот подзапрос или любое из его полей.

Приведем более сложный запрос. Здесь требуется вывести список из двух колонок. В первой колонке – имя работника. Во второй колонке – список занимаемых им в разное время должностей в течение периода работа на данном предприятии. Для формирования списка используется агрегатная функция *listagg*.

```
select e.first_name||' '||e.last_name
      , f.joblist
from hr.employees e
      inner join (
          select d.employee_id
                , listagg(c.job_title,'; ') as joblist
          from hr.jobs c
                inner join (
```

```

select b1.employee_id
      , b1.job_id from (
          SELECT a.employee_id
                , a.job_id
          from hr.job_history a
          union all
          select b.employee_id
                , b.job_id
          from hr.employees b ) b1
      ) d on d.job_id=c.job_id
group by d.employee_id) f on e.employee_id =
f.employee_id

```

Цветом выделены подзапросы разного уровня вложенности. Существуют и другие средства составить запрос, отвечающие на поставленную задачу. Цель этого примера продемонстрировать механизм использования вложенных запросов в качестве источника данных раздела FROM предложения SELECT.

Порядок выполнения работы

Составить следующие запросы

- Составить запрос к БД HR. Результат запроса должен содержать две колонки. Первая колонка - имя и фамилия работника. Вторая колонка – первая должность, которую он занял при поступлении на предприятие. Указание: необходимо использовать механизм вложенных запросов при построении предложения SQL.
- Составить запрос к БД HR. Результат запроса должен содержать две колонки. Первая колонка – название должности. Вторая колонка перечень имен и фамилий сотрудников, работающих в этой должности в настоящее время. В качестве разделителя между реквизитами сотрудников использовать символ «точка с запятой» Указание: необходимо использовать механизм вложенных запросов при построении предложения SQL.
- Составить запрос к БД stackoverflow. Результат запроса должен содержать список имен пользователей, которые писали посты как в 2010 году так и в

2015 году. Для составления запроса обязательно требуется использовать вложенные запросы в качестве источника данных.

- Составить запрос к БД `stackoverflow`. Результат запроса должен содержать список имен пользователей, которые писали посты в 2010 году, но не написали ни одного поста в 2015 году. Для составления запроса обязательно требуется использовать вложенные запросы в качестве источника данных.
- Составить запрос к БД `stackoverflow`. Результат запроса должен содержать список имен пользователей, которые не написали ни одного поста в 2010 году, но писали посты в 2015 году. Для составления запроса обязательно требуется использовать вложенные запросы в качестве источника данных.

Содержание отчета:

1. Тема, цель лабораторной работы.
2. Примеры выполнения запросов к базе данных.
3. Составленные согласно заданию запросы и скриншоты полученных результатов.
5. Выводы.

Контрольные вопросы:

1. Опишите разновидности вложенных запросов.
2. Для чего используется функция `listagg()`? Объясните ее параметры и особенности использования.
3. Охарактеризуйте механизм ссылок на поля вложенного запроса.
4. Можно ли использовать вложенные запросы в операциях `JOIN`?
5. Во втором примере вложенного запроса список занимаемых должностей не упорядочен хронологически. Какие изменения нужно внести в запрос для упорядочения элементов списка должностей каждого сотрудника в хронологическом порядке?
6. Назовите количество вложенных запросов в примерах указанных выше.

7. Предложите свои примеры использования вложенных запросов в качестве источника данных раздела FROM предложения SELECT.

Файл: Лабораторная работа 8
Каталог: C:\Users\sss\Documents
Шаблон: C:\Users\sss\AppData\Roaming\Microsoft\Шаблоны\Normal.dotm
Заголовок:
Содержание:
Автор: sss
Ключевые слова:
Заметки:
Дата создания: 14.11.2018 9:49:00
Число сохранений: 30
Дата сохранения: 18.11.2018 15:51:00
Сохранил: sss
Полное время правки: 283 мин.
Дата печати: 18.11.2018 15:51:00
При последней печати
 страниц: 6
 слов: 1 243 (прибл.)
 знаков: 7 088 (прибл.)