

ЛАБОРАТОРНАЯ РАБОТА №5

Тема: Структура предложения SELECT. Методы конструирования списка полей запроса (продолжение).

Цель: Пробрести навыки составления списка полей запроса с использованием агрегатных функций и внутренних 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>.

Использование агрегатных функций

Стандарт языка SQL предусматривает использование пяти агрегатных функций. Каждая такая функция на вход принимает 0 или более строк и выдает на выходе одно значение. Перечислим эти функции

count() – функция подсчета количества строк.

sum() - функция суммирования заданного выражения на множестве строк.

avg() - подсчет среднего значения заданного выражения на множестве строк.

min() - возвращает минимальное значение выражения на множестве строк.

max() - возвращает максимальное значение выражения на множестве строк.

Отдельные марки СУБД поддерживают дополнительные агрегатные функции. Но перечисленные выше 5 агрегатных функций входят в стандарт языка SQL.

Примеры.

Подсчет количества строк в таблице posts

```
select count(*) from posts
```

Определение максимального оклада среди окладов сотрудников

```
select max(salary) from employees
```

Внутренние запросы в качестве элементов списка полей.

В реальных SQL запросах часто встречаются внутренние подзапросы. Их можно встретить в разных ролях и частях основного SQL запроса. Уровень вложенности запросов может быть весьма глубоким. Одним из мест, где часто используются внутренние подзапросы, является список полей основного запроса. Любой элемент списка полей должен возвращать строго одно значение. Такой конструкцией возвращающей одно значение может быть внутренний запрос.

Внутренние запросы в качестве элемента списка полей часто используются в качестве инструмента связывания данных из различных источников (таблиц)

Пусть требуется вывести для каждого департамента из БД HR его название и адрес. Анализ структуры БД показывает, что название департамента (department_name) хранится в таблице departments. Адрес департамента содержится (поле street_address) содержится в таблице locations. В языке SQL существует несколько способов связывания таблиц в один источник данных запроса. Одним из таких способов является построение внутреннего запроса, который будет использоваться в качестве элемента списка полей запроса. В рассматриваемом примере запрос будет выглядеть следующим образом.

```
select department_name
```

```
, (select street_address from locations
      where departments.location_id=locations.location_id)
from departments
```

В приведенном операторе выборки вложенным запросом будет выражение `, (select street_address from locations where departments.location_id = locations.location_id)`. Этот запрос имеет несколько особенностей. Во-первых, он является коррелированным (связанным). Это означает, что попытка его выполнить самостоятельно закончится неудачно. Причиной это является использование во внутреннем запросе данных их внешнего запроса (`departments.location_id`). Во-вторых, следует отметить типичную ситуацию, когда в двух таблицах имеется поля с одинаковым именем. В нашем случае поле `location_id` присутствует в таблицах `departments` и `locations`. Для снятия неопределенности по поводу одинаковых имен полей в SQL принято уточнять название поля префиксом – названием таблицы. Например, `departments.location_id` этот идентификатор относится к полю `location_id` таблицы `departments`. Имеется и другой способ уточнения принадлежности поля – это механизм присвоения временных псевдонимов таблицам. Синоним записывается сразу после названия таблицы. В качестве таких псевдонимов принято использовать строчные латинские буквы. Затем он помещается перед названием поля, принадлежность которого требуется уточнить. С использованием этого механизма предыдущий пример может быть переписан следующим образом:

```
select a.department_name
      , (select street_address from locations b
      where a.location_id=b.location_id)
from departments a
```

Иногда можно встретить использование не связанных (не коррелированных) внутренних запросов в качестве элемента списка полей запроса. Отметим, что не связанный запрос может быть выполнен отдельно,

т. е. независимо от внешнего запроса. Например, требуется подсчитать количество записей в таблице employees и количество записей в таблице departments.

```
select (select count(*) from employees)
       , (select count(*) from departments)
       from dual
```

В рассмотренном примере внутренние запросы (select count(*) from employees) и (select count(*) from departments) являются не связанными. Они могут выполнены самостоятельно.

Следует напомнить, что таблица dual в СУБД ORACLE используется в тех случаях, когда смысл запроса вообще не требует использования таблиц. Но синтаксические правила обязывают использовать раздел FROM. В СУБД MsSQL можно опустить раздел FROM. Эта возможность является расширением этой СУБД. Подобные расширения предусмотрены в СУБД MySQL и PostgreSQL.

Порядок выполнения работы

Порядок выполнения работы.

Составить следующие запросы

- Подсчитать количество разных названия улиц в таблице tb_elekrostal_2018. Указание: использовать модификатор для задания уникальности выражений.
- Подсчитать сумму премии сотрудникам из таблицы employees.
- Составить запрос к БД stackoverflow. Результат запроса должен содержать две колонки. Первая колонка содержание поста. Вторая колонка имя пользователя, создавшего этот пост. Указание: необходимо использовать внутренний запрос для вывода второй колонки.
- Подсчитать количество типов улиц в таблице tb_elektrostal_2018.

- Составить запрос (СУБД stackoverflow), который возвращает одну строку из трех чисел. Первое число – количество строк в таблице posts. Второе число – количество строк в таблице users. Третье число – произведение количества строк таблицы posts на количество строк таблицы comments.

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. Какие агрегатные функции Вы знаете?
2. Охарактеризуйте особенности использования агрегатных функций.
3. Какие входные параметры указываются в агрегатных функциях COUNT() и SUM().
4. Опишите особенности использования внутренних запросов в качестве элементов списка полей предложения.
6. Отличия не связанных и связанных внутренних запросов.
7. Цели использования таблицы dual. В СУБД какой марки используется эта таблица.
8. Приведите пример, когда внутренний запрос, используемый в качестве элемента списка полей предложения SELECT возвращает несколько строк. Опишите реакцию среды выполнения.
9. Приведите пример, когда внутренний запрос, используемый в качестве элемента списка полей предложения SELECT возвращает строку, которая состоит из нескольких полей. Опишите реакцию среды выполнения.

Файл: Лабораторная работа 5
Каталог: C:\Users\sss\Documents
Шаблон: C:\Users\sss\AppData\Roaming\Microsoft\Шаблоны\Normal.dotm
Заголовок:
Содержание:
Автор: sss
Ключевые слова:
Заметки:
Дата создания: 06.11.2018 20:59:00
Число сохранений: 33
Дата сохранения: 08.11.2018 22:11:00
Сохранил: sss
Полное время правки: 375 мин.
Дата печати: 08.11.2018 22:11:00
При последней печати
 страниц: 5
 слов: 1 090 (прибл.)
 знаков: 6 214 (прибл.)