

ЛАБОРАТОРНАЯ РАБОТА №10

Тема: Раздел WHERE предложения SELECT. Предикаты EXISTS, ANY, SOME, ALL.

Цель: Пробрести навыки использования предикатов EXISTS, ANY, SOME, ALL при составлении логических условий включения строк в результирующий набор данных.

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>.

В разделе WHERE задаются условия отбора записей в результат запроса, т.е. условие которому должна удовлетворять запись, чтобы попасть в результат запроса, аналог операции селекции в реляционной алгебре.

В выражении условий раздела WHERE могут быть использованы следующие предикаты:

Использование подзапросов в разделе **WHERE** предложения **SELECT** возможно несколькими способами. Стандартом языка предусмотрены несколько предикатов, специально предназначенных для анализа результатов выполнения подзапросов:

- предикаты сравнения «для некоторых/для всех» - **ANY, SOME, ALL**;
- предикат вхождения **IN**;
- предикат существования **EXISTS**.

В первом способе конструкция запроса может выглядеть следующим образом:

SELECT... FROM...

WHERE Выражение <cmp>
[**ANY**|**SOME**|**ALL**] (SELECT...);
где <cmp> - оператор сравнения.

Как правило, выражение включает поле из списка полей внешней SQL-инструкции или функцию от этих полей. Внутренняя инструкция **SELECT** должна возвращать набор данных по одному полю или по вычисляемому полю, при принципиальной сравнимости с выражением во внешней SQL-инструкции (главным образом по типу данных).

Предикаты **ANY** и **SOME**, являющиеся синонимами, используются для отбора в главной SQL-инструкции тех записей, которые удовлетворяют сравнению с какой-либо записью (т. е. хотя бы с одной), из отобранных во внутренней инструкции **SELECT**.

Для примера рассмотрим запрос к БД HR, который выводит список работников (таблица **employees**), сменивших должность не менее двух раз. Для анализа количества изменений должности можно использовать таблицу **JOB_HISTORY**. Код текущей должности для каждого работника указан в таблице **employees**. Если в таблице **JOB_HISTORY** есть не менее двух записей по работнику, то он менял должность не менее двух раз. Сформулировать SQL запрос для решения поставленной задачи можно разными способами. Приведем ее решение с помощью предиката **ANY**.

```
SELECT first_name||' '||last_name
FROM employees
WHERE employee_id = any (
    SELECT employee_id
    FROM job_history
    GROUP BY employee_id
    HAVING count(*)>=2
);
```

Предикат **ALL** (для всех) используется для отбора в главном запросе только тех записей, которые удовлетворяют сравнению одновременно со всеми записями, отобранными в подчиненном запросе. Пример выполнения запроса с предикатом **ALL** может быть следующим. Пусть требуется вывести всех сотрудников, имеющих профессию **Programmer**, которые получают

оклад (salary) больше всех (любого) сотрудника, имеющего профессию Stock Manager.

```
SELECT first_name||' '||last_name
      FROM employees a
         inner join jobs b using(job_id)
      WHERE b.job_title='Programmer'
         and a.salary>all (
           SELECT salary
             FROM employees a1
                inner join jobs b1 using(job_id)
           WHERE b1.job_title='Stock Manager')
```

Принцип действия запроса с использованием предиката IN в поиске среди результирующего набора записей внешней SQL-инструкции тех записей, для которых значение определенного выражения входит в список записей, отбираемых внутренней инструкцией SELECT. Конструкция запроса с предикатом **IN** выглядит следующим образом:

```
SELECT ... FROM ...
WHERE Выражение [NOT]IN (SELECT...);
```

В качестве примера перепишем запрос о сотрудниках, сменивших должность не менее 2-х раз, с использованием предиката **IN**.

```
SELECT first_name||' '||last_name
      FROM employees
      WHERE employee_id IN (
          SELECT employee_id
            FROM job_history
          GROUP BY employee_id
          HAVING count(*)>=2
        );
```

Следует добавить, что предикат **NOT IN** используется для отбора во внешней SQL-инструкции только тех записей, которые содержат значения, не совпадающие ни с одним из отобранных внутренней инструкцией SELECT.

В третьем способе построения подчиненного запроса предикат **EXISTS** (с необязательным зарезервированным словом **NOT**) используется в логическом выражении для определения того, должен ли подчиненный запрос возвращать какие-либо записи. Исходя из этого, каждая запись,

отбираемая во внешней SQL-инструкции, идет в итоговый набор данных только тогда, когда при ее условиях отбирается (существует) хотя бы одна запись по внутренней инструкции SELECT. Конструкция запроса в этом случае может выглядеть следующим образом:

```
SELECT...FROM...WHERE ([NOT]Exists(SELECT...)) ;
```

В качестве примера приведен запрос о сотрудниках, сменивших должность не менее 2-х раз, с использованием предиката **EXISTS**.

```
ELECT first_name||' '||last_name  
FROM employees  
WHERE exists (  
        SELECT *  
        FROM job_history  
        WHERE      job_history.employee_id      =  
employees.employee_id  
        GROUP BY employee_id  
        HAVING  count(*)>=2  
);
```

Следует заметить, что использование предиката **NOT EXISTS** сформирует список сотрудников, сменивших должность менее 2-х раз. В последнем примере можно также увидеть, что альтернативным решением для реализации такого запроса является использование запроса на внутреннее соединение (INNER JOIN).

Порядок выполнения работы

Порядок выполнения работы.

Составить следующие запросы

- С использованием предиката **ANY** составить запрос к БД HR. Результат запроса должен содержать две колонки. Первая колонка - имя и фамилия работника (first_name, last_name). Вторая колонка – размер заработной платы (salary). В результат нужно включить только данные о работниках департамента с местом расположения в Зимбабве, которые получают зарплату размером менее размера заработной платы, получаемой хоть каким-либо работником департамента, расположенного в Лондоне.

- С использованием предиката **ALL** составить запрос к БД HR. Результат запроса должен содержать две колонки. Первая колонка - имя и фамилия работника (first_name, last_name). Вторая колонка – размер заработной платы (salary). В результат нужно включить только данные о работниках департамента с местом расположения в Лондоне, которые получают зарплату размером более размера заработной платы, получаемой каждым работником департамента, расположенного в Зимбабве.
- С использованием предиката **IN** составить запрос к БД HR. Результат запроса должен содержать две колонки. Первая колонка - имя и фамилия работника (first_name, last_name). Вторая колонка – название должности. В результат нужно включить только данные о работниках департаментов, расположенных в Европе
- С использованием предиката **EXISTS** составить запрос к БД HR. Результат запроса должен содержать две колонки. Первая колонка - имя и фамилия работника (first_name, last_name). Вторая колонка – название должности. В результат нужно включить только данные о работниках департаментов, расположенных в Европе
- Составить запрос к БД stackoverflow. Результат запроса должен содержать список имен пользователей, из России, которые создали постов больше хоть какого-то пользователя из Польши. Указание необходимо использовать предикат **SOME**.
- Составить запрос к БД stackoverflow. Результат запроса должен содержать список имен пользователей, из России, которые создали постов больше любого какого-то пользователя из Польши. Указание необходимо использовать предикат **ALL**.
- Составить запрос к БД stackoverflow. Результат запроса должен содержать список имен пользователей, из России, которые создали более 10. Указание необходимо использовать предикат **EXISTS**.

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 назначение предиката **EXISTS** и особенности его использования в разделе **WHERE** предложения **SELECT**.
2. Опишите назначение предиката **ALL** и особенности его использования в разделе **WHERE** предложения **SELECT**.
3. Опишите назначение предиката **ANY** и особенности его использования в разделе **WHERE** предложения **SELECT**.
4. Опишите назначение предиката **SOME** и особенности его использования в разделе **WHERE** предложения **SELECT**.
5. Опишите назначение предиката **IN** и особенности его использования в разделе **WHERE** предложения **SELECT**.
6. Какие поля должен возвращать подзапрос, используемый в предикате **EXISTS**?
7. Какие поля должен возвращать подзапрос, используемый в предикате **IN**?
8. Какие поля должен возвращать подзапрос, используемый в предикате **ALL**?
9. Опишите особенности использования логической операции **NOT** с предикатами **ANY**, **SOME**, **EXISTS**, **ALL**, **IN**. Приведите пример запроса с использованием логической операции **NOT** с предикатом **EXISTS**.

Файл: Лабораторная работа 10
Каталог: E:\Projects\documents\md\predmety\BD_2018
Шаблон: C:\Users\sss\AppData\Roaming\Microsoft\Шаблоны\Normal.dotm
Заголовок:
Содержание:
Автор: sss
Ключевые слова:
Заметки:
Дата создания: 15.11.2018 20:53:00
Число сохранений: 47
Дата сохранения: 17.11.2018 17:02:00
Сохранил: sss
Полное время правки: 178 мин.
Дата печати: 17.11.2018 17:02:00
При последней печати
 страниц: 6
 слов: 1 386 (прибл.)
 знаков: 7 901 (прибл.)