

### ЛАБОРАТОРНАЯ РАБОТА №3

*Тема:* Простые запросы к базам данных на языке SQL.

*Цель:* Пробрести навыки использования оператора SELECT языка SQL.

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. Владельцы проекта разрешают удаленный доступ к одной из последних версий базы данных вопросов и ответов с помощью стандартных SQL запросов. Для получения доступа к базе данных регистраций не требуется. Адрес ресурса <http://data.stackexchange.com/stackoverflow/query/new>.

База данных имеет довольно сложную структуру. Но там ее часть, которая будет использована в лабораторных работах, описывается следующими таблицами.

**Users** — таблица с информацией о пользователях Поля (приведены только те, которые могут понадобиться):

Id — идентификатор пользователя;

Reputation — число обозначающее репутацию пользователя;

DisplayName — имя пользователя;

Location — местоположение пользователя.

**Posts** — таблица с информацией о сообщениях (постах)

Поля (приведены только те, которые могут понадобиться):

Id — идентификатор поста;

PostTypeId — тип поста, 1 — вопрос, 2 — ответ;

ParentId — (ссылка на) идентификатор вопроса (присутствует только для ответов — постов с PostTypeId = 2); CreationDate — дата и время создания поста;

Score — количество очков, полученных за это сообщение;  
OwnerUserId — (ссылка на) идентификатор пользователя;  
Tags — строка тегов (тем), к которым относится вопрос, разделённых  
“,”.

**Tags** — таблица с перечнем всех тегов Поля (необходимые):

Id — идентификатор тега;

TagName — название тега.

**PostTags** — таблица, связывающая вопросы и теги Поля:

PostId — ссылка на идентификатор поста;

TagId — ссылка на идентификатор тега.

Оператор SELECT используется для отбора записей, удовлетворяющих сложным критериям поиска, и имеет следующий формат:

SELECT [DISTINCT] { \* \ < Список полей > }

FROM < Список таблиц >

[WHERE < Условия отбора >]

[ORDER BY < Список полей для сортировки >]

[GROUP BY <Список полей для группирования >]

[HAVING < Условия группирования >]

[UNION <Вложенный оператор SELECT>]

Результат выполнения SQL-запроса, заданного оператором SELECT, представляет собой выборку записей, отвечающих заданным в нем условиям. При рассмотрении оператора SELECT будем предполагать, что SQL-запрос набран и выполнен с помощью в инструменте SQL Commands компонента SQL Workshop облачной среды APEX.ORACLE.COM. Регистрация в этой среде – это задача лабораторной работы №1. Напоминаем, что средой выполнения запросов является СУБД ORACLE. Помимо этого в Вашей рабочей области должны находиться база данных HR и таблица TB\_ELEKTROSTAL\_2018.

В результирующем наборе данных могут быть разрешены или запрещены повторяющиеся записи. Этим режимом управляет описатель DISTINCT. Если он отсутствует, то в наборе данных разрешаются повторяющиеся записи.

В оператор SELECT обязательно включается список полей и операнд FROM, остальные операнды могут отсутствовать. В списке операнда FROM перечисляются имена таблиц, для которых отбираются записи. Список должен содержать как минимум одно поле .

Операнд WHERE задает условие отбора. Выражение, описывающие условия отбора, является логическим. Его элементами могут быть имена полей, операции сравнения, скобки, специальные функции LIKE, NULL, IN и другие.

Операнд GROUP BY позволяет выделять группы записей в результирующем наборе данных.

Операнд HAVING используется для отбора записей внутри групп.

Операнд ORDER BY содержит список полей, определяющих порядок сортировки записей результирующего набора данных.

Оператор SELECT используется также внутри других операторов, например, операторов модификации записей, обеспечивая для их выполнения требуемый отбор записей.

### **Порядок выполнения работы**

Задан набор таблиц базы данных HR и таблица TB\_ELEKTROSTAL\_2018. Структура всех таблиц описана в лабораторной работе №1.

Порядок выполнения работы.

Выполнить следующие запросы.

Выбор всех записей таблицы и всех ее столбцов:

```
select * from departments
```

Выбор всех данных из заданной колонки таблицы:

```
select job_title  
      from jobs
```

Выбор всех данных из нескольких колонок таблицы:

```
select first_name  
      , last_name  
      , salary from employees
```

Выбор из таблицы данных, удовлетворяющих заданному условию:

```
select fio  
      from tb_elektrostal_2018  
      where typstr='проспект'
```

Выбор данных из таблицы, удовлетворяющих заданному условию и сортировка результатов:

```
select fio  
      from tb_elektrostal_2018  
      where typstr='проспект'  
      order by fio
```

Выбор данных из таблицы, удовлетворяющих заданному условию и сортировка результатов по убыванию:

```
select fio  
      from tb_elektrostal_2018  
      where typstr='проспект'  
      order by fio desc
```

Сортировка результата с использованием номера колонки:

```
select fio
, namestr
from tb_elektrostal_2018
where typstr='проспект'
order by 2
```

Сортировка по нескольким столбцам:

```
select fio
,namestr
from tb_elektrostal_2018
where typstr='проспект'
order by 2,1
```

Подсчет количества записей:

```
select count (*)
      from tb_elektrostal_2018
```

Запрос к БД StatckOverflow

```
select count (*)
      from users
```

Подсчет количества записей с использованием фильтра:

```
select count (*)
      from tb_elektrostal_2018
      where typstr='проспект'
```

Использование группировки:

```
select typstr
, count (*)
      from tb_elektrostal_2018
group by typstr
```

Использование группировки с сортировкой:

```
select typstr
      , count (*)
      from tb_elektrostal_2018
group by typstr
order by 2
```

Использование группировки с сортировкой по убыванию:

```
select typstr
      , count (*)
      from tb_elektrostal_2018
group by typstr
order by 2 desc
```

Использование соединений:

```
select a.first_name
      , a.last_name
      , b.department_name
      from employees a, departments b
      where a.department_id=b.department_id
```

Количество работников в отделах:

```
select b.department_name
      , count(*)
      from employees a, departments b
      where a.department_id=b.department_id
group by b.department_name
```

Составить следующие запросы

- Вывести список сотрудников, получающих заработную плату большую чем у непосредственного руководителя

- Вывести список сотрудников, получающих максимальную заработную плату в своем отделе
- Вывести список ID отделов, количество сотрудников в которых не превышает 3 человек
- Вывести список сотрудников, не имеющих назначенного руководителя, работающего в том же отделе
- Найти список ID отделов с максимальной суммарной зарплатой сотрудников
- Вывести идентификаторы пользователей, отвечавших в 2014 году и не задававших вопросов в 2015
- Найти пользователей, задавших более 50 вопросов

#### **Содержание отчета:**

1. Тема, цель лабораторной работы.
2. Примеры выполнения запросов к базе данных.
3. Примеры выполнения запросов
5. Выводы.

#### **Контрольные вопросы:**

1. Что является простым запросом к базе данных?
2. Для чего служит оператор SELECT, какие использует операнды?
3. Какую функцию выполняет в операторе SELECT команда WHERE?

Файл: Лабораторная работа 2  
Каталог: C:\Users\sss\Documents  
Шаблон: C:\Users\sss\AppData\Roaming\Microsoft\Шаблоны\Normal.dotm  
Заголовок:  
Содержание:  
Автор: sss  
Ключевые слова:  
Заметки:  
Дата создания: 13.09.2018 20:32:00  
Число сохранений: 14  
Дата сохранения: 15.09.2018 12:05:00  
Сохранил: sss  
Полное время правки: 93 мин.  
Дата печати: 15.09.2018 12:05:00  
При последней печати  
    страниц: 7  
    слов: 1 086 (прибл.)  
    знаков: 6 196 (прибл.)