

Федеральное государственное автономное образовательное учреждение высшего
образования «Национальный исследовательский университет ИТМО»
**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
“НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО”**

Факультет Программной Инженерии и Компьютерной Техники
Дисциплина: «Базы данных»

ОТЧЁТ

по лабораторной работе №2
Вариант №776

Выполнил:

Студент группы Р3108

Грищенко Андрей Викторович

Проверил:

Афанасьев Дмитрий Борисович

Оглавление

Оглавление.....	2
Текст задания.....	2
Реализация запросов на SQL.....	3
Запрос 1:.....	3
Запрос 2:.....	4
Запрос 3:.....	4
Запрос 4:.....	4
Запрос 5:.....	5
Запрос 6:.....	5
Запрос 7:.....	6
Вывод.....	7

Текст задания

По варианту, выданному преподавателем, составить и выполнить запросы к базе данных "Учебный процесс".

Команда для подключения к базе данных `uscheb`:

```
psql -h pg -d usheb
```

Составить запросы на языке SQL (пункты 1-7).

1. Сделать запрос для получения атрибутов из указанных таблиц, применив фильтры по указанным условиям:
Н_ТИПЫ_ВЕДОМОСТЕЙ, Н_ВЕДОМОСТИ.
Вывести атрибуты: Н_ТИПЫ_ВЕДОМОСТЕЙ.НАИМЕНОВАНИЕ, Н_ВЕДОМОСТИ.ИД.
Фильтры (AND):
а) Н_ТИПЫ_ВЕДОМОСТЕЙ.НАИМЕНОВАНИЕ = Ведомость.
б) Н_ВЕДОМОСТИ.ДАТА = 2010-06-18.
с) Н_ВЕДОМОСТИ.ДАТА > 1998-01-05.
Вид соединения: INNER JOIN.
2. Сделать запрос для получения атрибутов из указанных таблиц, применив фильтры по указанным условиям:
Таблицы: Н_ЛЮДИ, Н_ВЕДОМОСТИ, Н_СЕССИЯ.
Вывести атрибуты: Н_ЛЮДИ.ОТЧЕСТВО, Н_ВЕДОМОСТИ.ДАТА, Н_СЕССИЯ.ЧЛВК_ИД.
Фильтры (AND):
а) Н_ЛЮДИ.ОТЧЕСТВО = Сергеевич.

- б) Н_ВЕДОМОСТИ.ЧЛВК_ИД > 153285.
Вид соединения: INNER JOIN.
3. Составить запрос, который ответит на вопрос, есть ли среди студентов ФКТИУ люди без ИНН.
 4. Выдать различные отчества студентов и число людей с каждой из этих отчеств, ограничив список отчествами, встречающимися ровно 10 раз на заочной форме обучения.
Для реализации использовать соединение таблиц.
 5. Выведите таблицу со средним возрастом студентов во всех группах (Группа, Средний возраст), где средний возраст равен минимальному возрасту в группе 3100.
 6. Получить список студентов, отчисленных до первого сентября 2012 года с очной формы обучения. В результат включить:
номер группы;
номер, фамилию, имя и отчество студента;
номер пункта приказа;
Для реализации использовать подзапрос с EXISTS.
 7. Вывести список людей, не являющихся или не являвшихся студентами ФКТИУ (данные, о которых отсутствуют в таблице Н_УЧЕНИКИ). В запросе нельзя использовать DISTINCT.

Реализация запросов на SQL

Запрос 1:

“Сделать запрос для получения атрибутов из указанных таблиц, применив фильтры по указанным условиям:

Н_ТИПЫ_ВЕДОМОСТЕЙ, Н_ВЕДОМОСТИ.

Вывести атрибуты: Н_ТИПЫ_ВЕДОМОСТЕЙ.НАИМЕНОВАНИЕ, Н_ВЕДОМОСТИ.ИД.

Фильтры (AND):

а) Н_ТИПЫ_ВЕДОМОСТЕЙ.НАИМЕНОВАНИЕ = Ведомость.

б) Н_ВЕДОМОСТИ.ДАТА = 2010-06-18.

с) Н_ВЕДОМОСТИ.ДАТА > 1998-01-05.

Вид соединения: INNER JOIN.”

```
SELECT НАИМЕНОВАНИЕ,  
       Н_ВЕДОМОСТИ.ИД  
FROM Н_ВЕДОМОСТИ  
      JOIN Н_ТИПЫ_ВЕДОМОСТЕЙ  
      ON Н_ВЕДОМОСТИ.ТВ_ИД = Н_ТИПЫ_ВЕДОМОСТЕЙ.ИД  
WHERE НАИМЕНОВАНИЕ = 'Ведомость'  
      AND Н_ВЕДОМОСТИ.ДАТА = '2010-06-18'  
      AND Н_ВЕДОМОСТИ.ДАТА > '1998-01-05';
```

Запрос 2:

“Сделать запрос для получения атрибутов из указанных таблиц, применив фильтры по указанным условиям:

Таблицы: Н_ЛЮДИ, Н_ВЕДОМОСТИ, Н_СЕССИЯ.

Вывести атрибуты: Н_ЛЮДИ.ОТЧЕСТВО, Н_ВЕДОМОСТИ.ДАТА,
Н_СЕССИЯ.ЧЛВК_ИД.

Фильтры (AND):

а) Н_ЛЮДИ.ОТЧЕСТВО = Сергеевич.

б) Н_ВЕДОМОСТИ.ЧЛВК_ИД > 153285.

Вид соединения: INNER JOIN.”

```
SELECT ОТЧЕСТВО,  
       Н_ВЕДОМОСТИ.ДАТА,  
       Н_СЕССИЯ.ЧЛВК_ИД  
FROM Н_ЛЮДИ  
     JOIN Н_СЕССИЯ  
         ON Н_СЕССИЯ.ЧЛВК_ИД = Н_ЛЮДИ.ИД  
     JOIN Н_ВЕДОМОСТИ  
         ON Н_ВЕДОМОСТИ.ЧЛВК_ИД = Н_ЛЮДИ.ИД  
WHERE Н_ВЕДОМОСТИ.ЧЛВК_ИД > 153285  
      AND Н_ЛЮДИ.ОТЧЕСТВО = 'Сергеевич';
```

Запрос 3:

“Составить запрос, который ответит на вопрос, есть ли среди студентов ФКТИУ люди без ИНН.”

```
SELECT EXISTS(  
    SELECT 1  
    FROM Н_ЛЮДИ  
         JOIN Н_УЧЕНИКИ  
             ON Н_УЧЕНИКИ.ЧЛВК_ИД = Н_ЛЮДИ.ИД  
         JOIN Н_ПЛАНЫ  
             ON Н_ПЛАНЫ.ИД = Н_УЧЕНИКИ.ПЛАН_ИД  
         JOIN Н_ОТДЕЛЫ  
             ON Н_ОТДЕЛЫ.ИД = Н_ПЛАНЫ.ОТД_ИД  
    WHERE Н_ОТДЕЛЫ.КОРОТКОЕ_ИМЯ = 'КТИУ'  
          AND Н_ЛЮДИ.ИНН NOTNULL  
);
```

Запрос 4:

“Выдать различные отчества студентов и число людей с каждой из этих отчеств, ограничив список отчествами, встречающимися ровно 10 раз на заочной форме обучения.

Для реализации использовать соединение таблиц.”

```
SELECT ОТЧЕСТВО,  
       COUNT(ОТЧЕСТВО)  
FROM Н_ЛЮДИ  
     JOIN Н_УЧЕНИКИ  
         ON Н_ЛЮДИ.ИД = Н_УЧЕНИКИ.ЧЛВК_ИД  
     JOIN Н_ПЛАНЫ  
         ON Н_УЧЕНИКИ.ПЛАН_ИД = Н_ПЛАНЫ.ИД  
     JOIN Н_ФОРМЫ_ОБУЧЕНИЯ  
         ON Н_ФОРМЫ_ОБУЧЕНИЯ.ИД = Н_ПЛАНЫ.ФО_ИД  
WHERE НАИМЕНОВАНИЕ = 'Заочная'  
     AND ОТЧЕСТВО IN (  
         SELECT ОТЧЕСТВО  
         FROM Н_ЛЮДИ  
         RIGHT JOIN Н_СЕССИЯ  
             ON Н_ЛЮДИ.ИД = Н_СЕССИЯ.ЧЛВК_ИД  
         WHERE Н_ЛЮДИ.ИД IS NOT NULL  
     )  
GROUP BY ОТЧЕСТВО  
HAVING COUNT(ОТЧЕСТВО) = 10;
```

Запрос 5:

“Выведите таблицу со средним возрастом студентов во всех группах (Группа, Средний возраст), где средний возраст равен минимальному возрасту в группе 3100.”

```
SELECT Н_УЧЕНИКИ.ГРУППА,  
       AVG(AGE(NOW(), Н_ЛЮДИ.ДАТА_РОЖДЕНИЯ))  
FROM Н_ЛЮДИ  
     JOIN Н_УЧЕНИКИ  
         ON Н_ЛЮДИ.ИД = Н_УЧЕНИКИ.ЧЛВК_ИД  
GROUP BY Н_УЧЕНИКИ.ГРУППА  
HAVING AVG(AGE(NOW(), Н_ЛЮДИ.ДАТА_РОЖДЕНИЯ)) =  
(  
    SELECT MIN(AGE(NOW(), Н_ЛЮДИ.ДАТА_РОЖДЕНИЯ))  
    FROM Н_ЛЮДИ  
    JOIN Н_УЧЕНИКИ  
        ON Н_ЛЮДИ.ИД = Н_УЧЕНИКИ.ЧЛВК_ИД  
    WHERE Н_УЧЕНИКИ.ГРУППА = '3100'  
        AND Н_ЛЮДИ.ДАТА_РОЖДЕНИЯ <= NOW()  
);
```

Запрос 6:

“Получить список студентов, отчисленных до первого сентября 2012 года с очной формы обучения. В результат включить:

номер группы;

номер, фамилию, имя и отчество студента;

номер пункта приказа;

Для реализации использовать подзапрос с EXISTS.”

```
SELECT ГРУППА,  
       ВНЕШ_УЧЕНИКИ.ИД,  
       ФАМИЛИЯ,  
       ИМЯ,  
       ОТЧЕСТВО,  
       П_ПРКОК_ИД  
FROM Н_ЛЮДИ  
     JOIN Н_УЧЕНИКИ AS ВНЕШ_УЧЕНИКИ  
         ON ВНЕШ_УЧЕНИКИ.ЧЛВК_ИД = Н_ЛЮДИ.ИД  
     JOIN Н_ПЛАНЫ  
         ON ВНЕШ_УЧЕНИКИ.ПЛАН_ИД = Н_ПЛАНЫ.ИД  
     JOIN Н_ФОРМЫ_ОБУЧЕНИЯ  
         ON Н_ПЛАНЫ.ФО_ИД = Н_ФОРМЫ_ОБУЧЕНИЯ.ИД  
         AND НАИМЕНОВАНИЕ = 'Заочная'  
WHERE EXISTS  
(  
    SELECT *  
    FROM Н_УЧЕНИКИ  
    WHERE ПРИЗНАК = 'отчисл'  
          AND СОСТОЯНИЕ = 'утвержден'  
          AND ИД = ВНЕШ_УЧЕНИКИ.ИД  
          AND DATE(КОНЕЦ) < '2012-09-01'  
);
```

Запрос 7:

“Вывести список людей, не являющихся или не являвшихся студентами ФКТИУ (данные, о которых отсутствуют в таблице Н_УЧЕНИКИ). В запросе нельзя использовать DISTINCT.”

```
SELECT Н_ЛЮДИ.ИД,  
       ФАМИЛИЯ,  
       ИМЯ,  
       ОТЧЕСТВО  
FROM Н_ЛЮДИ  
WHERE NOT EXISTS (  
    SELECT *  
    FROM Н_УЧЕНИКИ  
         JOIN Н_ПЛАНЫ  
            ON Н_ПЛАНЫ.ИД = Н_УЧЕНИКИ.ПЛАН_ИД  
         JOIN Н_ОТДЕЛЫ  
            ON Н_ОТДЕЛЫ.ИД = Н_ПЛАНЫ.ОТД_ИД  
            AND Н_ОТДЕЛЫ.КОРОТКОЕ_ИМЯ = 'КТиУ'  
    WHERE  
        Н_УЧЕНИКИ.ЧЛВК_ИД = Н_ЛЮДИ.ИД  
);
```

Вывод

В ходе проделанной лабораторной работы, я изучил DML SQL, узнал что он предназначен для работы с данными, хранящимися в базе данных, познакомился с основными функциями языка SQL, а также написал свои собственные запросы.