

Федеральное государственное автономное образовательное учреждение
высшего образования «Национальный исследовательский университет
ИТМО»

Факультет программной инженерии и компьютерной техники
Направление подготовки: 09.03.04 – Программная инженерия, Системное и
прикладное программное обеспечение

Дисциплина: Базы данных

Лабораторная работа №2

Вариант №312110

Выполнил:

Карнажицкий Максим Романович

Группа: Р3111

Проверила:

Харитоновна Анастасия Евгеньевна

г. Санкт-Петербург, 2025

Оглавление

| | |
|---------------------------------|---|
| Текст задания..... | 3 |
| Реализация запросов на SQL..... | 5 |
| Задание 1..... | 5 |
| Задание 2..... | 5 |
| Задание 3..... | 5 |
| Задание 4..... | 6 |
| Задание 5..... | 6 |
| Задание 6..... | 7 |
| Задание 7..... | 8 |
| Вывод..... | 9 |

Текст задания

1. Сделать запрос для получения атрибутов из указанных таблиц, применив фильтры по указанным условиям:
Таблицы: Н_ЛЮДИ, Н_СЕССИЯ.
Вывести атрибуты: Н_ЛЮДИ.ФАМИЛИЯ, Н_СЕССИЯ.УЧГОД.
Фильтры (AND):
 - а) Н_ЛЮДИ.ОТЧЕСТВО > Владимирович.
 - б) Н_СЕССИЯ.ДАТА > 2002-01-04.
 - с) Н_СЕССИЯ.ДАТА < 2002-01-04.Вид соединения: LEFT JOIN.
2. Сделать запрос для получения атрибутов из указанных таблиц, применив фильтры по указанным условиям:
Таблицы: Н_ЛЮДИ, Н_ОБУЧЕНИЯ, Н_УЧЕНИКИ.
Вывести атрибуты: Н_ЛЮДИ.ОТЧЕСТВО, Н_ОБУЧЕНИЯ.ЧЛВК_ИД, Н_УЧЕНИКИ.ГРУППА.
Фильтры: (AND)
 - а) Н_ЛЮДИ.ИМЯ > Александр.
 - б) Н_ОБУЧЕНИЯ.ЧЛВК_ИД < 163484.Вид соединения: RIGHT JOIN.
3. Вывести число фамилий и имен без учета повторений.
При составлении запроса нельзя использовать DISTINCT.
4. Выдать различные имена студентов и число людей с каждой из этих имен, ограничив список именами, встречающимися более 50 раз на ФКТИУ.
Для реализации использовать подзапрос.
5. Выведите таблицу со средними оценками студентов группы 4100 (Номер, ФИО, Ср_оценка), у которых средняя оценка не больше максимальной оценк(е|и) в группе 1101.
6. Получить список студентов, зачисленных после первого сентября 2012 года на первый курс заочной формы обучения (специальность: 230101).
В результат включить:
номер группы;
номер, фамилию, имя и отчество студента;
номер и состояние пункта приказа;
Для реализации использовать подзапрос с EXISTS.

7. Вывести список студентов, имеющих одинаковые имена, но не совпадающие ид.

Реализация запросов на SQL

Задание 1

```
SELECT Н_ЛЮДИ.ФАМИЛИЯ, Н_СЕССИЯ.УЧГОД FROM Н_СЕССИЯ
LEFT JOIN Н_ЛЮДИ ON Н_СЕССИЯ.ЧЛВК_ИД = Н_ЛЮДИ.ИД
WHERE Н_ЛЮДИ.ОТЧЕСТВО > 'Владимирович'
AND Н_СЕССИЯ.ДАТА > '2002-01-04'
AND Н_СЕССИЯ.ДАТА < '2002-01-04';
```

Этот запрос не даст результатов, потому что условие невозможно. Последнее условие я убрал, и получил корректный запрос.

```
SELECT Н_ЛЮДИ.ФАМИЛИЯ, Н_СЕССИЯ.УЧГОД FROM Н_СЕССИЯ
LEFT JOIN Н_ЛЮДИ ON Н_СЕССИЯ.ЧЛВК_ИД = Н_ЛЮДИ.ИД
WHERE Н_ЛЮДИ.ОТЧЕСТВО > 'Владимирович'
AND Н_СЕССИЯ.ДАТА > '2002-01-04';
```

Задание 2

```
SELECT Н_ЛЮДИ.ОТЧЕСТВО, Н_ОБУЧЕНИЯ.ЧЛВК_ИД, Н_УЧЕНИКИ.ГРУППА FROM
Н_ОБУЧЕНИЯ
RIGHT JOIN Н_ЛЮДИ ON Н_ОБУЧЕНИЯ.ЧЛВК_ИД = Н_ЛЮДИ.ИД
RIGHT JOIN Н_УЧЕНИКИ ON Н_ЛЮДИ.ИД = Н_УЧЕНИКИ.ЧЛВК_ИД
WHERE Н_ЛЮДИ.ИМЯ > 'Александр'
AND Н_ОБУЧЕНИЯ.ЧЛВК_ИД < 163484;
```

Задание 3

```
WITH unique_fam_amount AS (
    SELECT COUNT(ФАМИЛИЯ) AS fcount FROM Н_ЛЮДИ
    GROUP BY ФАМИЛИЯ
    HAVING COUNT(ФАМИЛИЯ) = 1
), unique_name_amount AS (
    SELECT COUNT(ИМЯ) AS ncount FROM Н_ЛЮДИ
    GROUP BY ИМЯ
    HAVING COUNT(ИМЯ) = 1
)
SELECT
    (SELECT COUNT(*) FROM unique_fam_amount) AS колво_уник_фамилий,
    (SELECT COUNT(*) FROM unique_name_amount) AS колво_уник_имен;
```

Задание 4

```
WITH names_50_ktu AS (  
    SELECT Н_ЛЮДИ.ИМЯ as name FROM Н_УЧЕНИКИ  
    JOIN Н_ЛЮДИ ON Н_УЧЕНИКИ.ЧЛВК_ИД = Н_ЛЮДИ.ИД  
    JOIN Н_ГРУППЫ_ПЛАНОВ ON Н_УЧЕНИКИ.ГРУППА = Н_ГРУППЫ_ПЛАНОВ.ГРУППА  
    JOIN Н_ПЛАНЫ ON Н_ГРУППЫ_ПЛАНОВ.ПЛАН_ИД = Н_ПЛАНЫ.ИД  
    JOIN Н_ОТДЕЛЫ ON Н_ПЛАНЫ.ОТД_ИД = Н_ОТДЕЛЫ.ИД  
    WHERE Н_ОТДЕЛЫ.КОРОТКОЕ_ИМЯ = 'КТиУ' AND ИМЯ <> '.' AND ИМЯ <> '  
    GROUP BY name  
    HAVING COUNT(DISTINCT Н_ЛЮДИ.ИД) > 50  
)  
  
SELECT Н_ЛЮДИ.ИМЯ, COUNT(*) AS total_count FROM Н_УЧЕНИКИ  
JOIN Н_ЛЮДИ ON Н_УЧЕНИКИ.ЧЛВК_ИД = Н_ЛЮДИ.ИД  
WHERE Н_ЛЮДИ.ИМЯ IN (SELECT name FROM names_50_ktu)  
GROUP BY Н_ЛЮДИ.ИМЯ  
ORDER BY total_count DESC;
```

Задание 5

```
WITH max_mark_1101 AS (  
    SELECT MAX(Н_ВЕДОМОСТИ.ОЦЕНКА::Integer) as max_mark FROM  
    Н_ВЕДОМОСТИ  
    JOIN Н_УЧЕНИКИ ON Н_ВЕДОМОСТИ.ЧЛВК_ИД = Н_УЧЕНИКИ.ЧЛВК_ИД  
    WHERE Н_УЧЕНИКИ.ГРУППА = '1101'  
    AND Н_ВЕДОМОСТИ.ОЦЕНКА IN ('5', '4', '3', '2')  
)  
  
SELECT Н_УЧЕНИКИ.ИД, Н_ЛЮДИ.ИМЯ, Н_ЛЮДИ.ФАМИЛИЯ, Н_ЛЮДИ.ОТЧЕСТВО,  
AVG(Н_ВЕДОМОСТИ.ОЦЕНКА::Integer) as avg_mark FROM Н_УЧЕНИКИ  
JOIN Н_ВЕДОМОСТИ ON Н_УЧЕНИКИ.ЧЛВК_ИД = Н_ВЕДОМОСТИ.ЧЛВК_ИД  
JOIN Н_ЛЮДИ ON Н_УЧЕНИКИ.ЧЛВК_ИД = Н_ЛЮДИ.ИД  
WHERE Н_УЧЕНИКИ.ГРУППА = '4100'  
AND Н_ВЕДОМОСТИ.ОЦЕНКА IN ('5', '4', '3', '2')  
AND Н_ВЕДОМОСТИ.ОЦЕНКА::Integer <= (SELECT max_mark FROM  
max_mark_1101)  
GROUP BY Н_УЧЕНИКИ.ИД, Н_ЛЮДИ.ИМЯ, Н_ЛЮДИ.ФАМИЛИЯ, Н_ЛЮДИ.ОТЧЕСТВО  
ORDER BY avg_mark DESC;
```

Задание 6

```
SELECT
    Н_УЧЕНИКИ.ГРУППА,
    Н_УЧЕНИКИ.ИД,
    Н_ЛЮДИ.ИМЯ,
    Н_ЛЮДИ.ФАМИЛИЯ,
    Н_ЛЮДИ.ОТЧЕСТВО,
    Н_УЧЕНИКИ.П_ПРКОК_ИД AS НОМЕР_ПРИКАЗА,
    Н_УЧЕНИКИ.СОСТОЯНИЕ AS СОСТОЯНИЕ_ПРИКАЗА
FROM Н_УЧЕНИКИ
JOIN Н_ЛЮДИ ON Н_УЧЕНИКИ.ЧЛВК_ИД = Н_ЛЮДИ.ИД
WHERE Н_УЧЕНИКИ.НАЧАЛО > '2012-09-01'
AND EXISTS (
    SELECT 1 FROM Н_ПЛАНЫ
    INNER JOIN Н_ФОРМЫ_ОБУЧЕНИЯ ON Н_ПЛАНЫ.ФО_ИД = Н_ФОРМЫ_ОБУЧЕНИЯ.ИД
    INNER JOIN Н_НАПРАВЛЕНИЯ_СПЕЦИАЛ ON Н_ПЛАНЫ.НАПС_ИД =
Н_НАПРАВЛЕНИЯ_СПЕЦИАЛ.ИД
    INNER JOIN Н_НАПР_СПЕЦ ON Н_НАПРАВЛЕНИЯ_СПЕЦИАЛ.НС_ИД =
Н_НАПР_СПЕЦ.ИД
    WHERE Н_ПЛАНЫ.ИД = Н_УЧЕНИКИ.ПЛАН_ИД
    AND Н_ФОРМЫ_ОБУЧЕНИЯ.НАИМЕНОВАНИЕ LIKE '%заочная%'
    AND Н_НАПР_СПЕЦ.КОД_НАПРСПЕЦ = '230101'
);
```

Запрос не дал результатов, потому что записей старше 2011 года в таблице нет. Помимо этого, намек на заочную учебу нет (только в 2006 есть очно-заочная форма). Ниже обновленный запрос, который дает результаты.

```
--      2012-09-01  ->  2006-09-01
--      Заочная    ->  Очно-заочная

SELECT DISTINCT ON (Н_ЛЮДИ.ИД)
    Н_УЧЕНИКИ.ГРУППА,
    Н_УЧЕНИКИ.ИД,
    Н_ЛЮДИ.ИМЯ,
    Н_ЛЮДИ.ФАМИЛИЯ,
    Н_ЛЮДИ.ОТЧЕСТВО,
    Н_УЧЕНИКИ.П_ПРКОК_ИД AS НОМЕР_ПРИКАЗА,
    Н_УЧЕНИКИ.СОСТОЯНИЕ AS СОСТОЯНИЕ_ПРИКАЗА
FROM Н_УЧЕНИКИ
JOIN Н_ЛЮДИ ON Н_УЧЕНИКИ.ЧЛВК_ИД = Н_ЛЮДИ.ИД
WHERE Н_УЧЕНИКИ.НАЧАЛО > '2006-09-01'
```



```

AND EXISTS (
    SELECT 1 FROM Н_ПЛАНЫ
    INNER JOIN Н_ФОРМЫ_ОБУЧЕНИЯ ON Н_ПЛАНЫ.ФО_ИД = Н_ФОРМЫ_ОБУЧЕНИЯ.ИД
    INNER JOIN Н_НАПРАВЛЕНИЯ_СПЕЦИАЛ ON Н_ПЛАНЫ.НАПС_ИД =
Н_НАПРАВЛЕНИЯ_СПЕЦИАЛ.ИД
    INNER JOIN Н_НАПР_СПЕЦ ON Н_НАПРАВЛЕНИЯ_СПЕЦИАЛ.НС_ИД =
Н_НАПР_СПЕЦ.ИД
    WHERE Н_ПЛАНЫ.ИД = Н_УЧЕНИКИ.ПЛАН_ИД
    AND Н_ФОРМЫ_ОБУЧЕНИЯ.НАИМЕНОВАНИЕ LIKE '%заочная%'
    AND Н_НАПР_СПЕЦ.КОД_НАПРСПЕЦ = '230101'
);

```

Задание 7

```

WITH repeat_names AS (
    SELECT Н_ЛЮДИ.ИМЯ AS name FROM Н_УЧЕНИКИ
    INNER JOIN Н_ЛЮДИ ON Н_УЧЕНИКИ.ЧЛВК_ИД = Н_ЛЮДИ.ИД
    GROUP BY name
    HAVING COUNT(DISTINCT Н_ЛЮДИ.ИД) > 1
)
SELECT DISTINCT Н_ЛЮДИ.ИД, Н_ЛЮДИ.ИМЯ, Н_ЛЮДИ.ФАМИЛИЯ,
Н_ЛЮДИ.ОТЧЕСТВО AS name FROM Н_УЧЕНИКИ
INNER JOIN Н_ЛЮДИ ON Н_УЧЕНИКИ.ЧЛВК_ИД = Н_ЛЮДИ.ИД
WHERE Н_ЛЮДИ.ИМЯ IN (SELECT name FROM repeat_names) AND Н_ЛЮДИ.ИМЯ
<> '.'
ORDER BY Н_ЛЮДИ.ИМЯ ASC;

```

Пришлось указывать явно уникальность Н_ЛЮДИ.ИД, так как один человек может несколько раз числиться в студентах под разными номерами!

Вывод

В ходе лабораторной работы я научился быстро переключать раскладку на клавиатуре и писать сложные SQL-запросы, наконец разобрался с JOIN'ами. Познакомился с подзапросами и их вариациями.