

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ  
ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ  
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«Санкт-Петербургский национальный исследовательский университет  
информационных технологий, механики и оптики»

ФАКУЛЬТЕТ ПРОГРАММНОЙ ИНЖЕНЕРИИ И  
КОМПЬЮТЕРНОЙ ТЕХНИКИ

## **ЛАБОРАТОРНАЯ РАБОТА № 2**

по дисциплине  
«Базы Данных»

Вариант № 312515

**Выполнил:**

Студент группы Р3125

Шадрухин Александр Сергеевич

**Преподаватель:**

Егошин Алексей Васильевич

г. Санкт-Петербург, 2023

## Задание.

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

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.ЧЛВК\_ИД.

Фильтры (AND):

а) Н\_ЛЮДИ.ОТЧЕСТВО = Георгиевич.

б) Н\_СЕССИЯ.ЧЛВК\_ИД < 100622.

Вид соединения: RIGHT JOIN.

```
SELECT Н_ЛЮДИ.ФАМИЛИЯ, Н_СЕССИЯ.ЧЛВК_ИД
FROM Н_ЛЮДИ
      RIGHT JOIN Н_СЕССИЯ ON Н_ЛЮДИ.ИД = Н_СЕССИЯ.ЧЛВК_ИД
WHERE Н_ЛЮДИ.ОТЧЕСТВО = 'Георгиевич' AND Н_СЕССИЯ.ЧЛВК_ИД <
'100622';
```

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.ИД, Н\_ОБУЧЕНИЯ.НЗК, Н\_УЧЕНИКИ.ГРУППА.

Фильтры: (AND)

а) Н\_ЛЮДИ.ИД = 142095.

б) Н\_ОБУЧЕНИЯ.ЧЛВК\_ИД < 163276.

с) Н\_УЧЕНИКИ.ГРУППА > 1101.

Вид соединения: INNER JOIN.

```
SELECT Н_ЛЮДИ.ИД, Н_ОБУЧЕНИЯ.НЗК, Н_УЧЕНИКИ.ГРУППА
FROM Н_ЛЮДИ
      INNER JOIN Н_ОБУЧЕНИЯ ON Н_ЛЮДИ.ИД = Н_ОБУЧЕНИЯ.ЧЛВК_ИД
      INNER JOIN Н_УЧЕНИКИ ON Н_ЛЮДИ.ИД = Н_УЧЕНИКИ.ЧЛВК_ИД
WHERE Н_ЛЮДИ.ИД = 142095 AND Н_ОБУЧЕНИЯ.ЧЛВК_ИД < '163276' AND
Н_УЧЕНИКИ.ГРУППА > '1101';
```

3) Составить запрос, который ответит на вопрос, есть ли среди студентов группы 3102 те, кто не имеет отчества.

```
SELECT COUNT(*) <> 0 AS "Есть ли в группе люди без отчества?"
FROM Н_ЛЮДИ
      RIGHT JOIN Н_УЧЕНИКИ ON Н_ЛЮДИ.ИД = Н_УЧЕНИКИ.ЧЛВК_ИД
WHERE Н_УЧЕНИКИ.ГРУППА = '3102' AND Н_ЛЮДИ.ОТЧЕСТВО IS NULL;
```

4) Найти группы, в которых в 2011 году было менее 5 обучающихся студентов на кафедре вычислительной техники.

Для реализации использовать подзапрос.

```

SELECT ГРУППЫ_ВТ.ГРУППА, ГРУППЫ_ВТ.КОЛИЧЕСТВО
FROM (SELECT Н_УЧЕНИКИ.ГРУППА, count(Н_УЧЕНИКИ.ИД) AS КОЛИЧЕСТВО
      FROM Н_УЧЕНИКИ
      JOIN Н_ПЛАНЫ ON Н_УЧЕНИКИ.ПЛАН_ИД = Н_ПЛАНЫ.ИД
AND Н_ПЛАНЫ.УЧЕБНЫЙ_ГОД = '2010/2011' AND Н_ПЛАНЫ.УЧЕБНЫЙ_ГОД =
'2011/2012'
      JOIN Н_ОТДЕЛЫ ON Н_ОТДЕЛЫ.ИД = Н_ПЛАНЫ.ОТД_ИД AND
Н_ОТДЕЛЫ.КОРОТКОЕ_ИМЯ = 'ВТ'
      GROUP BY Н_УЧЕНИКИ.ГРУППА) AS ГРУППЫ_ВТ
WHERE ГРУППЫ_ВТ.КОЛИЧЕСТВО < 5;

```

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

```

SELECT * FROM
  (SELECT ГРУППА,
    floor(AVG(date_part('year', age (now(),
ДАТА_РОЖДЕНИЯ)))) AS Возраст
  FROM Н_УЧЕНИКИ
    JOIN Н_ЛЮДИ ON Н_УЧЕНИКИ.ЧЛВК_ИД = Н_ЛЮДИ.ИД
  GROUP BY ГРУППА) AS res
WHERE Возраст=
  (SELECT floor(AVG(date_part('year', age (now(),
ДАТА_РОЖДЕНИЯ))))
  FROM Н_УЧЕНИКИ
    JOIN Н_ЛЮДИ ON Н_УЧЕНИКИ.ЧЛВК_ИД = Н_ЛЮДИ.ИД
  WHERE ГРУППА='3100'
  GROUP BY ГРУППА)
Order BY Возраст DESC;

```

6) Получить список студентов, зачисленных ровно первого сентября 2012 года на первый курс очной или заочной формы обучения (специальность: Программная инженерия). В результат включить:  
 номер группы;  
 номер, фамилию, имя и отчество студента;  
 номер и состояние пункта приказа;  
 Для реализации использовать подзапрос с IN.

```

SELECT ГРУППА, ФАМИЛИЯ, ИМЯ, ОТЧЕСТВО, П_ПРКОК_ИД
FROM (Н_УЧЕНИКИ students
      JOIN Н_ЛЮДИ people ON students.ЧЛВК_ИД=people.ИД) AS res
WHERE res.НАЧАЛО = '2012-09-01'
AND res.ГРУППА IN (
  SELECT gr.ГРУППА
  FROM Н_ГРУППЫ_ПЛАНОВ gr
  JOIN Н_ПЛАНЫ plan ON gr.ПЛАН_ИД = plan.ИД
  JOIN Н_НАПРАВЛЕНИЯ_СПЕЦИАЛ sp ON sp.ИД = plan.НАПС_ИД
  JOIN Н_НАПР_СПЕЦ ns ON ns.ИД = sp.НС_ИД
  WHERE ns.НАИМЕНОВАНИЕ = 'Программная инженерия'
  AND (plan.ФО_ИД = 1 OR plan.ФО_ИД = 3));

```

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

```
SELECT ФАМИЛИЯ, ИМЯ,  
       ИД FROM Н_ЛЮДИ person1  
WHERE EXISTS (SELECT 1  
              FROM Н_ЛЮДИ person2  
              WHERE person1.ФАМИЛИЯ=person2.ФАМИЛИЯ  
                 AND person1.ИД<>person2.ИД) ORDER BY ФАМИЛИЯ;
```

## Вывод

Во время выполнения лабораторной работы я овладел навыками работы с запросами в SQL, включая использование подзапросов, условий, преобразования и сортировкой данных, а также с различными функциями.