Министерство образования и науки РФ

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 № 3**

по дисциплине

‘Информационные системы и базы данных’

Вариант №1955

*Выполнил:*

Студент группы P33111

Павлов Александр Сергеевич

*Преподаватель:*

Харитонова А.Е.



Санкт-Петербург, 2023

# Описание задания

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.ИД, Н\_ВЕДОМОСТИ.ЧЛВК\_ИД.  
   Фильтры (AND):  
   a) Н\_ТИПЫ\_ВЕДОМОСТЕЙ.НАИМЕНОВАНИЕ < Экзаменационный лист.  
   b) Н\_ВЕДОМОСТИ.ЧЛВК\_ИД > 105590.  
   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.ИД, Н\_ОБУЧЕНИЯ.НЗК, Н\_УЧЕНИКИ.НАЧАЛО.  
   Фильтры: (AND)  
   a) Н\_ЛЮДИ.ФАМИЛИЯ > Афанасьев.  
   b) Н\_ОБУЧЕНИЯ.ЧЛВК\_ИД < 163484.  
   c) Н\_УЧЕНИКИ.ИД > 150308.  
   Вид соединения: RIGHT JOIN.
3. Составить запрос, который ответит на вопрос, есть ли среди студентов группы 3102 люди без ИНН.
4. В таблице Н\_ГРУППЫ\_ПЛАНОВ найти номера планов, по которым обучается (обучалось) менее 2 групп на очной форме обучения.  
   Для реализации использовать подзапрос.
5. Выведите таблицу со средними оценками студентов группы 4100 (Номер, ФИО, Ср\_оценка), у которых средняя оценка равна максимальной оценк(е|и) в группе 1100.
6. Получить список студентов, отчисленных после первого сентября 2012 года с заочной формы обучения (специальность: Программная инженерия). В результат включить:  
   номер группы;  
   номер, фамилию, имя и отчество студента;  
   номер пункта приказа;  
   Для реализации использовать подзапрос с IN.
7. Вывести список студентов, имеющих одинаковые фамилии, но не совпадающие даты рождения.

# Запросы

1. SELECT Н\_ТИПЫ\_ВЕДОМОСТЕЙ.ИД, Н\_ВЕДОМОСТИ.ЧЛВК\_ИД

FROM Н\_ТИПЫ\_ВЕДОМОСТЕЙ

LEFT JOIN Н\_ВЕДОМОСТИ ON Н\_ТИПЫ\_ВЕДОМОСТЕЙ.ИД = Н\_ВЕДОМОСТИ.ТВ\_ИД

WHERE Н\_ТИПЫ\_ВЕДОМОСТЕЙ.НАИМЕНОВАНИЕ < 'Экзаменационный лист'

AND Н\_ВЕДОМОСТИ.ЧЛВК\_ИД > 105590;

2. SELECT Н\_ЛЮДИ.ИД, Н\_ОБУЧЕНИЯ.НЗК, Н\_УЧЕНИКИ.НАЧАЛО

FROM Н\_ЛЮДИ

RIGHT JOIN Н\_ОБУЧЕНИЯ ON Н\_ЛЮДИ.ИД = Н\_ОБУЧЕНИЯ.ЧЛВК\_ИД

RIGHT JOIN Н\_УЧЕНИКИ ON Н\_ЛЮДИ.ИД = Н\_УЧЕНИКИ.ЧЛВК\_ИД

WHERE Н\_ЛЮДИ.ФАМИЛИЯ > 'Афанасьев'

AND Н\_ОБУЧЕНИЯ.ЧЛВК\_ИД < 163484

AND Н\_УЧЕНИКИ.ИД > 150308;

3. SELECT EXISTS(

SELECT \*

FROM Н\_УЧЕНИКИ

JOIN Н\_ЛЮДИ ON Н\_УЧЕНИКИ.ИД = Н\_ЛЮДИ.ИД

WHERE Н\_УЧЕНИКИ.ГРУППА = '3102'

AND Н\_ЛЮДИ.ИНН is NULL

);

4. SELECT Н\_ГРУППЫ\_ПЛАНОВ.ПЛАН\_ИД

FROM Н\_ГРУППЫ\_ПЛАНОВ

JOIN (

SELECT Н\_ПЛАНЫ.ИД

FROM Н\_ПЛАНЫ

JOIN Н\_ФОРМЫ\_ОБУЧЕНИЯ ON Н\_ПЛАНЫ.ФО\_ИД = Н\_ФОРМЫ\_ОБУЧЕНИЯ.ИД

WHERE Н\_ФОРМЫ\_ОБУЧЕНИЯ.НАИМЕНОВАНИЕ = 'Очная'

) AS Н\_ОЧНЫЕ\_ПЛАНЫ\_ОБУЧЕНИЯ ON Н\_ГРУППЫ\_ПЛАНОВ.ПЛАН\_ИД = Н\_ОЧНЫЕ\_ПЛАНЫ\_ОБУЧЕНИЯ.ИД

GROUP BY Н\_ГРУППЫ\_ПЛАНОВ.ПЛАН\_ИД

HAVING COUNT(Н\_ГРУППЫ\_ПЛАНОВ.ПЛАН\_ИД) < 2;

5. SELECT Н\_ВЕДОМОСТИ.ЧЛВК\_ИД, Н\_ЛЮДИ.ФАМИЛИЯ, Н\_ЛЮДИ.ИМЯ, Н\_ЛЮДИ.ОТЧЕСТВО, ROUND(AVG(Н\_ВЕДОМОСТИ.ОЦЕНКА::int), 2) AS "СРЕДНЯЯ\_ОЦЕНКА"

FROM Н\_ВЕДОМОСТИ

JOIN Н\_УЧЕНИКИ 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 ~ '^[2-5]$'

GROUP BY Н\_ВЕДОМОСТИ.ЧЛВК\_ИД, Н\_ЛЮДИ.ФАМИЛИЯ, Н\_ЛЮДИ.ИМЯ, Н\_ЛЮДИ.ОТЧЕСТВО

HAVING AVG(Н\_ВЕДОМОСТИ.ОЦЕНКА::int) = (

SELECT MAX(Н\_ВЕДОМОСТИ.ОЦЕНКА::int)

FROM Н\_ВЕДОМОСТИ

JOIN Н\_УЧЕНИКИ ON Н\_УЧЕНИКИ.ИД = Н\_ВЕДОМОСТИ.ЧЛВК\_ИД

WHERE Н\_УЧЕНИКИ.ГРУППА = '1100'

AND Н\_ВЕДОМОСТИ.ОЦЕНКА ~ '^[2-5]$'

);

6. SELECT Н\_УЧЕНИКИ.ГРУППА, Н\_ЛЮДИ.ИД, Н\_ЛЮДИ.ФАМИЛИЯ,

Н\_ЛЮДИ.ИМЯ, Н\_ЛЮДИ.ОТЧЕСТВО, Н\_УЧЕНИКИ.П\_ПРКОК\_ИД

FROM Н\_УЧЕНИКИ

JOIN Н\_ЛЮДИ ON Н\_ЛЮДИ.ИД = Н\_УЧЕНИКИ.ЧЛВК\_ИД

JOIN Н\_ПЛАНЫ ON Н\_ПЛАНЫ.ИД = Н\_УЧЕНИКИ.ПЛАН\_ИД

JOIN Н\_ФОРМЫ\_ОБУЧЕНИЯ ON Н\_ПЛАНЫ.ФО\_ИД = Н\_ФОРМЫ\_ОБУЧЕНИЯ.ИД

AND Н\_ФОРМЫ\_ОБУЧЕНИЯ.НАИМЕНОВАНИЕ = 'Заочная'

JOIN Н\_НАПРАВЛЕНИЯ\_СПЕЦИАЛ ON Н\_ПЛАНЫ.НАПС\_ИД = Н\_НАПРАВЛЕНИЯ\_СПЕЦИАЛ.ИД

JOIN Н\_НАПР\_СПЕЦ ON Н\_НАПРАВЛЕНИЯ\_СПЕЦИАЛ.НС\_ИД = Н\_НАПР\_СПЕЦ.ИД

AND Н\_НАПР\_СПЕЦ.НАИМЕНОВАНИЕ = 'Программная инженерия'

WHERE Н\_УЧЕНИКИ.ИД IN (

SELECT Н\_УЧЕНИКИ.ИД

FROM Н\_УЧЕНИКИ

WHERE Н\_УЧЕНИКИ.КОНЕЦ > '2012-09-01'::timestamp

AND Н\_УЧЕНИКИ.ПРИЗНАК = 'отчисл'

);

7. SELECT DISTINCT Н\_ЛЮДИ.ИД, Н\_ЛЮДИ.ФАМИЛИЯ, Н\_ЛЮДИ.ИМЯ, Н\_ЛЮДИ.ОТЧЕСТВО, Н\_ЛЮДИ.ДАТА\_РОЖДЕНИЯ

FROM Н\_УЧЕНИКИ

JOIN Н\_ЛЮДИ ON Н\_ЛЮДИ.ИД = Н\_УЧЕНИКИ.ЧЛВК\_ИД

JOIN (

SELECT DISTINCT Н\_ЛЮДИ.ИД, Н\_ЛЮДИ.ФАМИЛИЯ, Н\_ЛЮДИ.ИМЯ, Н\_ЛЮДИ.ОТЧЕСТВО, Н\_ЛЮДИ.ДАТА\_РОЖДЕНИЯ

FROM Н\_УЧЕНИКИ

JOIN Н\_ЛЮДИ ON Н\_ЛЮДИ.ИД = Н\_УЧЕНИКИ.ЧЛВК\_ИД

) AS s2 ON Н\_ЛЮДИ.ФАМИЛИЯ = s2.ФАМИЛИЯ AND Н\_ЛЮДИ.ДАТА\_РОЖДЕНИЯ <> s2.ДАТА\_РОЖДЕНИЯ

ORDER BY Н\_ЛЮДИ.ФАМИЛИЯ, Н\_ЛЮДИ.ИМЯ;

# Вывод

При выполнении лабораторной работы я познакомился с основными функциями языка SQL и диалекта PostgreSQL. Научился писать запросы, отсеивать и сортировать полученные данные.