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 № 312508

**Выполнил**:

Студент группы P3125

Михайлов Павел

Максимович

**Преподаватель:** Егошин Алексей

Васильевич

г. Санкт-Петербург, 2023

# Задание.

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

Вариант: 3125666

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

1. Сделать запрос для получения атрибутов из указанных таблиц, применив фильтры по указанным условиям:  
   Таблицы: Н\_ЛЮДИ, Н\_СЕССИЯ.  
   Вывести атрибуты: Н\_ЛЮДИ.ИД, Н\_СЕССИЯ.ЧЛВК\_ИД.  
   Фильтры (AND):  
   a) Н\_ЛЮДИ.ИМЯ = Николай.  
   b) Н\_СЕССИЯ.УЧГОД < 2008/2009.  
   Вид соединения: INNER JOIN.

**SELECT** Н\_ЛЮДИ.ИД,

Н\_СЕССИЯ.ЧЛВК\_ИД,

**FROM** Н\_СЕССИЯ

**INNER** **JOIN** Н\_ЛЮДИ **ON** Н\_ЛЮДИ.ИД=Н\_СЕССИЯ.ЧЛВК\_ИД

**WHERE** Н\_ЛЮДИ.ИМЯ='Николай'

**AND** to\_number(split\_part(Н\_СЕССИЯ.УЧГОД, '/', 1), '9999') < 2008;

1. 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.НАЧАЛО.  
   Фильтры: (AND)  
   a) Н\_ЛЮДИ.ИМЯ < Ярослав.  
   b) Н\_ОБУЧЕНИЯ.НЗК = 999080.  
   c) Н\_УЧЕНИКИ.НАЧАЛО < 2009-02-09.  
   Вид соединения: LEFT JOIN.

**SELECT** p.ОТЧЕСТВО,

o.ЧЛВК\_ИД,

s.НАЧАЛО

**FROM** Н\_ЛЮДИ p

**LEFT** **JOIN** Н\_ОБУЧЕНИЯ o **ON** p.ИД=o.ЧЛВК\_ИД

**LEFT** **JOIN** Н\_УЧЕНИКИ s **ON** s.ЧЛВК\_ИД=p.ИД

**WHERE** p.ИМЯ<'Ярослав'

**AND** o.НЗК='999080'

**AND** (s.НАЧАЛО < '2009-02-09'

**OR** s.НАЧАЛО **IS** **NULL**);

1. Вывести число отчеств без учета повторений.  
   При составлении запроса нельзя использовать DISTINCT.

**SELECT** **count**(\*) **AS** КОЛИЧЕСТВО

**FROM**

(**SELECT** (ОТЧЕСТВО) **AS** ОТЧЕСТВО

**FROM** Н\_ЛЮДИ

**WHERE** ОТЧЕСТВО **IS** **NOT** **NULL**

**GROUP** **BY** ОТЧЕСТВО) **AS** foo;

1. В таблице Н\_ГРУППЫ\_ПЛАНОВ найти номера планов, по которым обучается (обучалось) менее 2 групп на кафедре вычислительной техники.  
   Для реализации использовать подзапрос.

**SELECT** \*

**FROM**

(**SELECT** НОМЕР,

**count**(ГРУППА) **AS** КОЛИЧЕСТВО\_ГРУПП

**FROM** Н\_ГРУППЫ\_ПЛАНОВ gp

**JOIN** Н\_ПЛАНЫ p **ON** gp.ПЛАН\_ИД=p.ИД

**JOIN** Н\_ОТДЕЛЫ o **ON** o.ИД=p.ОТД\_ИД

**WHERE** o.КОРОТКОЕ\_ИМЯ=’КТиУ’

**GROUP** **BY** НОМЕР) **AS** foo

**WHERE** foo.КОЛИЧЕСТВО\_ГРУПП<2;

1. Выведите таблицу со средним возрастом студентов во всех группах (Группа, Средний возраст), где средний возраст равен минимальному возрасту в группе 1100.

**SELECT** \*

**FROM**

(**SELECT** ГРУППА,

floor(**AVG**(date\_part('year', age (now(), ДАТА\_РОЖДЕНИЯ)))) **AS** avg\_age

**FROM** Н\_УЧЕНИКИ s

**JOIN** Н\_ЛЮДИ p **ON** s.ЧЛВК\_ИД=p.ИД

**GROUP** **BY** ГРУППА) **AS** res

**WHERE** avg\_age=

(**SELECT** **MIN**(date\_part('year', age (now(), ДАТА\_РОЖДЕНИЯ)))

**FROM** Н\_УЧЕНИКИ s

**JOIN** Н\_ЛЮДИ p **ON** s.ЧЛВК\_ИД=p.ИД

**WHERE** ГРУППА='1100'

**GROUP** **BY** ГРУППА);

1. Получить список студентов, отчисленных после первого сентября 2012 года с очной или 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 EXISTS.

**SELECT** ГРУППА,

ФАМИЛИЯ,

ИМЯ,

ОТЧЕСТВО,

П\_ПРКОК\_ИД

**FROM** (Н\_УЧЕНИКИ s

**JOIN** Н\_ЛЮДИ p **ON** s.ЧЛВК\_ИД=p.ИД) **AS** res

**WHERE** res.ПРИЗНАК='отчисл'

**AND** res.КОНЕЦ > '2012-09-01'

**AND** **EXISTS**

(**SELECT** 1

**FROM** Н\_ГРУППЫ\_ПЛАНОВ gp

**JOIN** Н\_ПЛАНЫ pl **ON** gp.ПЛАН\_ИД = pl.ИД

**JOIN** Н\_НАПРАВЛЕНИЯ\_СПЕЦИАЛ sp **ON** sp.ИД=pl.НАПС\_ИД

**JOIN** Н\_НАПР\_СПЕЦ ns **ON** ns.ИД = sp.НС\_ИД

**WHERE** ns.НАИМЕНОВАНИЕ = 'Программная инженерия'

**AND** gp.ГРУППА=res.ГРУППА

**AND** (pl.ФО\_ИД=1

**OR** pl.ФО\_ИД=3) );

1. Вывести список студентов, имеющих одинаковые отчества, но не совпадающие даты рождения.

**SELECT** ФАМИЛИЯ,

ИМЯ,

ДАТА\_РОЖДЕНИЯ

**FROM** Н\_ЛЮДИ p1

**WHERE** **EXISTS**

(**SELECT** 1

**FROM** Н\_ЛЮДИ p2

**WHERE** p1.ОТЧЕСТВО=p2.ОТЧЕСТВО

**AND** p1.ДАТА\_РОЖДЕНИЯ<>p2.ДАТА\_РОЖДЕНИЯ)

**ORDER** **BY** ФАМИЛИЯ;

# Вывод

В процессе выполнения лабораторной работы я научился работать с запросами, подзапросами, различными соединениями таблиц, использованием условий и переводом типов данных. Также в процессе выполнения были использованы встроенные psql функции(такие как функция получения текущей даты и времени now(), count(), avg()).