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 КОМПЬЮТЕРНЫХ ТЕХНОЛОГИЙ

**ЛАБОРАТОРНАЯ РАБОТА №2**

по дисциплине

«Информационные системы и базы данных»

Вариант № 1654

***Выполнил:***

Студент группы P3116

Векшин Арсений

Иванович

***Преподаватель:***

Горбунов Михаил   
Витальевич

Содержание

**Добавьте заголовки ("Формат > Стили абзацев"), чтобы они появились в оглавлении.**

# Задание

Задание.

По варианту, выданному преподавателем, составить и выполнить запросы к [базе данных "Учебный процесс"](https://se.ifmo.ru/documents/10180/733702/%D0%91%D0%94+%D0%A3%D1%87%D0%B5%D0%B1%D0%BD%D1%8B%D0%B9+%D0%9F%D1%80%D0%BE%D1%86%D0%B5%D1%81%D1%81.pdf/2eae3fcd-ea34-4496-924b-6ee4e889a9e5).

Команда для подключения к базе данных ucheb:

*psql -h pg -d ucheb*

Отчёт по лабораторной работе должен содержать:

1. Текст задания.
2. Реализацию запросов на SQL.
3. Выводы по работе.

Темы для подготовки к защите лабораторной работы:

1. SQL
2. Соединение таблиц
3. Подзапросы
4. Представления
5. Последовательности

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 = Перезачет.  
    b) Н\_ВЕДОМОСТИ.ДАТА < 2022-06-08.  
    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):  
    a) Н\_ЛЮДИ.ОТЧЕСТВО = Владимирович.  
    b) Н\_ВЕДОМОСТИ.ИД > 1250972.  
    c) Н\_СЕССИЯ.ИД = 27640.  
    Вид соединения: INNER JOIN.
3. Составить запрос, который ответит на вопрос, есть ли среди студентов группы 3102 те, кто не имеет отчества.
4. Выдать различные отчества преподавателей и число людей с каждой из этих отчеств, ограничив список отчествами, встречающимися ровно 10 раз на на заочной форме обучения.  
   Для реализации использовать подзапрос.
5. Выведите таблицу со средним возрастом студентов во всех группах (Группа, Средний возраст), где средний возраст меньше максимального возраста в группе 1100.
6. Получить список студентов, зачисленных после первого сентября 2012 года на первый курс очной или заочной формы обучения. В результат включить:  
   номер группы;  
   номер, фамилию, имя и отчество студента;  
   номер и состояние пункта приказа;  
   Для реализации использовать подзапрос с EXISTS.
7. Сформировать запрос для получения числа в группе No 3100 хорошистов.

# Исходный код программы

/\*

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 = Перезачет.

b) Н\_ВЕДОМОСТИ.ДАТА < 2022-06-08.

Вид соединения: INNER JOIN.

\*/

select "Н\_ВЕДОМОСТИ"."ТВ\_ИД", "Н\_ВЕДОМОСТИ"."ЧЛВК\_ИД"

from "Н\_ВЕДОМОСТИ"

inner join "Н\_ТИПЫ\_ВЕДОМОСТЕЙ" on "Н\_ВЕДОМОСТИ"."ТВ\_ИД" = "Н\_ТИПЫ\_ВЕДОМОСТЕЙ"."ИД"

where "Н\_ТИПЫ\_ВЕДОМОСТЕЙ"."НАИМЕНОВАНИЕ" = 'Перезачет'

and "Н\_ВЕДОМОСТИ"."ДАТА" < *cast*('2022-06-08' as timestamp with time zone)

;

/\*

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):

a) Н\_ЛЮДИ.ОТЧЕСТВО = Владимирович.

b) Н\_ВЕДОМОСТИ.ИД > 1250972.

c) Н\_СЕССИЯ.ИД = 27640.

Вид соединения: INNER JOIN.

\*/

select "Н\_ЛЮДИ"."ОТЧЕСТВО", "Н\_ВЕДОМОСТИ"."ДАТА", "Н\_СЕССИЯ"."ЧЛВК\_ИД"

from "Н\_ЛЮДИ"

inner join "Н\_ВЕДОМОСТИ" on "Н\_ЛЮДИ"."ИД" = "Н\_ВЕДОМОСТИ"."ЧЛВК\_ИД"

inner join "Н\_СЕССИЯ" on "Н\_ЛЮДИ"."ИД" = "Н\_СЕССИЯ"."ЧЛВК\_ИД"

where "Н\_ЛЮДИ"."ОТЧЕСТВО" = 'Владимирович'

and "Н\_ВЕДОМОСТИ"."ИД" > 1250972

and "Н\_СЕССИЯ"."ИД" = 27640

;

/\*

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

\*/

select *exists*(select \*

from "Н\_УЧЕНИКИ"

inner join "Н\_ЛЮДИ" НЛ on "Н\_УЧЕНИКИ"."ЧЛВК\_ИД" = НЛ."ИД"

where "Н\_УЧЕНИКИ"."ГРУППА" = '3102'

and (НЛ."ОТЧЕСТВО" IS NULL

or НЛ."ОТЧЕСТВО" ='.'));

/\*

Выдать различные отчества преподавателей и число людей с каждой из этих отчеств,

ограничив список отчествами, встречающимися ровно 10 раз на на заочной форме обучения.

Для реализации использовать подзапрос.

\*/

select "Н\_ЛЮДИ"."ОТЧЕСТВО" as Отчество,

*count*(\*) as count

from "Н\_ЛЮДИ"

where "Н\_ЛЮДИ"."ОТЧЕСТВО" in (

select DISTINCT "Н\_ЛЮДИ"."ОТЧЕСТВО" as Отчество

from "Н\_ЛЮДИ"

inner join "Н\_УЧЕНИКИ" on "Н\_ЛЮДИ"."ИД" = "Н\_УЧЕНИКИ"."ЧЛВК\_ИД"

where "Н\_УЧЕНИКИ"."ПЛАН\_ИД" in (

select "Н\_ПЛАНЫ"."ПЛАН\_ИД"

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\_КВАЛИФИКАЦИИ" on "Н\_НАПРАВЛЕНИЯ\_СПЕЦИАЛ"."КВАЛ\_ИД" = "Н\_КВАЛИФИКАЦИИ"."ИД"

where "Н\_ФОРМЫ\_ОБУЧЕНИЯ"."НАИМЕНОВАНИЕ" = 'Заочная'

and "Н\_КВАЛИФИКАЦИИ"."НАИМЕНОВАНИЕ" like '%Педагог%'

))

group by "Н\_ЛЮДИ"."ОТЧЕСТВО" having *count*(\*) = 10;

/\*

Выведите таблицу со средним возрастом студентов во всех группах (Группа, Средний возраст),

где средний возраст меньше максимального возраста в группе 1100.

\*/

with GROUP\_AVG\_AGE as (

select "Н\_УЧЕНИКИ"."ГРУППА" as Группа,

*avg*(*age*("Н\_ПЛАНЫ"."ДАТА\_УТВЕРЖДЕНИЯ", НЛ."ДАТА\_РОЖДЕНИЯ")) as Ср\_Возраст

from "Н\_УЧЕНИКИ"

inner join "Н\_ЛЮДИ" НЛ on "Н\_УЧЕНИКИ"."ЧЛВК\_ИД" = НЛ."ИД"

inner join "Н\_ПЛАНЫ" on "Н\_УЧЕНИКИ"."ПЛАН\_ИД" = "Н\_ПЛАНЫ"."ИД"

where "Н\_УЧЕНИКИ"."НАЧАЛО" > НЛ."ДАТА\_РОЖДЕНИЯ"

group by "Н\_УЧЕНИКИ"."ГРУППА")

select \* from GROUP\_AVG\_AGE

where GROUP\_AVG\_AGE.Ср\_Возраст < (

select *max*(*age*("Н\_ПЛАНЫ"."ДАТА\_УТВЕРЖДЕНИЯ", НЛ."ДАТА\_РОЖДЕНИЯ"))

from "Н\_УЧЕНИКИ"

join "Н\_ЛЮДИ" НЛ on "Н\_УЧЕНИКИ"."ЧЛВК\_ИД" = НЛ."ИД"

inner join "Н\_ПЛАНЫ" on "Н\_УЧЕНИКИ"."ПЛАН\_ИД" = "Н\_ПЛАНЫ"."ИД"

where "Н\_УЧЕНИКИ"."ГРУППА" = '1100'

and "Н\_УЧЕНИКИ"."НАЧАЛО" > НЛ."ДАТА\_РОЖДЕНИЯ"

)

order by GROUP\_AVG\_AGE.Ср\_Возраст;

/\*

Получить список студентов, зачисленных после первого сентября 2012 года на первый курс очной или заочной формы обучения. В результат включить:

номер группы;

номер, фамилию, имя и отчество студента;

номер и состояние пункта приказа;

Для реализации использовать подзапрос с EXISTS.

\*/

select НУ."ГРУППА" as Группа,

"Н\_ЛЮДИ"."ИД" as isu,

*CONCAT*("Н\_ЛЮДИ"."ФАМИЛИЯ", ' ', "Н\_ЛЮДИ"."ИМЯ", ' ', "Н\_ЛЮДИ"."ОТЧЕСТВО") as ФИО,

НУ."П\_ПРКОК\_ИД" as Номер\_приказа,

НУ."СОСТОЯНИЕ" as Состояние,

НУ."НАЧАЛО"

from "Н\_ЛЮДИ"

join "Н\_УЧЕНИКИ" НУ on "Н\_ЛЮДИ"."ИД" = НУ."ЧЛВК\_ИД"

where НУ."НАЧАЛО" > *cast*('1.09.2012' as timestamp with time zone)

and *exists*(

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'

or "Н\_ФОРМЫ\_ОБУЧЕНИЯ"."НАИМЕНОВАНИЕ" = 'Заочная')

and "Н\_ПЛАНЫ"."КУРС" = 1

and НУ."ПЛАН\_ИД" = "Н\_ПЛАНЫ"."ИД"

)

;

/\*

Сформировать запрос для получения числа в группе No 3100 хорошистов.

\*/

select DISTINCT on ("ФИО")

"Н\_ЛЮДИ"."ИД" as isu,

*CONCAT*("Н\_ЛЮДИ"."ФАМИЛИЯ", ' ', "Н\_ЛЮДИ"."ИМЯ", ' ', "Н\_ЛЮДИ"."ОТЧЕСТВО") as ФИО,

"Н\_УЧЕНИКИ"."ГРУППА"

from "Н\_ЛЮДИ"

join "Н\_ВЕДОМОСТИ" on "Н\_ЛЮДИ"."ИД" = "Н\_ВЕДОМОСТИ"."ЧЛВК\_ИД"

join "Н\_УЧЕНИКИ" on "Н\_ЛЮДИ"."ИД" = "Н\_УЧЕНИКИ"."ЧЛВК\_ИД"

where "Н\_УЧЕНИКИ"."ГРУППА" = '3100'

and not *exists*(

select 1 from "Н\_ВЕДОМОСТИ"

inner join "Н\_ОЦЕНКИ" on "Н\_ОЦЕНКИ"."КОД" = "Н\_ВЕДОМОСТИ"."ОЦЕНКА"

where "Н\_ЛЮДИ"."ИД" = "Н\_ВЕДОМОСТИ"."ЧЛВК\_ИД"

and "Н\_ОЦЕНКИ"."СОРТ" not in (1,2,5))

and *exists*(

select 1 from "Н\_ВЕДОМОСТИ"

inner join "Н\_ОЦЕНКИ" on "Н\_ОЦЕНКИ"."КОД" = "Н\_ВЕДОМОСТИ"."ОЦЕНКА"

where "Н\_ЛЮДИ"."ИД" = "Н\_ВЕДОМОСТИ"."ЧЛВК\_ИД"

and "Н\_ОЦЕНКИ"."СОРТ" =2)

;

Вывод