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

Факультет программной инженерии и компьютерной техники

**Лабораторная работа №2**

Вариант №18009

Выполнил

Макогон Ярослав Вадимович

Номер группы: P3118

Проверила

Бострикова Д. К.

**Содержание**

[Задание 3](#_Toc194753551)

[Решение 4](#_Toc194753552)

[Вывод 11](#_Toc194753553)

### 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

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

1. Сделать запрос для получения атрибутов из указанных таблиц, применив фильтры по указанным условиям:  
   Таблицы: Н\_ОЦЕНКИ, Н\_ВЕДОМОСТИ.  
   Вывести атрибуты: Н\_ОЦЕНКИ.ПРИМЕЧАНИЕ, Н\_ВЕДОМОСТИ.ЧЛВК\_ИД.  
   Фильтры (AND):  
   a) Н\_ОЦЕНКИ.КОД > 3.  
   b) Н\_ВЕДОМОСТИ.ЧЛВК\_ИД > 142390.  
   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.ИД, Н\_СЕССИЯ.ЧЛВК\_ИД.  
   Фильтры (AND):  
   a) Н\_ЛЮДИ.ИМЯ > Николай.  
   b) Н\_ВЕДОМОСТИ.ДАТА < 2022-06-08.  
   c) Н\_СЕССИЯ.ИД > 27640.  
   Вид соединения: RIGHT JOIN.
3. Вывести число студентов вечерней формы обучения, которые не имеет отчества.  
   Ответ должен содержать только одно число.
4. Найти группы, в которых в 2011 году было менее 10 обучающихся студентов на ФКТИУ.  
   Для реализации использовать соединение таблиц.
5. Выведите таблицу со средними оценками студентов группы 4100 (Номер, ФИО, Ср\_оценка), у которых средняя оценка не больше минимальной оценк(е|и) в группе 3100.
6. Получить список студентов, отчисленных до первого сентября 2012 года с очной формы обучения (специальность: 230101). В результат включить:  
   номер группы;  
   номер, фамилию, имя и отчество студента;  
   номер пункта приказа;  
   Для реализации использовать подзапрос с EXISTS.
7. Вывести список студентов, имеющих одинаковые имена, но не совпадающие ид.

### Решение

*1. Сделать запрос для получения атрибутов из указанных таблиц, применив фильтры по указанным условиям:*

*Таблицы: Н\_ОЦЕНКИ, Н\_ВЕДОМОСТИ.*

*Вывести атрибуты: Н\_ОЦЕНКИ.ПРИМЕЧАНИЕ, Н\_ВЕДОМОСТИ.ЧЛВК\_ИД.*

*Фильтры (AND):*

*a) Н\_ОЦЕНКИ.КОД > 3.*

*b) Н\_ВЕДОМОСТИ.ЧЛВК\_ИД > 142390.*

*Вид соединения: INNER JOIN.*

SELECT

Н\_ОЦЕНКИ.ПРИМЕЧАНИЕ,

Н\_ВЕДОМОСТИ.ЧЛВК\_ИД

FROM

Н\_ВЕДОМОСТИ

INNER JOIN

Н\_ОЦЕНКИ ON Н\_ВЕДОМОСТИ.ОЦЕНКА = Н\_ОЦЕНКИ.КОД

WHERE

Н\_ОЦЕНКИ.**КОД** IN ('1', '2', '3', '4', '5')-- тк Н\_ОЦЕНКИ.КОД может быть нечисловой строкой

AND CAST(Н\_ОЦЕНКИ.КОД AS int) > 3

AND Н\_ВЕДОМОСТИ.ЧЛВК\_ИД > 142390;

Вывод:

ПРИМЕЧАНИЕ | ЧЛВК\_ИД

------------+---------

хорошо | 142391

хорошо | 142393

хорошо | 142394

отлично | 142395

хорошо | 142396

отлично | 142397

хорошо | 142398

отлично | 142399

хорошо | 142422

отлично | 142400

хорошо | 142401

отлично | 153942

хорошо | 153355

отлично | 146246

хорошо | 146107

…

*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.ИД, Н\_СЕССИЯ.ЧЛВК\_ИД.*

*Фильтры (AND):*

*a) Н\_ЛЮДИ.ИМЯ > Николай.*

*b) Н\_ВЕДОМОСТИ.ДАТА < 2022-06-08.*

*c) Н\_СЕССИЯ.ИД > 27640.*

*Вид соединения: RIGHT JOIN.*

SELECT

Н\_ЛЮДИ.ОТЧЕСТВО,

Н\_ВЕДОМОСТИ.ИД,

Н\_СЕССИЯ.ЧЛВК\_ИД

FROM

Н\_ЛЮДИ

RIGHT JOIN

Н\_ВЕДОМОСТИ ON Н\_ЛЮДИ.ИД = Н\_ВЕДОМОСТИ.ЧЛВК\_ИД

RIGHT JOIN

Н\_СЕССИЯ ON Н\_ЛЮДИ.ИД = Н\_СЕССИЯ.ЧЛВК\_ИД

WHERE

Н\_ЛЮДИ.ИМЯ > 'Николай'

-- AND Н\_ВЕДОМОСТИ.ДАТА < CAST('2022-06-08' AS DATE) -- (1)

-- AND Н\_СЕССИЯ.ИД > 27640; -- (2)

Вывод:

ОТЧЕСТВО | ИД | ЧЛВК\_ИД

------------+--------+---------

Дмитриевич | 180645 | 117954

Дмитриевич | 180603 | 117954

Дмитриевич | 180618 | 117954

Дмитриевич | 180631 | 117954

Дмитриевич | 180660 | 117954

Дмитриевич | 180673 | 117954

Дмитриевич | 180686 | 117954

Дмитриевич | 180701 | 117954

Дмитриевич | 223435 | 117954

Дмитриевич | 189736 | 117954

Дмитриевич | 221585 | 117954

Дмитриевич | 221623 | 117954

…

C (1) и (2) – пустой вывод из-за неравномерных данных в бд

*3. Вывести число студентов вечерней формы обучения, которые не имеет отчества. Ответ должен содержать только одно число.*

SELECT

COUNT(\*)

FROM

Н\_УЧЕНИКИ

INNER JOIN

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

INNER JOIN

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

RIGHT JOIN

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

WHERE

Н\_ЛЮДИ.ОТЧЕСТВО = '.'

AND Н\_ФОРМЫ\_ОБУЧЕНИЯ.НАИМЕНОВАНИЕ ~ 'вечерняя';

Вывод:

count

-------

41

*4. Найти группы, в которых в 2011 году было менее 10 обучающихся студентов на ФКТИУ.*

*Для реализации использовать соединение таблиц.*

SELECT

Н\_УЧЕНИКИ.ГРУППА

FROM

Н\_УЧЕНИКИ

INNER JOIN

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

INNER JOIN

Н\_ОТДЕЛЫ ON Н\_ПЛАНЫ.ОТД\_ИД = Н\_ОТДЕЛЫ.ИД

WHERE

Н\_ОТДЕЛЫ.КОРОТКОЕ\_ИМЯ = 'КТиУ'

AND Н\_ПЛАНЫ.УЧЕБНЫЙ\_ГОД = '2010/2011'

GROUP BY

Н\_УЧЕНИКИ.ГРУППА

HAVING

COUNT(\*) < 10;

Вывод:

ГРУППА

--------

5509

6114

5114

6120

*5. Выведите таблицу со средними оценками студентов группы 4100 (Номер, ФИО, Ср\_оценка), у которых средняя оценка не больше минимальной оценк(е|и) в группе 3100.*

SELECT

u1.ИД AS Номер,

Н\_ЛЮДИ.ИМЯ || ' ' || Н\_ЛЮДИ.ФАМИЛИЯ || ' ' || Н\_ЛЮДИ.ОТЧЕСТВО AS ФИО,

AVG(CAST(v1.ОЦЕНКА AS int))

FROM

Н\_УЧЕНИКИ u1

INNER JOIN

Н\_ЛЮДИ ON u1.ЧЛВК\_ИД = Н\_ЛЮДИ.ИД

INNER JOIN

Н\_ВЕДОМОСТИ v1 USING(ЧЛВК\_ИД)

WHERE

v1.ОЦЕНКА IN ('1', '2', '3', '4', '5')

GROUP BY

u1.ИД,

Н\_ЛЮДИ.ИМЯ,

Н\_ЛЮДИ.ФАМИЛИЯ,

Н\_ЛЮДИ.ОТЧЕСТВО

HAVING

AVG(CAST(v1.ОЦЕНКА AS int)) <= ALL(

SELECT

CAST(v2.ОЦЕНКА AS int)

FROM

Н\_ВЕДОМОСТИ v2

INNER JOIN

Н\_УЧЕНИКИ u2 USING(ЧЛВК\_ИД)

WHERE

v2.ОЦЕНКА ~ '^\d+$'

AND u2.ГРУППА = '3100'

);

Вывод:

Номер | ФИО | avg

--------+--------------------------------+--------------------

22848 | Николай Сергеева Рашидович | 2.0000000000000000

23041 | Андрей Мишин Александрович | 2.0000000000000000

23042 | Полина Чивилихин Николаевич | 2.0000000000000000

31421 | Андрей Мишин Александрович | 2.0000000000000000

34232 | Николай Сергеева Рашидович | 2.0000000000000000

34242 | Полина Чивилихин Николаевич | 2.0000000000000000

37578 | Полина Чивилихин Николаевич | 2.0000000000000000

37579 | Николай Сергеева Рашидович | 2.0000000000000000

48846 | Антон Целиков Сергеевна | 2.0000000000000000

…

*6. Получить список студентов, отчисленных до первого сентября 2012 года с очной формы обучения (специальность: 230101). В результат включить:*

*номер группы;*

*номер, фамилию, имя и отчество студента;*

*номер пункта приказа;*

*Для реализации использовать подзапрос с EXISTS.*

SELECT

Н\_УЧЕНИКИ.ГРУППА,

Н\_УЧЕНИКИ.ИД,

Н\_ЛЮДИ.ИМЯ || ' ' || Н\_ЛЮДИ.ФАМИЛИЯ || ' ' || Н\_ЛЮДИ.ОТЧЕСТВО AS ФИО

FROM

Н\_УЧЕНИКИ

INNER JOIN

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

WHERE

Н\_УЧЕНИКИ.ПРИЗНАК = 'отчисл'

AND Н\_УЧЕНИКИ.КОНЕЦ < DATE('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 = 'Очная'

AND Н\_НАПР\_СПЕЦ.КОД\_НАПРСПЕЦ = '230101'

);

Вывод:

ГРУППА | ИД | ФИО

--------+--------+------------------------------------

2101 | 203985 | Ирина Родионов Дмитриевич

2101 | 203165 | Галина Федорова Павловна

2101 | 201735 | Роман Лучников Александровна

2101 | 177524 | . Иванов Викторович

2100 | 183057 | Михаил Смирновский Евгеньевич

1103 | 107887 | Артем Соколов Игоревич

1100 | 109414 | Евгений Кандиболоцкий Анатольевна

2120 | 177520 | Дмитрий Суворов Владимирович

2101 | 225936 | Роман Лучников Александровна

2101 | 225516 | Галина Федорова Павловна

…

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

SELECT

u1.ИД,

p1.ИМЯ || ' ' || p1.ФАМИЛИЯ || ' ' || p1.ОТЧЕСТВО AS ФИО

FROM

Н\_УЧЕНИКИ u1

INNER JOIN

Н\_ЛЮДИ p1 ON u1.ЧЛВК\_ИД = p1.ИД

WHERE EXISTS(

SELECT

\*

FROM

Н\_УЧЕНИКИ u2

INNER JOIN

Н\_ЛЮДИ p2 ON u2.ЧЛВК\_ИД = p2.ИД

WHERE

p1.ИМЯ ~ '^\w+$'

AND u1.ИД != u2.ИД

AND p1.ИМЯ = p2.ИМЯ

)

ORDER BY

p1.ИМЯ;

Вывод:

ИД | ФИО

--------+------------------------------------------

26603 | Адам Смирнов Сергеевич

66638 | Адам Смирнов Сергеевич

34937 | Адам Смирнов Сергеевич

46076 | Адам Смирнов Сергеевич

8274 | Адам Смирнов Сергеевич

17235 | Адам Смирнов Сергеевич

80819 | Адам Смирнов Сергеевич

…

*Доп. Список студенток старше 30 которы сдавали ссесию в аудитории 3 и 4 этажа (1 цифра) получивших оценку 4 и выше. ср балл < 4*

SELECT

Н\_ЛЮДИ.ИМЯ || ' ' || Н\_ЛЮДИ.ФАМИЛИЯ || ' ' || Н\_ЛЮДИ.ОТЧЕСТВО AS ФИО,

AGE(Н\_ЛЮДИ.ДАТА\_РОЖДЕНИЯ),

Н\_СЕССИЯ.АУДИТОРИЯ,

AVG(CAST(v1.ОЦЕНКА AS int)) AS Ср\_оценка,

v2.ОЦЕНКА

FROM

Н\_УЧЕНИКИ

INNER JOIN

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

INNER JOIN

Н\_ВЕДОМОСТИ v1 ON Н\_УЧЕНИКИ.ЧЛВК\_ИД = v1.ЧЛВК\_ИД

INNER JOIN

Н\_СЕССИЯ ON Н\_УЧЕНИКИ.ЧЛВК\_ИД = Н\_СЕССИЯ.ЧЛВК\_ИД

INNER JOIN

Н\_СОДЕРЖАНИЯ\_ЭЛЕМЕНТОВ\_СТРОК ON Н\_СОДЕРЖАНИЯ\_ЭЛЕМЕНТОВ\_СТРОК.ИД = Н\_СЕССИЯ.СЭС\_ИД

INNER JOIN

Н\_ВЕДОМОСТИ v2 ON Н\_СОДЕРЖАНИЯ\_ЭЛЕМЕНТОВ\_СТРОК.ИД = v2.СЭС\_ИД

WHERE

v1.ОЦЕНКА IN ('1', '2', '3', '4', '5')

AND v2.ОЦЕНКА IN ('1', '2', '3', '4', '5')

AND Н\_ЛЮДИ.ПОЛ = 'Ж'

AND AGE(Н\_ЛЮДИ.ДАТА\_РОЖДЕНИЯ) > INTERVAL '30 years'

AND Н\_СЕССИЯ.АУДИТОРИЯ ~ '^[34]'

GROUP BY

Н\_ЛЮДИ.ИМЯ,

Н\_ЛЮДИ.ФАМИЛИЯ,

Н\_ЛЮДИ.ОТЧЕСТВО,

Н\_СЕССИЯ.АУДИТОРИЯ,

Н\_ЛЮДИ.ДАТА\_РОЖДЕНИЯ,

v2.ОЦЕНКА

HAVING

AVG(CAST(v1.ОЦЕНКА AS int)) < 4

AND CAST(v2.ОЦЕНКА AS int) >= 4;

Вывод:

ФИО | age | АУДИТОРИЯ | Ср\_оценка | ОЦЕНКА

-----+-----+-----------+-----------+--------

Пусто из-за неравномерности данных в бд.

### Вывод

### Познакомился с механикой соединения таблиц в SQL (JOIN).

### Узнал, как работать с подзапросами в SQL.