

Университет ИТМО, факультет ПИиКТ

Лабораторная работа №2

Вариант 4652

Дисциплина: Информационные системы и базы данных

Выполнил: Чангалиди Антон

Группа: Р33113

Преподаватель:  
Николаев Владимир Вячеславович

г. Санкт-Петербург

2020 г.

## ЛАБОРАТОРНАЯ РАБОТА № 2

### Задание

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

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):  
а) Н\_ОЦЕНКИ.КОД > осв.  
б) Н\_ВЕДОМОСТИ.ДАТА > 2010-06-18.  
Вид соединения: RIGHT 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.ДАТА.  
Фильтры (AND):  
а) Н\_ЛЮДИ.ИД = 100012.  
б) Н\_ВЕДОМОСТИ.ДАТА = 2022-06-08.  
с) Н\_СЕССИЯ.ДАТА > 2002-01-04.  
Вид соединения: INNER JOIN.
3. Вывести число студентов вечерней формы обучения, которые младше 20 лет.  
Ответ должен содержать только одно число.
4. В таблице Н\_ГРУППЫ\_ПЛАНОВ найти номера планов, по которым обучается (обучалось) более 2 групп ФКТИУ.  
Для реализации использовать соединение таблиц.
5. Выведите таблицу со средним возрастом студентов во всех группах (Группа, Средний возраст), где средний возраст больше минимального возраста в группе 3100.
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 EXISTS.
7. Сформировать запрос для получения числа в группе No 3100 отличников.

### Реализация запросов на SQL:

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):

а) Н\_ОЦЕНКИ.КОД > осв.

б) Н\_ВЕДОМОСТИ.ДАТА > 2010-06-18.

Вид соединения: RIGHT JOIN.

```
SELECT "Н_ОЦЕНКИ"."ПРИМЕЧАНИЕ",  
       "Н_ВЕДОМОСТИ"."ЧЛВК_ИД"  
FROM "Н_ОЦЕНКИ"  
      RIGHT JOIN "Н_ВЕДОМОСТИ" ON "Н_ОЦЕНКИ"."КОД" = "Н_ВЕДОМОСТИ"."ОЦЕНКА"  
WHERE "Н_ОЦЕНКИ"."КОД" > 'осв'  
AND "Н_ВЕДОМОСТИ"."ДАТА" > '2010-06-18';
```

Ничего не вывело, так как "Н\_ОЦЕНКИ"."КОД" = 'осв' - максимальное значение (больше него быть не может):

	КОД	ПРИМЕЧАНИЕ	СОРТ
1	осв	освобождение	8
2	неявка	неявка	7
3	незач	незачет	6
4	зачет	зачет	5
5	99	диплом с отличием	<null>
6	5	отлично	1
7	4	хорошо	2
8	3	удовлетворительно	3
9	2	неудовлетворительно	4

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.ДАТА.

Фильтры (AND):

а) Н\_ЛЮДИ.ИД = 100012.

б) Н\_ВЕДОМОСТИ.ДАТА = 2022-06-08.

с) Н\_СЕССИЯ.ДАТА > 2002-01-04.

Вид соединения: INNER JOIN.

```
SELECT "Н_ЛЮДИ"."ОТЧЕСТВО",  
       "Н_ВЕДОМОСТИ"."ИД",  
       "Н_СЕССИЯ"."ДАТА"  
FROM "Н_ЛЮДИ"  
      INNER JOIN "Н_СЕССИЯ" on "Н_ЛЮДИ"."ИД" = "Н_СЕССИЯ"."ЧЛВК_ИД"  
      INNER JOIN "Н_ВЕДОМОСТИ" on "Н_СЕССИЯ"."СЭС_ИД" = "Н_ВЕДОМОСТИ"."СЭС_ИД"  
WHERE "Н_ЛЮДИ"."ИД" = '100012'  
AND "Н_ВЕДОМОСТИ"."ДАТА" = '2022-06-08'  
AND "Н_СЕССИЯ"."ДАТА" > '2002-01-04';
```

Ничего не вывело, так как "Н\_ВЕДОМОСТИ"."ДАТА" = '2022-06-08' - год из будущего, а базы из прошлого:)

3. Вывести число студентов вечерней формы обучения, которые младше 20 лет.

Ответ должен содержать только одно число.

```
SELECT COUNT("Н_УЧЕНИКИ"."ИД")  
FROM "Н_УЧЕНИКИ"  
      JOIN "Н_ЛЮДИ" нл on "Н_УЧЕНИКИ"."ЧЛВК_ИД" = нл."ИД" AND date_part('year',  
age(нл."ДАТА_РОЖДЕНИЯ", :date)) < 20
```

```
JOIN "Н_ПЛАНЫ" НП on "Н_УЧЕНИКИ"."ПЛАН_ИД" = НП."ПЛАН_ИД"
JOIN "Н_ФОРМЫ_ОБУЧЕНИЯ" НФО on НП."ФО_ИД" = НФО."ИД" AND НФО."ИМЯ_В_ИМИН_ПАДЕЖЕ" =
'вечерняя';
```

1 row ▾	
	count ▾
1	485

4. В таблице Н\_ГРУППЫ\_ПЛАНОВ найти номера планов, по которым обучается (обучалось) более 2 групп ФКТИУ.







Для реализации использовать соединение таблиц.

```
SELECT НГП."ПЛАН_ИД"
FROM "Н_ГРУППЫ_ПЛАНОВ" НГП
      JOIN "Н_ПЛАНЫ" НП on НГП."ПЛАН_ИД" = НП."ИД"
      JOIN "Н_ОТДЕЛЫ" НО on НП."ОТД_ИД" = НО."ИД"
WHERE НО."КОРОТКОЕ_ИМЯ" = 'КТИУ'
GROUP BY НГП."ПЛАН_ИД"
HAVING COUNT(*) > 2;
```

68 rows ▾	
	"ПЛАН_ИД" ▾
1	2837
2	5603
3	601
4	276
5	2144
6	5284

5. Выведите таблицу со средним возрастом студентов во всех группах (Группа, Средний возраст), где средний возраст больше минимального возраста в группе 3100.

```
SELECT НУ."ГРУППА",
      AVG(date_part('year', age("ДАТА_РОЖДЕНИЯ"::date))) AS "Средний возраст"
FROM "Н_УЧЕНИКИ" НУ
      JOIN "Н_ЛЮДИ" НЛ on НУ."ЧЛВК_ИД" = НЛ."ИД"
GROUP BY НУ."ГРУППА"
HAVING AVG(date_part('year', age("ДАТА_РОЖДЕНИЯ"::date))) >
      (SELECT MIN(date_part('year', age(НЛ."ДАТА_РОЖДЕНИЯ"::date)))
FROM "Н_УЧЕНИКИ" НУ
      JOIN "Н_ЛЮДИ" НЛ on НУ."ЧЛВК_ИД" = НЛ."ИД" AND НУ."ГРУППА" = '3100');
```

< < 158 rows > >			   		
	 "ГРУППА" ▾		 "Средний возраст" ▾		
1	2538		30.953051643192488		
2	6103		36.052287581699346		
3	254		26.586046511627906		
4	3108		28.25		
5	6100		32.58620689655172		

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 EXISTS.

```
SELECT НУ."ГРУППА",
       "Н_люди"."ИД",
       "Н_люди"."ФАМИЛИЯ",
       "Н_люди"."ИМЯ",
       "Н_люди"."ОТЧЕСТВО",
       НУ."П_ПРКОК_ИД"
FROM "Н_УЧЕНИКИ" НУ
      JOIN "Н_ЛЮДИ" ON "Н_ЛЮДИ"."ИД" = НУ."ЧЛВК_ИД"
      JOIN "Н_ПЛАНЫ" ON НУ."ПЛАН_ИД" = "Н_ПЛАНЫ"."ИД"
      JOIN "Н_ФОРМЫ_ОБУЧЕНИЯ" ON "Н_ПЛАНЫ"."ФО_ИД" = "Н_ФОРМЫ_ОБУЧЕНИЯ"."ИД"
      AND ("Н_ФОРМЫ_ОБУЧЕНИЯ"."НАИМЕНОВАНИЕ" = 'Заочная' OR "Н_ФОРМЫ_ОБУЧЕНИЯ"."НАИМЕНОВАНИЕ" =
'Очная')
      JOIN "Н_НАПРАВЛЕНИЯ_СПЕЦИАЛ" ON "Н_ПЛАНЫ"."НАПС_ИД" = "Н_НАПРАВЛЕНИЯ_СПЕЦИАЛ"."ИД"
      JOIN "Н_НАПР_СПЕЦ" ON "Н_НАПР_СПЕЦ"."ИД" = "Н_НАПРАВЛЕНИЯ_СПЕЦИАЛ"."НС_ИД"
      AND "Н_НАПР_СПЕЦ"."НАИМЕНОВАНИЕ" = 'Программная инженерия'
WHERE EXISTS (SELECT *
              FROM "Н_УЧЕНИКИ"
              WHERE "Н_УЧЕНИКИ"."НАЧАЛО" = '01-09-2012');
```

Нет результатов \\_0\_0/.

7. Сформировать запрос для получения числа в группе No 3100 отличников.

```
SELECT COUNT(*)
FROM (
      SELECT "Н_люди"."ИД"
      FROM "Н_ЛЮДИ"
           JOIN "Н_УЧЕНИКИ" ON "Н_УЧЕНИКИ"."ЧЛВК_ИД" = "Н_ЛЮДИ"."ИД" AND
           "Н_УЧЕНИКИ"."ГРУППА" = '3100'
           JOIN "Н_ВЕДОМОСТИ" ON "Н_ВЕДОМОСТИ"."ЧЛВК_ИД" = "Н_ЛЮДИ"."ИД" AND
           "ОЦЕНКА" NOT IN ('осв', 'неявка', 'зачет', 'незач')
      GROUP BY "Н_люди"."ИД"
      HAVING AVG(CAST("ОЦЕНКА" AS NUMERIC)) >= 5.0
    ) as genius;
```

Output	
COUNT(*):begin	
1 row	
count	
1	11

## **Выводы**

В ходе выполнения лабораторной работы я глубже изучил DML: с помощью комбинаций команд `join`, `select`, `and`, `exists` я выполнил нетривиальные запросы к базе данных, в том числе с агрегациями, и использования подзапросов.