Факультет Программной Инженерии и Компьютерной Техники

Лабораторная работа №2

Вариант 676

Выполнил:

Шмунк Андрей Александрович

Группа P3108

Преподаватели:

Афанасьев Дмитрий Борисович

Николаев Владимир Вячеславович

Оглавление

[Текст задания 3](#_Toc163056797)

[Реализация запросов на SQL 4](#_Toc163056798)

[Вывод 7](#_Toc163056799)

# Текст задания

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:  
   Н\_ТИПЫ\_ВЕДОМОСТЕЙ, Н\_ВЕДОМОСТИ.  
   Вывести атрибуты: Н\_ТИПЫ\_ВЕДОМОСТЕЙ.НАИМЕНОВАНИЕ, Н\_ВЕДОМОСТИ.ИД.  
   Фильтры (AND):  
   a) Н\_ТИПЫ\_ВЕДОМОСТЕЙ.НАИМЕНОВАНИЕ = Перезачет.  
   b) Н\_ВЕДОМОСТИ.ЧЛВК\_ИД = 153285.  
   c) Н\_ВЕДОМОСТИ.ЧЛВК\_ИД < 117219.  
   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.УЧГОД.  
   Фильтры (AND):  
   a) Н\_ЛЮДИ.ИД = 100012.  
   b) Н\_ВЕДОМОСТИ.ИД > 1250981.  
   Вид соединения: INNER JOIN.
3. Вывести число фамилий и имен без учета повторений.  
   При составлении запроса нельзя использовать DISTINCT.
4. Выдать различные фамилии преподавателей и число людей с каждой из этих фамилий, ограничив список фамилиями, встречающимися менее 50 раз на на заочной форме обучения.  
   Для реализации использовать соединение таблиц.
5. Выведите таблицу со средними оценками студентов группы 4100 (Номер, ФИО, Ср\_оценка), у которых средняя оценка равна средней оценк(е|и) в группе 1101.
6. Получить список студентов, отчисленных до первого сентября 2012 года с заочной формы обучения. В результат включить:  
   номер группы;  
   номер, фамилию, имя и отчество студента;  
   номер пункта приказа;  
   Для реализации использовать подзапрос с EXISTS.
7. Вывести список людей, не являющихся или не являвшихся студентами СПбГУ ИТМО (данные, о которых отсутствуют в таблице Н\_УЧЕНИКИ). В запросе нельзя использовать DISTINCT.

# Реализация запросов на SQL

/\*

Сделать запрос для получения атрибутов из указанных таблиц, применив фильтры по указанным условиям:  
Н\_ТИПЫ\_ВЕДОМОСТЕЙ, Н\_ВЕДОМОСТИ.  
Вывести атрибуты: Н\_ТИПЫ\_ВЕДОМОСТЕЙ.НАИМЕНОВАНИЕ, Н\_ВЕДОМОСТИ.ИД.  
Фильтры (AND):  
a) Н\_ТИПЫ\_ВЕДОМОСТЕЙ.НАИМЕНОВАНИЕ = Перезачет.  
b) Н\_ВЕДОМОСТИ.ЧЛВК\_ИД = 153285.  
c) Н\_ВЕДОМОСТИ.ЧЛВК\_ИД < 117219.  
Вид соединения: RIGHT JOIN.

\*/

**SELECT** НАИМЕНОВАНИЕ,

Н\_ВЕДОМОСТИ.ИД

**FROM** Н\_ВЕДОМОСТИ

**RIGHT JOIN** Н\_ТИПЫ\_ВЕДОМОСТЕЙ

**ON** Н\_ВЕДОМОСТИ.ВЕД\_ИД = Н\_ТИПЫ\_ВЕДОМОСТЕЙ.ИД

**WHERE** НАИМЕНОВАНИЕ = 'Перезачет'

**AND** ЧЛВК\_ИД = 153285

**AND** ЧЛВК\_ИД < 117219;

/\*

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.УЧГОД.  
Фильтры (AND):  
a) Н\_ЛЮДИ.ИД = 100012.  
b) Н\_ВЕДОМОСТИ.ИД > 1250981.  
Вид соединения: INNER JOIN.

\*/

**SELECT** ОТЧЕСТВО,

Н\_ВЕДОМОСТИ.ИД,

УЧГОД

**FROM** Н\_ЛЮДИ

**JOIN** Н\_СЕССИЯ

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

**JOIN** Н\_ВЕДОМОСТИ

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

**WHERE** Н\_ЛЮДИ.ИД = 100012

**AND** Н\_ВЕДОМОСТИ.ИД > 1250981;

/\*

Вывести число фамилий и имен без учета повторений.  
При составлении запроса нельзя использовать DISTINCT.

\*/

**SELECT COUNT**(ФАМИЛИЯ)

**FROM**

(**SELECT** ФАМИЛИЯ, ИМЯ **FROM** Н\_ЛЮДИ **GROUP BY** ФАМИЛИЯ, ИМЯ) **AS** foo;

/\*

Выдать различные фамилии преподавателей и число людей с каждой из этих фамилий, ограничив список фамилиями, встречающимися менее 50 раз на на заочной форме обучения.  
Для реализации использовать соединение таблиц.

\*/

**SELECT** ФАМИЛИЯ,

**COUNT**(ФАМИЛИЯ)

**FROM** Н\_ЛЮДИ

**JOIN** Н\_УЧЕНИКИ

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

**JOIN** Н\_ПЛАНЫ

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

**JOIN** Н\_ФОРМЫ\_ОБУЧЕНИЯ

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

**WHERE** НАИМЕНОВАНИЕ = 'Заочная'

**AND** ФАМИЛИЯ **IN** (

**SELECT** ФАМИЛИЯ

**FROM** Н\_ЛЮДИ

**RIGHT JOIN** Н\_СЕССИЯ

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

**WHERE** Н\_ЛЮДИ.ИД **IS NOT NULL**

)

**GROUP BY** ФАМИЛИЯ

**HAVING** COUNT(ФАМИЛИЯ) < 50;

/\*

Выведите таблицу со средними оценками студентов группы 4100 (Номер, ФИО, Ср\_оценка), у которых средняя оценка равна средней оценк(е|и) в группе 1101.

\*/

**SELECT** Н\_ЛЮДИ.ИД,

**AVG**(**CAST**(ОЦЕНКА **AS NUMERIC**)),

ФАМИЛИЯ,

ИМЯ,

ОТЧЕСТВО

**FROM** Н\_ЛЮДИ

**JOIN** Н\_УЧЕНИКИ

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

**JOIN** Н\_ВЕДОМОСТИ

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

**AND** ОЦЕНКА **NOT IN** ( 'осв', 'неявка', 'зачет', 'незач' )

**GROUP BY** Н\_ЛЮДИ.ИД,

ФАМИЛИЯ,

ИМЯ,

ОТЧЕСТВО

**HAVING AVG**(**CAST**(ОЦЕНКА **AS NUMERIC**)) =

(

**SELECT** **AVG**(**CAST**(ОЦЕНКА **AS NUMERIC**))

**FROM** Н\_УЧЕНИКИ

**JOIN** Н\_ВЕДОМОСТИ

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

**AND** Н\_УЧЕНИКИ.ГРУППА = '1101'

**AND** ОЦЕНКА **NOT IN** ( 'осв', 'неявка', 'зачет', 'незач' )

);

/\*

Получить список студентов, отчисленных до первого сентября 2012 года с заочной формы обучения. В результат включить:  
номер группы;  
номер, фамилию, имя и отчество студента;  
номер пункта приказа;  
Для реализации использовать подзапрос с EXISTS.

\*/

**SELECT** ГРУППА,

ВНЕШ\_УЧЕНИКИ.ИД,

ФАМИЛИЯ,

ИМЯ,

ОТЧЕСТВО,

П\_ПРКОК\_ИД

**FROM** Н\_ЛЮДИ

**JOIN** Н\_УЧЕНИКИ AS ВНЕШ\_УЧЕНИКИ

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

**JOIN** Н\_ПЛАНЫ

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

**JOIN** Н\_ФОРМЫ\_ОБУЧЕНИЯ

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

**AND** НАИМЕНОВАНИЕ = 'Заочная'

**WHERE EXISTS**

(

**SELECT** \*

**FROM** Н\_УЧЕНИКИ

**WHERE** ПРИЗНАК = 'отчисл'

**AND** СОСТОЯНИЕ = 'утвержден'

**AND** ИД = ВНЕШ\_УЧЕНИКИ.ИД

**AND** **DATE**(КОНЕЦ) < '2012-09-01'

);

/\*

Вывести список людей, не являющихся или не являвшихся студентами СПбГУ ИТМО (данные, о которых отсутствуют в таблице Н\_УЧЕНИКИ). В запросе нельзя использовать DISTINCT.

\*/

**SELECT** Н\_ЛЮДИ.ИД,

ФАМИЛИЯ,

ИМЯ,

ОТЧЕСТВО

**FROM** Н\_ЛЮДИ

**LEFT JOIN** Н\_УЧЕНИКИ

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

**WHERE** Н\_УЧЕНИКИ.ИД **IS NULL**;

# Вывод

В ходе данной работы я познакомился с основными функциями языка SQL. Научился писать запросы, писать, отсеивать и сортировать полученные данные. В результате был изучен DML SQL, предназначенный для работы с данными, хранящимися внутри базы данных.