

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ
ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ «Санкт-Петербургский национальный исследовательский
университет ИТМО»

ФАКУЛЬТЕТ ПРОГРАММНОЙ ИНЖЕНЕРИИ И КОМПЬЮТЕРНОЙ
ТЕХНИКИ

ЛАБОРАТОРНАЯ РАБОТА №2

по дисциплине «Информационные системы и базы данных»

Вариант № 367101

Выполнил:

Студент группы Р3116

Билошицкий Михаил Владимирович

Преподаватель:

Горбунов Михаил Витальевич

Санкт-Петербург, 2023

Содержание

Содержание	2
Задание.....	3
Текст задания	4
Реализация запросов на SQL	5
Вывод.....	9

Задание

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
```

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

Введите вариант:

Внимание! У разных вариантов разный текст задания!

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):
а) Н_ТИПЫ_ВЕДОМОСТЕЙ.НАИМЕНОВАНИЕ > Ведомость.
б) Н_ВЕДОМОСТИ.ИД > 1250981.
Вид соединения: RIGHT JOIN.
2. Сделать запрос для получения атрибутов из указанных таблиц, применив фильтры по указанным условиям:
Таблицы: Н_ЛЮДИ, Н_ОБУЧЕНИЯ, Н_УЧЕНИКИ.
Вывести атрибуты: Н_ЛЮДИ.ИМЯ, Н_ОБУЧЕНИЯ.ЧЛВК_ИД, Н_УЧЕНИКИ.НАЧАЛО.
Фильтры: (AND)
а) Н_ЛЮДИ.ФАМИЛИЯ < Ёлкин.
б) Н_ОБУЧЕНИЯ.ЧЛВК_ИД > 112514.
с) Н_УЧЕНИКИ.ИД > 100410.
Вид соединения: RIGHT JOIN.
3. Вывести число студентов ФКТИУ, которые не имеют отчества.
Ответ должен содержать только одно число.
4. Выдать различные имена людей и число людей с каждой из этих имен, ограничив список именами, встречающимися ровно 10 раз 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 1101.
6. Получить список студентов, зачисленных ровно первого сентября 2012 года на первый курс очной или заочной формы обучения (специальность: 230101). В результат включить:
номер группы;
номер, фамилию, имя и отчество студента;
номер и состояние пункта приказа;
Для реализации использовать подзапрос с IN.
7. Вывести список студентов, имеющих одинаковые фамилии, но не совпадающие даты рождения.

Текст задания

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):
а) Н_ТИПЫ_ВЕДОМОСТЕЙ.НАИМЕНОВАНИЕ > Ведомость.
б) Н_ВЕДОМОСТИ.ИД > 1250981.
Вид соединения: RIGHT JOIN.
2. Сделать запрос для получения атрибутов из указанных таблиц, применив фильтры по указанным условиям:
Таблицы: Н_ЛЮДИ, Н_ОБУЧЕНИЯ, Н_УЧЕНИКИ.
Вывести атрибуты: Н_ЛЮДИ.ИМЯ, Н_ОБУЧЕНИЯ.ЧЛВК_ИД,
Н_УЧЕНИКИ.НАЧАЛО.
Фильтры: (AND)
а) Н_ЛЮДИ.ФАМИЛИЯ < Ёлкин.
б) Н_ОБУЧЕНИЯ.ЧЛВК_ИД > 112514.
с) Н_УЧЕНИКИ.ИД > 100410.
Вид соединения: RIGHT JOIN.
3. Вывести число студентов ФКТИУ, которые не имеет отчества.
Ответ должен содержать только одно число.
4. Выдать различные имена людей и число людей с каждой из этих имен, ограничив список именами, встречающимися ровно 10 раз 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 1101.
6. Получить список студентов, зачисленных ровно первого сентября 2012 года на первый курс очной или заочной формы обучения (специальность: 230101). В результат включить:
номер группы;
номер, фамилию, имя и отчество студента;
номер и состояние пункта приказа;
Для реализации использовать подзапрос с IN.
7. Вывести список студентов, имеющих одинаковые фамилии, но не совпадающие даты рождения.

Реализация запросов на SQL

```
-- Задание 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):  
-- а) Н_ТИПЫ_ВЕДОМОСТЕЙ.НАИМЕНОВАНИЕ > Ведомость.  
-- б) Н_ВЕДОМОСТИ.ИД > 1250981.  
-- Вид соединения: RIGHT JOIN.
```

```
SELECT Н_ТИПЫ_ВЕДОМОСТЕЙ.ИД, Н_ВЕДОМОСТИ.ЧЛВК_ИД  
FROM Н_ТИПЫ_ВЕДОМОСТЕЙ  
RIGHT JOIN Н_ВЕДОМОСТИ ON Н_ТИПЫ_ВЕДОМОСТЕЙ.ИД = Н_ВЕДОМОСТИ.ТВ_ИД  
WHERE Н_ТИПЫ_ВЕДОМОСТЕЙ.НАИМЕНОВАНИЕ > 'Ведомость' AND Н_ВЕДОМОСТИ.ИД > 1250981;
```

```
-- Задание 2.  
-- Сделать запрос для получения атрибутов из указанных таблиц, применив фильтры по  
указанным условиям:  
-- Таблицы: Н_ЛЮДИ, Н_ОБУЧЕНИЯ, Н_УЧЕНИКИ.  
-- Вывести атрибуты: Н_ЛЮДИ.ИМЯ, Н_ОБУЧЕНИЯ.ЧЛВК_ИД, Н_УЧЕНИКИ.НАЧАЛО.  
-- Фильтры: (AND)  
-- а) Н_ЛЮДИ.ФАМИЛИЯ < Ёлкин.  
-- б) Н_ОБУЧЕНИЯ.ЧЛВК_ИД > 112514.  
-- с) Н_УЧЕНИКИ.ИД > 100410.  
-- Вид соединения: RIGHT JOIN.
```

```
SELECT Н_ЛЮДИ.ИМЯ, Н_ОБУЧЕНИЯ.ЧЛВК_ИД, Н_УЧЕНИКИ.НАЧАЛО  
FROM Н_ОБУЧЕНИЯ  
RIGHT JOIN Н_ЛЮДИ ON Н_ОБУЧЕНИЯ.ЧЛВК_ИД = Н_ЛЮДИ.ИД  
RIGHT JOIN Н_УЧЕНИКИ ON Н_ОБУЧЕНИЯ.ЧЛВК_ИД = Н_УЧЕНИКИ.ЧЛВК_ИД  
WHERE Н_ЛЮДИ.ФАМИЛИЯ < 'Ёлкин' AND  
      Н_ОБУЧЕНИЯ.ЧЛВК_ИД > 112514 AND  
      Н_УЧЕНИКИ.ИД > 100410;
```

```
-- Задание 3.  
-- Вывести число студентов ФКТИУ, которые не имеет отчества.
```

-- Ответ должен содержать только одно число.

```
SELECT COUNT(*) FROM Н_люди
WHERE (Н_люди.ОТЧЕСТВО SIMILAR TO '[\.\s]*' OR Н_люди.ОТЧЕСТВО IS NULL) AND
      (Н_люди.ИД = ANY (SELECT Н_УЧЕНИКИ.ЧЛВК_ИД
                        FROM Н_УЧЕНИКИ
                        JOIN Н_ПЛАНЫ ON Н_УЧЕНИКИ.ПЛАН_ИД = Н_ПЛАНЫ.ИД
                        JOIN Н_ОТДЕЛЫ ON Н_ОТДЕЛЫ.ИД = Н_ПЛАНЫ.ОТД_ИД
                        WHERE
                            Н_ОТДЕЛЫ.ОТД_ИД = ANY (SELECT Н_ОТДЕЛЫ.ИД FROM Н_ОТДЕЛЫ WHERE
Н_ОТДЕЛЫ.КОРОТКОЕ_ИМЯ='КТиУ') OR
                            Н_ОТДЕЛЫ.ИД = (SELECT Н_ОТДЕЛЫ.ИД FROM Н_ОТДЕЛЫ WHERE
Н_ОТДЕЛЫ.КОРОТКОЕ_ИМЯ='КТиУ')
                        ));
```

-- Задание 4.

-- Выдать различные имена людей и число людей с каждой из этих имен,
-- ограничив список именами, встречающимися ровно 10 раз на ФКТИУ.
-- Для реализации использовать соединение таблиц.

```
SELECT Н_люди.ИМЯ as Имя, COUNT(*) as count FROM Н_люди
WHERE (Н_люди.ИД = ANY (SELECT Н_УЧЕНИКИ.ЧЛВК_ИД
                        FROM Н_УЧЕНИКИ
                        JOIN Н_ПЛАНЫ ON Н_УЧЕНИКИ.ПЛАН_ИД = Н_ПЛАНЫ.ИД
                        JOIN Н_ОТДЕЛЫ ON Н_ОТДЕЛЫ.ИД = Н_ПЛАНЫ.ОТД_ИД
                        WHERE
                            Н_ОТДЕЛЫ.ОТД_ИД = ANY (SELECT Н_ОТДЕЛЫ.ИД FROM Н_ОТДЕЛЫ WHERE
Н_ОТДЕЛЫ.КОРОТКОЕ_ИМЯ='КТиУ') OR
                            Н_ОТДЕЛЫ.ИД = (SELECT Н_ОТДЕЛЫ.ИД FROM Н_ОТДЕЛЫ WHERE
Н_ОТДЕЛЫ.КОРОТКОЕ_ИМЯ='КТиУ')
                        ))
GROUP BY Н_люди.ИМЯ HAVING COUNT(*) >= 10;
```

-- Задание 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 1101.

```
WITH AVG_GRADES AS (
```

```

SELECT
    Н_люди.ИД AS Номер,
    CONCAT(Н_люди.ИМЯ, ' ', Н_люди.ФАМИЛИЯ, ' ', Н_люди.ОТЧЕСТВО) AS ФИО,
    AVG(CAST(Н_ВЕДОМОСТИ.ОЦЕНКА AS INTEGER)) AS Ср_оценка
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_ВЕДОМОСТИ.ОЦЕНКА SIMILAR TO '\d+' AND
    Н_УЧЕНИКИ.ГРУППА = '4100'
GROUP BY Н_люди.ИД, Н_люди.ИМЯ, Н_люди.ФАМИЛИЯ, Н_люди.ОТЧЕСТВО)
SELECT * FROM AVG_GRADES
WHERE AVG_GRADES.Ср_оценка <= (
    SELECT MIN(CAST(Н_ВЕДОМОСТИ.ОЦЕНКА AS INTEGER))
    FROM Н_ВЕДОМОСТИ
    JOIN Н_люди ON Н_ВЕДОМОСТИ.ЧЛВК_ИД=Н_люди.ИД
    JOIN Н_УЧЕНИКИ ON Н_ВЕДОМОСТИ.ЧЛВК_ИД=Н_УЧЕНИКИ.ЧЛВК_ИД
    WHERE
        Н_ВЕДОМОСТИ.ОЦЕНКА SIMILAR TO '\d+' AND
        Н_УЧЕНИКИ.ГРУППА = '1101')
ORDER BY AVG_GRADES.Ср_оценка DESC;

```

```

-- Задание 6.
-- Получить список студентов, зачисленных ровно первого сентября 2012 года
-- на первый курс очной или заочной формы обучения (специальность: 230101). В
результат включить:
-- номер группы;
-- номер, фамилию, имя и отчество студента;
-- номер и состояние пункта приказа;
-- Для реализации использовать подзапрос с IN.

```

```

SELECT
    Н_УЧЕНИКИ.ГРУППА as НОМЕР_ГРУППЫ,
    Н_люди.ИД as НОМЕР,
    CONCAT(Н_люди.ФАМИЛИЯ, ' ', Н_люди.ИМЯ, ' ', Н_люди.ОТЧЕСТВО) as ФИО,
    Н_УЧЕНИКИ.ПРИЗНАК as СОСТОЯНИЕ
FROM Н_люди
JOIN Н_УЧЕНИКИ ON Н_люди.ИД = Н_УЧЕНИКИ.ЧЛВК_ИД
WHERE

```

```

H_УЧЕНИКИ.ПЛАН_ИД IN (
    SELECT H_ПЛАНЫ.ИД FROM H_ПЛАНЫ
    JOIN H_ФОРМЫ_ОБУЧЕНИЯ ON H_ПЛАНЫ.ФО_ИД = H_ФОРМЫ_ОБУЧЕНИЯ.ИД
    WHERE H_ПЛАНЫ.НАПС_ИД = (
        SELECT H_НАПРАВЛЕНИЯ_СПЕЦИАЛ.ИД FROM H_НАПРАВЛЕНИЯ_СПЕЦИАЛ
        WHERE H_НАПРАВЛЕНИЯ_СПЕЦИАЛ.НС_ИД = (
            SELECT H_НАПР_СПЕЦ.ИД FROM H_НАПР_СПЕЦ
            WHERE H_НАПР_СПЕЦ.КОД_НАПРСПЕЦ = '230101'
        )
    )
) AND
(H_ФОРМЫ_ОБУЧЕНИЯ.НАИМЕНОВАНИЕ = 'Очная' OR
H_ФОРМЫ_ОБУЧЕНИЯ.НАИМЕНОВАНИЕ = 'Заочная') AND
(DATE(H_УЧЕНИКИ.НАЧАЛО) = DATE('2012-09-01'))
);

```

-- Задание 7.

-- Вывести список студентов, имеющих одинаковые фамилии, но не совпадающие даты рождения.

```

SELECT DISTINCT
    s1.ИМЯ,
    s1.ФАМИЛИЯ,
    s1.ОТЧЕСТВО,
    s1.ДАТА_РОЖДЕНИЯ
FROM H_ЛЮДИ s1
JOIN H_ЛЮДИ s2 ON
    s1.ФАМИЛИЯ = s2.ФАМИЛИЯ AND
    DATE(s1.ДАТА_РОЖДЕНИЯ) != DATE(s2.ДАТА_РОЖДЕНИЯ)
ORDER BY 2;

```


Вывод

В ходе проведения лабораторной работы я научился правильно составлять запросы к базам данных на основе ее инфологической модели, закрепил навык знания языка запросов SQL, изучил как реализуются соединения таблиц, подзапросы, представления и последовательности. Полученные знания понадобятся нам в процессе дальнейшего обучения.