Санкт-Петербургский Национальный Исследовательский Университет ИТМО

Факультет Программной Инженерии и Компьютерной Техники



Вариант № 1341

Лабораторная работа №2

По дисциплине

Базы Данных

Выполнил:

Захра дарабзадех  
студент группы P3130.

Преподаватель:

Перцев Тимофей Сергеевич

Санкт-Петербург 2025 г.

1. **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.НАИМЕНОВАНИЕ, Н\_ВЕДОМОСТИ.ЧЛВК\_ИД.  
   Фильтры (AND):  
   a) Н\_ТИПЫ\_ВЕДОМОСТЕЙ.НАИМЕНОВАНИЕ < Ведомость.  
   b) Н\_ВЕДОМОСТИ.ИД < 39921.  
   Вид соединения: LEFT 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.ИД, Н\_ВЕДОМОСТИ.ИД, Н\_СЕССИЯ.ЧЛВК\_ИД.  
   Фильтры (AND):  
   a) Н\_ЛЮДИ.ИД = 152862.  
   b) Н\_ВЕДОМОСТИ.ЧЛВК\_ИД < 153285.  
   c) Н\_СЕССИЯ.ДАТА = 2012-01-25.  
   Вид соединения: INNER JOIN.
3. Составить запрос, который ответит на вопрос, есть ли среди студентов ФКТИУ те, кто старше 25 лет.
4. Найти группы, в которых в 2011 году было ровно 5 обучающихся студентов на ФКТИУ.  
   Для реализации использовать подзапрос.
5. Выведите таблицу со средними оценками студентов группы 4100 (Номер, ФИО, Ср\_оценка), у которых средняя оценка не меньше максимальной оценк(е|и) в группе 1101.
6. Получить список студентов, зачисленных после первого сентября 2012 года на первый курс очной формы обучения. В результат включить:  
   номер группы;  
   номер, фамилию, имя и отчество студента;  
   номер и состояние пункта приказа;  
   Для реализации использовать соединение таблиц.
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.
8. **Реализация запросов на SQL**

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

-- Таблицы: Н\_ТИПЫ\_ВЕДОМОСТЕЙ, Н\_ВЕДОМОСТИ.

-- Вывести атрибуты: Н\_ТИПЫ\_ВЕДОМОСТЕЙ.НАИМЕНОВАНИЕ, Н\_ВЕДОМОСТИ.ЧЛВК\_ИД.

-- Фильтры (AND):

-- a) Н\_ТИПЫ\_ВЕДОМОСТЕЙ.НАИМЕНОВАНИЕ < Ведомость.

-- b) Н\_ВЕДОМОСТИ.ИД < 39921.

-- Вид соединения: LEFT JOIN.

**SELECT** Н\_ТИПЫ\_ВЕДОМОСТЕЙ.НАИМЕНОВАНИЕ, Н\_ВЕДОМОСТИ.ЧЛВК\_ИД

**FROM** Н\_ТИПЫ\_ВЕДОМОСТЕЙ

**LEFT** **JOIN** Н\_ВЕДОМОСТИ **ON** Н\_ТИПЫ\_ВЕДОМОСТЕЙ.ИД = Н\_ВЕДОМОСТИ.ТВ\_ИД

**WHERE**

Н\_ТИПЫ\_ВЕДОМОСТЕЙ.НАИМЕНОВАНИЕ < 'Ведомость'

**AND** Н\_ВЕДОМОСТИ.ИД < **39921**;

-- 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.ИД, Н\_ВЕДОМОСТИ.ИД, Н\_СЕССИЯ.ЧЛВК\_ИД.

-- Фильтры (AND):

-- a) Н\_ЛЮДИ.ИД = 152862.

-- b) Н\_ВЕДОМОСТИ.ЧЛВК\_ИД < 153285.

-- c) Н\_СЕССИЯ.ДАТА = 2012-01-25.

-- Вид соединения: INNER JOIN.

**SELECT** Н\_ЛЮДИ.ИД, Н\_ВЕДОМОСТИ.ИД, Н\_СЕССИЯ.ЧЛВК\_ИД

**FROM** Н\_ЛЮДИ

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

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

**WHERE** Н\_ЛЮДИ.ИД = **152862**

**AND** Н\_ВЕДОМОСТИ.ЧЛВК\_ИД < **153285**

**AND** Н\_СЕССИЯ.ДАТА = '2012-01-25';

-- 3. Составить запрос, который ответит на вопрос, есть ли среди студентов ФКТИУ те, кто старше 25 лет.

**SELECT**

**CASE**

**WHEN** **COUNT** (\*) > **0** **THEN** 'Да, среди студентов ФКТИУ есть те, кому больше 25 лет'

**ELSE** 'Нет, среди студентов ФКТИУ нет тех, кому больше 25 лет'

**END** **AS** результат

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

**WHERE** Н\_ОТДЕЛЫ.КОРОТКОЕ\_ИМЯ = 'ФКТИУ'

**AND** **EXTRACT** (**YEAR** **FROM** AGE(Н\_ЛЮДИ.ДАТА\_РОЖДЕНИЯ)) > **25**;

-- 4. Найти группы, в которых в 2011 году было ровно 5 обучающихся студентов на ФКТИУ.

-- Для реализации использовать подзапрос.

**SELECT** ГРУППА

**FROM** (

**SELECT** Н\_УЧЕНИКИ.ГРУППА,

**COUNT**(Н\_УЧЕНИКИ.ЧЛВК\_ИД) **AS** КОЛИЧЕСТВО

**FROM** Н\_УЧЕНИКИ

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

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

**WHERE**

Н\_ОТДЕЛЫ.КОРОТКОЕ\_ИМЯ = 'ФКТИУ'

**AND** **EXTRACT** (**YEAR** **FROM** Н\_УЧЕНИКИ.НАЧАЛО) = **2011**

**GROUP** **BY**

Н\_УЧЕНИКИ.ГРУППА

) **AS** ГРУППЫ\_СТУДЕНТОВ

**WHERE** "КОЛИЧЕСТВО" = **5**;

-- 5. Выведите таблицу со средними оценками студентов группы 4100 (Номер, ФИО, Ср\_оценка),

-- у которых средняя оценка не меньше максимальной оценк(е|и) в группе 1101.

**SELECT**

Н\_УЧЕНИКИ.ГРУППА **AS** Номер,Н\_ЛЮДИ.ФАМИЛИЯ || ' ' || Н\_ЛЮДИ.ИМЯ || ' ' || COALESCE(Н\_ЛЮДИ.ОТЧЕСТВО, '') **AS** ФИО,

**AVG** (Н\_ВЕДОМОСТИ.ОЦЕНКА ::numeric) **AS** Ср\_оценка

**FROM** Н\_УЧЕНИКИ

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

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

**WHERE** Н\_УЧЕНИКИ.ГРУППА = '4100'

**GROUP** **BY** Н\_УЧЕНИКИ.ГРУППА, Н\_ЛЮДИ.ФАМИЛИЯ, Н\_ЛЮДИ.ИМЯ, Н\_ЛЮДИ.ОТЧЕСТВО

**HAVING**

**AVG** (Н\_ВЕДОМОСТИ.ОЦЕНКА ::numeric) >= (

**SELECT** **MAX** (Н\_ВЕДОМОСТИ.ОЦЕНКА::numeric)

**FROM** Н\_ВЕДОМОСТИ

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

**WHERE** Н\_УЧЕНИКИ.ГРУППА = '1101'

);

-- 6. Получить список студентов, зачисленных после первого сентября 2012 года на первый курс очной формы обучения.

-- В результат включить:

-- номер группы;

-- номер, фамилию, имя и отчество студента;

-- номер и состояние пункта приказа;

**SELECT**

Н\_УЧЕНИКИ.ГРУППА,

Н\_ЛЮДИ.ИД,

Н\_ЛЮДИ.ФАМИЛИЯ,

Н\_ЛЮДИ.ИМЯ,

Н\_ЛЮДИ.ОТЧЕСТВО,

Н\_УЧЕНИКИ.П\_ПРКОК\_ИД,

Н\_УЧЕНИКИ.СОСТОЯНИЕ

**FROM** Н\_УЧЕНИКИ

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

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

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

**WHERE** Н\_УЧЕНИКИ.ПРИЗНАК = 'зачисл'

**AND** Н\_УЧЕНИКИ.НАЧАЛО > '2012-09-01'

**AND** Н\_УЧЕНИКИ.КУРС = '1'

**AND** Н\_ФОРМЫ\_ОБУЧЕНИЯ.НАИМЕНОВАНИЕ = 'Очная';

-- 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.

**SELECT**

Н\_ЛЮДИ.ИД,

Н\_ЛЮДИ.ФАМИЛИЯ,

Н\_ЛЮДИ.ИМЯ,

Н\_ЛЮДИ.ОТЧЕСТВО

**FROM** Н\_ЛЮДИ

**WHERE** **NOT** **EXISTS** (

**SELECT** **1**

**FROM** Н\_УЧЕНИКИ

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

);

--количество студентов, которые на момент зачисление, были меньше 18 лет

SELECT COUNT(\*) AS количество\_студентов\_моложе\_18

FROM Н\_УЧЕНИКИ

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

WHERE AGE(Н\_УЧЕНИКИ.Дата\_Зачисления, Н\_ЛЮДИ.Дата\_Рождения) < INTERVAL '18 years';

1. **Вывод**

При выполнении лабораторной работы я познакомился с основными функциями языка SQL и диалекта PostgreSQL. Научился писать запросы, получать, агрегировать, отсеивать и сортировать полученные данные с использованием различных синтаксических конструкций языка. В результате был освоен язык DML SQL, предназначенный для работы с данными, хранящимися внутри базы данных.