Санкт-Петербургский национально исследовательский университет

информационных технологий, механики и оптики

Факультет программной инженерии и компьютерной техники



**Лабораторная работа №3 по предмету**

**“Информационные системы и базы данных”**

Вариант №313089

Выполнил: Балтабаев Дамир

Группа: P33121

Преподаватель: Шешуков Дмитрий Михайлович

г. Санкт-Петербург

2022

**Задание**

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

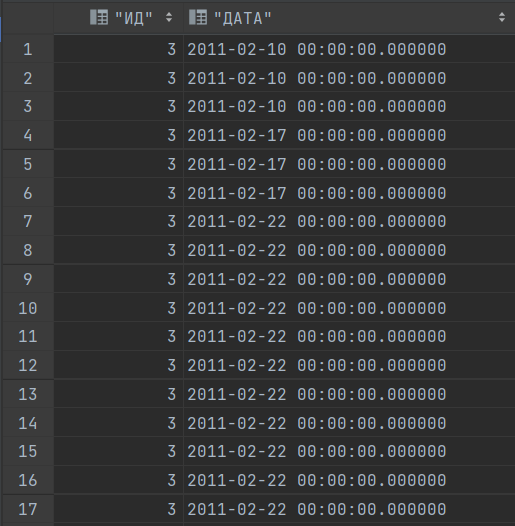
Составить запросы на языке SQL (пункты 1-7).

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.ДАТА.  
   Фильтры (AND):  
   a) Н\_ТИПЫ\_ВЕДОМОСТЕЙ.НАИМЕНОВАНИЕ = Перезачет.  
   b) Н\_ВЕДОМОСТИ.ДАТА > 1998-01-05.  
   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.ИД, Н\_ВЕДОМОСТИ.ДАТА, Н\_СЕССИЯ.ЧЛВК\_ИД.  
   Фильтры (AND):  
   a) Н\_ЛЮДИ.ОТЧЕСТВО = Георгиевич.  
   b) Н\_ВЕДОМОСТИ.ДАТА < 2022-06-08.  
   c) Н\_СЕССИЯ.ЧЛВК\_ИД < 151200.  
   Вид соединения: LEFT JOIN.
3. Вывести число студентов вечерней формы обучения, которые без ИНН.  
   Ответ должен содержать только одно число.
4. Выдать различные отчества людей и число людей с каждой из этих отчеств, ограничив список отчествами, встречающимися более 50 раз на на очной форме обучения.  
   Для реализации использовать подзапрос.
5. Выведите таблицу со средними оценками студентов группы 4100 (Номер, ФИО, Ср\_оценка), у которых средняя оценка не больше максимальной оценк(е|и) в группе 3100.
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 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.

**Запросы**

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.ДАТА.  
   Фильтры (AND):  
   a) Н\_ТИПЫ\_ВЕДОМОСТЕЙ.НАИМЕНОВАНИЕ = Перезачет.  
   b) Н\_ВЕДОМОСТИ.ДАТА > 1998-01-05.  
   Вид соединения: INNER JOIN.

SELECT "Н\_ТИПЫ\_ВЕДОМОСТЕЙ"."ИД", "Н\_ВЕДОМОСТИ"."ДАТА"  
FROM "Н\_ВЕДОМОСТИ"  
 INNER JOIN "Н\_ТИПЫ\_ВЕДОМОСТЕЙ" ON "Н\_ВЕДОМОСТИ"."ТВ\_ИД" = "Н\_ТИПЫ\_ВЕДОМОСТЕЙ"."ИД"  
WHERE ("Н\_ТИПЫ\_ВЕДОМОСТЕЙ"."НАИМЕНОВАНИЕ" = 'Перезачет'  
 AND "Н\_ВЕДОМОСТИ"."ДАТА" > '1998-01-05')



1. 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.ДАТА, Н\_СЕССИЯ.ЧЛВК\_ИД.  
   Фильтры (AND):  
   a) Н\_ЛЮДИ.ОТЧЕСТВО = Георгиевич.  
   b) Н\_ВЕДОМОСТИ.ДАТА < 2022-06-08.  
   c) Н\_СЕССИЯ.ЧЛВК\_ИД < 151200.  
   Вид соединения: LEFT JOIN.

SELECT "Н\_ЛЮДИ"."ИД", "Н\_ВЕДОМОСТИ"."ДАТА", "Н\_СЕССИЯ"."ЧЛВК\_ИД"  
FROM "Н\_ЛЮДИ"  
 LEFT JOIN "Н\_ВЕДОМОСТИ" ON "Н\_ВЕДОМОСТИ"."ЧЛВК\_ИД" = "Н\_ЛЮДИ"."ИД"  
 LEFT JOIN "Н\_СЕССИЯ" ON "Н\_СЕССИЯ"."ЧЛВК\_ИД" = "Н\_ЛЮДИ"."ИД"  
WHERE ("Н\_ЛЮДИ"."ОТЧЕСТВО" = 'Георгиевич' AND  
 "Н\_ВЕДОМОСТИ"."ДАТА" < '2022-06-08' AND  
 "Н\_СЕССИЯ"."ЧЛВК\_ИД" < 151200)

Изображение выглядит как текст

Автоматически созданное описание

1. Вывести число студентов вечерней формы обучения, которые без ИНН.

Ответ должен содержать только одно число.

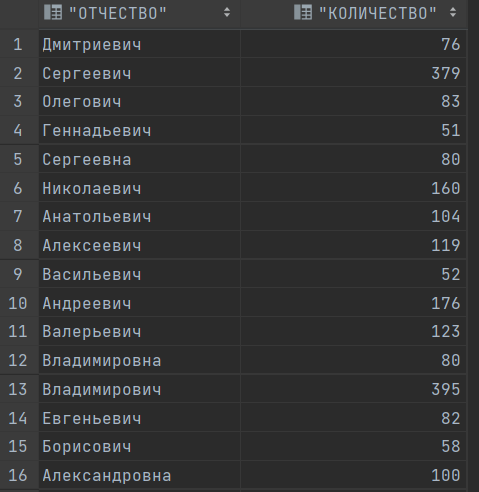
SELECT *COUNT*(DISTINCT "Н\_ЛЮДИ"."ИД")  
FROM "Н\_ЛЮДИ"  
 INNER JOIN "Н\_УЧЕНИКИ" ON "Н\_ЛЮДИ"."ИД" = "Н\_УЧЕНИКИ"."ЧЛВК\_ИД"  
 INNER JOIN "Н\_ПЛАНЫ" ON "Н\_УЧЕНИКИ"."ПЛАН\_ИД" = "Н\_ПЛАНЫ"."ИД"  
 INNER JOIN "Н\_ФОРМЫ\_ОБУЧЕНИЯ" ON "Н\_ФОРМЫ\_ОБУЧЕНИЯ"."ИД" = "Н\_ПЛАНЫ"."ФО\_ИД"  
WHERE ("Н\_ФОРМЫ\_ОБУЧЕНИЯ"."НАИМЕНОВАНИЕ" = 'Очно-заочная(вечерняя)' AND  
 "Н\_ЛЮДИ"."ИНН" IS NULL)

Изображение выглядит как текст

Автоматически созданное описание

1. Выдать различные отчества людей и число людей с каждой из этих отчеств, ограничив список отчествами, встречающимися более 50 раз на очной форме обучения.  
   Для реализации использовать подзапрос.

SELECT "Н\_ЛЮДИ"."ОТЧЕСТВО", *COUNT*(\*) AS "КОЛИЧЕСТВО"  
FROM "Н\_ЛЮДИ"  
WHERE ("Н\_ЛЮДИ"."ИД" IN (SELECT "Н\_УЧЕНИКИ"."ЧЛВК\_ИД"  
 FROM "Н\_УЧЕНИКИ"  
 WHERE "Н\_УЧЕНИКИ"."ПЛАН\_ИД" IN (SELECT "Н\_ПЛАНЫ"."ИД"  
 FROM "Н\_ПЛАНЫ"  
 WHERE "Н\_ПЛАНЫ"."ФО\_ИД" IN (SELECT "Н\_ФОРМЫ\_ОБУЧЕНИЯ"."ИД"  
 FROM "Н\_ФОРМЫ\_ОБУЧЕНИЯ"  
 WHERE "Н\_ФОРМЫ\_ОБУЧЕНИЯ"."НАИМЕНОВАНИЕ" = 'Очная')))  
 AND "Н\_ЛЮДИ"."ОТЧЕСТВО" <> '.')  
GROUP BY "Н\_ЛЮДИ"."ОТЧЕСТВО"  
HAVING *COUNT*(\*) > 50



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

SELECT "Н\_УЧЕНИКИ"."ИД", "Н\_ЛЮДИ"."ФАМИЛИЯ", "Н\_ЛЮДИ"."ИМЯ","Н\_ЛЮДИ"."ОТЧЕСТВО", *AVG*(*CAST*("Н\_ВЕДОМОСТИ"."ОЦЕНКА" AS INT)) as "СРЕДНЯЯ ОЦЕНКА"  
FROM "Н\_ЛЮДИ"  
 JOIN "Н\_УЧЕНИКИ" ON "Н\_ЛЮДИ"."ИД" = "Н\_УЧЕНИКИ"."ЧЛВК\_ИД"  
 JOIN "Н\_ВЕДОМОСТИ" ON "Н\_ЛЮДИ"."ИД" = "Н\_ВЕДОМОСТИ"."ЧЛВК\_ИД"  
WHERE ("Н\_УЧЕНИКИ"."ГРУППА" = '4100' AND  
 "Н\_ВЕДОМОСТИ"."ОЦЕНКА" <> 'осв' AND  
 "Н\_ВЕДОМОСТИ"."ОЦЕНКА" <> 'неявка' AND  
 "Н\_ВЕДОМОСТИ"."ОЦЕНКА" <> 'незач' AND  
 "Н\_ВЕДОМОСТИ"."ОЦЕНКА" <> 'зачет')  
GROUP BY "Н\_УЧЕНИКИ"."ИД", "Н\_ЛЮДИ"."ФАМИЛИЯ","Н\_ЛЮДИ"."ИМЯ","Н\_ЛЮДИ"."ОТЧЕСТВО"  
HAVING (*AVG*(*CAST*("Н\_ВЕДОМОСТИ"."ОЦЕНКА" AS INT)) < (SELECT *MAX*(*CAST*("Н\_ВЕДОМОСТИ"."ОЦЕНКА" AS INT))  
 FROM "Н\_ВЕДОМОСТИ"  
 JOIN "Н\_УЧЕНИКИ" ON "Н\_ВЕДОМОСТИ"."ЧЛВК\_ИД" = "Н\_УЧЕНИКИ"."ЧЛВК\_ИД"  
 WHERE ("Н\_УЧЕНИКИ"."ГРУППА" = '3100' AND  
 "Н\_ВЕДОМОСТИ"."ОЦЕНКА" <> 'осв' AND  
 "Н\_ВЕДОМОСТИ"."ОЦЕНКА" <> 'неявка' AND  
 "Н\_ВЕДОМОСТИ"."ОЦЕНКА" <> 'незач' AND  
 "Н\_ВЕДОМОСТИ"."ОЦЕНКА" <> 'зачет')))

Изображение выглядит как текст

Автоматически созданное описание

1. 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 соединение таблиц.

SELECT "Н\_УЧЕНИКИ"."ГРУППА",  
 "Н\_УЧЕНИКИ"."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\_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"  
 JOIN "Н\_НАПРАВЛЕНИЯ\_СПЕЦИАЛ" ON "Н\_ПЛАНЫ"."НАПС\_ИД" = "Н\_НАПРАВЛЕНИЯ\_СПЕЦИАЛ"."ИД"  
 JOIN "Н\_НАПР\_СПЕЦ" ON "Н\_НАПРАВЛЕНИЯ\_СПЕЦИАЛ"."НС\_ИД" = "Н\_НАПР\_СПЕЦ"."ИД"  
WHERE (("Н\_ФОРМЫ\_ОБУЧЕНИЯ"."НАИМЕНОВАНИЕ" = 'Очная' OR  
 "Н\_ФОРМЫ\_ОБУЧЕНИЯ"."НАИМЕНОВАНИЕ" = 'Очно-заочная(вечерняя)' OR  
 "Н\_ФОРМЫ\_ОБУЧЕНИЯ"."НАИМЕНОВАНИЕ" = 'Заочная')  
 AND "Н\_ПЛАНЫ"."КУРС" = 1  
 AND "Н\_УЧЕНИКИ"."НАЧАЛО" = '2012-09-01'  
 AND "Н\_НАПР\_СПЕЦ"."КОД\_НАПРСПЕЦ" = '230101')

Изображение выглядит как текст

Автоматически созданное описание

1. 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 "Н\_ЛЮДИ"."ИД" NOT IN (SELECT "Н\_УЧЕНИКИ"."ЧЛВК\_ИД"  
 FROM "Н\_УЧЕНИКИ")  
GROUP BY "Н\_ЛЮДИ"."ИД", "Н\_ЛЮДИ"."ФАМИЛИЯ", "Н\_ЛЮДИ"."ИМЯ", "Н\_ЛЮДИ"."ОТЧЕСТВО"

Изображение выглядит как текст

Автоматически созданное описание

**Вывод**

В ходе выполнения данной лабораторной работы я реализовал множество SQL запросов, поработав с DML. Также познакомился с командами соединения таблиц, такими как: INNER JOIN, LEFT JOIN, RIGHT JOIN. Поработал с такими командами как GROUP BY и HAVING, а также использовал подзапросы.