

**Университет ИТМО**

**Факультет ПИиКТ**

Лабораторная работа №3 по дисциплине  
“Информационные системы и базы данных”

(3ий курс бакалавриата ФПИиКТ)

**Студент:**

Иванов Евгений Дмитриевич

Группа Р33111

**Преподаватель:**

Харитоновна Анастасия Евгеньевна

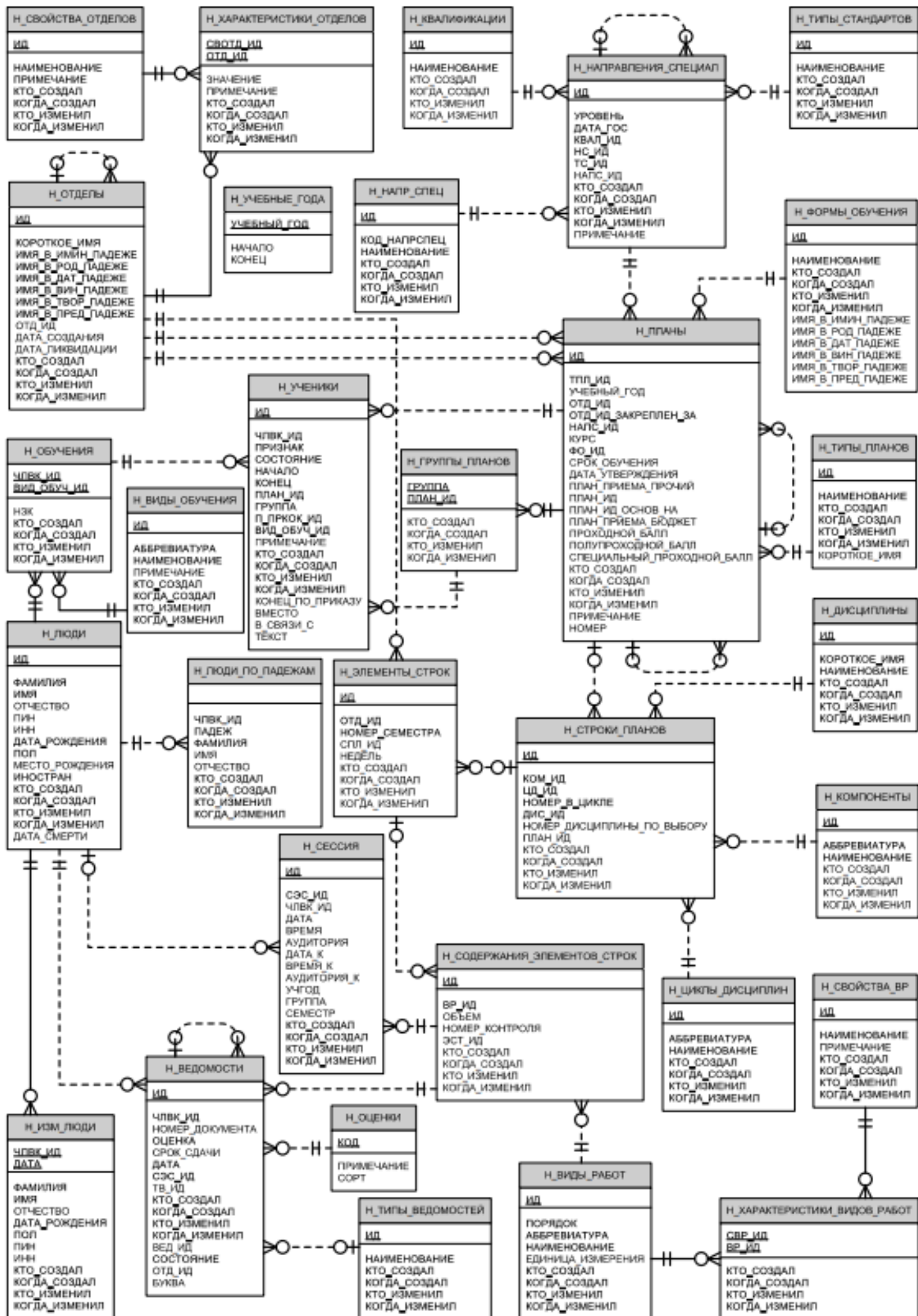
## Вариант: 1523

### Условие варианта:

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):  
а) Н\_ТИПЫ\_ВЕДОМОСТЕЙ.ИД < 3.  
б) Н\_ВЕДОМОСТИ.ДАТА < 2022-06-08.  
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.ИМЯ, Н\_ВЕДОМОСТИ.ЧЛВК\_ИД, Н\_СЕССИЯ.ДАТА.  
Фильтры (AND):  
а) Н\_ЛЮДИ.ОТЧЕСТВО < Владимирович.  
б) Н\_ВЕДОМОСТИ.ЧЛВК\_ИД > 153285.  
с) Н\_СЕССИЯ.УЧГОД > 2003/2004.  
Вид соединения: INNER JOIN.
3. Вывести число фамилий и отчеств без учета повторений.  
При составлении запроса нельзя использовать DISTINCT.
4. Найти группы, в которых в 2011 году было менее 5 обучающихся студентов на кафедре вычислительной техники.  
Для реализации использовать подзапрос.
5. Выведите таблицу со средним возрастом студентов во всех группах (Группа, Средний возраст), где средний возраст меньше максимального возраста в группе 1101.
6. Получить список студентов, зачисленных до первого сентября 2012 года на первый курс 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 на ФКТИУ троечников.

### Инфологическая модель “Учебный процесс”:



## Запросы:

```
-- Тестовый запрос
select *
from "Н_ВЕДОМОСТИ";

-- 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.НАИМЕНОВАНИЕ,
Н_ВЕДОМОСТИ.ИД.
-- Фильтры (AND):
-- а) Н_ТИПЫ_ВЕДОМОСТЕЙ.ИД < 3.
-- б) Н_ВЕДОМОСТИ.ДАТА < 2022-06-08.
-- Вид соединения: RIGHT JOIN.
SELECT "НАИМЕНОВАНИЕ", НВ."ИД"
FROM "Н_ТИПЫ_ВЕДОМОСТЕЙ" AS ТВ
      RIGHT OUTER JOIN "Н_ВЕДОМОСТИ" AS НВ ON ТВ."ИД" =
НВ."ТВ_ИД"
WHERE ТВ."ИД" < 3
      AND НВ."ДАТА" < DATE '2022-06-08';

-- Задание №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.ИМЯ, Н_ВЕДОМОСТИ.ЧЛВК_ИД,
Н_СЕССИЯ.ДАТА.
-- Фильтры (AND):
-- а) Н_ЛЮДИ.ОТЧЕСТВО < Владимирович.
-- б) Н_ВЕДОМОСТИ.ЧЛВК_ИД > 153285.
-- с) Н_СЕССИЯ.УЧГОД > 2003/2004.
-- Вид соединения: INNER JOIN.
SELECT ЛЮДИ."ИМЯ", НВ."ЧЛВК_ИД", СЕССИЯ."ДАТА",
СЕССИЯ."УЧГОД"
FROM "Н_ЛЮДИ" AS ЛЮДИ
      INNER JOIN "Н_ВЕДОМОСТИ" AS НВ ON ЛЮДИ."ИД" =
НВ."ЧЛВК_ИД"
```

```
INNER JOIN "Н_СЕССИЯ" AS СЕССИЯ ON ЛЮДИ."ИД" =
СЕССИЯ."ЧЛВК_ИД"
WHERE ЛЮДИ."ОТЧЕСТВО" < 'Владимирович' -- сравнение стандартное,
посимвольное, не по длине слова.
AND НВ."ЧЛВК_ИД" > 153285
AND СЕССИЯ."УЧГОД" > '2003/2004';
-- будет нормально работать, так как сравнивает посимвольно, а года
всегда имеют такой формат.
-- ORDER BY НВ."ЧЛВК_ИД" DESC
-- Запрос не даст результатов, так как максимальный ЧЛВК_ИД =
134262, а следует искать > 153285
```

```
-- Задание №3
-- Вывести число фамилий и отчеств без учета повторений.
-- При составлении запроса нельзя использовать DISTINCT.
```

```
SELECT count(*) AS last_name_count
FROM (SELECT count("ФАМИЛИЯ") AS last_name
      FROM "Н_ЛЮДИ" AS ЛЮДИ
      GROUP BY "ФАМИЛИЯ") AS ЛФ;
SELECT count(*) AS father_name_count
FROM (SELECT count("ОТЧЕСТВО") AS father_name
      FROM "Н_ЛЮДИ" AS ЛЮДИ
      GROUP BY "ОТЧЕСТВО") AS ЛФ;
```

```
-- Задание №4
-- Найти группы, в которых в 2011 году было менее 5 обучающихся
студентов на кафедре вычислительной техники.
-- Для реализации использовать подзапрос.
```

```
SELECT "ГРУППА"
FROM (SELECT count(*) as count_st, "ГРУППА"
      FROM "Н_УЧЕНИКИ" AS ST
      WHERE ST."НАЧАЛО" >= TIMESTAMP '2011-09-01 00:00:00.000000'
            AND ST."КОНЕЦ" < TIMESTAMP '2012-09-01 00:00:00.000000'
            GROUP BY "ГРУППА") AS inner_sql
WHERE count_st < 5;
```

```
-- Задание №5
-- Выведите таблицу со средним возрастом студентов во всех группах
```

```

(Группа, Средний возраст),
-- где средний возраст меньше максимального возраста в группе 1101.
SELECT ST."ГРУППА", AVG(date_part('year',
age(PEOP."ДАТА_РОЖДЕНИЯ"))) as avg_age
FROM "Н_УЧЕНИКИ" AS ST
    INNER JOIN "Н_ЛЮДИ" AS PEOP on ST."ЧЛВК_ИД" = PEOP."ИД"
GROUP BY ST."ГРУППА"
HAVING AVG(date_part('year', age(PEOP."ДАТА_РОЖДЕНИЯ"))) <
    (SELECT MAX(date_part('year', age(PEOP."ДАТА_РОЖДЕНИЯ")))
    FROM "Н_УЧЕНИКИ" AS ST
        INNER JOIN "Н_ЛЮДИ" AS PEOP on ST."ЧЛВК_ИД" =
PEOP."ИД"
    WHERE ST."ГРУППА" = '1101');
-- результат данного подзапроса по поиску максимального возраста в
группе 1101 = 38

```

-- Задание №6

-- Получить список студентов, зачисленных до первого сентября 2012 года на первый курс 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 sp_type."НАИМЕНОВАНИЕ",
    "УЧЕБНЫЙ_ГОД",
    st."ГРУППА",
    peop."ФАМИЛИЯ",
    peop."ИМЯ",
    peop."ОТЧЕСТВО",
    st."НАЧАЛО",
    st."КОНЕЦ"
FROM "Н_НАПРАВЛЕНИЯ_СПЕЦИАЛ" as sp
    INNER JOIN "Н_НАПР_СПЕЦ" sp_type on sp."НС_ИД" =
sp_type."ИД"
    INNER JOIN "Н_ПЛАНЫ" as plans on plans."НАПС_ИД" =
sp."ИД"
    INNER JOIN "Н_УЧЕНИКИ" as st on plans."ИД" =
st."ПЛАН_ИД"
    INNER JOIN "Н_ЛЮДИ" as peop on peop."ИД" = st."ЧЛВК_ИД"

```

```

WHERE EXISTS(
    SELECT *
    FROM "Н_НАПР_СПЕЦ" as sttt
    WHERE sttt."ИД" = 741
)
AND sp_type."ИД" = 741 -- Направление программной инженерии
AND plans."КУРС" = 1 -- Курс зачисления
AND st."НАЧАЛО" < '2012-09-01';

-- Задание №7
-- Сформировать запрос для получения числа на ФКТИУ троечников.
SELECT count(*)
FROM (SELECT peop."ИД", count(*)
      FROM "Н_ВЕДОМОСТИ" AS ved
           INNER JOIN "Н_ЛЮДИ" AS peop on ved."ЧЛВК_ИД" =
peop."ИД"
           INNER JOIN "Н_УЧЕНИКИ" AS st on st."ЧЛВК_ИД" =
peop."ИД"
           INNER JOIN "Н_ПЛАНЫ" AS plans on plans."ИД" =
st."ПЛАН_ИД"
           INNER JOIN "Н_НАПРАВЛЕНИЯ_СПЕЦИАЛ" as sp on
sp."ИД" = plans."НАПС_ИД"
           INNER JOIN "Н_НАПР_СПЕЦ" as sp_type on sp."НС_ИД"
= sp_type."ИД"
      WHERE ved."ОЦЕНКА" = '3'
      AND sp_type."ИД" IN (741, 772, 317, 316, 16)
      GROUP BY peop."ИД") AS Ис;

SELECT peop."ИД", count(*) as number_of_3
FROM "Н_ВЕДОМОСТИ" as ved
      INNER JOIN "Н_ЛЮДИ" as peop on ved."ЧЛВК_ИД" = peop."ИД"
      INNER JOIN "Н_УЧЕНИКИ" as st on st."ЧЛВК_ИД" = peop."ИД"
      INNER JOIN "Н_ПЛАНЫ" as plans on plans."ИД" =
st."ПЛАН_ИД"
      INNER JOIN "Н_НАПРАВЛЕНИЯ_СПЕЦИАЛ" as sp on sp."ИД" =
plans."НАПС_ИД"
      INNER JOIN "Н_НАПР_СПЕЦ" as sp_type on sp."НС_ИД" =
sp_type."ИД"
WHERE ved."ОЦЕНКА" = '3'
GROUP BY peop."ИД";

```

**Вывод:** Поработал с разными запросами к большой связной базе(сделанной с малым количеством комментариев). Узнал новые операторы SQL.