

Университет ИТМО
Факультет программной инженерии и компьютерной техники

Информационные системы и базы данных

Лабораторная работа №2

Выполнила: Машина Е.А.

Группа Р33113

Преподаватель: Николаев В.В.

Санкт-Петербург
2020 г.

Задание

Please, enter your variant: 3685

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:
Таблицы: Н_ЛЮДИ, Н_СЕССИЯ.
Вывести атрибуты: Н_ЛЮДИ.ИМЯ, Н_СЕССИЯ.ЧЛВК_ИД.
Фильтры (AND):
а) Н_ЛЮДИ.ИД < 100865.
б) Н_СЕССИЯ.ЧЛВК_ИД > 100622.
Вид соединения: INNER 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.ОТЧЕСТВО, Н_ОБУЧЕНИЯ.ЧЛВК_ИД, Н_УЧЕНИКИ.ГРУППА.
Фильтры: (AND)
а) Н_ЛЮДИ.ОТЧЕСТВО < Владимирович.
б) Н_ОБУЧЕНИЯ.ЧЛВК_ИД < 113409.
с) Н_УЧЕНИКИ.ГРУППА > 5100.
Вид соединения: INNER JOIN.
- 3. Вывести число студентов ФКТИУ, которые не имеет отчества.
Ответ должен содержать только одно число.
- 4. Найти группы, в которых в 2011 году было более 10 обучающихся студентов на ФКТИУ.
Для реализации использовать соединение таблиц.
- 5. Выведите таблицу со средними оценками студентов группы 4100 (Номер, ФИО, Ср_оценка), у которых средняя оценка больше минимальной оценк(е)и в группе 1101.
- 6. Получить список студентов, зачисленных ровно 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 подзапрос с EXISTS.
- 7. Вывести список людей, не являющихся или не являвшихся студентами ФКТИУ (данные, о которых отсутствуют в таблице Н_УЧЕНИКИ). В запросе нельзя использовать DISTINCT.

Выполнение

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

Таблицы: Н_ЛЮДИ, Н_СЕССИЯ.

Вывести атрибуты: Н_ЛЮДИ.ИМЯ, Н_СЕССИЯ.ЧЛВК_ИД.

Фильтры (AND):

а) Н_ЛЮДИ.ИД < 100865.

б) Н_СЕССИЯ.ЧЛВК_ИД > 100622.

Вид соединения: INNER JOIN.

```
SELECT "Н_ЛЮДИ"."ИМЯ",  
       "Н_СЕССИЯ"."ЧЛВК_ИД"  
FROM "Н_ЛЮДИ"  
      INNER JOIN "Н_СЕССИЯ" ON "Н_ЛЮДИ"."ИД" = "Н_СЕССИЯ"."ЧЛВК_ИД"  
WHERE "Н_ЛЮДИ"."ИД" < 100865 AND  
      "Н_СЕССИЯ"."ЧЛВК_ИД" > 100622;
```

	"ИМЯ"	"ЧЛВК_ИД"			
1	Юрий	100657			
2	Юрий	100657			
3	Александр	100633			
4	Александр	100633	479	Александр	100627
5	Сергей	100656	480	Александр	100666
6	Юрий	100657	481	Игорь	100628

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.ОТЧЕСТВО, Н_ОБУЧЕНИЯ.ЧЛВК_ИД, Н_УЧЕНИКИ.ГРУППА.

Фильтры: (AND)

а) Н_ЛЮДИ.ОТЧЕСТВО < Владимирович.

б) Н_ОБУЧЕНИЯ.ЧЛВК_ИД < 113409.

с) Н_УЧЕНИКИ.ГРУППА > 5100.

Вид соединения: INNER JOIN.

```
SELECT "Н_люди"."ОТЧЕСТВО",
       "Н_обучения"."ЧЛВК_ИД",
       "Н_ученики"."ГРУППА"
FROM "Н_люди"
      INNER JOIN "Н_обучения" ON "Н_обучения"."ЧЛВК_ИД" = "Н_люди"."ИД"
      INNER JOIN "Н_ученики" ON "Н_ученики"."ЧЛВК_ИД" = "Н_люди"."ИД"
WHERE "Н_люди"."ОТЧЕСТВО" < 'Владимирович' AND
      "Н_обучения"."ЧЛВК_ИД" < 113409 AND
      "Н_ученики"."ГРУППА" > '5100';
```

	"ОТЧЕСТВО"	"ЧЛВК_ИД"	"ГРУППА"
1	Александровна	112977	595
2	Александровна	112977	595
3	Александрович	110136	553
4	Александрович	110136	653
331	Борисович	111831	550
332	Борисович	111831	5895
333	Борисович	111831	550

3. Вывести число студентов ФКТИУ, которые не имеет отчества.

Ответ должен содержать только одно число.

```
SELECT COUNT("Н_ученики"."ИД")
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" AND
                        "Н_отделы"."КОРОТКОЕ_ИМЯ" = 'КТИУ'
      JOIN "Н_люди" ON "Н_ученики"."ЧЛВК_ИД" = "Н_люди"."ИД" AND
                    ("Н_люди"."ОТЧЕСТВО" is null or "Н_люди"."ОТЧЕСТВО" =
'.');
```

	count
1	949

4. Найти группы, в которых в 2011 году было более 10 обучающихся студентов на ФКТИУ.

Для реализации использовать соединение таблиц.

```
SELECT "Н_группы_планов"."ГРУППА"
FROM "Н_группы_планов"
      JOIN "Н_планы" ON "Н_группы_планов"."ПЛАН_ИД" = "Н_планы"."ИД"
      JOIN "Н_ученики" ON "Н_группы_планов"."ГРУППА" = "Н_ученики"."ГРУППА" AND
                        "НАЧАЛО" < '01.01.2012' AND
                        "КОНЕЦ" >= '01.01.2011'
WHERE (SELECT "Н_отделы"."КОРОТКОЕ_ИМЯ"
      FROM "Н_отделы"
      WHERE "Н_отделы"."КОРОТКОЕ_ИМЯ" = 'КТИУ') = 'КТИУ'
```

```
GROUP BY "Н_ГРУППЫ_ПЛАНОВ"."ГРУППА"
HAVING COUNT(*) < 10;
```

	"ГРУППА"		
1	5121		
2	218		
3	439	20	6109
4	5109	21	438
5	217	22	3895

5. Выведите таблицу со средними оценками студентов группы 4100 (Номер, ФИО, Ср_оценка), у которых средняя оценка больше минимальной оценк(е|и) в группе 1101.

```
SELECT "Н_ЛЮДИ"."ИД",
       "ФАМИЛИЯ",
       "ИМЯ",
       "ОТЧЕСТВО",
       AVG(CAST("ОЦЕНКА" AS NUMERIC)) AS "СРЕДНЯЯ_ОЦЕНКА"
FROM "Н_ЛЮДИ"
     JOIN "Н_УЧЕНИКИ" ON "Н_УЧЕНИКИ"."ЧЛВК_ИД" = "Н_ЛЮДИ"."ИД" AND
                          "Н_УЧЕНИКИ"."ГРУППА" = '4100'
     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 MIN(EXP.MARK)
                                           FROM
                                           (
                                           SELECT AVG(CAST("ОЦЕНКА" AS
NUMERIC)) AS MARK
                                           FROM "Н_УЧЕНИКИ"
                                           JOIN "Н_ВЕДОМОСТИ" ON
"Н_УЧЕНИКИ"."ГРУППА" = '1101' AND
"Н_УЧЕНИКИ"."ЧЛВК_ИД" = "Н_ВЕДОМОСТИ"."ЧЛВК_ИД" AND
"ОЦЕНКА" NOT IN('осв', 'неявка', 'зачет', 'незач')
                                           ) EXP);
```

Запросу соответствует 69 ответов

6. Получить список студентов, зачисленных ровно 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 подзапрос с EXISTS.

```
SELECT "ГРУППА",
       "ЧЛВК_ИД",
       "ФАМИЛИЯ",
       "ИМЯ",
       "ОТЧЕСТВО",
       "В_СВЯЗИ_С"
FROM "Н_УЧЕНИКИ"
     JOIN "Н_ПЛАНЫ" ON "Н_УЧЕНИКИ"."ПЛАН_ИД" = "Н_ПЛАНЫ"."ИД"
     JOIN "Н_ЛЮДИ" ON "Н_УЧЕНИКИ"."ЧЛВК_ИД" = "Н_ЛЮДИ"."ИД"
```

```
WHERE EXISTS(SELECT "ИД"
              FROM "Н_ФОРМЫ_ОБУЧЕНИЯ"
              WHERE "НАИМЕНОВАНИЕ" = 'Очная' AND
                    "НАЧАЛО" = '01.09.2012');
```

Запросу отвечает 0 человек

7. Вывести список людей, не являющихся или не являвшихся студентами ФКТИУ (данные, о которых отсутствуют в таблице Н_УЧЕНИКИ). В запросе нельзя использовать DISTINCT.

```
SELECT "ИД",
       "ФАМИЛИЯ",
       "ИМЯ",
       "ОТЧЕСТВО"
FROM "Н_ЛЮДИ" AS people
WHERE NOT EXISTS(SELECT *
                  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"
                  AND
                  "Н_ОТДЕЛЫ"."КОРОТКОЕ_ИМЯ" = 'КТИУ'
                  JOIN "Н_ЛЮДИ" ON "Н_УЧЕНИКИ"."ЧЛВК_ИД" = "Н_ЛЮДИ"."ИД"
                  WHERE "Н_УЧЕНИКИ"."ЧЛВК_ИД" = people."ИД");
```

	"ИД"	"ФАМИЛИЯ"	"ИМЯ"	"ОТЧЕСТВО"
1	114093	Попович	Игорь	Васильевич
2	114992	Жилин	Роман	Алексеевич
3	114993	Зубович	Иван	Александрович
4	114995	Косарева	Елена	Владимировна
498	116138	Гречишкин	Дмитрий	Олегович
499	116184	Лебедев	Сергей	Владимирович
500	156964	Кожевников	Иван	Андреевич

Вывод

В ходе выполнения данной лабораторной работы я более подробно изучила язык SQL, совершила запросы к нескольким таблицам, путем склеивания таблиц, познакомилась с понятием подзапроса и использовала их в запросах.

Проделав все запросы и изучив семь вариантов склеивания таблиц я пришла к выводу, что выбор правильного варианта склеивания позволит избежать лишних подзапросов. Также узнала о том, что по умолчанию в PostgreSQL join выполняется, как inner join.