

**/ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
“НАЦИОНАЛЬНЫЙ ИСЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО”**

Факультет Программной Инженерии и Компьютерной Техники

Дисциплина: «Базы Данных»

ОТЧЁТ

по лабораторной работе №4

Вариант 10094

Выполнил:

Студент группы Р3110

Федоров Евгений Константинович

Проверил:

Бострикова Дарья Константиновна

Санкт-Петербург 2024

Оглавление

Задание: 3

Реализация запросов: 4

Индексы 8

Планы 9

Планы для первого запроса 9

Планы для второго запроса: 11

Задание:

Составить запросы на языке SQL (пункты 1-2).

Для каждого запроса предложить индексы, добавление которых уменьшит время выполнения запроса (указать таблицы/атрибуты, для которых нужно добавить индексы, написать тип индекса; объяснить, почему добавление индекса будет полезным для данного запроса).

Для запросов 1-2 необходимо составить возможные планы выполнения запросов. Планы составляются на основании предположения, что в таблицах отсутствуют индексы. Из составленных планов необходимо выбрать оптимальный и объяснить свой выбор.

Изменяются ли планы при добавлении индекса и как?

Для запросов 1-2 необходимо добавить в отчет вывод команды EXPLAIN ANALYZE [запрос]

Подробные ответы на все вышеперечисленные вопросы должны присутствовать в отчете (планы выполнения запросов должны быть нарисованы, ответы на вопросы - представлены в текстовом виде).

1. Сделать запрос для получения атрибутов из указанных таблиц, применив фильтры по указанным условиям:
Таблицы: Н_ЛЮДИ, Н_СЕССИЯ.
Вывести атрибуты: Н_ЛЮДИ.ИД, Н_СЕССИЯ.ЧЛВК_ИД.
Фильтры (AND):
 - а) Н_ЛЮДИ.ОТЧЕСТВО < Александрович.
 - б) Н_СЕССИЯ.ЧЛВК_ИД > 106059.
 - в) Н_СЕССИЯ.ЧЛВК_ИД < 105948q.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.ФАМИЛИЯ, Н_ВЕДОМОСТИ.ИД, Н_СЕССИЯ.ИД.
Фильтры (AND):
 - а) Н_ЛЮДИ.ИД > 100865.
 - б) Н_ВЕДОМОСТИ.Дата < 1998-01-05.Вид соединения: INNER JOIN.

Реализация запросов:

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

Таблицы: Н_ЛЮДИ, Н_СЕССИЯ.

Вывести атрибуты: Н_ЛЮДИ.ИД, Н_СЕССИЯ.ЧЛВК_ИД.

Фильтры (AND):

а) Н_ЛЮДИ.ОТЧЕСТВО < Александрович.

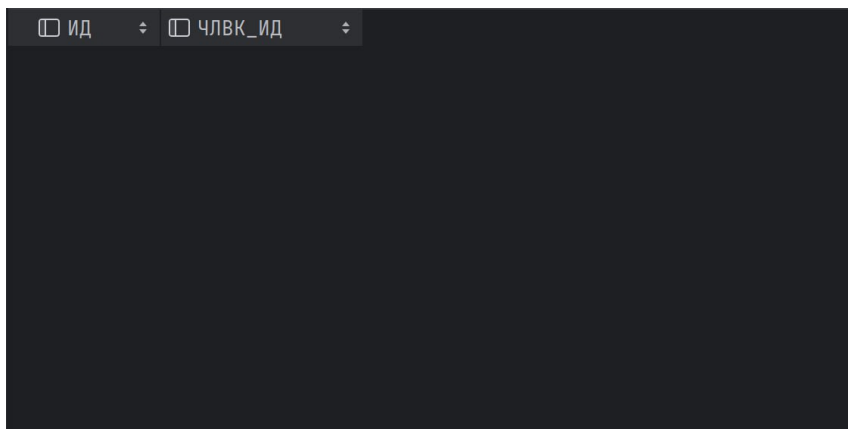
б) Н_СЕССИЯ.ЧЛВК_ИД > 106059.

с) Н_СЕССИЯ.ЧЛВК_ИД < 105948.

Вид соединения: RIGHT JOIN.

```
select Н_ЛЮДИ.ИД, "Н_СЕССИЯ".ЧЛВК_ИД
from "Н_ЛЮДИ"
    right join "Н_СЕССИЯ" on "Н_ЛЮДИ"."ИД" = "Н_СЕССИЯ"."ЧЛВК_ИД"
where "Н_ЛЮДИ"."ОТЧЕСТВО" < 'Александрович'
and "Н_СЕССИЯ"."ИД" > 106059
and "Н_СЕССИЯ"."ЧЛВК_ИД" < 105948
```

Вывод:



ИД	ЧЛВК_ИД
----	---------

Вывод пустой т.к. нет записей Н_СЕССИЯ.ЧЛВК_ИД > 106059, поменяем знак неравенства.

Тогда вывод будет следующим:

	ИД	ЧЛВК_ИД
1	100629	100629
2	100629	100629
3	100629	100629
4	100629	100629
5	100629	100629
6	100629	100629
7	100629	100629
8	100629	100629
9	100275	100275
10	100629	100629
11	100058	100058
12	100629	100629
13	100629	100629
14	100629	100629
15	100275	100275
16	100275	100275
17	100275	100275
18	100275	100275
19	100058	100058
20	100629	100629
21	100629	100629
22	100629	100629

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.ФАМИЛИЯ, Н_ВЕДОМОСТИ.ИД, Н_СЕССИЯ.ИД.

Фильтры (AND):

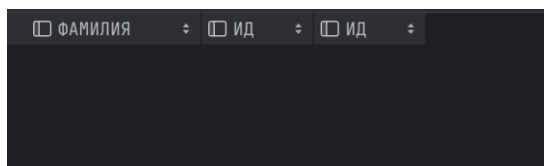
а) Н_ЛЮДИ.ИД > 100865.

б) Н_ВЕДОМОСТИ.ДАТА < 1998-01-05.

Вид соединения: INNER JOIN.

```
select "Н_ЛЮДИ".ФАМИЛИЯ, "Н_ВЕДОМОСТИ".ИД, "Н_СЕССИЯ".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" > 100865 and "Н_ВЕДОМОСТИ"."ДАТА" < '1998-01-05'
```

Вывод:



ФАМИЛИЯ	ИД	ИД
---------	----	----

Вывод пустой т.к. нет записей "Н_ВЕДОМОСТИ"."ДАТА" < '1998-01-05'

, поменяем знак неравенства.

Тогда вывод будет следующим:

	📄 ФАМИЛИЯ	📄 ИД	📄 ИД
1	Егоров	88677	936
2	Егоров	88678	936
3	Егоров	88679	936
4	Егоров	88680	936
5	Егоров	88681	936
6	Егоров	88682	936
7	Егоров	88683	936
8	Егоров	88684	936
9	Егоров	88685	936
10	Подоляк	52822	1166
11	Подоляк	57130	1166
12	Подоляк	57131	1166
13	Подоляк	77941	1166
14	Подоляк	77942	1166
15	Подоляк	75299	1166
16	Подоляк	75300	1166
17	Подоляк	75301	1166
18	Подоляк	75303	1166
19	Подоляк	77743	1166
20	Подоляк	77744	1166

Индексы

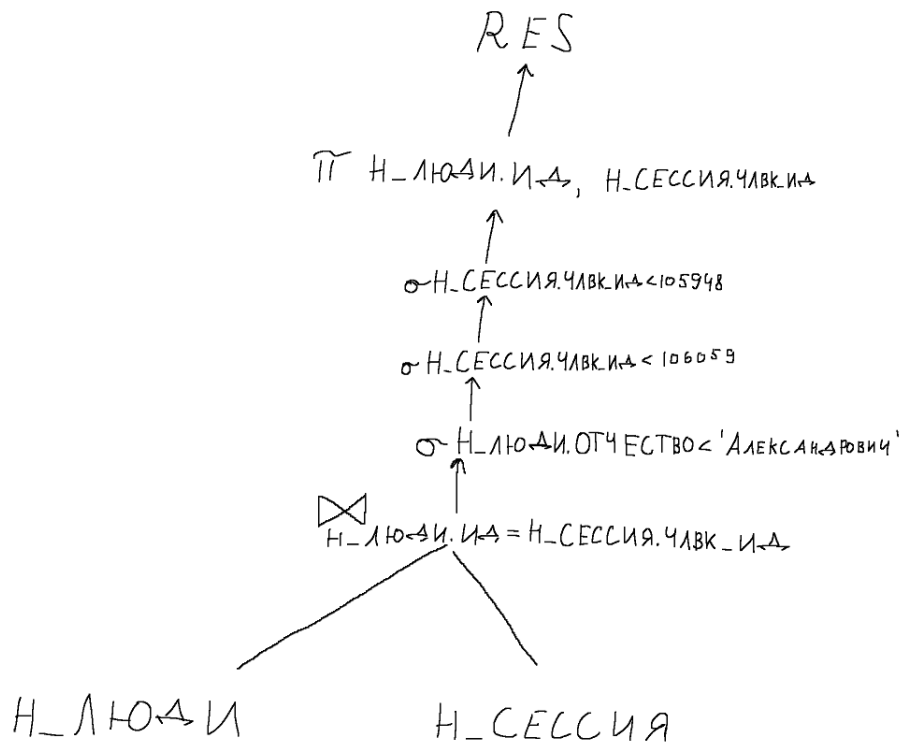
Для 1-го запроса добавим индекс (B-tree) на атрибут Н_ЛЮДИ.ИД , тогда увеличится скорость соединения таблиц. Так же добавим индекс(b-tree) на атрибут Н_ВЕДОМОСТИ.ЧЛВК_ИД , что увеличит фильтрацию таблицы Н_ВЕДОМОСТИ по условию ЧЛВК_ИД < 105948 and ЧЛВК_ИД < 106059 и увеличит скорость соединения таблиц.

Для 2-го запроса добавим индекс (B-tree) на атрибут Н_ЛЮДИ.ИД , тогда ускорится скорость фильтрации таблицы Н_ЛЮДИ по условию ИД >100865 и увеличится скорость соединения таблиц. Так же добавим индекс(b-tree) на атрибут Н_ВЕДОМОСТИ.ДАТА , что увеличит фильтрацию таблицы Н_ВЕДОМОСТИ по условию ДАТА > '1998-01-05'.

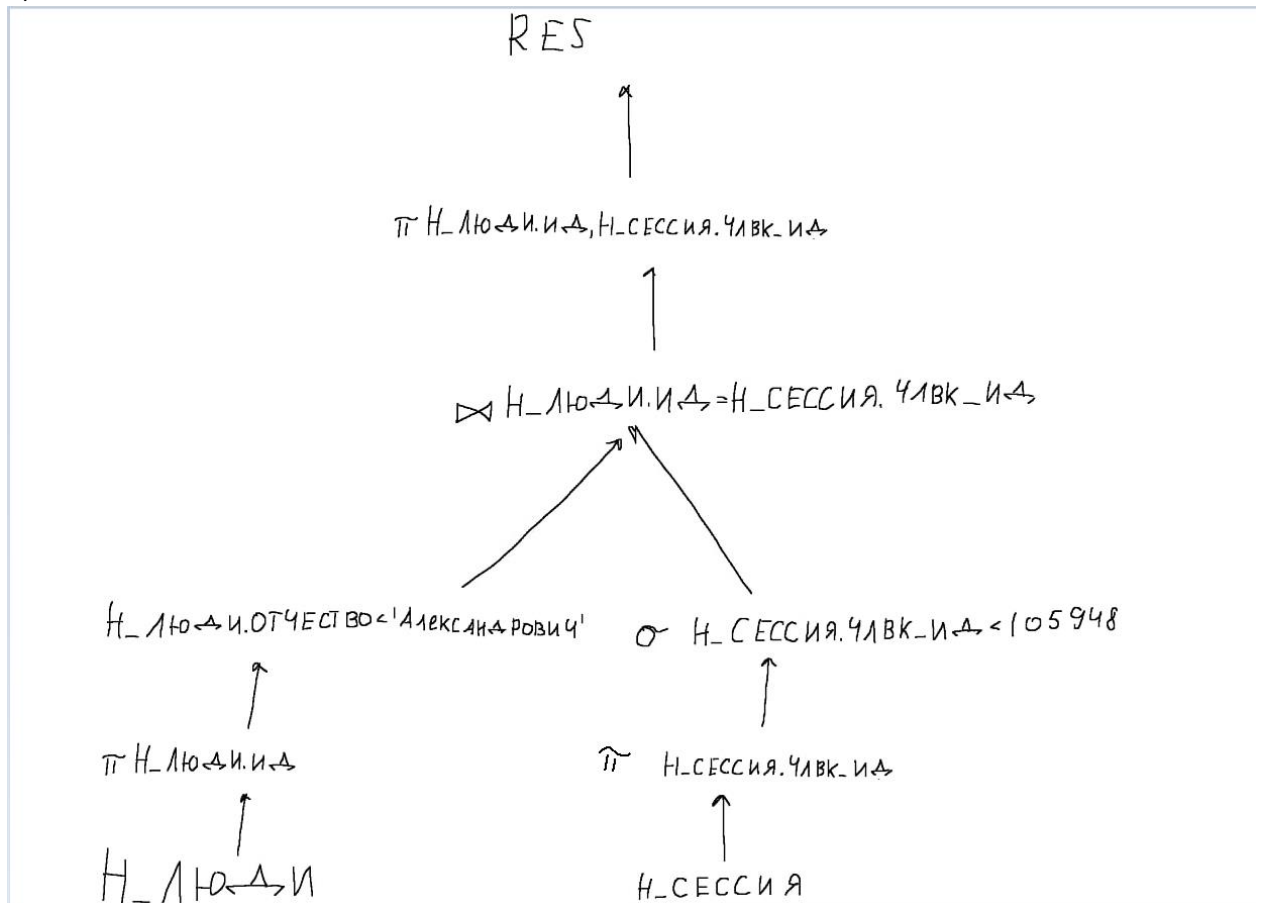
Планы

Планы для первого запроса

1)



2)



Второй план выполнится быстрее, так как фильтрация происходит то джоина таблиц, а так же нет лишних пунктов фильтрации, то есть ЧЛВК_ИД < 106059 and ЧЛВ_ИД < 105948. Так же будет использоваться индексный скан, что увеличит скорость соединения таблиц, тк есть существуют индексы на атрибут Н_ЛЮДИ.ИД и атрибут Н_СЕССИЯ.ЧЛВК_ИД, по которым они соединяются.

```
Nested Loop (cost=0.29..391.10 rows=123 width=8) (actual
time=0.176..1.946 rows=233 loops=1)

"  -> Seq Scan on "Н_СЕССИЯ" (cost=0.00..127.28 rows=2631 width=4)
(actual time=0.014..0.725 rows=2633 loops=1)"

"      Filter: (("ИД" < 106059) AND ("ЧЛВК_ИД" < 105948))"

      Rows Removed by Filter: 1119

      -> Memoize (cost=0.29..1.10 rows=1 width=4) (actual
time=0.000..0.000 rows=0 loops=2633)

"      Cache Key: "Н_СЕССИЯ"."ЧЛВК_ИД"

      Cache Mode: logical

      Hits: 2535 Misses: 98 Evictions: 0 Overflows: 0 Memory
Usage: 7kB

"      -> Index Scan using "ЧЛВК_ПК" on "Н_ЛЮДИ"
(cost=0.28..1.09 rows=1 width=4) (actual time=0.003..0.003 rows=0
loops=98)"

"      Index Cond: ("ИД" = "Н_СЕССИЯ"."ЧЛВК_ИД")"

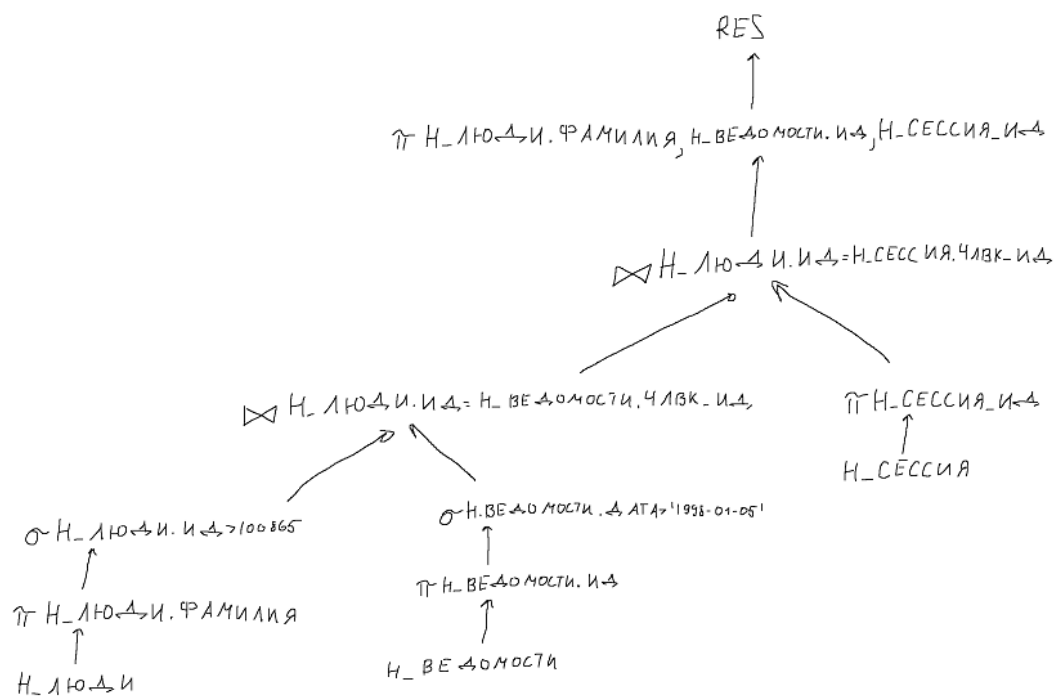
"      Filter: (("ОТЧЕСТВО")::text < 'Александрович'::text)"

      Rows Removed by Filter: 1

Planning Time: 0.438 ms

Execution Time: 2.005 ms
```

1)



Второй план будет выполняться быстрее, тк вся фильтрация происходит до соединения таблиц. Индексы так же положительно влияют на скорость nested loop, благодаря индексу на атрибуте Н_ЛЮДИ.ИД, что увеличит скорость соединения таблиц и Н_ВЕДОМОСТИ.ДАТА, что увеличит скорость фильтрации таблицы.

```
Nested Loop (cost=0.60..4700.69 rows=136649 width=24) (actual
time=0.315..6.062 rows=6370 loops=1)
"  Join Filter: ("Н_ЛЮДИ"."ИД" = "Н_ВЕДОМОСТИ"."ЧЛВК_ИД") "
    -> Nested Loop (cost=0.29..357.52 rows=3222 width=28) (actual
time=0.024..2.455 rows=1583 loops=1)
        -> Seq Scan on "Н_СЕССИЯ" (cost=0.00..108.52 rows=3752
width=8) (actual time=0.007..0.476 rows=3752 loops=1)
            -> Memoize (cost=0.29..0.87 rows=1 width=20) (actual
time=0.000..0.000 rows=0 loops=3752)
                Cache Key: "Н_СЕССИЯ"."ЧЛВК_ИД"
                Cache Mode: logical
                Hits: 3571 Misses: 181 Evictions: 0 Overflows: 0
Memory Usage: 20kB
                    -> Index Scan using "ЧЛВК_ПК" on "Н_ЛЮДИ"
(cost=0.28..0.86 rows=1 width=20) (actual time=0.002..0.002 rows=1
loops=181)
                        Index Cond: (("ИД" = "Н_СЕССИЯ"."ЧЛВК_ИД")
AND ("ИД" > 100865))
                            -> Memoize (cost=0.30..6.16 rows=66 width=8) (actual
time=0.000..0.001 rows=4 loops=1583)
                                Cache Key: "Н_СЕССИЯ"."ЧЛВК_ИД"
                                Cache Mode: logical
                                Hits: 1444 Misses: 139 Evictions: 0 Overflows: 0 Memory
Usage: 53kB
                                    -> Index Scan using "ВЕД_ЧЛВК_ФК_ИФК" on "Н_ВЕДОМОСТИ"
(cost=0.29..6.15 rows=66 width=8) (actual time=0.002..0.010 rows=8
loops=139)
                                        Index Cond: ("ЧЛВК_ИД" = "Н_СЕССИЯ"."ЧЛВК_ИД")
                                            Filter: ("ДАТА" > '1998-01-05 00:00:00'::timestamp
without time zone)
                                                Rows Removed by Filter: 0

Planning Time: 1.793 ms
Execution Time: 6.494 ms
```

Вывод:

В ходе выполнения лабораторной работы разобрался в работе индексов, разобрался в работе разных видов join'ов. Научились составлять планы и искать оптимальные среди них.