Санкт-Петербургский Национальный Исследовательский Университет ИТМО

Факультет Программной Инженерии и Компьютерной Техники



Вариант № 461842

Лабораторная работа №4

По дисциплине

Базы Данных

Выполнил студент группы P3130:

Толстых Мария

Преподаватель:

Барсуков Илья Александрович

Санкт-Петербург 2023 г.

# 1. Текст задания

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.  
Изменятся ли планы при добавлении индекса и как?

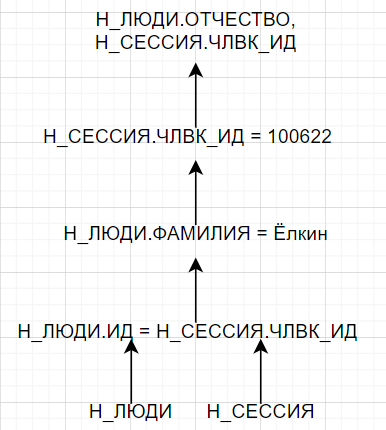
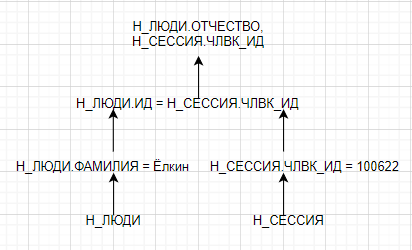
Для запросов 1-2 необходимо добавить в отчет вывод команды EXPLAIN ANALYZE [запрос]

**2. Реализация первого запроса**

Сделать запрос для получения атрибутов из указанных таблиц, применив фильтры по указанным условиям:  
Таблицы: Н\_ЛЮДИ, Н\_СЕССИЯ.  
Вывести атрибуты: Н\_ЛЮДИ.ОТЧЕСТВО, Н\_СЕССИЯ.ЧЛВК\_ИД.  
Фильтры (AND):  
a) Н\_ЛЮДИ.ФАМИЛИЯ = Ёлкин.  
b) Н\_СЕССИЯ.ЧЛВК\_ИД > 151200.  
c) Н\_СЕССИЯ.ЧЛВК\_ИД = 100622.  
Вид соединения: INNER JOIN.

SELECT Н\_ЛЮДИ.ОТЧЕСТВО, Н\_СЕССИЯ.ЧЛВК\_ИД  
FROM Н\_ЛЮДИ  
 JOIN Н\_СЕССИЯ ON Н\_ЛЮДИ.ИД = Н\_СЕССИЯ.ЧЛВК\_ИД  
WHERE ( Н\_ЛЮДИ.ФАМИЛИЯ = 'Ёлкин' AND Н\_СЕССИЯ.ЧЛВК\_ИД = 100622);

Планы выполнения:

1.  2. 

Следует выбрать оптимальный план, чтобы сократить время поиска, можно осуществить объединение только нужной выборки, а не всех сущностей. Если размер промежуточных данных становится меньше, значит план будет более оптимальным. Второй план является оптимальным.

Индексы:

CREATE INDEX ON "Н\_ЛЮДИ" USING HASH ("ИД");  
CREATE INDEX ON "Н\_ЛЮДИ" USING BTREE ("ФАМИЛИЯ");  
CREATE INDEX ON "Н\_СЕССИЯ" USING BTREE ("ЧЛВК\_ИД");

Используются операторы сравнения в выборке, следовательно оптимально использование BTREE, а соединение таблиц с использованием ИД оптимально использование HASH.

Explain Analyse:

Nested Loop (cost=0.56..14.59 rows=73 width=24) (actual time=0.033..0.034 rows=0 loops=1)

" -> Index Scan using ""ЧЛВК\_PK"" on ""Н\_ЛЮДИ"" (cost=0.28..8.30 rows=1 width=24) (actual time=0.033..0.033 rows=0 loops=1)"

" Index Cond: (""ИД"" = 100622)"

" Filter: ((""ФАМИЛИЯ"")::text = 'Ёлкин'::text)"

Rows Removed by Filter: 1

" -> Index Only Scan using ""SYS\_C003500\_IFK"" on ""Н\_СЕССИЯ"" (cost=0.28..5.56 rows=73 width=4) (never executed)"

" Index Cond: (""ЧЛВК\_ИД"" = 100622)"

Heap Fetches: 0

Planning Time: 0.190 ms

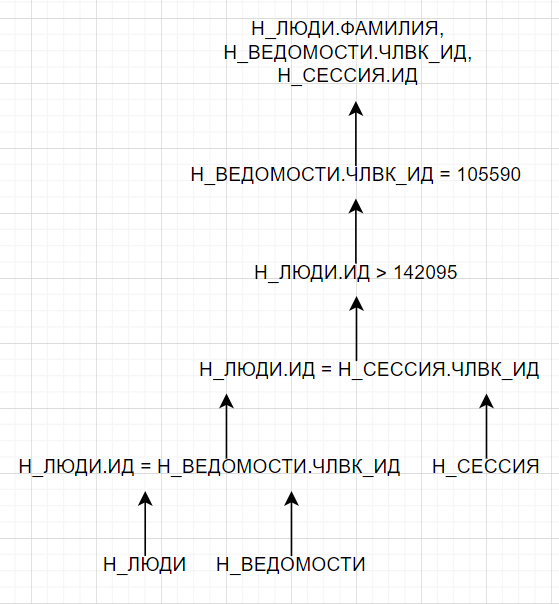
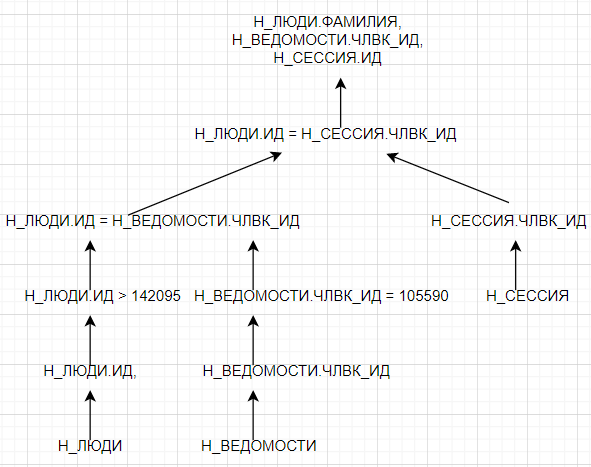
Execution Time: 0.068 ms

**3. Реализация второго запроса**

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.ЧЛВК\_ИД, Н\_СЕССИЯ.ИД.  
Фильтры (AND):  
a) Н\_ЛЮДИ.ИД > 142095.  
b) Н\_ВЕДОМОСТИ.ЧЛВК\_ИД = 105590.  
Вид соединения: INNER JOIN.

SELECT Н\_ЛЮДИ.ФАМИЛИЯ, Н\_ВЕДОМОСТИ.ЧЛВК\_ИД, Н\_СЕССИЯ.ИД  
FROM Н\_ЛЮДИ  
 INNER JOIN Н\_ВЕДОМОСТИ ON Н\_ЛЮДИ.ИД = Н\_ВЕДОМОСТИ.ЧЛВК\_ИД  
 INNER JOIN Н\_СЕССИЯ ON Н\_ЛЮДИ.ИД = Н\_СЕССИЯ.ЧЛВК\_ИД  
WHERE (Н\_ЛЮДИ.ИД > 142095 AND Н\_ВЕДОМОСТИ.ЧЛВК\_ИД = 105590);

Планы выполнения:

1.2.

Из-за того, что выборка происходит на более ранних этапах, идет соединение только нужных атрибутов, и размер промежуточных данных меньше. Второй план является оптимальным.

Индексы:

CREATE INDEX ON "Н\_ЛЮДИ" USING HASH ("ИД");  
CREATE INDEX ON "Н\_СЕССИЯ" USING HASH ("ЧЛВК ИД");  
CREATE INDEX ON "Н\_ВЕДОМОСТИ" USING HASH ("ЧЛВК ИД");

Соединение таблиц с использованием ИД – оптимально использование HASH.

Explain Analyse:

Nested Loop (cost=5.01..79.46 rows=1280 width=24) (actual time=0.065..0.067 rows=0 loops=1)

" -> Index Only Scan using ""ВЕД\_ЧЛВК\_FK\_IFK"" on ""Н\_ВЕДОМОСТИ"" (cost=0.29..5.42 rows=64 width=4) (actual time=0.038..0.043 rows=27 loops=1)"

" Index Cond: (""ЧЛВК\_ИД"" = 105590)"

Heap Fetches: 0

-> Materialize (cost=4.72..58.10 rows=20 width=24) (actual time=0.001..0.001 rows=0 loops=27)

-> Nested Loop (cost=4.72..58.00 rows=20 width=24) (actual time=0.010..0.011 rows=0 loops=1)

" -> Index Scan using ""ЧЛВК\_PK"" on ""Н\_ЛЮДИ"" (cost=0.28..8.30 rows=1 width=20) (actual time=0.010..0.010 rows=0 loops=1)"

" Index Cond: ((""ИД"" > 142095) AND (""ИД"" = 105590))"

" -> Bitmap Heap Scan on ""Н\_СЕССИЯ"" (cost=4.44..49.50 rows=20 width=8) (never executed)"

" Recheck Cond: (""ЧЛВК\_ИД"" = 105590)"

" -> Bitmap Index Scan on ""SYS\_C003500\_IFK"" (cost=0.00..4.43 rows=20 width=0) (never executed)"

" Index Cond: (""ЧЛВК\_ИД"" = 105590)"

Planning Time: 1.042 ms

Execution Time: 0.159 ms

# Вывод

При выполнении лабораторной работы я узнала, для чего нужно использование индексов, а именно для ускорения обработки запросов в SQL. Изучила поэтапное выполнение запроса, варианты построения планов выполнения и влияние индексов на процесс.\