Министерство образования и науки РФ

Федеральное государственное автономное

образовательное учреждение высшего образования

«Национальный исследовательский университет ИТМО»

**факультет программной инженерии и компьютерной техники**

**Лабораторная работа №4**

по дисциплине

Информационные системы и базы данных

Вариант 312493

Выполнил:

Студент группы P33302

Ершова А. И.

Преподаватель:

Шешуков Д. М.

Санкт-Петербург, 2022

Описание 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?

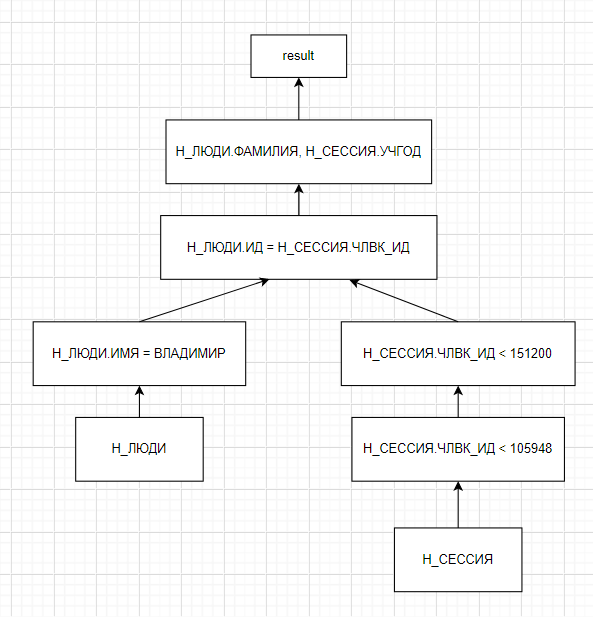
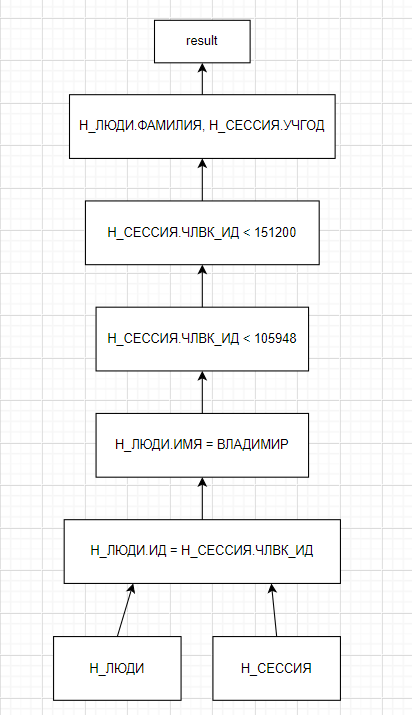
Для запросов 1–2 необходимо добавить в отчет вывод команды EXPLAIN ANALYZE [запрос]

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.ФАМИЛИЯ, Н\_СЕССИЯ.УЧГОД.  
Фильтры (AND):  
a) Н\_ЛЮДИ.ИМЯ = Владимир.  
b) Н\_СЕССИЯ.ЧЛВК\_ИД < 105948.  
c) Н\_СЕССИЯ.ЧЛВК\_ИД < 151200.  
Вид соединения: INNER JOIN.

SELECT Н\_ЛЮДИ.ФАМИЛИЯ, Н\_СЕССИЯ.УЧГОД  
FROM Н\_ЛЮДИ  
 INNER JOIN Н\_СЕССИЯ ON Н\_ЛЮДИ.ИД = Н\_СЕССИЯ.ЧЛВК\_ИД  
WHERE ( Н\_ЛЮДИ.ИМЯ = 'Владимир' AND Н\_СЕССИЯ.ЧЛВК\_ИД < 105948 AND Н\_СЕССИЯ.ЧЛВК\_ИД < 151200);

*Планы выполнения:*

**

Во втором плане происходит объединение только нужной выборки, а не всех сущностей. Размер промежуточных данных меньше, значит этот план является оптимальным.

*Индексы:*

CREATE INDEX ON "Н\_ЛЮДИ" USING HASH ("ИД");  
CREATE INDEX ON "Н\_ЛЮДИ" USING BTREE ("ОТЧЕСТВО");

CREATE INDEX ON "Н\_СЕССИЯ" USING BTREE ("УЧГОД");  
CREATE INDEX ON "Н\_СЕССИЯ" USING BTREE ("ЧЛВК\_ИД");

Выборка происходит с использованием операторов сравнения, поэтому оптимально использование BTREE. Соединение таблиц с использованием ИД – оптимально использование HASH.

*Explain Analyze:*

Nested Loop (cost=0.29..279.69 rows=49 width=26) (actual time=0.087..2.077 rows=173 loops=1)

-> Seq Scan on "Н\_СЕССИЯ" (cost=0.00..127.28 rows=2631 width=14) (actual time=0.023..0.864 rows=2633 loops=1)

Filter: (("ЧЛВК\_ИД" < 105948) AND ("ЧЛВК\_ИД" < 151200))

Rows Removed by Filter: 1119

-> Memoize (cost=0.29..0.49 rows=1 width=20) (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\_PK" on "Н\_ЛЮДИ" (cost=0.28..0.48 rows=1 width=20) (actual time=0.003..0.003 rows=0 loops=98)

Index Cond: ("ИД" = "Н\_СЕССИЯ"."ЧЛВК\_ИД")

Filter: (("ИМЯ")::text = 'Владимир'::text)

Rows Removed by Filter: 1

Planning Time: 1.052 ms

Execution Time: 2.135 ms

Изображение выглядит как текст

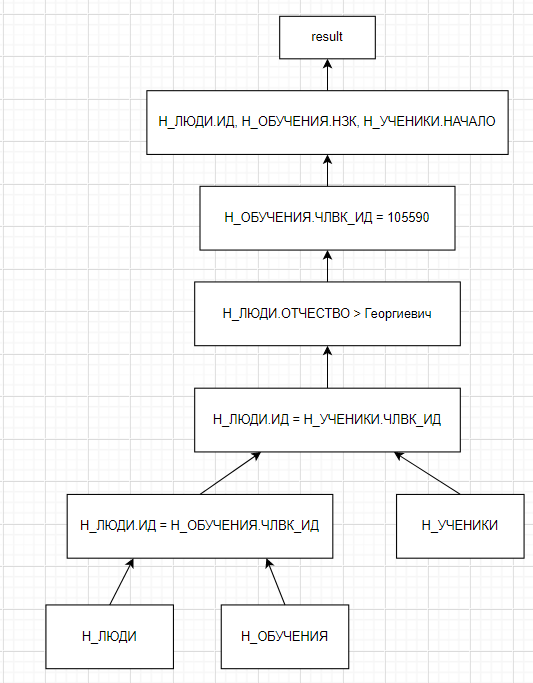
Автоматически созданное описание

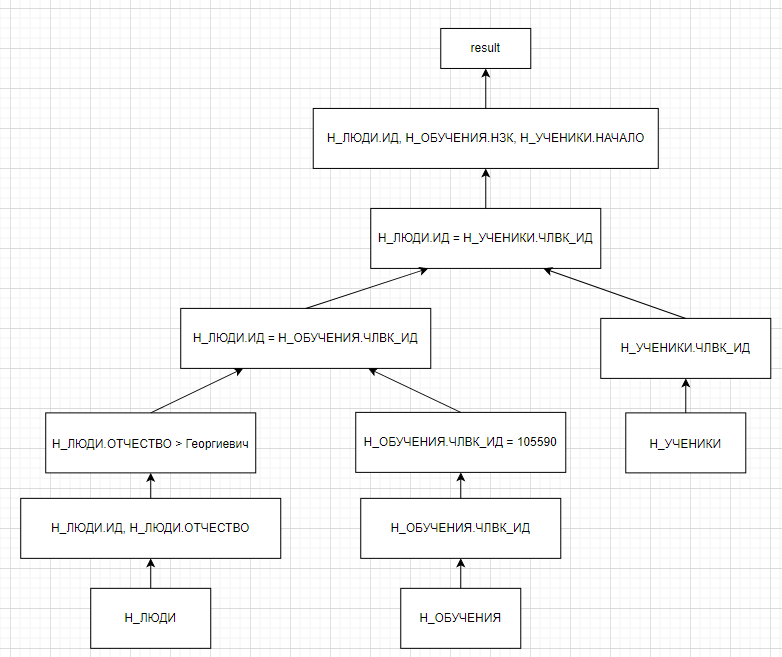
Реализация второго запроса

Сделать запрос для получения атрибутов из указанных таблиц, применив фильтры по указанным условиям:  
Таблицы: Н\_ЛЮДИ, Н\_ОБУЧЕНИЯ, Н\_УЧЕНИКИ.  
Вывести атрибуты: Н\_ЛЮДИ.ИД, Н\_ОБУЧЕНИЯ.НЗК, Н\_УЧЕНИКИ.НАЧАЛО.  
Фильтры: (AND)  
a) Н\_ЛЮДИ.ОТЧЕСТВО > Георгиевич.  
b) Н\_ОБУЧЕНИЯ.ЧЛВК\_ИД = 105590.  
Вид соединения: INNER JOIN.

SELECT Н\_ЛЮДИ.ИД, Н\_ОБУЧЕНИЯ.НЗК, Н\_УЧЕНИКИ.НАЧАЛО  
FROM Н\_ЛЮДИ  
 INNER JOIN Н\_ОБУЧЕНИЯ ON Н\_ЛЮДИ.ИД = Н\_ОБУЧЕНИЯ.ЧЛВК\_ИД  
 INNER JOIN Н\_УЧЕНИКИ ON Н\_ЛЮДИ.ИД = Н\_УЧЕНИКИ.ЧЛВК\_ИД  
WHERE (Н\_ЛЮДИ.ОТЧЕСТВО > 'Георгиевич' AND Н\_ОБУЧЕНИЯ.ЧЛВК\_ИД = 105590);

*Планы выполнения:*





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

*Индексы:*

CREATE INDEX "Н\_ЛЮДИ" USING HASH ("ИД");

CREATE INDEX "Н\_ЛЮДИ" USING BTREE ("ОТЧЕСТВО");

CREATE INDEX "Н\_ОБУЧЕНИЯ" USING BTREE ("ЧЛВК\_ИД");

Выборка происходит с использованием операторов сравнения, поэтому оптимально использование BTREE. Соединение таблиц с использованием ИД – оптимально использование HASH.

Nested Loop (cost=4.89..39.61 rows=5 width=18) (actual time=0.016..0.017 rows=0 loops=1)

-> Nested Loop (cost=0.56..16.61 rows=1 width=10) (actual time=0.016..0.016 rows=0 loops=1)

-> Index Scan using "ЧЛВК\_PK" on "Н\_ЛЮДИ" (cost=0.28..8.30 rows=1 width=4) (actual time=0.015..0.015 rows=0 loops=1)

Index Cond: ("ИД" = 105590)

Filter: (("ОТЧЕСТВО")::text > 'Георгиевич'::text)

Rows Removed by Filter: 1

-> Index Scan using "ОБУЧ\_ЧЛВК\_FK\_I" on "Н\_ОБУЧЕНИЯ" (cost=0.28..8.30 rows=1 width=10) (never executed)

Index Cond: ("ЧЛВК\_ИД" = 105590)

-> Bitmap Heap Scan on "Н\_УЧЕНИКИ" (cost=4.33..22.95 rows=5 width=12) (never executed)

Recheck Cond: ("ЧЛВК\_ИД" = 105590)

-> Bitmap Index Scan on "УЧЕН\_ОБУЧ\_FK\_I" (cost=0.00..4.32 rows=5 width=0) (never executed)

Index Cond: ("ЧЛВК\_ИД" = 105590)

Planning Time: 1.122 ms

Execution Time: 0.066 ms

Изображение выглядит как текст

Автоматически созданное описание

Выводы

При выполнении лабораторной работы я познакомилась с использованием индексов для ускорения обработки запросов в SQL. Изучила, как поэтапно происходит выполнение запроса, какие планы могут использоваться и как индексы влияют на процесс.