НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО

Факультет программной инженерии и компьютерных технологии

Изображение выглядит как Шрифт, логотип, Графика, белый

Автоматически созданное описание

Базы Данных

**Лабораторная работа № 4**

Выполнил студент

Еманов И.С.

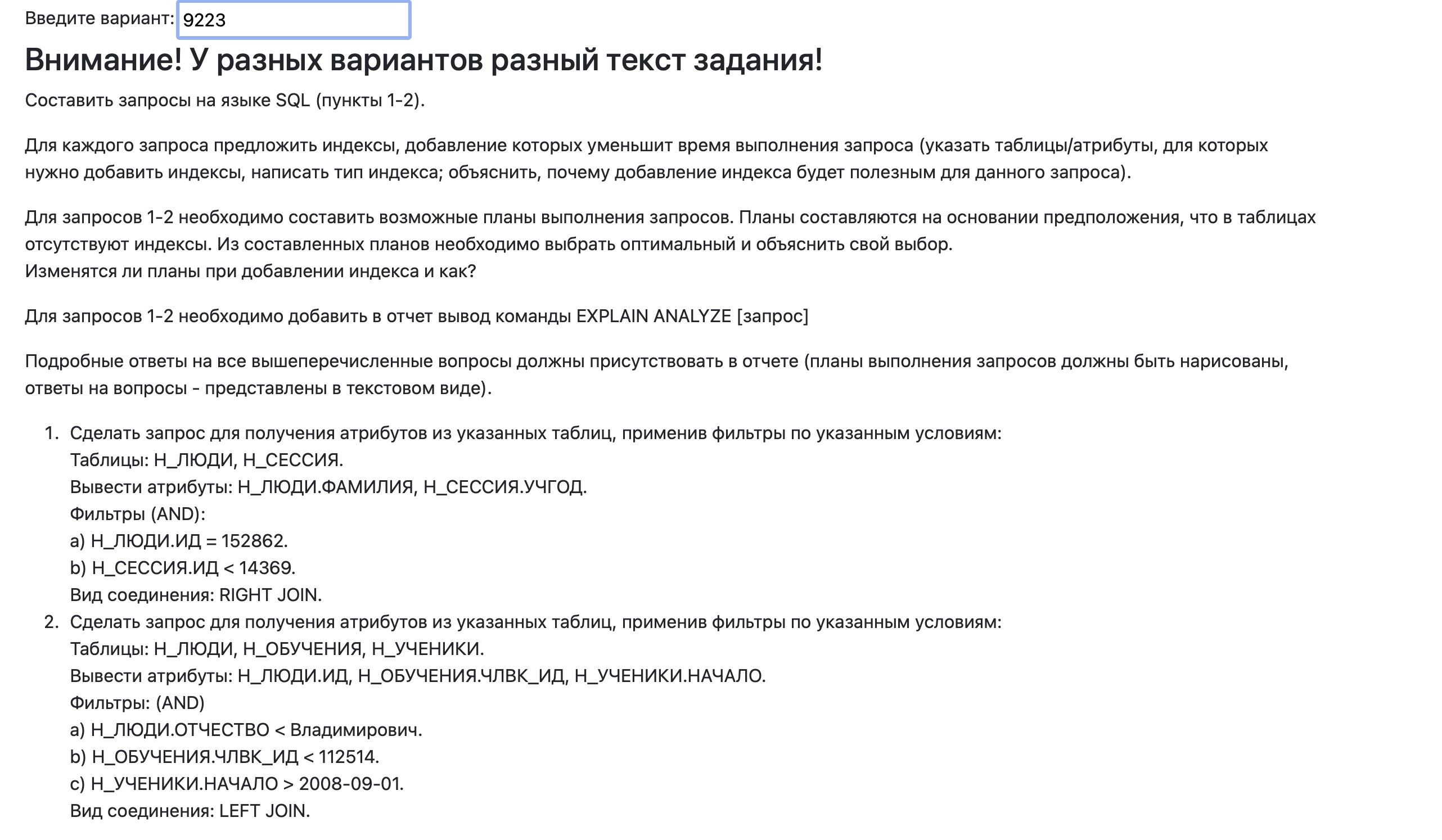
Группа № P3118

Преподаватель:

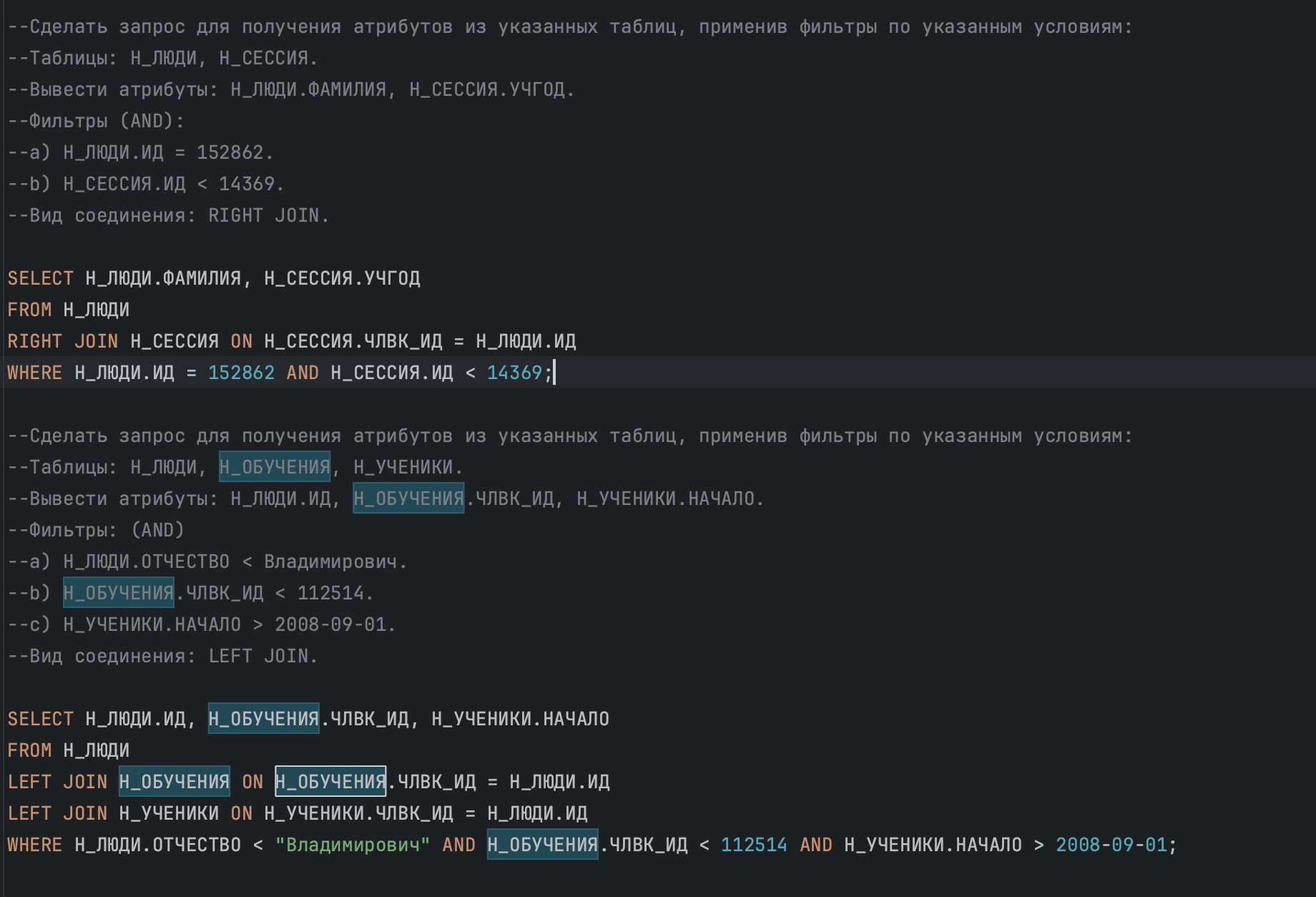
Лисицина В.В.

г. Санкт-Петербург

**2024**



**1. Запросы**

****

**EXPLAIN ANALYZE:**

**1 запрос:**

Nested Loop (cost=4.58..19.63 rows=1 width=26) (actual time=0.066..0.067 rows=0 loops=1)

-> Index Scan using "ЧЛВК\_PK" on "Н\_ЛЮДИ" (cost=0.28..8.30 rows=1 width=20) (actual time=0.052..0.053 rows=1 loops=1)

Index Cond: ("ИД" = 152862)

-> Bitmap Heap Scan on "Н\_СЕССИЯ" (cost=4.30..11.32 rows=1 width=14) (actual time=0.009..0.009 rows=0 loops=1)

Recheck Cond: ("ЧЛВК\_ИД" = 152862)

Filter: ("ИД" < 14369)

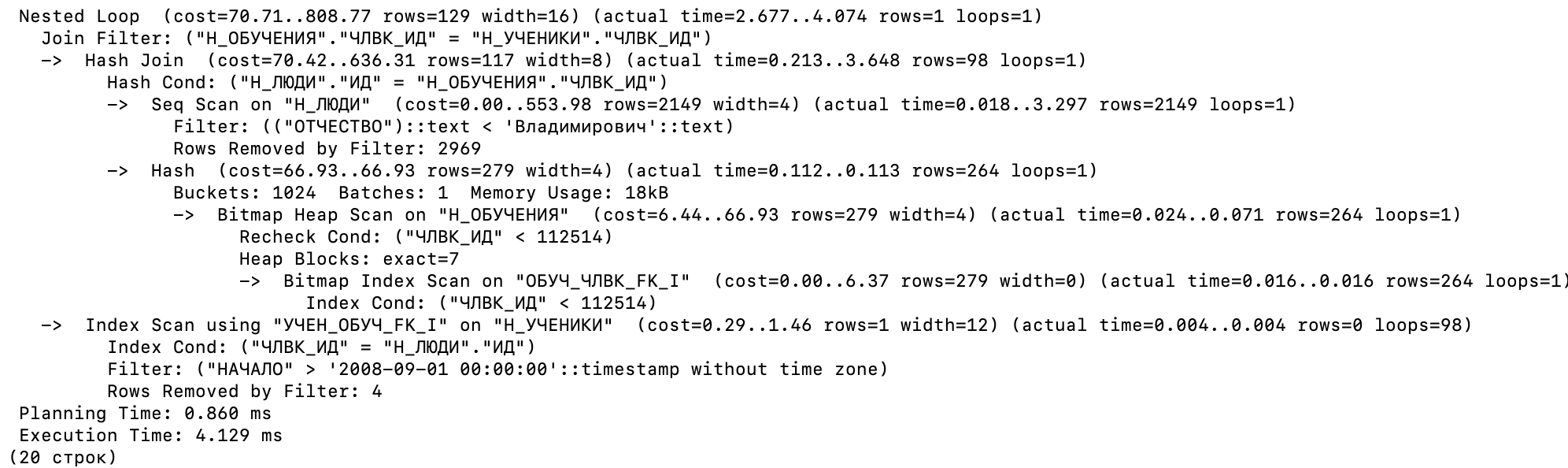
-> Bitmap Index Scan on "SYS\_C003500\_IFK" (cost=0.00..4.29 rows=2 width=0) (actual time=0.007..0.007 rows=0 loops=1)

Index Cond: ("ЧЛВК\_ИД" = 152862)

Planning Time: 1.144 ms

Execution Time: 0.179 ms

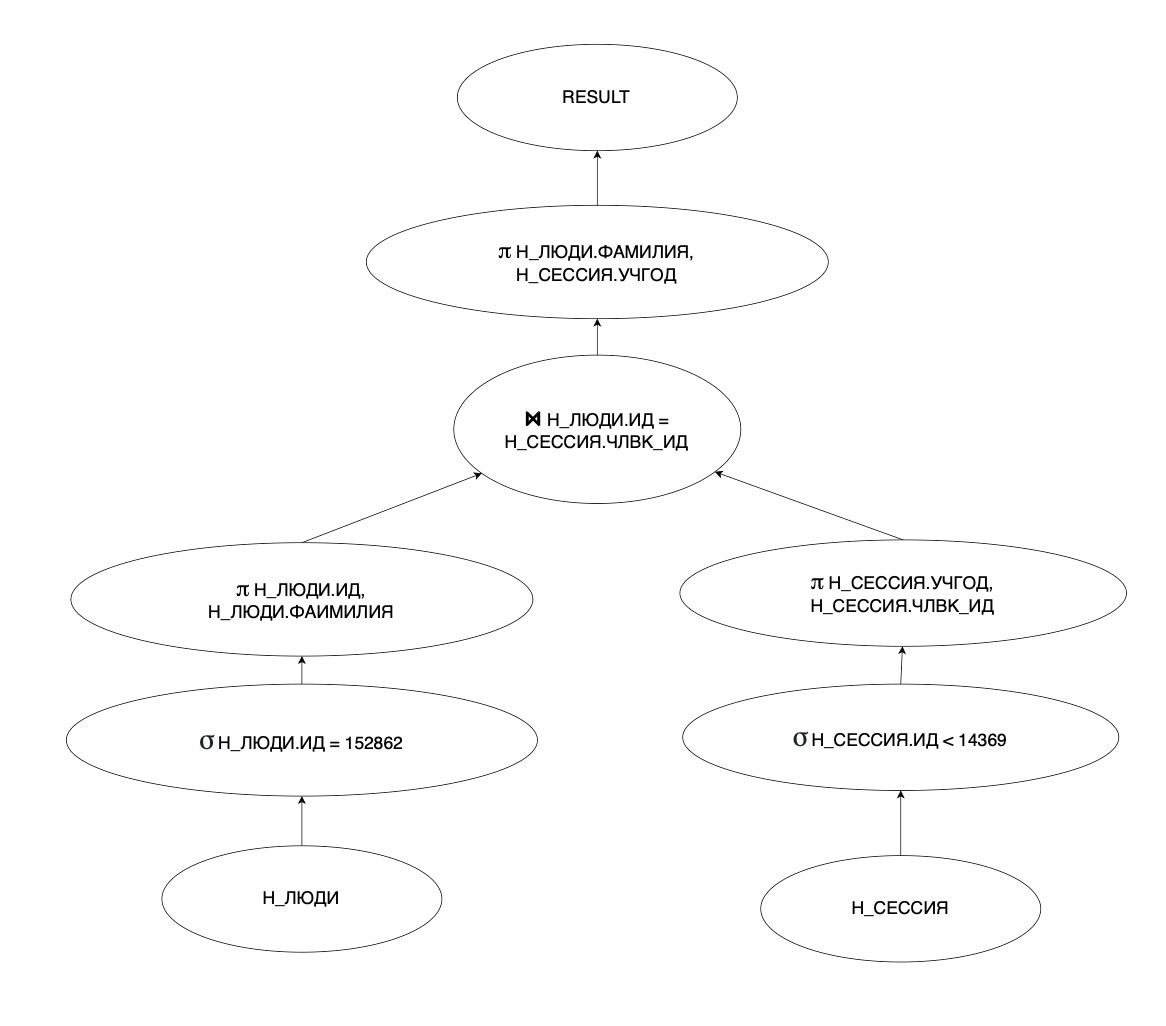
**2 запрос:**

****

**2. Планы выполнения запросов:**

**1 запрос:**

**1 план.**

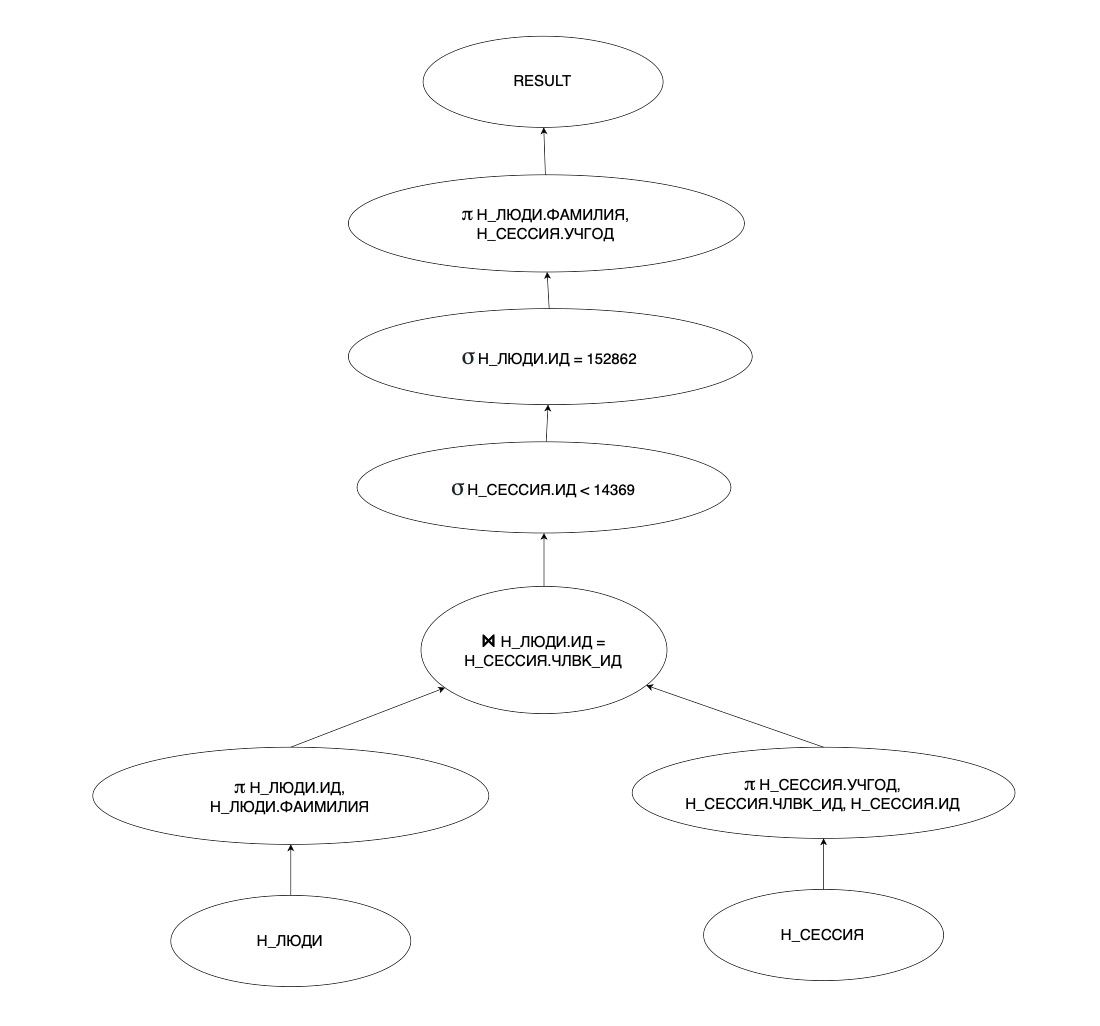
****

Этот план является оптимальным, так как:

1. выборка и проекции сделаны как можно раньше

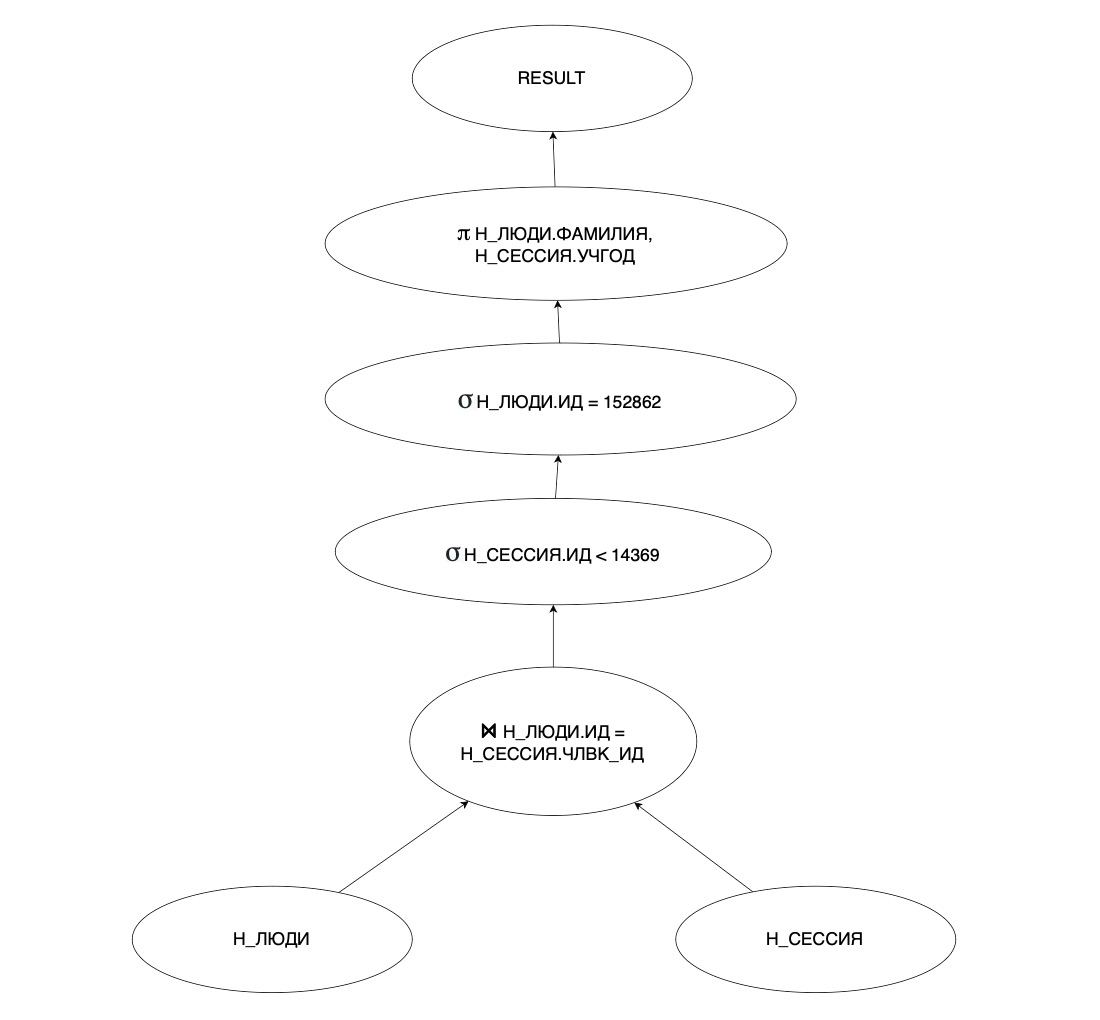
2. мы уменьшили размеры промежуточных данных (т.е. уменьшили число операция чтения записи во внешнюю память)

**2 план.**

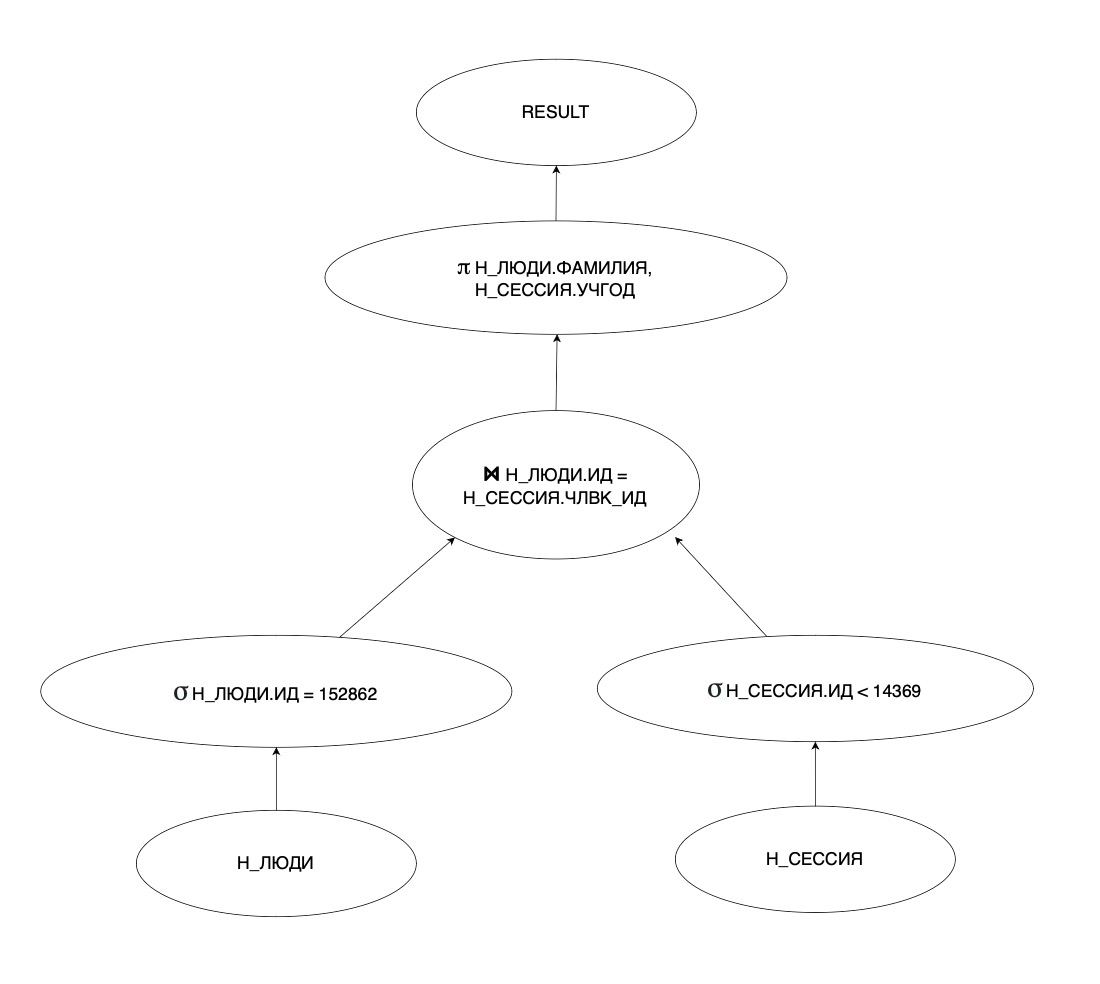
****

Выборка прошла позже => было увеличено кол-во операций чтения записи в память.

**3 план.**

****

**4 план**

****

**2 запрос:**

**1 план.**

****

Выборка и проекция выполнена как можно раньше, соответственно, лучший план выполнения.

**2 план.**

****

Наименее оптимизированный план выполнения, но фильтрация происходит в конце

**3 план.**

****

Совсем плохой план, фильтрации и проекция строятся в конце выполнения.

**4 план.**

Тут используем закон о переносе проекции за операцию соединенияя.

****

**5 план.**

****

**6 план.**

****

**Добавление индексов:**

**Для первого запроса**:

* Можно создать **индекс** по двум столбцам `Н\_СЕССИЯ.ЧЛВК\_ИД` и `Н\_СЕССИЯ.ИД`, так соединение будет проходить быстрее.
* Hash-Индекс или b-tree по столбцу `Н\_ЛЮДИ.ИД`:

Это основной индекс для таблицы `Н\_ЛЮДИ`, так как он используется в условии `WHERE Н\_ЛЮДИ.ИД = 152862`.

* B-tree-Индекс по столбцу `Н\_СЕССИЯ.ИД`:

Этот индекс ускорит проверку условия `Н\_СЕССИЯ.ИД < 14369` в запросе.

**Для второго запроса:**

* **B-tree-Индекс по столбцу `Н\_ЛЮДИ.ОТЧЕСТВО`:**

Это позволит быстро найти строки в таблице `Н\_ЛЮДИ`, которые удовлетворяют условию where.

* **B-tree-Индекс по столбцу `Н\_УЧЕНИКИ.ЧЛВК\_ИД`:**

Ускорит поиск в таблице `Н\_УЧЕНИКИ` строк, которые соответствуют условию where.

* **B-tree-Индекс по столбцу `Н\_УЧЕНИКИ.НАЧАЛО`:**

Это позволит быстро найти строки в таблице `Н\_УЧЕНИКИ`, которые удовлетворяют условию where.

**Вывод:**

Во время выполнения работы я познакомился с понятием индексов и их видами, оптимизацией запросов. Я узнал, как использовать команду EXPLAIN и читать ее вывод.