Федеральное государственное автономное образовательное учреждение высшего образования «Национальный исследовательский университет ИТМО»

Факультет программной инженерии и компьютерных технологий

Лабораторная работа №4

Вариант 3221

Выполнила:

Павличенко Софья Алексеевна, Р3115

Проверил:

Мартин Райла

Санкт-Петербург 2024г.

Оглавление

[Задание 3](#_Toc167570556)

[Запрос 1 4](#_Toc167570557)

[Индексы, добавление которых уменьшит время выполнения запроса 4](#_Toc167570558)

[Возможные планы выполнения запросов 5](#_Toc167570559)

[Результат команды EXPLAIN ANALYZE 6](#_Toc167570560)

[Изменения при добавлении индексов 7](#_Toc167570561)

[Запрос 2 7](#_Toc167570562)

[Индексы, добавление которых уменьшит время выполнения запроса 7](#_Toc167570563)

[Возможные планы выполнения запросов 8](#_Toc167570564)

[Результат команды EXPLAIN ANALYZE 9](#_Toc167570565)

[Изменения при добавлении индексов 10](#_Toc167570566)

[Заключение 11](#_Toc167570567)

# 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.  
Изменятся 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.УЧГОД.  
   Фильтры (AND):  
   a) Н\_ЛЮДИ.ФАМИЛИЯ < Петров.  
   b) Н\_СЕССИЯ.ДАТА > 2012-01-25.  
   c) Н\_СЕССИЯ.ДАТА < 2012-01-25.  
   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.ИМЯ, Н\_ВЕДОМОСТИ.ИД, Н\_СЕССИЯ.ЧЛВК\_ИД.  
   Фильтры (AND):  
   a) Н\_ЛЮДИ.ОТЧЕСТВО > Сергеевич.  
   b) Н\_ВЕДОМОСТИ.ДАТА > 2022-06-08.  
   Вид соединения: LEFT JOIN.

# Запрос 1

SELECT Н\_ЛЮДИ.ИД, Н\_СЕССИЯ.УЧГОД FROM Н\_ЛЮДИ  
RIGHT JOIN Н\_СЕССИЯ ON Н\_СЕССИЯ.ЧЛВК\_ИД = Н\_ЛЮДИ.ИД  
 WHERE Н\_ЛЮДИ.ФАМИЛИЯ < 'Петров' AND Н\_СЕССИЯ.ДАТА > '2012-01-25 00:00:00' AND Н\_СЕССИЯ.ДАТА < '2012-01-25 00:00:00';

## Индексы, добавление которых уменьшит время выполнения запроса

CREATE INDEX ИНД\_Н\_СЕССИЯ\_ЧЛВК\_ИД ON Н\_СЕССИЯ(ЧЛВК\_ИД);  
CREATE INDEX ИНД\_Н\_ЛЮДИ\_ИД ON Н\_ЛЮДИ(ИД);

Данные B-tree индексы будут полезны для запроса, так как ускорят операцию RIGHT JOIN, которая использует атрибуты Н\_СЕССИЯ.ЧЛВК\_ИД и Н\_ЛЮДИ.ИД, уменьшив количество строк, которые нужно сканировать в обеих таблицах.

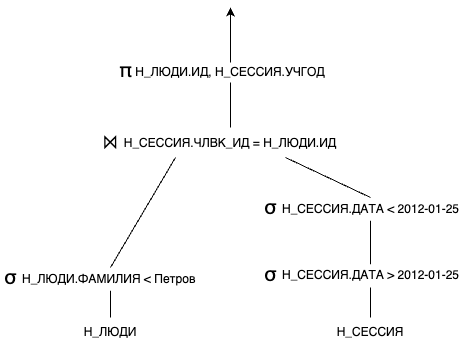
CREATE INDEX ИНД\_Н\_ЛЮДИ\_ФАМИЛИИ ON Н\_ЛЮДИ(ФАМИЛИЯ);

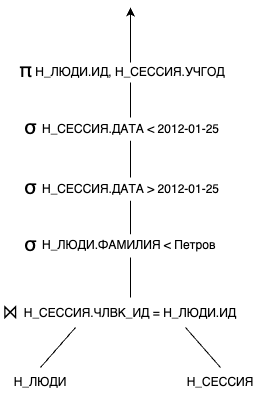
Данный B-tree индекс будет полезен для запроса, так как ускорит операцию фильтрации атрибута Н\_ЛЮДИ.ФАМИЛИЯ, уменьшив количество строк, которые нужно проверить в таблице Н\_ЛЮДИ.

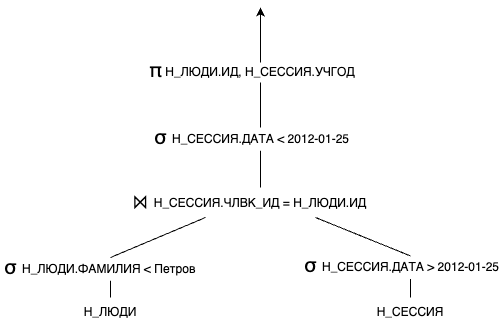
CREATE INDEX ИНД\_Н\_СЕССИЯ\_ДАТА ON Н\_СЕССИЯ(ДАТА);

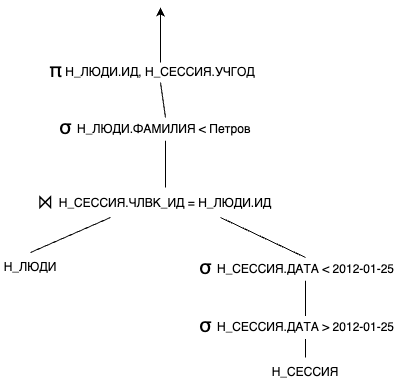
Данный B-tree индекс будет полезен для запроса, так как ускорит операцию фильтрации атрибута Н\_СЕССИЯ.ДАТА, позволив быстро находить строки в таблице Н\_СЕССИЯ, соответствующие указанному диапазону дат.

## Возможные планы выполнения запросов



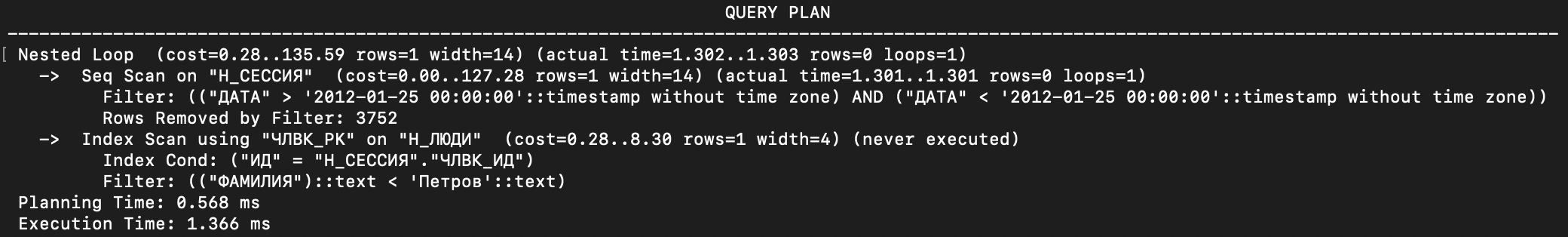






Оптимальным является 4 план выполнения. Сначала выполняется фильтрация таблицы Н\_СЕССИЯ по атрибуту ДАТА, в результате удаляются все строчки, так как оба условия фильтрации не выполнимы одновременно. Дальнейшая обработка не выполняется, так как для Nested Loop внешнее соотношение будет пустое (Н\_СЕССИЯ). Таким образом план в самом начале выполняет эффективную фильтрацию, значительно уменьшая количество строк для последующих операций до нуля.

Результат команды EXPLAIN ANALYZE:



## Изменения при добавлении индексов

При добавлении индексов запрос выполнится быстрее за счёт более быстрой фильтрации благодаря индексу на таблицу Н\_СЕССИЯ на атрибут ДАТА. Другие индексы использованы не будут, так как после фильтрации по ДАТА не останется строк для дальнейшей обработки.

Вместо полного сканирования таблиц будет использоваться индексное сканирование (Index Scan), и фильтрация произойдёт с помощью условия Index Cond.

# Запрос 2

SELECT Н\_ЛЮДИ.ИМЯ, Н\_ВЕДОМОСТИ.ИД, Н\_СЕССИЯ.ЧЛВК\_ИД FROM Н\_ЛЮДИ  
LEFT JOIN Н\_СЕССИЯ ON Н\_СЕССИЯ.ЧЛВК\_ИД = Н\_ЛЮДИ.ИД  
LEFT JOIN Н\_ВЕДОМОСТИ ON Н\_ВЕДОМОСТИ.ЧЛВК\_ИД = Н\_ЛЮДИ.ИД  
 WHERE Н\_ЛЮДИ.ОТЧЕСТВО > 'Сергеевич' AND Н\_ВЕДОМОСТИ.ДАТА > '2022-06-08 00:00:00';

## Индексы, добавление которых уменьшит время выполнения запроса

CREATE INDEX ИНД\_Н\_ЛЮДИ\_ИД ON Н\_ЛЮДИ(ИД); CREATE INDEX ИНД\_Н\_СЕССИЯ\_ЧЛВК\_ИД ON Н\_СЕССИЯ(ЧЛВК\_ИД);  
CREATE INDEX ИНД\_Н\_ВЕДОМОСТИ\_ЧЛВК\_ИД ON Н\_ВЕДОМОСТИ(ЧЛВК\_ИД);

Данные B-tree индексы будут полезны для запроса, так как ускорят операцию RIGHT JOIN, которая использует атрибуты Н\_ВЕДОМОСТИ.ЧЛВК\_ИД, Н\_СЕССИЯ.ЧЛВК\_ИД и Н\_ЛЮДИ.ИД, уменьшив количество строк, которые нужно сканировать в таблицах.

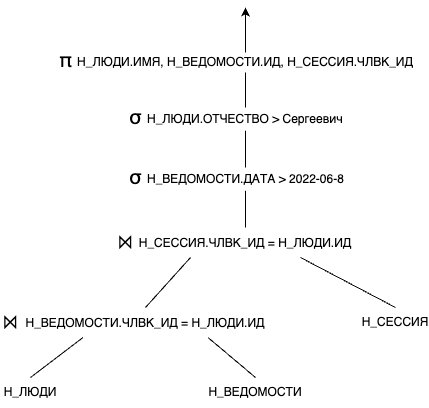
CREATE INDEX ИНД\_Н\_ЛЮДИ\_ОТЧЕСТВО ON Н\_ЛЮДИ(ОТЧЕСТВО);

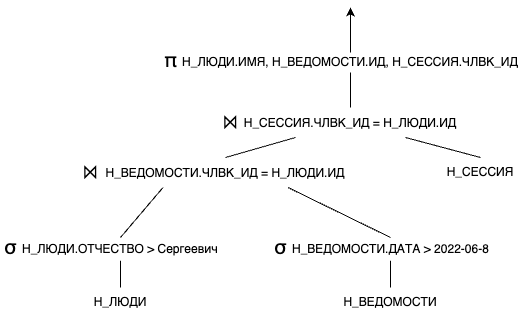
Данный B-tree индекс будет полезен для запроса, так как ускорит операцию фильтрации атрибута Н\_ЛЮДИ.ОТЧЕСТВО, уменьшив количество строк, которые нужно проверить в таблице Н\_ЛЮДИ.

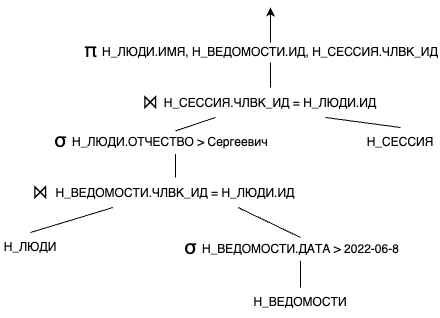
CREATE INDEX ИНД\_Н\_ВЕДОМОСТИ\_ДАТА ON Н\_ВЕДОМОСТИ(ДАТА);

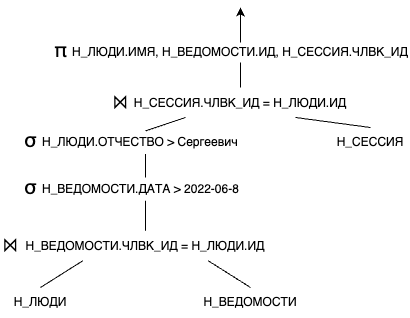
Данный B-tree индекс будет полезен для запроса, так как ускорит операцию фильтрации атрибута Н\_ВЕДОМОСТИ.ДАТА, позволив быстро находить строки в таблице Н\_ВЕДОМОСТИ, соответствующие указанному диапазону дат.

## Возможные планы выполнения запросов





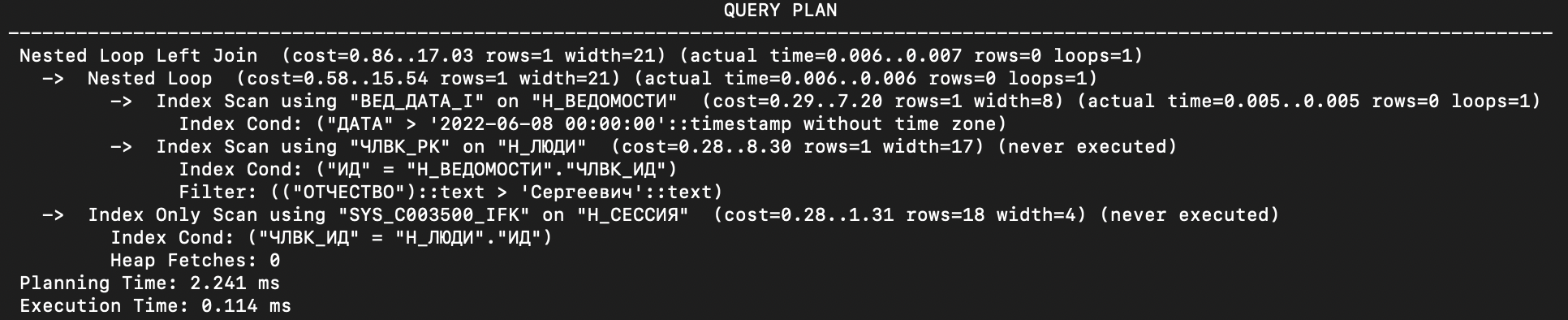




Оптимальным является 3 план выполнения. Сначала выполняется фильтрация таблицы Н\_ВЕДОМОСТИ по атрибуту ДАТА, в результате удаляются все строки, так как ни одна строчка таблицы не соответствует условию фильтрации. Дальнейший Nested Loop так же возвращает пустой результат.

Таким образом план в самом начале выполняет эффективную фильтрацию, значительно уменьшая количество строк для последующих операций до нуля.

Результат команды EXPLAIN ANALYZE:



## Изменения при добавлении индексов

При добавлении индексов запрос выполнится быстрее за счёт более быстрой фильтрации благодаря индексу на таблицу Н\_ВЕДОМОСТИ на атрибут ДАТА. Другие индексы использованы не будут, так как после фильтрации по ДАТА не останется строк для дальнейшей обработки.

Вместо полного сканирования таблиц будет использоваться индексное сканирование (Index Scan), и фильтрация произойдёт с помощью условия Index Cond.