

Министерство науки и высшего образования Российской Федерации  
федеральное государственное автономное образовательное учреждение высшего  
образования «Национальный исследовательский университет ИТМО»

Факультет программной инженерии и компьютерной техники

Лабораторная работа по  
Бадам данных №4  
Вариант №3528

Работу выполнил:

Агаев Х. Р.

Группа:

P3234

Санкт-Петербург,

2023

# СОДЕРЖАНИЕ

Стр.

<b>1</b>	<b>Текст задания .....</b>	<b>3</b>
<b>2</b>	<b>Запросы 1 и 2 .....</b>	<b>5</b>
<b>3</b>	<b>Использование индексов для запросов.....</b>	<b>6</b>
3.1	Индексы для 1-го запроса.....	6
3.2	Индексы для 2-го запроса.....	6
<b>4</b>	<b>Возможные планы выполнения запросов.....</b>	<b>8</b>
4.1	Планы выполнения 1-го запроса .....	8
4.2	Планы выполнения 2-го запроса .....	9
<b>5</b>	<b>EXPLAIN ANALYZE для запросов.....</b>	<b>11</b>
5.1	EXPLAIN ANALYZE для 1-го запроса .....	11
5.2	EXPLAIN ANALYZE для 2-го запроса .....	11
	<b>ЗАКЛЮЧЕНИЕ .....</b>	<b>13</b>

## 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].

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.ИМЯ, Н\_СЕССИЯ.ЧЛВК\_ИД.
  - Фильтры (AND):
    - a) Н\_ЛЮДИ.ИМЯ > Николай.
    - b) Н\_СЕССИЯ.УЧГОД = 2011/2012.
  - Вид соединения: LEFT 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.ДАТА, Н\_СЕССИЯ.УЧГОД.
  - Фильтры (AND):
    - a) Н\_ЛЮДИ.ИД = 152862.
    - b) Н\_ВЕДОМОСТИ.ЧЛВК\_ИД = 142390.
    - c) Н\_СЕССИЯ.ДАТА = 2012-01-25.

- Вид соединения: RIGHT JOIN.

## 2 Запросы 1 и 2

Запросы, с которыми мы будем в дальнейшем работать, выглядят следующим образом (см. листинг 2.1 и листинг 2.2).

Листинг 2.1: Код 1-го запроса

```
SELECT Н_ЛЮДИ.ИМЯ, Н_СЕССИЯ.ЧЛВК_ИД FROM
    Н_ЛЮДИ LEFT JOIN Н_СЕССИЯ ON Н_ЛЮДИ.ИД = Н
        _СЕССИЯ.ЧЛВК_ИД
    WHERE Н_ЛЮДИ.ИМЯ > 'Николай' AND Н_СЕССИЯ.УЧГОД =
        '2011/2012';
```

Листинг 2.2: Код 2-го запроса

```
SELECT Н_ЛЮДИ.ФАМИЛИЯ, Н_ВЕДОМОСТИ.ДАТА, Н
    _СЕССИЯ.УЧГОД FROM
    Н_ЛЮДИ RIGHT JOIN Н_ВЕДОМОСТИ ON Н_ЛЮДИ.ИД = Н
        _ВЕДОМОСТИ.ЧЛВК_ИД
    RIGHT JOIN Н_СЕССИЯ ON Н_ВЕДОМОСТИ.ЧЛВК_ИД = Н
        _СЕССИЯ.ЧЛВК_ИД
    WHERE Н_ЛЮДИ.ИД = 152862 AND Н_ВЕДОМОСТИ.ЧЛВК_ИД =
        142390 AND Н_СЕССИЯ.ДАТА = '2012-01-25';
```

## **3 Использование индексов для запросов**

### **3.1 Индексы для 1-го запроса**

Для 1-го запроса можно использовать следующие индексы, которые сократят его время выполнения:

- Индекс для поля Н\_ЛЮДИ.ИМЯ типа В-Tree;
- Индекс для поля Н\_СЕССИЯ.УЧГОД типа В-Tree;
- Индекс для поля Н\_СЕССИЯ.ЧЛВК\_ИД типа В-Tree.

Индекс для поля Н\_ЛЮДИ.ИМЯ типа В-Tree позволит проводить операцию WHERE значительно быстрее, потому что без него пришлось бы смотреть все строки таблицы Н\_ЛЮДИ. В данном случае тип индекса В-Tree актуален вследствие сравнения имени с другими именами ( $>$ ), которое производится в соответствии с лексиграфическим порядком.

Индекс для поля Н\_СЕССИЯ.УЧГОД типа В-Tree позволит проводить операцию WHERE значительно быстрее, потому что без него пришлось бы смотреть все строки таблицы Н\_СЕССИЯ. В данном случае тип индекса В-Tree актуален так как В-Tree также может использоваться для поиска соответствующего значения в таблице.

Индекс для поля Н\_СЕССИЯ.ЧЛВК\_ИД типа В-Tree позволит проводить соединение таблиц быстрее. Без использования индекса, СУБД бы проводила соединение таблиц используя вложенный цикл, а с ним же на каждую запись внешней таблицы Н\_ЛЮДИ будет искааться запись Н\_СЕССИЯ по индексу для поля ЧЛВК\_ИД, что достаточно быстро.

### **3.2 Индексы для 2-го запроса**

Для 2-го запроса можно использовать следующие индексы, которые сократят его время выполнения:

- Индекс для поля Н\_ЛЮДИ.ИД типа В-Tree;
- Индекс для поля Н\_ВЕДОМОСТИ.ЧЛВК\_ИД типа В-Tree;
- Индекс для поля Н\_СЕССИЯ.ДАТА типа В-Tree.

Индекс для поля Н\_ЛЮДИ.ИД типа В-Tree позволит проводить соединение таблиц быстрее. Без использования индекса, СУБД бы проводила соединение таблиц используя вложенный цикл, а с ним же на каждую запись внешней таблицы Н\_ВЕДОМОСТИ будет искааться запись Н\_ЛЮДИ по индексу для поля ИД, что достаточно быстро.

Индекс для поля Н\_ВЕДОМОСТИ.ЧЛВК\_ИД типа В-Tree позволит проводить соединение таблиц и операцию WHERE быстрее. Без использования индекса, СУБД бы проводила соединение таблиц используя вложенный цикл, а с ним же на каждую запись внешней таблицы Н\_СЕССИЯ будет искааться запись Н\_ВЕДОМОСТИ по индексу для поля ИД, что достаточно быстро. В данном случае тип индекса В-Tree актуален так как В-Tree также может использоваться для поиска соответствующего значения в таблице.

Индекс для поля Н\_СЕССИЯ.ДАТА типа В-Tree позволит проводить операцию WHERE значительно быстрее, потому что без него пришлось бы смотреть все строки таблицы Н\_СЕССИЯ. В данном случае тип индекса В-Tree актуален так как В-Tree также может использоваться для поиска соответствующего значения в таблице.

## 4 Возможные планы выполнения запросов

### 4.1 Планы выполнения 1-го запроса

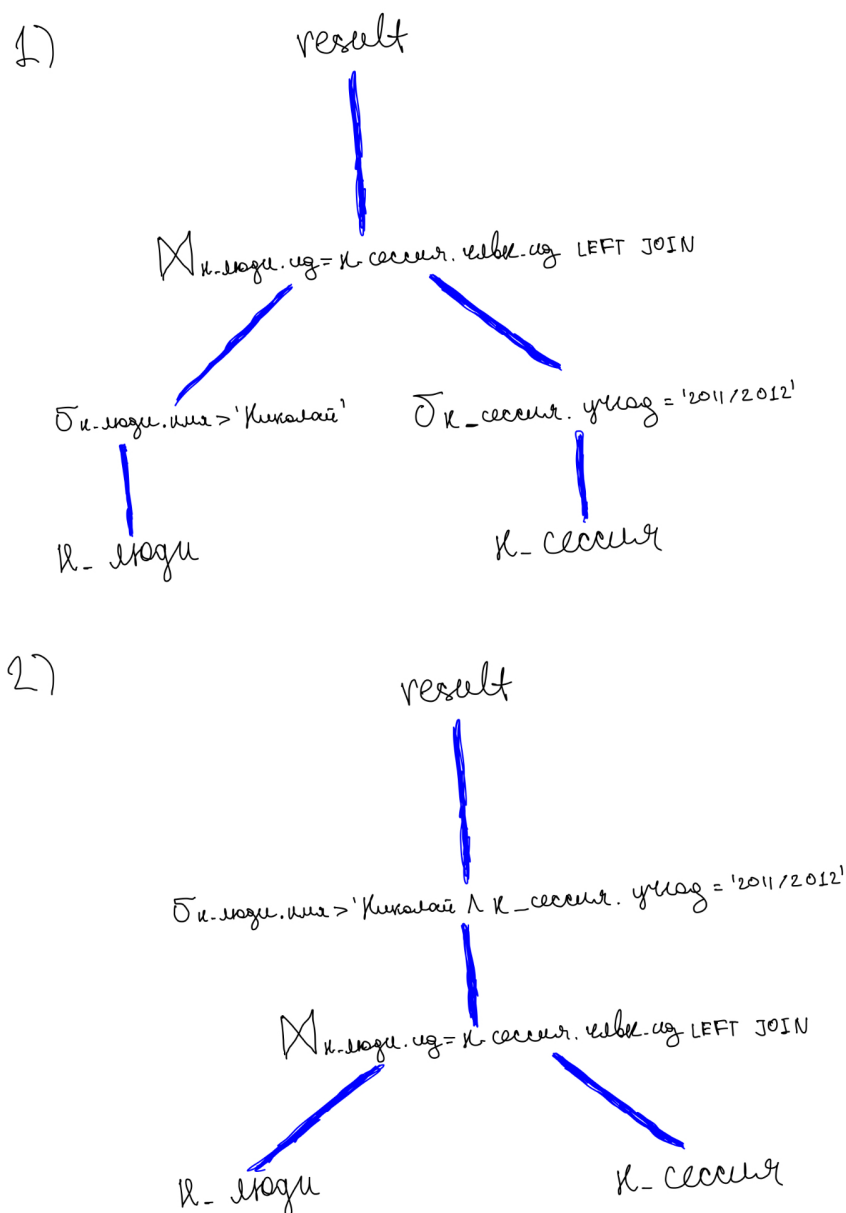


Рисунок 4.1 — Планы выполнения 1-го запроса (без индексов) добавить проекции для 1-го и 2-го планов

В данном запросе оптимальным планом будет являться 1-ый план, так как в нем фильтры для таблиц H\_СЕССИЯ и H\_ЛЮДИ применяются до выполнения соединения, что позволяет уменьшить количество соединяемых строк и увеличить скорость обработки запроса.



При использовании индексов для 1-го запроса структура планов останется такой же, но при этом в полях Н\_ЛЮДИ.ИМЯ и Н\_СЕССИЯ.УЧГОД будет использоваться индексный поиск, что увеличит скорость обработки запроса. При этом 1-ый запрос будет также оптимальнее для выполнения.

## 4.2 Планы выполнения 2-го запроса

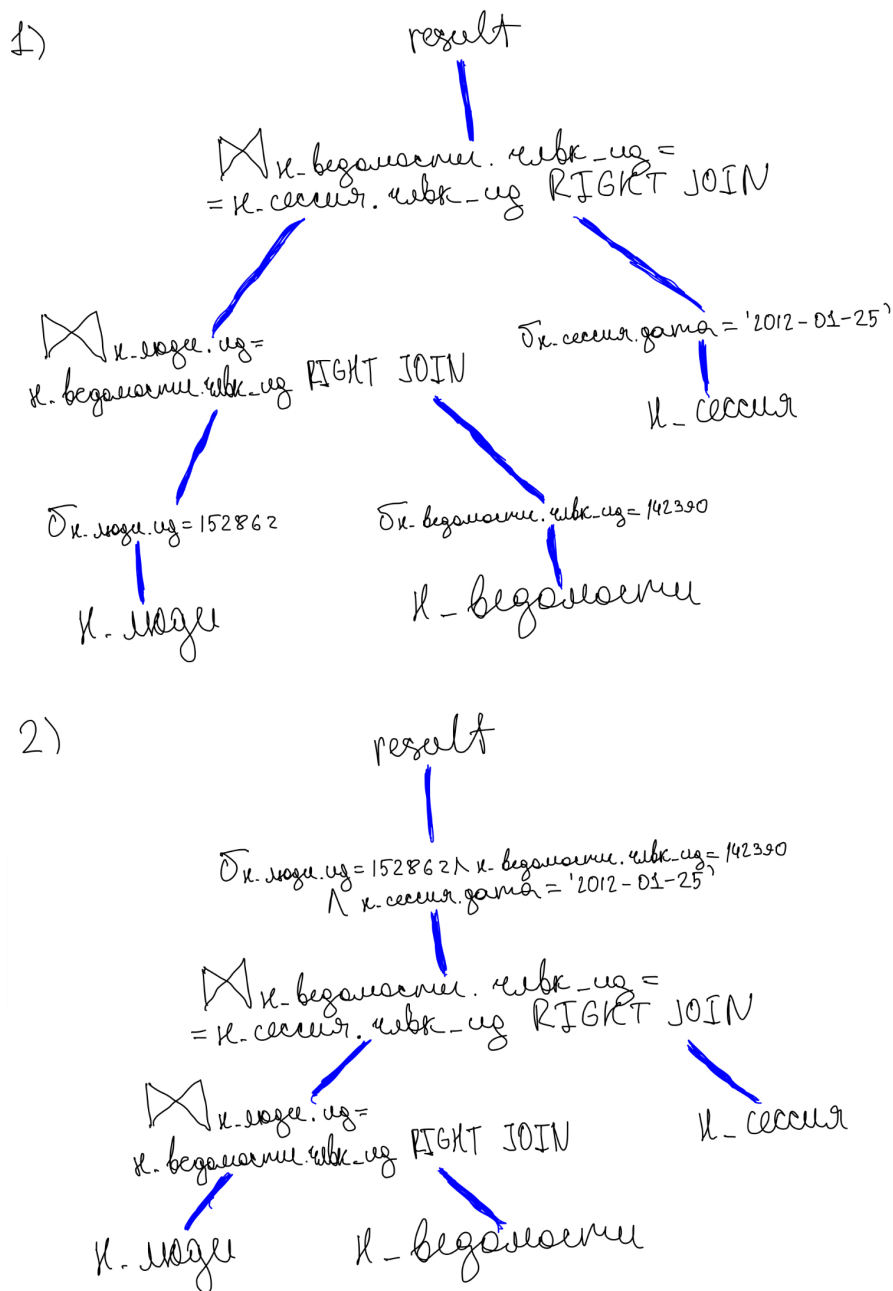


Рисунок 4.2 — Планы выполнения 2-го запроса (без индексов)

В данном запросе оптимальным планом будет являться 1-ый план, так как в нем фильтры для таблиц Н\_ЛЮДИ, Н\_ВЕДОМОСТИ и Н\_СЕССИЯ применяются до выполнения соединения, что позволяет уменьшить количество соединяемых строк и увеличить скорость обработки запроса.

При использовании индексов для 2-го запроса структура планов останется такой же, но при этом в полях Н\_ЛЮДИ.ИД, Н\_ВЕДОМОСТИ.ЧЛВК\_ИД и Н\_СЕССИЯ.ДАТА будет использоваться индексный поиск, что увеличит скорость обработки запроса. При этом 2-ый запрос будет также оптимальнее для выполнения.

## 5 EXPLAIN ANALYZE для запросов

### 5.1 EXPLAIN ANALYZE для 1-го запроса

Листинг 5.1: EXPLAIN ANALYZE для 1-го запроса

```
Hash Join  (cost=178.19..296.64 rows=40 width=17)
(actual time=3.545..3.688 rows=26 loops=1)
  Hash Cond: ("Н_СЕССИЯ"."ЧЛВК_ИД" =
              "Н_ЛЮДИ"."ИД")
-> Seq Scan on "Н_СЕССИЯ"  (cost=0.00..117.90
   rows=209 width=4) (actual time=0.082..0.715
   rows=209 loops=1)
   Filter: (("УЧГОД")::text = '2011/2012'::text)
   Rows Removed by Filter: 3543
-> Hash  (cost=163.97..163.97 rows=1137 width=17)
(actual time=2.936..2.937 rows=1141 loops=1)
  Buckets: 2048  Batches: 1  Memory Usage: 72kB
  -> Seq Scan on "Н_ЛЮДИ"
      (cost=0.00..163.97 rows=1137 width=17)
      (actual time=0.008..2.703 rows=1141
      loops=1)
      Filter: (("ИМЯ")::text > 'Николай'::text)
      Rows Removed by Filter: 3977
Planning Time: 0.629 ms
Execution Time: 3.718 ms
```

Как можно заметить из плана выполнения запроса, был применен Hash Join вследствие большого количества соединяемых строк, а значит что индексы на столбцы, по которым происходит соединение, применены **не будут**.

### 5.2 EXPLAIN ANALYZE для 2-го запроса

Листинг 5.2: EXPLAIN ANALYZE для 2-го запроса

```
Result (cost=0.00..0.00 rows=0 width=34) (actual
  time=0.001..0.002 rows=0 loops=1)
  One-Time Filter: false
Planning Time: 0.240 ms
Execution Time: 0.016 ms
```

Как можно заметить из плана выполнения запроса, результат запроса оказался пустым.

## **ЗАКЛЮЧЕНИЕ**

При выполнении данной лабораторной работы были изучены индексы и работа с ними, построены планы выполнений запросов.