

Министерство высшего образования и науки Российской Федерации  
Национальный научно-исследовательский университет ИТМО  
Факультет программной инженерии и компьютерной техники

ЛАБОРАТОРНАЯ РАБОТА №4

по дисциплине

**«БАЗЫ ДАННЫХ».**

Вариант №676.

Выполнил:

Петров Вячеслав Маркович,

Студент группы Р3108.

Преподаватель:

Афанасьев Дмитрий Борисович

Санкт-Петербург, 2024

## Оглавление

Задание.....	3
Запрос №1 .....	4
Задание .....	4
Запрос .....	4
Планы выполнения .....	5
Индексы.....	5
EXPLAIN ANALYZE .....	5
Запрос №2 .....	7
Задание .....	7
Запрос .....	7
Планы выполнения .....	8
Индексы.....	8
EXPLAIN ANALYZE .....	9
Вывод.....	10

# 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. Изменяются 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\_ТИПЫ\_ВЕДОМОСТЕЙ, Н\_ВЕДОМОСТИ.  
Вывести атрибуты: Н\_ТИПЫ\_ВЕДОМОСТЕЙ.НАИМЕНОВАНИЕ, Н\_ВЕДОМОСТИ.ИД.  
Фильтры (AND):  
а) Н\_ТИПЫ\_ВЕДОМОСТЕЙ.НАИМЕНОВАНИЕ = Перезачет.  
б) Н\_ВЕДОМОСТИ.ЧЛВК\_ИД = 153285.  
с) Н\_ВЕДОМОСТИ.ЧЛВК\_ИД < 117219.  
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.ОТЧЕСТВО, Н\_ВЕДОМОСТИ.ИД, Н\_СЕССИЯ.УЧГОД.  
Фильтры (AND):  
а) Н\_ЛЮДИ.ИД = 100012.  
б) Н\_ВЕДОМОСТИ.ИД > 1250981.  
Вид соединения: INNER JOIN.

# Запрос №1

## Задание

Сделать запрос для получения атрибутов из указанных таблиц, применив фильтры по указанным условиям:

Таблицы: Н\_ТИПЫ\_ВЕДОМОСТЕЙ, Н\_ВЕДОМОСТИ.

Вывести атрибуты: Н\_ТИПЫ\_ВЕДОМОСТЕЙ.НАИМЕНОВАНИЕ, Н\_ВЕДОМОСТИ.ИД.

Фильтры (AND):

а) Н\_ТИПЫ\_ВЕДОМОСТЕЙ.НАИМЕНОВАНИЕ = 'Перезачет'.

б) Н\_ВЕДОМОСТИ.ЧЛВК\_ИД = 153285.

с) Н\_ВЕДОМОСТИ.ЧЛВК\_ИД < 117219.

Вид соединения: RIGHT JOIN.

## Запрос

```
SELECT НАИМЕНОВАНИЕ,  
       Н_ВЕДОМОСТИ.ИД  
FROM Н_ВЕДОМОСТИ  
      RIGHT JOIN Н_ТИПЫ_ВЕДОМОСТЕЙ  
      ON Н_ВЕДОМОСТИ.ВЕД_ИД = Н_ТИПЫ_ВЕДОМОСТЕЙ.ИД  
WHERE НАИМЕНОВАНИЕ = 'Перезачет'  
      AND ЧЛВК_ИД = 153285  
      AND ЧЛВК_ИД < 117219;
```

Результат:

```
НАИМЕНОВАНИЕ | ИД  
-----+-----
```

(0 строк)

## Планы выполнения

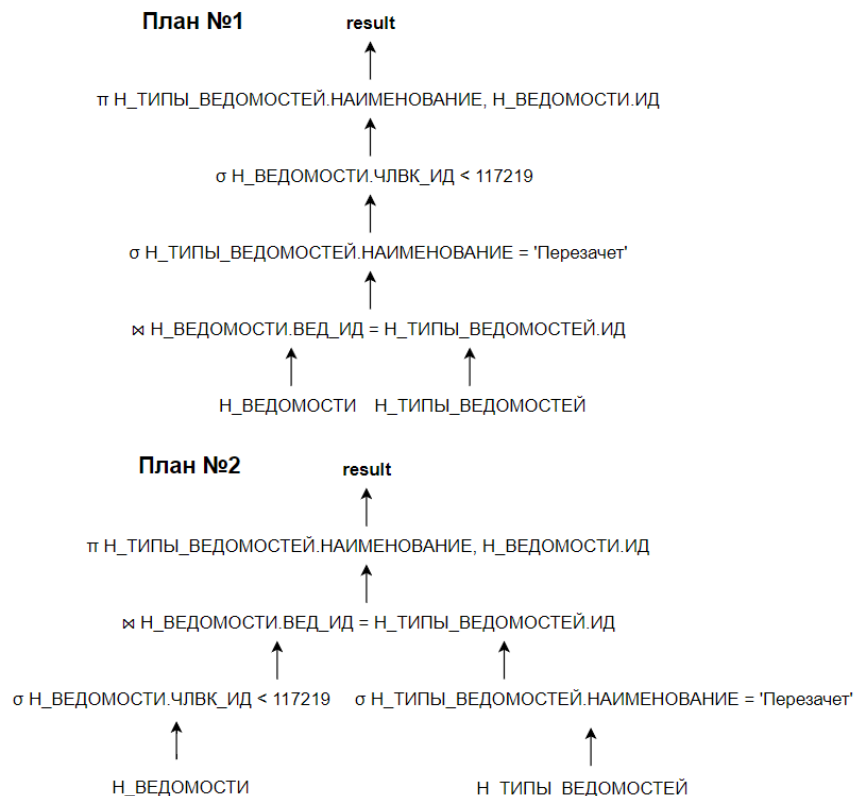


Рисунок 1 - планы выполнения запроса №1

Оптимальным является план №2, так как он производит объединение таблиц по уже выбранным ранее атрибутам, а не по таблицам целиком.

## Индексы

```
CREATE INDEX "ИНД_ВЕД_ИД" ON "Н_ВЕДОМОСТИ" USING HASH("ВЕД_ИД");
CREATE INDEX "ИНД_ВЕД_НАИМ" ON "Н_ТИПЫ_ВЕДОМОСТЕЙ" USING
HASH("НАИМЕНОВАНИЕ");
CREATE INDEX "ИНД_ВЕД_ЧЛВК" ON "Н_ВЕДОМОСТИ" USING
BTREE("ЧЛВК_ИД");
```

Выборка происходит с использованием операторов сравнения, поэтому оптимально использование BTREE. При этом для операции равенства оптимально использование HASH.

Добавление этих индексов должно ускорить выполнение запросов, так как по перечисленным полям происходит выборка с использованием оператора сравнения, а также теперь соединение таблиц будет происходить быстрее.

При добавлении индексов планы выполнения запросов изменятся, так как вместо полного сканирования таблиц будет производиться индексный скан и Nested Loops Join станет быстрее.

## EXPLAIN ANALYZE

```
EXPLAIN ANALYZE
SELECT НАИМЕНОВАНИЕ,
```

```
      H_ВЕДОМОСТИ.ИД
FROM H_ВЕДОМОСТИ
      RIGHT JOIN H_ТИПЫ_ВЕДОМОСТЕЙ
      ON H_ВЕДОМОСТИ.ВЕД_ИД = H_ТИПЫ_ВЕДОМОСТЕЙ.ИД
WHERE НАИМЕНОВАНИЕ = 'Перезачет'
      AND ЧЛВК_ИД = 153285
      AND ЧЛВК_ИД < 117219;
```

Nested Loop (cost=0.29..42.32 rows=1 width=30) (actual time=0.016..0.017 rows=0 loops=1)

Join Filter: ("H\_ВЕДОМОСТИ"."ВЕД\_ИД" = "H\_ТИПЫ\_ВЕДОМОСТЕЙ"."ИД")

-> Seq Scan on "H\_ТИПЫ\_ВЕДОМОСТЕЙ" (cost=0.00..1.04 rows=1 width=30) (actual time=0.010..0.010 rows=1 loops=1)

Filter: (("НАИМЕНОВАНИЕ")::text = 'Перезачет'::text)

Rows Removed by Filter: 2

-> Index Scan using "ВЕД\_ЧЛВК\_FK\_IFK" on "H\_ВЕДОМОСТИ" (cost=0.29..41.14 rows=12 width=8) (actual time=0.004..0.004 rows=0 loops=1)

Index Cond: (("ЧЛВК\_ИД" < 117219) AND ("ЧЛВК\_ИД" = 153285))

Planning Time: 0.997 ms

Execution Time: 0.071 ms

## Запрос №2

### Задание

Сделать запрос для получения атрибутов из указанных таблиц, применив фильтры по указанным условиям:

Таблицы: Н\_ЛЮДИ, Н\_ВЕДОМОСТИ, Н\_СЕССИЯ.

Вывести атрибуты: Н\_ЛЮДИ.ОТЧЕСТВО, Н\_ВЕДОМОСТИ.ИД, Н\_СЕССИЯ.УЧГОД.

Фильтры (AND):

а) Н\_ЛЮДИ.ИД = 100012.

б) Н\_ВЕДОМОСТИ.ИД > 1250981.

Вид соединения: INNER JOIN.

### Запрос

```
SELECT ОТЧЕСТВО,  
       Н_ВЕДОМОСТИ.ИД,  
       УЧГОД  
FROM Н_ЛЮДИ  
     JOIN Н_СЕССИЯ  
       ON Н_ЛЮДИ.ИД = Н_СЕССИЯ.ЧЛВК_ИД  
     JOIN Н_ВЕДОМОСТИ  
       ON Н_ВЕДОМОСТИ.ЧЛВК_ИД = Н_ЛЮДИ.ИД  
WHERE Н_ЛЮДИ.ИД = 100012  
       AND Н_ВЕДОМОСТИ.ИД > 1250981;
```

Результат:

ОТЧЕСТВО | ИД | УЧГОД

-----+-----+-----

(0 строк)

## Планы выполнения

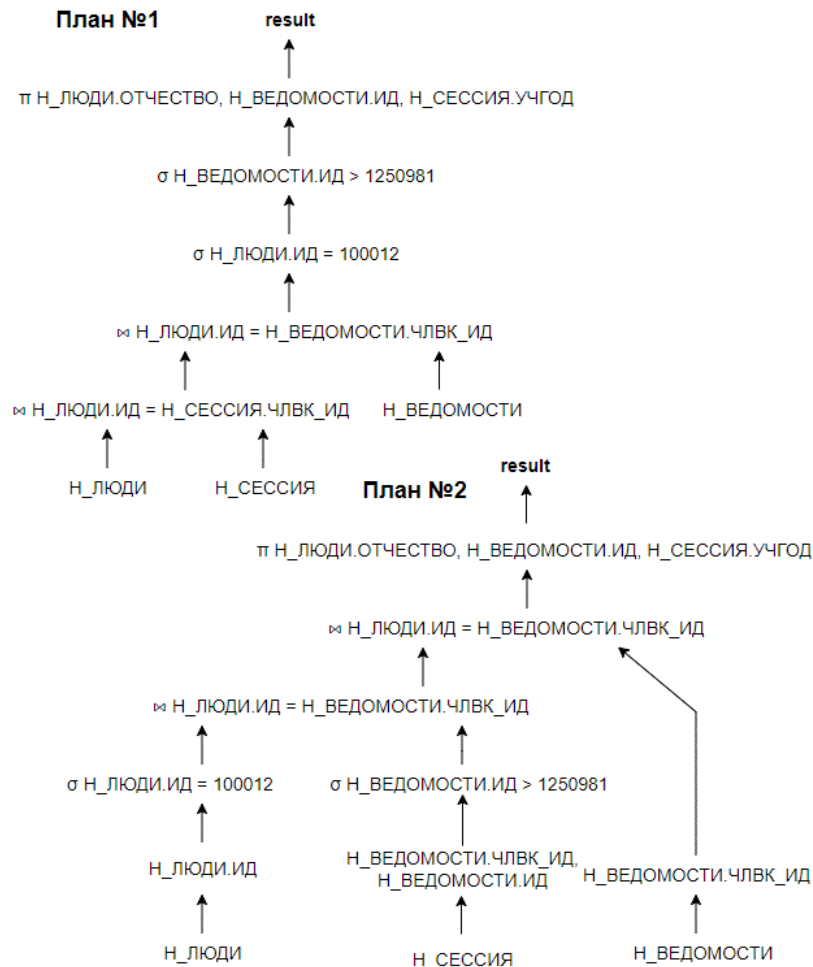


Рисунок 2 - планы выполнения запроса №2

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 "ИНД_ЛЮДИ_ИД" ON "Н_ЛЮДИ" USING HASH("ИД");  
CREATE INDEX "ИНД_ВЕД_ИД" ON "Н_ВЕДОМОСТИ" USING BTREE("ИД");  
CREATE INDEX "ИНД_ВЕД_ЧЛВК" ON "Н_ВЕДОМОСТИ" USING HASH("ЧЛВК_ИД");
```

Выборка происходит с использованием операторов сравнения, поэтому оптимально использование BTREE. При этом для операции равенства оптимально использование HASH.

Добавление этих индексов должно ускорить выполнение запросов, так как по перечисленным полям происходит выборка с использованием оператора сравнения, а также теперь соединение таблиц будет происходить быстрее.

При добавлении индексов планы выполнения запросов изменятся, так как вместо полного сканирования таблиц будет производиться индексный скан и Nested Loops Join станет быстрее.



## EXPLAIN ANALYZE

EXPLAIN ANALYZE

```
SELECT ОТЧЕСТВО,  
       H_ВЕДОМОСТИ.ИД,  
       УЧГОД  
FROM H_ЛЮДИ  
      JOIN H_СЕССИЯ  
        ON H_ЛЮДИ.ИД = H_СЕССИЯ.ЧЛВК_ИД  
      JOIN H_ВЕДОМОСТИ  
        ON H_ВЕДОМОСТИ.ЧЛВК_ИД = H_ЛЮДИ.ИД  
WHERE H_ЛЮДИ.ИД = 100012  
      AND H_ВЕДОМОСТИ.ИД > 1250981;
```

Nested Loop (cost=4.90..231.68 rows=48 width=34) (actual time=0.009..0.009 rows=0 loops=1)

-> Index Scan using "БЕД\_ЧЛВК\_FK\_IFK" on "H\_ВЕДОМОСТИ" (cost=0.29..199.53 rows=8 width=8) (actual time=0.008..0.008 rows=0 loops=1)

Index Cond: ("ЧЛВК\_ИД" = 100012)

Filter: ("ИД" > 1250981)

-> Materialize (cost=4.61..31.56 rows=6 width=34) (never executed)

-> Nested Loop (cost=4.61..31.53 rows=6 width=34) (never executed)

-> Index Scan using "ЧЛВК\_ПК" on "H\_ЛЮДИ" (cost=0.28..8.30 rows=1 width=24) (never executed)

Index Cond: ("ИД" = 100012)

-> Bitmap Heap Scan on "H\_СЕССИЯ" (cost=4.33..23.17 rows=6 width=14) (never executed)

Recheck Cond: ("ЧЛВК\_ИД" = 100012)

-> Bitmap Index Scan on "SYS\_C003500\_IFK" (cost=0.00..4.33 rows=6 width=0) (never executed)

Index Cond: ("ЧЛВК\_ИД" = 100012)

Planning Time: 0.603 ms

Execution Time: 0.064 ms

## Вывод

При выполнении лабораторной работы я познакомился с использованием индексов для ускорения обработки запросов в SQL, а также планами выполнения запросов, их построением и анализом.