

Лабораторная работа № 4
по дисциплине базы данных
Индексы
Вариант № 1678

Выполнил:
студент группы Р3116
Сиразетдинов А. Н
Преподаватель:
Горбунов М. В.

г. Санкт-Петербург
2023г.

Содержание

1	Текст задания	3
2	Запрос 1	4
2.1	Реализация запроса на SQL	4
2.2	Использование индексов	4
2.3	Планы выполнения запроса	4
2.4	Оптимальный план	6
2.5	Выполнение команды EXPLAIN ANALYZE	6
3	Запрос 2	7
3.1	Реализация запроса на SQL	7
3.2	Использование индексов	7
3.3	Планы выполнения запросов	8
3.4	Оптимальный план	9
3.5	Выполнение команды EXPLAIN ANALYSE	9
4	Вывод	10

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. 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:

Н_ТИПЫ_ВЕДОМОСТЕЙ, Н_ВЕДОМОСТИ.

Вывести атрибуты: Н_ТИПЫ_ВЕДОМОСТЕЙ.ИД, Н_ВЕДОМОСТИ.ЧЛВК_ИД.

Фильтры (AND):

(a) Н_ТИПЫ_ВЕДОМОСТЕЙ.НАИМЕНОВАНИЕ > Ведамость.

(b) Н_ВЕДОМОСТИ.ИД > 1250981.

Вид соединения: RIGHT JOIN.

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

Таблицы: Н_ЛЮДИ, Н_ОБУЧЕНИЯ, Н_УЧЕНИКИ.

Вывести атрибуты: Н_ЛЮДИ.ИМЯ, Н_ОБУЧЕНИЯ.ЧЛВК_ИД, Н_УЧЕНИКИ.НАЧАЛО.

Фильтры: (AND)

(a) Н_ЛЮДИ.ФАМИЛИЯ < Ёлкин.

(b) Н_ОБУЧЕНИЯ.ЧЛВК_ИД > 112514.

(c) Н_УЧЕНИКИ.ИД > 100410.

Вид соединения: RIGHT JOIN.

2 Запрос 1

2.1 Реализация запроса на SQL

```
SELECT
  Н_ТИПЫ_ВЕДОМОСТЕЙ.ИД,
  Н_ВЕДОМОСТИ.ЧЛВК_ИД
FROM
  "Н_ТИПЫ_ВЕДОМОСТЕЙ"
RIGHT JOIN "Н_ВЕДОМОСТИ"
ON
  Н_ВЕДОМОСТИ."ТВ_ИД" = Н_ТИПЫ_ВЕДОМОСТЕЙ."ИД"
WHERE
  Н_ТИПЫ_ВЕДОМОСТЕЙ.НАИМЕНОВАНИЕ > 'Ведомость'
AND Н_ВЕДОМОСТИ.ИД > 1250981;
```

2.2 Использование индексов

Таблица Н_ТИПЫ_ВЕДОМОСТЕЙ

Индекс b-tree на атрибуте НАИМЕНОВАНИЕ не уменьшит время выполнения запроса, потому что таблица содержит малое количество строк.

Хеш индекс на атрибуте ИД уменьшит время выполнения запроса потому что обращаться к хеш-таблице быстрее и hash-join работает быстрее nested-loop join

Индексы GiST, SP_GiST, GIN, BRIN не могут быть использованы из-за неподходящих условий выборки.

Таблица Н_ВЕДОМОСТИ

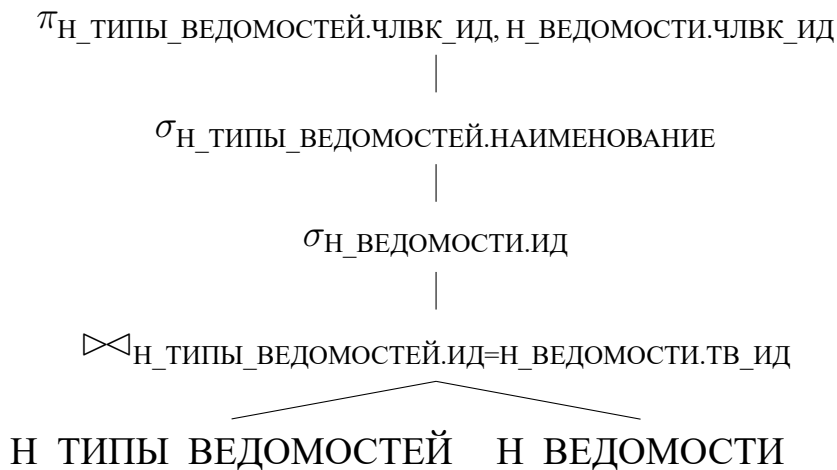
Индекс b-tree на атрибуте ИД уменьшит время выполнения запроса, потому что индекс b-tree хорошо подходит для операций выборок больше определенного значения, а так же таблица имеет большое количество строк до фильтрации и небольшое после

Хеш индекс не требуется этой таблице, потому что join будет производиться с помощью Hash join, а хешироваться будет таблица Н_ТИПЫ_ВЕДОМОСТЕЙ, потому что она меньше

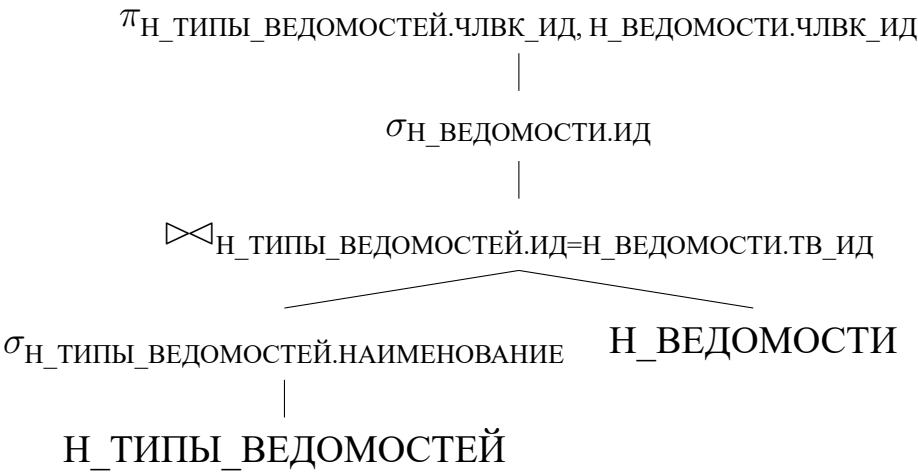
Индексы GiST, SP_GiST, GIN, BRIN не могут быть использованы из-за неподходящих условий выборки.

2.3 Планы выполнения запроса

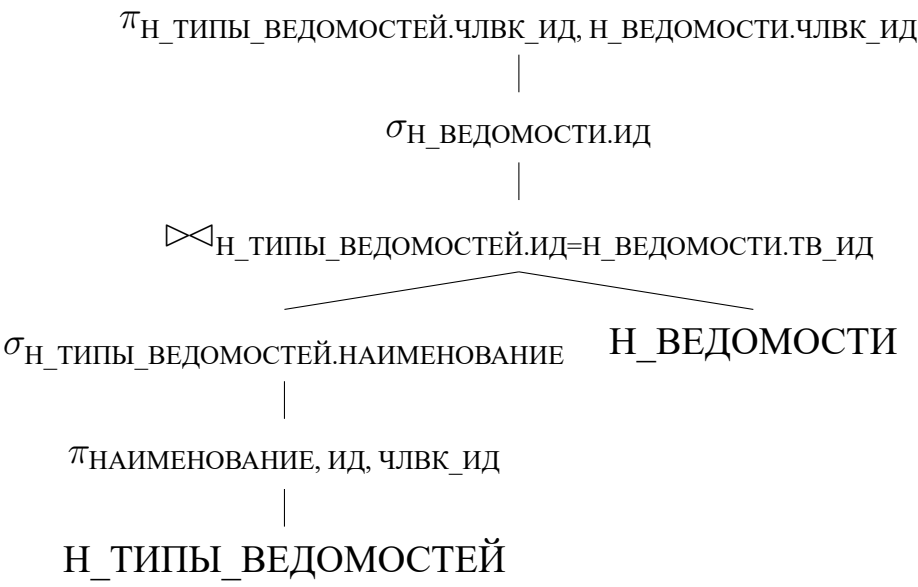
План 1



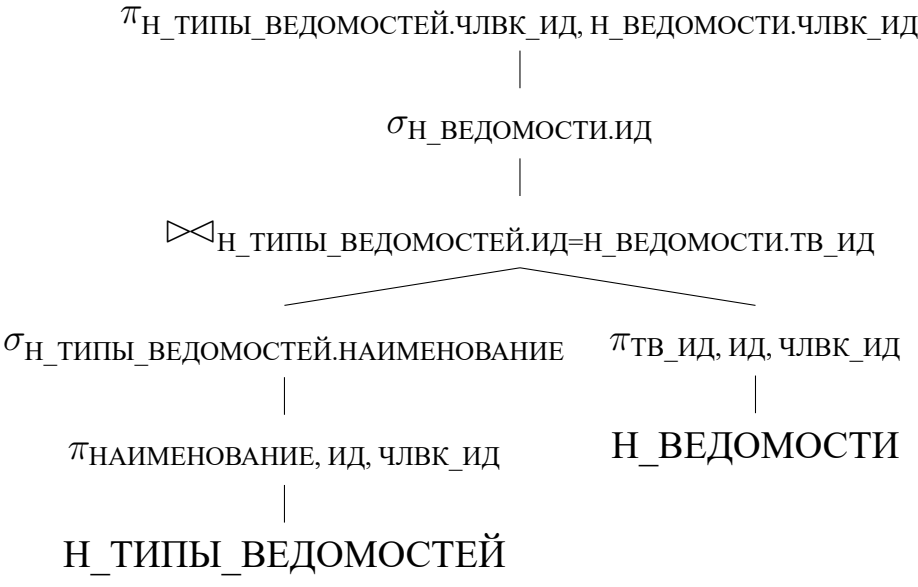
План 2



План 3



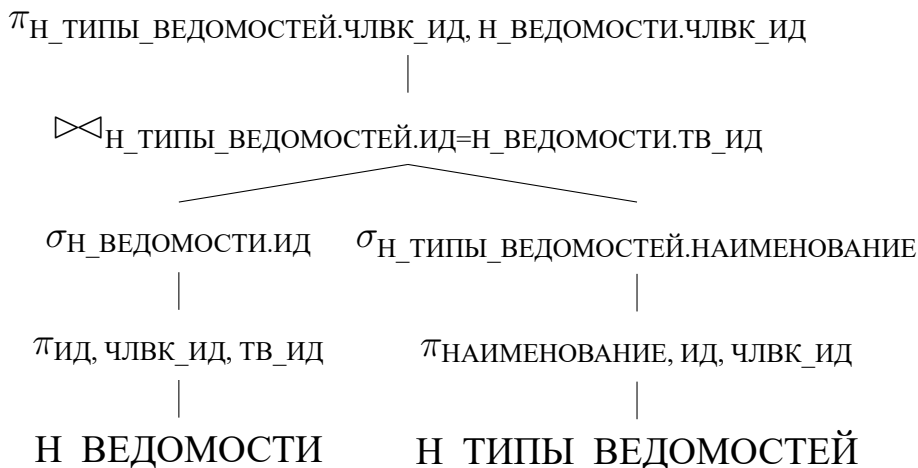
План 4



План 5



План 6



2.4 Оптимальный план

Самый оптимальный план - План 6, потому что он использует левостороннее дерево и все выборки и проекции сделаны максимально рано.

2.5 Выполнение команды EXPLAIN ANALYZE

```
Hash Join (cost=491.19..5047.40 rows=8676 width=8) (actual time=1.650..10.511 rows=6789 loops=1)
  Hash Cond: ("Н_ВЕДОМОСТИ"."ТВ_ИД" = "Н_ТИПЫ_ВЕДОМОСТЕЙ"."ИД")
    -> Bitmap Heap Scan on "Н_ВЕДОМОСТИ" (cost=490.14..4881.51 rows=26029 width=8) (actual
time=1.484..5.907 rows=26125 loops=1)
      Recheck Cond: ("ИД" > 1250981)
      Heap Blocks: exact=642
      -> Bitmap Index Scan on "ВЕД_ПК" (cost=0.00..483.64 rows=26029 width=0) (actual
time=1.376..1.376 rows=26125 loops=1)
        Index Cond: ("ИД" > 1250981)
    -> Hash (cost=1.04..1.04 rows=1 width=4) (actual time=0.065..0.066 rows=2 loops=1)
      Buckets: 1024 Batches: 1 Memory Usage: 9kB
      -> Seq Scan on "Н_ТИПЫ_ВЕДОМОСТЕЙ" (cost=0.00..1.04 rows=1 width=4) (actual
time=0.053..0.056 rows=2 loops=1)
        Filter: (("НАИМЕНОВАНИЕ")::text > 'Ведомость'::text)
        Rows Removed by Filter: 1
Planning Time: 1.005 ms
Execution Time: 10.903 ms
```

3 Запрос 2

3.1 Реализация запроса на SQL

```
SELECT
  Н_ЛЮДИ.ИМЯ,
  Н_ОБУЧЕНИЯ.ЧЛВК_ИД,
  Н_УЧЕНИКИ.НАЧАЛО
FROM
  "Н_ЛЮДИ"
RIGHT JOIN "Н_ОБУЧЕНИЯ"
ON
  Н_ЛЮДИ."ИД" = Н_ОБУЧЕНИЯ."ЧЛВК_ИД"
RIGHT JOIN "Н_УЧЕНИКИ"
ON
  Н_УЧЕНИКИ."ЧЛВК_ИД" = Н_ОБУЧЕНИЯ."ЧЛВК_ИД"
WHERE
  Н_ЛЮДИ.ФАМИЛИЯ < 'Елена'
  AND Н_ОБУЧЕНИЯ.ЧЛВК_ИД > 112514
  AND Н_УЧЕНИКИ.ИД > 100410;
```

3.2 Использование индексов

Таблица Н_ЛЮДИ

Индекс b-tree на атрибуте ФАМИЛИЯ уменьшит время выполнения запроса, потому что индекс b-tree хорошо подходит для операций выборки меньше определенного значения фильтрация обладает большой выборкой и убирает немалое количество строк

Хеш индексы этой таблице не требуются, для конвейерной обработки данных из-за использования индекса btree. Для hash join будет быстрее вызвать операцию hash, чем воспользоваться существующим индексом

Индексы GiST, SP_GiST, GIN, BRIN не могут быть использованы из-за неподходящих условий выборки.

Таблица Н_ОБУЧЕНИЯ

Индекс b-tree для атрибута ЧЛВК_ИД не уменьшит время выполнения запроса, потому что фильтр, хоть и обладает большой выборкой, но уберет 5% строк.

Хеш-индекс не требуется для таблицы Н_ОБУЧЕНИЯ, потому что при объединении будет использоваться hash-join, который захеширует меньшую таблицу ЛЮДИ

Индексы GiST, SP_GiST, GIN, BRIN не могут быть использованы из-за неподходящих условий выборки.

Таблица Н_УЧЕНИКИ

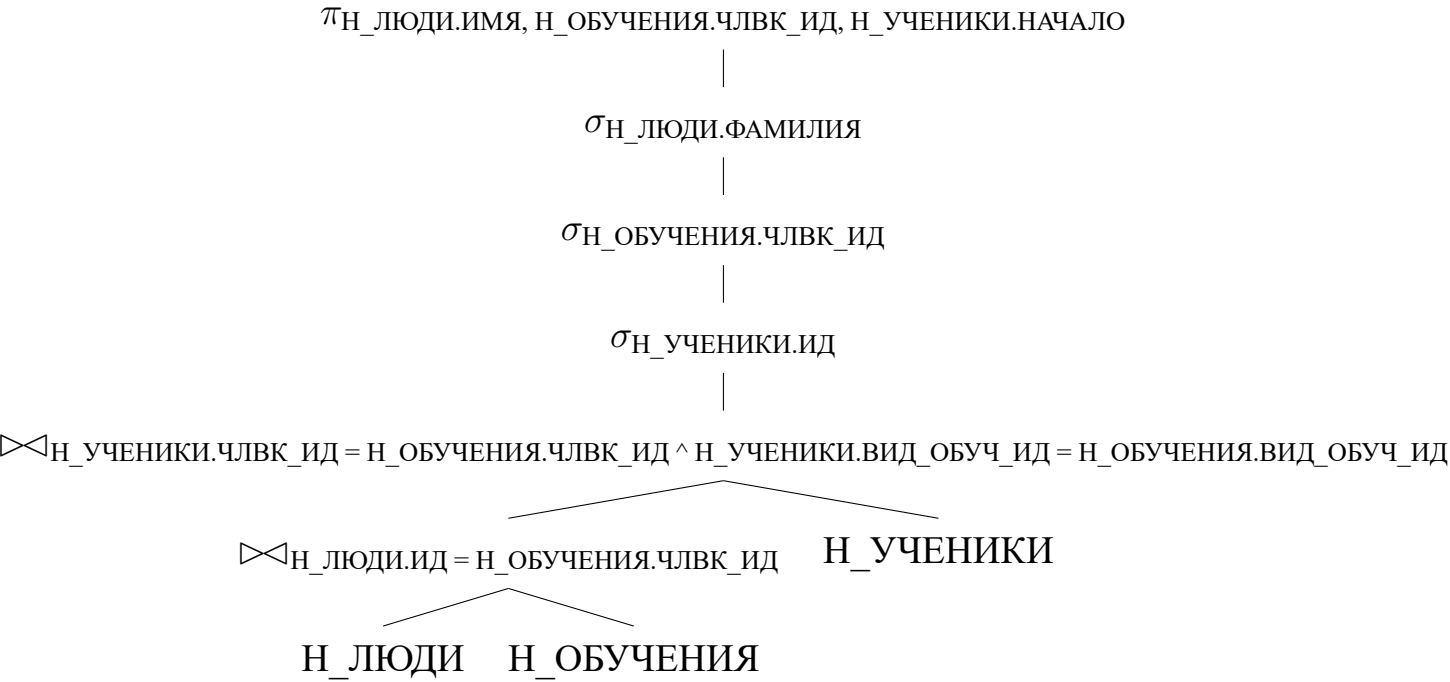
Индекс b-tree на атрибуте ИД не уменьшит время исполнения запроса, потому что фильтрация хоть и обладает большой выборкой, но оставит большое количество строк.

Хеш индекс не требуется для таблицы Н_УЧЕНИКИ, так как будет произведен hash-join, который будет хешировать меньшую таблицу

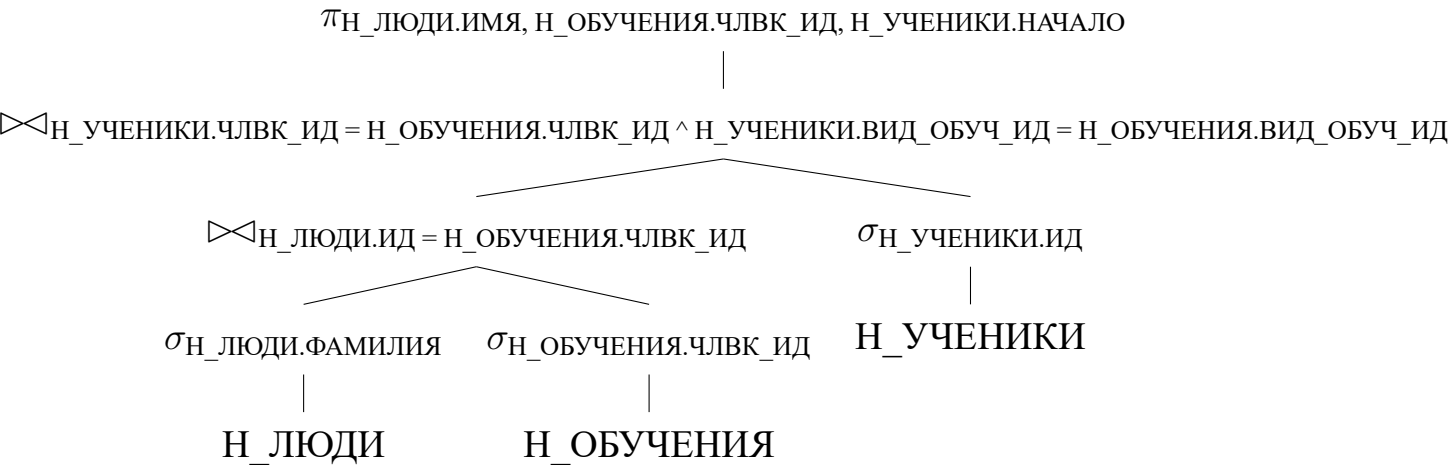
Индексы GiST, SP_GiST, GIN, BRIN не могут быть использованы из-за неподходящих условий выборки.

3.3 Планы выполнения запросов

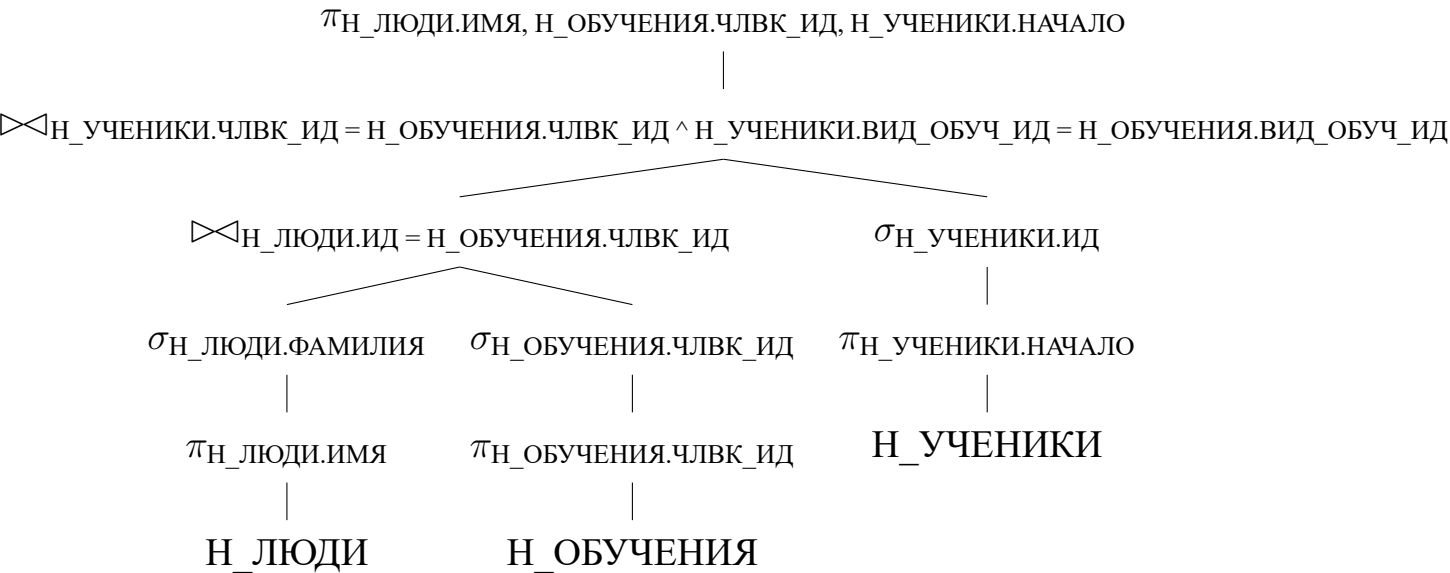
План 1



План 2



План 3



3.4 Оптимальный план

Самый оптимальный план - План 3, потому что он использует левостороннее дерево и все выборки и проекции сделаны максимально рано.

3.5 Выполнение команды EXPLAIN ANALYSE

```
Hash Join (cost=316.31..1213.73 rows=2565 width=25) (actual time=3.051..8.692 rows=2661 loops=1)
  Hash Cond: ("Н_УЧЕНИКИ"."ЧЛВК_ИД" = "Н_ЛЮДИ"."ИД")
  -> Seq Scan on "Н_УЧЕНИКИ" (cost=0.00..832.39 rows=10504 width=12) (actual time=0.087..3.943 rows=10511 loops=1)
    Filter: ("ИД" > 100410)
    Rows Removed by Filter: 12800
  -> Hash (cost=300.68..300.68 rows=1250 width=21) (actual time=2.946..2.949 rows=1238 loops=1)
    Buckets: 2048 Batches: 1 Memory Usage: 83kB
    -> Hash Join (cost=168.46..300.68 rows=1250 width=21) (actual time=1.172..2.695 rows=1238 loops=1)
      Hash Cond: ("Н_ОБУЧЕНИЯ"."ЧЛВК_ИД" = "Н_ЛЮДИ"."ИД")
      -> Seq Scan on "Н_ОБУЧЕНИЯ" (cost=0.00..119.76 rows=4741 width=4) (actual time=0.005..0.870 rows=4756 loops=1)
        Filter: ("ЧЛВК_ИД" > 112514)
        Rows Removed by Filter: 265
      -> Hash (cost=151.60..151.60 rows=1349 width=17) (actual time=1.145..1.147 rows=1339 loops=1)
        Buckets: 2048 Batches: 1 Memory Usage: 84kB
        -> Bitmap Heap Scan on "Н_ЛЮДИ" (cost=34.74..151.60 rows=1349 width=17) (actual time=0.455..0.891 rows=1339 loops=1)
          Recheck Cond: (("ФАМИЛИЯ")::text < 'Ёлкин')::text
          Heap Blocks: exact=99
        -> Bitmap Index Scan on "ФАМ_ЛЮД" (cost=0.00..34.40 rows=1349 width=0) (actual time=0.438..0.438 rows=1339 loops=1)
          Index Cond: (("ФАМИЛИЯ")::text < 'Ёлкин')::text

Planning Time: 2.194 ms
Execution Time: 8.926 ms
```

4 Вывод

При выполнении данной лабораторной работы я узнал понятие индексов в базах данных и как их использовать. Научился составлять планы и выбирать наиболее выгодный. Узнал про виды оптимизации соединения таблиц