

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

Факультет Программной инженерии и компьютерной техники

Отчёт по лабораторной работе №4  
Информационные системы и базы данных  
Вариант № 3199

Выполнил: студент группы Р3131

Серов А. А.

Проверил: Вербовой А.А.

Санкт-Петербург 2025г.

## 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:  
Таблицы: Н\_ЛЮДИ, Н\_СЕССИЯ.  
Вывести атрибуты: Н\_ЛЮДИ.ИД, Н\_СЕССИЯ.ИД.  
Фильтры (AND):  
а) Н\_ЛЮДИ.ИМЯ < Александр.  
б) Н\_СЕССИЯ.УЧГОД > 2001/2002.  
с) Н\_СЕССИЯ.УЧГОД = 2001/2002.  
Вид соединения: LEFT 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.ИД, Н\_ОБУЧЕНИЯ.ЧЛВК\_ИД, Н\_УЧЕНИКИ.ГРУППА.  
Фильтры: (AND)  
а) Н\_ЛЮДИ.ОТЧЕСТВО > Владимирович.  
б) Н\_ОБУЧЕНИЯ.НЗК > 999080.  
Вид соединения: LEFT JOIN.

## 2) Запросы на языке SQL:

**Первый запрос:**

```
SELECT
  Н_ЛЮДИ.ИД,
  Н_СЕССИЯ.ИД
FROM
  Н_ЛЮДИ
LEFT JOIN
  Н_СЕССИЯ ON Н_ЛЮДИ.ИД = Н_СЕССИЯ.ЧЛВК_ИД
WHERE
  Н_ЛЮДИ.ИМЯ < 'Александр'
  AND (Н_СЕССИЯ.УЧГОД > '2001/2002' OR Н_СЕССИЯ.УЧГОД = '2001/2002');
```

**Второй запрос:**

```
SELECT
  Н_ЛЮДИ.ИД,
  Н_ОБУЧЕНИЯ.ЧЛВК_ИД,
  Н_УЧЕНИКИ.ГРУППА
FROM
  Н_ЛЮДИ
LEFT JOIN
  Н_ОБУЧЕНИЯ ON Н_ЛЮДИ.ИД = Н_ОБУЧЕНИЯ.ЧЛВК_ИД
LEFT JOIN
  Н_УЧЕНИКИ ON Н_ОБУЧЕНИЯ.ЧЛВК_ИД = Н_УЧЕНИКИ.ЧЛВК_ИД
WHERE
  Н_ЛЮДИ.ОТЧЕСТВО > 'Владимирович'
  AND Н_ОБУЧЕНИЯ.НЗК > '999080';
```

### **3) Предложенные индексы**

**Для запроса 1:**

```
CREATE INDEX idx_сессия ON Н_СЕССИЯ(ЧЛВК_ИД, УЧГОД);
```

Тип индекса: B-tree (по умолчанию для PostgreSQL)

Польза для запроса:

Ускоряет соединение таблиц по полю ЧЛВК\_ИД (LEFT JOIN условие)

Оптимизирует фильтрацию по условиям на УЧГОД (оба условия в WHERE)

Эффективность: Составной индекс покрывает сразу две операции - JOIN и фильтрацию, что дает максимальный прирост производительности для данного запроса.

**Для запроса 2:**

```
CREATE INDEX idx_обучения ON Н_ОБУЧЕНИЯ(ЧЛВК_ИД, НЗК);
```

Тип индекса: B-tree (по умолчанию для PostgreSQL)

Польза для запроса:

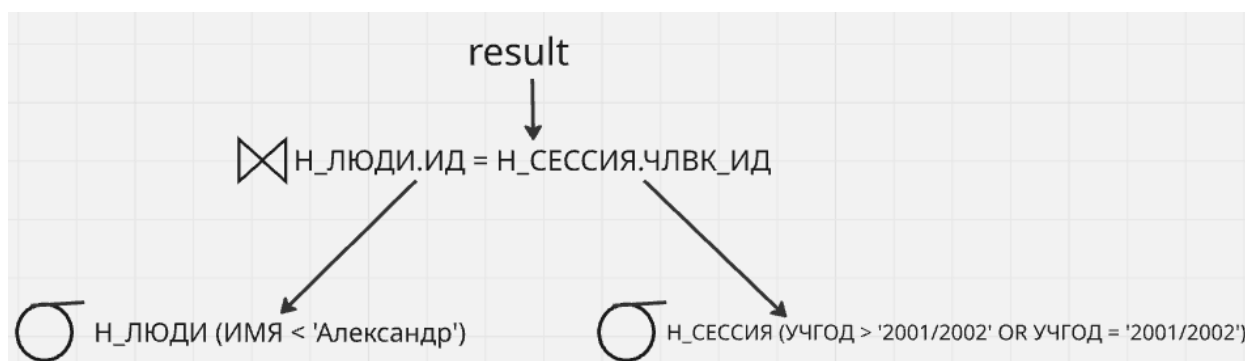
Оптимизирует соединение таблиц по полю ЧЛВК\_ИД (LEFT JOIN условие)

Ускоряет фильтрацию по условию НЗК > '999080'

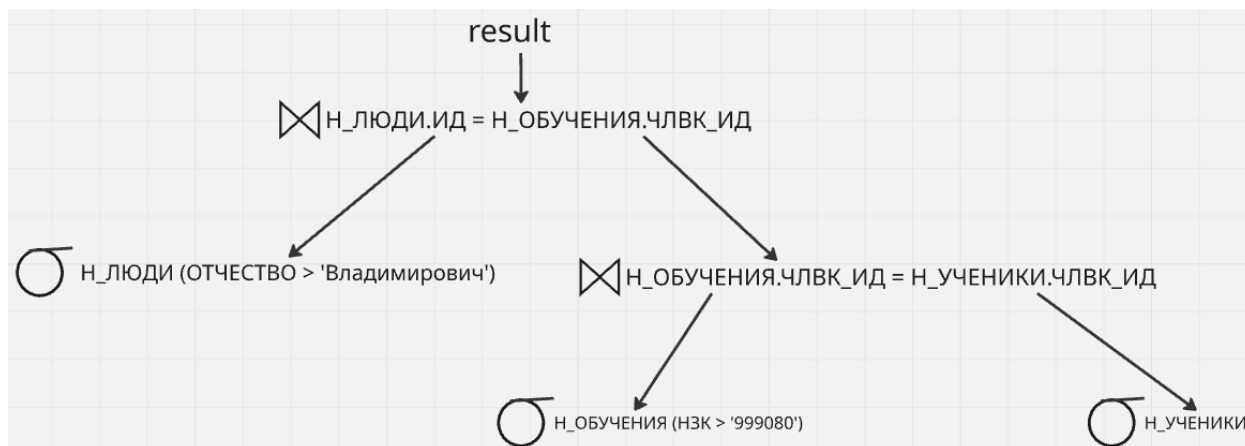
Эффективность: Один составной индекс заменяет два отдельных, эффективно обслуживая оба условия запроса. Особенно полезен, так как оба условия используются совместно в запросе.

#### 4) Возможные планы выполнения

**Запрос 1:**



**Запрос 2:**



## 5) EXPLAIN ANALYZE

### Запрос 1:

#### QUERY PLAN

Nested Loop (cost=0.29..295.94 rows=116 width=8) (actual time=0.815..3.882 rows=44 loops=1)

-> Seq Scan on "Н\_СЕССИЯ" (cost=0.00..127.28 rows=3634 width=8) (actual time=0.015..1.969 rows=3752 loops=1)

Filter: (((("УЧГОД")::text > '2001/2002')::text) OR ((("УЧГОД")::text = '2001/2002')::text))

-> Memoize (cost=0.29..0.44 rows=1 width=4) (actual time=0.000..0.000 rows=0 loops=3752)

Cache Key: "Н\_СЕССИЯ"."ЧЛВК\_ИД"

Cache Mode: logical

Hits: 3571 Misses: 181 Evictions: 0 Overflows: 0 Memory Usage: 13kB

-> Index Scan using "ЧЛВК\_ПК" on "Н\_ЛЮДИ" (cost=0.28..0.43 rows=1 width=4) (actual time=0.003..0.003 rows=0 loops=181)

Index Cond: ("ИД" = "Н\_СЕССИЯ"."ЧЛВК\_ИД")

Filter: (("ИМЯ")::text < 'Александр')::text)

Rows Removed by Filter: 1

Planning Time: 0.491 ms

Execution Time: 3.943 ms

(13 строк)

## Запрос 2:

### QUERY PLAN

---

Nested Loop Left Join (cost=4.61..151.11 rows=5 width=12) (actual time=1.219..1.221 rows=0 loops=1)

- > Nested Loop (cost=0.28..128.07 rows=1 width=8) (actual time=1.219..1.220 rows=0 loops=1)
  - > Seq Scan on "Н\_ОБУЧЕНИЯ" (cost=0.00..119.76 rows=1 width=4) (actual time=1.218..1.219 rows=0 loops=1)
    - Filter: (("НЗК")::text > '999080'::text)
    - Rows Removed by Filter: 5021
  - > Index Scan using "ЧЛВК\_ПК" on "Н\_ЛЮДИ" (cost=0.28..8.30 rows=1 width=4) (never executed)
    - Index Cond: ("ИД" = "Н\_ОБУЧЕНИЯ"."ЧЛВК\_ИД")
    - Filter: (("ОТЧЕСТВО")::text > 'Владимирович'::text)
- > Bitmap Heap Scan on "Н\_УЧЕНИКИ" (cost=4.33..22.99 rows=5 width=8) (never executed)
  - Recheck Cond: ("Н\_ОБУЧЕНИЯ"."ЧЛВК\_ИД" = "ЧЛВК\_ИД")
  - > Bitmap Index Scan on "УЧЕН\_ОБУЧ\_FK\_I" (cost=0.00..4.32 rows=5 width=0) (never executed)
    - Index Cond: ("ЧЛВК\_ИД" = "Н\_ОБУЧЕНИЯ"."ЧЛВК\_ИД")

Planning Time: 0.743 ms

Execution Time: 1.283 ms

(14 строк)

## 6) Вывод

В результате выполнения лабораторной работы я вспомнил, как делать запросы на SQL, научился составлять планы их выполнения, а также смог составить для каждого из запросов EXPLAIN ANALYZE.