



Факультет программной инженерии и компьютерной техники  
Информационные системы и базы данных

Лабораторная работа № 4  
Вариант 1151

Преподаватель: Харитонов А. Е.  
Выполнила: Блинова М.А.  
Группа: Р33301

## 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:

Н\_ТИПЫ\_ВЕДОМОСТЕЙ, Н\_ВЕДОМОСТИ.

Вывести атрибуты: Н\_ТИПЫ\_ВЕДОМОСТЕЙ.НАИМЕНОВАНИЕ, Н\_ВЕДОМОСТИ.ЧЛВК\_ИД.

Фильтры (AND):

- a) Н\_ТИПЫ\_ВЕДОМОСТЕЙ.ИД = 1.
- b) Н\_ВЕДОМОСТИ.ЧЛВК\_ИД < 163249.
- c) Н\_ВЕДОМОСТИ.ЧЛВК\_ИД < 105590.

Вид соединения: INNER 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.ОТЧЕСТВО, Н\_ОБУЧЕНИЯ.НЗК, Н\_УЧЕНИКИ.НАЧАЛО.

Фильтры: (AND)

- a) Н\_ЛЮДИ.ФАМИЛИЯ > Соколов.
- b) Н\_ОБУЧЕНИЯ.ЧЛВК\_ИД < 105590.

Вид соединения: LEFT JOIN.

## Реализация запросов на SQL:

```
1. SELECT type.НАИМЕНОВАНИЕ, ved.ЧЛВК_ИД FROM Н_ТИПЫ_ВЕДОМОСТЕЙ AS type JOIN
Н_ВЕДОМОСТИ AS ved ON type.ИД = ved.ТВ_ИД WHERE (

    type.ИД = 1 AND

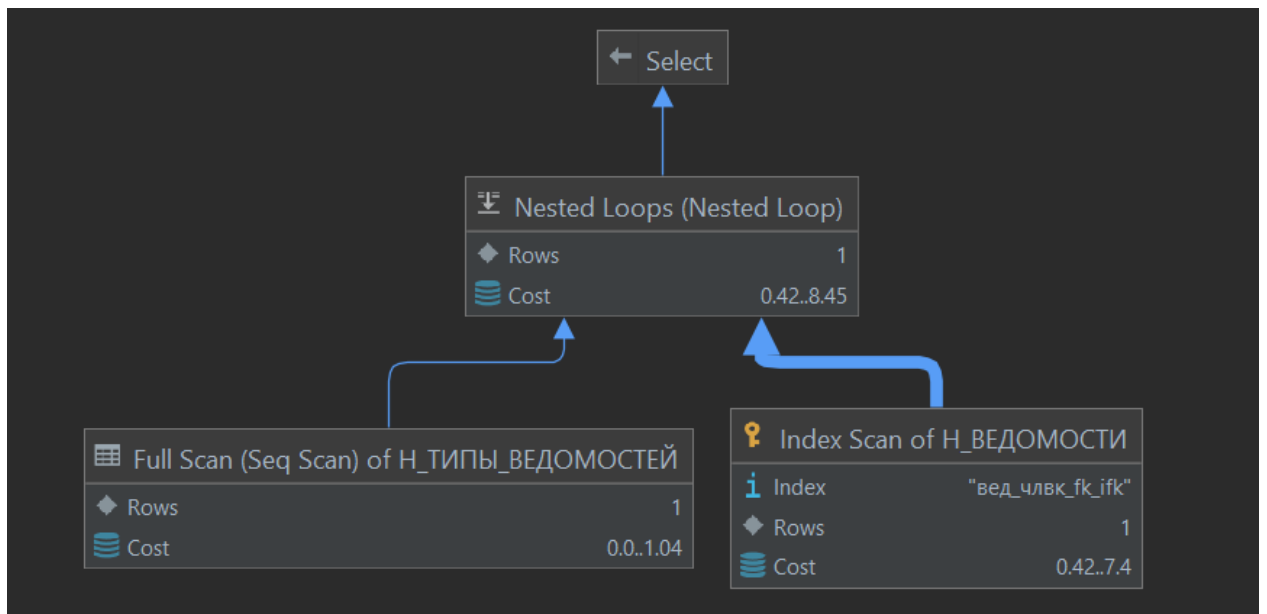
    ved.ЧЛВК_ИД < 163249 AND

    ved.ЧЛВК_ИД < 105590

);
```

```

1
-----
QUERY PLAN
-----
Nested Loop  (cost=0.42..8.45 rows=1 width=422) (actual time=0.015..0.015 rows=0 loops=1)
-> Seq Scan on "Н_ТИПЫ_ВЕДОМОСТЕЙ" type  (cost=0.00..1.04 rows=1 width=422) (actual time=0.008..0.009 rows=1 loops=1)
    Filter: ("ИД" = 1)
    Rows Removed by Filter: 2
-> Index Scan using "ВЕД_ЧЛВК_FK_IFK" on "Н_ВЕДОМОСТИ" ved  (cost=0.42..7.40 rows=1 width=8) (actual time=0.004..0.004 rows=0 loops=1)
    Index Cond: (("ЧЛВК_ИД" < 163249) AND ("ЧЛВК_ИД" < 105590))
    Filter: ("ТВ_ИД" = 1)
Planning time: 0.200 ms
Execution time: 0.074 ms
(9 rows)
```



Оптимизировать этот запрос не нужно. Для таблицы “Н\_ВЕДОМОСТИ” планировщик сам создает индекс по столбцу «ЧЛВК\_ИД», по которому идет отбор данных, значит здесь создание еще какого-либо индекса не имеет смысла

“Н\_ТИПЫ\_ВЕДОМОСТЕЙ” сканируется полностью, тем не менее индекс таблице не нужен, т.к. она содержит всего 3 записи и навряд ли их количество изменится. Поэтому добавление индекса здесь будет невыгодно.

2. **SELECT** people.ОТЧЕСТВО, studies.НЗК, students.НАЧАЛО **FROM** Н\_ЛЮДИ **AS** people **LEFT JOIN** Н\_ОБУЧЕНИЯ **AS** studies **ON** people.ИД = studies.ЧЛВК\_ИД **LEFT JOIN** Н\_УЧЕНИКИ **AS** students **ON** students.ЧЛВК\_ИД = people.ИД **WHERE** (

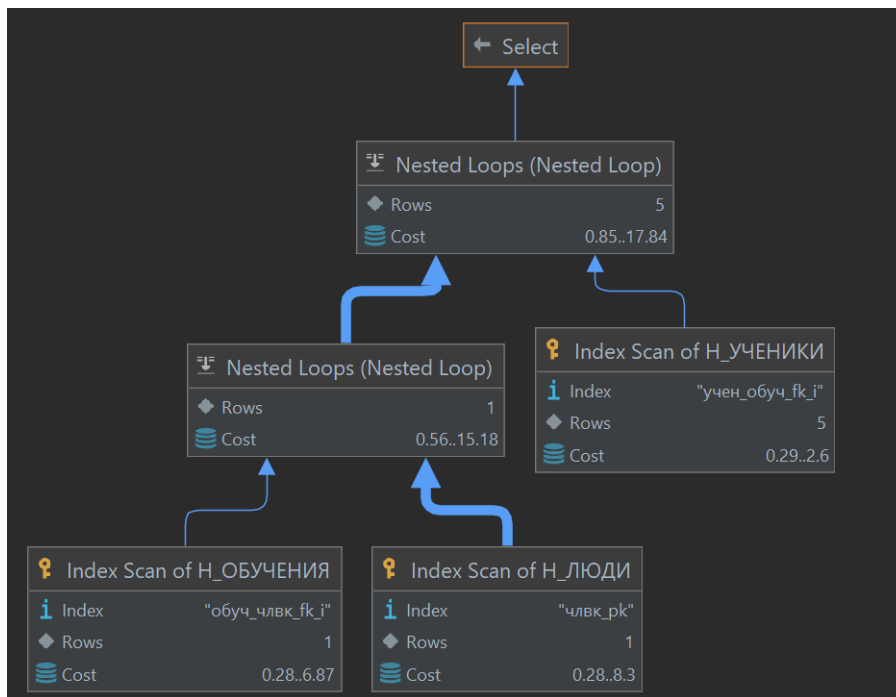
people.ФАМИЛИЯ > 'Соколов' **AND**

studies.ЧЛВК\_ИД < 105590

);

```

----- QUERY PLAN -----
Nested Loop Left Join (cost=0.85..17.84 rows=5 width=34) (actual time=0.005..0.005 rows=0 loops=1)
-> Nested Loop (cost=0.56..15.18 rows=1 width=30) (actual time=0.004..0.004 rows=0 loops=1)
-> Index Scan using "ОБУЧ_ЧЛВК_ФК_I" on "Н_ОБУЧЕНИЯ" studies (cost=0.28..6.87 rows=1 width=10) (actual time=0.003..0.003 rows=0 loops=1)
    Index Cond: ("ЧЛВК_ИД" < 105590)
-> Index Scan using "ЧЛВК_РК" on "Н_ЛЮДИ" people (cost=0.28..8.30 rows=1 width=24) (never executed)
    Index Cond: ("ИД" = studies."ЧЛВК_ИД")
    Filter: (("ФАМИЛИЯ")::text > 'Соколов'::text)
-> Index Scan using "УЧЕН_ОБУЧ_ФК_I" on "Н_УЧЕНИКИ" students (cost=0.29..2.60 rows=5 width=12) (never executed)
    Index Cond: ("ЧЛВК_ИД" = people."ИД")
Planning time: 2.309 ms
Execution time: 0.139 ms
(11 rows)
  
```



Здесь мы имеем обращение к трём таблицам, при этом, составных условий для какого-либо из полей нет.

В “H\_ОБУЧЕНИЯ” индекс уже построен по необходимому полю (ЧЛВК\_ИД, которое нам и нужно).

Для таблицы “H\_ЛЮДИ” выгоднее использовать индекс по “ФАМИЛИЯ”. Потому что в условии мы выбираем все фамилии больше 'Соколов', их может быть очень много и с помощью индекса по “ФАМИЛИЯ” они обработаются быстрее чем по “ЧЛВК\_ИД”.

Последняя таблица – “H\_УЧЕНИКИ”, в которой планировщик так же сам предлагает построить нужный индекс (ЧЛВК\_ИД).

Поэтому, здесь оптимизация запроса так же не требуется.

### Вывод:

В процессе выполнения лабораторной я немного научилась оптимизировать БД путём создания индексов – структур данных, распределяющих записи из таблиц по заданному полю, что ускоряет обращение к ним, а также в процессе выборки данных планировщиком.