Федеральное государственное автономное образовательное учреждение высшего

образования «Национальный исследовательский университет ИТМО»

Факультет программной инженерии и компьютерной техники

**Отчет**

**По лабораторной работе №4**

По дисциплине «Базы данных»

Вариант 1341

Выполнил: Чураков Александр Алексеевич, группа P3131

Преподаватель: Харитонова Анастасия Евгеньевна

Санкт-Петербург

~ 2024 ~

# 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.  
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:  
   Н\_ТИПЫ\_ВЕДОМОСТЕЙ, Н\_ВЕДОМОСТИ.  
   Вывести атрибуты: Н\_ТИПЫ\_ВЕДОМОСТЕЙ.НАИМЕНОВАНИЕ, Н\_ВЕДОМОСТИ.ЧЛВК\_ИД.  
   Фильтры (AND):  
   a) Н\_ТИПЫ\_ВЕДОМОСТЕЙ.НАИМЕНОВАНИЕ < Ведомость.  
   b) Н\_ВЕДОМОСТИ.ИД < 39921.  
   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.ИД, Н\_ВЕДОМОСТИ.ИД, Н\_СЕССИЯ.ЧЛВК\_ИД.  
   Фильтры (AND):  
   a) Н\_ЛЮДИ.ИД = 152862.  
   b) Н\_ВЕДОМОСТИ.ЧЛВК\_ИД < 153285.  
   c) Н\_СЕССИЯ.ДАТА = 2012-01-25.  
   Вид соединения: INNER JOIN.

# Реализация на SQL

1)

explain analyse  
select t."НАИМЕНОВАНИЕ", v."ЧЛВК\_ИД"  
from "Н\_ТИПЫ\_ВЕДОМОСТЕЙ" t  
 left join "Н\_ВЕДОМОСТИ" v on v."ТВ\_ИД" = t."ИД"  
where "НАИМЕНОВАНИЕ" < 'Ведомость'  
 and v."ИД" < 39921  
;

2)

explain analyse  
select p."ИД" as people\_id, v."ИД" as ved\_id, s."ЧЛВК\_ИД" as session\_man\_id  
from "Н\_ЛЮДИ" p  
 inner join "Н\_ВЕДОМОСТИ" v on v."ЧЛВК\_ИД" = p."ИД"  
 inner join "Н\_СЕССИЯ" s on s."ЧЛВК\_ИД" = p."ИД"  
where p."ИД" = 152862  
 and v."ЧЛВК\_ИД" < 153285  
 and s."ДАТА" = '2012-01-25'

# Уменьшение времени выполнения запросов

## Запрос №1

### Индексы, добавление которых уменьшит время выполнения запроса

Таблица Н\_ТИПЫ\_ВЕДОМОСТИ:

Индекс на атрибуте НАИМЕНОВАНИЕ ускорит фильтрацию строк с условием   
Н\_ТИПЫ\_ВЕДОМОСТЕЙ.НАИМЕНОВАНИЕ < Ведомость.

### Возможные планы выполнения без индексов

**План 1**

1. Сначала происходит последовательное сканирование таблицы "Н\_ТИПЫ\_ВЕДОМОСТЕЙ" с применением фильтра Н\_ТИПЫ\_ВЕДОМОСТЕЙ.НАИМЕНОВАНИЕ < Ведомость.
2. Затем для каждой строки из таблицы Н\_ТИПЫ\_ВЕДОМОСТЕЙ происходит выполнение вложенного цикла.
3. Для каждой строки из таблицы Н\_ТИПЫ\_ВЕДОМОСТЕЙ происходит поиск соответствующих строк в таблице "Н\_ВЕДОМОСТИ”.
4. Применяется фильтр к результатам сканирования таблицы Н\_ВЕДОМОСТИ: ""ИД"" < 39921.

**План 2**

1. Последовательное сканирование таблицы "Н\_ВЕДОМОСТИ" с фильтром "ИД" < 39921.
2. Последовательное сканирование таблицы "Н\_ТИПЫ\_ВЕДОМОСТЕЙ"
3. Затем происходит хеширование результатов сканирования таблицы "Н\_ТИПЫ\_ВЕДОМОСТЕЙ" на основе значения столбца ""ИД".
4. Выполняется хеш-объединение между результатами сканирования таблицы "Н\_ВЕДОМОСТИ" и хешированными результатами сканирования таблицы "Н\_ТИПЫ\_ВЕДОМОСТЕЙ". Объединение происходит по условию "v.""ТВ\_ИД"" = t.""ИД"".

Оптимальным является план 2, так как фильтр по ИД выполняется до соединения таблиц, что уменьшает количество строк для обработки.

При добавлении индексов вместо полного скана таблиц будет использоваться индексный скан

### План выполнения

Удалим условие "НАИМЕНОВАНИЕ" < 'Ведомость' и увеличим максимальный ID иначе нет подходящих значений.

Hash Join (cost=1.07..7468.62 rows=98061 width=30) (actual time=2.036..56.851 rows=98307 loops=1)  
" Hash Cond: (v.""ТВ\_ИД"" = t.""ИД"")"  
" -> Seq Scan on ""Н\_ВЕДОМОСТИ"" v (cost=0.00..6846.50 rows=98061 width=8) (actual time=2.015..30.363 rows=98307 loops=1)"  
" Filter: (""ИД"" < 399220)"  
 Rows Removed by Filter: 124133  
 -> Hash (cost=1.03..1.03 rows=3 width=30) (actual time=0.012..0.014 rows=3 loops=1)  
 Buckets: 1024 Batches: 1 Memory Usage: 9kB  
" -> Seq Scan on ""Н\_ТИПЫ\_ВЕДОМОСТЕЙ"" t (cost=0.00..1.03 rows=3 width=30) (actual time=0.004..0.005 rows=3 loops=1)"  
Planning Time: 0.300 ms  
Execution Time: 61.568 ms

Изображение выглядит как текст, снимок экрана, Шрифт, линия

Автоматически созданное описание

# Запрос №2

## Индексы, добавление которых уменьшит время выполнения запроса

Индекс на атрибуте ДАТА таблицы Н\_СЕССИЯ ускорит фильтрацию строк с условием Н\_СЕССИЯ.ДАТА = 2012-01-25.

### Возможные планы выполнения без индексов

**План 1**

1. Полный скан таблицы Н\_ЛЮДИ и поиск строки с ИД=1528623
2. Полный скан таблицы Н\_ВЕДОМОСТИ с применением фильтра v."ЧЛВК\_ИД" < 153285
3. Полный скан таблицы Н\_СЕССИЯ с применением фильтра s."ДАТА" = '2012-01-25'
4. Соединение таблиц с использованием Hash Join по атрибутам

**План 2**

1. Сначала выполняется вложенная операция "Nested Loop", которая состоит из двух частей:

а) Сканирование таблицы "Н\_ЛЮДИ" (обозначается как p). Поиск выполняется по ключу "ИД", равному 1528623.

б) Битовое сканирование кучи таблицы "Н\_СЕССИЯ" (обозначается как s). Производится поиск записей, где значение поля "ЧЛВК\_ИД" равно 1528623, и где дата равна '2012-01-25'.

2. Затем результаты вложенной операции "Nested Loop" объединяются с помощью еще одной операции "Nested Loop". Для каждой строки, полученной из предыдущей операции, выполняется сканирование таблицы "Н\_ВЕДОМОСТИ" (обозначается как v). Поиск выполняется по ключу "ЧЛВК\_ИД", который меньше 153285 и равен 1528623.

Оптимальным является план 2, потому что фильтрация данных выполняется до соединения таблиц, что уменьшает количество строк для обработки

При добавлении индексов вместо полного скана таблиц будет использоваться индексный скан

### План выполнения

Удалим "ДАТА" = '2012-01-25' из фильтра и изменим условие на "ИД" = 110136, чтобы результат запроса был не пустым.

Nested Loop (cost=5.18..297.59 rows=2604 width=12) (actual time=0.063..0.294 rows=798 loops=1)  
" -> Index Scan using ""ВЕД\_ЧЛВК\_FK\_IFK"" on ""Н\_ВЕДОМОСТИ"" v (cost=0.29..190.58 rows=62 width=8) (actual time=0.022..0.030 rows=19 loops=1)"  
" Index Cond: ((""ЧЛВК\_ИД"" < 153285) AND (""ЧЛВК\_ИД"" = 110136))"  
 -> Materialize (cost=4.89..74.57 rows=42 width=8) (actual time=0.002..0.008 rows=42 loops=19)  
 -> Nested Loop (cost=4.89..74.36 rows=42 width=8) (actual time=0.036..0.081 rows=42 loops=1)  
" -> Index Only Scan using ""ЧЛВК\_PK"" on ""Н\_ЛЮДИ"" p (cost=0.28..4.30 rows=1 width=4) (actual time=0.010..0.011 rows=1 loops=1)"  
" Index Cond: (""ИД"" = 110136)"  
 Heap Fetches: 0  
" -> Bitmap Heap Scan on ""Н\_СЕССИЯ"" s (cost=4.61..69.64 rows=42 width=4) (actual time=0.020..0.058 rows=42 loops=1)"  
" Recheck Cond: (""ЧЛВК\_ИД"" = 110136)"  
 Heap Blocks: exact=21  
" -> Bitmap Index Scan on ""SYS\_C003500\_IFK"" (cost=0.00..4.60 rows=42 width=0) (actual time=0.012..0.012 rows=42 loops=1)"  
" Index Cond: (""ЧЛВК\_ИД"" = 110136)"  
Planning Time: 0.241 ms  
Execution Time: 0.376 ms

Изображение выглядит как текст, Шрифт, число, снимок экрана

Автоматически созданное описание

# Вывод

В ходе выполнения лабораторной работы я освоил работу с реляционной алгеброй и научился строить планы выполнения запросов, а также их диаграммы. Я изучил различные виды индексов и узнал, как использовать их для оптимизации запросов