САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ

ИССЛЕДОВАТЕЛЬСКАЯ

НАЦИОНАЛЬНАЯ НАУЧНО-ОБРАЗОВАТЕЛЬНАЯ

КОРПОРАЦИЯ ИТМО

Факультет программной инженерии и компьютерной техники

Направление подготовки 09.03.04 Программная инженерия

Дисциплина «Информационные системы и базы данных»

**Отчет**

По лабораторной работе №4

Вариант 3111457225

Студент

*Казаев Максим*

*Р3111*

Преподаватель

*Николаев Владимир Вячеславович*

Практик

*Чупанов Аликылыч Алибекович*

Санкт-Петербург, 2023 г.

Оглавление

[Текст задания 2](#_Toc137138592)

[Запросы 3](#_Toc137138593)

[Уменьшение времени исполнения 1го запроса 3](#_Toc137138594)

[Индексы 3](#_Toc137138595)

[Возможные планы выполнения 3](#_Toc137138596)

[План 1: 3](#_Toc137138597)

[План 2: 4](#_Toc137138598)

[Оптимальный план: 5](#_Toc137138599)

[Изменения с добавлением индексов 5](#_Toc137138600)

[Объяснение 5](#_Toc137138601)

[Уменьшение времени исполнения 2го запроса 6](#_Toc137138602)

[Индексы 6](#_Toc137138603)

[Возможные планы выполнения 6](#_Toc137138604)

[План 1: 6](#_Toc137138605)

[План 2: 6](#_Toc137138606)

[Оптимальный план: 7](#_Toc137138607)

[Изменения с добавлением индексов 7](#_Toc137138608)

[Объяснение 7](#_Toc137138609)

[Вывод 8](#_Toc137138610)

# 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.  
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.ИД, Н\_ВЕДОМОСТИ.ДАТА.  
   Фильтры (AND):   
   a) Н\_ТИПЫ\_ВЕДОМОСТЕЙ.НАИМЕНОВАНИЕ > Экзаменационный лист.  
   b) Н\_ВЕДОМОСТИ.ЧЛВК\_ИД = 142390.  
   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.ОТЧЕСТВО, Н\_ВЕДОМОСТИ.ИД, Н\_СЕССИЯ.ЧЛВК\_ИД.  
   Фильтры (AND):   
   a) Н\_ЛЮДИ.ИД > 100865.  
   b) Н\_ВЕДОМОСТИ.ЧЛВК\_ИД = 105590.  
   c) Н\_СЕССИЯ.ДАТА = 2004-01-17.  
   Вид соединения: RIGHT JOIN.

# Запросы

1) SELECT Н\_ТИПЫ\_ВЕДОМОСТЕЙ.ИД, Н\_ВЕДОМОСТИ.ДАТА

FROM Н\_ТИПЫ\_ВЕДОМОСТЕЙ

LEFT JOIN Н\_ВЕДОМОСТИ ON Н\_ТИПЫ\_ВЕДОМОСТЕЙ.ИД = Н\_ВЕДОМОСТИ.ТВ\_ИД

WHERE Н\_ТИПЫ\_ВЕДОМОСТЕЙ.НАИМЕНОВАНИЕ > 'Экзаменационный лист'

AND Н\_ВЕДОМОСТИ.ЧЛВК\_ИД = 142390;

2) SELECT Н\_ЛЮДИ.ОТЧЕСТВО, Н\_ВЕДОМОСТИ.ИД, Н\_СЕССИЯ.ЧЛВК\_ИД

FROM Н\_ЛЮДИ

RIGHT JOIN Н\_ВЕДОМОСТИ ON Н\_ЛЮДИ.ИД=Н\_ВЕДОМОСТИ.ЧЛВК\_ИД

RIGHT JOIN Н\_СЕССИЯ ON Н\_ЛЮДИ.ИД = Н\_СЕССИЯ.ЧЛВК\_ИД

WHERE Н\_ЛЮДИ.ИД > 100865 AND Н\_ВЕДОМОСТИ.ЧЛВК\_ИД = 105590 AND Н\_СЕССИЯ.ДАТА = '2004-01-17';

# Уменьшение времени исполнения 1-го запроса

## Индексы

Н\_ТИПЫ\_ВЕДОМОСТЕЙ:

1. Индекс на атрибут Н\_ТИПЫ\_ВЕДОМОСТИ.ИД (Hash) для ускорения объединения таблиц
2. Индекс на атрибут Н\_ТИПЫ\_ВЕДОМОСТЕЙ.НАИМЕНОВАНИЕ (B-tree) его можно убрать, так как он замедлит выполнение запроса( 3 элемента)

Н\_ВЕДОМОСТИ:

1. Индекс на атрибут Н\_ВЕДОМОСТИ.ТВ\_ИД (Hash) для ускорения объединения таблиц
2. Индекс на атрибут Н\_ВЕДОМОСТИ.ЧЛВК\_ИД (B-tree) для ускорения фильтрации

## Возможные планы выполнения

### План 1:

1. Полный скан Н\_ТИПЫ\_ВЕДОМОСТИ с применением фильтра Н\_ТИПЫ\_ВЕДОМОСТИ > Экзаменационный лист
2. Полный скан Н\_ ВЕДОМОСТИ с применением фильтра Н\_ ВЕДОМОСТИ = 142390
3. Соединение таблиц с использованием Nested Loops JOIN по атрибутам ИД, ТВ\_ИД

Изображение выглядит как текст, рукописный текст, бумага, блокнот

Автоматически созданное описание

### План 2:

1. Соединение таблиц с использованием Nested Loops JOIN по атрибутам ИД, ТВ\_ИД
2. Полный скан Н\_ТИПЫ\_ВЕДОМОСТИ с применением фильтра Н\_ТИПЫ\_ВЕДОМОСТИ > Экзаменационный лист
3. Полный скан Н\_ ВЕДОМОСТИ с применением фильтра Н\_ ВЕДОМОСТИ = 142390

Изображение выглядит как текст, рукописный текст, бумага, блокнот

Автоматически созданное описание

### Оптимальный план:

План 1 будет актуальнее, т. к. данные фильтруются до соединения таблиц, что уменьшает количество строк для обработки.

### Изменения с добавлением индексов

1. Вместо полного скана будет использован индексный скан
2. Nested Loops JOIN будет произведен быстрее, благодаря индексу по атрибутам ИД и ТВ\_ИД.

### Объяснение

Nested Loop (cost=0.29..197.63 rows=21 width=12) (actual time=0.076..0.077 rows=0 loops=1)

Join Filter: ("Н\_ТИПЫ\_ВЕДОМОСТЕЙ"."ИД" = "Н\_ВЕДОМОСТИ"."ТВ\_ИД")

-> Seq Scan on "Н\_ТИПЫ\_ВЕДОМОСТЕЙ" (cost=0.00..1.04 rows=1 width=4) (actual time=0.075..0.076 rows=0 loops=1)

Filter: (("НАИМЕНОВАНИЕ")::text > 'Экзаменационный лист'::text)

Rows Removed by Filter: 3

-> Index Scan using "ВЕД\_ЧЛВК\_FK\_IFK" on "Н\_ВЕДОМОСТИ" (cost=0.29..195.79 rows=64 width=12) (never executed)

Index Cond: ("ЧЛВК\_ИД" = 142390)

Planning Time: 0.266 ms

Execution Time: 0.112 ms

# Уменьшение времени исполнения 2-го запроса

## Индексы

Н\_ЛЮДИ:

1. Индекс на атрибут ИД(Hash) для ускорения объединения

Н\_СЕССИЯ:

1. Индекс на атрибут ЧЛВК\_ИД(Hash) для ускорения объединения
2. Индекс на атрибут Дата(B-tree) для ускорения фильтрации

Н\_ВЕДОМОСТИ:

1. Индекс на атрибут ЧЛВК\_ИД(Hash) для ускорения объединения

## Возможные планы выполнения

### План 1:

1. Полный скан Н\_СЕССИЯ
2. Полный скан Н\_ЛЮДИ
3. Полный скан Н\_ВЕДОМОСТИ
4. Объеденить с помощью Nested Loop JOIN Н\_ЛЮДИ c Н\_ВЕДОМОСТИ по Н\_ЛЮДИ.ИД= Н\_ВЕДОМОСТИ.ЧЛВК\_ИД
5. Объеденить с помощью Nested Loop JOIN результат c Н\_СЕССИЯ Н\_ЛЮДИ.ИД= Н\_СЕССИЯ.ЧЛВК\_ИД
6. Полный скан Н\_ВЕДОМОСТИ с применением фильтра Н\_ВЕДОМОСТИ.ЧЛВК\_ИД = 105590
7. Полный скан Н\_Люди с применением фильтра Н\_ЛЮДИ.ИД = 100865
8. Полный скан Н \_СЕССИЯ с применением фильтра Н \_СЕССИЯ.ДАТА = 2004.01.17

Изображение выглядит как текст, рукописный текст, бумага, Бумажное изделие

Автоматически созданное описание

### План 2:

1. Полный скан Н \_СЕССИЯ Н \_СЕССИЯ.ЧЛВК\_ИД , с применением фильтра Н \_СЕССИЯ.ДАТА = 2004.01.17
2. Полный скан Н\_Люди Н\_ЛЮДИ.ОТЧЕСТВО, с применением фильтра Н\_ЛЮДИ.ИД = 100865
3. Полный скан Н\_ВЕДОМОСТИ с применением фильтра Н\_ВЕДОМОСТИ.ЧЛВК\_ИД = 105590
4. Объеденить с помощью Nested Loop JOIN Н\_ЛЮДИ c Н\_ВЕДОМОСТИ по Н\_ЛЮДИ.ИД= Н\_ВЕДОМОСТИ.ЧЛВК\_ИД
5. Объеденить с помощью Nested Loop JOIN результат c Н\_СЕССИЯ Н\_ЛЮДИ.ИД= Н\_СЕССИЯ.ЧЛВК\_ИД

Изображение выглядит как текст, рукописный текст, бумага, чернила

Автоматически созданное описание

### Оптимальный план:

План 2 будет актуальнее, т.к. данные фильтруются до соединения таблиц, что уменьшает количество строк для обработки, также т.к. количество данных уже сокращено и не имеет смысла использовать Hash JOIN, т.к. он не актуален для небольшого количества строк

### Изменения с добавлением индексов

1. Вместо полного скана таблиц будет использован индексный скан
2. Nested Loop JOIN будет быстрее, благодаря индексам на атрибутах Н\_СЕССИЯ, ЧЛВК\_ИД

### Объяснение

Nested Loop (cost=5.01..254.28 rows=64 width=28) (actual time=0.059..0.061 rows=0 loops=1)

-> Nested Loop (cost=4.71..57.85 rows=1 width=28) (actual time=0.059..0.060 rows=0 loops=1)

-> Index Scan using "ЧЛВК\_PK" on "Н\_ЛЮДИ" (cost=0.28..8.30 rows=1 width=24) (actual time=0.015..0.015 rows=1 loops=1)

Index Cond: (("ИД" > 100865) AND ("ИД" = 105590))

-> Bitmap Heap Scan on "Н\_СЕССИЯ" (cost=4.43..49.54 rows=1 width=4) (actual time=0.039..0.040 rows=0 loops=1)

Recheck Cond: ("ЧЛВК\_ИД" = 105590)

Filter: ("ДАТА" = '2004-01-17 00:00:00'::timestamp without time zone)

Rows Removed by Filter: 20

Heap Blocks: exact=9

-> Bitmap Index Scan on "SYS\_C003500\_IFK" (cost=0.00..4.43 rows=20 width=0) (actual time=0.012..0.012 rows=20 loops=1)

Index Cond: ("ЧЛВК\_ИД" = 105590)

-> Index Scan using "ВЕД\_ЧЛВК\_FK\_IFK" on "Н\_ВЕДОМОСТИ" (cost=0.29..195.79 rows=64 width=8) (never executed)

Index Cond: ("ЧЛВК\_ИД" = 105590)

Planning Time: 0.320 ms

Execution Time: 0.134 ms

Вывод:

В ходе выполнения работы я научился строить планы выполнения запросов и учитывать работу индексов