Министерство образования и науки РФ

Федеральное государственное автономное

образовательное учреждение высшего образования

«Санкт-Петербургский национальный исследовательский университет

информационных технологий, механики и оптики»

**факультет программной инженерии и компьютерной техники**

**ЛАБОРАТОРНАЯ РАБОТА № 4**

по дисциплине

‘Информационные системы и базы данных’

Вариант №1966

*Выполнил:*

Студент группы P33111

Павлов Александр Сергеевич

*Преподаватель:*

Харитонова А.Е.

Изображение выглядит как Шрифт, логотип, Графика, белый

Автоматически созданное описание

Санкт-Петербург, 2023

# Описание 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.ИД > 1.  
   b) 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\_ВЕДОМОСТИ, Н\_СЕССИЯ.  
   Вывести атрибуты: Н\_ЛЮДИ.ИД, Н\_ВЕДОМОСТИ.ИД, Н\_СЕССИЯ.ИД.  
   Фильтры (AND):  
   a) Н\_ЛЮДИ.ИД < 100012.  
   b) Н\_ВЕДОМОСТИ.ЧЛВК\_ИД < 142390.  
   c) Н\_СЕССИЯ.УЧГОД > 2003/2004.  
   Вид соединения: RIGHT JOIN.

# Запросы

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

FROM Н\_ТИПЫ\_ВЕДОМОСТЕЙ

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

WHERE Н\_ТИПЫ\_ВЕДОМОСТЕЙ.ИД > 1

AND Н\_ВЕДОМОСТИ.ЧЛВК\_ИД > 105590;

Explain Analyze:

Изображение выглядит как текст, снимок экрана, число, Шрифт

Автоматически созданное описание

Изображение выглядит как текст, снимок экрана, линия, диаграмма

Автоматически созданное описание

Изображение выглядит как текст, снимок экрана, Шрифт, линия

Автоматически созданное описание

Оптимальным запросом является запрос №3, так как в ином другом варианте объединение таблиц будет производиться либо со всем, либо с большинством объемом данных двух таблиц, что заметно снизит эффективность запроса.

Индексы, добавление которых уменьшит время выполнения запроса:

1. CREATE INDEX Н\_ТИПЫ\_ВЕДОМОСТЕЙ\_ИД\_ИНДЕКС ON Н\_ТИПЫ\_ВЕДОМОСТЕЙ (“ИД”);
2. CREATE INDEX Н\_ВЕДОМОСТИ\_ЧЛВК\_ИД\_ИНДЕКС ON Н\_ВЕДОМОСТИ (“ЧЛВК\_ИД”);
3. CREATE INDEX Н\_ВЕДОМОСТИ\_ТВ\_ИД\_ИНДЕКС ON Н\_ВЕДОМОСТИ USING HASH (“ТВ\_ИД”);

Первые два индекса должны ускорить выполнение запроса, так как у нас происходит выборка бОльших или мЕньших значений. Третий индекс должен ускорить объединение таблиц, так как поможет эффективнее выбирать строки, основываясь на хэше.

2. SELECT Н\_ЛЮДИ.ИД, Н\_ВЕДОМОСТИ.ИД, Н\_СЕССИЯ.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 > 100012

AND Н\_ВЕДОМОСТИ.ЧЛВК\_ИД < 142390

AND Н\_СЕССИЯ.УЧГОД > '2003/2004';

Explain Analyze:

Изображение выглядит как текст, снимок экрана, число, Шрифт

Автоматически созданное описание

Изображение выглядит как текст, диаграмма, линия, Параллельный

Автоматически созданное описание

Изображение выглядит как текст, диаграмма, линия, снимок экрана

Автоматически созданное описание

Оптимальным запросом является запрос №4, так как в ином другом варианте объединение таблиц будет производиться либо со всем, либо с большинством объемом данных таблиц, что заметно снизит эффективность запроса.

Индексы, добавление которых уменьшит время выполнения запроса:

1. CREATE INDEX Н\_ЛЮДИ \_ИД\_ИНДЕКС ON Н\_ЛЮДИ (“ИД”);
2. CREATE INDEX Н\_ВЕДОМОСТИ\_ЧЛВК\_ИД\_ИНДЕКС ON Н\_ВЕДОМОСТИ (“ЧЛВК\_ИД”);
3. CREATE INDEX Н\_СЕССИЯ\_УЧГОД\_ИНДЕКС ON Н\_СЕССИЯ (“УЧГОД”);

Эти индексы должны ускорить выполнение запроса, так как у нас происходит выборка бОльших или мЕньших значений.

# Вывод

При выполнении лабораторной работы я познакомился с использованием индексов для ускорения обработки запросов в SQL. Изучил, как поэтапно происходит выполнение запроса, какие планы могут использоваться и как индексы влияют на процесс.