**Министерство образования и науки**

**Российской Федерации**

**САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО**

Факультет программной инженерии и компьютерной техники

Дисциплина: Базы данных

**Лабораторная работа №4**

Вариант 281920

Выполнил студент группы Р3133 Анисимов Максим Дмитриевич

Проверил Барсуков Илья Александрович

Санкт-Петербург

2022 г

Оглавление

[Описание задания 3](#_Toc136416283)

[Реализация первого запроса 3](#_Toc136416284)

[Планы выполнения: 3](#_Toc136416285)

[Создание индексов 4](#_Toc136416286)

[Explain analyze: 5](#_Toc136416287)

[Реализация второго запроса 5](#_Toc136416288)

[План выполнения: 6](#_Toc136416289)

[Индексы 7](#_Toc136416290)

[Explain analyze: 7](#_Toc136416291)

[Выводы 7](#_Toc136416292)

# 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]

# Реализация первого запроса

Сделать запрос для получения атрибутов из указанных таблиц, применив фильтры по указанным условиям:  
Таблицы: Н\_ЛЮДИ, Н\_ВЕДОМОСТИ.  
Вывести атрибуты: Н\_ЛЮДИ.ФАМИЛИЯ, Н\_ВЕДОМОСТИ.ДАТА.  
Фильтры (AND):  
a) Н\_ЛЮДИ.ФАМИЛИЯ > Афанасьев.  
b) Н\_ВЕДОМОСТИ.ЧЛВК\_ИД < 105590.  
Вид соединения: INNER JOIN.

SELECT Н\_ЛЮДИ.ФАМИЛИЯ, Н\_ВЕДОМОСТИ.ДАТА

FROM Н\_ЛЮДИ

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

WHERE (Н\_ЛЮДИ.ФАМИЛИЯ > 'Афанасьев' AND Н\_ВЕДОМОСТИ.ЧЛВК\_ИД < 105590);

# Планы выполнения:

Первый план:

Изображение выглядит как снимок экрана, текст, Шрифт, черный

Автоматически созданное описание

Второй план:

Изображение выглядит как снимок экрана, черный, текст

Автоматически созданное описание

Во втором плане происходит объединение только нужной выборки, а не всех сущностей. Размер промежуточных данных меньше, значит этот план является оптимальным.

# Создание индексов

CREATE INDEX ON "Н\_ЛЮДИ" USING HASH ("ИД");

CREATE INDEX ON "Н\_ЛЮДИ" USING BTREE ("ФАМИЛИЯ");

CREATE INDEX ON "Н\_ВЕДОМОСТИ" USING BTREE ("ЧЛВК\_ИД");

Выборка происходит с использованием операторов сравнения, поэтому оптимально использование BTREE. Соединение таблиц с использованием ИД – оптимально использование HASH

EXPLAIN ANALYZE SELECT Н\_ЛЮДИ.ФАМИЛИЯ, Н\_ВЕДОМОСТИ.ДАТА

FROM Н\_ЛЮДИ

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

WHERE (Н\_ЛЮДИ.ФАМИЛИЯ > 'Афанасьев' AND Н\_ВЕДОМОСТИ.ЧЛВК\_ИД < 105590);

# Explain analyze:

QUERY PLAN

----------------------------------------------------------------------------------------------------------------------------------------

Nested Loop (cost=0.58..15.57 rows=1 width=24) (actual time=0.005..0.006 rows=0 loops=1)

-> Index Scan using "ВЕД\_ЧЛВК\_FK\_IFK" on "Н\_ВЕДОМОСТИ" (cost=0.29..7.27 rows=1 width=12) (actual time=0.004..0.004 rows=0 loops=1)

Index Cond: ("ЧЛВК\_ИД" < 105590)

-> Index Scan using "ЧЛВК\_PK" on "Н\_ЛЮДИ" (cost=0.28..8.30 rows=1 width=20) (never executed)

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

Filter: (("ФАМИЛИЯ")::text > 'Афанасьев'::text)

Planning Time: 0.477 ms

Execution Time: 0.056 ms

# Реализация второго запроса

1. Сделать запрос для получения атрибутов из указанных таблиц, применив фильтры по указанным условиям:  
   Таблицы: Н\_ЛЮДИ, Н\_ОБУЧЕНИЯ, Н\_УЧЕНИКИ.  
   Вывести атрибуты: Н\_ЛЮДИ.ФАМИЛИЯ, Н\_ОБУЧЕНИЯ.ЧЛВК\_ИД, Н\_УЧЕНИКИ.ИД.  
   Фильтры: (AND)  
   a) Н\_ЛЮДИ.ИД > 152862.  
   b) Н\_ОБУЧЕНИЯ.НЗК < 001000.  
   c) Н\_УЧЕНИКИ.НАЧАЛО < 2011-11-21.  
   Вид соединения: INNER JOIN.

SELECT Н\_ЛЮДИ.ФАМИЛИЯ, Н\_ОБУЧЕНИЯ.ЧЛВК\_ИД, Н\_УЧЕНИКИ.ИД

FROM Н\_ЛЮДИ

INNER JOIN Н\_ОБУЧЕНИЯ ON Н\_ЛЮДИ.ИД = Н\_ОБУЧЕНИЯ.ЧЛВК\_ИД

INNER JOIN Н\_УЧЕНИКИ ON Н\_ЛЮДИ.ИД = Н\_УЧЕНИКИ.ЧЛВК\_ИД

WHERE (Н\_ЛЮДИ.ИД > 152862 AND Н\_ОБУЧЕНИЯ.НЗК < '001000' AND Н\_УЧЕНИКИ.НАЧАЛО < '2011-11-21');

# План выполнения:

Первый план:

Изображение выглядит как черный, темнота

Автоматически созданное описание

Второй план:

Изображение выглядит как черный, темнота

Автоматически созданное описание

Во втором плане происходит объединение только нужной выборки, а не всех сущностей. Размер промежуточных данных меньше, значит этот план является оптимальным.

# Индексы

CREATE INDEX ON "Н\_ЛЮДИ" USING HASH ("ИД");  
CREATE INDEX ON "Н\_ОБУЧЕНИЯ" USING BTREE ("НЗК");  
CREATE INDEX ON "Н\_УЧЕНИКИ" USING BTREE ("НАЧАЛО");

Выборка происходит с использованием операторов сравнения, поэтому оптимально использование BTREE.

# Explain analyze:

QUERY PLAN

--------------------------------------------------------------------------------------------------------------------

Nested Loop (cost=0.57..131.34 rows=5 width=24) (actual time=1.282..1.283 rows=0 loops=1)

-> Nested Loop (cost=0.28..128.08 rows=1 width=24) (actual time=1.282..1.283 rows=0 loops=1)

-> Seq Scan on "Н\_ОБУЧЕНИЯ" (cost=0.00..119.76 rows=1 width=4) (actual time=1.282..1.282 rows=0 loops=1)

Filter: (("НЗК")::text < '001000'::text)

Rows Removed by Filter: 5021

-> Index Scan using "ЧЛВК\_PK" on "Н\_ЛЮДИ" (cost=0.28..8.30 rows=1 width=20) (never executed)

Index Cond: (("ИД" = "Н\_ОБУЧЕНИЯ"."ЧЛВК\_ИД") AND ("ИД" > 152862))

-> Index Scan using "УЧЕН\_ОБУЧ\_FK\_I" on "Н\_УЧЕНИКИ" (cost=0.29..3.20 rows=5 width=8) (never executed)

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

Filter: ("НАЧАЛО" < '2011-11-21 00:00:00'::timestamp without time zone)

Planning Time: 0.732 ms

Execution Time: 1.330 ms

# Выводы

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.