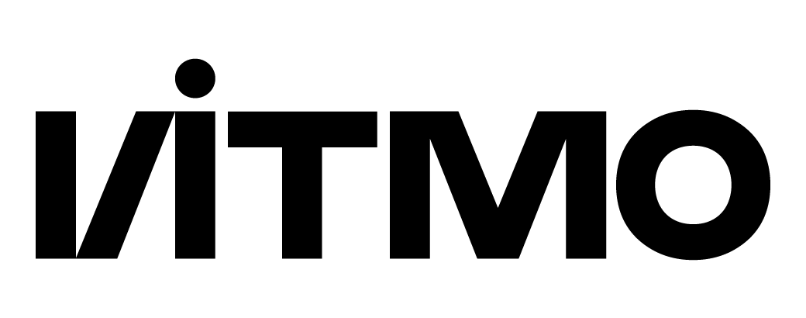
Федеральное государственное автономное образовательное учреждение высшего образования

«Национальный исследовательский университет ИТМО»

Факультет Программной Инженерии и Компьютерной Техники

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

****

Лабораторная работа №4

по дисциплине “Информационные системы и базы данных”

Выполнил:

Студент группы P3110  
Конкин Вадим Вадимович

Преподаватель:

Николаев Владимир Вячеславович

г. Санкт-Петербург

2023 г.

Оглавление

[Текст задания 3](#_Toc135922984)

[Запрос 1 3](#_Toc135922985)

[Запрос 2 4](#_Toc135922986)

[Вывод 6](#_Toc135922987)

# 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:  
   Таблицы: Н\_ОЦЕНКИ, Н\_ВЕДОМОСТИ.  
   Вывести атрибуты: Н\_ОЦЕНКИ.КОД, Н\_ВЕДОМОСТИ.ИД.  
   Фильтры (AND):  
   a) Н\_ОЦЕНКИ.КОД > 4.  
   b) Н\_ВЕДОМОСТИ.ЧЛВК\_ИД = 163249.  
   Вид соединения: RIGHT 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.ИМЯ, Н\_ВЕДОМОСТИ.ДАТА, Н\_СЕССИЯ.УЧГОД.  
   Фильтры (AND):  
   a) Н\_ЛЮДИ.ОТЧЕСТВО > Георгиевич.  
   b) Н\_ВЕДОМОСТИ.ИД < 39921.  
   c) Н\_СЕССИЯ.ДАТА > 2002-01-04.  
   Вид соединения: INNER JOIN.

# Запрос 1

1. Сделать запрос для получения атрибутов из указанных таблиц, применив фильтры по указанным условиям:  
   Таблицы: Н\_ОЦЕНКИ, Н\_ВЕДОМОСТИ.  
   Вывести атрибуты: Н\_ОЦЕНКИ.КОД, Н\_ВЕДОМОСТИ.ИД.  
   Фильтры (AND):  
   a) Н\_ОЦЕНКИ.КОД > 4.  
   b) Н\_ВЕДОМОСТИ.ЧЛВК\_ИД = 163249.  
   Вид соединения: RIGHT JOIN.

SELECT "Н\_ОЦЕНКИ"."КОД", "Н\_ВЕДОМОСТИ"."ИД" FROM "Н\_ВЕДОМОСТИ" RIGHT JOIN "Н\_ОЦЕНКИ" ON "Н\_ВЕДОМОСТИ"."ОЦЕНКА" = "Н\_ОЦЕНКИ"."КОД"  
WHERE "Н\_ОЦЕНКИ"."КОД" > 4 AND "Н\_ВЕДОМОСТИ"."ЧЛВК\_ИД" = 163249;

В этом запросе используются операторы сравнения. Для таких операторов лучше всего подходит индекс B-Tree. Поэтому для атрибутов ) Н\_ОЦЕНКИ.КОД и Н\_ВЕДОМОСТИ.ЧЛВК\_ИД создадим индекс. В PostgreSQL по умолчанию выбирается индекс B-Tree.

CREATE INDEX ИНД\_ВЕД\_ЧЛВК\_ИД on Н\_ВЕДОМОСТИ(ЧЛВК\_ИД)

Но стоит упомянуть, что для всех UNIQUE атрибутов PostgreSQL автоматически создает индекс, поэтому создавать индекс на Н\_ОЦЕНКИ.КОД будет излишним.

Планы выполнения запроса: Изображение выглядит как текст, диаграмма, снимок экрана, линия

Автоматически созданное описание

Изображение выглядит как диаграмма, линия, текст, График

Автоматически созданное описание

Второй план оптимальный

При добавлении индексов:

Вместо полного скана таблиц будет использоваться индексный скан.

Nested Loops Join станет быстрее благодаря индексу на атрибуте Н\_ВЕДОМОСТИ.ЧЛВК\_ИД

EXPLAIN ANALYZE

Nested Loop (cost=0.29..197.17 rows=11 width=38) (actual time=0.034..0.044 rows=10 loops=1)

-> Seq Scan on ""Н\_ОЦЕНКИ"" (cost=0.00..1.11 rows=1 width=34) (actual time=0.016..0.016 rows=1 loops=1)

Filter: ((""КОД"")::text = '5'::text)

Rows Removed by Filter: 8

-> Index Scan using ""ВЕД\_ЧЛВК\_FK\_IFK"" on ""Н\_ВЕДОМОСТИ"" (cost=0.29..195.95 rows=11 width=10) (actual time=0.016..0.024 rows=10 loops=1)

Index Cond: (""ЧЛВК\_ИД"" = 163249)

Filter: ((""ОЦЕНКА"")::text = '5'::text)

Rows Removed by Filter: 9

Planning Time: 0.169 ms

Execution Time: 0.073 ms

# 

# Запрос 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.ИМЯ, Н\_ВЕДОМОСТИ.ДАТА, Н\_СЕССИЯ.УЧГОД.  
Фильтры (AND):  
a) Н\_ЛЮДИ.ОТЧЕСТВО > Георгиевич.  
b) Н\_ВЕДОМОСТИ.ИД < 39921.  
c) Н\_СЕССИЯ.ДАТА > 2002-01-04.  
Вид соединения: INNER JOIN.

SELECT "Н\_ЛЮДИ"."ИМЯ", "Н\_ВЕДОМОСТИ"."ДАТА", "Н\_СЕССИЯ"."УЧГОД" FROM "Н\_ВЕДОМОСТИ"  
 JOIN "Н\_ЛЮДИ" ON "Н\_ВЕДОМОСТИ"."ЧЛВК\_ИД" = "Н\_ЛЮДИ"."ИД"  
 JOIN "Н\_СЕССИЯ" ON "Н\_ЛЮДИ"."ИД" = "Н\_СЕССИЯ"."ЧЛВК\_ИД"  
WHERE "Н\_ЛЮДИ"."ОТЧЕСТВО" > 'Георгиевич' AND "Н\_ВЕДОМОСТИ"."ИД" < 39921 AND "Н\_СЕССИЯ"."ДАТА" > '2002-01-04'::timestamp;

В этом запросе используются операторы сравнения. Для таких операторов лучше всего подходит индекс B-Tree. Поэтому для атрибутов Н\_ЛЮДИ.ОТЧЕСТВО, Н\_ВЕДОМОСТИ.ИД и Н\_СЕССИЯ.ДАТА создадим индекс. В PostgreSQL по умолчанию выбирается индекс B-Tree.

CREATE INDEX ИНД\_СЕССИЯ\_ДАТА on Н\_СЕССИЯ(ДАТА)

CREATE INDEX ИНД\_ЛЮДИ\_ОТЧЕСТВО on Н\_ЛЮДИ(ОТЧЕСТВО)

Но стоит упомянуть, что для всех UNIQUE атрибутов PostgreSQL автоматически создает индекс, поэтому создавать индекс на Н\_ВЕДОМОСТИ.ИД будет излишним.

Планы выполнения запроса:

Изображение выглядит как текст, диаграмма, снимок экрана, линия

Автоматически созданное описание

Изображение выглядит как текст, диаграмма, линия, График

Автоматически созданное описание

Изображение выглядит как текст, диаграмма, линия, График

Автоматически созданное описание

EXPLAIN ANALYZE

Nested Loop (cost=0.98..16.54 rows=1 width=31) (actual time=0.004..0.005 rows=0 loops=1)

" Join Filter: (""Н\_ВЕДОМОСТИ"".""ЧЛВК\_ИД"" = ""Н\_СЕССИЯ"".""ЧЛВК\_ИД"")"

-> Nested Loop (cost=0.70..15.56 rows=1 width=29) (actual time=0.004..0.004 rows=0 loops=1)

" -> Index Scan using ""ВЕД\_PK"" on ""Н\_ВЕДОМОСТИ"" (cost=0.42..7.25 rows=1 width=12) (actual time=0.004..0.004 rows=0 loops=1)"

" Index Cond: (""ИД"" < 39921)"

" -> Index Scan using ""ЧЛВК\_PK"" on ""Н\_ЛЮДИ"" (cost=0.28..8.30 rows=1 width=17) (never executed)"

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

" Filter: ((""ОТЧЕСТВО"")::text > 'Георгиевич'::text)"

" -> Index Scan using ""SYS\_C003500\_IFK"" on ""Н\_СЕССИЯ"" (cost=0.28..0.78 rows=17 width=14) (never executed)"

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

" Filter: (""ДАТА"" > '2002-01-04 00:00:00'::timestamp without time zone)"

Planning Time: 1.406 ms

Execution Time: 0.041 ms

# Вывод

Дополнил свои знания про индексы. Теперь я сумею определить, какой тип нужно использовать в своей задаче. Поработал с командами EXPLAIN и EXPLAIN ANALYZE. Буду применять их в работе.