Федеральное государственное автономное образовательное учреждение

высшего образования

«Национальный исследовательский университет ИТМО»

Факультет программной инженерии и компьютерной техники

Лабораторная работа по Базам Данных №4

Вариант 31049

Выполнил  
Пчелкин Илья Игоревич

P3106

Проверил

Вербовой А. А.

Санкт-Петербург 2025

Оглавление

[Задание 3](#_Toc196232686)

[1 запрос 4](#_Toc196232687)

[2 запрос 4](#_Toc196232688)

[Возможные планы выполнения запроса №1: 5](#_Toc196232689)

[Возможные планы выполнения запроса №2: 6](#_Toc196232690)

[EXPLAIN ANALYZE 1 запроса: 7](#_Toc196232691)

[EXPLAIN ANALYZE 2 запроса: 7](#_Toc196232692)

# 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:  
   Таблицы: Н\_ОЦЕНКИ, Н\_ВЕДОМОСТИ.  
   Вывести атрибуты: Н\_ОЦЕНКИ.ПРИМЕЧАНИЕ, Н\_ВЕДОМОСТИ.ИД.  
   Фильтры (AND):  
   a) Н\_ОЦЕНКИ.КОД > 5.  
   b) Н\_ВЕДОМОСТИ.ДАТА = 2010-06-18.  
   Вид соединения: RIGHT JOIN.
2. 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.НАЧАЛО.  
   Фильтры: (AND)  
   a) Н\_ЛЮДИ.ИД < 100012.  
   b) Н\_ОБУЧЕНИЯ.НЗК > 001000.  
   c) Н\_УЧЕНИКИ.ГРУППА = 4103.  
   Вид соединения: RIGHT JOIN.

# 1 запрос

-- 1. Сделать запрос для получения атрибутов из указанных таблиц, применив фильтры по указанным условиям:  
-- Таблицы: Н\_ОЦЕНКИ, Н\_ВЕДОМОСТИ.  
-- Вывести атрибуты: Н\_ОЦЕНКИ.ПРИМЕЧАНИЕ, Н\_ВЕДОМОСТИ.ИД.  
-- Фильтры (AND):  
-- a) Н\_ОЦЕНКИ.КОД > 5.  
-- b) Н\_ВЕДОМОСТИ.ДАТА = 2010-06-18.  
-- Вид соединения: RIGHT JOIN.  
  
select Н\_ОЦЕНКИ.ПРИМЕЧАНИЕ, Н\_ВЕДОМОСТИ.ИД  
from Н\_ВЕДОМОСТИ  
 right join Н\_ОЦЕНКИ on Н\_ВЕДОМОСТИ.ОЦЕНКА = Н\_ОЦЕНКИ.КОД  
where Н\_ОЦЕНКИ.КОД like '^\d+$'  
 and Н\_ОЦЕНКИ.КОД::integer < 5  
 and Н\_ВЕДОМОСТИ.ДАТА::date = '2010-06-18';

Создание индексов:

-- т.к. людей с одной оценкой много, создадим индекс для оценки чтобы поиск выполнялся только по строкам с нужной оценкой а также для более быстрого соединения таблиц (B-tree)  
create index grade\_index on Н\_ОЦЕНКИ (КОД);  
  
-- т.к. дат очень много, нужен индекс для того чтобы не перебирать все даты с 1998 г. (B-tree)  
 create index date\_index on Н\_ВЕДОМОСТИ (ДАТА);

# 2 запрос

-- 2. 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.НАЧАЛО.  
-- Фильтры: (AND)  
-- a) Н\_ЛЮДИ.ИД < 100012.  
-- b) Н\_ОБУЧЕНИЯ.НЗК > 001000.  
-- c) Н\_УЧЕНИКИ.ГРУППА = 4103.  
-- Вид соединения: RIGHT JOIN.  
  
  
select Н\_ЛЮДИ.ФАМИЛИЯ, Н\_ОБУЧЕНИЯ.ЧЛВК\_ИД, Н\_УЧЕНИКИ.НАЧАЛО  
from Н\_УЧЕНИКИ  
 right join Н\_ОБУЧЕНИЯ on Н\_ОБУЧЕНИЯ.ЧЛВК\_ИД = Н\_УЧЕНИКИ.ЧЛВК\_ИД  
 right join Н\_ЛЮДИ on Н\_ЛЮДИ.ИД = Н\_ОБУЧЕНИЯ.ЧЛВК\_ИД  
where Н\_ЛЮДИ.ИД < 100012  
 and Н\_ОБУЧЕНИЯ.НЗК::integer > 001000  
 and Н\_УЧЕНИКИ.ГРУППА = '4103';

Создание индексов:

-- для более быстрого соединения (B-tree)  
create index person\_id\_index\_1 on Н\_УЧЕНИКИ (ЧЛВК\_ИД);  
  
create index person\_id\_index\_2 on Н\_ОБУЧЕНИЯ (ЧЛВК\_ИД);  
  
create index person\_id\_index\_3 on Н\_ЛЮДИ (ИД);  
  
-- для поиска соответствия сразу в нужной нам группе (B-tree)  
create index group\_index on Н\_УЧЕНИКИ (ГРУППА);  
-- для быстрого перехода к группе кортежей которые удовлетворяют условию (B-tree)  
create index nzk\_index on Н\_ОБУЧЕНИЯ (НЗК);

# Возможные планы выполнения запроса №1:

1)

Изображение выглядит как текст, снимок экрана, Шрифт, линия

Автоматически созданное описание

2)

Изображение выглядит как текст, снимок экрана, линия, диаграмма

Автоматически созданное описание

План выполнения 2 будет эффективнее, т. к. выборка кортежей произойдет до соединения таблиц => соединяются только кортежи подходящие под условие

# Возможные планы выполнения запроса №2:

1)

Изображение выглядит как текст, диаграмма, линия, График

Автоматически созданное описание

2)

Изображение выглядит как текст, диаграмма, линия, График

Автоматически созданное описание

План выполнения 2 будет эффективнее, т. к. выборка кортежей произойдет до соединения таблиц => соединяются только кортежи подходящие под условие

# EXPLAIN ANALYZE 1 запроса:

QUERY PLAN   
-----------------------------------------------------------------------------  
 Nested Loop (cost=273.82..4812.77 rows=124 width=422) (actual time=2.222..32.640 rows=94 loops=1)  
 -> Seq Scan on "Н\_ОЦЕНКИ" (cost=0.00..1.18 rows=1 width=452) (actual time=0.040..0.063 rows=3 loops=1)  
 Filter: ((("КОД")::text ~ '^\d+$'::text) AND (("КОД")::integer < 5))  
 Rows Removed by Filter: 6  
 -> Bitmap Heap Scan on "Н\_ВЕДОМОСТИ" (cost=273.82..4810.35 rows=124 width=10) (actual time=1.417..10.840 rows=31 loops=3)  
 Recheck Cond: (("ОЦЕНКА")::text = ("Н\_ОЦЕНКИ"."КОД")::text)  
 Filter: (("ДАТА")::date = '2010-06-18'::date)  
 Rows Removed by Filter: 25127  
 Heap Blocks: exact=8770  
 -> Bitmap Index Scan on "ВЕД\_ОЦЕНКА\_I" (cost=0.00..273.79 rows=24716 width=0) (actual time=1.038..1.038 rows=25158 loops=3)  
 Index Cond: (("ОЦЕНКА")::text = ("Н\_ОЦЕНКИ"."КОД")::text)  
 Planning Time: 0.312 ms  
 Execution Time: 32.699 ms  
(13 строк)

# EXPLAIN ANALYZE 2 запроса:

QUERY PLAN   
-----------------------------------------------------------------------------  
 Nested Loop (cost=0.85..18.48 rows=1 width=28) (actual time=0.005..0.007 rows=0 loops=1)  
 -> Nested Loop (cost=0.56..16.62 rows=1 width=24) (actual time=0.005..0.006 rows=0 loops=1)  
 -> Index Scan using "ЧЛВК\_PK" on "Н\_ЛЮДИ" (cost=0.28..8.30 rows=1 width=20) (actual time=0.004..0.005 rows=0 loops=1)  
 Index Cond: ("ИД" < 100012)  
 -> Index Scan using "ОБУЧ\_ЧЛВК\_FK\_I" on "Н\_ОБУЧЕНИЯ" (cost=0.28..8.31 rows=1 width=4) (never executed)  
 Index Cond: ("ЧЛВК\_ИД" = "Н\_ЛЮДИ"."ИД")  
 Filter: (("НЗК")::integer > 1000)  
 -> Index Scan using "УЧЕН\_ОБУЧ\_FK\_I" on "Н\_УЧЕНИКИ" (cost=0.29..1.85 rows=1 width=12) (never executed)  
 Index Cond: ("ЧЛВК\_ИД" = "Н\_ОБУЧЕНИЯ"."ЧЛВК\_ИД")  
 Filter: (("ГРУППА")::text = '4103'::text)  
 Planning Time: 0.697 ms  
 Execution Time: 0.053 ms  
(12 строк)

При добавлении индексов Seq Scan(полное сканирование таблицы) заменится на Index Scan(сканирование таблицы по индексу) и время выполнения запроса уменьшится