



Факультет программной инженерной и компьютерной техники
Базы Данных

Лабораторная работа №4

Вариант 10083

Выполнил: Алхимовици Арсений 408138

Принял: Бострикова Дарья Константиновна

Р3110

Санкт-Петербург, 2024

Текст Задания

Введите вариант:

Внимание! У разных вариантов разный 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_ЛЮДИ, Н_СЕССИЯ.

Вывести атрибуты: Н_ЛЮДИ.ФАМИЛИЯ, Н_СЕССИЯ.ДАТА.

Фильтры (AND):

a) Н_ЛЮДИ.ИМЯ = Владимир.

b) Н_СЕССИЯ.ЧЛВК_ИД > 106059.

c) Н_СЕССИЯ.ЧЛВК_ИД > 100622.

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.ИД, Н_ОБУЧЕНИЯ.НЗК, Н_УЧЕНИКИ.ИД.

Фильтры (AND):

a) Н_ЛЮДИ.ИД > 152862.

b) Н_ОБУЧЕНИЯ.НЗК < 001000.

Вид соединения: LEFT JOIN.

Реализация SQL-запросов

```
SELECT Н_ЛЮДИ.ФАМИЛИЯ, Н_СЕССИЯ.ДАТА
FROM Н_ЛЮДИ RIGHT JOIN Н_СЕССИЯ ON Н_ЛЮДИ.ИД = Н_СЕССИЯ.ЧЛВК_ИД
WHERE Н_ЛЮДИ.ИМЯ = 'Владимир' AND Н_СЕССИЯ.ЧЛВК_ИД > 106059;
--вывод (надо брать меньше ид)
```

ФАМИЛИЯ	ДАТА
character varying (25)	timestamp without time zone

Для ид больше 100059

	ФАМИЛИЯ	ДАТА
	character varying (25)	timestamp without time zone
1	Воронин	2007-01-11 00:00:00
2	Воронин	2005-01-08 00:00:00
3	Воронин	2009-01-19 00:00:00
4	Корепанов	2011-01-19 00:00:00
5	Корепанов	2007-06-21 00:00:00
6	Корепанов	2005-06-18 00:00:00
7	Воронин	2005-12-29 00:00:00
8	Воронин	2009-01-12 00:00:00
9	Воронин	2009-01-19 00:00:00
10	Корепанов	[null]
11	Корепанов	[null]
12	Корепанов	2010-01-22 00:00:00

```
SELECT Н_ЛЮДИ.ИД, Н_ОБУЧЕНИЯ.НЗК, Н_УЧЕНИКИ.ИД
```

```

FROM Н_ЛЮДИ LEFT JOIN Н_УЧЕНИКИ ON Н_ЛЮДИ.ИД = Н_УЧЕНИКИ.ЧЛВК_ИД
LEFT JOIN Н_ОБУЧЕНИЯ ON Н_ЛЮДИ.ИД = Н_ОБУЧЕНИЯ.ЧЛВК_ИД
WHERE Н_ЛЮДИ.ИД > 152862 AND Н_ОБУЧЕНИЯ.НЗК::int < 001000;

-- ВЫВОД

```

	ИД integer	НЗК character varying (8)	ИД integer
1	160540	11	242144

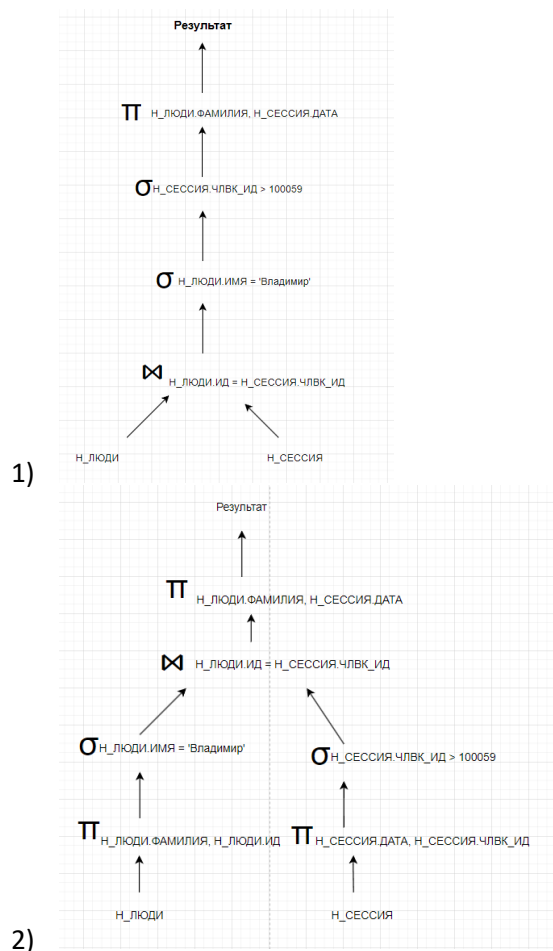
Индексы

Для 1-ого запроса надо сделать индекс(B-tree) на поле Н_ЛЮДИ.ИМЯ тк тогда ускорится время фильтрации таблицы Н_ЛЮДИ по условию, что имя должно быть “Владимир”. Так же можно добавить индекс (B-tree) на поле Н_СЕССИЯ.ЧЛВК_ИД тк это ускорит фильтрацию таблицы Н_СЕССИЯ по условию Н_СЕССИЯ.ЧЛВК_ИД>106059 и так же ускорит время соединения таблиц.

Для 2-ого запроса надо сделать индексы (B-tree) на полях Н_ОБУЧЕНИЯ.НЗК и Н_ОБУЧЕНИЕ.ЧЛВК_ИД для таблицы Н_ОБУЧЕНИЯ тк это ускорит фильтрацию по условию Н_ОБУЧЕНИЯ.НЗК::int > 001000 и ускорит время соединения таблиц соответственно. Так же надо добавить индекс (B-tree) на таблицу Н_УЧЕНИКИ на поле Н_УЧЕНИКИ.ЧЛВК_ИД, чтобы ускорить соединение таблиц.

Планы

Рассмотрим возможные планы выполнения для 1-ого запроса:



Второй план будет более оптимальным тк фильтрация выполняется до соединения таблиц, тем самым меньше строк для обработки => быстрее выполнение.

При добавлении индекса вместо полного скана будет использоваться индексный скан, Nested Loops Join будет работать быстрее тк есть индекс на Н_СЕССИЯ.ЧЛВК_ИД.

Nested Loop (cost=8.46..647.37 rows=9 width=24) (actual time=0.607..0.608 rows=0 loops=1)

-> Bitmap Heap Scan on "Н_СЕССИЯ" (cost=8.16..85.43 rows=501 width=12) (actual time=0.038..0.203 rows=502 loops=1)

Recheck Cond: ("ЧЛВК_ИД" > 106059)

Heap Blocks: exact=68

-> Bitmap Index Scan on "SYS_C003500_IFK" (cost=0.00..8.04 rows=501 width=0) (actual time=0.026..0.027 rows=502 loops=1)

Index Cond: ("ЧЛВК_ИД" > 106059)

-> Memoize (cost=0.29..3.21 rows=1 width=20) (actual time=0.001..0.001 rows=0 loops=502)

Cache Key: ""Н_СЕССИЯ"". ""ЧЛВК_ИД""

Cache Mode: logical

Hits: 427 Misses: 75 Evictions: 0 Overflows: 0 Memory Usage: 5kB

-> Index Scan using ""ЧЛВК_РК"" on ""Н_ЛЮДИ"" (cost=0.28..3.20 rows=1 width=20) (actual time=0.003..0.003 rows=0 loops=75)

Index Cond: (""ИД"" = ""Н_СЕССИЯ"". ""ЧЛВК_ИД"")

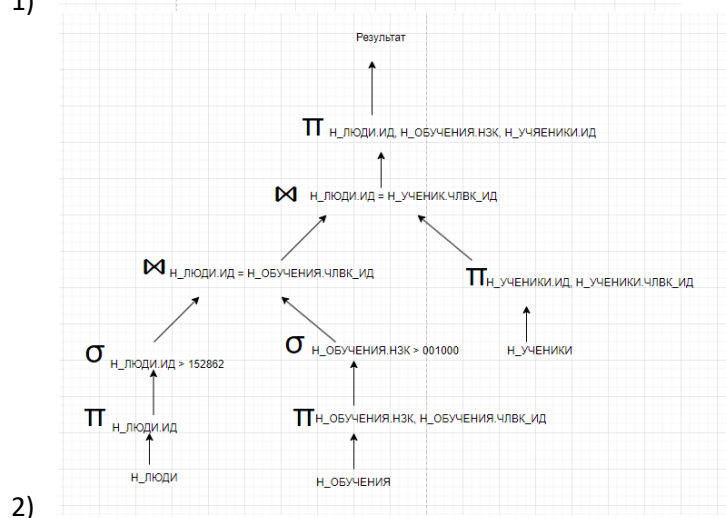
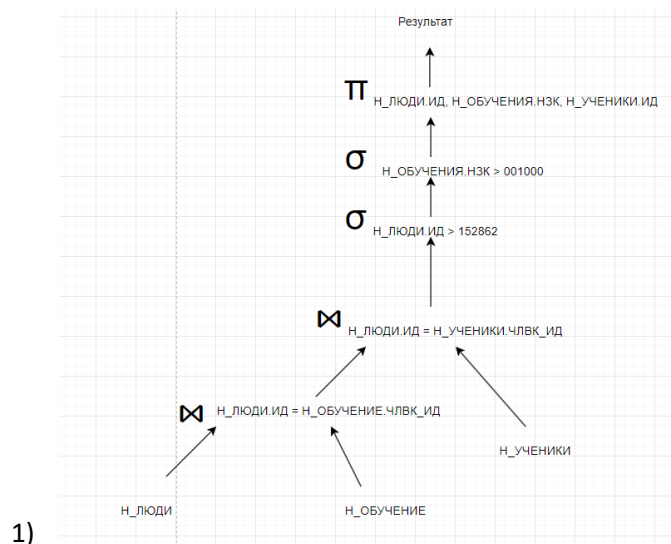
Filter: ((""ИМЯ"")::text = 'Владимир'::text)

Rows Removed by Filter: 1

Planning Time: 0.317 ms

Execution Time: 0.652 ms

Рассмотрим планы для 2-ого запроса:



Второй план будет оптимальнее так же как и в 1 запросе из-за того что мы используем фильтрацию, до того как соединять таблицы => запрос выполнится быстрее.

При добавлении индекса вместо полного скана будет использоваться индексный скан => запрос выполниться еще быстрее так к тому же из-за того что у нас есть индекс на Н_ОБУЧЕНИЯ.ЧЛВК_ИД и Н_УЧЕНИКИ.ЧЛВК_ИД Nested Loops Join будет работать еще быстрее.

Nested Loop Left Join (cost=45.14..1062.04 rows=1220 width=14)
(actual time=0.689..0.915 rows=1 loops=1)

-> Hash Join (cost=44.85..194.11 rows=268 width=10) (actual
time=0.680..0.905 rows=1 loops=1)

Hash Cond: ("Н_ОБУЧЕНИЯ"."ЧЛВК_ИД" = "Н_ЛЮДИ"."ИД")

-> Seq Scan on "Н_ОБУЧЕНИЯ" (cost=0.00..144.87 rows=1674
width=10) (actual time=0.192..0.670 rows=2 loops=1)

Filter: (("НЗК")::integer < 1000)

Rows Removed by Filter: 5019

-> Hash (cost=34.61..34.61 rows=819 width=4) (actual
time=0.225..0.225 rows=827 loops=1)

Buckets: 1024 Batches: 1 Memory Usage: 38kB

-> Index Only Scan using "ЧЛВК_РК" on "Н_ЛЮДИ"
(cost=0.28..34.61 rows=819 width=4) (actual time=0.008..0.106
rows=827 loops=1)

Index Cond: ("ИД" > 152862)

Heap Fetches: 0

-> Index Scan using "УЧЕН_ОБУЧ_FK_I" on "Н_УЧЕНИКИ"
(cost=0.29..3.19 rows=5 width=8) (actual time=0.007..0.007 rows=1
loops=1)

Index Cond: ("ЧЛВК_ИД" = "Н_ЛЮДИ"."ИД")

Planning Time: 0.451 ms

Execution Time: 0.959 ms

Вывод:

В ходе данной лабораторной работы я научился работать с индексами, попрактиковался в составлении запросов и планов. Поработал с учебной базой данных. Научился оптимизировать свои запросы.