



Федеральное государственное автономное образовательное
учреждение высшего образования

«Национальный исследовательский университет ИТМО»

Факультет ПИ и КТ

Лабораторная работа №4
по дисциплине: «Базы данных»
Вариант 311113567

Выполнил:

Болорболд Аригуун,

группа Р3111

Преподаватель:

Николаев Владимир Вячеславович

Практик:

Чупанов Аликылыч Алибекович

IT's *more than a*
UNIVERSITY

Санкт-Петербург

2023

1. 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. Изменяются ли планы при добавлении индекса и как?

Для запросов 1-2 необходимо добавить в отчет вывод команды EXPLAIN ANALYZE [запрос].

Запросы:

1. Сделать запрос для получения атрибутов из указанных таблиц, применив фильтры по указанным условиям:

Таблицы: Н_ЛЮДИ, Н_СЕССИЯ.

Вывести атрибуты: Н_ЛЮДИ.ИМЯ, Н_СЕССИЯ.ДАТА.

Фильтры (AND):

а) Н_ЛЮДИ.ИД = 100865.

б) Н_СЕССИЯ.ДАТА = 2012-01-25.

с) Н_СЕССИЯ.ДАТА < 2002-01-04.

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)

а) Н_ЛЮДИ.ИД > 152862.

б) Н_ОБУЧЕНИЯ.НЗК < 933232.

Вид соединения: LEFT JOIN.

2. Реализация запросов на SQL:

2.1.

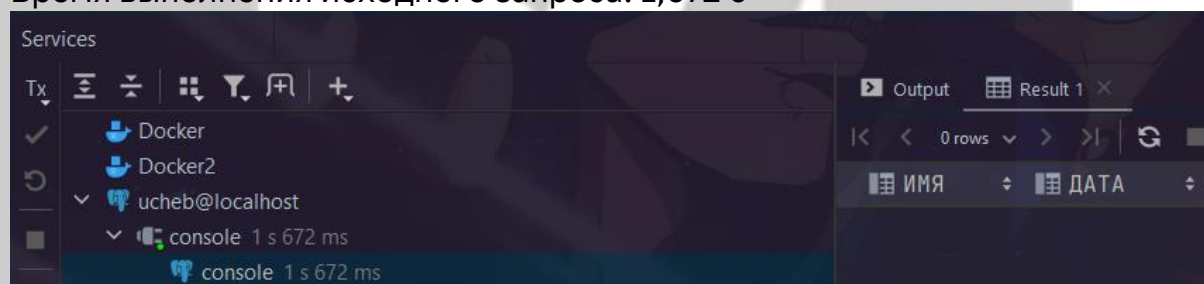
```
SELECT Н_люди.Имя, Н_сессия.Дата FROM Н_люди  
RIGHT JOIN Н_сессия ON Н_люди.Ид = Н_сессия.Члвк_Ид  
WHERE Н_люди.Ид = 100865  
AND Н_сессия.Дата = '2012-01-25'  
AND Н_сессия.Дата < '2002-01-04';
```

2.2.

```
SELECT Н_люди.Фамилия, Н_обучения.Члвк_Ид, Н_ученики.Начало FROM Н_люди  
LEFT JOIN Н_обучения ON Н_люди.Ид = Н_обучения.Члвк_Ид  
LEFT JOIN Н_ученики ON Н_обучения.Члвк_Ид = Н_ученики.Члвк_Ид  
WHERE Н_люди.Ид > 152862  
AND Н_обучения.НЗК < '933232';
```

3. Уменьшение выполнения времени 1 запроса:

Время выполнения исходного запроса: 1,672 с



Индексы, добавления которых уменьшит время выполнения запроса:

a) На таблице Н_люди:

Индекс на атрибуте Н_люди.Ид (B-tree) ускорит фильтрацию с условием «Н_люди.Ид = 100865».

b) На таблице Н_сессия:

Индекс на атрибуте Н_сессия.Члвк_Ид (B-tree) ускорит соединение таблиц по атрибуту Члвк_Ид.
Индекс на атрибуте Н_сессия.Дата (B-tree) ускорит фильтрацию с условиями «Н_сессия.Дата = 2012-01-25» и «Н_сессия.Дата < 2002-01-04».

Добавление индексов на указанные столбцы позволит ускорить поиск нужных записей по фильтрам в запросе.

Возможные планы выполнения запроса без индексов:

План 1:

- Полный скан таблицы Н_люди;

- Полный скан таблицы Н_СЕССИЯ с применением фильтра по условию «Н_СЕССИЯ.ДАТА = 2012-01-25».
- Полный скан таблицы Н_СЕССИЯ с применением фильтра по условиям «Н_СЕССИЯ.ДАТА < 2002-01-04».
- Соединение таблиц с использованием Nested Loops Join по атрибуту ЧЛВК_ИД.
- Фильтрация результата соединения по условию «Н_ЛЮДИ.ИД = 100865».

План 2:

- Полный скан таблицы Н_ЛЮДИ с применением фильтра по условию «Н_ЛЮДИ.ИД = 100865».
- Полный скан таблицы Н_СЕССИЯ с применением фильтра по условию «Н_СЕССИЯ.ДАТА = 2012-01-25».
- Полный скан таблицы Н_СЕССИЯ с применением фильтра по условиям «Н_СЕССИЯ.ДАТА < 2002-01-04».
- Соединение таблиц с использованием Nested Loops Join по атрибуту ЧЛВК_ИД.

Оптимальный план: второй, так как фильтрация данных выполняется до соединения таблиц, что уменьшает количество строк для обработки, а следовательно, и время выполнения запроса.

При добавлении индексов планы выполнения запросов изменятся:

- Вместо полного скана таблиц будет использоваться индексный скан.
- Nested Loops Join станет быстрее благодаря индексу на атрибуте ЧЛВК_ИД.

4. План выполнения 1 запроса:

Nested Loop (cost=4.68..50.32 rows=1 width=21) (actual time=0.125..0.126 rows=0 loops=1)

-> Index Scan using "ЧЛВК_ПК" on "Н_ЛЮДИ" (cost=0.28..8.30 rows=1 width=17) (actual time=0.041..0.042 rows=1 loops=1)

Index Cond: ("ИД" = 100865)

-> Bitmap Heap Scan on "Н_СЕССИЯ" (cost=4.39..42.01 rows=1 width=12) (actual time=0.080..0.081 rows=0 loops=1)

Recheck Cond: ("ЧЛВК_ИД" = 100865)

Filter: (("DATA" < '2002-01-04 00:00:00'::timestamp without time zone) AND ("DATA" = '2012-01-25 00:00:00'::timestamp without time zone))

Rows Removed by Filter: 15

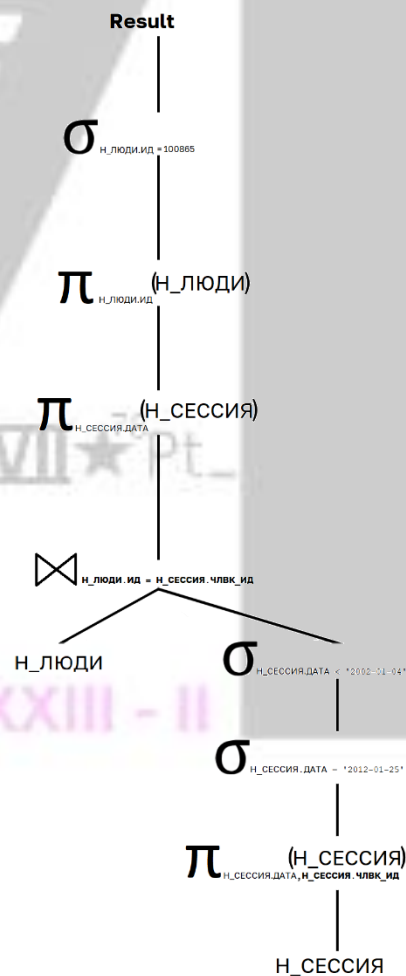
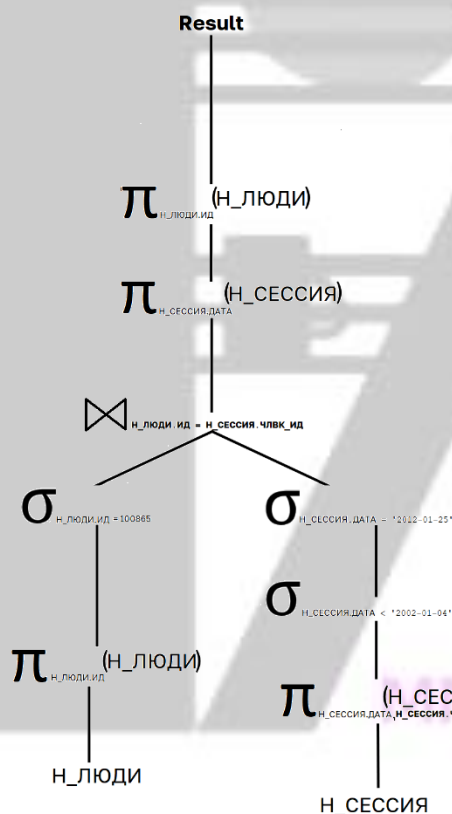
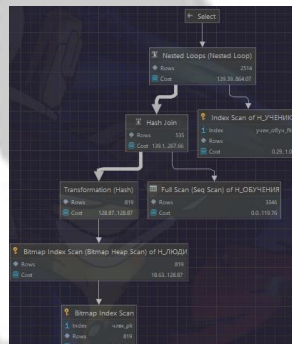
Heap Blocks: exact=14

-> Bitmap Index Scan on "SYS_C003500_IFK" (cost=0.00..4.39 rows=15 width=0) (actual time=0.046..0.047 rows=15 loops=1)

Index Cond: ("ЧЛВК_ИД" = 100865)

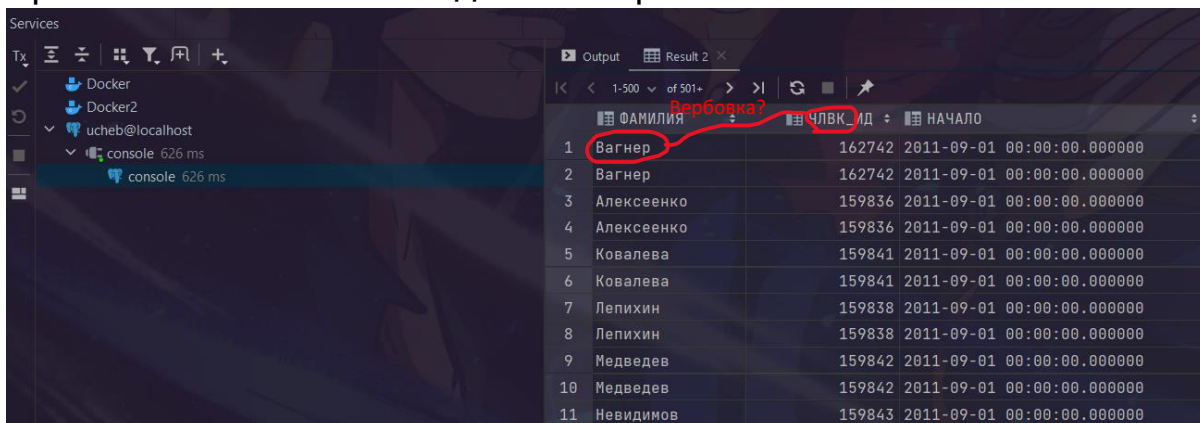
Planning Time: 0.861 ms

Execution Time: 0.207 ms



5. Уменьшение выполнения времени 2 запроса:

Время выполнения исходного запроса: 626 мс



	ФАМИЛИЯ	ЧЛВК_ИД	НАЧАЛО
1	Вагнер	162742	2011-09-01 00:00:00.000000
2	Вагнер	162742	2011-09-01 00:00:00.000000
3	Алексеев	159836	2011-09-01 00:00:00.000000
4	Алексеев	159836	2011-09-01 00:00:00.000000
5	Ковалева	159841	2011-09-01 00:00:00.000000
6	Ковалева	159841	2011-09-01 00:00:00.000000
7	Лепихин	159838	2011-09-01 00:00:00.000000
8	Лепихин	159838	2011-09-01 00:00:00.000000
9	Медведев	159842	2011-09-01 00:00:00.000000
10	Медведев	159842	2011-09-01 00:00:00.000000
11	Невидимов	159843	2011-09-01 00:00:00.000000

Индексы, добавления которых уменьшит время выполнения запроса:

а) На таблице Н_ЛЮДИ:

Индекс на атрибуте Н_ЛЮДИ.ИД (B-tree) ускорит фильтрацию с условием «Н_ЛЮДИ.ИД > 152862».

б) На таблице Н_ОБУЧЕНИЯ:

Индекс на атрибуте Н_ОБУЧЕНИЯ.НЗК (B-tree) ускорит фильтрацию с условием «Н_ОБУЧЕНИЯ.НЗК < 933232».

в) На таблице Н_УЧЕНИКИ:

Возможно добавление индекса на атрибуте Н_УЧЕНИКИ.ЧЛВК_ИД (B-tree), чтобы ускорить соединение таблиц по атрибуту ЧЛВК_ИД, но это вряд ли будет способствовать уменьшению времени выполнения запроса и

Добавление индексов на указанные столбцы позволит ускорить поиск нужных записей по фильтрам в запросе.

Возможные планы выполнения запроса без индексов:

План 1:

- Полный скан таблицы Н_ЛЮДИ с применением фильтра по условию «Н_ЛЮДИ.ИД > 152862».
- Полный скан таблицы Н_ОБУЧЕНИЯ с применением фильтра по условию «Н_ОБУЧЕНИЯ.НЗК < 933232».
- Соединение таблиц с использованием Nested Loops Join по атрибуту ЧЛВК_ИД.
- Полный скан таблицы Н_УЧЕНИКИ.

- Соединение таблиц с использованием Nested Loops Join по атрибуту ЧЛВК_ИД.

План 2:

- Полный скан таблицы Н_ЛЮДИ с применением фильтра по условию «Н_ЛЮДИ.ИД > 152862».
- Полный скан таблицы Н_ОБУЧЕНИЯ с применением фильтра по условию «Н_ОБУЧЕНИЯ.НЗК < 933232».
- Соединение таблиц с использованием Hash Join по атрибуту ЧЛВК_ИД.
- Полный скан таблицы Н_УЧЕНИКИ.
- Соединение таблиц с использованием Hash Join по атрибуту ЧЛВК_ИД.

Оптимальный план: первый, так как использование Nested Loops Join предпочтительнее по сравнению в Hash Join-ом в случаях, когда обрабатываемые наборы данных небольшие, и когда заранее известно, что результаты фильтрации будут содержать мало строк.

При добавлении индексов планы выполнения запросов изменятся:

- Вместо полного скана таблиц будет использоваться индексный скан.
- Nested Loops Join станет быстрее благодаря индексу на атрибутам Н_ОБУЧЕНИЯ.ЧЛВК_ИД и Н_УЧЕНИКИ.ЧЛВК_ИД.

6. План выполнения 2 запроса:

Nested Loop Left Join (cost=139.39..864.07 rows=2514 width=28) (actual time=1.139..5.935 rows=2190 loops=1)

-> Hash Join (cost=139.10..267.66 rows=535 width=20) (actual time=1.115..2.722 rows=803 loops=1)

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

-> Seq Scan on "Н_ОБУЧЕНИЯ" (cost=0.00..119.76 rows=3346 width=4) (actual time=0.013..1.627 rows=3347 loops=1)

Filter: (("НЗК")::text < '933232')::text)

Rows Removed by Filter: 1674

-> Hash (cost=128.87..128.87 rows=819 width=20) (actual time=0.486..0.487 rows=827 loops=1)

Buckets: 1024 Batches: 1 Memory Usage: 52kB

-> Bitmap Heap Scan on "Н_ЛЮДИ" (cost=18.63..128.87 rows=819 width=20) (actual time=0.079..0.324 rows=827 loops=1)

Recheck Cond: ("ИД" > 152862)

-> Bitmap Index Scan on "ЧЛВК_ПК" (cost=0.00..18.42 rows=819 width=0) (actual time=0.052..0.052 rows=827 loops=1)

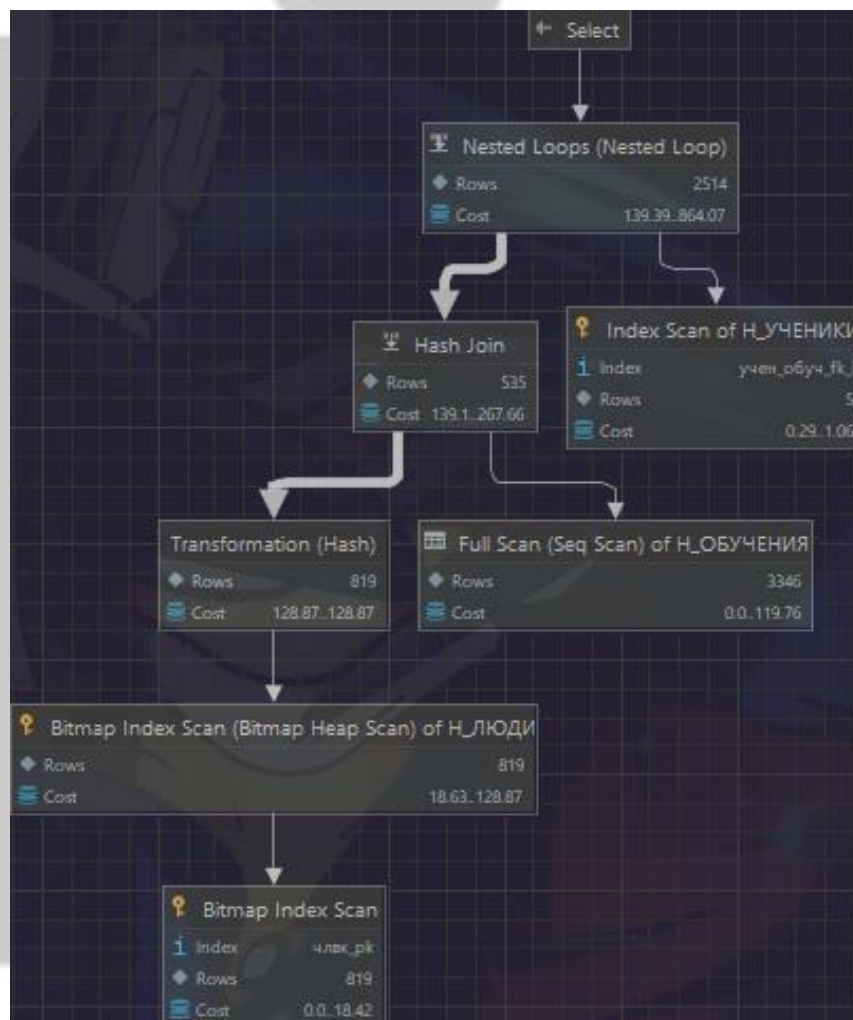
Index Cond: ("ИД" > 152862)

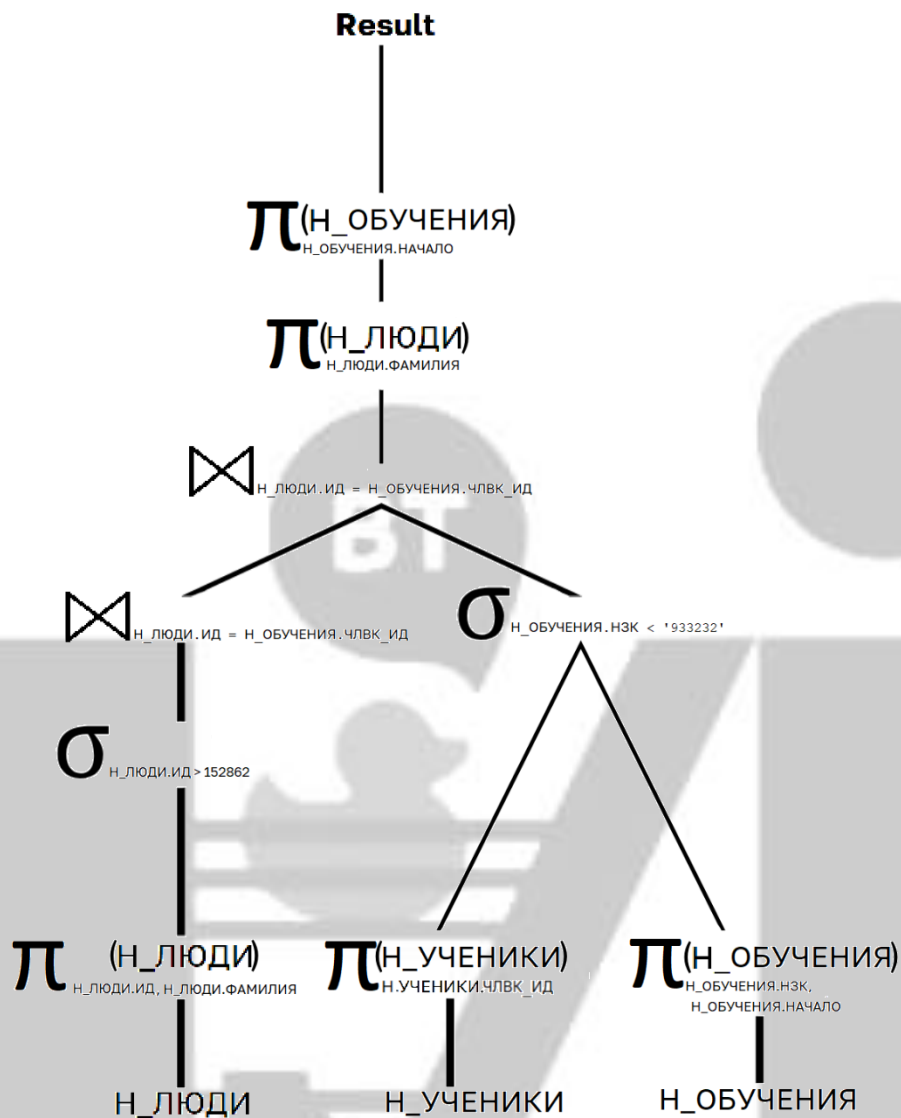
-> Index Scan using "УЧЕН_ОБУЧ_ФК_I" on "Н_УЧЕНИКИ" (cost=0.29..1.06 rows=5 width=12) (actual time=0.002..0.003 rows=3 loops=803)

Index Cond: ("ЧЛВК_ИД" = "Н_ОБУЧЕНИЯ"."ЧЛВК_ИД")

Planning Time: 1.391 ms

Execution Time: 6.140 ms





7. Вывод:

В ходе выполнения лабораторной работы я освоил тему реляционной алгебры, планы выполнения запроса и составления деревьев планирования. Также изучил индексы в PostgreSQL, который является одним из двух методов оптимизации скорости выполнения запроса. В целом, здесь заканчивается курс базы данных, но кроме PostgreSQL-а есть множество других тип базы данных, поэтому обязательно буду заниматься их изучением.