



Лабораторная работа №4

по дисциплине: Информационные системы и базы данных

Вариант: 1350

Выполнил: Неграш Андрей, Р33301

Преподаватель: Харитоновна Анастасия Евгеньевна

Санкт-Петербург, 2022

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.

Изменяются 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:
Н_ТИПЫ_ВЕДОМОСТЕЙ, Н_ВЕДОМОСТИ.
Вывести атрибуты: Н_ТИПЫ_ВЕДОМОСТЕЙ.ИД, Н_ВЕДОМОСТИ.ДАТА.
Фильтры (AND):
 - а) Н_ТИПЫ_ВЕДОМОСТЕЙ.НАИМЕНОВАНИЕ = Ведомость.
 - б) Н_ВЕДОМОСТИ.ЧЛВК_ИД < 105590.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.ГРУППА.
Фильтры: (AND)
 - а) Н_ЛЮДИ.ИМЯ = Ярослав.
 - б) Н_ОБУЧЕНИЯ.НЗК = 999080.
 - с) Н_УЧЕНИКИ.НАЧАЛО < 2011-11-21.Вид соединения: RIGHT JOIN.

Реализация запросов на SQL

1) SELECT

```
        Н_ТИПЫ_ВЕДОМОСТЕЙ.ИД,  
        Н_ВЕДОМОСТИ.ДАТА  
FROM Н_ТИПЫ_ВЕДОМОСТЕЙ  
RIGHT JOIN Н_ВЕДОМОСТИ  
ON Н_ТИПЫ_ВЕДОМОСТЕЙ.ИД = Н_ВЕДОМОСТИ.ВЕД_ИД  
WHERE  
        Н_ТИПЫ_ВЕДОМОСТЕЙ.НАИМЕНОВАНИЕ = 'Ведомость' AND  
        Н_ВЕДОМОСТИ.ЧЛВК_ИД < 105590
```

Результат:

ИД | ДАТА

----+-----

(0 строк)

QUERY PLAN

Nested Loop (cost=0.29..8.32 rows=1 width=12) (actual time=0.029..0.031 rows=0 loops=1)

Join Filter: ("Н_ТИПЫ_ВЕДОМОСТЕЙ"."ИД" = "Н_ВЕДОМОСТИ"."ВЕД_ИД")

-> Seq Scan on "Н_ТИПЫ_ВЕДОМОСТЕЙ" (cost=0.00..1.04 rows=1 width=4) (actual time=0.022..0.024 rows=1 loops=1)

Filter: (("НАИМЕНОВАНИЕ")::text = 'Ведомость'::text)

Rows Removed by Filter: 2

-> Index Scan using "ВЕД_ЧЛВК_FK_IFK" on "Н_ВЕДОМОСТИ" (cost=0.29..7.27 rows=1 width=12) (actual time=0.003..0.003 rows=0 loops=1)

Index Cond: ("ЧЛВК_ИД" < 105590)

Planning Time: 0.281 ms

Execution Time: 0.065 ms

(9 строк)

2) SELECT

```
        Н_ЛЮДИ.ФАМИЛИЯ,  
        Н_ОБУЧЕНИЯ.ЧЛВК_ИД,  
        Н_УЧЕНИКИ.ГРУППА  
FROM Н_ЛЮДИ  
RIGHT JOIN Н_ОБУЧЕНИЯ  
ON Н_ЛЮДИ.ИД = Н_ОБУЧЕНИЯ.ЧЛВК_ИД  
RIGHT JOIN Н_УЧЕНИКИ  
ON Н_ОБУЧЕНИЯ.ЧЛВК_ИД = Н_УЧЕНИКИ.ЧЛВК_ИД  
WHERE  
        Н_ЛЮДИ.ИМЯ = 'Ярослав' AND  
        Н_ОБУЧЕНИЯ.НЗК = '999080' AND  
        Н_УЧЕНИКИ.НАЧАЛО < DATE('2011-11-21')
```

Результат:

ФАМИЛИЯ | ЧЛВК_ИД | ГРУППА

-----+-----+-----

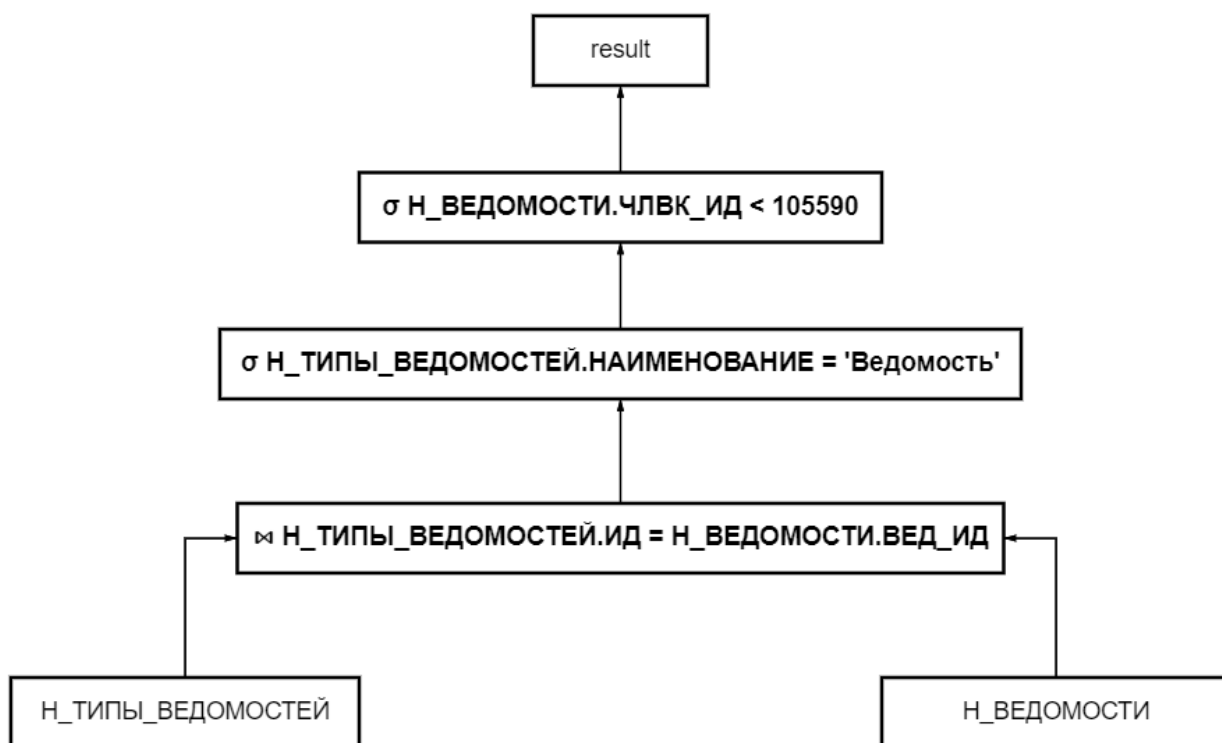
(0 строк)

QUERY PLAN

Nested Loop (cost=4.61..144.45 rows=5 width=24) (actual time=0.564..0.565 rows=0 loops=1)
Join Filter: ("Н_ОБУЧЕНИЯ"."ЧЛВК_ИД" = "Н_ЛЮДИ"."ИД")
-> Nested Loop (cost=4.33..142.77 rows=5 width=12) (actual time=0.134..0.546 rows=6 loops=1)
-> Seq Scan on "Н_ОБУЧЕНИЯ" (cost=0.00..119.76 rows=1 width=4) (actual time=0.112..0.513 rows=1 loops=1)
Filter: (("НЗК")::text = '999080'::text)
Rows Removed by Filter: 5020
-> Bitmap Heap Scan on "Н_УЧЕНИКИ" (cost=4.33..22.96 rows=5 width=8) (actual time=0.017..0.027 rows=6 loops=1)
Recheck Cond: ("ЧЛВК_ИД" = "Н_ОБУЧЕНИЯ"."ЧЛВК_ИД")
Filter: ("НАЧАЛО" < '2011-11-21'::date)
Heap Blocks: exact=6
-> Bitmap Index Scan on "УЧЕН_ОБУЧ_FK_I" (cost=0.00..4.32 rows=5 width=0) (actual time=0.010..0.010 rows=6 loops=1)
Index Cond: ("ЧЛВК_ИД" = "Н_ОБУЧЕНИЯ"."ЧЛВК_ИД")
-> Index Scan using "ЧЛВК_РК" on "Н_ЛЮДИ" (cost=0.28..0.32 rows=1 width=20) (actual time=0.003..0.003 rows=0 loops=6)
Index Cond: ("ИД" = "Н_УЧЕНИКИ"."ЧЛВК_ИД")
Filter: (("ИМЯ")::text = 'Ярослав'::text)
Rows Removed by Filter: 1
Planning Time: 0.668 ms
Execution Time: 0.612 ms
(18 строк)

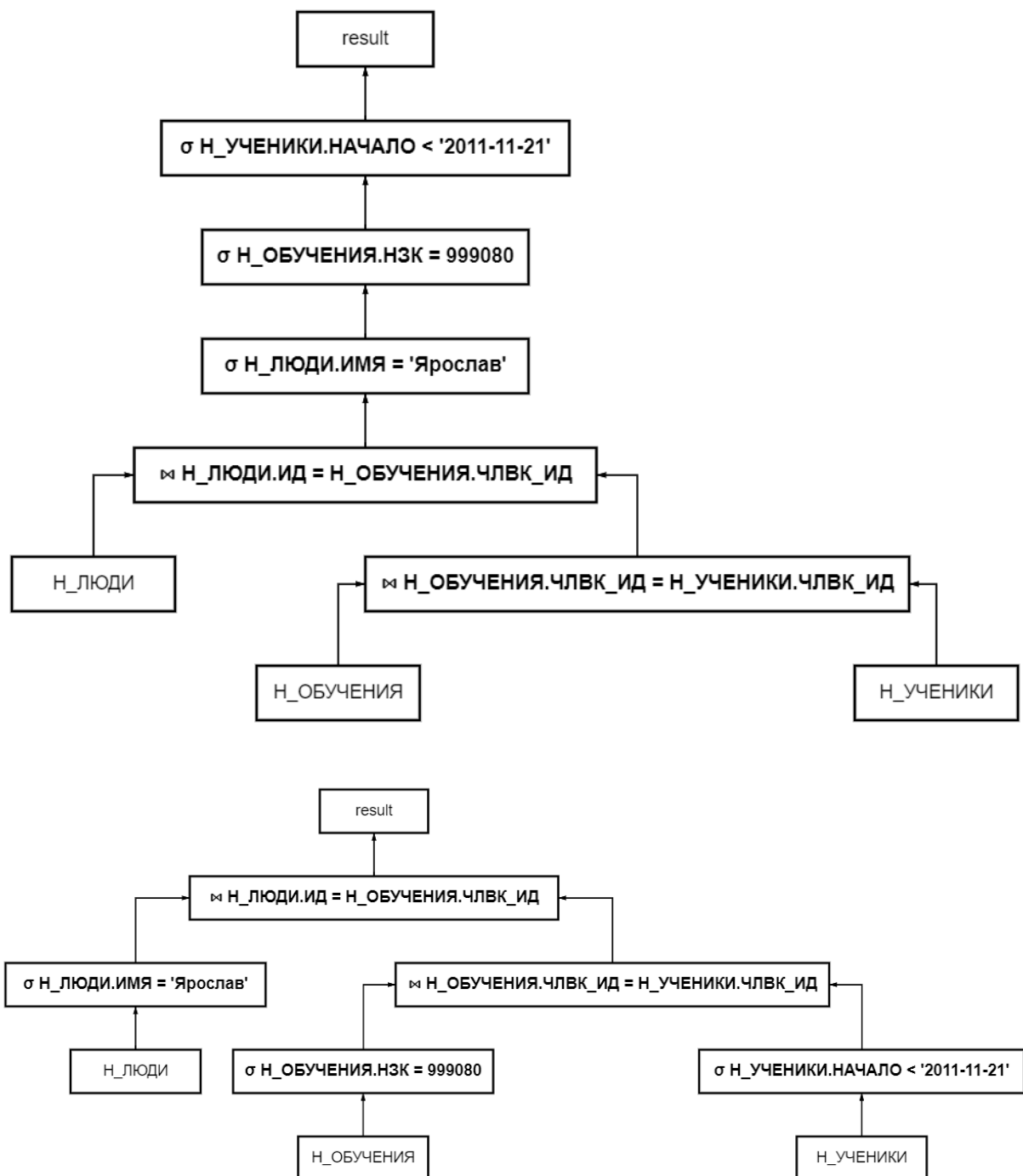
Планы выполнения запросов

1)



Оптимальным планом запроса является второй, так как мы объединяем не целые сущности, а только нужные нам выборки из них, следовательно, размер промежуточного отношения меньше.

2)



В данном запросе оптимальным планом также будет второй вариант. За счет раннего использования выборки и проекции происходит соединение не целых отношений, а только нужных нам атрибутов. Следовательно, промежуточные данные меньше.

Добавление индекса

1)

```
CREATE INDEX "ТИПЫ_ВЕД_ХЭШ" ON "Н_ТИПЫ_ВЕДОМОСТЕЙ" USING HASH ("ИД");  
CREATE INDEX "ЧЛВК_ИД_ИНД" ON "Н_ВЕДОМОСТИ" USING BTREE ("ЧЛВК_ИД");  
CREATE INDEX "ТИПЫ_ВЕДОМОСТЕЙ_ИНД" ON "Н_ТИПЫ_ВЕДОМОСТЕЙ" USING BTREE  
("НАИМЕНОВАНИЕ");
```

2)

```
CREATE INDEX "ЛЮДИ_ХЭШ" ON "Н_ЛЮДИ" USING HASH ("ИД");  
CREATE INDEX "ИМЯ_ИНД" ON "Н_ЛЮДИ" USING BTREE ("ИМЯ");  
CREATE INDEX "НЗК_ИНД" ON "Н_ОБУЧЕНИЕ" USING BTREE ("НЗК");  
CREATE INDEX "ДАТА_НАЧАЛА" ON "Н_УЧЕНИКИ" USING BTREE ("НАЧАЛО");
```

Добавление этих индексов может существенно ускорить запросы, т.к. по данным атрибутам идет выборка с операторами "=" и "<", а также соединение таблиц.

Вывод

Итак, в ходе данной лабораторной работы я ознакомился с планами выполнения запросов и индексами, а также разработал и выполнил необходимые по заданию запросы к базе данных «Учебный процесс».