|  |  |  |
| --- | --- | --- |
| Лабораторная работа №5  Индексы, хранимые процедуры, представления | Ф.И.О. | Авдеев И. А. |
| Группа | ИВТ-263 |
| Преподаватель | Соколов А.А. |
| Дата сдачи |  |

**Ход выполнения работы:**

Индексы

1. Добавить в большинство основных таблиц в БД **минимум по 1000 записей** (если в другие по логике невозможно) через файлы **CSV**с помощью генераторов данных, **Excel**или других инструментов сохранения в CSV-файлы, а также команды **COPY**(**LOAD DATA**в MySQL и т.п. в других СУБД).

2. Запустить самые сложные, составные запросы из лабораторной работы №4 и **измерить их время выполнения** с помощью приведённых в лекции инструментов.

3. Выбрать из них **5-10 самых затратных по времени**и **проанализировать**их с помощью команды **EXPLAIN**. Также использовать **визуализатор планов** в графической оболочке СУБД. Построить **визуальные планы выполнения запросов**.

Также можно выбрать те запросы, которые будут потенциально самыми**часто используемыми**, **популярными** у клиентов вашей БД (приложений), которые будет рационально ускорить для ускорения работы приложения. Например, запрос поиска книги по названию и/или автору (скорее всего, чаще книги будут искать именно так).

Добавить**скриншоты планов** в протокол, результаты вызовов функции **EXPLAIN**(особенно важно для PostgreSQL, т.к. на графическом плане не отображаются дополнительные данные)и время выполнения**до**дальнейших оптимизаций.

4. С помощью **добавления индексов** в свою БД ускорить эти запросы. Приложить в протокол **код выбранных SQL-запросов и код добавления индексов**, дать комментарии, почему были выбраны именно такие индексы.

5. **Построить новые визуальные планы** выполнения запросов с индексами и **повторно использовать команду EXPLAIN**, добавить их в протокол, а также новую измеренную скорость выполнения запросов. Проанализировать изменения в скорости выполнения запросов, сделать выводы.

Хранимые процедуры и функции

1. Написать для своей БД 3**процедуры и 3 функции**, а также написать как минимум по **1 SQL-запросу вызова**каждой из них для тестирования результатов их работы.

2. В процедуры и функции **обязательно передавать параметры,** **различных видов**, как минимум **в 3 из них использовать локальные переменные.**

3. В процедурах и функциях можно использовать созданные в 4-й лабораторной работе запросы. Как минимум **в 3 из них** обязательно использовать операторы **IF, CASE, LOOP, ISNULL**и др.

4. **В протокол включить**для каждой процедуры и функции:

* краткое описание того, что процедура или функция делает
* код процедуры или функции
* скриншот с результатами вызова процедуры или функции с данными из вашей БД

Представления

1. Создать**3 представления** с помощью оператора **VIEW**, желательно объединяющих данные из **нескольких таблиц.**

2. Написать **SQL-запросы вызова** всех созданных представлений.

3. **В протокол включить:**

* краткое описание того, что представление хранит
* код создания представления
* скриншот результата вызова SQL-запроса обращения к представлению с данными из вашей БД

1) select p.price, p.model, p.name, m.adress as 'Адрес магазина', mn.name, mn.adress as 'Адрес производителя'

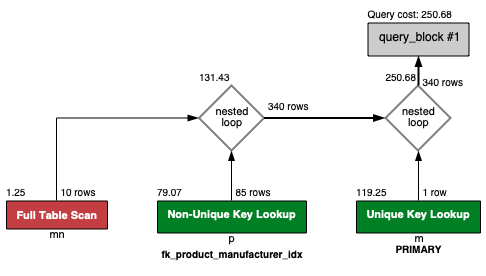
from product p

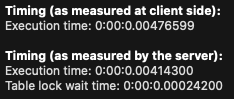
join music\_store m on p.id\_music\_store=m.idmusic\_store

join manufacturer mn on p.id\_manufacturer=mn.idmanufacturer

where p.price< 50000;

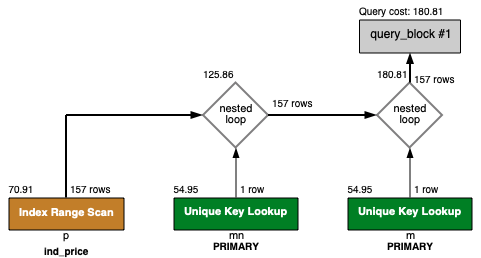
**Без индексов:**

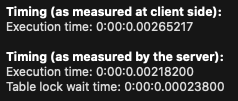
****



**С индексами:**

create index ind\_price on product(price);





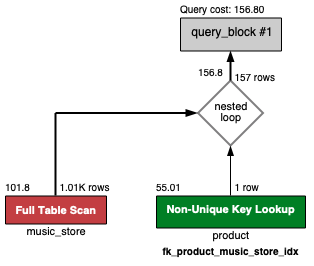
2) select product.name

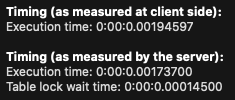
from product

join music\_store on product.id\_music\_store=music\_store.idmusic\_store

where music\_store.adress = 'Михайловка';

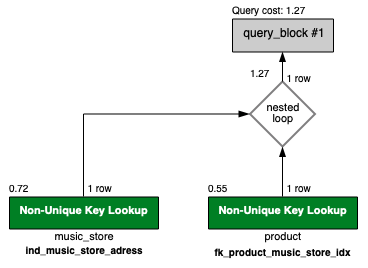
**Без индексов:**

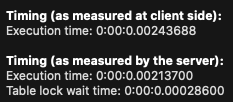




**С индексами:**

create index ind\_music\_store\_adress on music\_store(adress);





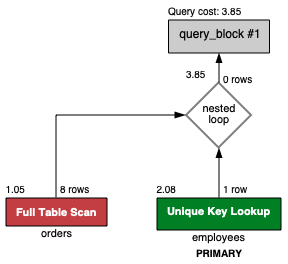
3) select orders.date

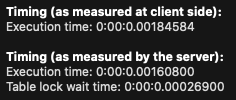
from orders

join employees on orders.id\_employee=employees.idemployees

where employees.post = 'Продавец';

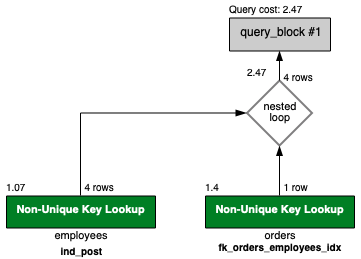
**Без индексов:**

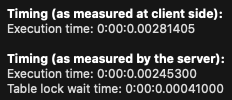
****

****

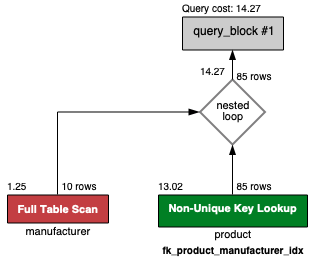
**С индексами:**

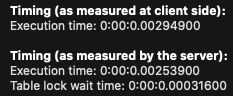
create index ind\_post on employees(post);





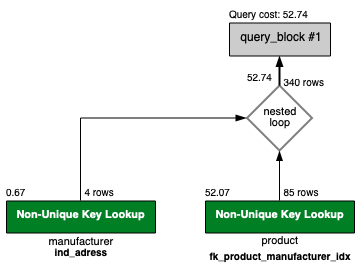
**Без индексов:**

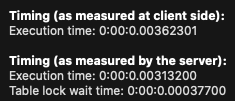




**С индексами:**

create index ind\_adress on manufacturer(adress);





**Без индексов:**

5) select product.name

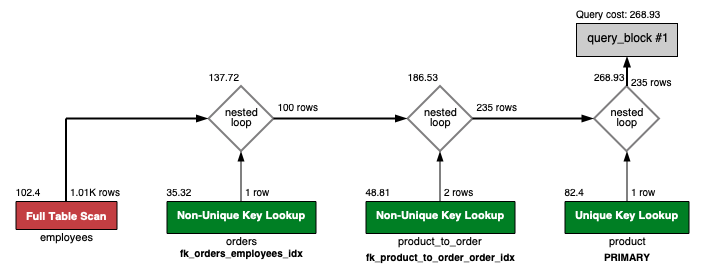
from employees

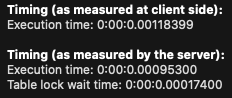
join orders on orders.id\_employee=employees.idemployees

join product\_to\_order on product\_to\_order.id\_order=orders.idorders

join product on product\_to\_order.id\_product=product.idproduct

where employees.fio = 'Doig';

****

****

**С индексами:**

create index ind\_fio on employees(fio);

