

МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра МО ЭВМ

ОТЧЕТ
по лабораторной работе №4
по дисциплине «Базы данных»
Тема: Нагрузочное тестирование БД

Студент гр. 1303

Гирман А.В.

Преподаватель

Заславский М.М.

Санкт-Петербург

2023

Цель работы.

Заполнить большим количеством тестовых данных, измерить время выполнения запросов. Измерить влияние (или его отсутствие) индексов на скорость выполнения запросов.

Задание.

Вариант 5

- Написать скрипт, заполняющий БД большим количеством тестовых данных.
- Измерить время выполнения запросов, написанных в ЛР3.

Проверить для числа записей:

100 записей в каждой таблице

1.000 записей

10.000 записей

100.000 записей

1.000.000 записей

Все запросы выполнять с фиксированным ограничением на вывод (LIMIT), т.к. запросы без LIMIT всегда будет выполняться $O(n)$ от количества записей.

Для измерения использовать фактическое (не процессорное и т.п.) время. Для node.js есть console.time и console.timeEnd.

- Добавить в БД индексы (хотя бы 5 штук). Измерить влияние (или его отсутствие) индексов на скорость выполнения запросов.

Выполнение работы.

Для выполнения работы был написан скрипт, который создаёт массив с данными и заполняет его нужным количеством информации.

Для генерации данных использовался модуль @faker-js/faker.

Измерение времени на заполнение таблиц данными:

- 100 записей – 178.347ms
- 1.000 записей – 678.541ms

- 10.000 записей – 4.466s
- 100.000 записей – 43.876s

Таблица 1. Время на выполнение запросов.

вид\Запросы	1й	2й	3й	4й	5й	6й
100 с LIMIT	10.715ms	4.238ms	5.982ms	3.095ms	4.276ms	4.505ms
1000 с LIMIT	9.984ms	3.852ms	6.299ms	3.654ms	4.393ms	7.05ms
10000 с LIMIT	10.618ms	4.242ms	7.821ms	2.987ms	5.185ms	7.647ms
100000 с LIMIT	8.771ms	4.152ms	6.775ms	3.147ms	5.352ms	7.071ms
100 с ORDER	11.052ms	4.867ms	6.325ms	2.698ms	4.218ms	5.157ms
1000 с ORDER	9.509ms	4.005ms	6.528ms	3.647ms	4.46ms	8.466ms
10000 с ORDER	10.401ms	5.898ms	8.434ms	3.912ms	23.536ms	52.442ms
100000 с ORDER	11.629ms	60.468ms	29.928ms	13.301ms	241.423ms	567.2ms
100 с INDEX	13.898ms	5.259ms	9.158ms	2.946ms	4.218ms	4.3ms
1000 с INDEX	11.857ms	5.749ms	9.507ms	3.045ms	4.474ms	6.885ms
10000 с INDEX	10.364ms	4.534ms	9.444ms	3.502ms	21.853ms	45.912ms
100000 с INDEX	11.074ms	5.098ms	21.207ms	3.285ms	506.554ms	498.622ms

Из результатов, можно заметить, что запросы с LIMIT выполняются быстрее на 100.000 количестве данных, чем с INDEX и ORDER.

Вывод.

Написана таблица, в которой записаны данные о времени выполнения запросов, при различном количестве данных и с использованием LIMIT, ORDER, INDEX.

Приложение А

Ссылки

Pull Request: <https://github.com/moevm/sql-2023-1303/pull/56>