|  |  |  |
| --- | --- | --- |
|  |  |  |
| МИНОБРНАУКИ РОССИИ | | |
| Федеральное государственное бюджетное образовательное учреждение  высшего образования  **«МИРЭА – Российский технологический университет»**  **РТУ МИРЭА** | | |

Институт Информационных технологий (ИТ)

Кафедра Инструментального и прикладного программного обеспечения

(ИиППО)

**ОТЧЕТ О ПРАКТИЧЕСКОЙ РАБОТЕ №10**

**по дисциплине**

«Технологии обработки транзакций клиент-серверных приложений»

Выполнил студент группы ИКБО-20-21 Сидоров С.Д.

Принял Маличенко С.В.

Москва 2024**ЗАДАНИЕ НА ПРАКТИЧЕСКУЮ РАБОТУ**

1. Влияние подготовки на выполнение долгого оператора. Вычислите среднюю стоимость одного заказа. Посчитайте среднее время выполнения этого запроса. Подготовьте оператор для этого запроса. Снова посчитайте среднее время выполнения. Во сколько раз ускорилось выполнение?
2. Влияние подготовки на выполнение коротких операторов. Многократно запросите данные о одном и том же заказе. Посчитайте среднее время выполнения. Подготовьте оператор для этого запроса. Снова посчитайте среднее время выполнения. Во сколько раз ускорилось выполнение в этом случае?

**РЕФЕРАТ**

Отчёт 13 страниц, 6 рисунков, 5 источников, 1 приложения.

POSTGRESQL, РАСШИРЕННЫЕ ЗАПРОСЫ, ПЛАНИРОВАНИЕ, ОПТИМИЗАЦИЯ, SQL

Объектом разработки является база данных.

Цель работы – изучение и практическое применение методов оптимизации транзакционных запросов в базах данных.

В процессе работы производилось исследование способов повышения производительности и эффективности операций с данными.

Результатом являются сведения по различным способам повышения производительности и эффективности операций с данными.

**ВВЕДЕНИЕ**

[ВВЕДЕНИЕ 6](#_Toc162362314)

[1. ВЫПОЛНЕНИЕ ПРАКТИЧЕСКОЙ РАБОТЫ 7](#_Toc162362315)

[1.1 Выполнение задания 1 7](#_Toc162362316)

[1.2 Выполнение задания 2 8](#_Toc162362317)

[ЗАКЛЮЧЕНИЕ 10](#_Toc162362318)

[СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 11](#_Toc162362319)

[ПРИЛОЖЕНИЕ А 12](#_Toc162362320)

**ПЕРЕЧЕНЬ СОКРАЩЕНИЙ И ОБОЗНАЧЕНИЙ**

БД – база данных;

СУБД – система управления базами данных.

**ВВЕДЕНИЕ**

В современном мире, где данные играют важнейшую роль в различных сферах, от бизнеса до повседневной жизни, вопрос обеспечения высокой скорости обработки большого объёма данных является одним из ключевых. Для решения задач увеличения производительности в современных СУБД применяется оптимизация запросов.

Целью данной практической работы является изучение и практическое применение методов оптимизации транзакционных запросов в базах данных.

1. **ВЫПОЛНЕНИЕ ПРАКТИЧЕСКОЙ РАБОТ****Ы**
   1. **Выполнение задания 1**

Перед выполнением задания была создана база данных, содержащая таблицу sales\_order с количеством записей 100005, что представлено на рисунке 1.

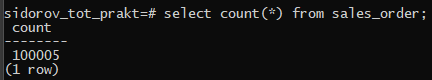


Рисунок 1 – Вычисление количества записей в таблице sales\_order

Для выполнения задания был использован цикл, проводящий расчет средней стоимости одного заказа 1000 раз, для более точного измерения. В результате исполнение заняло 6745 миллисекунд, что представлено на рисунке 2.

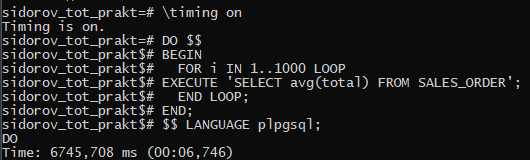


Рисунок 2 – Выполнение 1000 запросов на расчет стоимости одного заказа

Для расчета стоимости одного заказа с подготовкой оператора, также использовался цикл, проводящий расчет средней стоимости одного заказа 1000 раз. В результате исполнения заняло 6556 миллисекунд, что представлено на рисунке 3.

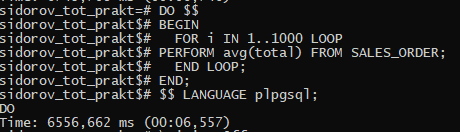


Рисунок 3 – Выполнение 1000 запросов на расчет стоимости одного заказа с подготовкой оператора

Следовательно, при выполнении долгого оператора без подготовки среднее время исполнения составило 6,7 миллисекунд, а при использовании подготовки среднее время составило 6,6 миллисекунд. Из чего следует, что при использовании долгого оператора использование предварительной подготовки влияет на время исполнения не значительно.

* 1. **Выполнение задания 2**

Перед выполнением задания была создана новая база данных с таблицей sales\_order, содержащей 105 записей, а также одна из записей была обновлена для дальнейшего поиска в запросе, что представлено на рисунке 4.



Рисунок 4 – Изменение одной из записей заказов

Для выполнения задания был использован цикл, производящий поиск заказа 100000 раз, для более точного измерения. В результате исполнение заняло 1877 миллисекунд, что представлено на рисунке 5.

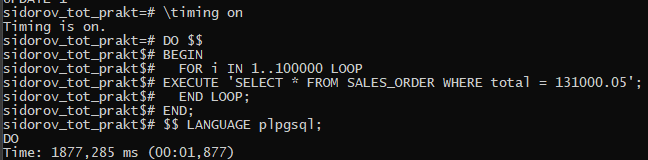


Рисунок 5 – Выполнение 100000 запросов на получение заказа

Для поиска одного заказа с подготовкой оператора, также использовался цикл, проводящий поиск одного заказа с подготовкой оператора 1000 раз. В результате исполнение заняло 888 миллисекунд, что представлено на рисунке 6.

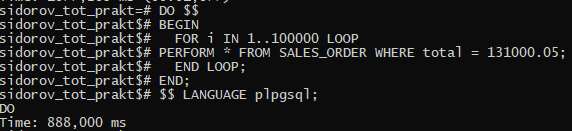


Рисунок 6 – Выполнение 100000 запросов на получение заказа с подготовкой оператора

Следовательно, при выполнении быстрого оператора без подготовки среднее время исполнения заняло 0.019 миллисекунд, а при использовании предварительной подготовки 0.0.09 миллисекунд. Из чего следует, что при использовании предварительной подготовки для простого оператора позволяет значительно уменьшить время исполнения.

# ЗАКЛЮЧЕНИЕ

В ходе выполнения практической работы были получены практические навыки по применению методов оптимизации транзакционных запросов в базах данных. Также были получены знания о способах повышения производительности и эффективности операций с данными, проводимых в рамках транзакций.

# СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. PostgreSQL: Официальная документация [Электронный ресурс] – URL: https://www.postgresql.org/docs/ (дата обращения: 14.02.2024).
2. PostgreSQL Tutorial for Beginners [Электронный ресурс] – URL: https://www.tutorialspoint.com/postgresql/index.htm (дата обращения: 14.02.2024).
3. PostgreSQL: Википедия [Электронный ресурс] – URL: https://en.wikipedia.org/wiki/PostgreSQL (дата обращения: 14.02.2024).
4. Лузанов П.В. и др. Postgres. Первое знакомство. [Электронный ресурс] – URL: https://postgrespro.ru/education/books/introbook (дата обращения: 22.02.2024)
5. Новиков Б. А. Лекции Основы технологий баз данных. [Электронный ресурс] – URL: https://postgrespro.ru/education/university/dbtech (дата обращения: 22.02.2024)

# ПРИЛОЖЕНИЕ А

**ФРАГМЕНТЫ КОДА РАЗРАБОТАННОГО ПРИЛОЖЕНИЯ**

Листинг 1 – SQL код для выполнения задания 1

\timing on

DO $$

BEGIN

FOR i IN 1..1000 LOOP

EXECUTE 'SELECT avg(total) FROM SALES\_ORDER';

END LOOP;

END;

$$ LANGUAGE plpgsql;

DO

DO $$

BEGIN

FOR i IN 1..1000 LOOP

PERFORM avg(total) FROM SALES\_ORDER;

END LOOP;

END;

$$ LANGUAGE plpgsql;

DO

\timing off

Листинг 2 – SQL код для выполнения задания 2 практической работы

UPDATE SALES\_ORDER SET total = 131000.05 where order\_id = 63;

\timing on

DO $$

BEGIN

FOR i IN 1..100000 LOOP

EXECUTE 'SELECT \* FROM SALES\_ORDER WHERE total = 131000.05';

END LOOP;

END;

$$ LANGUAGE plpgsql;

DO

DO $$

BEGIN

FOR i IN 1..100000 LOOP

PERFORM \* FROM SALES\_ORDER WHERE total = 131000.05;

END LOOP;

END;

$$ LANGUAGE plpgsql;

DO

\timing off