|  |  |  |
| --- | --- | --- |
|  |  |  |
| 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 №3**

**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. В первом сеансе начните новую транзакцию с уровнем изоляции Serializable. Вычислите количество заказов с суммой 20 000 рублей.
2. Во втором сеансе начните новую транзакцию с уровнем изоляции Serializable. Вычислите количество заказов с суммой 30 000 рублей.
3. В первом сеансе добавьте новый заказ на 30 000 рублей и снова вычислите количество заказов с суммой 20 000 рублей.
4. Во втором сеансе добавьте новый заказ на 20 000 рублей и снова вычислите количество заказов с суммой 30 000 рублей.
5. Зафиксируйте транзакции в обоих сеансах.

Соответствует ли результат ожиданиями? Можно ли сериализовать эти транзакции (иными словами, можно ли представить такой порядок последовательного выполнения этих транзакций, при котором результат совпадет с тем, что получился при параллельном выполнении)?

**РЕФЕРАТ**

Отчет 11 страниц, 7 рисунков, 5 источников, 1 приложение.

POSTGRESQL, УРОВЕНЬ ИЗОЛЯЦИИ, ТАНЗАКЦИИ, СЕРИАЛИЗАЦИЯ, SQL

Объектом разработки является база данных.

Цель работы – разработка и заполнение базы данных.

В процессе работы проводилось практическое использование базы данных в виде создания и заполнения различных таблиц.

Результатом является подготовленное рабочее место с установленным PostgreSQL и заполненная база данных.

**СОДЕРЖАНИЕ**

[**ВВЕДЕНИЕ** 6](#_Toc159594186)

[**1.** **ВЫПОЛНЕНИЕ ПРАКТИЧЕСКОЙ РАБОТЫ** 7](#_Toc159594187)

[1.1 Базовая работа с транзакциями 7](#_Toc159594188)

[1.2 Уровень изоляции Read Committed 12](#_Toc159594189)

[1.3 Уровень изоляции Repeatable Read 13](#_Toc159594190)

[1.3.1 Задание 1 13](#_Toc159594191)

[1.3.2 Задание 2 14](#_Toc159594192)

[ЗАКЛЮЧЕНИЕ 17](#_Toc159594193)

[СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 18](#_Toc159594194)

[ПРИЛОЖЕНИЕ А 19](#_Toc159594195)

[**ФРАГМЕНТЫ КОДА РАЗРАБОТАННОГО ПРИЛОЖЕНИЯ** 19](#_Toc159594196)

**ПЕРЕЧЕНЬ СОКРАЩЕНИЙ И ОБОЗНАЧЕНИЙ**

БД – база данных;

СУБД – система управления базами данных.

**ВВЕДЕНИЕ**

В современной сфере баз данных, PostgreSQL занимает выдающуюся позицию, предоставляя мощный инструмент для управления информацией. Эта система управления базами данных предлагает механизм транзакций, который играет ключевую роль в обеспечении последовательности и консистентности изменений в базе данных.

Транзакции в PostgreSQL выполняют множество задач, включая поддержку атомарности, согласованности, изолированности и долговечности (ACID). Эти принципы являются фундаментальными для обеспечения надёжности и предсказуемости работы с данными. В данном контексте, изучение и понимание принципов транзакций в PostgreSQL становится необходимым для разработчиков и администраторов баз данных, стремящихся обеспечить эффективное управление информацией в сложных и динамичных проектах.

Целью данной работы является практическое исследование аспектов работы с транзакциями в базах данных на примере системы управления PostgreSQL. Работа сфокусирована на сценариях использования уровня изоляции Serializable и отличиях этого уровня от других.

1. **ВЫПОЛНЕНИЕ ПРАКТИЧЕСКОЙ РАБОТЫ**

Перед выполнением практической работы база данных была приведена в изначальное состояние. Содержимое таблицы sales\_order продемонстрировано на рисунке 1.

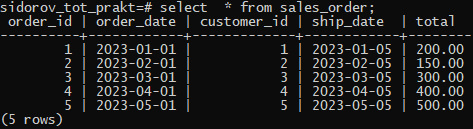


Рисунок 1 – Содержимое таблицы sales\_order до начала работы

После отчистки таблицы, внутри первого сеанса была начата транзакция с уровнем изоляции Serializable и вычислено количество заказов на сумму 20000, что продемонстрировано на рисунке 2.

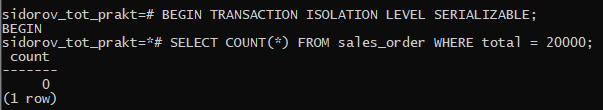


Рисунок 2 – Начало транзакции в сеансе 1 с уровнем изоляции Serializable и вычисление количества заказов с суммой 20000

Параллельно с сеансом 1, был создан второй сеанс, в котором была начата новая транзакция с уровнем изоляции Serializable, а также вычислено количество заказов с суммой равной 30000, что представлено на рисунке 3.

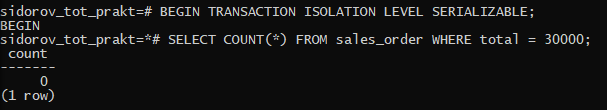


Рисунок 3 – Начало транзакции в сеансе 2 с уровнем изоляции Serializable и вычисление количества заказов с суммой 30000

После был выполнено добавление заказа на сумму 30000 в первом сеансе и вычисление количества заказов с суммой равно 20000, которых также не оказалось, что представлено на рисунке 4.

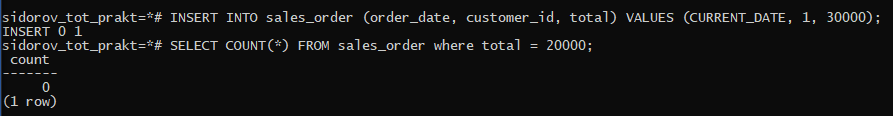


Рисунок 4 – Добавление заказа на 30000 в первом сеансе и вычисление количества заказов с суммой 20000

Далее, был выполнен переход во второй сеанс, где был добавлен заказ на сумму 20000 и были вычислены заказы с суммой равной 30000, которых также не оказалось, что представлено на рисунке 5.

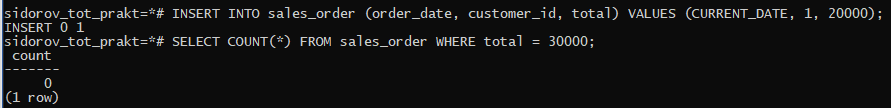


Рисунок 5 - Добавление заказа на 20000 во втором сеансе и вычисление количества заказов с суммой 30000

После всех изменений, в первом сеансе была произведена фиксация, которая прошла успешно, что видно на рисунке 6.



Рисунок 6 – Фиксация изменений в первом сеансе

После фиксации изменений в первом сеансе, была произведена попытка произвести фиксацию во втором сеансе. Система не дала произвести транзакцию, так как их не возможно сериализовать из-за взаимного внесения изменения в ту часть таблицы, которую читая другая транзакция, что продемонстрировано на рисунке 7.

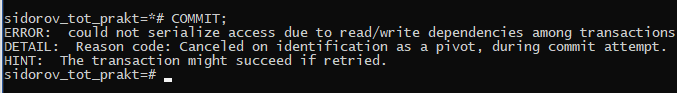


Рисунок 7 – Попытка фиксации изменений во втором сеансе

# ЗАКЛЮЧЕНИЕ

В ходе выполнения практической работы было осуществлено детальное изучение процессов транзакций в системе управления базами данных PostgreSQL. Была проведена работа с использованием транзакций с уровнем изоляции Serializable, были получены знания по использованию данного уровня изоляции и по дополнительным ограничениям накладываемым им.

# 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 практической работы

--Сессия 1--

BEGIN TRANSACTION ISOLATION LEVEL SERIALIZABLE;

SELECT COUNT(\*) FROM sales\_order WHERE total = 20000;

--Сессия 2--

BEGIN TRANSACTION ISOLATION LEVEL SERIALIZABLE;

SELECT COUNT(\*) FROM sales\_order WHERE total = 30000;

--Сессия 1--

INSERT INTO sales\_order (order\_date, customer\_id, total) VALUES (CURRENT\_DATE, 1, 30000);

SELECT COUNT(\*) FROM sales\_order where total = 20000;

--Сессия 2--

INSERT INTO sales\_order (order\_date, customer\_id, total) VALUES (CURRENT\_DATE, 1, 20000);

SELECT COUNT(\*) FROM sales\_order WHERE total = 30000;

--Сессия 1--

COMMIT;

--Сессия 2--

COMMIT;