

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение

высшего образования

**«МИРЭА – Российский технологический университет»**

**РТУ МИРЭА**

|  |
| --- |
|  |

Институт информационных технологий

Кафедра инструментального и прикладного программного обеспечения (ИППО)

**ОТЧЕТ**

**ПО ПРАКТИЧЕСКОЙ РАБОТЕ №2**

**по дисциплине**

«Технологии обработки транзакций клиент-серверных приложений»

Выполнил студент группы ИКБО-16-21 Белослудцев Е.Д.

Принял Маличенко С. В.

Практическая работа выполнена «\_\_» \_\_\_\_\_\_\_ 2024 г.

«Зачтено» «\_\_» \_\_\_\_\_\_\_ 2024 г.

Москва 2024

**ЗАДАНИЕ НА ПРАКТИЧЕСКУЮ РАБОТУ**

Цель данной работы заключается в изучении уровней изоляции транзакций, точек сохранения, а также исследовании их влиянии на обработку данных в базе.

Необходимо выполнить следующие задачи:

* + 1. Начать транзакцию (командой BEGIN) и создать новый заказ в таблице sales\_order с сегодняшней датой. Добавить два предмета в таблицу item, связанных с созданным заказом. Представить, что пользователь не подтвердил заказ и все введенные данные необходимо отменить. Выполнить отмену транзакции и проверить, что никакой добавленной информации действительно не осталось.
    2. Представить сценарий, в котором нужно отменить не все данные, а только последний из добавленных предметов. Для этого повторить все действия из предыдущего упражнения, но перед добавлением каждого предмета создавать точку сохранения (с одним и тем же именем). После ввода второго предмета выполнить откат к точке сохранения. Проверить, что заказ и первый предмет остались.
    3. В рамках той же транзакции добавить еще один предмет и зафиксировать транзакцию
    4. Перед началом выполнения задания проверить, что в таблице sales\_order нет заказов на сумму to 1 000 рублей.
    - в первом сеансе начать транзакцию (командой BEGIN). Выполнить обновление таблицы sales\_order: увеличить total в два раза в тех строках, где сумма равна 1 000 рублей.
    - во втором сеансе начать транзакцию (командой BEGIN). Вставить в таблицу sales\_order новый заказ на 1 000 рублей и зафиксировать транзакцию.
    - в первом сеансе повторить обновление таблицы sales\_order и зафиксировать транзакцию.
    1. Повторить предыдущее упражнение, но начать транзакцию в первом сеансе с уровнем изоляции транзакций Repeatable Read.
    2. Выполнить указанные действия в двух сеансах:
    - в первом сеансе начать новую транзакцию с уровнем изоляции Repeatable Read. Вычислить количество заказов с суммой 20 000 рублей.
    - во втором сеансе начать новую транзакцию с уровнем изоляции Repeatable Read. Вычислить количество заказов с суммой 30 000 рублей.
    - в первом сеансе добавить новый заказ на 30 000 рублей и снова вычислить количество заказов с суммой 20 000 рублей.
    - во втором сеансе добавить новый заказ на 20 000 рублей и снова вычислить количество заказов с суммой 30 000 рублей.
    - зафиксировать транзакции в обоих сеансах.

**РЕФЕРАТ**

Белослудцев Е.Д., Практическая работа направления подготовки «Программная инженерия» на тему «Создание базы данных»: М. 2024 г., МИРЭА – Российский технологический университет (РТУ МИРЭА), Институт информационных технологий (ИИТ), кафедра инструментального и прикладного программного обеспечения (ИиППО) – 22 стр., 22 рис., 8 источн. (в т.ч 1 на английском яз.)

Ключевые слова: POSTGRESQL, БАЗА ДАННЫХ, ТАБЛИЦА, SQL, READ COMMITTED, REPEATABLE READ, ИЗОЛЯЦИЯ, ТРАНЗАКЦИЯ.

Целью работы является изучение уровней изоляции транзакций, точек сохранения, а также исследование их влияния на обработку данных в базе.

Belosludtsev E.D., Practical work of the direction of training “Software Engineering” on the theme “Database creation”: M. 2024, MIREA – Russian Technological University (RTU MIREA), Institute of Information of Instrumental and Applied Software (IAPS) – 22 p., 22 fig., 8 sources (inc. 1 in English)

Keywords: POSTGRESQL, DATABASE, TABLE, SQL, READ COMMITTED, REPEATABLE READ, ISOLATION, TRANSACTION.

The aim of the work is to study transaction isolation levels, save points and analyze its effects on data processing in the database.

**ПЕРЕЧЕНЬ СОКРАЩЕНИЙ**

|  |  |  |
| --- | --- | --- |
| SQL | — | Structured Query Language |
| СУБД | — | Система управления базой данных |

**ПЕРЕЧЕНЬ ТЕРМИНОВ И ОПРЕДЕЛЕНИЙ**

1. **База данных** – совокупность данных, хранимых в соответствии со схемой данных, манипулирование которыми выполняют в соответствии с правилами средств моделирования данных.
2. **Таблица** – совокупность связанных данных, хранящихся в структурированном виде в базе данных.
3. **SQL-скрипт** – сохраненная в текстовом формате SQL инструкция, с помощью которой создаются объекты базы данных.
4. **Транзакция** – последовательность операторов языка SQL, которая рассматривается как некоторое неделимое действие над базой данных, осмысленное с точки зрения пользователя.
5. **Изоляция** – свойство, за счет соблюдения которого СУБД способна параллельно обслуживать много сеансов, не жертвуя корректностью данных.
6. **Атомарность** – свойство, за счет которого любая транзакция выполняется целиком, либо не выполняется совсем.
7. **Согласованность** – свойство, благодаря которому в начале работы транзакции данные находятся в согласованном состоянии, и в конце своей работы транзакция оставляет их согласованными.
8. **Долговечность** – свойство, которое гарантирует, что зафиксированные данные не пропадут даже в случае сбоя.

**СОДЕРЖАНИЕ**

[ВВЕДЕНИЕ 8](#_Toc159951947)

[ХОД РАБОТЫ 9](#_Toc159951948)

[ЗАКЛЮЧЕНИЕ 21](#_Toc159951949)

[СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 22](#_Toc159951950)

# ВВЕДЕНИЕ

Уровни изоляции транзакций и точки сохранения играют значительную роль в обеспечении целостности данных и предотвращении конфликтов при одновременном доступе нескольких транзакций. Они определяют, какие изменения, сделанные одной транзакцией, будут видимы другим транзакциям. Точки сохранения, также известные как точки восстановления, позволяют сохранить состояние базы данных на определенный момент времени.

В ходе работы будут изучены различные уровни изоляции, такие как Read Committed и Repeatable Read. Будут проведены эксперименты, чтобы проанализировать, как эти уровни влияют на параллельное выполнение транзакций и общую консистентность данных.

# ХОД РАБОТЫ

Для начала проверим наполнение таблиц sales\_order и item перед началом выполнения транзакции (рис. 1 и рис.2)

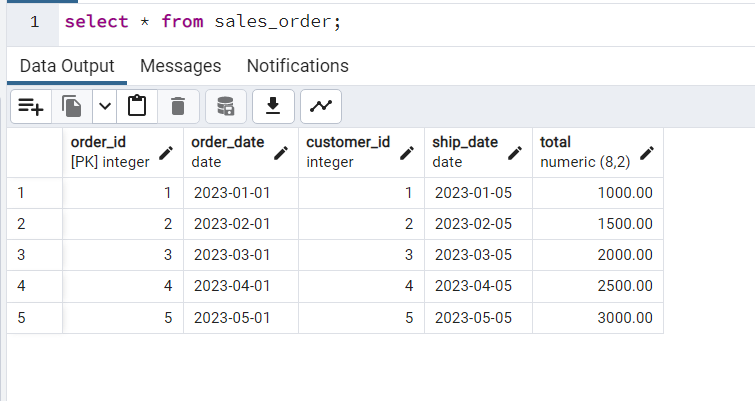


Рисунок 1 – Таблица sales\_order перед началом транзакции

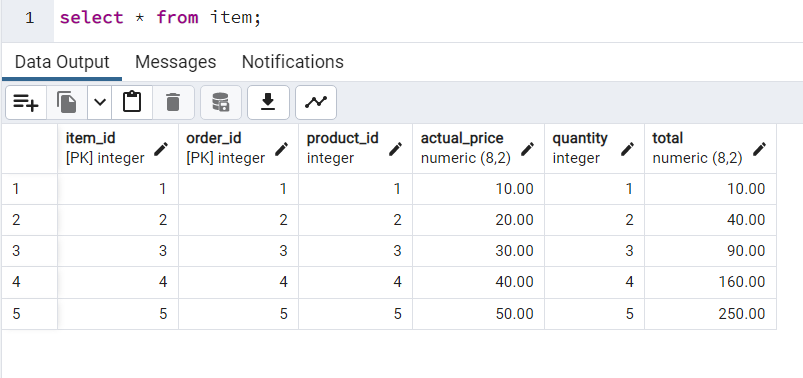


Рисунок 2 – Таблица item перед началом транзакции

Далее начнем, транзакцию, в которой будет выполнена вставка нового заказа, а также добавление в него двух предметов (рис. 3)

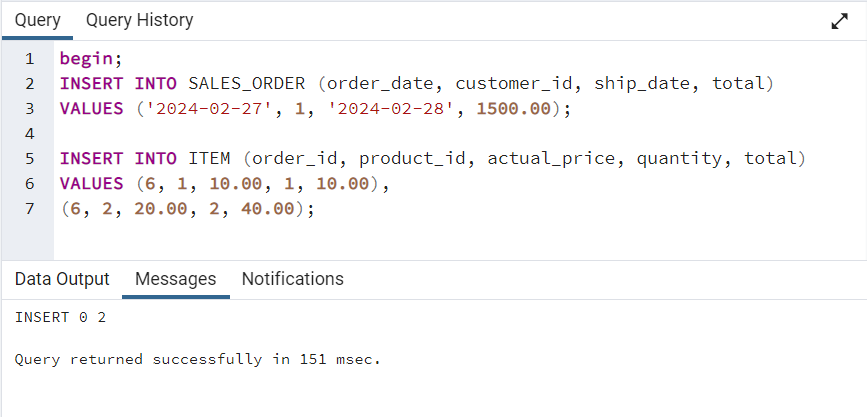


Рисунок 3 – Начало транзакции с добавлением в таблицы sales\_order и item новых данных

Также необходимо откатить транзакцию и проверить данные в таблицах – рисунки 4 – 5.

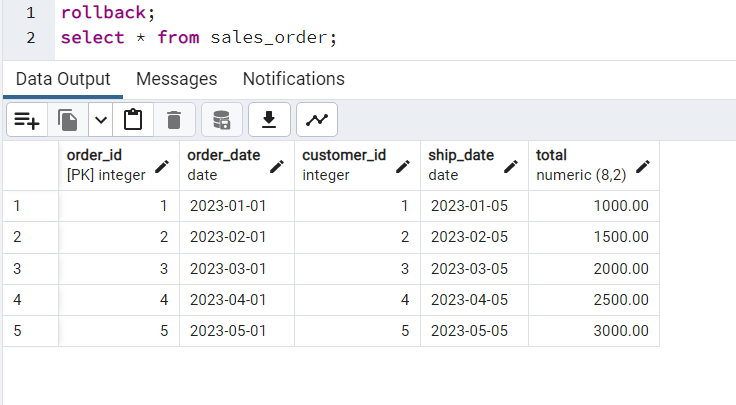


Рисунок 4 – Данные в таблице sales\_order после отката транзакции

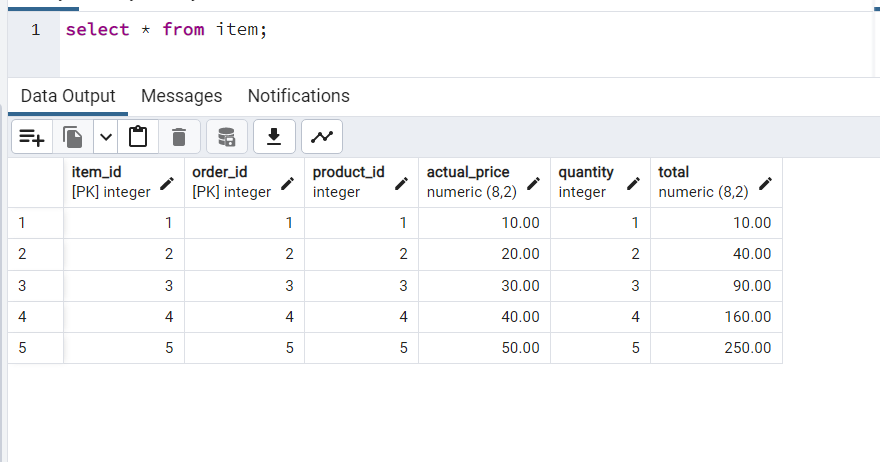


Рисунок 5 – Данные в таблице item после отката транзакции

Теперь необходимо отменить только последний добавленный предмет. Для этого перед каждым добавлением будет создана точка сохранения и произведен откат после вставки второго предмета.

Для начало начнем транзакцию и добавить новые данные в таблицу sales\_order – рисунок 6.

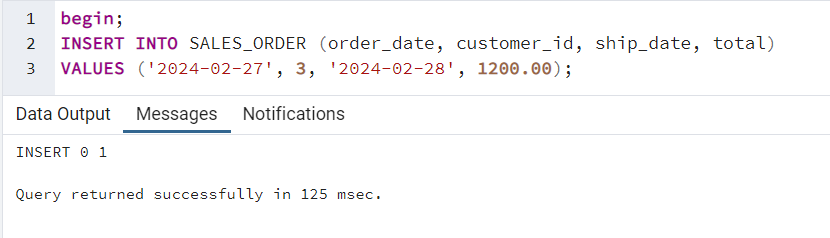


Рисунок 6 – Начало транзакции и добавление данных в таблицу sales\_order

Теперь добавим два предмета в таблицу item, связанных с созданным заказом, не забыв создать точку сохранения после каждого успешного выполнения – рисунок 7.

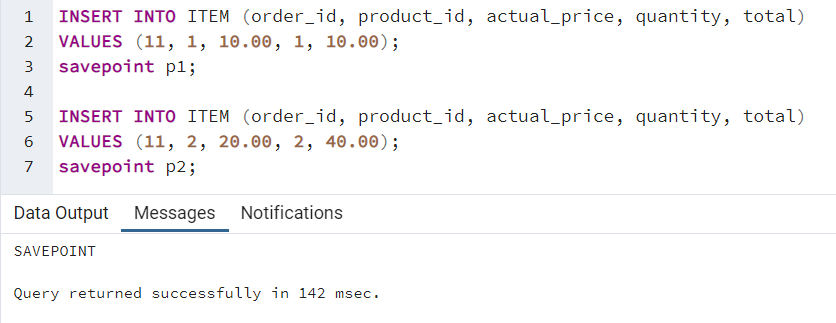


Рисунок 7 – Добавление двух товаров в таблицу item и добавление точек сохранения

Теперь произведем частичный откат транзакции на одну точку сохранения и проверим содержимое двух таблиц – рисунки 8 и 9.

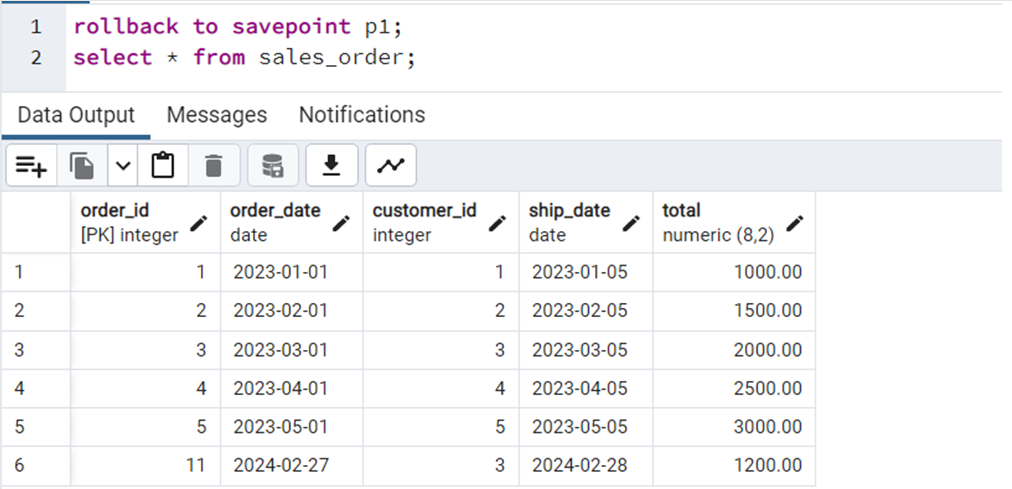


Рисунок 8 – Содержимое таблицу sales\_order после частичного отката

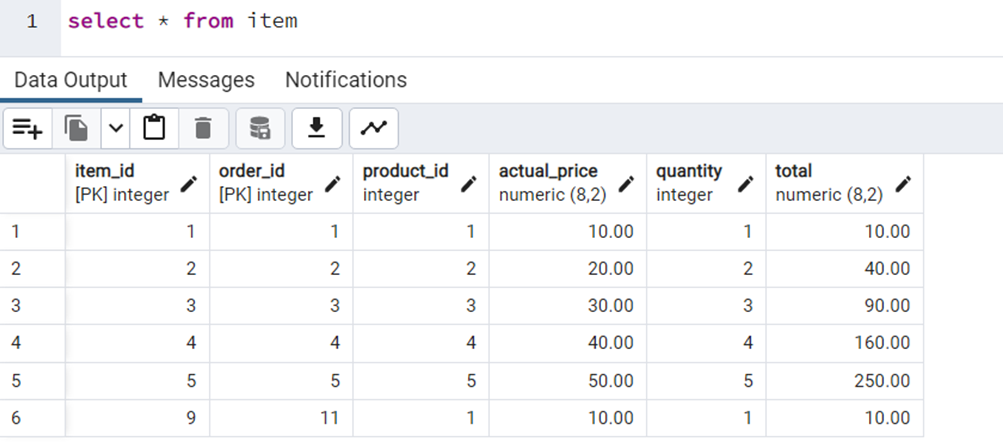


Рисунок 9 – Содержимое таблицу item после частичного отката

Наконец, добавляется еще один предмет и транзакция фиксируется – рисунок 10.

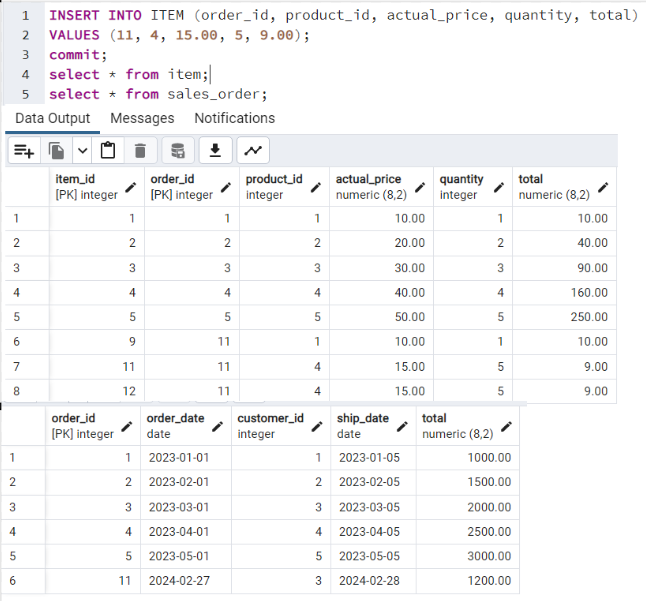


Рисунок 10 – Добавление предмета и коммит

Следующим шагом будет изучен уровень изоляции Read Committed. В первом сеансе начата транзакция по удвоению поля total записей, у которых это поле имеет значение 1000 – рисунок 11.

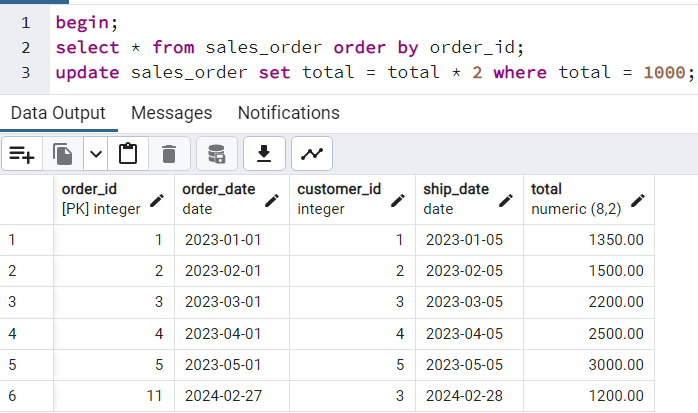


Рисунок 11 – Исходные данные и начало первой транзакции

Также параллельно с этим будет выполнена транзакция по вставке нового заказа, значение поля total которого равно 1000 – рисунок 12.

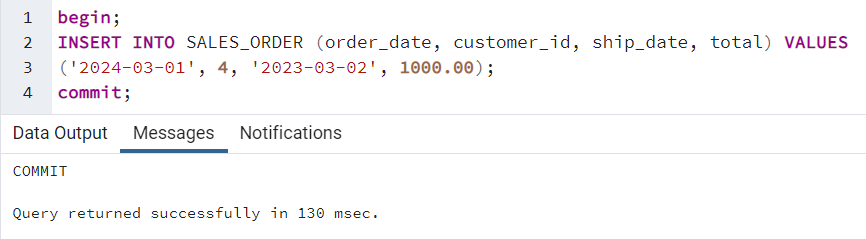


Рисунок 12 – Выполнение второй транзакции

Наконец, в первом сеансе будет повторно произведено обновление таблицы и зафиксирована транзакция – рисунок 13.

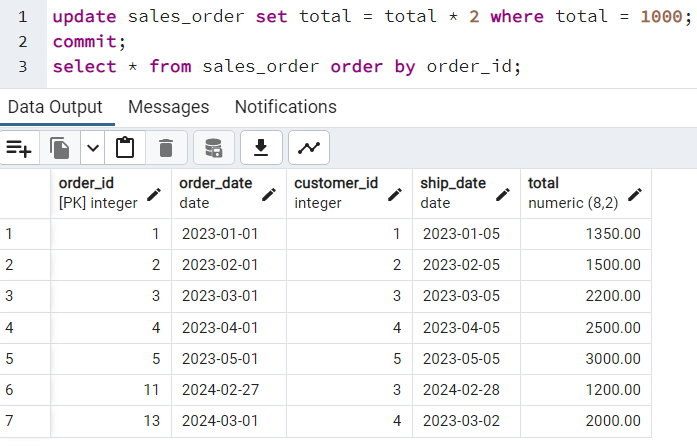


Рисунок 13 – Результат после повторного обновления и фиксации первой транзакции

По результатам выполнения транзакций, сумма заказа стала равной 2000 рублей. Такой эффект допускается при уровне изоляции Read Committed, так как во время выполнения одной транзакции в ее область видимости входят зафиксированные изменения других.

Последний шаг в данной работе – изучить уровень изоляции транзакций Repeatable Read. Для начала будет также рассмотрен случай с заказом и полем total со значением 1000, но на этот раз с данным уровнем изоляции (рис. 14-16)

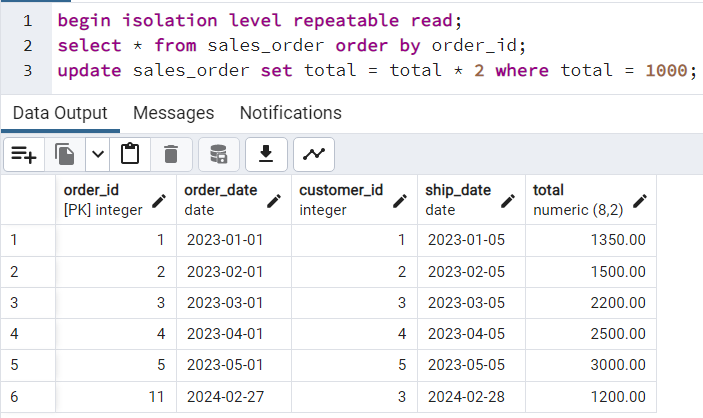


Рисунок 14 – Начало первой транзакции

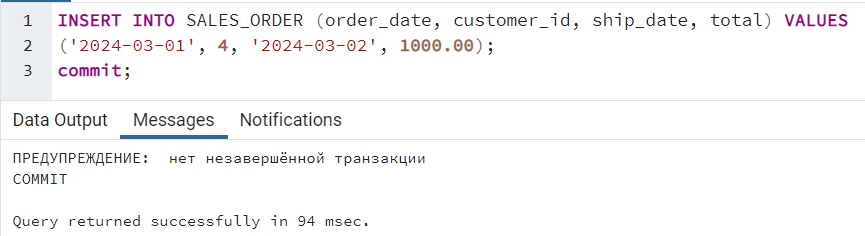


Рисунок 15 – Выполнение второй транзакции

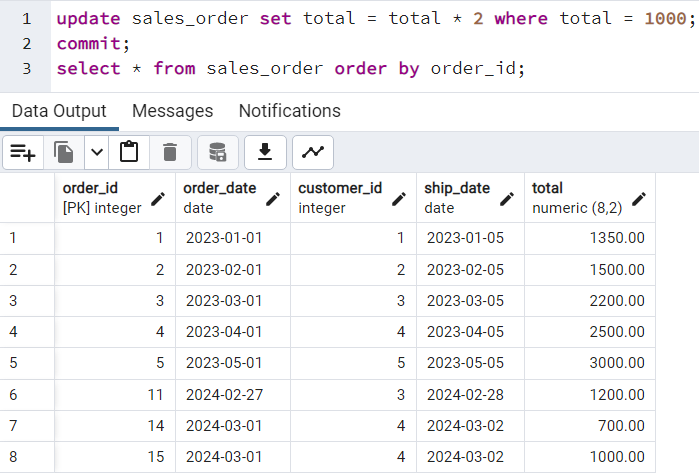


Рисунок 16 – Результат после повторного обновления и фиксации первой транзакции

В данном случае, по результатам выполнения транзакций поле total сохранило значение 1000, так как первая транзакция изолирована от изменений, зафиксированных другими транзакциями после начала текущей. В этом и проявилось отличие между уровнями изоляции Read Committed и Repeatable Read.

В качестве последнего упражнения также будут параллельно выполнены две транзакции. В первом сеансе начата транзакция с уровнем изоляции Repeatable Read и вычислено количество заказов с суммой 20 000 рублей – рисунок 18.

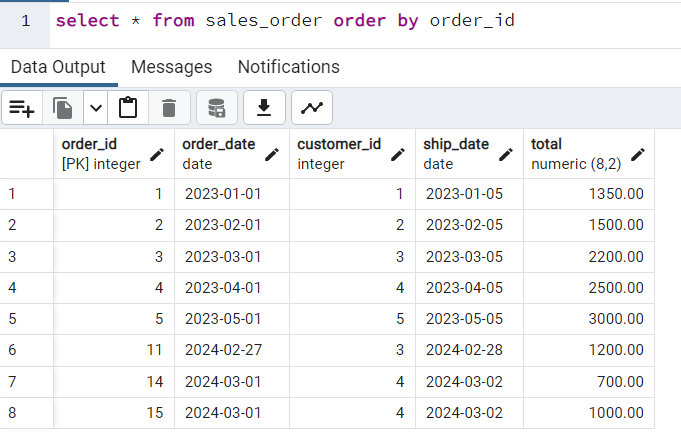


Рисунок 17 – Исходные данные таблицы sales\_order

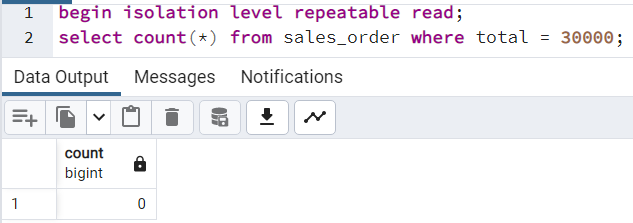


Рисунок 18 – Начало первой транзакции

Во втором сеансе начата транзакция с таким же уровнем изоляции, вычислено количество заказов с суммой 30 000 рублей – рисунок 19.

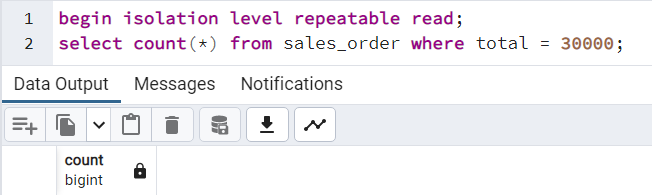


Рисунок 19 – Начало второй транзакции

Далее, в первом сеансе добавлен новый заказ на 20 000 рублей и снова вычислено количество заказов с суммой 30 000 рублей – рисунок 20.

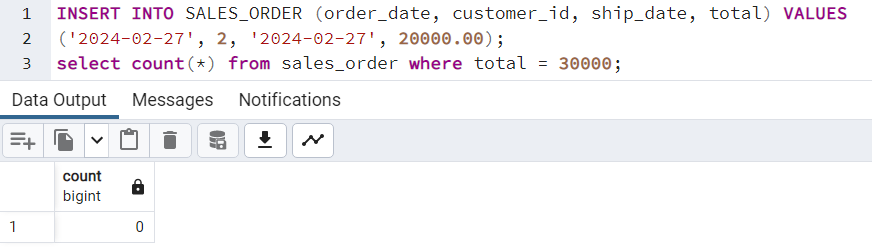


Рисунок 20 – Добавление заказа и повторное вычисление (первая транзакция)

Во втором сеансе также добавлен заказ (total = 30 000) и вычислено количество заказов с полем total = 20 000 – рисунок 21.

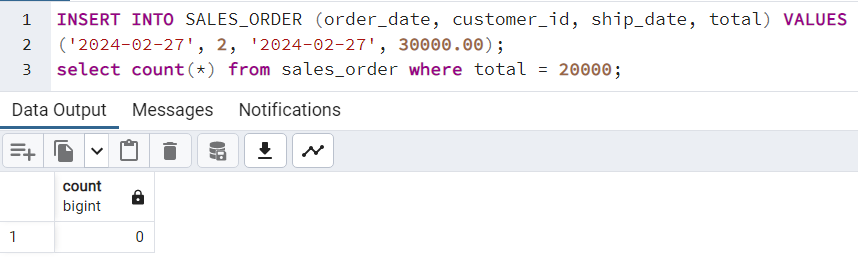


Рисунок 21 – Добавление заказ и повторное вычисление (вторая транзакция)

Наконец, произведена фиксация обеих транзакций и выведена итоговая таблица с полученными данными – рисунок 22.

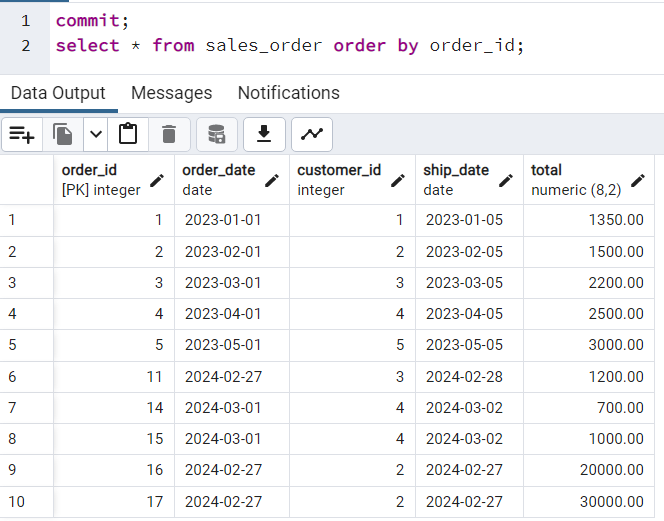


Рисунок 22 – Результат фиксации изменений

Благодаря уровню изоляции Repeatable Read данные, прочитанные в начале транзакции, остаются постоянными до ее завершения.

# ЗАКЛЮЧЕНИЕ

В результате данной работы проведены эксперименты и изучены уровни изоляции транзакций и точек сохранения. Уровни изоляции позволяют балансировать между параллелизмом выполнения транзакций и обеспечением консистентности данных. Точки сохранения предоставляют возможность восстановления базы данных до определенного момента, что полезно в случае ошибок или отката транзакций. Грамотное использование уровней изоляции и точек сохранения является ключевым элементом обеспечения стабильной и эффективной работы баз данных в различных сценариях использования.

# СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Новиков Б. А. Основы технологий баз данных: учеб. пособие / Б. А. Новиков, Е. А. Горшкова, Н. Г. Графеева; под ред. Е. В. Рогова. — 2-е изд. — М.: ДМК Пресс, 2020. — 582 с. – URL: <https://postgrespro.ru/education/books/dbtech> (дата обращения: 18.02.2024)
2. Новиков Б. А. Лекции Основы технологий баз данных. – URL: <https://postgrespro.ru/education/university/dbtech> (дата обращения: 17.12.2023)
3. Моргунов, Е. П. PostgreSQL. Основы языка SQL: учеб. пособие / Е. П. Моргунов; под ред. Е. В. Рогова, П. В. Лузанова. — СПб.: БХВ-Петербург, 2018. — 336 с.: ил. – URL: <https://postgrespro.ru/education/books/sqlprimer> (дата обращения: 18.02.2024)
4. Моргунов, Е. П. Технологии разработки программ на основе инструментария с открытым исходным кодом. Вводный курс: учеб. пособие / Е. П. Моргунов, О. Н. Моргунова, В. В. Тынченко; НИИ СУВПТ. – Красноярск, 2006. – 148 с. – URL: <http://www.morgunov.org/docs/free_soft_tech.pdf> (дата обращения: 18.02.2024)
5. Лузанов П.В. и др. Postgres. Первое знакомство. – 178 с. – URL: <https://postgrespro.ru/education/books/introbook> (дата обращения: 17.12.2023)
6. BEGIN, COMMIT, ROLLBACK (работа с транзакциями) – URL: <https://ydb.tech/docs/ru/postgresql/statements/begin_commit_rollback> (дата обращения: 18.02.2024)
7. Postgres Pro Standard: 13.2 Изоляция транзакций – URL: <https://postgrespro.ru/docs/postgrespro/16/transaction-iso> (дата обращения: 18.02.2024)
8. Isolation Levels – part IX: Read Committed – URL: <https://dev.to/franckpachot/isolation-levels-part-ix-read-committed-3lll> (дата обращения: 18.02.2024)