Министерство науки и высшего образования РФ

Федеральное государственное автономное

образовательное учреждение высшего образования

**«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»**

Институт космических и информационных технологий

институт

Программная инженерия

кафедра

**ОТЧЁТ О ПРАКТИЧЕСКОЙ РАБОТЕ №7**

Транзакции в PostgreSQL.

тема

Преподаватель А. Д. Вожжов

подпись, дата инициалы, фамилия

Студент КИ23-17/2Б, 032319859 В. А. Серебров

номер группы, зачетной книжки подпись, дата инициалы, фамилия

Красноярск 2025

**СОДЕРЖАНИЕ**

[1 Цели 3](#_Toc191494455)

[2 Задачи 3](#_Toc191494456)

[3 Ход работы 3](#_Toc191494457)

[4 Вывод 8](#_Toc191494458)

# 1 Цели

В рамках данной практической работы необходимо изучить теоретический материал по работе с транзакциями в СУБД PostgreSQL.

# 2 Задачи

Для выполнения практической работы необходимо выполнить следующие задачи:

- изучить учебный материал (глава 9 учебного пособия);

- выполнить контрольные задания (с нечетными номерами).

# 3 Ход работы

На рисунке 1 представлены две параллельно работающие транзакции, при чем в первой транзакции происходит блокировка строк через FOR UPDATE.

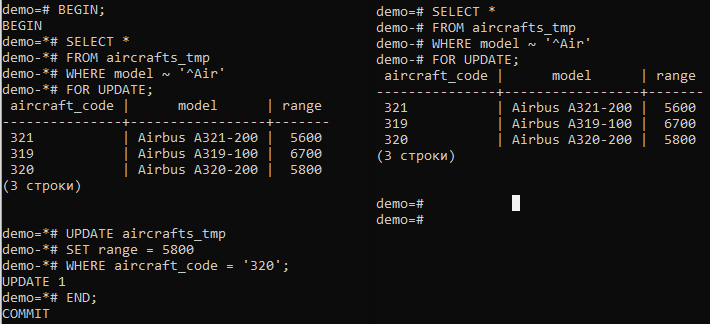


Рисунок 1 – Транзакции и блокировка строк

Кроме того, можно заблокировать всю таблицу. На рисунке 2 представлены две параллельно работающие транзакции с использованием блокировки всей таблицы.

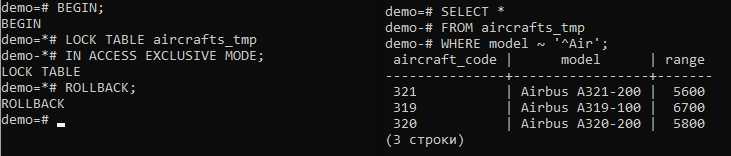


Рисунок 2 – Транзакции с блокировкой таблицы

На рисунке 3 представлены одновременные изменения строки в двух параллельных транзакциях.

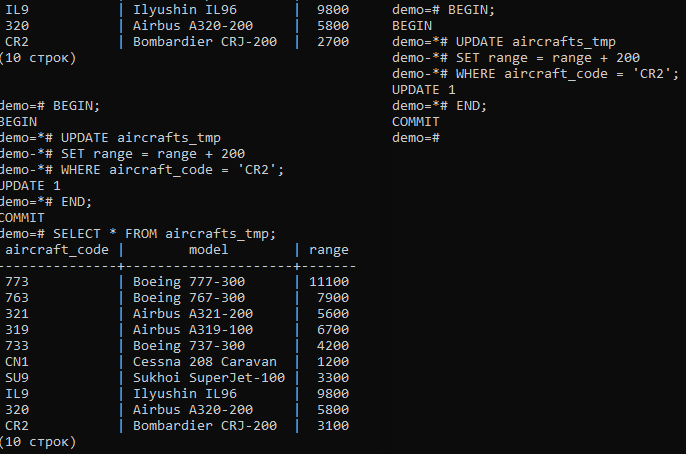


Рисунок 3 – Изменение данных

Видно, что оба изменения сохранились и феномен потерянного обновления не проявился.

На рисунке 4 мы обновляли данные в строке на конкретные значения.

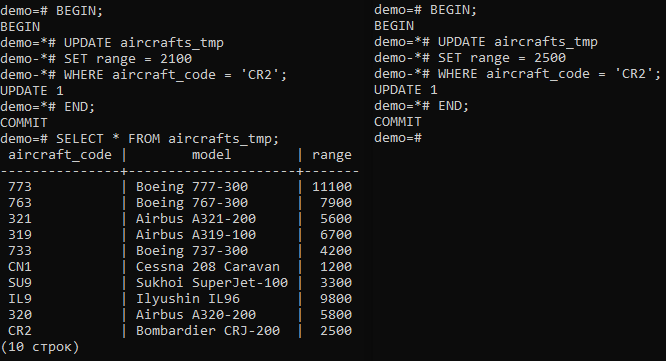


Рисунок 4 – Обновление данных

Получается, что мы потеряли значение 2100 при обновлении данных во второй транзакции. Чтобы исправить это, можно использовать блокировку строк или всей таблицы.

На рисунке 5 представлена работа команды SELECT … FOR UPDATE в случае, когда выбираемое в второй транзакции множество строк является подмножеством множества строк, выбираемых в первой транзакции.

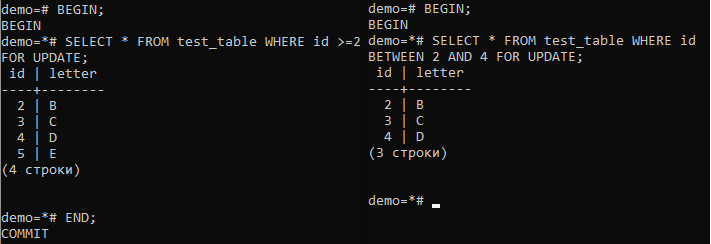


Рисунок 5 – Подмножество

На рисунке 6 представлена работа команды SELECT … FOR UPDATE в случае, когда выбираемое в второй транзакции множество строк является надмножеством множества строк, выбираемых в первой транзакции.

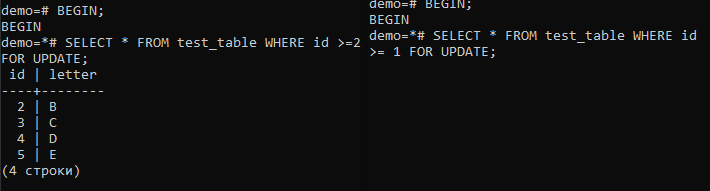


Рисунок 6 – Надмножество

На рисунке 7 представлена работа команды SELECT … FOR UPDATE в случае, когда выбираемое в второй транзакции множество строк является пересечением множества строк, выбираемых в обоих транзакции.

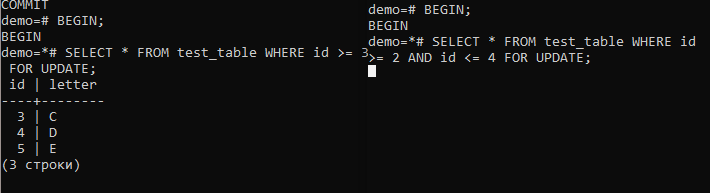


Рисунок 7 – Пересечение

На рисунке 8 представлена работа команды SELECT … FOR UPDATE в случае, когда выбираемое в второй транзакции множество строк не пересекается с множеством строк, выбираемых в первой транзакции.

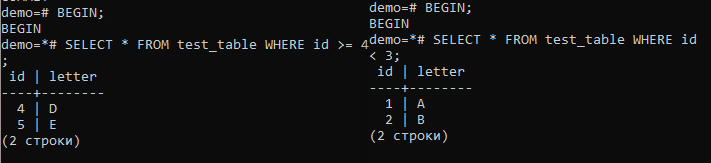


Рисунок 8 – Множества строк не пересекаются

На рисунке 9 видно, что при параллельном запуске нескольких транзакций, пока не закончится одна, другая не сможет обновить данные в строке.

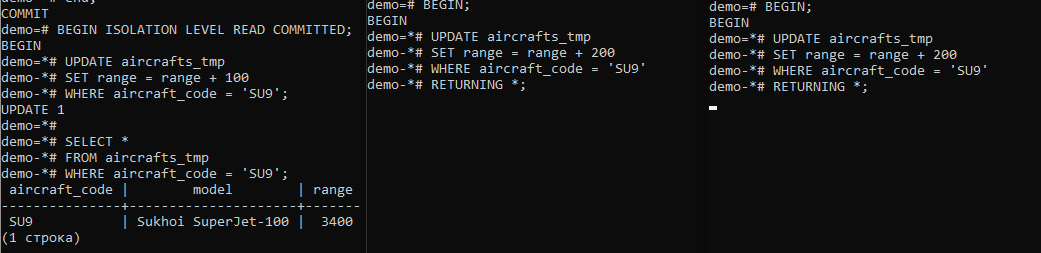


Рисунок 9 – Транзакции

На рисунке 10 представлен результат обновления данных строке после каждой из трех транзакций.

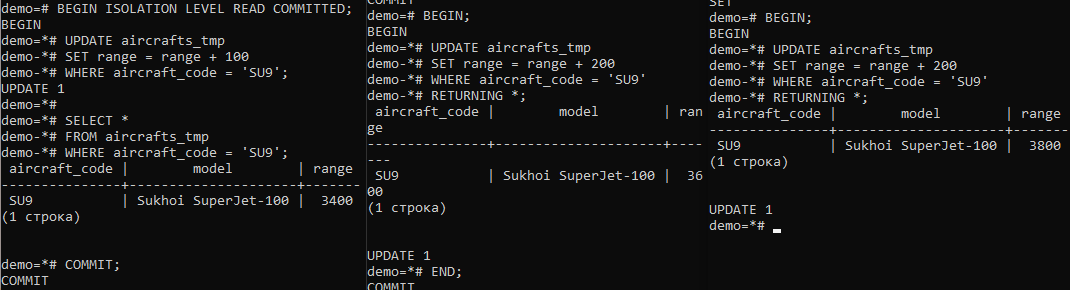


Рисунок 10 – Результат

На рисунке 11 представлен результат обновления данных с уровнем изоляции SERIALIZABLE и использованием индексов.

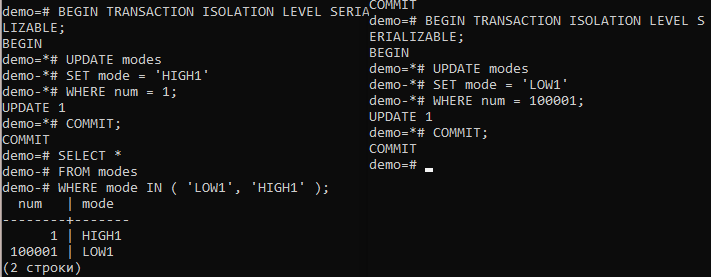


Рисунок 11 – Использование индексов

# 4 Вывод

Был изучен теоретический материал и получены навыки работы с транзакциями в СУБД PostgreSQL.