Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Новосибирский государственный технический университет»

Кафедра вычислительной техники

**Отчет**

По лабораторной работе №2

По дисциплине: «Базы данных»

Транзакции и ограничения в среде PostgreSQL

Вариант 10

Выполнили: Павлюк А.С., Преподаватель: Харюткина С. А.

Казанцев К. О.

Группа: АВТ-008

Факультет: АВТ

Новосибирск, 2022

Оглавление

[Цель работы 3](#_Toc113314340)

[Задание 3](#_Toc113314341)

[Структура базы данных 4](#_Toc113314342)

[Ход работы 4](#_Toc113314343)

[Вывод 8](#_Toc113314344)

[Приложение 9](#_Toc113314345)

Цель работы

Приобрести навыки создания пользователей баз данных, работы с транзакциями в среде PostgreSQL. Изучить основные команды по работе с базой данных. Получение навыков работы с ограничениями.

Задание

Ознакомиться с теоретическими сведениями о возможностях создания пользователей баз данных, использования транзакций в PostgreSQL. Создать нового пользователя и зайти под его именем. Ознакомиться с теоретическими сведениями о возможностях создания ограничений в PostgreSQL. Наложить ограничения согласно своему варианту. Проверить работоспособность ограничений путем добавления в таблицы данных, удовлетворяющих и не удовлетворяющих условиям ограничений. Создать транзакционный блок, в котором производится добавление в таблицы произвольных полей, создать несколько производных таблиц, просмотреть структуру измененных таблиц. Не завершая транзакции параллельно запустить еще одно окно терминала, подключиться к базе и попробовать добавить и удалить записи в таблицы. Сделать откат транзакций, просмотреть структуру таблиц. Просмотреть и проанализировать полученную в результате выполнения операций информацию.

*Вариант 10:* заполнить базу данных туристического агентства (при необходимости). Ограничить заполнение поля «вид тура» следующими 4 наименованиями: автобусный, железнодорожный, авиа. Возраст руководителей туров не должен быть менее 25 лет. Стоимость тура не должна быть отрицательной. Количество участников тура не должно превышать 25 человек и быть менее 10 человек. Продемонстрировать результаты работы.

Структура базы данных

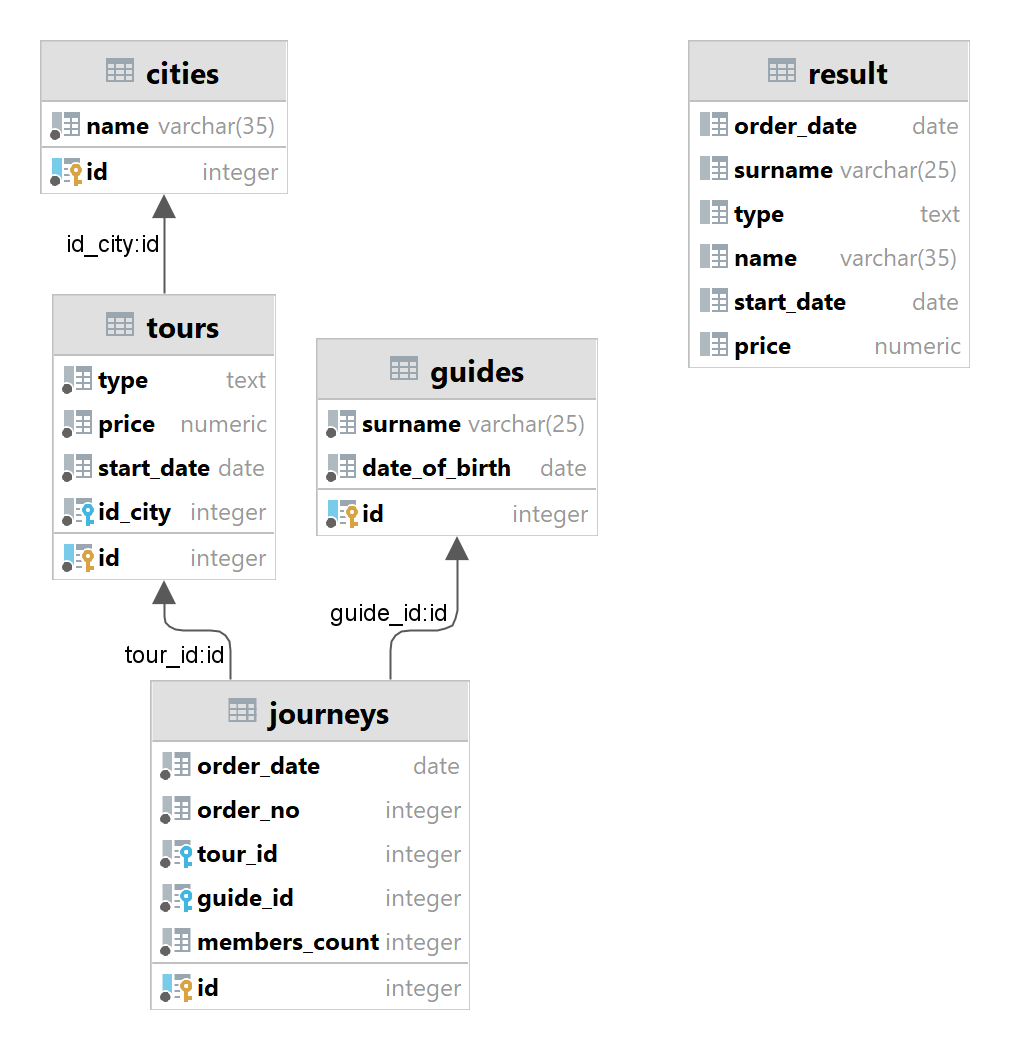
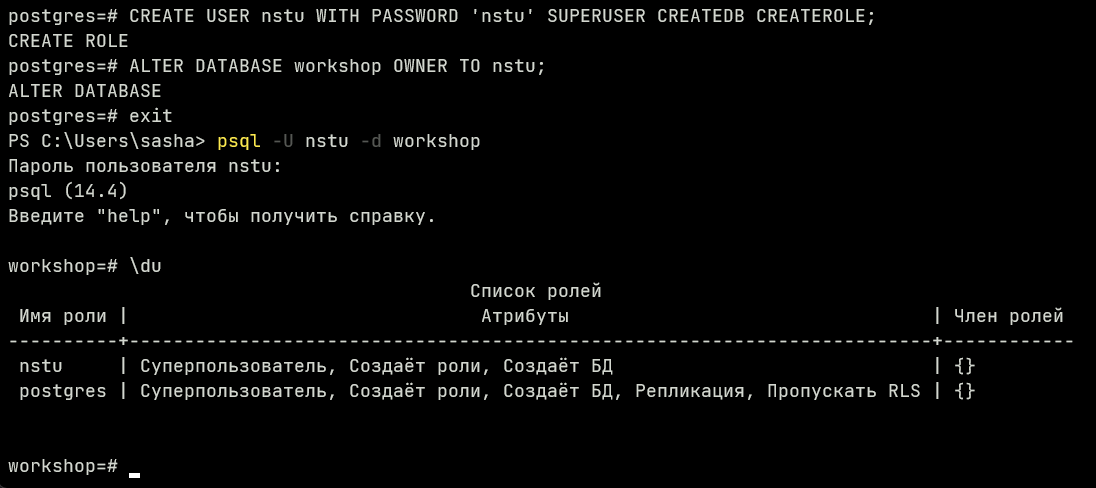


Рисунок 1. Структура базы данных

Ход работы

Создадим суперпользователя **nstu** с возможностью создавать базы данных и других пользователей и передадим ему базу данных **tour\_agency**.



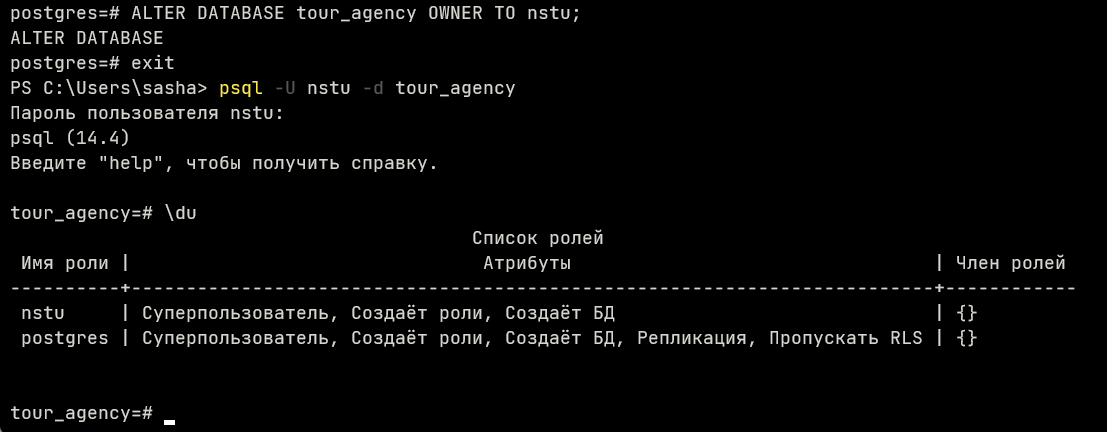


Рисунок 2. Создание новой роли

Далее обновим таблицы, задав ограничения согласно варианту.

**Ограничение** (constraint) представляет собой особый атрибут таблицы, который устанавливает критерии допустимости для содержимого ее полей.

Ограничения задаются в секции CONSTRAINT при создании таблицы командой CREATE TABLE. Они делятся на два типа - ограничения полей и ограничения таблиц.

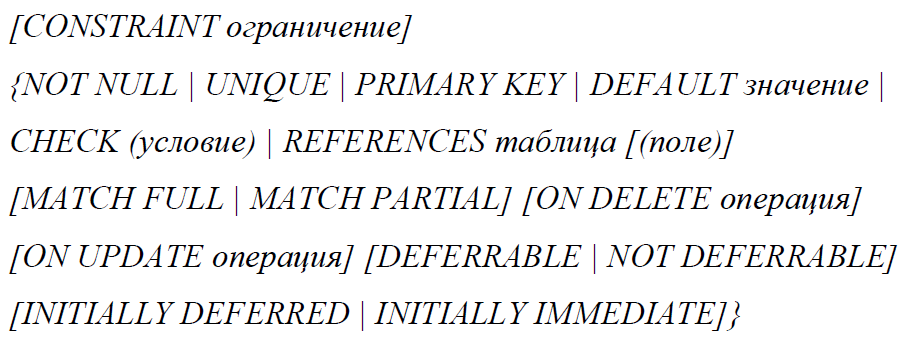


Рисунок 3. Синтаксис ограничений

Добавим ограничение на тип туров.

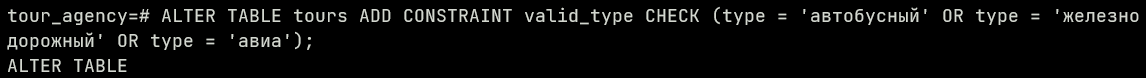


Рисунок 4. Ограничение типов туров

Далее добавим ограничение по возрасту для гидов.

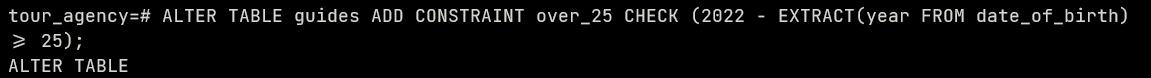


Рисунок 5. Добавление ограничения по возрасту мастеров

Стоимость туров должна быть положительной.



Рисунок 6. Ограничение положительной суммы

Добавим ограничение на количество участников.

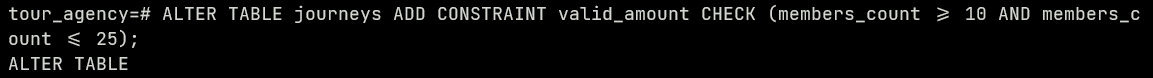


Рисунок 7. Ограничение на число участников

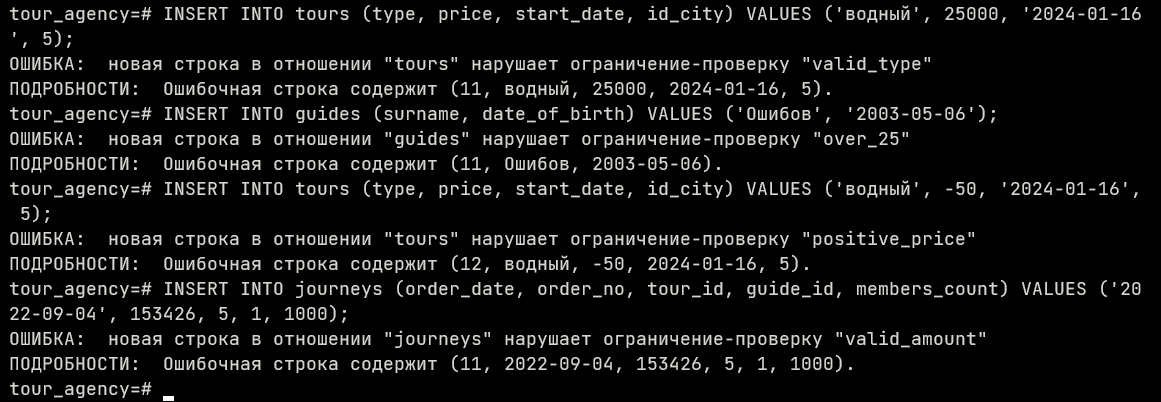


Рисунок 8. Проверка ограничений

Перейдем к транзакциям.

**Транзакции** являются одним из фундаментальных концептов всех СУБД. Сущность транзакции состоит в связывании нескольких шагов в одну операцию по принципу «все или ничего». Внутренние промежуточные состояния между шагами не видны для других конкурирующих транзакций и если во время выполнения транзакции случится ошибка, которая помешает транзакции завершится, то в базе данных никаких изменений сделано не будет.

Транзакция является атомарным действием с точки зрения других транзакций и либо она завершится полностью успешно, либо никакие действия, составляющие транзакцию, выполнены не будут.

В PostgreSQL транзакция - список команд SQL, которые находятся внутри блока, начинающегося командой BEGIN и заканчивающегося командой COMMIT.

Если во время выполнения транзакции мы решаем, что не хотим завершать ее, то мы вместо команды COMMIT вводим команду ROLLBACK и все наши изменения от начала транзакции, будут отменены.

Добавим поле **population** в таблицу **cities**.

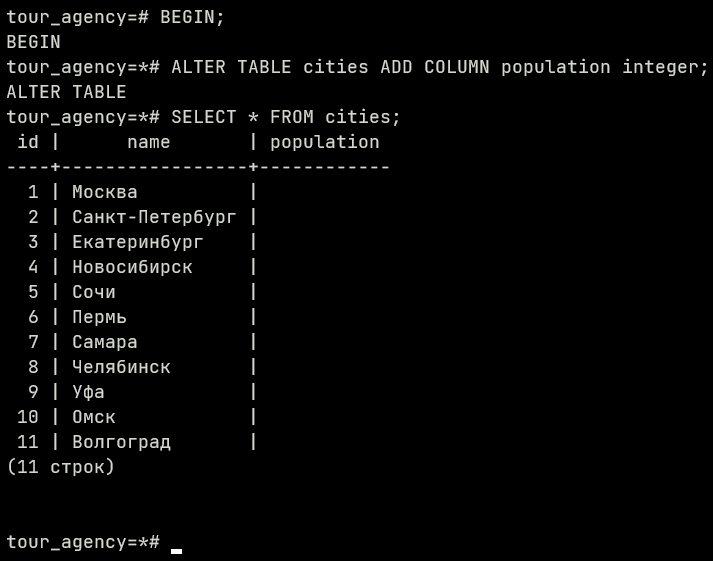


Рисунок 9. Блок транзакции

В другом окне терминала попробуем получить выборку из этой таблицы.

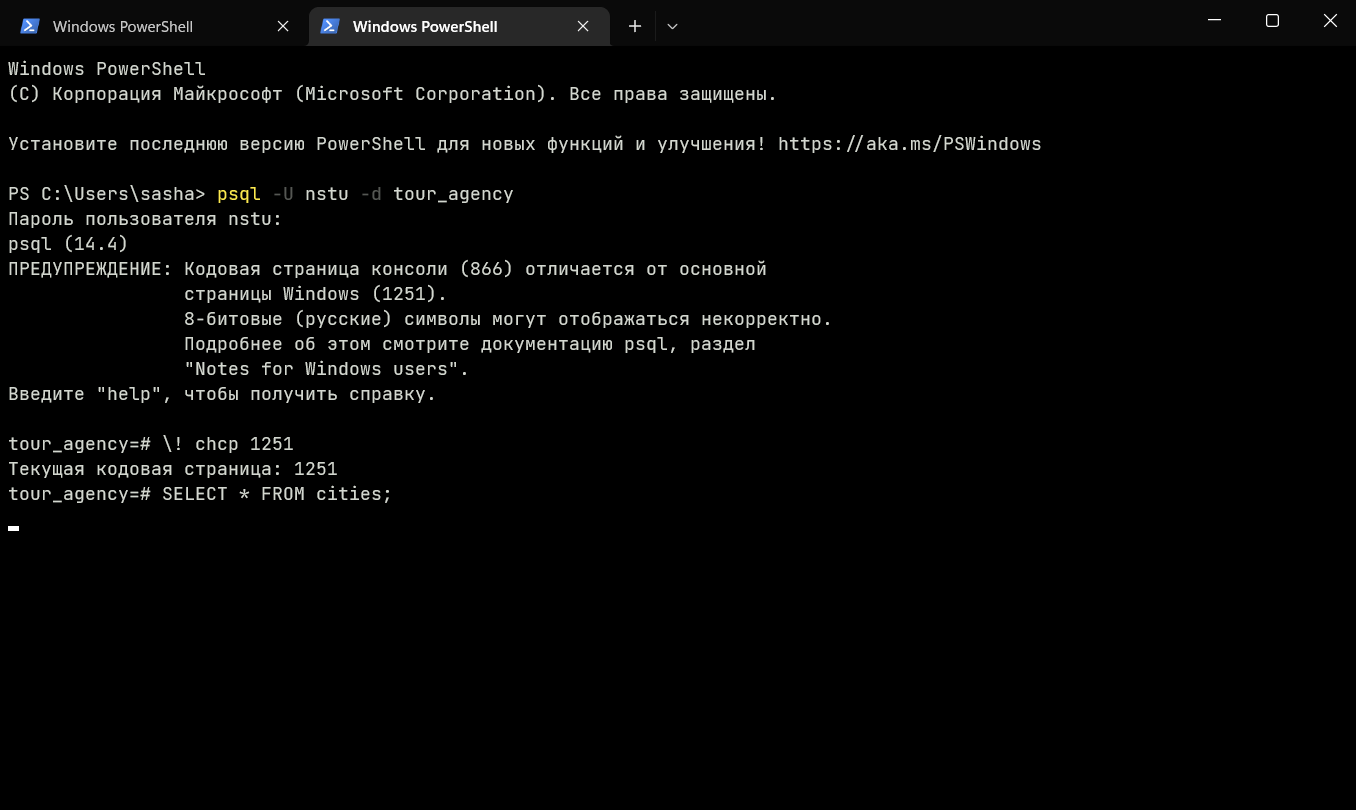


Рисунок 10. Ожидание завершения транзакции

Оно будет ждать завершения или отката транзакции. Все изменения не учтены.

После отката изменений получаем результат.

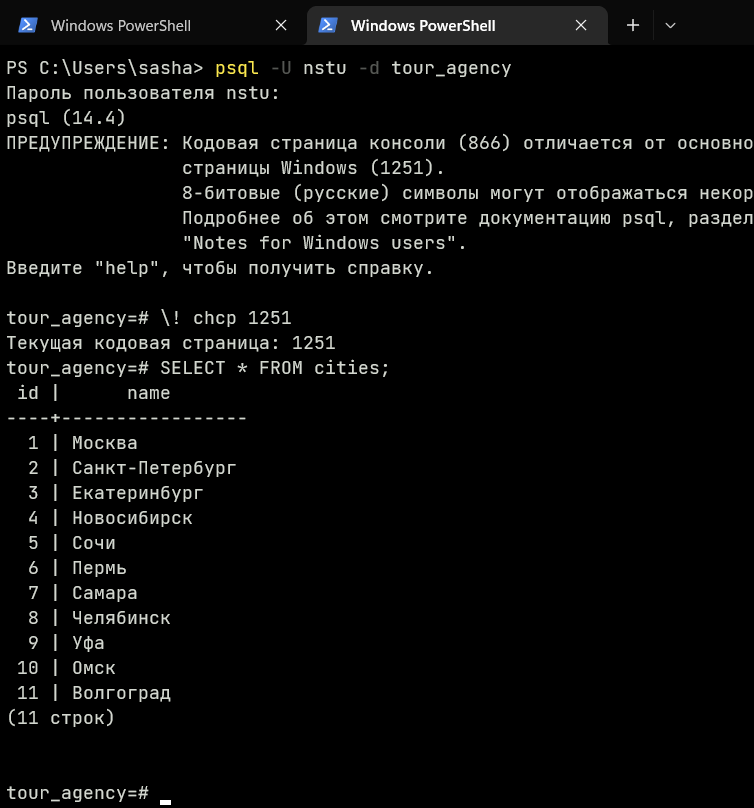


Рисунок 11. Результат после отката транзакции

На рис. 11 видно, что изменения не применились.

Вывод

В результате выполнения лабораторной работы были приобретены навыки создания пользователей баз данных, работы с транзакциями в среде PostgreSQL. Также были получены навыки работы с ограничениями.

Приложение

CREATE USER nstu WITH PASSWORD 'nstu' SUPERUSER CREATEDB CREATEROLE;

ALTER DATABASE workshop OWNER TO nstu;

ALTER TABLE tours ADD CONSTRAINT valid\_type CHECK (type = 'автобусный' OR type = 'железнодорожный' OR type = 'авиа');

ALTER TABLE guides ADD CONSTRAINT over\_25 CHECK (2022 - EXTRACT(year FROM date\_of\_birth) >= 25);

ALTER TABLE tours ADD CONSTRAINT positive\_price CHECK (price > 0);

ALTER TABLE journeys ADD CONSTRAINT valid\_amount CHECK (members\_count >= 10 AND members\_count <= 25);

BEGIN;

ALTER TABLE cities ADD COLUMN population integer;

ROLLBACK;