Министерство науки и высшего образования Российской Федерации

Федеральное государственное автономное образовательное учреждение высшего образования

«САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ

УНИВЕРСИТЕТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ, МЕХАНИКИ И ОПТИКИ»

Факультет среднего профессионального образования

**ОТЧЕТ**

**о Лабораторной работе № 6**

по теме: СОЗДАНИЕ ТАБЛИЦ БАЗЫ ДАННЫХ POSTGRESQL. ЗАПОЛНЕНИЕ ТАБЛИЦ РАБОЧИМИ ДАННЫМИ

по дисциплине: Основы проектирования баз данных

Специальность:

09.02.07 Информационные системы и программирование

|  |  |
| --- | --- |
| Проверил:  Говорова М.М. \_\_\_\_\_\_\_\_\_\_  Дата: «\_\_» \_\_\_\_\_\_\_ 20\_\_г.  Оценка \_\_\_\_\_\_\_\_\_\_\_\_\_\_ | Выполнил:  студент группы Y2336  \_\_\_\_\_\_ Заерко М.В. |

Санкт-Петербург 2019

1. Цель работы

Овладеть практическими навыками создания таблиц базы данных PostgreSQL 10 (11), заполнения их рабочими данными, резервного копирования и восстановления БД.

1. Практическое задание

Практическое задание включает в себя следующие пункты:

1. Создать базу данных с использованием pgAdmin 4 (согласно индивидуальному заданию.
2. Создать схему в составе базы данных.
3. Создать таблицы базы данных.
4. Установить ограничения на данные: Primary Key, Unique, Check, Foreign Key.
5. Заполнить таблицы БД рабочими данными.
6. Создать резервную копию БД.

Указание:

Создать две резервные копии:

* с расширением CUSTOM для восстановления БД;
* с расширением PLAIN для листинга (в отчете);
* при создании резервных копий БД настроить параметры Dump options для Type of objects и Queries;

1. Восстановить БД на другом ПК.
2. Выполнение лабораторной работы
3. Схема базы данных

Схема базы данных представлена на рис. 1.

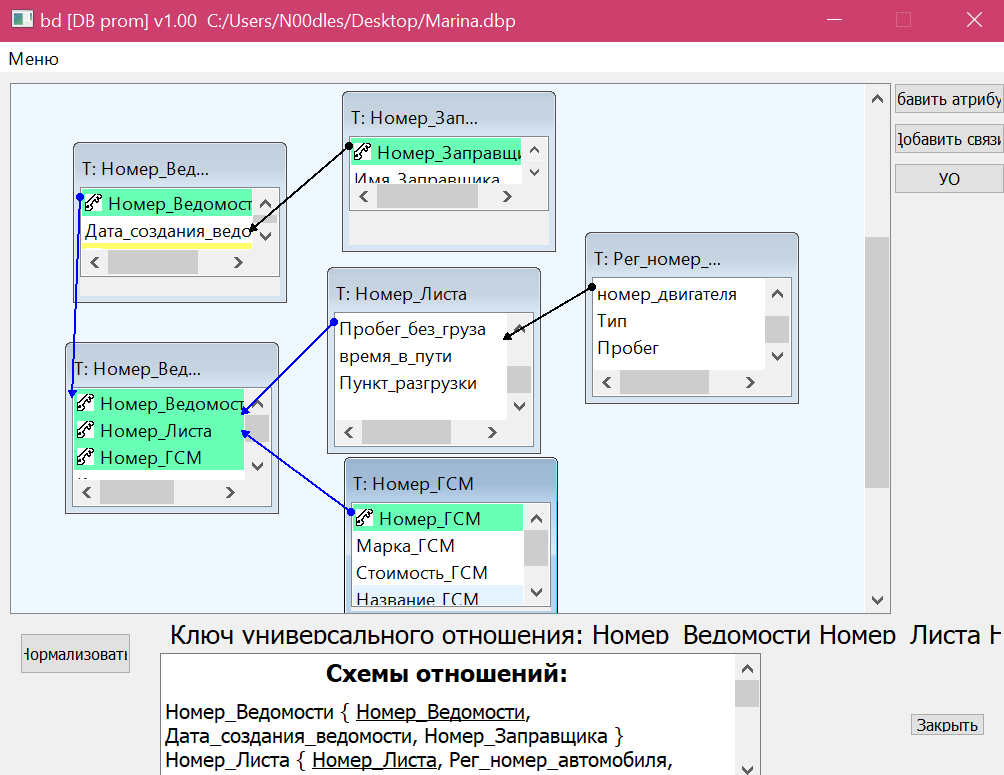


Рисунок 1 — Схема базы данных

1. Скрипт работы с базой данных

Листинг скрипта созданной базы данных представлен в листинге 1:

Листинг 1 – Скрипт работы с базой данных

DROP DATABASE auto\_base;

CREATE DATABASE auto\_base WITH TEMPLATE = template0 ENCODING = 'UTF8' LC\_COLLATE = 'Russian\_Russia.1251' LC\_CTYPE = 'Russian\_Russia.1251';

ALTER DATABASE auto\_base OWNER TO postgres;

ALTER TABLE ONLY public.statement DROP CONSTRAINT statement\_fkey;

ALTER TABLE ONLY public.refuel DROP CONSTRAINT refuel\_n\_statement\_fkey;

ALTER TABLE ONLY public.refuel DROP CONSTRAINT refuel\_n\_list\_fkey;

ALTER TABLE ONLY public.refuel DROP CONSTRAINT refuel\_n\_fuel\_fkey;

ALTER TABLE ONLY public.list DROP CONSTRAINT list\_n\_num\_auto\_fkey;

ALTER TABLE ONLY public.statement DROP CONSTRAINT statement\_pkey;

ALTER TABLE ONLY public.refueler DROP CONSTRAINT refueler\_pkey;

ALTER TABLE ONLY public.list DROP CONSTRAINT list\_pkey;

ALTER TABLE ONLY public.fuel DROP CONSTRAINT fuel\_pkey;

ALTER TABLE ONLY public.auto DROP CONSTRAINT auto\_pkey;

DROP TABLE public.statement;

DROP TABLE public.refueler;

DROP TABLE public.refuel;

DROP TABLE public.list;

DROP TABLE public.fuel;

DROP TABLE public.auto;

CREATE TABLE public.fuel (

n\_fuel integer NOT NULL,

mark\_fuel text NOT NULL,

sum\_fuel double precision NOT NULL,

name\_fuel text NOT NULL

);

ALTER TABLE public.fuel OWNER TO postgres;

CREATE TABLE public.list (

n\_list integer NOT NULL,

dist\_with\_weight double precision NOT NULL,

time\_way double precision NOT NULL,

point\_end text NOT NULL,

n\_num\_auto text NOT NULL

);

CREATE TABLE public.refuel (

n\_list integer NOT NULL,

n\_fuel integer NOT NULL,

n\_statement integer NOT NULL,

sum double precision NOT NULL,

num\_fuel double precision NOT NULL

);

CREATE TABLE public.refueler (

n\_refueler integer NOT NULL,

name\_refueler text NOT NULL

);

CREATE TABLE public.statement (

n\_statement integer NOT NULL,

creation\_date date NOT NULL,

n\_refueler integer NOT NULL

);

ALTER TABLE ONLY public.auto

ADD CONSTRAINT auto\_pkey PRIMARY KEY (n\_reg\_auto);

ALTER TABLE ONLY public.fuel

ADD CONSTRAINT fuel\_pkey PRIMARY KEY (n\_fuel);

ALTER TABLE ONLY public.list

ADD CONSTRAINT list\_pkey PRIMARY KEY (n\_list);

ALTER TABLE ONLY public.refueler

ADD CONSTRAINT refueler\_pkey PRIMARY KEY (n\_refueler);

ALTER TABLE ONLY public.statement

ADD CONSTRAINT statement\_pkey PRIMARY KEY (n\_statement);

ALTER TABLE ONLY public.refuel

ADD CONSTRAINT refuel\_n\_fuel\_fkey FOREIGN KEY (n\_fuel) REFERENCES public.fuel(n\_fuel);

ALTER TABLE ONLY public.refuel

ADD CONSTRAINT refuel\_n\_list\_fkey FOREIGN KEY (n\_list) REFERENCES public.list(n\_list);

ALTER TABLE ONLY public.refuel

ADD CONSTRAINT refuel\_n\_statement\_fkey FOREIGN KEY (n\_statement) REFERENCES public.statement(n\_statement);

ALTER TABLE ONLY public.statement

ADD CONSTRAINT statement\_fkey FOREIGN KEY (n\_refueler) REFERENCES public.refueler(n\_refueler);

INSERT INTO public.auto (n\_reg\_auto, mark, n\_engine, type, distance) VALUES ('м335шк', 'КамАЗ', 'QG16000000P', 'Грузовой', 100000);

INSERT INTO public.auto (n\_reg\_auto, mark, n\_engine, type, distance) VALUES ('л227ка', 'Volvo', 'QR20000000A', 'Грузовой', 57666);

INSERT INTO public.auto (n\_reg\_auto, mark, n\_engine, type, distance) VALUES ('х405ов', 'Газель', '5А62881', 'Легковой', 130.5);

INSERT INTO public.fuel (n\_fuel, mark\_fuel, sum\_fuel, name\_fuel) VALUES (1, 'Газпромнефть', 48, '95');

INSERT INTO public.fuel (n\_fuel, mark\_fuel, sum\_fuel, name\_fuel) VALUES (2, 'Газпромнефть', 42.600000000000001, '92');

INSERT INTO public.fuel (n\_fuel, mark\_fuel, sum\_fuel, name\_fuel) VALUES (3, 'Газпромнефть', 53, 'ДТ');

INSERT INTO public.fuel (n\_fuel, mark\_fuel, sum\_fuel, name\_fuel) VALUES (4, 'Газпромнефть', 44.899999999999999, 'ДТ');

INSERT INTO public.list (n\_list, dist\_with\_weight, time\_way, point\_end, n\_num\_auto) VALUES (1, 100, 25.5, 'Санкт-Петербург', 'х405ов');

INSERT INTO public.list (n\_list, dist\_with\_weight, time\_way, point\_end, n\_num\_auto) VALUES (2, 90000, 72, 'Челябинск', 'м335шк');

INSERT INTO public.list (n\_list, dist\_with\_weight, time\_way, point\_end, n\_num\_auto) VALUES (3, 50000, 35, 'Краснодар', 'л227ка');

INSERT INTO public.refuel (n\_list, n\_fuel, n\_statement, sum, num\_fuel) VALUES (3, 1, 1, 2400, 50);

INSERT INTO public.refuel (n\_list, n\_fuel, n\_statement, sum, num\_fuel) VALUES (2, 3, 2, 3000, 70);

INSERT INTO public.refueler (n\_refueler, name\_refueler) VALUES (1, 'Иванов Владимир Владиславович');

INSERT INTO public.refueler (n\_refueler, name\_refueler) VALUES (2, 'Ланец Антон Борисович');

INSERT INTO public.refueler (n\_refueler, name\_refueler) VALUES (3, 'Мурышев Рустам Олегович');

INSERT INTO public.statement (n\_statement, creation\_date, n\_refueler) VALUES (1, '2019-01-25', 3);

INSERT INTO public.statement (n\_statement, creation\_date, n\_refueler) VALUES (2, '2019-01-25', 2);

INSERT INTO public.statement (n\_statement, creation\_date, n\_refueler) VALUES (3, '2019-01-25', 1);

INSERT INTO public.statement (n\_statement, creation\_date, n\_refueler) VALUES (4, '2019-02-26', 3);

INSERT INTO public.statement (n\_statement, creation\_date, n\_refueler) VALUES (5, '2019-02-27', 3);

1. Вывод

В процессе выполнения данной лабораторной работы были приобретены навыки создания баз данных в PostgreSQL, в том числе создания таблиц базы данных, заполнения ее рабочими данными и создания ее резервных копий.