**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**

**«САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»**

**Факультет инфокоммуникационных технологий**

**Дисциплина:**

«Проектирование и реализация баз данных»

**ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №1**

«**СОЗДАНИЕ БД POSTGRESQL В PGADMIN. РЕЗЕРВНОЕ КОПИРОВАНИЕ И ВОССТАНОВЛЕНИЕ БД**»

**Выполнил:**

студент группы K32392

Бочкарь Артём Артёмович

*\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

(подпись)

**Проверил(а):**

Говорова Марина Михайловна

*\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

(отметка о выполнении)

*\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

(подпись)

Санкт-Петербург

2023 г.

**Цель работы 1.1:** овладеть практическими навыками установки СУБД PostgreSQL и создания базы данных в pgadmin 4.

**Практическое задание 1.1:**

1. Установить СУБД PostgreSQL 1Х.
2. Создать базу данных с использованием pgadmin 4.

**Цель работы 1.2:** овладеть практическими навыками создания таблиц базы данных PostgreSQL 1Х, заполнения их рабочими данными, резервного копирования и восстановления БД.

**Практическое задание 1.2:**

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.

**Выполнение**

**Наименование БД:** hotel

**ERD диаграмма:**

**Изображение выглядит как текст, снимок экрана, диаграмма, Параллельный

Автоматически созданное описание**

Рисунок 1 - ERD диаграмма

**Dump БД:** dump базы данных в двух вариантах, сгенерированный в pgAdmin, приложен к отчету. Файлы: restaurant\_custom\_dump.sql, restaurant\_plain\_dump.sql

**Скрипты работы с БД для создания таблиц:**

CREATE TABLE "lab\_1".employee (

id\_employee BIGINT GENERATED ALWAYS AS IDENTITY PRIMARY KEY,

full\_name VARCHAR(60) NOT NULL,

post VARCHAR(20) NOT NULL,

category VARCHAR(20) NOT NULL,

salary INT NOT NULL CHECK (salary > 0),

number\_of\_passport VARCHAR(11) NOT NULL

);

CREATE TABLE "lab\_1".promotion (

id\_promotion BIGINT GENERATED ALWAYS AS IDENTITY PRIMARY KEY,

holiday VARCHAR(20),

benefit VARCHAR(20) NOT NULL,

season VARCHAR(6)

);

CREATE TABLE "lab\_1".conveniences (

id\_conveniences BIGINT GENERATED ALWAYS AS IDENTITY PRIMARY KEY,

category VARCHAR(40) NOT NULL,

name VARCHAR(40) NOT NULL

);

CREATE TABLE "lab\_1".visitor (

id\_visitor BIGINT GENERATED ALWAYS AS IDENTITY PRIMARY KEY,

name VARCHAR(20) NOT NULL,

surname VARCHAR(20) NOT NULL,

patronymic VARCHAR(20) NOT NULL,

adress VARCHAR(60) NOT NULL,

number\_of\_passport VARCHAR(11) NOT NULL

);

CREATE TABLE "lab\_1".shift (

id\_shift BIGINT GENERATED ALWAYS AS IDENTITY PRIMARY KEY,

start\_time TIMESTAMP NOT NULL,

end\_time TIMESTAMP NOT NULL,

break\_length VARCHAR(1) NOT NULL,

CHECK (start\_time < end\_time)

);

CREATE TABLE "lab\_1".type (

id\_type BIGINT GENERATED ALWAYS AS IDENTITY PRIMARY KEY,

price INT NOT NULL CHECK (price > 0),

number\_of\_seats INT NOT NULL CHECK (number\_of\_seats > 0 AND number\_of\_seats <= 10),

conveniences\_id BIGINT,

title VARCHAR(10) NOT NULL,

FOREIGN KEY(conveniences\_id) REFERENCES "lab\_1".conveniences(id\_conveniences) ON DELETE CASCADE

);

CREATE TABLE "lab\_1".hotel\_room (

id\_hotel\_room BIGINT GENERATED ALWAYS AS IDENTITY PRIMARY KEY,

employment\_status BOOLEAN NOT NULL,

type\_id BIGINT NOT NULL,

cleaning\_status BOOLEAN NOT NULL,

FOREIGN KEY(type\_id) REFERENCES "lab\_1".type(id\_type) ON DELETE CASCADE

);

CREATE TABLE "lab\_1".hotel (

id\_hotel BIGINT GENERATED ALWAYS AS IDENTITY PRIMARY KEY,

name VARCHAR(20) NOT NULL,

adress VARCHAR(60) NOT NULL,

promotion\_id BIGINT NOT NULL,

hotel\_room\_id BIGINT NOT NULL,

employee\_id BIGINT NOT NULL,

UNIQUE (promotion\_id, hotel\_room\_id, employee\_id),

FOREIGN KEY(promotion\_id) REFERENCES "lab\_1".promotion(id\_promotion) ON DELETE CASCADE,

FOREIGN KEY(hotel\_room\_id) REFERENCES "lab\_1".hotel\_room(id\_hotel\_room) ON DELETE CASCADE,

FOREIGN KEY(employee\_id) REFERENCES "lab\_1".employee(id\_employee) ON DELETE CASCADE

);

CREATE TABLE "lab\_1".registration (

date\_of\_registration TIMESTAMP NOT NULL,

date\_of\_departure TIMESTAMP NOT NULL,

hotel\_room\_id BIGINT NOT NULL,

employee\_id BIGINT NOT NULL,

visitor\_id BIGINT NOT NULL,

PRIMARY KEY (hotel\_room\_id, employee\_id, visitor\_id),

FOREIGN KEY(hotel\_room\_id) REFERENCES "lab\_1".hotel\_room(id\_hotel\_room) ON DELETE CASCADE,

FOREIGN KEY(employee\_id) REFERENCES "lab\_1".employee(id\_employee) ON DELETE CASCADE,

FOREIGN KEY(visitor\_id) REFERENCES "lab\_1".visitor(id\_visitor) ON DELETE CASCADE,

reservation\_status BOOLEAN NOT NULL,

lateness\_status BOOLEAN NOT NULL,

CHECK (date\_of\_registration < date\_of\_departure)

);

CREATE TABLE "lab\_1".work\_schedule (

id\_work\_schedule BIGINT GENERATED ALWAYS AS IDENTITY PRIMARY KEY,

shift\_id BIGINT NOT NULL,

employee\_id BIGINT NOT NULL,

FOREIGN KEY(shift\_id) REFERENCES "lab\_1".shift(id\_shift) ON DELETE CASCADE,

FOREIGN KEY(employee\_id) REFERENCES "lab\_1".employee(id\_employee) ON DELETE CASCADE

);

**Скрипты работы с БД для вставки данных:**

INSERT INTO

"lab\_1".employee (full\_name, post, category, salary, number\_of\_passport)

VALUES

(

'Андреев Артем Сергеевич',

'Администратор',

'Первая',

40000,

'4015 535678'

),

(

'Павлов ‘Кирилл Андреевич',

'Администратор',

'Вторая',

50000,

'4012 536587'

),

(

'Задохин Александр Антонович',

'Повар',

'Третья',

80000,

'4013 125689'

),

(

'Уколов Георгий Владимирович',

'Беллбой',

'Первая',

25000,

'4010 908045'

),

(

'Зайцев Олег Андреевич',

'Уборщик',

'Вторая',

20000,

'4016 341223'

);

INSERT INTO

"lab\_1".promotion (holiday, benefit, season)

VALUES

('Новый год', '20%', ''),

('День рождения', '15%', ''),

('', '1%', 'Весна'),

('', '1%', 'Осень'),

('', '5%', 'Зима'),

('', '3%', 'Лето');

INSERT INTO

"lab\_1".conveniences (category, name)

VALUES

('Техника', 'Кондиционер'),

('Техника', 'Телевизор'),

('Техника', 'Микроволновка'),

('Услуга', 'Массаж'),

('Услуга', 'Завтрак в номер');

INSERT INTO

"lab\_1".visitor (name, surname, patronymic, adress, number\_of\_passport)

VALUES

(

'Артём',

'Бочкарь',

'Артёмович',

'Санкт-Петербург, улица Бронная, дом 15/4, кв. 1',

'4013 457890'

),

(

'Лобус',

'Виктория',

'Витальевна',

'Санкт-Петербург, улица Бронная, дом 15/4, кв. 1',

'4013 673412'

),

(

'Комаров',

'Алексей',

'Иванович','Мурманск, проспкт Сергея Приминина, дом 37, кв. 16',

'4011 554421'

),

(

'Исаев',

'Батыр',

'Бактыбекович',

'Москва, улица Советников, дом 12, кв. 567',

'4010 425363'

),

(

'Кислый',

'Иван',

'Петрович',

'Пермь, улица Горская, дом 3, кв. 114',

'4006 571973'

);

INSERT INTO

"lab\_1".shift (start\_time, end\_time, break\_length)

VALUES

('2023-03-13 10:00:00', '2023-03-13 19:00:00', 1),

('2023-03-13 15:00:00', '2023-03-14 10:00:00', 1),

('2023-03-13 20:00:00', '2023-03-14 05:00:00', 1),

('2023-03-14 10:00:00', '2023-03-14 11:00:00', 1);

INSERT INTO

"lab\_1".type (price, number\_of\_seats, conveniences\_id, title)

VALUES

(15000, 8, 21, 'Президент'),

(2500, 2, 22, 'Классик'),

(4000, 3, 23, 'Классик'),

(8000, 4, 24, 'Люкс'),

(10000, 6, 25, 'Люкс');

INSERT INTO

"lab\_1".hotel\_room (employment\_status, type\_id, cleaning\_status)

VALUES

(FALSE, 16, TRUE),

(FALSE, 17, FALSE),

(TRUE, 18, FALSE),

(FALSE, 19, TRUE),

(TRUE, 20, FALSE);

INSERT INTO

"lab\_1".hotel (name, adress, promotion\_id, hotel\_room\_id, employee\_id)

VALUES

('Гранд', 'Москва, малая Садовая, дом 32', 33, 6, 26);

INSERT INTO

"lab\_1".registration (date\_of\_registration, date\_of\_departure, hotel\_room\_id, employee\_id, visitor\_id, reservation\_status, lateness\_status)

VALUES

(

'2023-04-11 11:11:11',

'2023-04-22 22:22:22',

6,

30,

21,

TRUE,

FALSE

),

(

'2023-04-20 13:34:11',

'2023-04-21 15:11:54',

7,

29,

23,

FALSE,

FALSE

),

(

'2023-03-07 18:17:45',

'2023-03-10 11:43:17',

8,

27,

25,

TRUE,

TRUE

),

(

'2023-01-01 11:00:12',

'2023-01-05 10:16:23',

9,

28,

24,

FALSE,

FALSE

),

(

'2023-02-12 14:15:16',

'2023-02-24 13:56:34',

10,

26,

22,

FALSE,

FALSE

);

INSERT INTO

"lab\_1".work\_schedule (shift\_id, employee\_id)

VALUES

(17, 27),

(17, 26),

(18, 30),

(19, 29),

(20, 28);

**Выводы**

В процессе выполнения лабораторной работы удалось более детально ознакомится с работой в pgAdmin 4, получить практические навыки создания таблиц, установки ограничений на таблицы, создания и восстановления резервных копий баз данных.