

**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 К32392

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 1X.
2. Создать базу данных с использованием pgadmin 4.

Цель работы 1.2: овладеть практическими навыками создания таблиц базы данных PostgreSQL 1X, 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 .

7. Восстановить БД.

Выполнение

Наименование БД: hotel

ERD диаграмма:

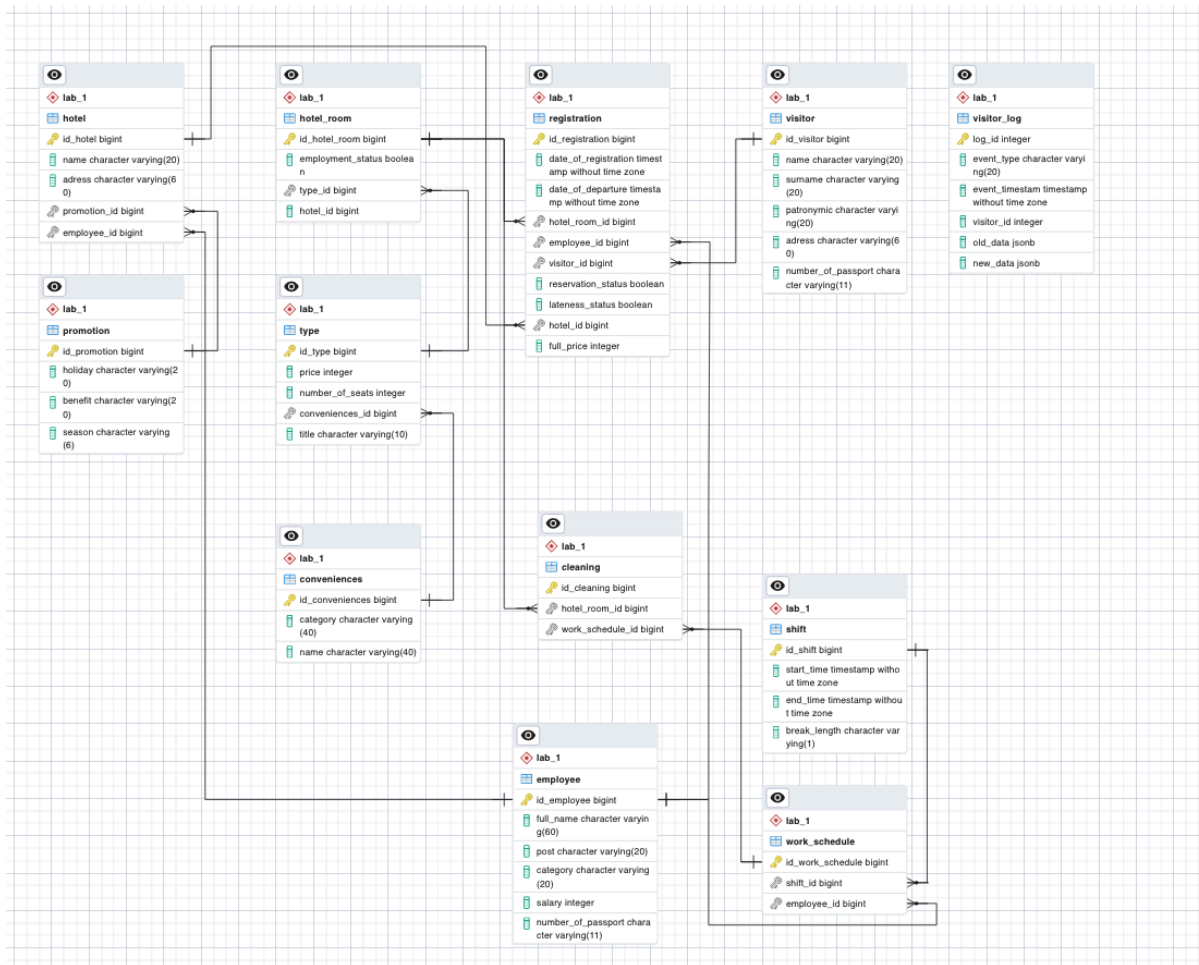


Рисунок 1 - ERD диаграмма

Дамп БД: дампы 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,
    hotel_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,  
    employee_id BIGINT NOT NULL,  
    UNIQUE (promotion_id, employee_id),  
    FOREIGN KEY(promotion_id) REFERENCES "lab_1".promotion(id_promotion) ON  
DELETE CASCADE,  
    FOREIGN KEY(employee_id) REFERENCES "lab_1".employee(id_employee) ON  
DELETE CASCADE  
);
```

```
CREATE TABLE "lab_1".registration (  
    id_registration BIGINT GENERATED ALWAYS AS IDENTITY PRIMARY KEY,  
    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,  
    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".cleaning (  
    id_cleaning BIGINT GENERATED ALWAYS AS IDENTITY PRIMARY KEY,  
    hotel_room_id BIGINT NOT NULL,  
    FOREIGN KEY(hotel_room_id) REFERENCES "lab_1".hotel_room(id_hotel_room) ON  
DELETE CASCADE  
);
```

```
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, проспект 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, 1, 'Президент'),
(2500, 2, 2, 'Классик'),
(4000, 3, 3, 'Классик'),
(8000, 4, 4, 'Люкс'),
(10000, 6, 5, 'Люкс');

INSERT INTO

"lab_1".hotel (name, adress, promotion_id, employee_id)

VALUES

('Гранд', 'Москва, улица малая Садовая, дом 32', 1, 5),
('Леон', 'Санкт-Петербург, улица Достоевская, дом 13', 2, 3),
('Оазис', 'Пермь, улица Петра Великого, дом 17', 3, 4),
('Шишка', 'Москва, улица Сморчкова, дом 100', 4, 1),
('Пальма', 'Дубайск, улица Шейха, дом 1', 5, 2);

INSERT INTO

"lab_1".hotel_room (employment_status, type_id, hotel_id, cleaning_status)

VALUES

(FALSE, 6, 1, TRUE),
(FALSE, 7, 1, FALSE),
(TRUE, 8, 1, FALSE),
(FALSE, 9, 1, TRUE),
(TRUE, 10, 1, FALSE);

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,
 1,
 1,
 TRUE,
 FALSE
),


```
(
    '2023-04-20 13:34:11',
    '2023-04-21 15:11:54',
    7,
    2,
    3,
    FALSE,
    FALSE
),
(
    '2023-03-07 18:17:45',
    '2023-03-10 11:43:17',
    8,
    3,
    5,
    TRUE,
    TRUE
),
(
    '2023-01-01 11:00:12',
    '2023-01-05 10:16:23',
    9,
    4,
    4,
    FALSE,
    FALSE
),
(
    '2023-02-12 14:15:16',
    '2023-02-24 13:56:34',
    10,
    5,
    2,
    FALSE,
    FALSE
);
```

```
INSERT INTO
    "lab_1".work_schedule (shift_id, employee_id)
VALUES
    (1, 1),
    (1, 2),
    (2, 3),
    (4, 5),
    (3, 4);
```

```
INSERT INTO
    "lab_1".cleaning (hotel_room_id, work_schedule_id)
VALUES
    (6, 2),
    (7, 3),
    (8, 4),
    (7, 5),
    (9, 6);
```

Выводы

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