

Министерство науки и высшего образования Российской Федерации
федеральное государственное автономное образовательное учреждение
высшего образования
«Национальный исследовательский университет ИТМО»

Факультет инфокоммуникационных технологий

Лабораторная работа №1
«Создание БД в СУБД PostgreSQL.
Резервное копирование и восстановление
БД»
по дисциплине:
«Проектирование и реализация баз
данных»

Выполнила:
студентка II курса ИКТ
группы К3242
Скокова Алина Викторовна

Проверила:
Говорова Марина Михайловна

Санкт-Петербург
2022

Цель работы: овладение практическими навыками установки СУБД PostgreSQL и создания базы данных в pgadmin 4, создания таблиц базы данных PostgreSQL 12, заполнения их рабочими данными, резервного копирования и восстановления БД.

Практическое задание.

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.
7. Восстановить БД.

Выполнение.

Наименование БД – «telephone_exchange_db».

Схема логической модели базы данных, сгенерированная в Generate ERD, представлена на рисунке 1.

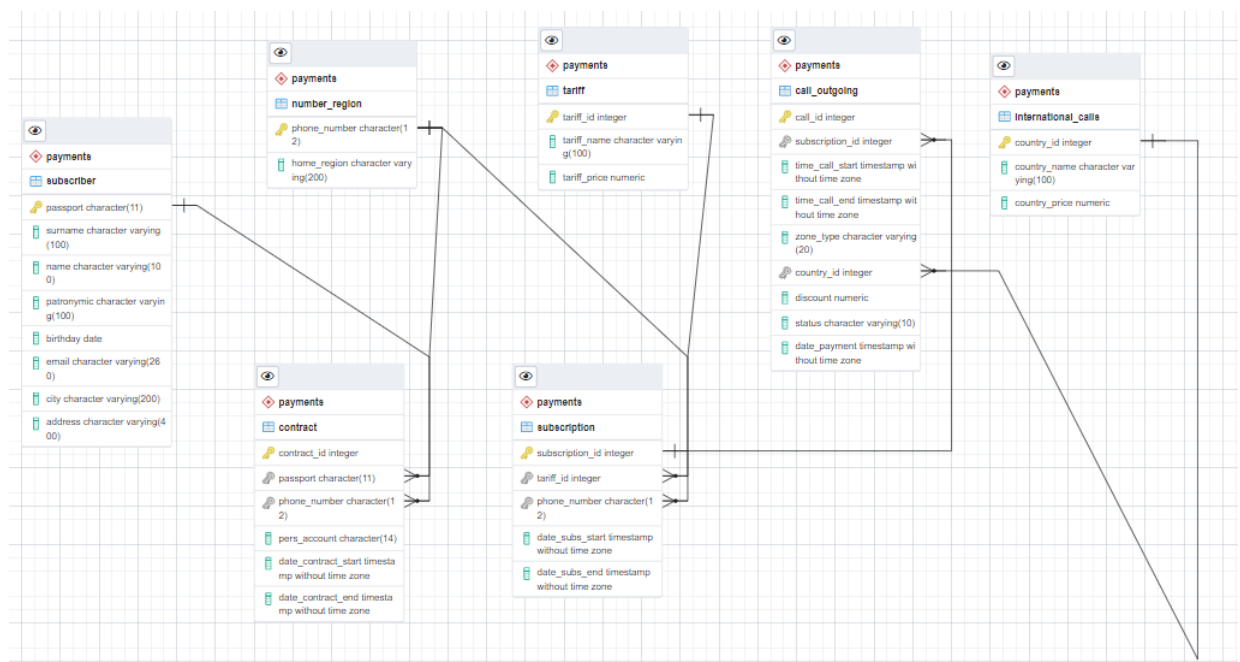


Рисунок 1 – Схема логической модели БД

Скрипт работы с базой данных:

-- создание БД

```
CREATE DATABASE telephone_exchange_db WITH TEMPLATE = template0  
ENCODING = 'UTF8' LC_COLLATE = 'Russian_Russia.1251' LC_CTYPE =  
'Russian_Russia.1251';
```

```
ALTER DATABASE telephone_exchange_db OWNER TO postgres;
```

```
\connect telephone_exchange_db
```

```
SET statement_timeout = 0;  
SET lock_timeout = 0;  
SET idle_in_transaction_session_timeout = 0;  
SET client_encoding = 'UTF8';  
SET standard_conforming_strings = on;  
SELECT pg_catalog.set_config('search_path', '', false);  
SET check_function_bodies = false;  
SET xmloption = content;  
SET client_min_messages = warning;  
SET row_security = off;
```

-- создание схемы payments

```
CREATE SCHEMA payments;
```

```
ALTER SCHEMA payments OWNER TO postgres;
```

```
SET default_tablespace = '';
```

```
SET default_table_access_method = heap;
```

-- создание таблицы.call_outgoing

```
CREATE TABLE payments.call_outgoing (  
    call_id integer NOT NULL,  
    subscription_id integer NOT NULL,  
    time_call_start timestamp without time zone NOT NULL,  
    time_call_end timestamp without time zone NOT NULL,  
    zone_type character varying(20) NOT NULL,  
    country_id integer,  
    discount numeric NOT NULL,  
    status character varying(10) NOT NULL,  
    date_payment timestamp without time zone  
);
```

```
ALTER TABLE payments.call_outgoing OWNER TO postgres;
```

-- автоматическая генерация значений первичного ключа

```
ALTER TABLE payments.call_outgoing ALTER COLUMN call_id ADD  
GENERATED ALWAYS AS IDENTITY (  
    SEQUENCE NAME payments.call_outgoing_call_id_seq  
    START WITH 1  
    INCREMENT BY 1  
    NO MINVALUE  
    NO MAXVALUE  
    CACHE 1  
);
```

-- создание таблицы contract

```
CREATE TABLE payments.contract (  
    contract_id integer NOT NULL,  
    passport character(11) NOT NULL,  
    phone_number character(12) NOT NULL,  
    pers_account character(14) NOT NULL,  
    date_contract_start timestamp without time zone NOT NULL,  
    date_contract_end timestamp without time zone  
);
```

```
ALTER TABLE payments.contract OWNER TO postgres;
```

-- автоматическая генерация значений первичного ключа

```
ALTER TABLE payments.contract ALTER COLUMN contract_id ADD  
GENERATED ALWAYS AS IDENTITY (  
    SEQUENCE NAME payments.contract_contract_id_seq  
    START WITH 1  
    INCREMENT BY 1  
    NO MINVALUE  
    NO MAXVALUE  
    CACHE 1  
);
```

-- создание таблицы international_calls

```
CREATE TABLE payments.international_calls (  
    country_id integer NOT NULL,  
    country_name character varying(100) NOT NULL,  
    country_price numeric NOT NULL  
);
```

```
ALTER TABLE payments.international_calls OWNER TO postgres;
```

-- автоматическая генерация значений первичного ключа

```
ALTER TABLE payments.international_calls ALTER COLUMN country_id  
ADD GENERATED ALWAYS AS IDENTITY (  
    SEQUENCE NAME payments.international_calls_country_id_seq  
    START WITH 1  
    INCREMENT BY 1  
    NO MINVALUE  
    NO MAXVALUE  
    CACHE 1  
);
```

-- создание таблицы number_region

```
CREATE TABLE payments.number_region (  
    phone_number character(12) NOT NULL,  
    home_region character varying(200) NOT NULL  
);
```

```
ALTER TABLE payments.number_region OWNER TO postgres;
```

-- создание таблицы subscriber

```
CREATE TABLE payments.subscriber (  
    passport character(11) NOT NULL,  
    surname character varying(100) NOT NULL,  
    name character varying(100) NOT NULL,  
    patronymic character varying(100),  
    birthday date NOT NULL,  
    email character varying(260) NOT NULL,
```

```
city character varying(200) NOT NULL,  
address character varying(400) NOT NULL  
);
```

```
ALTER TABLE payments.subscriber OWNER TO postgres;
```

-- создание таблицы subscription

```
CREATE TABLE payments.subscription (  
    subscription_id integer NOT NULL,  
    tariff_id integer NOT NULL,  
    phone_number character(12) NOT NULL,  
    date_subs_start timestamp without time zone NOT NULL,  
    date_subs_end timestamp without time zone  
);
```

```
ALTER TABLE payments.subscription OWNER TO postgres;
```

-- автоматическая генерация значений первичного ключа

```
ALTER TABLE payments.subscription ALTER COLUMN subscription_id ADD  
GENERATED ALWAYS AS IDENTITY (  
    SEQUENCE NAME payments.subscription_subscription_id_seq  
    START WITH 1  
    INCREMENT BY 1  
    NO MINVALUE  
    NO MAXVALUE  
    CACHE 1  
);
```

-- создание таблицы tariff

```
CREATE TABLE payments.tariff (  
    tariff_id integer NOT NULL,  
    tariff_name character varying(100) NOT NULL,  
    tariff_price numeric NOT NULL  
);
```

```
ALTER TABLE payments.tariff OWNER TO postgres;
```

-- автоматическая генерация значений первичного ключа

```
ALTER TABLE payments.tariff ALTER COLUMN tariff_id ADD GENERATED  
ALWAYS AS IDENTITY (  
    SEQUENCE NAME payments.tariff_tariff_id_seq  
    START WITH 1  
    INCREMENT BY 1  
    NO MINVALUE  
    NO MAXVALUE  
    CACHE 1  
);
```

-- добавление данных в таблицу call_outgoing

```
INSERT INTO payments.call_outgoing OVERRIDING SYSTEM VALUE  
VALUES (1, 1, '2022-06-03 08:00:00', '2022-06-03 08:10:00', 'город', NULL, 0.0,  
'оплачено', '2022-06-03 08:15:00');
```

```
INSERT INTO payments.call_outgoing OVERRIDING SYSTEM VALUE  
VALUES (2, 4, '2022-06-03 08:00:00', '2022-06-03 08:30:00', 'город', NULL, 0.0,  
'ожидание', NULL);
```

```
INSERT INTO payments.call_outgoing OVERRIDING SYSTEM VALUE  
VALUES (3, 1, '2022-06-05 21:14:00', '2022-06-05 21:17:00', 'город', NULL, 0.2,  
'оплачено', '2022-06-05 21:18:00');
```


INSERT INTO payments.call_outgoing OVERRIDING SYSTEM VALUE
VALUES (5, 7, '2022-06-13 20:00:00', '2022-06-13 21:17:00', 'CHГ', 1, 0.2,
'ожидание', NULL);

INSERT INTO payments.call_outgoing OVERRIDING SYSTEM VALUE
VALUES (6, 7, '2022-06-13 22:00:00', '2022-06-13 22:45:00', 'CHГ', 1, 0.2,
'ожидание', NULL);

INSERT INTO payments.call_outgoing OVERRIDING SYSTEM VALUE
VALUES (7, 13, '2022-06-14 14:00:00', '2022-06-14 14:10:00', 'дальнее
зарубежье', 5, 0.0, 'оплачено', '2022-06-14 14:15:00');

INSERT INTO payments.call_outgoing OVERRIDING SYSTEM VALUE
VALUES (8, 14, '2022-06-14 14:00:00', '2022-06-14 14:19:00', 'город', NULL,
0.0, 'оплачено', '2022-06-14 14:15:00');

INSERT INTO payments.call_outgoing OVERRIDING SYSTEM VALUE
VALUES (9, 4, '2022-06-14 17:54:00', '2022-06-14 18:30:00', 'город', NULL, 0.0,
'оплачено', '2022-06-14 19:00:00');

INSERT INTO payments.call_outgoing OVERRIDING SYSTEM VALUE
VALUES (10, 10, '2022-08-25 19:32:00', '2022-08-25 19:38:00', 'CHГ', 2, 0.2,
'оплачено', '2022-08-25 19:38:00');

INSERT INTO payments.call_outgoing OVERRIDING SYSTEM VALUE
VALUES (11, 10, '2022-08-29 02:00:00', '2022-08-29 02:05:00', 'город', NULL,
0.0, 'ожидание', NULL);

INSERT INTO payments.call_outgoing OVERRIDING SYSTEM VALUE
VALUES (12, 13, '2022-08-29 12:00:00', '2022-08-29 12:25:00', 'дальнее
зарубежье', 5, 0.0, 'оплачено', '2022-08-29 12:25:00');

INSERT INTO payments.call_outgoing OVERRIDING SYSTEM VALUE
VALUES (13, 10, '2022-08-29 15:17:00', '2022-08-29 15:34:00', 'город', NULL,
0.0, 'оплачено', '2022-08-29 15:34:00');

INSERT INTO payments.call_outgoing OVERRIDING SYSTEM VALUE
VALUES (14, 6, '2022-08-23 11:17:00', '2022-09-23 11:20:00', 'город', NULL,
0.0, 'оплачено', '2022-09-23 11:21:00');

```
INSERT INTO payments.call_outgoing OVERRIDING SYSTEM VALUE  
VALUES (15, 1, '2022-08-25 20:00:00', '2022-09-25 20:20:00', 'город', NULL,  
0.2, 'оплачено', '2022-09-25 20:23:00');
```

```
INSERT INTO payments.call_outgoing OVERRIDING SYSTEM VALUE  
VALUES (16, 7, '2022-08-25 20:00:00', '2022-09-25 20:20:00', 'ЧГ', 1, 0.2,  
'ожидание', NULL);
```

-- добавление данных в таблицу contract

```
INSERT INTO payments.contract OVERRIDING SYSTEM VALUE VALUES  
(1, '1111 111111', '+79316538912', '1111 1111 1111', '2001-03-06 00:00:00',  
NULL);
```

```
INSERT INTO payments.contract OVERRIDING SYSTEM VALUE VALUES  
(2, '2232 227422', '+74951473804', '2222 2222 2222', '2020-01-23 00:00:00',  
NULL);
```

```
INSERT INTO payments.contract OVERRIDING SYSTEM VALUE VALUES  
(3, '3668 208482', '+79220711918', '3333 3333 3333', '2020-01-23 15:00:00',  
NULL);
```

```
INSERT INTO payments.contract OVERRIDING SYSTEM VALUE VALUES  
(4, '4900 836477', '+79229001231', '4444 4444 4444', '2020-08-17 00:00:00',  
'2020-12-29 00:00:00');
```

```
INSERT INTO payments.contract OVERRIDING SYSTEM VALUE VALUES  
(5, '4900 836477', '+79223214545', '4444 4444 4444', '2021-01-17 12:45:00',  
NULL);
```

```
INSERT INTO payments.contract OVERRIDING SYSTEM VALUE VALUES  
(6, '5116 343435', '+79319319331', '5555 5555 5555', '2021-01-17 16:30:00',  
NULL);
```

```
INSERT INTO payments.contract OVERRIDING SYSTEM VALUE VALUES  
(7, '6666 648381', '+79315680002', '6666 6666 6666', '2021-03-04 00:00:00',  
NULL);
```

```
INSERT INTO payments.contract OVERRIDING SYSTEM VALUE VALUES  
(8, '7271 000100', '+79812342324', '7777 7777 7777', '2021-03-04 00:00:00',  
'2022-03-04 00:00:00');
```

```
INSERT INTO payments.contract OVERRIDING SYSTEM VALUE VALUES  
(9, '8543 123456', '+79117775456', '8888 8888 8888', '2021-06-16 00:00:00',  
NULL);
```

```
INSERT INTO payments.contract OVERRIDING SYSTEM VALUE VALUES  
(11, '9000 111112', '+79317776688', '9999 9999 9999', '2021-06-17 17:00:00',  
NULL);
```

```
INSERT INTO payments.contract OVERRIDING SYSTEM VALUE VALUES  
(12, '1010 674583', '+79588652495', '1010 1010 1010', '2022-02-23 11:15:00',  
NULL);
```

-- добавление данных в таблицу international_calls

```
INSERT INTO payments.international_calls OVERRIDING SYSTEM VALUE  
VALUES (1, 'Армения', 30.00);
```

```
INSERT INTO payments.international_calls OVERRIDING SYSTEM VALUE  
VALUES (2, 'Казахстан', 30.00);
```

```
INSERT INTO payments.international_calls OVERRIDING SYSTEM VALUE  
VALUES (3, 'Турция', 39.99);
```

```
INSERT INTO payments.international_calls OVERRIDING SYSTEM VALUE  
VALUES (5, 'Германия', 169.99);
```

```
INSERT INTO payments.international_calls OVERRIDING SYSTEM VALUE  
VALUES (4, 'Финляндия', 87.99);
```

-- добавление данных в таблицу number_region

```
INSERT INTO payments.number_region VALUES ('+74951473804',  
'Волгоградская область');
```

```
INSERT INTO payments.number_region VALUES ('+79220711918', 'Москва и  
МО');
```

```

INSERT INTO payments.number_region VALUES ('+79223214545', 'Москва и
МО');
INSERT INTO payments.number_region VALUES ('+79229001231', 'Москва и
МО');
INSERT INTO payments.number_region VALUES ('+79316538912', 'Санкт-
Петербург и ЛО');
INSERT INTO payments.number_region VALUES ('+79319319331', 'Санкт-
Петербург и ЛО');
INSERT INTO payments.number_region VALUES ('+79315680002', 'Санкт-
Петербург и ЛО');
INSERT INTO payments.number_region VALUES ('+79317776688', 'Санкт-
Петербург и ЛО');
INSERT INTO payments.number_region VALUES ('+79310102038', 'Санкт-
Петербург и ЛО');
INSERT INTO payments.number_region VALUES ('+79171111111', 'Санкт-
Петербург и ЛО');
INSERT INTO payments.number_region VALUES ('+79812342324',
'Республика Башкортостан');
INSERT INTO payments.number_region VALUES ('+79128888885',
'Республика Башкортостан');
INSERT INTO payments.number_region VALUES ('+79588652495',
'Свердловская область');
INSERT INTO payments.number_region VALUES ('+79580548521',
'Свердловская область');
INSERT INTO payments.number_region VALUES ('+79117775456',
'Краснодарский край');

```

-- добавление данных в таблицу subscriber

```

INSERT INTO payments.subscriber VALUES ('1111 111111', 'Иванов', 'Иван',
'Иванович', '1976-01-01', 'ivanov@mail.ru', 'Санкт-Петербург', 'ул. Большая, 4');

```

```
INSERT INTO payments.subscriber VALUES ('2232 227422', 'Юдина', 'Лада',  
'Сергеевна', '1991-03-15', 'yulaaad@mail.ru', 'Ногинск', 'пер. Космонавтов, 48');  
INSERT INTO payments.subscriber VALUES ('3668 208482', 'Мишина',  
'Ксения', 'Львовна', '1988-02-10', 'ksumi@gmail.com', 'Ростов', 'пл. Славы, 02');  
INSERT INTO payments.subscriber VALUES ('4900 836477', 'Иванов',  
'Александр', 'Петрович', '1989-08-10', 'ivanovalex@gmail.com', 'Дмитров', 'наб.  
Гоголя, 92');
```

```
INSERT INTO payments.subscriber VALUES ('5116 343435', 'Аксенов',  
'Дмитрий', 'Максимович', '1993-01-26', 'dmitriymak@gmail.com', 'Санкт-  
Петербург', 'наб. Обводного канала, 83');
```

```
INSERT INTO payments.subscriber VALUES ('6666 648381', 'Костин', 'Виль',  
'Викторович', '1960-05-03', 'kostinvilv@mail.ru', 'Санкт-Петербург', 'ул.  
Беринга, 6');
```

```
INSERT INTO payments.subscriber VALUES ('7271 000100', 'Шилова',  
'Анастасия', 'Борисовна', '1998-11-30', 'nastyanastya@mail.ru', 'Егорьевск', 'наб.  
Ломоносова, 86');
```

```
INSERT INTO payments.subscriber VALUES ('8543 123456', 'Кондратьева',  
'Елена-Мария', 'Сергеевна', '1998-01-31', 'kondrat@mail.ru', 'Москва', 'ул.  
Широкая, 14');
```

```
INSERT INTO payments.subscriber VALUES ('9000 111112', 'Самойлов',  
'Нестор', 'Алексеевич', '2001-04-17', 'nestorrr01@gmail.com', 'Наро-Фоминск',  
'пер. Ладыгина, 43');
```

```
INSERT INTO payments.subscriber VALUES ('1010 674583', 'Мясникова',  
'Дарья', 'Олеговна', '1989-12-22', 'mas22dar@gmail.com', 'Луховицы', 'шоссе  
Космонавтов, 61');
```

-- добавление данных в таблицу subscription

```
INSERT INTO payments.subscription OVERRIDING SYSTEM VALUE  
VALUES (1, 2, '+79316538912', '2001-03-06 02:00:00', NULL);
```

```

INSERT INTO payments.subscription OVERRIDING SYSTEM VALUE
VALUES (3, 1, '+74951473804', '2001-01-23 00:00:00', NULL);
INSERT INTO payments.subscription OVERRIDING SYSTEM VALUE
VALUES (4, 1, '+79220711918', '2001-01-23 15:15:00', NULL);
INSERT INTO payments.subscription OVERRIDING SYSTEM VALUE
VALUES (5, 4, '+79229001231', '2020-08-17 00:00:00', '2020-12-28 00:00:00');
INSERT INTO payments.subscription OVERRIDING SYSTEM VALUE
VALUES (6, 4, '+79223214545', '2021-01-20 00:00:00', NULL);
INSERT INTO payments.subscription OVERRIDING SYSTEM VALUE
VALUES (7, 4, '+79319319331', '2021-01-18 00:00:00', NULL);
INSERT INTO payments.subscription OVERRIDING SYSTEM VALUE
VALUES (8, 3, '+79315680002', '2021-03-04 00:00:00', NULL);
INSERT INTO payments.subscription OVERRIDING SYSTEM VALUE
VALUES (10, 3, '+79812342324', '2021-03-04 00:00:00', '2021-10-01 00:00:00');
INSERT INTO payments.subscription OVERRIDING SYSTEM VALUE
VALUES (11, 5, '+79812342324', '2021-10-01 00:00:00', '2022-03-04 00:00:00');
INSERT INTO payments.subscription OVERRIDING SYSTEM VALUE
VALUES (12, 5, '+79117775456', '2021-06-28 00:00:00', NULL);
INSERT INTO payments.subscription OVERRIDING SYSTEM VALUE
VALUES (13, 1, '+79317776688', '2021-06-17 17:02:00', NULL);
INSERT INTO payments.subscription OVERRIDING SYSTEM VALUE
VALUES (14, 1, '+79588652495', '2022-02-23 11:15:00', NULL);

```

-- добавление данных в таблицу tariff

```

INSERT INTO payments.tariff OVERRIDING SYSTEM VALUE VALUES (1,
'Все включено', 29.99);
INSERT INTO payments.tariff OVERRIDING SYSTEM VALUE VALUES (2,
'Без переплат', 9.99);
INSERT INTO payments.tariff OVERRIDING SYSTEM VALUE VALUES (3,
'Средненький', 19.99);

```

```
INSERT INTO payments.tariff OVERRIDING SYSTEM VALUE VALUES (4,  
'Для щедрых', 119.99);
```

```
INSERT INTO payments.tariff OVERRIDING SYSTEM VALUE VALUES (5,  
'Семейный', 17.99);
```

-- ограничение на первичный ключ

```
ALTER TABLE ONLY payments.call_outgoing  
    ADD CONSTRAINT call_outgoing_pkey PRIMARY KEY (call_id);
```

-- ограничение на возраст

```
ALTER TABLE payments.subscriber  
    ADD CONSTRAINT ch_birthday CHECK ((date_part('year'::text,  
age((birthday)::timestamp with time zone)) >= (18)::double precision)) NOT  
VALID;
```

-- ограничение на список стран

```
ALTER TABLE payments.international_calls  
    ADD CONSTRAINT ch_country_name CHECK (((country_name)::text = ANY  
((ARRAY['Армения'::character varying, 'Казахстан'::character varying,  
'Турция'::character varying, 'Финляндия'::character varying,  
'Германия'::character varying])::text[]))) NOT VALID;
```

-- проверка последовательности дат

```
ALTER TABLE payments.contract  
    ADD CONSTRAINT ch_date_contract_end CHECK ((date_contract_end >  
date_contract_start)) NOT VALID;
```

-- проверка последовательности дат

```
ALTER TABLE payments.subscription
```

```
ADD CONSTRAINT ch_date_subs_end CHECK ((date_subs_end >
date_subs_start)) NOT VALID;
```

-- ограничение на возможные значения скидки

```
ALTER TABLE payments.call_outgoing
```

```
ADD CONSTRAINT ch_discount CHECK ((discount = ANY (ARRAY[0.0,
0.2]))) NOT VALID;
```

-- ограничение на формат почты

```
ALTER TABLE payments.subscriber
```

```
ADD CONSTRAINT ch_email CHECK (((email)::text ~* '.*@.*':text)) NOT
VALID;
```

-- ограничение на формат имени

```
ALTER TABLE payments.subscriber
```

```
ADD CONSTRAINT ch_name CHECK (((name)::text ~* '^[a-яА-Я\s-
]+$':text)) NOT VALID;
```

-- ограничение на формат паспортных данных

```
ALTER TABLE payments.subscriber
```

```
ADD CONSTRAINT ch_passport CHECK ((passport ~* '^[0-9]{4}[\s][0-
9]{6}$':text)) NOT VALID;
```

-- ограничение на формат отчества

```
ALTER TABLE payments.subscriber
```

```
ADD CONSTRAINT ch_patronymic CHECK (((patronymic)::text ~* '^[a-яА-
Я\s-]+$':text)) NOT VALID;
```


-- ограничение на формат лицевого счета

```
ALTER TABLE payments.contract
```

```
ADD CONSTRAINT ch_pers_account CHECK ((pers_account ~* '^[0-9]{4}[\s][0-9]{4}[\s][0-9]{4}$'::text)) NOT VALID;
```

-- ограничение на формат номера телефона

```
ALTER TABLE payments.number_region
```

```
ADD CONSTRAINT ch_phone_number CHECK ((phone_number ~* '^[+][7][0-9]{10}$'::text)) NOT VALID;
```

-- ограничение на список значений домашнего региона

```
ALTER TABLE payments.number_region
```

```
ADD CONSTRAINT ch_region CHECK (((home_region)::text = ANY ((ARRAY['Санкт-Петербург и ЛО'::character varying, 'Волгоградская область'::character varying, 'Краснодарский край'::character varying, 'Москва и МО'::character varying, 'Республика Башкортостан'::character varying, 'Свердловская область'::character varying]))::text[[]])) NOT VALID;
```

-- ограничение на список значений статуса оплаты

```
ALTER TABLE payments.call_outgoing
```

```
ADD CONSTRAINT ch_status CHECK (((status)::text = ANY ((ARRAY['ожидание'::character varying, 'оплачено'::character varying]))::text[[]])) NOT VALID;
```

-- ограничение на формат фамилии

```
ALTER TABLE payments.subscriber
```

```
ADD CONSTRAINT ch_surname CHECK (((surname)::text ~* '^[a-яА-Я\s-]+$',::text)) NOT VALID;
```

-- ограничение на порядок времени начала и окончания звонка

```
ALTER TABLE payments.call_outgoing  
    ADD CONSTRAINT ch_time_call_end CHECK ((time_call_end >=  
time_call_start)) NOT VALID;
```

-- ограничение на список значений зоны

```
ALTER TABLE payments.call_outgoing  
    ADD CONSTRAINT ch_zone_type CHECK (((zone_type)::text = ANY  
((ARRAY['город'::character varying, 'республика'::character varying,  
'CHГ'::character varying, 'дальнее зарубежье'::character varying]))::text[])) NOT  
VALID;
```

-- ограничение на первичный ключ

```
ALTER TABLE ONLY payments.contract  
    ADD CONSTRAINT contract_pkey PRIMARY KEY (contract_id);
```

-- ограничение на первичный ключ

```
ALTER TABLE ONLY payments.international_calls  
    ADD CONSTRAINT international_calls_pkey PRIMARY KEY (country_id);
```

-- ограничение на первичный ключ

```
ALTER TABLE ONLY payments.number_region  
    ADD CONSTRAINT number_region_pkey PRIMARY KEY (phone_number);
```

-- ограничение на первичный ключ

```
ALTER TABLE ONLY payments.subscriber  
    ADD CONSTRAINT subscriber_pkey PRIMARY KEY (passport);
```

-- ограничение на первичный ключ

```
ALTER TABLE ONLY payments.subscription
```

```
ADD CONSTRAINT subscription_pkey PRIMARY KEY (subscription_id);
```

-- ограничение на первичный ключ

```
ALTER TABLE ONLY payments.tariff
```

```
ADD CONSTRAINT tariff_pkey PRIMARY KEY (tariff_id);
```

-- ограничение на уникальность почты

```
ALTER TABLE ONLY payments.subscriber
```

```
ADD CONSTRAINT u_email UNIQUE (email);
```

-- ограничение на уникальность составного первичного ключа (заменен на суррогатный)

```
ALTER TABLE ONLY payments.contract
```

```
ADD CONSTRAINT u_pass_tel_date UNIQUE (passport, phone_number,  
date_contract_start);
```

-- ограничение на уникальность составного первичного ключа (заменен на суррогатный)

```
ALTER TABLE ONLY payments.call_outgoing
```

```
ADD CONSTRAINT u_subs_time UNIQUE (subscription_id, time_call_start);
```

-- ограничение на уникальность составного первичного ключа (заменен на суррогатный)

```
ALTER TABLE ONLY payments.subscription
```

```
ADD CONSTRAINT u_tariff_phone_date UNIQUE (tariff_id, phone_number,  
date_subs_start);
```

-- ограничение на внешний ключ

```
ALTER TABLE ONLY payments.call_outgoing  
    ADD CONSTRAINT fk_country_id FOREIGN KEY (country_id)  
REFERENCES payments.international_calls(country_id) NOT VALID;
```

-- ограничение на внешний ключ

```
ALTER TABLE ONLY payments.contract  
    ADD CONSTRAINT fk_passport FOREIGN KEY (passport) REFERENCES  
payments.subscriber(passport) NOT VALID;
```

-- ограничение на внешний ключ

```
ALTER TABLE ONLY payments.contract  
    ADD CONSTRAINT fk_phone_number FOREIGN KEY (phone_number)  
REFERENCES payments.number_region(phone_number) NOT VALID;
```

-- ограничение на внешний ключ

```
ALTER TABLE ONLY payments.subscription  
    ADD CONSTRAINT fk_phone_number FOREIGN KEY (phone_number)  
REFERENCES payments.number_region(phone_number) NOT VALID;
```

-- ограничение на внешний ключ

```
ALTER TABLE ONLY payments.call_outgoing  
    ADD CONSTRAINT fk_subscription_id FOREIGN KEY (subscription_id)  
REFERENCES payments.subscription(subscription_id);
```

-- ограничение на внешний ключ

```
ALTER TABLE ONLY payments.subscription  
    ADD CONSTRAINT fk_tariff_id FOREIGN KEY (tariff_id) REFERENCES  
payments.tariff(tariff_id);
```

Выводы.

Были получены практические навыки установки СУБД PostgreSQL и создания базы данных в pgadmin 4, создания схемы в составе базы данных, создания таблиц базы данных PostgreSQL 12 с установлением ограничений на данные, заполнения таблиц рабочими данными, резервного копирования и восстановления базы данных.