Министерство науки и высшего образования Российской Федерации

федеральное государственное автономное образовательное учреждение
высшего образования
«Национальный исследовательский университет ИТМО»

Факультет инфокоммуникационных технологий

Лабораторная работа №1.2

«Создание таблиц базы данных POSTGRESQL. Заполнение таблиц рабочими данными»

по дисциплине:
«Проектирование и реализация баз данных»

Выполнил:
студентка 2 курса ИКТ
группы К3242
Ф.И.О.: Быкова С.М.

Проверил:
Говорова Марина Михайловна

Санкт-Петербург
2022

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

Оборудование: компьютерный класс.

Программное обеспечение: СУБД PostgreSQL 1X, pgAdmin 4.

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.

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.

Ход работы:

Предметная область «Книжное издательство». Наименование БД «Books»

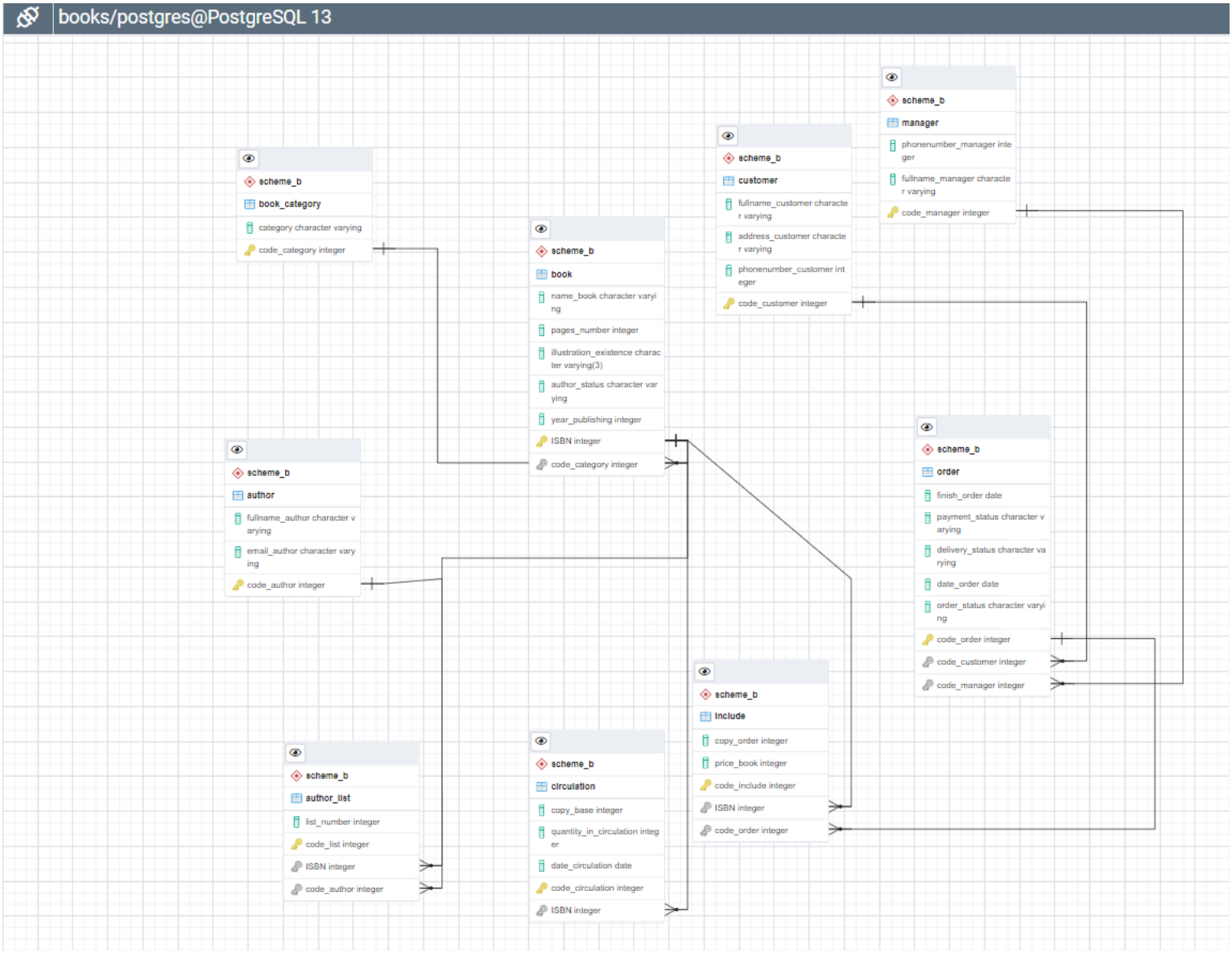

Рисунок 1 – Логическая схема модели базы данных, сгенерированная

SQL-скрипт с процессом создания базы данных:

--создание БД

```
CREATE DATABASE books WITH TEMPLATE = template0 ENCODING
    = 'UTF8' LOCALE = 'Russian_Russia.1251';

ALTER DATABASE books OWNER TO postgres;
```

```
\connect books


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 scheme_b

```
CREATE SCHEMA scheme_b;

ALTER SCHEMA scheme_b OWNER TO postgres;

SET default_tablespace = '';

SET default_table_access_method = heap;
```

-- создание таблицы со списками авторов

```
CREATE TABLE scheme_b.author_list (
    list_number integer NOT NULL,
    code_list integer NOT NULL,
    "ISBN" integer NOT NULL,
    code_author integer NOT NULL
);
```

ALTER TABLE scheme_b.author_list OWNER TO postgres;

-- создание внешних ключей

CREATE SEQUENCE scheme_b."author_list_ISBN_seq"

AS integer

START WITH 1

INCREMENT BY 1

NO MINVALUE

NO MAXVALUE

CACHE 1;

ALTER TABLE scheme_b."author_list_ISBN_seq" OWNER TO postgres;

ALTER SEQUENCE scheme_b."author_list_ISBN_seq" OWNED BY scheme_b.author_list."ISBN";

CREATE SEQUENCE scheme_b.author_list_code_author_seq

AS integer

START WITH 1

INCREMENT BY 1

NO MINVALUE

NO MAXVALUE

CACHE 1;

ALTER TABLE scheme_b.author_list_code_author_seq OWNER TO postgres;

ALTER SEQUENCE scheme_b.author_list_code_author_seq OWNED BY scheme_b.author_list.code_author;

--создание первичного ключа

CREATE SEQUENCE scheme_b.author_list_code_list_seq

```
    AS integer
    START WITH 1
    INCREMENT BY 1
    NO MINVALUE
    NO MAXVALUE
    CACHE 1;
ALTER TABLE scheme_b.author_list_code_list_seq OWNER TO postgres;
ALTER SEQUENCE scheme_b.author_list_code_list_seq OWNED BY scheme_b.author_list.code_list;
```

--создание таблицы, содержащей информацию о самой книге, и соответствующих первичного и внешних ключей

```
CREATE TABLE scheme_b.book (
    name_book character varying NOT NULL,
    pages_number integer NOT NULL,
    illustration_existence character varying NOT NULL,
    author_status character varying NOT NULL,
    year_publishing integer NOT NULL,
    "ISBN" integer NOT NULL,
    code_category integer NOT NULL
);
ALTER TABLE scheme_b.book OWNER TO postgres;
CREATE SEQUENCE scheme_b."book_ISBN_seq"
    AS integer
    START WITH 1
    INCREMENT BY 1
```

```
    NO MINVALUE
    NO MAXVALUE
    CACHE 1;
ALTER TABLE scheme_b."book_ISBN_seq" OWNER TO postgres;
ALTER SEQUENCE scheme_b."book_ISBN_seq" OWNED BY scheme_b.book."ISBN";
```

--создание таблицы с данными об авторах

```
CREATE TABLE scheme_b.author (
    fullname_author character varying NOT NULL,
    email_author character varying,
    code_author integer NOT NULL
);
ALTER TABLE scheme_b.author OWNER TO postgres;
CREATE SEQUENCE scheme_b.author_code_author_seq
    AS integer
    START WITH 1
    INCREMENT BY 1
    NO MINVALUE
    NO MAXVALUE
    CACHE 1;

ALTER TABLE scheme_b.author_code_author_seq OWNER TO postgres;
ALTER SEQUENCE scheme_b.author_code_author_seq OWNED BY scheme_b.author.code_author;
```

-- создание таблицы, содержащей информацию о возможных категориях книг

```
CREATE TABLE scheme_b.book_category (
    category character varying NOT NULL,
    code_category integer NOT NULL
);
ALTER TABLE scheme_b.book_category OWNER TO postgres;
CREATE SEQUENCE scheme_b.book_category_code_category_seq
    AS integer
    START WITH 1
    INCREMENT BY 1
    NO MINVALUE
    NO MAXVALUE
    CACHE 1;
ALTER TABLE scheme_b.book_category_code_category_seq OWNER TO postgres;
ALTER SEQUENCE scheme_b.book_category_code_category_seq OWNED BY scheme_b.book_category.code_category;
CREATE SEQUENCE scheme_b.book_code_category_seq
    AS integer
    START WITH 1
    INCREMENT BY 1
    NO MINVALUE
    NO MAXVALUE
    CACHE 1;
ALTER TABLE scheme_b.book_code_category_seq OWNER TO postgres;
ALTER SEQUENCE scheme_b.book_code_category_seq OWNED BY scheme_b.book.code_category;
```

-- создание таблицы с данными о тираже

```
CREATE TABLE scheme_b.circulation (
    copy_base integer,
    quantity_in_circulation integer NOT NULL,
    date_circulation date NOT NULL,
    code_circulation integer NOT NULL,
    "ISBN" integer NOT NULL
);


ALTER TABLE scheme_b.circulation OWNER TO postgres;
CREATE SEQUENCE scheme_b."circulation_ISBN_seq"
    AS integer
    START WITH 1
    INCREMENT BY 1
    NO MINVALUE
    NO MAXVALUE
    CACHE 1;
ALTER TABLE scheme_b."circulation_ISBN_seq" OWNER TO postgres;
ALTER SEQUENCE scheme_b."circulation_ISBN_seq" OWNED BY scheme_b.circulation."ISBN";


CREATE SEQUENCE scheme_b.circulation_code_circulation_seq
    AS integer
    START WITH 1
    INCREMENT BY 1
```

```
    NO MINVALUE
    NO MAXVALUE
    CACHE 1;
ALTER TABLE scheme_b.circulation_code_circulation_seq OWNER TO postgres;
ALTER SEQUENCE scheme_b.circulation_code_circulation_seq OWNED BY scheme_b.circulation.code_circulation;
-- создание таблицы с данными о клиенте
CREATE TABLE scheme_b.customer (
    fullname_customer character varying NOT NULL,
    address_customer character varying NOT NULL,
    phonenumber_customer bigint NOT NULL,
    code_customer integer NOT NULL
);
ALTER TABLE scheme_b.customer OWNER TO postgres;
CREATE SEQUENCE scheme_b.customer_code_customer_seq
    AS integer
    START WITH 1
    INCREMENT BY 1
    NO MINVALUE
    NO MAXVALUE
    CACHE 1;
ALTER TABLE scheme_b.customer_code_customer_seq OWNER TO postgres;
```

```
ALTER SEQUENCE scheme_b.customer_code_customer_seq OWNED BY scheme_b.customer.code_customer;

-- создание таблицы, связывающей заказ и клиента

CREATE TABLE scheme_b.include (
    copy_order integer NOT NULL,
    price_book integer NOT NULL,
    code_include integer NOT NULL,
    "ISBN" integer NOT NULL,
    code_order integer NOT NULL
);


ALTER TABLE scheme_b.include OWNER TO postgres;


CREATE SEQUENCE scheme_b."include_ISBN_seq"
    AS integer
    START WITH 1
    INCREMENT BY 1
    NO MINVALUE
    NO MAXVALUE
    CACHE 1;


ALTER TABLE scheme_b."include_ISBN_seq" OWNER TO postgres;

ALTER SEQUENCE scheme_b."include_ISBN_seq" OWNED BY scheme_b.include."ISBN";
```

```
CREATE SEQUENCE scheme_b.include_code_include_seq
    AS integer
    START WITH 1
    INCREMENT BY 1
    NO MINVALUE
    NO MAXVALUE
    CACHE 1;


ALTER TABLE scheme_b.include_code_include_seq OWNER TO postgres;

ALTER SEQUENCE scheme_b.include_code_include_seq OWNED BY scheme_b.include.code_include;

CREATE SEQUENCE scheme_b.include_code_order_seq
    AS integer
    START WITH 1
    INCREMENT BY 1
    NO MINVALUE
    NO MAXVALUE
    CACHE 1;

ALTER TABLE scheme_b.include_code_order_seq OWNER TO postgres;

ALTER SEQUENCE scheme_b.include_code_order_seq OWNED BY scheme_b.include.code_order;
```

--создание таблицы с данными о менеджерах

```
CREATE TABLE scheme_b.manager (
    phonenumber_manager bigint NOT NULL,
    fullname_manager character varying NOT NULL,
```

```
    code_manager integer NOT NULL
);


ALTER TABLE scheme_b.manager OWNER TO postgres;
CREATE SEQUENCE scheme_b.manager_code_manager_seq
    AS integer
    START WITH 1
    INCREMENT BY 1
    NO MINVALUE
    NO MAXVALUE
    CACHE 1;


ALTER TABLE scheme_b.manager_code_manager_seq OWNER TO postgres;
ALTER SEQUENCE scheme_b.manager_code_manager_seq OWNED BY scheme_b.manager.code_manager;
-- создание таблицы с данными о заказе
CREATE TABLE scheme_b."order" (
    finish_order date,
    payment_status character varying NOT NULL,
    delivery_status character varying NOT NULL,
    date_order date NOT NULL,
    order_status character varying NOT NULL,
    code_order integer NOT NULL,
    code_customer integer NOT NULL,
```

```
    code_manager integer NOT NULL
);



ALTER TABLE scheme_b."order" OWNER TO postgres;


CREATE SEQUENCE scheme_b.order_code_customer_seq
    AS integer
    START WITH 1
    INCREMENT BY 1
    NO MINVALUE
    NO MAXVALUE
    CACHE 1;



ALTER TABLE scheme_b.order_code_customer_seq OWNER TO postgres;


ALTER SEQUENCE scheme_b.order_code_customer_seq OWNED BY scheme_b."order".code_customer;


CREATE SEQUENCE scheme_b.order_code_manager_seq
    AS integer
    START WITH 1
    INCREMENT BY 1
    NO MINVALUE
```

```
    NO MAXVALUE
    CACHE 1;


ALTER TABLE scheme_b.order_code_manager_seq OWNER TO postgres;


ALTER SEQUENCE scheme_b.order_code_manager_seq OWNED BY scheme_b."order".code_manager;


CREATE SEQUENCE scheme_b.order_code_order_seq
    AS integer
    START WITH 1
    INCREMENT BY 1
    NO MINVALUE
    NO MAXVALUE
    CACHE 1;


ALTER TABLE scheme_b.order_code_order_seq OWNER TO postgres;

ALTER SEQUENCE scheme_b.order_code_order_seq OWNED BY scheme_b."order".code_order;
```

-- добавление данных в таблицу author

```
INSERT INTO scheme_b.author VALUES ('Martin Armstrong ', NULL, 1);

INSERT INTO scheme_b.author VALUES ('Vladimir Lastochkin', 'lasto4ka@gmail.com', 2);
```

```
INSERT INTO scheme_b.author VALUES ('Sergey N.M.', 'serg1234ey@gmail.com', 3);

INSERT INTO scheme_b.author VALUES ('Lilian Rose', 'liliya.rosetta@gmail.com', 4);

INSERT INTO scheme_b.author VALUES ('Charlotte Bronte', NULL, 5);

INSERT INTO scheme_b.author VALUES ('Agatha Christie', NULL, 6);

INSERT INTO scheme_b.author VALUES ('Mary Moore', 'mooremur@gmail.com', 7);

INSERT INTO scheme_b.author VALUES ('Vasiliy Gheltikh', 'gheltikh34@email.ru', 8);

INSERT INTO scheme_b.author VALUES ('Victoria Svobodina', 'freedom_69@email.ru', 9);

INSERT INTO scheme_b.author VALUES ('Sofia Emelkina', 'leviackerman_2512@gmail.com', 10);

INSERT INTO scheme_b.author VALUES ('Marianna Mir', 'marmir_234@gmail.com', 11);

INSERT INTO scheme_b.author VALUES ('Holly Polly Minor', 'holpolmin78493@gmail.com', 12);

INSERT INTO scheme_b.author VALUES ('Vasilisa Carnaval', 'carnavaltiktok@gmail.com', 13);

INSERT INTO scheme_b.author VALUES ('Bunin I.', NULL, 14);
```

--добавление данных в таблицу author_list

```
INSERT INTO scheme_b.author_list VALUES (1, 1, 131, 5);

INSERT INTO scheme_b.author_list VALUES (1, 2, 132, 1);

INSERT INTO scheme_b.author_list VALUES (1, 3, 133, 14);

INSERT INTO scheme_b.author_list VALUES (1, 4, 134, 2);

INSERT INTO scheme_b.author_list VALUES (1, 5, 135, 11);

INSERT INTO scheme_b.author_list VALUES (1, 6, 136, 3);
```

INSERT INTO scheme_b.author_list VALUES (1, 7, 137, 4);

INSERT INTO scheme_b.author_list VALUES (1, 8, 138, 6);

INSERT INTO scheme_b.author_list VALUES (1, 9, 139, 7);

INSERT INTO scheme_b.author_list VALUES (1, 10, 140, 8);

INSERT INTO scheme_b.author_list VALUES (1, 11, 141, 9);

INSERT INTO scheme_b.author_list VALUES (1, 12, 142, 10);

INSERT INTO scheme_b.author_list VALUES (1, 13, 143, 12);

INSERT INTO scheme_b.author_list VALUES (1, 14, 144, 13);

INSERT INTO scheme_b.author_list VALUES (2, 15, 134, 11);

--добавление данных в таблицу book

INSERT INTO scheme_b.book VALUES ('Jane Eyre', 705, 'no', 'Charlotte Bronte', 2019, 131, 1);

INSERT INTO scheme_b.book VALUES ('The Puppet Show', 456, 'yes', 'Martin Armstrong', 2020, 132, 8);

INSERT INTO scheme_b.book VALUES ('Dark Alleys', 239, 'no', 'Bunin I.', 2018, 133, 7);

INSERT INTO scheme_b.book VALUES ('The Flight', 374, 'yes', 'Vladimir Lastochkin

Marianna Mir', 2021, 134, 9);

INSERT INTO scheme_b.book VALUES ('Rose in the Sky', 232, 'yes', 'Marianna Mir', 2016, 135, 3);

INSERT INTO scheme_b.book VALUES ('First love', 127, 'no', 'Sergey N.M.

', 2014, 136, 2);

INSERT INTO scheme_b.book VALUES ('The Blueberry Eye', 542, 'no', 'Lilian Rose', 2010, 137, 10);

INSERT INTO scheme_b.book VALUES ('The Mystery of the Blue Train', 754, 'no', 'Agatha Christie

', 2012, 138, 8);

INSERT INTO scheme_b.book VALUES ('Penelopes diary', 227, 'yes', 'Mary Moore', 2007, 139, 4);

INSERT INTO scheme_b.book VALUES ('Attack on Titan', 1097, 'yes', 'Vasiliy Gheltikh

', 2016, 140, 9);

INSERT INTO scheme_b.book VALUES ('Devoted Assistant for an Idol

', 376, 'no', 'Victoria Svobodina', 2017, 141, 1);

INSERT INTO scheme_b.book VALUES ('The Rise of the Ackermans', 543, 'no', 'Sofia Emelkina', 2021, 142, 9);

INSERT INTO scheme_b.book VALUES ('Totally Spies', 2087, 'yes', 'Holly Polly Minor', 2014, 143, 2);

INSERT INTO scheme_b.book VALUES ('Success', 100, 'no', 'Vasilisa Carnaval', 2020, 144, 5);

--внесение данных в таблицу book_category

INSERT INTO scheme_b.book_category VALUES ('romance novel', 1);

INSERT INTO scheme_b.book_category VALUES ('adventures', 2);

INSERT INTO scheme_b.book_category VALUES ('fantasy', 3);

INSERT INTO scheme_b.book_category VALUES ('for children', 4);

INSERT INTO scheme_b.book_category VALUES ('business', 5);

INSERT INTO scheme_b.book_category VALUES ('historical', 6);

INSERT INTO scheme_b.book_category VALUES ('classical', 7);

INSERT INTO scheme_b.book_category VALUES ('detective', 8);

INSERT INTO scheme_b.book_category VALUES ('psycological', 9);

INSERT INTO scheme_b.book_category VALUES ('humorous', 10);

--внесение данных в таблицу circulation

```
INSERT INTO scheme_b.circulation VALUES (5690, 50000, '2019-01-17', 96, 131);

INSERT INTO scheme_b.circulation VALUES (23, 17000, '2020-08-14', 97, 132);

INSERT INTO scheme_b.circulation VALUES (NULL, 61000, '2018-09-23', 98, 133);

INSERT INTO scheme_b.circulation VALUES (12345, 15000, '2021-05-28', 99, 134);

INSERT INTO scheme_b.circulation VALUES (2016, 10000, '2016-04-24', 100, 135);

INSERT INTO scheme_b.circulation VALUES (489, 30000, '2020-02-14', 101, 136);

INSERT INTO scheme_b.circulation VALUES (21, 24500, '2019-07-12', 102, 137);

INSERT INTO scheme_b.circulation VALUES (2019, 67000, '2021-03-25', 103, 138);

INSERT INTO scheme_b.circulation VALUES (109, 6000, '2020-07-24', 104, 139);

INSERT INTO scheme_b.circulation VALUES (7098, 120000, '2020-11-09', 105, 140);

INSERT INTO scheme_b.circulation VALUES (2310, 15500, '2021-12-23', 106, 141);

INSERT INTO scheme_b.circulation VALUES (1100, 17800, '2021-10-11', 107, 142);

INSERT INTO scheme_b.circulation VALUES (NULL, 35000, '2019-08-12', 108, 143);

INSERT INTO scheme_b.circulation VALUES (12, 7800, '2021-09-25', 109, 144);

INSERT INTO scheme_b.circulation VALUES (1309, 25000, '2021-04-16', 110, 132);

INSERT INTO scheme_b.circulation VALUES (12703, 40000, '2022-01-18', 111, 143);
```

INSERT INTO scheme_b.circulation VALUES (2804, 31000, '2021-10-18', 112, 137);

--внесение данных в таблицу customer

INSERT INTO scheme_b.customer VALUES ('Sofia Bykova', 'SPb, street, flat

', 89213334499, 5);

INSERT INTO scheme_b.customer VALUES ('Giacopo Mingay', '87014 Schmedeman Court', 5013845847, 6);

INSERT INTO scheme_b.customer VALUES ('Faith Jahnig', '13 Texas Center', 5023193389, 7);

INSERT INTO scheme_b.customer VALUES ('Arte Paradine', '6118 Golden Leaf Circle', 4868635685, 8);

INSERT INTO scheme_b.customer VALUES ('Elvin Mackelworth', '24468 5th Crossing', 2402025297, 9);

INSERT INTO scheme_b.customer VALUES ('Max Kubasiewicz', '9187 Kings Hill', 8119748195, 10);

INSERT INTO scheme_b.customer VALUES ('Gerty Troughton', '374 South Lane', 9281367518, 11);

INSERT INTO scheme_b.customer VALUES ('Tomas Vasilic', '503 2nd Hill', 7187038087, 12);

INSERT INTO scheme_b.customer VALUES ('Lorianna Urquhart', '3685 Declaration Crossing', 1124894551, 13);

INSERT INTO scheme_b.customer VALUES ('Haskel Diss', '71 Fairfield Way', 7728126924, 14);

INSERT INTO scheme_b.customer VALUES ('Binni Dunston', '780 Fieldstone Road', 6334616878, 15);

--внесение данных в таблицу include

INSERT INTO scheme_b.include VALUES (1, 639, 1, 131, 75);

INSERT INTO scheme_b.include VALUES (2, 756, 2, 132, 76);

INSERT INTO scheme_b.include VALUES (1, 345, 3, 133, 77);

INSERT INTO scheme_b.include VALUES (1, 321, 4, 135, 78);

INSERT INTO scheme_b.include VALUES (1, 229, 5, 141, 79);

INSERT INTO scheme_b.include VALUES (3, 189, 6, 144, 80);

INSERT INTO scheme_b.include VALUES (1, 345, 7, 133, 81);

INSERT INTO scheme_b.include VALUES (1, 789, 8, 136, 82);

INSERT INTO scheme_b.include VALUES (1, 537, 9, 138, 83);

INSERT INTO scheme_b.include VALUES (1, 234, 10, 139, 84);

INSERT INTO scheme_b.include VALUES (1, 187, 11, 134, 85);

INSERT INTO scheme_b.include VALUES (1, 287, 12, 136, 86);

INSERT INTO scheme_b.include VALUES (1, 399, 13, 142, 76);

INSERT INTO scheme_b.include VALUES (1, 299, 14, 140, 77);

--внесение данных в таблицу manager

INSERT INTO scheme_b.manager VALUES (6704326847, 'Karlan Veazey', 1);

INSERT INTO scheme_b.manager VALUES (5907947489, 'Godiva Deble', 2);

INSERT INTO scheme_b.manager VALUES (1648508886, 'Vanessa Scrowton', 3);

INSERT INTO scheme_b.manager VALUES (4020084521, 'Demetri Trimnell', 4);

INSERT INTO scheme_b.manager VALUES (6283535528, 'Janie Hanbury-Brown', 5);

INSERT INTO scheme_b.manager VALUES (9369911607, 'Adina Bison', 6);

INSERT INTO scheme_b.manager VALUES (4293415613, 'Avram Tincey', 7);

INSERT INTO scheme_b.manager VALUES (2929374457, 'Maurie Shotter', 8);

--внесение данных в таблицу order

INSERT INTO scheme_b."order" VALUES ('2021-03-14', 'оплачено', 'самовывоз', '2021-03-04', 'выполнен', 75, 5, 1);

INSERT INTO scheme_b."order" VALUES ('2021-02-24', 'оплачено', 'с доставкой', '2021-02-17', 'выполнен', 76, 6, 1);

INSERT INTO scheme_b."order" VALUES (NULL, 'не оплачено', 'с доставкой', '2022-03-05', 'в обработке', 77, 6, 2);

INSERT INTO scheme_b."order" VALUES ('2022-01-02', 'оплачено', 'с доставкой', '2021-12-08', 'выполнен', 78, 7, 3);

INSERT INTO scheme_b."order" VALUES (NULL, 'оплачено', 'самовывоз', '2022-02-28', 'принят', 79, 8, 4);

INSERT INTO scheme_b."order" VALUES ('2022-01-21', 'оплачено', 'самовывоз', '2022-01-17', 'выполнен', 80, 9, 5);

INSERT INTO scheme_b."order" VALUES (NULL, 'не оплачено', 'с доставкой', '2022-02-20', 'в работе', 81, 10, 6);

INSERT INTO scheme_b."order" VALUES (NULL, 'не оплачено', 'с доставкой', '2022-03-04', 'в обработке', 82, 11, 7);

INSERT INTO scheme_b."order" VALUES (NULL, 'не оплачено', 'с доставкой', '2022-03-06', 'в работе', 83, 12, 8);

INSERT INTO scheme_b."order" VALUES (NULL, 'оплачено', 'самовывоз', '2022-02-26', 'в работе', 84, 13, 2);

INSERT INTO scheme_b."order" VALUES (NULL, 'не оплачено', 'с доставкой', '2022-02-24', 'принят', 85, 14, 3);

INSERT INTO scheme_b."order" VALUES (NULL, 'оплачено', 'самовывоз', '2022-02-27', 'принят', 86, 15, 4);

--создание ограничений

ALTER TABLE ONLY scheme_b.author_list

ADD CONSTRAINT author_list_pkey PRIMARY KEY (code_list);

ALTER TABLE ONLY scheme_b.author

ADD CONSTRAINT auto_gen1 UNIQUE (code_author);

```
ALTER TABLE ONLY scheme_b.author_list
    ADD CONSTRAINT auto_gen2 UNIQUE (code_list);

ALTER TABLE ONLY scheme_b.book
    ADD CONSTRAINT auto_gen3 UNIQUE ("ISBN");

ALTER TABLE ONLY scheme_b.customer
    ADD CONSTRAINT auto_gen4 UNIQUE (code_customer);

ALTER TABLE ONLY scheme_b.include
    ADD CONSTRAINT auto_gen6 UNIQUE (code_include);
ALTER TABLE ONLY scheme_b.manager
    ADD CONSTRAINT auto_gen7 UNIQUE (code_manager);
ALTER TABLE ONLY scheme_b.book_category
    ADD CONSTRAINT book_category_pkey PRIMARY KEY (code_category);

ALTER TABLE ONLY scheme_b.book
    ADD CONSTRAINT book_pkey PRIMARY KEY ("ISBN");

ALTER TABLE ONLY scheme_b.circulation
    ADD CONSTRAINT circulation_pkey PRIMARY KEY (code_circulation);

ALTER TABLE ONLY scheme_b.book_category
```

```
    ADD CONSTRAINT code_category UNIQUE (code_category);


ALTER TABLE ONLY scheme_b.circulation

    ADD CONSTRAINT code_circulation UNIQUE (code_circulation);


ALTER TABLE ONLY scheme_b."order"

    ADD CONSTRAINT code_order UNIQUE (code_order);


ALTER TABLE scheme_b.include

    ADD CONSTRAINT "copies in order" CHECK ((copy_order >= 0)) NOT
VALID;


ALTER TABLE ONLY scheme_b.customer

    ADD CONSTRAINT customer_pkey PRIMARY KEY (code_customer);


ALTER TABLE scheme_b."order"

    ADD CONSTRAINT date CHECK ((finish_order >= date_order)) NOT
VALID;


ALTER TABLE scheme_b."order"

    ADD CONSTRAINT delivery CHECK (((delivery_status)::text = ANY
((ARRAY['самовывоз'::character varying, 'с доставкой'::character
varying])::text[]))) NOT VALID;
```

```
ALTER TABLE scheme_b.book

    ADD CONSTRAINT illustrations CHECK (((illustration_existence)::text =
ANY (ARRAY[('yes'::character varying)::text, ('no'::character
varying)::text]))) NOT VALID;


ALTER TABLE ONLY scheme_b.include

    ADD CONSTRAINT include_pkey PRIMARY KEY (code_include);


ALTER TABLE ONLY scheme_b.manager

    ADD CONSTRAINT manager_pkey PRIMARY KEY (code_manager);


ALTER TABLE scheme_b."order"

    ADD CONSTRAINT "order" CHECK (((order_status)::text = ANY
((ARRAY['в обработке'::character varying, 'принят'::character varying, 'в
работе'::character varying, 'выполнен'::character varying])::text[]))) NOT
VALID;


ALTER TABLE ONLY scheme_b."order"

    ADD CONSTRAINT order_pkey PRIMARY KEY (code_order);

ALTER TABLE scheme_b.book

    ADD CONSTRAINT pages CHECK ((pages_number > 0)) NOT VALID;


ALTER TABLE scheme_b."order"

    ADD CONSTRAINT payment CHECK (((payment_status)::text = ANY
((ARRAY['оплачено'::character varying, 'не оплачено'::character
varying])::text[]))) NOT VALID;
```

```
ALTER TABLE ONLY scheme_b.author

    ADD CONSTRAINT pk1 PRIMARY KEY (code_author);


ALTER TABLE scheme_b.include

    ADD CONSTRAINT price_book CHECK ((price_book > 0)) NOT VALID;




ALTER TABLE scheme_b.circulation

    ADD CONSTRAINT quantity CHECK ((quantity_in_circulation >= 0)) NOT VALID;


ALTER TABLE scheme_b."order"

    ADD CONSTRAINT start_date CHECK ((date_order <= CURRENT_DATE)) NOT VALID;


ALTER TABLE scheme_b.book

    ADD CONSTRAINT year CHECK (((year_publishing >= 1000) AND (year_publishing < 9999))) NOT VALID;


CREATE INDEX "fki_ISBN" ON scheme_b.author_list USING btree ("ISBN");


CREATE INDEX fki_code_author ON scheme_b.author_list USING btree (code_author);
```

```
CREATE INDEX fki_code_category ON scheme_b.book USING btree (code_category);


CREATE INDEX fki_code_customer ON scheme_b."order" USING btree (code_customer);


CREATE INDEX fki_code_manager ON scheme_b."order" USING btree (code_manager);


CREATE INDEX fki_code_order ON scheme_b.include USING btree (code_order);


ALTER TABLE ONLY scheme_b.author_list

    ADD CONSTRAINT "ISBN" FOREIGN KEY ("ISBN") REFERENCES scheme_b.book("ISBN") DEFERRABLE INITIALLY DEFERRED;


ALTER TABLE ONLY scheme_b.circulation

    ADD CONSTRAINT "ISBN" FOREIGN KEY ("ISBN") REFERENCES scheme_b.book("ISBN") DEFERRABLE INITIALLY DEFERRED NOT VALID;


ALTER TABLE ONLY scheme_b.include

    ADD CONSTRAINT "ISBN" FOREIGN KEY ("ISBN") REFERENCES scheme_b.book("ISBN") DEFERRABLE INITIALLY DEFERRED NOT VALID;
```

ALTER TABLE ONLY scheme_b.author_list

ADD CONSTRAINT code_author FOREIGN KEY (code_author) REFERENCES scheme_b.author(code_author) DEFERRABLE INITIALLY DEFERRED NOT VALID;

ALTER TABLE ONLY scheme_b.book

ADD CONSTRAINT code_category FOREIGN KEY (code_category) REFERENCES scheme_b.book_category(code_category) DEFERRABLE INITIALLY DEFERRED;

ALTER TABLE ONLY scheme_b."order"

ADD CONSTRAINT code_customer FOREIGN KEY (code_customer) REFERENCES scheme_b.customer(code_customer) DEFERRABLE INITIALLY DEFERRED;

ALTER TABLE ONLY scheme_b."order"

ADD CONSTRAINT code_manager FOREIGN KEY (code_manager) REFERENCES scheme_b.manager(code_manager) DEFERRABLE INITIALLY DEFERRED NOT VALID;

ALTER TABLE ONLY scheme_b.include

ADD CONSTRAINT code_order FOREIGN KEY (code_order) REFERENCES scheme_b."order"(code_order) DEFERRABLE INITIALLY DEFERRED NOT VALID;

Вывод

В результате работы были усвоены практические данные, связанные с работой в pgAdmin 4: создание структуры базы данных, добавление различных ограничений, внесение в базу рабочих данных. На базовом уровне изучена работа с PostgreSQL 13. На практике применены знания SQL.