Министерство науки и высшего образования Российской Федерации

федеральное государственное автономное образовательное учреждение
высшего образования

«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»

ОТЧЕТ

по проектной работе

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

по дисциплине «**Проектирование и реализация баз данных**»

Автор:

Петросян Анна Мнацакановна

Факультет:

Инфокоммуникационных технологий

Группа:

К3242

Преподаватель: 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:

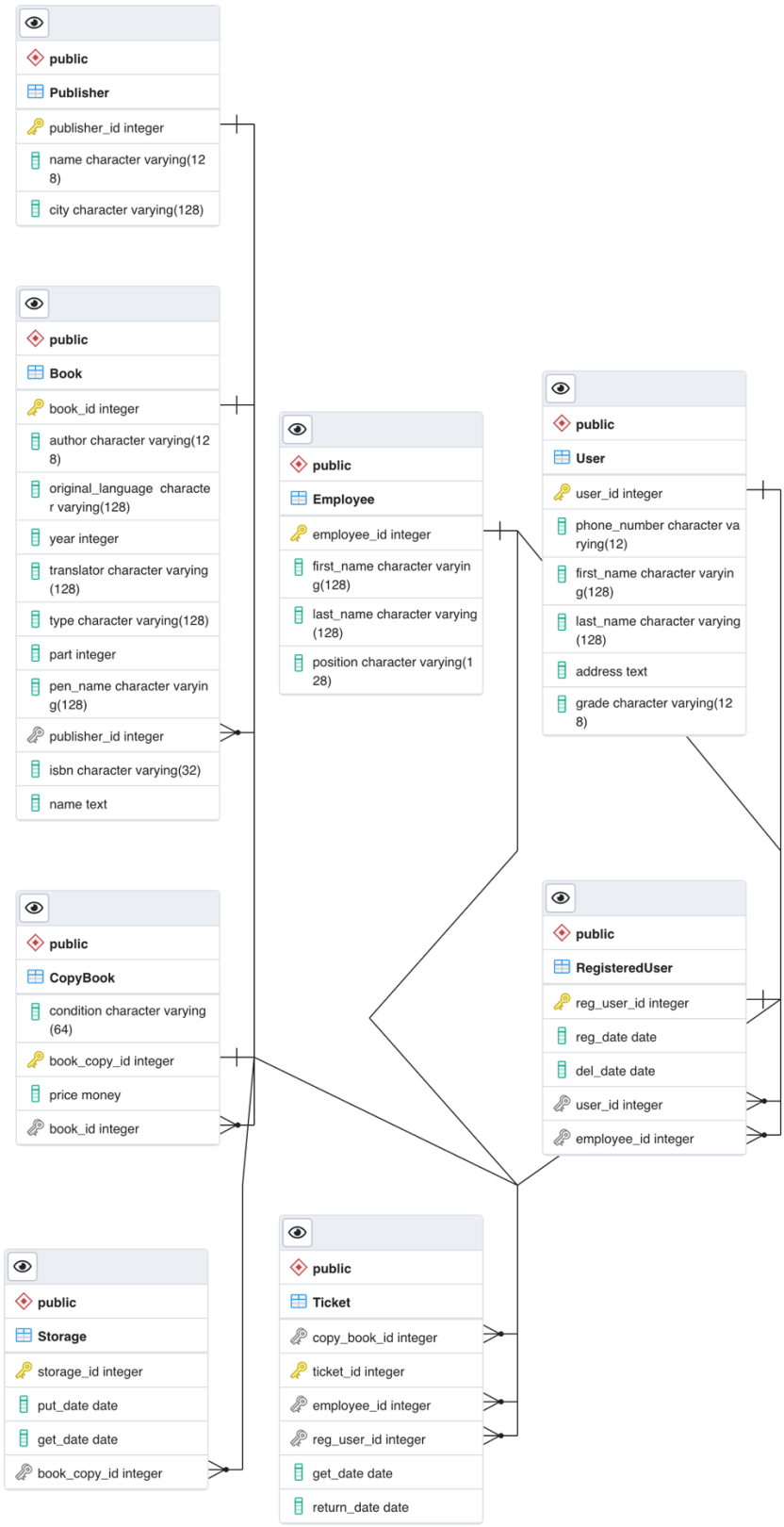
Создать две резервные копии:

- *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.

Выполнение:

- наименование БД - Библиотека
- схема логической модели базы данных, сгенерированная в Generate ERD:


public
Publisher
publisher_id integer
name character varying(128)
city character varying(128)
public
Book
book_id integer
author character varying(128)
original_language character varying(128)
year integer
translator character varying(128)
type character varying(128)
part integer
pen_name character varying(128)
publisher_id integer
isbn character varying(32)
name text
public
Employee
employee_id integer
first_name character varying(128)
last_name character varying(128)
position character varying(128)
public
User
user_id integer
phone_number character varying(12)
first_name character varying(128)
last_name character varying(128)
address text
grade character varying(128)
public
CopyBook
condition character varying(64)
book_copy_id integer
price money
book_id integer
public
RegisteredUser
reg_user_id integer
reg_date date
del_date date
user_id integer
employee_id integer
public
Storage
storage_id integer
put_date date
get_date date
book_copy_id integer
public
Ticket
copy_book_id integer
ticket_id integer
employee_id integer
reg_user_id integer
get_date date
return_date date

- dump, содержащий скрипты работы с БД:

Создание и заполнение (представлено по одному запросу на заполнение каждой таблицы)

```
--
-- PostgreSQL database dump
--

-- Dumped from database version 14.1
-- Dumped by pg_dump version 14.1

-- Started on 2022-03-01 15:40:46 MSK

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;

SET default_tablespace = '';

SET default_table_access_method = heap;

--
-- TOC entry 209 (class 1259 OID 16535)
-- Name: Book; Type: TABLE; Schema: public; Owner: postgres
--

CREATE TABLE public."Book" (
    book_id integer NOT NULL,
    author character varying(128),
    "original_language " character varying(128),
    year integer,
    translator character varying(128),
    type character varying(128) NOT NULL,
    part integer,
    pen_name character varying(128),
    publisher_id integer NOT NULL,
    isbn character varying(32) NOT NULL,
    name text
);


ALTER TABLE public."Book" OWNER TO postgres;

--
-- TOC entry 224 (class 1259 OID 16646)
-- Name: Book_book_id_seq; Type: SEQUENCE; Schema: public; Owner: postgres
--

ALTER TABLE public."Book" ALTER COLUMN book_id ADD GENERATED ALWAYS AS IDENTITY (
    SEQUENCE NAME public."Book_book_id_seq"
    START WITH 1
    INCREMENT BY 1
    NO MINVALUE
    NO MAXVALUE
    CACHE 1
);


--
-- TOC entry 211 (class 1259 OID 16553)
```

```
-- Name: CopyBook; Type: TABLE; Schema: public; Owner: postgres
--

CREATE TABLE public."CopyBook" (
    condition character varying(64) NOT NULL,
    book_copy_id integer NOT NULL,
    price money,
    book_id integer NOT NULL
);


ALTER TABLE public."CopyBook" OWNER TO postgres;

--
-- TOC entry 223 (class 1259 OID 16645)
-- Name: CopyBook_book_copy_id_seq; Type: SEQUENCE; Schema: public; Owner: postgres
--

ALTER TABLE public."CopyBook" ALTER COLUMN book_copy_id ADD GENERATED BY DEFAULT AS IDENTITY (
    SEQUENCE NAME public."CopyBook_book_copy_id_seq"
    START WITH 1
    INCREMENT BY 1
    NO MINVALUE
    NO MAXVALUE
    CACHE 1
);


--
-- TOC entry 213 (class 1259 OID 16589)
-- Name: Employee ; Type: TABLE; Schema: public; Owner: postgres
--

CREATE TABLE public."Employee " (
    employee_id integer NOT NULL,
    first_name character varying(128) NOT NULL,
    last_name character varying(128) NOT NULL,
    "position" character varying(128) NOT NULL
);


ALTER TABLE public."Employee " OWNER TO postgres;

--
-- TOC entry 222 (class 1259 OID 16644)
-- Name: Employee _employee_id_seq; Type: SEQUENCE; Schema: public; Owner: postgres
--

ALTER TABLE public."Employee " ALTER COLUMN employee_id ADD GENERATED ALWAYS AS IDENTITY (
    SEQUENCE NAME public."Employee _employee_id_seq"
    START WITH 1
    INCREMENT BY 1
    NO MINVALUE
    NO MAXVALUE
    CACHE 1
);


--
-- TOC entry 210 (class 1259 OID 16542)
-- Name: Publisher; Type: TABLE; Schema: public; Owner: postgres
--

CREATE TABLE public."Publisher" (
    publisher_id integer NOT NULL,
    name character varying(128) NOT NULL,
    city character varying(128) NOT NULL
);
```

```
ALTER TABLE public."Publisher" OWNER TO postgres;

--
-- TOC entry 221 (class 1259 OID 16643)
-- Name: Publisher_publisher_id_seq; Type: SEQUENCE; Schema: public; Owner: postgres
--

ALTER TABLE public."Publisher" ALTER COLUMN publisher_id ADD GENERATED ALWAYS AS IDENTITY (
    SEQUENCE NAME public."Publisher_publisher_id_seq"
    START WITH 1
    INCREMENT BY 1
    NO MINVALUE
    NO MAXVALUE
    CACHE 1
);


--
-- TOC entry 215 (class 1259 OID 16601)
-- Name: RegisteredUser ; Type: TABLE; Schema: public; Owner: postgres
--

CREATE TABLE public."RegisteredUser " (
    reg_user_id integer NOT NULL,
    reg_date date NOT NULL,
    del_date date NOT NULL,
    user_id integer NOT NULL,
    employee_id integer NOT NULL
);


ALTER TABLE public."RegisteredUser " OWNER TO postgres;

--
-- TOC entry 220 (class 1259 OID 16642)
-- Name: RegisteredUser _reg_user_id_seq; Type: SEQUENCE; Schema: public; Owner: postgres
--

ALTER TABLE public."RegisteredUser " ALTER COLUMN reg_user_id ADD GENERATED ALWAYS AS IDENTITY (
    SEQUENCE NAME public."RegisteredUser _reg_user_id_seq"
    START WITH 1
    INCREMENT BY 1
    NO MINVALUE
    NO MAXVALUE
    CACHE 1
);


--
-- TOC entry 212 (class 1259 OID 16575)
-- Name: Storage; Type: TABLE; Schema: public; Owner: postgres
--

CREATE TABLE public."Storage" (
    storage_id integer NOT NULL,
    put_date date NOT NULL,
    get_date date,
    book_copy_id integer NOT NULL
);


ALTER TABLE public."Storage" OWNER TO postgres;

--
-- TOC entry 219 (class 1259 OID 16641)
-- Name: Storage_storage_id_seq; Type: SEQUENCE; Schema: public; Owner: postgres
--
```

```
ALTER TABLE public."Storage" ALTER COLUMN storage_id ADD GENERATED BY DEFAULT AS IDENTITY (
    SEQUENCE NAME public."Storage_storage_id_seq"
    START WITH 1
    INCREMENT BY 1
    NO MINVALUE
    MAXVALUE 10000
    CACHE 1
);


--
-- TOC entry 216 (class 1259 OID 16616)
-- Name: Ticket; Type: TABLE; Schema: public; Owner: postgres
--

CREATE TABLE public."Ticket" (
    copy_book_id integer NOT NULL,
    ticket_id integer NOT NULL,
    employee_id integer NOT NULL,
    reg_user_id integer NOT NULL,
    get_date date NOT NULL,
    return_date date NOT NULL
);


ALTER TABLE public."Ticket" OWNER TO postgres;

--
-- TOC entry 218 (class 1259 OID 16640)
-- Name: Ticket_ticket_id_seq; Type: SEQUENCE; Schema: public; Owner: postgres
--

ALTER TABLE public."Ticket" ALTER COLUMN ticket_id ADD GENERATED ALWAYS AS IDENTITY (
    SEQUENCE NAME public."Ticket_ticket_id_seq"
    START WITH 1
    INCREMENT BY 1
    NO MINVALUE
    NO MAXVALUE
    CACHE 1
);


--
-- TOC entry 214 (class 1259 OID 16594)
-- Name: User; Type: TABLE; Schema: public; Owner: postgres
--

CREATE TABLE public."User" (
    user_id integer NOT NULL,
    phone_number character varying(12) NOT NULL,
    first_name character varying(128) NOT NULL,
    last_name character varying(128) NOT NULL,
    address text NOT NULL,
    grade character varying(128) NOT NULL
);


ALTER TABLE public."User" OWNER TO postgres;

--
-- TOC entry 217 (class 1259 OID 16639)
-- Name: User_user_id_seq; Type: SEQUENCE; Schema: public; Owner: postgres
--

ALTER TABLE public."User" ALTER COLUMN user_id ADD GENERATED ALWAYS AS IDENTITY (
    SEQUENCE NAME public."User_user_id_seq"
    START WITH 1
    INCREMENT BY 1
    NO MINVALUE
```

```
    NO MAXVALUE
    CACHE 1
);


--
-- TOC entry 3637 (class 0 OID 16535)
-- Dependencies: 209
-- Data for Name: Book; Type: TABLE DATA; Schema: public; Owner: postgres
--

COPY public."Book" (book_id, author, "original_language ", year, translator, type, part, pen_name, publisher_id, isbn, name)
FROM stdin;
1	Pupkin	RU	2000	\N	Magazine	1	\N	1	353535353	Tigers
2	Pupkin	RU	2001	\N	Magazine	1	\N	1	353535333	Monkeys
3	Ted	EN	2020	\N	Comics	1	\N	2	3535352233	Venom
4	John	EN	2015	\N	Comics	1	\N	2	111111111	Spider-Man
5	Petr	RU	2015	\N	Book	1	\N	3	111221111	Along
6	Vasily	EN	2015	\N	Book	1	\N	3	111112111	Who I Am?
7	Vladimir	RU	2015	\N	Guide	1	\N	4	121112111	Math
8	Artem	RU	2015	\N	Guide	1	\N	4	111112112	Economics
9	Armen	RU	2014	\N	Guide	1	\N	5	12221112112	Fresh
watermelons
10	Gurmen	RU	2013	\N	Guide	1	\N	5	1211112112	Cockroaches.
Bees. Ladybugs.
\.


--
-- TOC entry 3639 (class 0 OID 16553)
-- Dependencies: 211
-- Data for Name: CopyBook; Type: TABLE DATA; Schema: public; Owner: postgres
--

COPY public."CopyBook" (condition, book_copy_id, price, book_id) FROM stdin;
Good	1	$40.00	1
Normal	2	$30.00	1
New	3	$65.00	1
New	4	$70.00	2
Good	5	$60.00	2
Good	6	$60.00	2
Normal	7	$50.00	3
Normal	8	$50.00	3
Bad	9	$10.00	4
Bad	10	$10.00	5
\.


--
-- TOC entry 3641 (class 0 OID 16589)
-- Dependencies: 213
-- Data for Name: Employee ; Type: TABLE DATA; Schema: public; Owner: postgres
--

COPY public."Employee " (employee_id, first_name, last_name, "position") FROM stdin;
1	Gala	Dushina	Cleaner
2	Nadegda	Petrova	Librarian
3	Valeria	Rubina	Manager
\.


--
-- TOC entry 3638 (class 0 OID 16542)
-- Dependencies: 210
-- Data for Name: Publisher; Type: TABLE DATA; Schema: public; Owner: postgres
--

COPY public."Publisher" (publisher_id, name, city) FROM stdin;
1	Funny_Kids	Moscow
```

```
2	Marvel	NY
3	The_Best	Saint-Petersburg
4	ABC	Moscow
5	Garden_Flower	Sochi
\.


--
-- TOC entry 3643 (class 0 OID 16601)
-- Dependencies: 215
-- Data for Name: RegisteredUser ; Type: TABLE DATA; Schema: public; Owner: postgres
--

COPY public."RegisteredUser " (reg_user_id, reg_date, del_date, user_id, employee_id) FROM stdin;
3	2022-01-12	2032-01-12	2	2
4	2022-01-23	2032-01-23	3	2
5	2022-02-01	2032-02-01	7	2
\.


--
-- TOC entry 3640 (class 0 OID 16575)
-- Dependencies: 212
-- Data for Name: Storage; Type: TABLE DATA; Schema: public; Owner: postgres
--

COPY public."Storage" (storage_id, put_date, get_date, book_copy_id) FROM stdin;
1	2022-02-20	\N	1
2	2022-02-20	\N	2
3	2022-02-20	\N	3
4	2022-02-20	\N	4
5	2022-02-20	\N	5
6	2022-02-20	\N	6
7	2022-02-20	\N	7
8	2022-02-20	\N	8
9	2022-02-20	\N	9
10	2022-02-20	\N	10
\.


--
-- TOC entry 3644 (class 0 OID 16616)
-- Dependencies: 216
-- Data for Name: Ticket; Type: TABLE DATA; Schema: public; Owner: postgres
--

COPY public."Ticket" (copy_book_id, ticket_id, employee_id, reg_user_id, get_date, return_date) FROM stdin;
1	1	2	3	2022-02-19	2022-03-19
2	2	2	4	2022-01-23	2022-02-23
3	3	2	5	2022-01-02	2022-02-02
5	5	2	5	2022-01-01	2022-01-05
\.


--
-- TOC entry 3642 (class 0 OID 16594)
-- Dependencies: 214
-- Data for Name: User; Type: TABLE DATA; Schema: public; Owner: postgres
--

COPY public."User" (user_id, phone_number, first_name, last_name, address, grade) FROM stdin;
2	+79098674455	Vasya	Petrov	Main_street, 32	Student
3	+79098674456	Anna	Vorobyeva	Second_street,18	Student
4	+79098674457	Lev	Kalashov	Main_street, 2	Student
5	+79098674458	Tom	Voronzov	Main_street,76	Student
6	+79098674459	Sveta	Rubleva	Second_street, 45	Student
7	+79098674460	Vladimir	Pubin	Main_street, 1	Student
\.
```

```
--
-- TOC entry 3658 (class 0 OID 0)
-- Dependencies: 224
-- Name: Book_book_id_seq; Type: SEQUENCE SET; Schema: public; Owner: postgres
--

SELECT pg_catalog.setval('public."Book_book_id_seq"', 1, false);


--
-- TOC entry 3659 (class 0 OID 0)
-- Dependencies: 223
-- Name: CopyBook_book_copy_id_seq; Type: SEQUENCE SET; Schema: public; Owner: postgres
--

SELECT pg_catalog.setval('public."CopyBook_book_copy_id_seq"', 1, true);


--
-- TOC entry 3660 (class 0 OID 0)
-- Dependencies: 222
-- Name: Employee _employee_id_seq; Type: SEQUENCE SET; Schema: public; Owner: postgres
--

SELECT pg_catalog.setval('public."Employee _employee_id_seq"', 3, true);


--
-- TOC entry 3661 (class 0 OID 0)
-- Dependencies: 221
-- Name: Publisher_publisher_id_seq; Type: SEQUENCE SET; Schema: public; Owner: postgres
--

SELECT pg_catalog.setval('public."Publisher_publisher_id_seq"', 1, false);


--
-- TOC entry 3662 (class 0 OID 0)
-- Dependencies: 220
-- Name: RegisteredUser _reg_user_id_seq; Type: SEQUENCE SET; Schema: public; Owner: postgres
--

SELECT pg_catalog.setval('public."RegisteredUser _reg_user_id_seq"', 5, true);


--
-- TOC entry 3663 (class 0 OID 0)
-- Dependencies: 219
-- Name: Storage_storage_id_seq; Type: SEQUENCE SET; Schema: public; Owner: postgres
--

SELECT pg_catalog.setval('public."Storage_storage_id_seq"', 10, true);


--
-- TOC entry 3664 (class 0 OID 0)
-- Dependencies: 218
-- Name: Ticket_ticket_id_seq; Type: SEQUENCE SET; Schema: public; Owner: postgres
--

SELECT pg_catalog.setval('public."Ticket_ticket_id_seq"', 5, true);


--
-- TOC entry 3665 (class 0 OID 0)
-- Dependencies: 217
-- Name: User_user_id_seq; Type: SEQUENCE SET; Schema: public; Owner: postgres
--
```

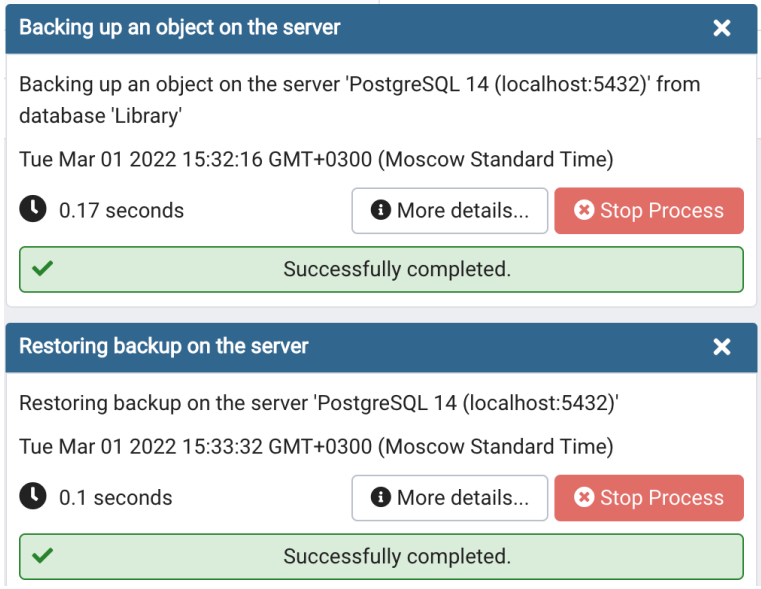
```
SELECT pg_catalog.setval('public."User_user_id_seq"', 7, true);


--
-- TOC entry 3469 (class 2606 OID 16541)
-- Name: Book Book_pkey; Type: CONSTRAINT; Schema: public; Owner: postgres
--

ALTER TABLE ONLY public."Book"
    ADD CONSTRAINT "Book_pkey" PRIMARY KEY (book_id);


--
-- TOC entry 3474 (class 2606 OID 16557)
-- Name: CopyBook CopyBook_pkey; Type: CONSTRAINT; Schema: public; Owner: postgres
--

ALTER TABLE ONLY public."CopyBook"
    ADD CONSTRAINT "CopyBook_pkey" PRIMARY KEY (book_copy_id);


--
-- TOC entry 3481 (class 2606 OID 16593)
-- Name: Employee  Employee _pkey; Type: CONSTRAINT; Schema: public; Owner: postgres
--

ALTER TABLE ONLY public."Employee "
    ADD CONSTRAINT "Employee _pkey" PRIMARY KEY (employee_id);


--
-- TOC entry 3472 (class 2606 OID 16546)
-- Name: Publisher Publisher_pkey; Type: CONSTRAINT; Schema: public; Owner: postgres
--

ALTER TABLE ONLY public."Publisher"
    ADD CONSTRAINT "Publisher_pkey" PRIMARY KEY (publisher_id);


--
-- TOC entry 3485 (class 2606 OID 16605)
-- Name: RegisteredUser  RegisteredUser _pkey; Type: CONSTRAINT; Schema: public; Owner: postgres
--

ALTER TABLE ONLY public."RegisteredUser "
    ADD CONSTRAINT "RegisteredUser _pkey" PRIMARY KEY (reg_user_id);


--
-- TOC entry 3477 (class 2606 OID 16579)
-- Name: Storage Storage_pkey; Type: CONSTRAINT; Schema: public; Owner: postgres
--

ALTER TABLE ONLY public."Storage"
    ADD CONSTRAINT "Storage_pkey" PRIMARY KEY (storage_id);


--
-- TOC entry 3487 (class 2606 OID 16620)
-- Name: Ticket Ticket_pkey; Type: CONSTRAINT; Schema: public; Owner: postgres
--

ALTER TABLE ONLY public."Ticket"
    ADD CONSTRAINT "Ticket_pkey" PRIMARY KEY (ticket_id);


--
-- TOC entry 3483 (class 2606 OID 16600)
```

```
-- Name: User User_pkey; Type: CONSTRAINT; Schema: public; Owner: postgres
--

ALTER TABLE ONLY public."User"
    ADD CONSTRAINT "User_pkey" PRIMARY KEY (user_id);


--
-- TOC entry 3489 (class 2606 OID 16637)
-- Name: Ticket bookCopy_id; Type: CONSTRAINT; Schema: public; Owner: postgres
--

ALTER TABLE ONLY public."Ticket"
    ADD CONSTRAINT "bookCopy_id" UNIQUE (copy_book_id);


--
-- TOC entry 3479 (class 2606 OID 16586)
-- Name: Storage book_copy_id; Type: CONSTRAINT; Schema: public; Owner: postgres
--

ALTER TABLE ONLY public."Storage"
    ADD CONSTRAINT book_copy_id UNIQUE (book_copy_id);


--
-- TOC entry 3467 (class 2606 OID 16638)
-- Name: Ticket get_date < return_date; Type: CHECK CONSTRAINT; Schema: public; Owner: postgres
--

ALTER TABLE public."Ticket"
    ADD CONSTRAINT "get_date < return_date" CHECK ((get_date < return_date)) NOT VALID;


--
-- TOC entry 3465 (class 2606 OID 16588)
-- Name: Storage get_date > put_date; Type: CHECK CONSTRAINT; Schema: public; Owner: postgres
--

ALTER TABLE public."Storage"
    ADD CONSTRAINT "get_date > put_date" CHECK ((get_date > put_date)) NOT VALID;


--
-- TOC entry 3466 (class 2606 OID 16647)
-- Name: RegisteredUser  reg_date_check; Type: CHECK CONSTRAINT; Schema: public; Owner: postgres
--

ALTER TABLE public."RegisteredUser "
    ADD CONSTRAINT reg_date_check CHECK ((reg_date < del_date)) NOT VALID;


--
-- TOC entry 3475 (class 1259 OID 16563)
-- Name: fki_book_id; Type: INDEX; Schema: public; Owner: postgres
--

CREATE INDEX fki_book_id ON public."CopyBook" USING btree (book_id);


--
-- TOC entry 3470 (class 1259 OID 16552)
-- Name: fki_publisher_id; Type: INDEX; Schema: public; Owner: postgres
--

CREATE INDEX fki_publisher_id ON public."Book" USING btree (publisher_id);


--
```

```
-- TOC entry 3494 (class 2606 OID 16611)
-- Name: RegisteredUser  Employee_id; Type: FK CONSTRAINT; Schema: public; Owner: postgres
--

ALTER TABLE ONLY public."RegisteredUser "
    ADD CONSTRAINT "Employee_id" FOREIGN KEY (employee_id) REFERENCES public."Employee "(employee_id);


--
-- TOC entry 3492 (class 2606 OID 16580)
-- Name: Storage Storage_book_copy_id_fkey; Type: FK CONSTRAINT; Schema: public; Owner: postgres
--

ALTER TABLE ONLY public."Storage"
    ADD CONSTRAINT "Storage_book_copy_id_fkey" FOREIGN KEY (book_copy_id) REFERENCES
public."CopyBook"(book_copy_id);


--
-- TOC entry 3493 (class 2606 OID 16606)
-- Name: RegisteredUser  User_id; Type: FK CONSTRAINT; Schema: public; Owner: postgres
--

ALTER TABLE ONLY public."RegisteredUser "
    ADD CONSTRAINT "User_id" FOREIGN KEY (user_id) REFERENCES public."User"(user_id);


--
-- TOC entry 3491 (class 2606 OID 16558)
-- Name: CopyBook book_id; Type: FK CONSTRAINT; Schema: public; Owner: postgres
--

ALTER TABLE ONLY public."CopyBook"
    ADD CONSTRAINT book_id FOREIGN KEY (book_id) REFERENCES public."Book"(book_id);


--
-- TOC entry 3497 (class 2606 OID 16631)
-- Name: Ticket copy_book_id; Type: FK CONSTRAINT; Schema: public; Owner: postgres
--

ALTER TABLE ONLY public."Ticket"
    ADD CONSTRAINT copy_book_id FOREIGN KEY (copy_book_id) REFERENCES public."CopyBook"(book_copy_id);


--
-- TOC entry 3496 (class 2606 OID 16626)
-- Name: Ticket employee_id; Type: FK CONSTRAINT; Schema: public; Owner: postgres
--

ALTER TABLE ONLY public."Ticket"
    ADD CONSTRAINT employee_id FOREIGN KEY (employee_id) REFERENCES public."Employee "(employee_id);


--
-- TOC entry 3490 (class 2606 OID 16547)
-- Name: Book publisher_id; Type: FK CONSTRAINT; Schema: public; Owner: postgres
--

ALTER TABLE ONLY public."Book"
    ADD CONSTRAINT publisher_id FOREIGN KEY (publisher_id) REFERENCES public."Publisher"(publisher_id);


--
-- TOC entry 3495 (class 2606 OID 16621)
-- Name: Ticket reg_user_id; Type: FK CONSTRAINT; Schema: public; Owner: postgres
--

ALTER TABLE ONLY public."Ticket"
```

```
    ADD CONSTRAINT reg_user_id FOREIGN KEY (reg_user_id) REFERENCES public."RegisteredUser "(reg_user_id);


-- Completed on 2022-03-01 15:40:47 MSK

--
-- PostgreSQL database dump complete
--
```

Резервная копия и восстановление



Вывод:

В ходе выполнения лабораторной я спроектировал базу данных и наполнил ее данными. Были установлены ограничения: Florien Keys, Check, Unique. Была построена схема, аналогично той, что мы делали ранее в Erwin Data Modeler. Так же я успешно сделал резервную копию и восстановил из нее БД.