

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ
СІКОРСЬКОГО»

Факультет прикладної математики
Кафедра програмного забезпечення комп'ютерних систем

ЗВІТ
з лабораторної роботи № 1
«Проектування бази даних та ознайомлення з базовими операціями
СУБД PostgreSQL»

Виконала:

студентка 3-го курсу, групи КП-82,
спеціальності 121 – Інженерія
програмного забезпечення
Лахман Ксенія Вікторівна

Київ – 2020

Хід роботи

Варіант: Бібліотека

Розроблена ER-модель (нормалізована):

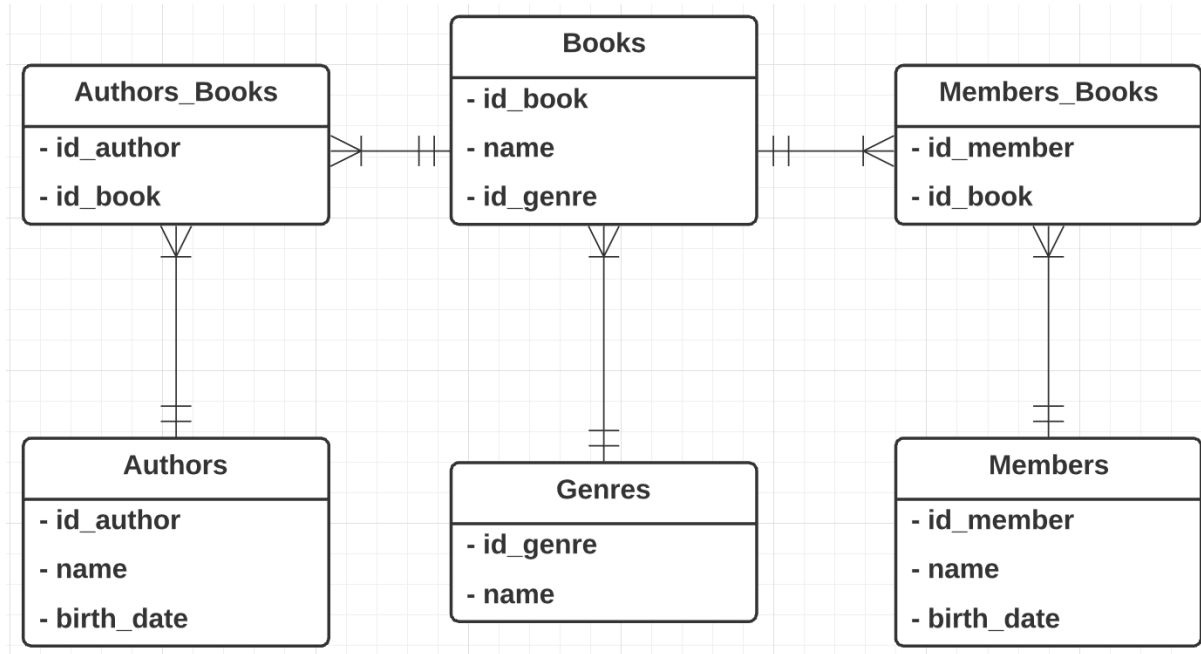


Рис. 1 ER-модель

Таблиці бази даних «Бібліотека»:




	 id_author [PK] integer		name text		birth_date date	
1		1	Joanne Rowling		1965-07-31	
2		2	Erich Paul Remark		1898-06-22	
3		3	Irwin Shaw		1913-02-27	
4		4	Arthur Conan Doyle		1859-05-22	

Рис. 2 *Authors*

	id_book [PK] integer	name text	id_genre integer
1	1	Pride and Prejudice	3
2	2	Harry Potter and the Philo...	1
3	3	Harry Potter and the Cha...	1
4	4	The Night in Lisbon	4

Рис. 3 *Books*

	id_genre [PK] integer	name text
1	1	Fantasy
2	2	Detective fiction
3	3	Romance novel
4	4	War novel

Рис. 4 *Genres*

	id_member [PK] integer	name text	birth_date date
1	1	Lakhman Kseniya	2001-02-06
2	2	Shapovalov Danylo	2001-08-19
3	3	Glushko Nastya	2001-01-19
4	4	Gorodivsky Vladislav	1999-09-18

Рис. 5 *Members*

	id_author integer	id_book integer
1	1	2
2	1	3
3	2	4

Рис. 6 *Authors_Books*:

	id_member integer	id_book integer
1	1	1
2	1	2
3	2	3
4	2	4

Рис. 7 *Members_Books*:

Скрипти, за допомогою якого було відображено таблиці та перевірено правильність побудованих зв'язків:

```
SELECT public."Authors".name, public."Books".name
FROM public."Authors" join public."Authors_Books"
ON public."Authors".id_author =
public."Authors_Books".id_author
join public."Books"
ON public."Authors_Books".id_book = public."Books".id_book;
```

Результат:

	name text	name text
1	Joanne Rowling	Harry Potter and the Philosopher`s Stone
2	Joanne Rowling	Harry Potter and the Chamber of Secrets
3	Erich Paul Rema...	The Night in Lisbon

```

SELECT public."Members".name, public."Books".name
FROM public."Members" join public."Members_Books"
ON public."Members".id_member =
public."Members_Books".id_member
join public."Books"
ON public."Members_Books".id_book = public."Books".id_book;

```

Результат:

	name text	name text
1	Lakhman Kseniya	Pride and Prejudice
2	Lakhman Kseniya	Harry Potter and the Philosopher`s Stone
3	Shapovalov Danylo	Harry Potter and the Chamber of Secrets
4	Shapovalov Danylo	The Night in Lisbon

```

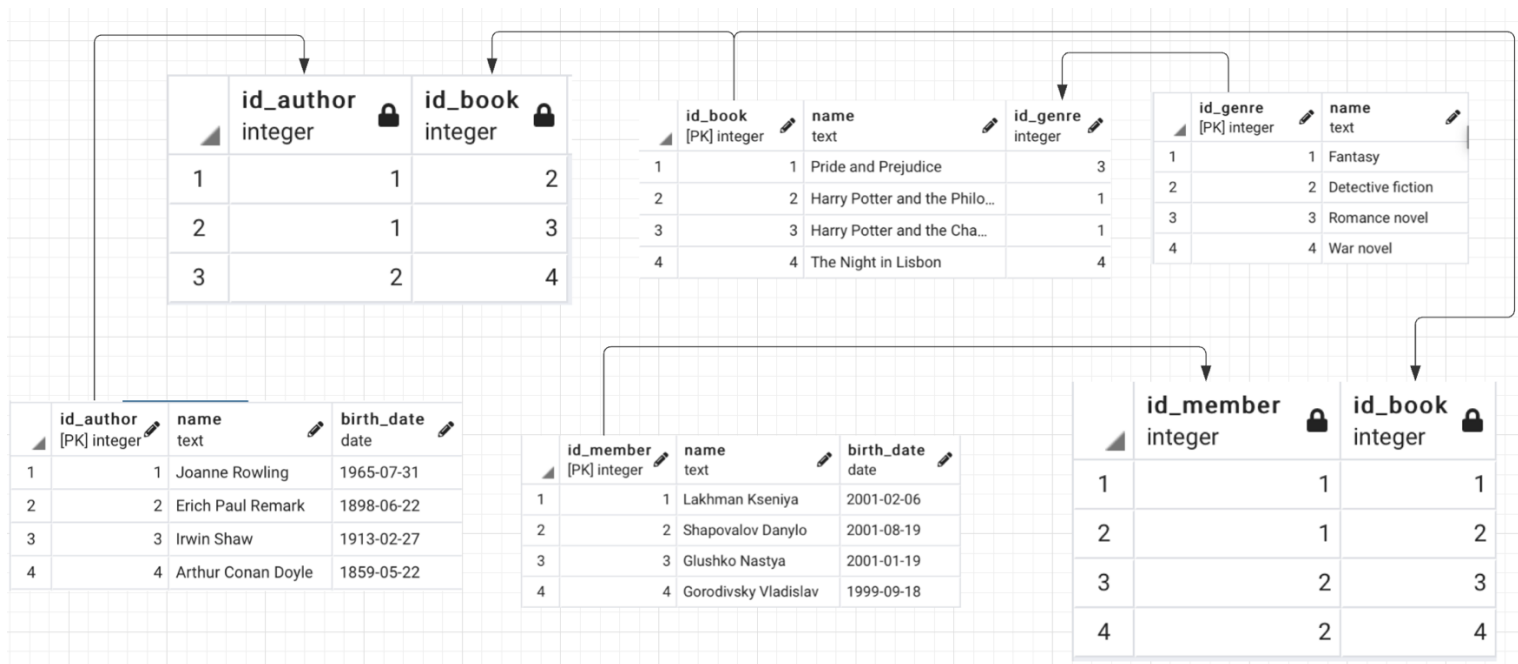
SELECT public."Books".name, public."Genres".name
FROM public."Genres" join public."Books"
ON public."Genres".id_genre = public."Books".id_genre

```

Результат:

	name text	name text
1	Pride and Prejudice	Romance novel
2	Harry Potter and the Philosopher`s Stone	Fantasy
3	Harry Potter and the Chamber of Secrets	Fantasy
4	The Night in Lisbon	War novel

Структура нормалізованої бази даних



SQL

1) Authors

```
CREATE TABLE public."Authors"
(
    id_author integer NOT NULL DEFAULT nextval('"Authors_id_author_seq"'::regclass),
    name text COLLATE pg_catalog."default" NOT NULL,
    birth_date date NOT NULL,
    CONSTRAINT "Authors_pkey" PRIMARY KEY (id_author)
)
WITH (
    OIDS = FALSE
)
TABLESPACE pg_default;

ALTER TABLE public."Authors"
    OWNER to postgres;
```

2) Books

```

CREATE TABLE public."Books"
(
    id_book integer NOT NULL DEFAULT nextval('"Books_id_book_seq" '::regclass),
    name text COLLATE pg_catalog."default" NOT NULL,
    id_genre integer NOT NULL,
    CONSTRAINT "Books_pkey" PRIMARY KEY (id_book),
    CONSTRAINT id_genre FOREIGN KEY (id_genre)
        REFERENCES public."Genres" (id_genre) MATCH SIMPLE
        ON UPDATE NO ACTION
        ON DELETE NO ACTION
        NOT VALID
)
WITH (
    OIDS = FALSE
)
TABLESPACE pg_default;

ALTER TABLE public."Books"
    OWNER to postgres;

```

3) Members

```

CREATE TABLE public."Members"
(
    id_member integer NOT NULL DEFAULT nextval('"Members_id_member_seq" '::regclass),
    name text COLLATE pg_catalog."default" NOT NULL,
    birth_date date NOT NULL,
    CONSTRAINT "Members_pkey" PRIMARY KEY (id_member)
)
WITH (
    OIDS = FALSE
)
TABLESPACE pg_default;

ALTER TABLE public."Members"
    OWNER to postgres;

```

4) Genres

```

CREATE TABLE public."Genres"
(
    id_genre integer NOT NULL DEFAULT nextval('"Genres_id_genre_seq"'::regclass),
    name text COLLATE pg_catalog."default" NOT NULL,
    CONSTRAINT "Genres_pkey" PRIMARY KEY (id_genre)
)
WITH (
    OIDS = FALSE
)
TABLESPACE pg_default;

ALTER TABLE public."Genres"
    OWNER to postgres;

```

5) Authors_Books

```

CREATE TABLE public."Authors_Books"
(
    id_author integer NOT NULL,
    id_book integer NOT NULL,
    CONSTRAINT id_author FOREIGN KEY (id_author)
        REFERENCES public."Authors" (id_author) MATCH SIMPLE
        ON UPDATE NO ACTION
        ON DELETE NO ACTION
        NOT VALID,
    CONSTRAINT id_book FOREIGN KEY (id_book)
        REFERENCES public."Books" (id_book) MATCH SIMPLE
        ON UPDATE NO ACTION
        ON DELETE NO ACTION
        NOT VALID
)
WITH (
    OIDS = FALSE
)
TABLESPACE pg_default;

ALTER TABLE public."Authors_Books"
    OWNER to postgres;

```

6) Members_Books


```
CREATE TABLE public."Members_Books"
(
    id_member integer NOT NULL,
    id_book integer NOT NULL,
    CONSTRAINT id_book FOREIGN KEY (id_book)
        REFERENCES public."Books" (id_book) MATCH SIMPLE
        ON UPDATE NO ACTION
        ON DELETE NO ACTION
        NOT VALID,
    CONSTRAINT id_member FOREIGN KEY (id_member)
        REFERENCES public."Members" (id_member) MATCH SIMPLE
        ON UPDATE NO ACTION
        ON DELETE NO ACTION
        NOT VALID
)
WITH (
    OIDS = FALSE
)
TABLESPACE pg_default;

ALTER TABLE public."Members_Books"
    OWNER to postgres;
```

ВИСНОВОК

При виконанні даної лабораторної роботи я навчилася створювати таблиці, будувати ER-модель та нормалізувати схеми баз даних, писати прості SQL скрипти та використовувати базовий функціонал pgAdmin4.