



НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ імені Ігоря Сікорського»

ФАКУЛЬТЕТ ПРИКЛАДНОЇ МАТЕМАТИКИ

**Кафедра системного програмування та спеціалізованих
комп'ютерних систем**

Лабораторна робота №1

з дисципліни

«Бази даних і засоби управління»

**Тема: «Створити БД в СУБД PostgreSQL
з допомогою конструктора PgAdmin 4»**

Виконав: студент III курсу

ФПМ групи KB-83

Панфілов Ярослав Юрійович

Перевірив: Павловський В.І.

Київ – 2020

Лабораторна робота №1.
Створити БД "Блог" в СУБД PostgreSQL
з допомогою конструктора pgAdmin 4

Мета роботи: створити БД "Блог" та сформувані обмеження цілісності на значення даних.

Порядок виконання роботи

1. Розробити концептуальну модель вибраного предметного середовища. Концептуальна модель предметного середовища "Блог" наводиться в Додатку А до лабораторної роботи;
2. Розробити логічну модель (схему даних) БД;
3. Вивчити склад та правила роботи с СУБД PostgreSQL;
4. Створити в СУБД PostgreSQL БД "Блог", використовуючи конструктори таблиць та стовпчиків. Схема даних БД "Блог" наводиться в Додатку Б до лабораторної роботи. Перелік атрибутів наводиться в Додатку В до лабораторної роботи;
5. Сформувані обмеження цілісності, що забезпечують:
 - унікальність та обов'язковість вводу первинних ключів для всіх таблиць;
 - перевірка на відповідність зовнішніх ключів таблиць;
 - обмеження на значення даних для атрибутів і вивід відповідних повідомлень при їх порушенні;
 - обов'язковість вводу даних атрибутів;
 - сформувані маску вводу для атрибутів;
6. Заповнити створену БД даними (порядку 5-10 записів в кожній таблиці).
7. Вивести вміст таблиць створеної БД.

Зміст звіту

1. Склад СУБД PostgreSQL;
2. Опис предметної галузі;
3. Концептуальна модель предметної області;
4. Логічна модель БД;
5. Список обмежень цілісності в термінах СУБД PostgreSQL;
6. Представлення БД в pgAdmin 4.

Опис предметної галузі

При проектуванні бази даних “Блог” можна виділити такі сутності: Стаття(Article), Тема статті(Theme), Користувач(User), Коментарі(Comment), Паспорт користувача(Passport).

До теми статті може належати декілька статей (один до багатьох).

Стаття може мати 0 або більше коментарів (один до багатьох).

Клієнт може написати 0 або більше коментарів (один до багатьох).

Користувач може прочитати багато статей і одна стаття може бути прочитана багатьма клієнтами (багато до багатьох).

Користувач повинен мати паспорт (один до одного).

Додаток А. Концептуальна модель предметної області “Блог”

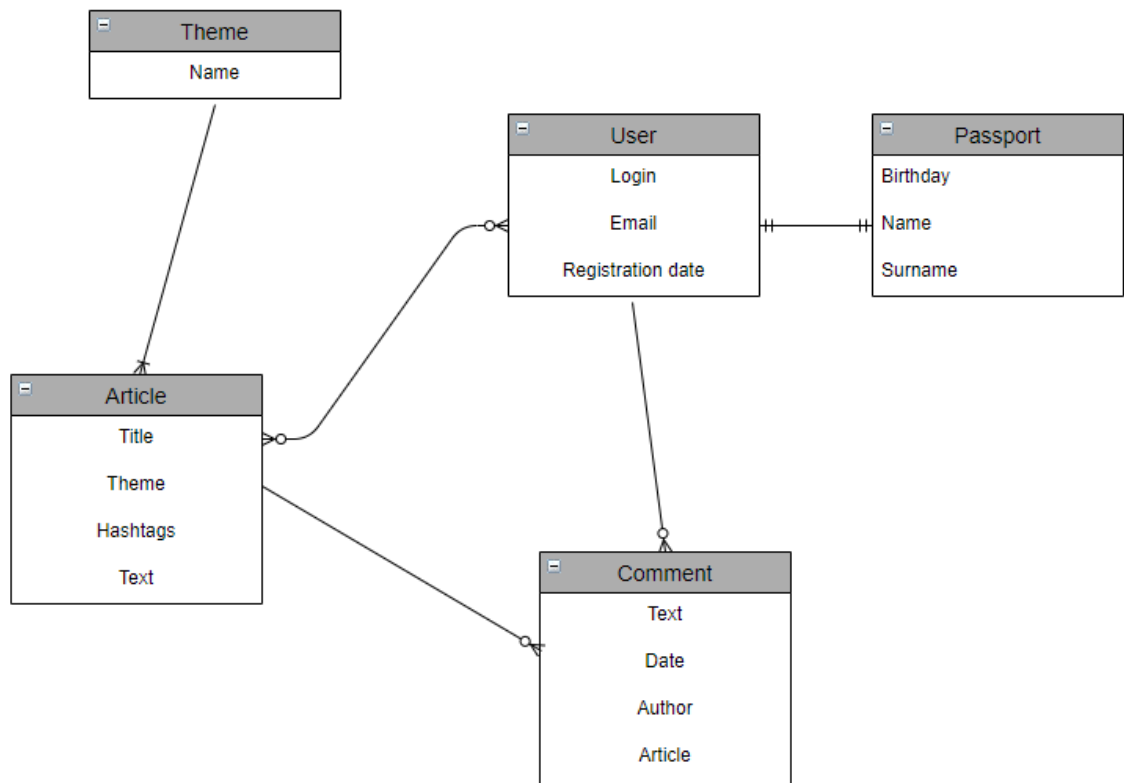


Рисунок 1 – Концептуальна модель предметної області "Блог"

Додаток Б. Структура БД “Блог”

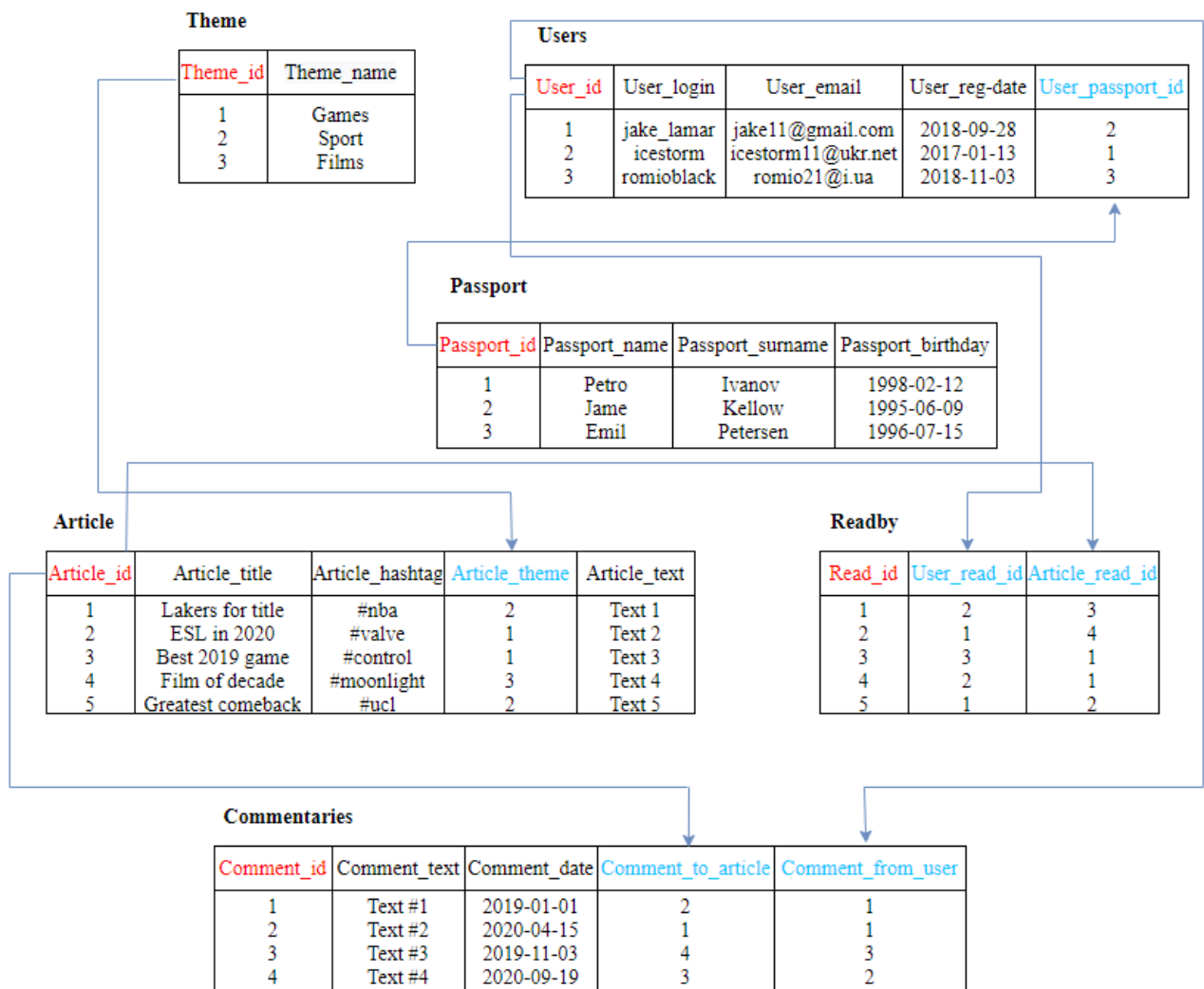


Рисунок 2 – структурна модель предметної області "Блог"

Додаток В. Опис структури БД “Блог”

ВІДНОШЕННЯ	АТРИБУТ	ТИП(Розмір)
Відношення “Theme” Містить інформацію про тему статті	Theme_id – унікальний ID теми в БД Theme_name – ім’я теми	Числовий Текстовий(20)
Відношення “Article” Містить інформацію про статтю	Article_id - унікальний ID статті в БД Article_title – заголовок статті Article_hashtag – хештег у статті Article_text – текст статті ArticleTheme_id – ID теми, до якої належить стаття	Числовий Текстовий(20) Текстовий(10) Текст Числовий
Відношення “Read” Вміщує інформацію про факт прочитання статті	Read_id – унікальний ID прочитання статті в БД UserRead_id – ID користувача, який прочитав статтю ArticleRead_id – ID статті, яка була прочитана	Числовий Числовий Числовий
Відношення “User” Вміщує інформацію про користувача в блозі	User_id – унікальний ID користувача User_login – логін користувача User_email – e-mail користувача User_Reg-Date– дата реєстрації користувача	Числовий Текстовий(20) Текстовий(40) Дата
Відношення “Comment” Вміщує в собі інформацію про коментар про статтю	Comment_id – унікальний ID коментаря Comment_text – текст коментаря Comment_date – дата написання коментаря UserComment_id – ID автора коментаря ArticleComment_id – ID статті до якої написано відгук	Числовий Текст Дата Числовий Числовий
Відношення “Passport” Вміщує в собі інформацію про коментар про паспорт користувача	Passport_id – унікальний ID паспорта Passport_name – ім’я в паспорті Passport_surname – прізвище в паспорті Passport_birthday – дата народження	Числовий Текстовий(50) Текстовий(50) Дата

Додаток Г. Логічна модель БД “Блог” (засобами SqlDBM)

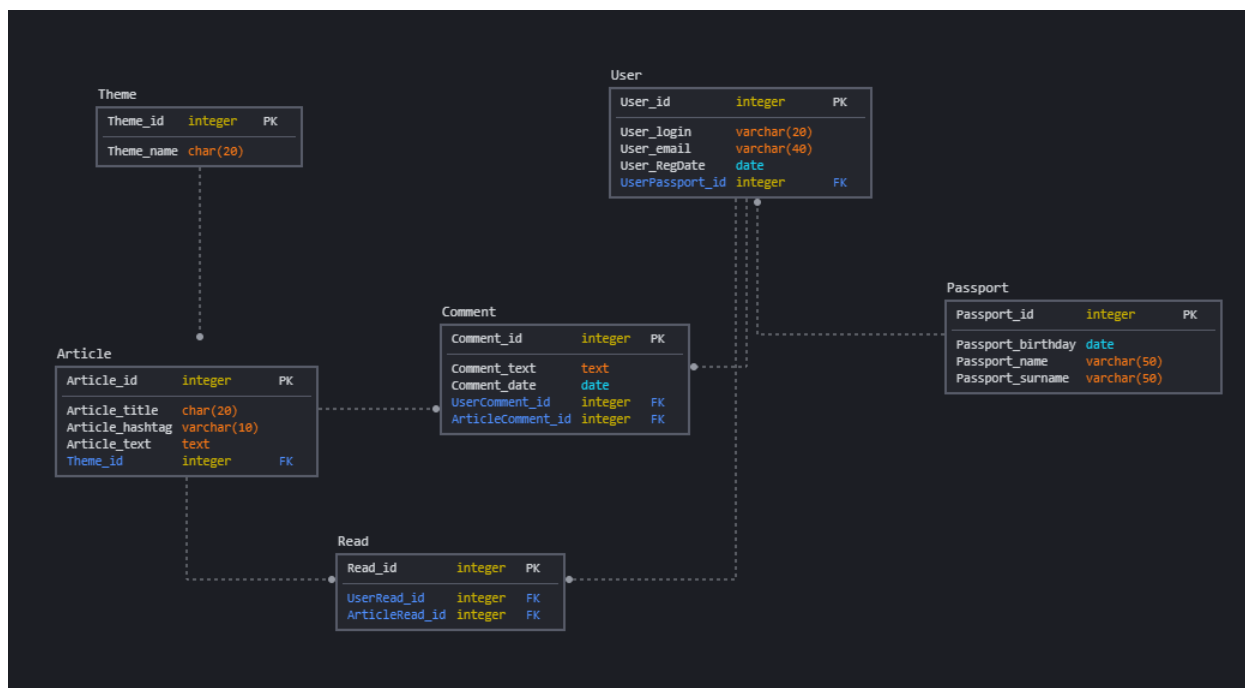


Рисунок 3 – Логічна модель предметної області "Блог"

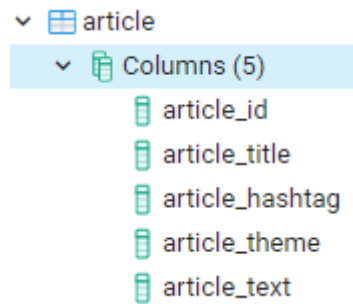
Схема бази даних відповідає 1НФ тому, що всі атрибути прості, повторень рядків немає і атрибути містять лише скалярні значення.


Схема бази даних відповідає 2НФ тому, що всі таблиці мають первинні ключі, і кожний неключовий атрибут залежить від первинного ключа, а не від його частини або іншого неключового атрибута.


Схема бази даних відповідає 3НФ тому, що всі атрибути залежні тільки від первинного ключа.






Додаток Д. Структура БД “Блог” в pgAdmin 4

Таблиця Article

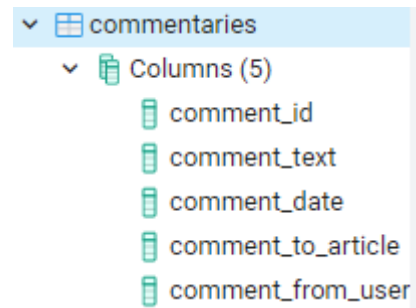



▼  article


▼  Columns (5)






 article_id
 article_title
 article_hashtag
 article_theme
 article_text

Таблиця Commentaries

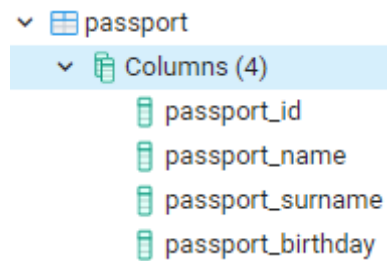



▼  commentaries


▼  Columns (5)





 comment_id
 comment_text
 comment_date
 comment_to_article
 comment_from_user

Таблиця Passport

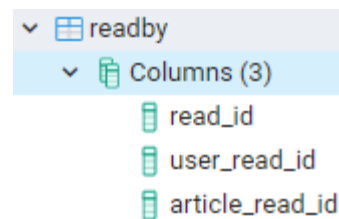



▼  passport


▼  Columns (4)




 passport_id
 passport_name
 passport_surname
 passport_birthday

Таблиця Readby

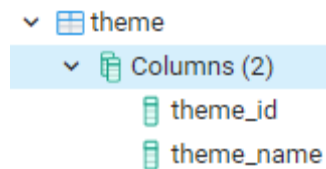



▼  readby


▼  Columns (3)



 read_id
 user_read_id
 article_read_id

Таблиця Theme

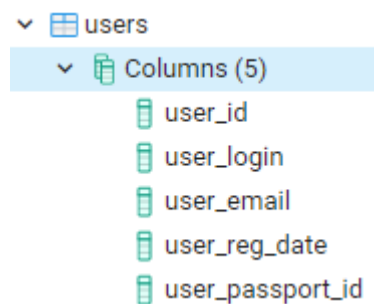



▼  theme


▼  Columns (2)






 theme_id
 theme_name

Таблиця Users



▼  users

▼  Columns (5)

 user_id
 user_login
 user_email
 user_reg_date
 user_passport_id

Опис таблиць в pgAdmin 4

Таблиця Article

```
CREATE TABLE public.article
(
    article_id integer NOT NULL DEFAULT nextval('article_article_id_seq'::regclass),
    article_title character varying(20) COLLATE pg_catalog."default" NOT NULL,
    article_hashtag character varying(10) COLLATE pg_catalog."default",
    article_theme integer NOT NULL,
    article_text text COLLATE pg_catalog."default" NOT NULL,
    CONSTRAINT article_idpk PRIMARY KEY (article_id),
    CONSTRAINT article_theme_idfk FOREIGN KEY (article_theme)
        REFERENCES public.theme (theme_id) MATCH SIMPLE
        ON UPDATE CASCADE
        ON DELETE CASCADE
)

TABLESPACE pg_default;

ALTER TABLE public.article
    OWNER to postgres;
```

Таблиця Commentaries

```
CREATE TABLE public.commentaries
(
    comment_id integer NOT NULL DEFAULT nextval('commentaries_comment_id_seq'::regclass),
    comment_text text COLLATE pg_catalog."default" NOT NULL,
    comment_date date NOT NULL,
    comment_to_article integer NOT NULL,
    comment_from_user integer NOT NULL,
    CONSTRAINT comment_idpk PRIMARY KEY (comment_id),
    CONSTRAINT comment_from_user_idfk FOREIGN KEY (comment_from_user)
        REFERENCES public.article (article_id) MATCH SIMPLE
        ON UPDATE CASCADE
        ON DELETE CASCADE,
    CONSTRAINT comment_to_article_idfk FOREIGN KEY (comment_to_article)
        REFERENCES public.article (article_id) MATCH FULL
        ON UPDATE CASCADE
        ON DELETE CASCADE
)

TABLESPACE pg_default;

ALTER TABLE public.commentaries
    OWNER to postgres;
```


Таблица Readby

```
CREATE TABLE public.readby
(
    read_id integer NOT NULL,
    user_read_id integer NOT NULL,
    article_read_id integer NOT NULL,
    CONSTRAINT read_idpk PRIMARY KEY (read_id),
    CONSTRAINT article_read_idfk FOREIGN KEY (article_read_id)
        REFERENCES public.article (article_id) MATCH SIMPLE
        ON UPDATE CASCADE
        ON DELETE CASCADE,
    CONSTRAINT user_read_idfk FOREIGN KEY (user_read_id)
        REFERENCES public.users (user_id) MATCH SIMPLE
        ON UPDATE CASCADE
        ON DELETE CASCADE
)

TABLESPACE pg_default;

ALTER TABLE public.readby
    OWNER to postgres;
```

Таблица Theme

```
CREATE TABLE public.theme
(
    theme_id integer NOT NULL DEFAULT nextval('theme_theme_id_seq'::regclass),
    theme_name character varying(20) COLLATE pg_catalog."default" NOT NULL,
    CONSTRAINT theme_id_pk PRIMARY KEY (theme_id)
)

TABLESPACE pg_default;

ALTER TABLE public.theme
    OWNER to postgres;
```

Таблица Users

```
CREATE TABLE public.users
(
    user_id integer NOT NULL,
    user_login character varying(20) COLLATE pg_catalog."default",
    user_email character varying(40) COLLATE pg_catalog."default",
    user_reg_date date,
    user_passport_id integer,
    CONSTRAINT user_idpk PRIMARY KEY (user_id),
    CONSTRAINT users_user_passport_id_key UNIQUE (user_passport_id),
    CONSTRAINT user_passport_idfk FOREIGN KEY (user_passport_id)
        REFERENCES public.passport (passport_id) MATCH SIMPLE
        ON UPDATE CASCADE
        ON DELETE CASCADE
)

TABLESPACE pg_default;

ALTER TABLE public.users
    OWNER to postgres;
```

Таблица Passport

```
CREATE TABLE public.passport
(
    passport_id integer NOT NULL,
    passport_name character varying(50) COLLATE pg_catalog."default",
    passport_surname character varying(50) COLLATE pg_catalog."default",
    passport_birthday date,
    CONSTRAINT passport_idpk PRIMARY KEY (passport_id)
)

TABLESPACE pg_default;

ALTER TABLE public.passport
    OWNER to postgres;
```

Вміст таблиць в pgAdmin 4

Таблиця Article

	article_id [PK] integer	article_title character varying (20)	article_hashtag character varying (10)	article_theme integer	article_text text
1	1	Lakers for title	#nba	2	Text of article 1
2	2	ESL in 2020	#valve	1	Text of article 2
3	3	Best 2019 game	#control	1	Text of article 3
4	4	Film of decade	#moonlight	3	Text of article 4
5	5	Greatest comeback	#UCL	2	Text of article 5

Таблиця Commentaries

	comment_id [PK] integer	comment_text text	comment_date date	comment_to_article integer	comment_from_user integer
1	1	Comment #1	2019-01-01	2	1
2	2	Comment #2	2020-04-15	1	1
3	3	Comment #3	2019-11-03	4	3
4	4	Comment #4	2020-09-19	3	2

Таблиця Readby

	read_id [PK] integer	user_read_id integer	article_read_id integer
1	1	2	3
2	2	1	4
3	3	3	1
4	4	2	1
5	5	1	2

Таблиця Theme

	theme_id [PK] integer	theme_name character varying (20)
1	1	Games
2	2	Sport
3	3	Films

Таблиця Users

	user_id [PK] integer	user_login character varying (20)	user_email character varying (40)	user_reg_date date	user_passport_id integer
1	1	jake_lamar	jake11@gmail.com	2018-09-28	2
2	2	icestorm	icestorm@ukr.net	2017-01-13	1
3	3	romioblack	romio21@i.ua	2018-11-03	3

Таблица Passport

	<div>passport_id</div> <div>[PK] integer</div>	<div>passport_name</div> <div>character varying (50)</div>	<div>passport_surname</div> <div>character varying (50)</div>	<div>passport_birthday</div> <div>date</div>
1	1	Petro	Ivanov	1998-02-12
2	2	Jame	Kellow	1995-06-09
3	3	Emil	Petersen	1996-07-15