

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 та засоби управління»**

Виконав: студент групи КВ-11

Угнівенко Ярослав

Контакт в telegram: @yar4ik4ik

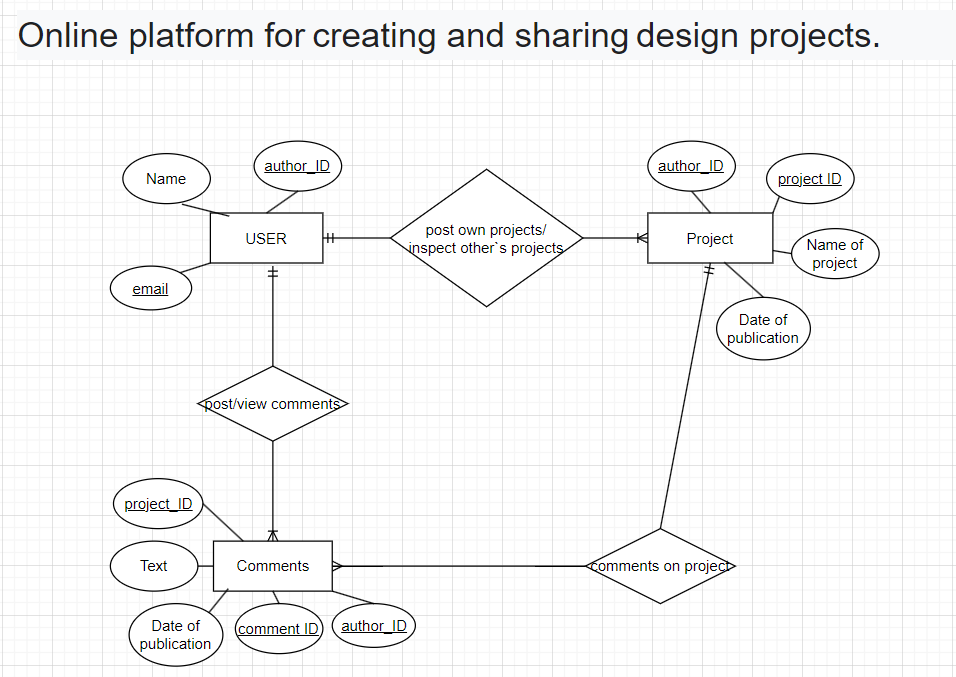
Київ

2023

**ПУНКТ №1**

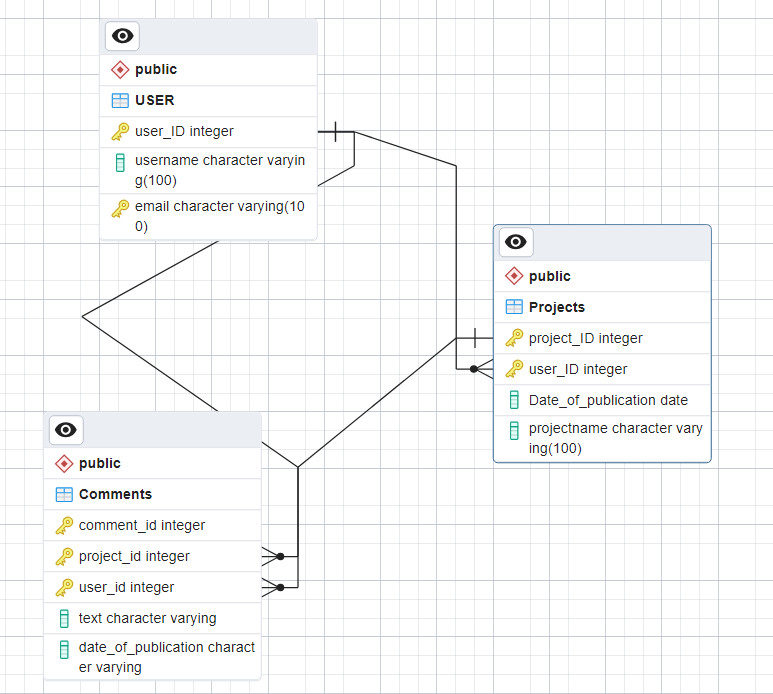
Для даної роботи було вибрано тему “ Онлайн-платформа для зберігання та обміну дизайнерськими проектами”. Тут наявні 3 сутності: User, Project і Comments. Перша сутність – User, має 3 атрибути: нік і ключові – ID, електронна пошта. Пошта є ключовими, для того, щоб на сайт можна було заходити за її допомоги, а також, щоб не було повторень і не можна було зареєструвати більше 1 аккаунта на одну пошту. ID – унікальний ключ, який присвоюється Usery у внутрішній базі і потрібен для його ідентифікації. Ім’я не є унікальним. Друга сутність – Project, має 4 атрибути: ім’я проекту, дата публікації проекту, ID проекту, який є ключовим і по ньому можна здійснити пошук саме цього проекту, а також зовнішній ключ – author\_ID. Остання сутність – Comments, в якій є атрибути тексту(сам коментар), дата публікації і 3 ключових поля, 2 з яких – зовнішні, а саме є comment\_id, який є унікальним полем для самої сутності, author\_id, який посилається на автора коментаря і project\_id, який посилається на проект, до якого доданий коментар.

Нотація “Чена”:



**ПУНКТ №2**

Схема згенерованa pgadmin4:



Як і було описано вище, маємо 3 сутності. Реалізація була придумана просто, існує User(далі – користувач), Projects(далі – проект) і Comments(далі – коментарі), для початку існує 1 користувач, який може мати багато проектів, після чого іде коментар під проектом, тобто під одним проектом може бути кілька коментарів, як від одного, так і від різних користувачів.

Отже, таблиця(фактично сутність) USER має зв’язок з іншими двома таблицями, як один до багатьох(1 користувач – кілька коментарів і проектів), таблиця Projects – має зв’язок багато до одного з таблицею USER і один до багатьох з таблицею Comments, а остання має зв’язки багато до одного з іншими таблицями. Всі вони пов’язані між собою за ключовими полями-ідентифікаторами, також є унікальні поля, у USER – user\_id, Projects – projects\_id, Comments – comment\_id. Можна побачити, що user\_id є в кожній сутності, але унікальним він являється тільки для сутності USER, всі інші просто наслідують його значення. Також у таблицях присутні неключові поля, такі, як назви і дати.

**ПУНКТ №3**

*Функціональні залежності:*

**USER**(*author\_id*, Name, email)

author\_id -> Name, email

author\_id -> Name

author\_id -> email

**Projects**(*project\_id*, *author\_id*, Name\_of\_project, Date\_of\_publication)

project\_id-> Name\_of\_project, Date\_of\_publication

project\_id-> Date\_of\_publication

project\_id-> Name\_of\_project

author\_id(foreign key), author\_id(USER) -> author\_id(Projects)

**Comments**(*comment\_id*, *project\_id*, *author\_id,* Text, Date\_of\_publication)

comment\_id-> Text, Date\_of\_publication

comment\_id-> Date\_of\_publication

comment\_id-> Text

author\_id(foreign key), author\_id(USER) -> author\_id(Comments)

project\_id(foreign key), project\_id(Projects) -> project\_id(Comments)

*Перша нормальна форма (1NF):*

Схема бази даних відповідає 1NF, оскільки кожен атрибут у реляційній таблиці має атомарне значення, тобто він не може бути масивом, списком або ієрархічною структурою.

*Друга нормальна форма (2NF):*

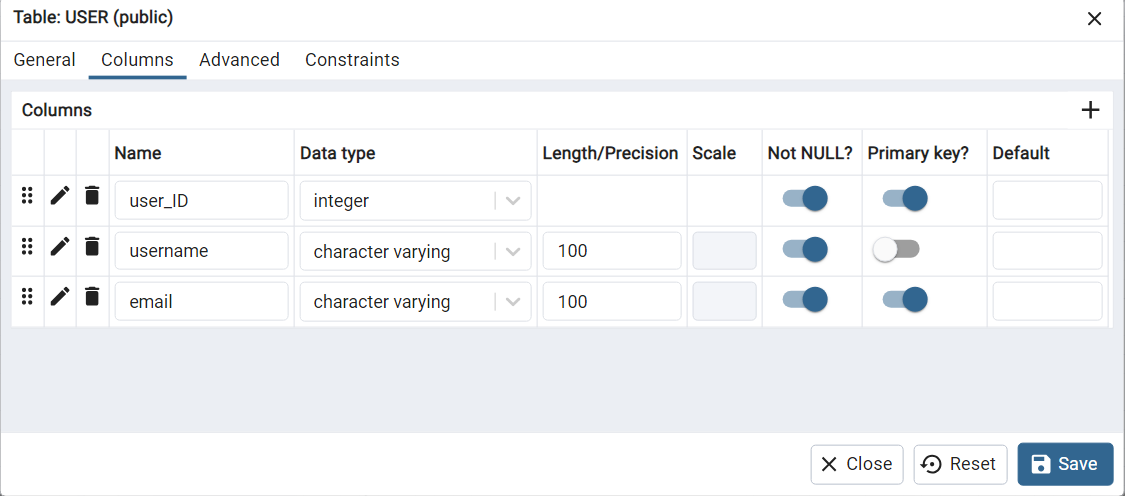
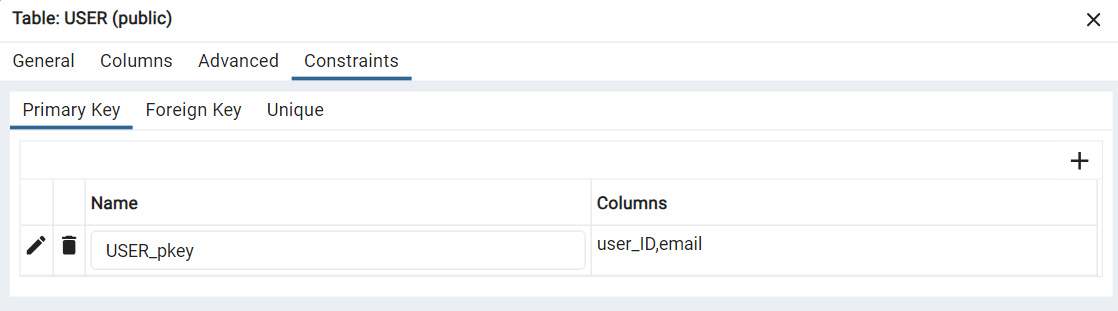
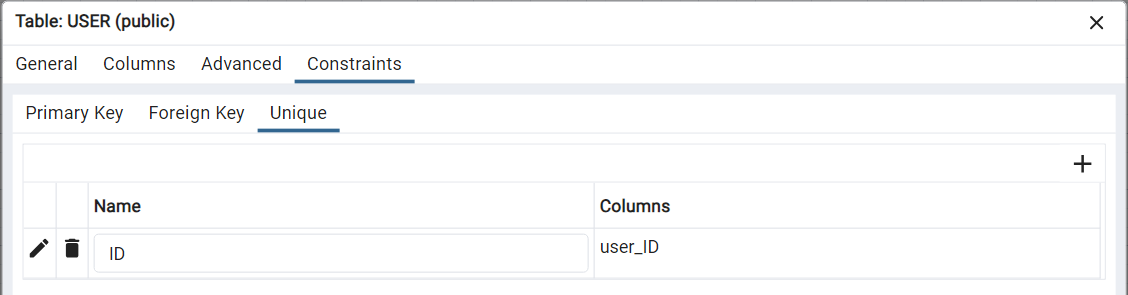
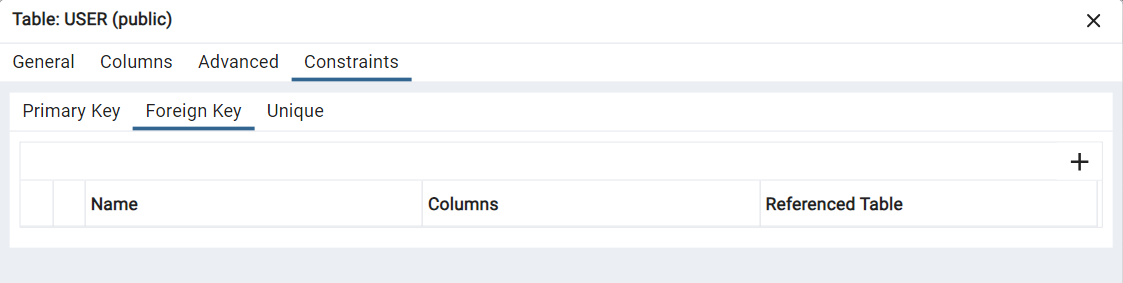
Схема бази даних відповідає 2NF, оскільки таблиця відповідає 1NF і кожен атрибут, що не входить в первинний ключ, повністю залежить від всіх атрибутів первинного ключа.

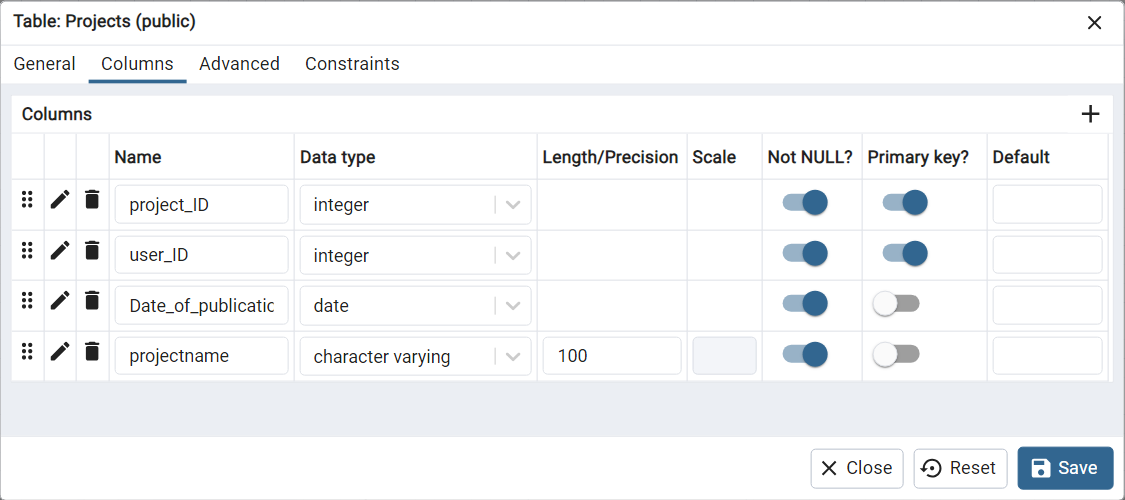
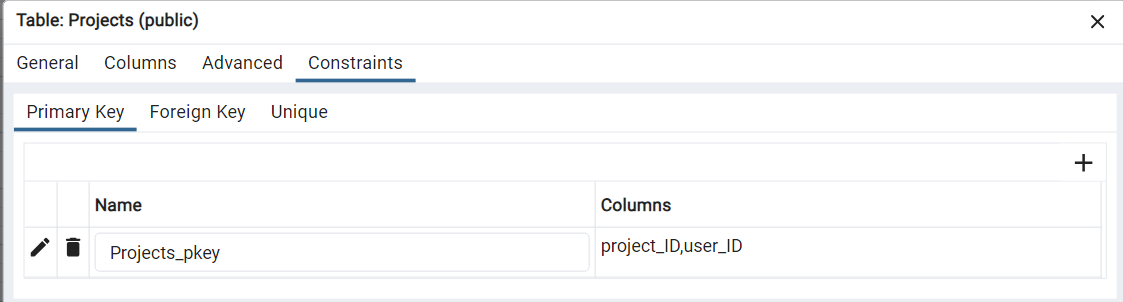
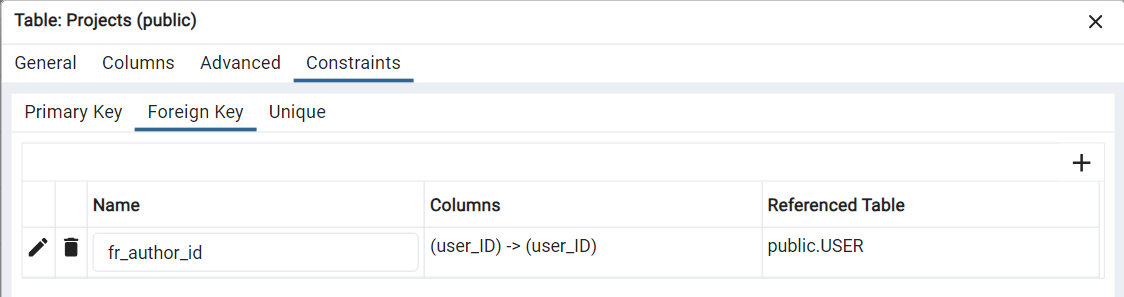
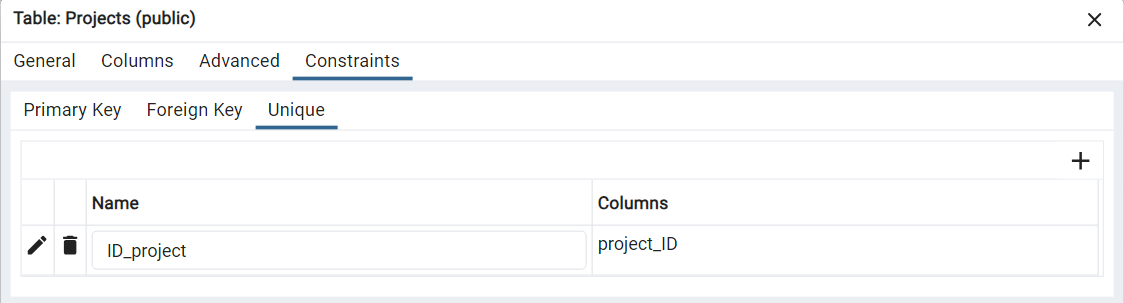
*Третя нормальна форма (3NF):*

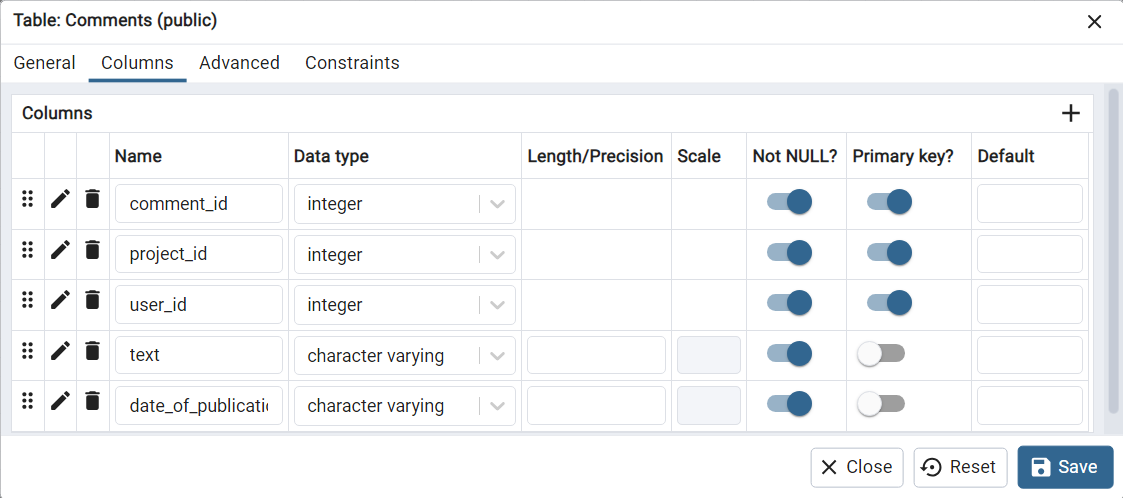
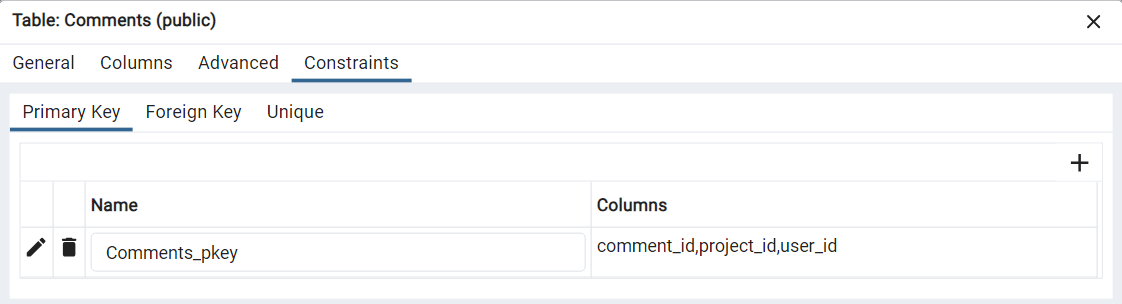
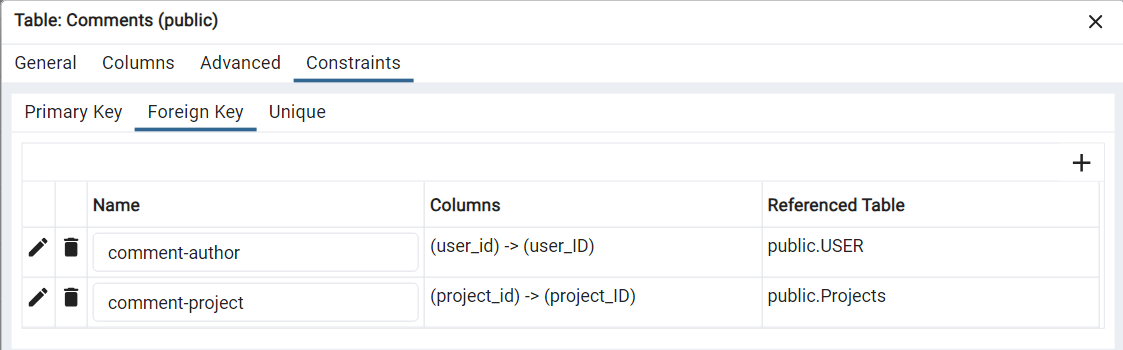
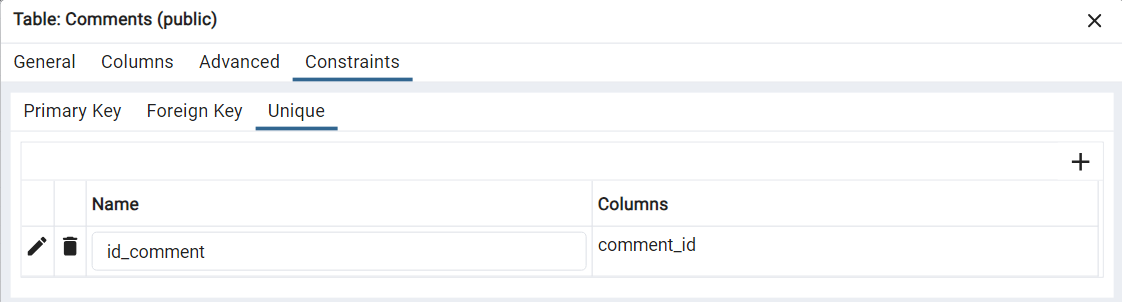
Схема бази даних відповідає 3NF, оскільки таблиця відповідає 2NF і кожен атрибут, що не входить в первинний ключ і не є частиною іншого атрибута (тобто атрибут не є транзитивно залежним від первинного ключа), повністю залежить від первинного ключа.

**ПУНКТ №4**

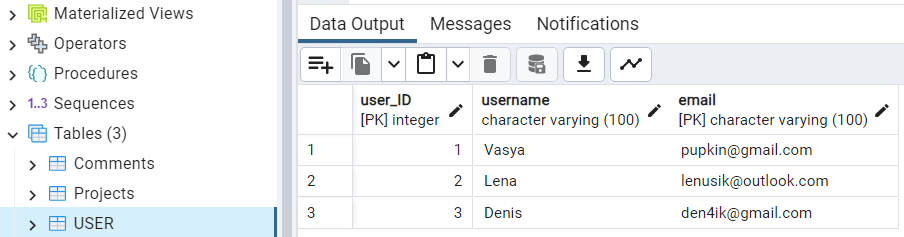
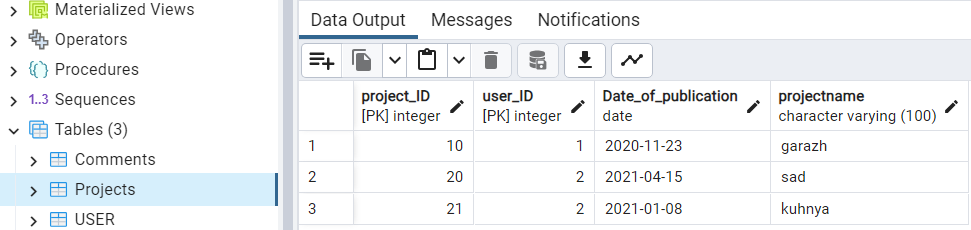
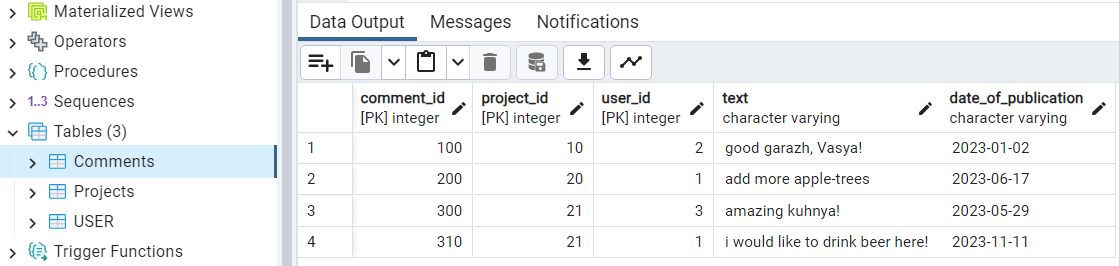
Копії екрану з pgAdmin4:

Дані в таблицях:

Посилання на гіт: [yaro-krasav4ik228/DataBases: dlya laboratornyh robit (github.com)](https://github.com/yaro-krasav4ik228/DataBases/)