

Міністерство освіти і науки України
Національний технічний університет України
«Київський політехнічний інститут»

Лабораторна робота №1
«Проектування бази даних та ознайомлення з
базовими операціями СУБД PostgreSQL»

Виконав студент групи: КВ-32

ПІБ: Клубук Максим Віталійович

Перевірив: _____

Мета роботи

здобуття вмінь проектування бази даних та практичних навичок створення реляційних баз даних за допомогою PostgreSQL

Варіант

Електронний журнал для ведення особистого щоденника.

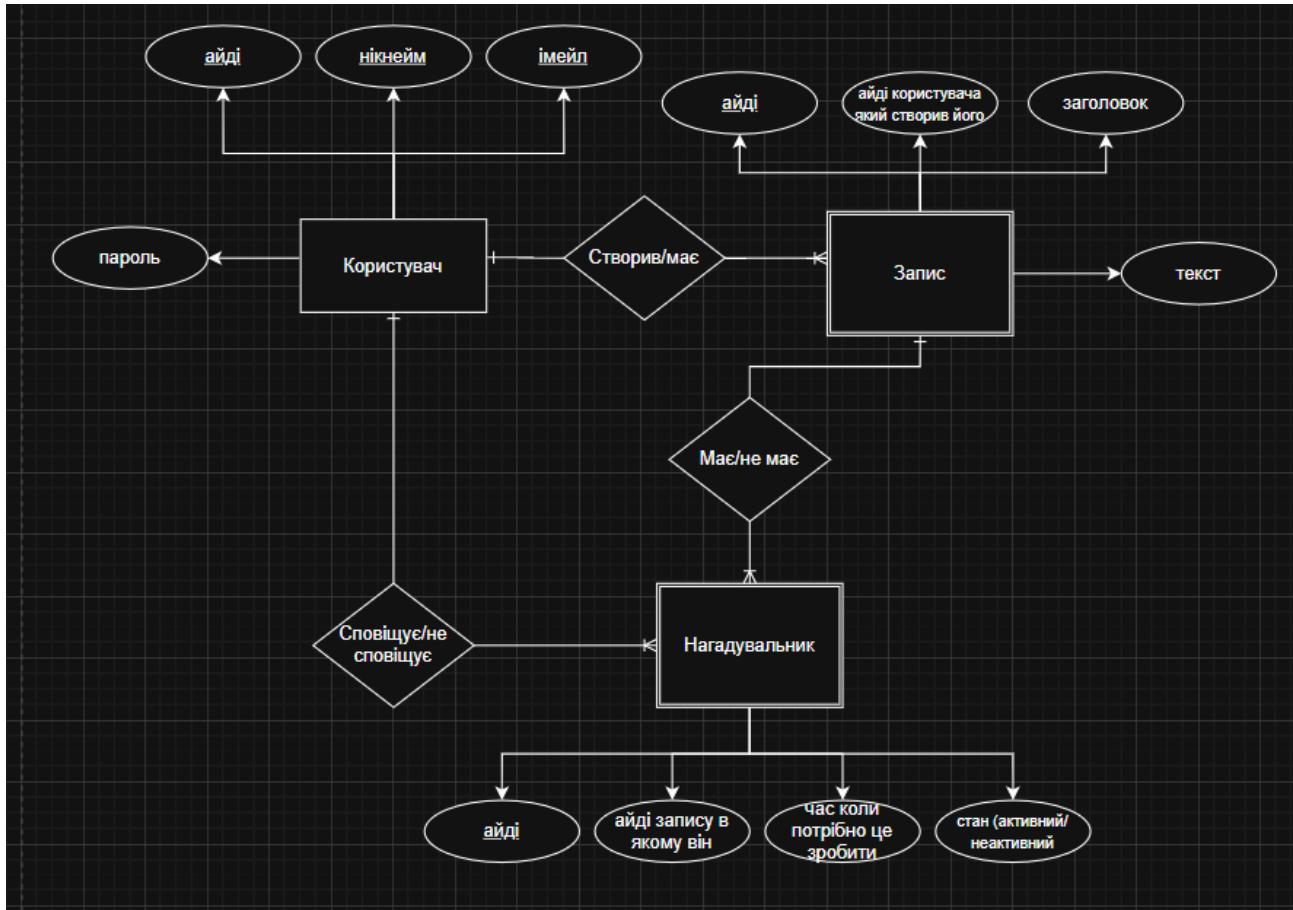
Хід роботи

- 1. Розробити модель «сущність-зв’язок» предметної галузі, обраної студентом самостійно, відповідно до пункту «Вимоги до ER-моделі».**

Для початку роботи розберемо всю ситуацію. В нас є користувач, який веде щоденник, тобто може робити багато записів в нього. Це вже дві сутності, сутність «Користувач» та сутність «Запис». Також непогано було б додати нагадувальник для виконання певних записів, наприклад я, як користувач, зробив запис «купити молоко в магазині» і хочу додати нагадування на певний час, коли це зробити, якщо забуду. Тому, додаємо також сутність «Нагадувальник», яка буде прив’язана до певного запису та сповіщати користувача про його обраний запис.

Тепер розберемо атрибути, «користувач» буде складатися з: ідентифікатор (id), ім’я (username), поштової скриньки (email), пароля (password). Сутність «запис» буде складатися з: ідентифікатора (entry_id), ідентифікатор користувача до якого він буде прив’язаний (user_id), заголовка (title), контексту або вмісту запису (text). І ще в нас є «Нагадувальник», тому він буде мати: номер (reminder_id), номер запису в якому він знаходиться (entry_id), час коли сповістити (remind_at), та його стан тобто увімкнений чи вимкнений (active).

Загалом, графічно це виглядатиме ось так:



Скріншот з сайту для розробки діаграм

В овалах описані атрибути, в прямокутниках сущності а ромби – відносини між сущностями. Один користувач має скільки завгодно записів, тобто тут в нас 1:N. Один запис може мати багато нагадувальників, тому також 1:N. Так само і з нагадувальником.

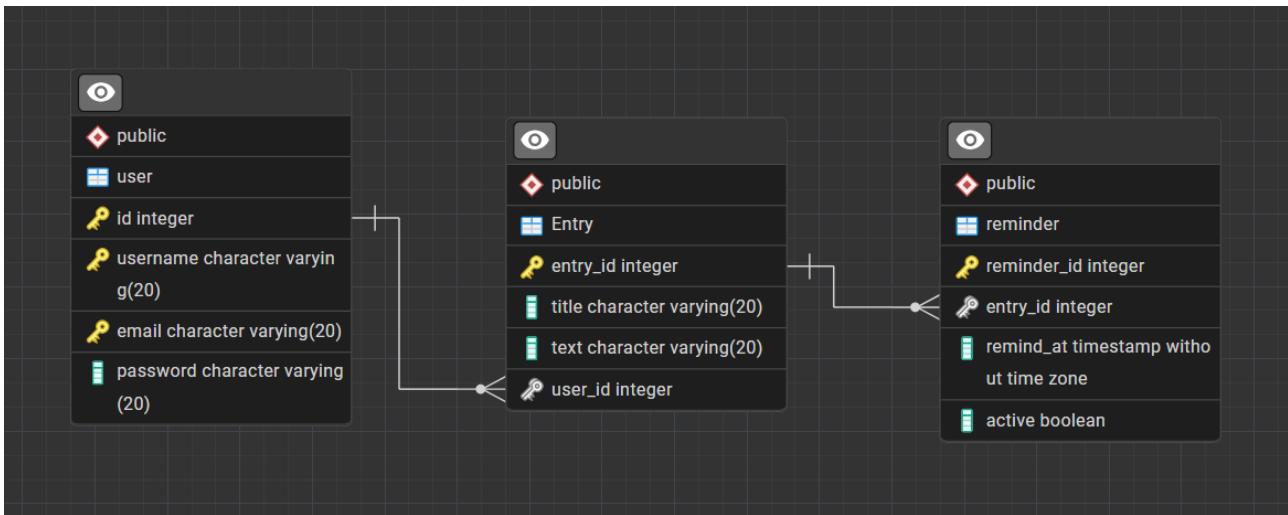
2. Перетворити розроблену модель у схему бази даних (таблиці) PostgreSQL.

Перейдемо до середовища pgAdmin4. Відкриємо Tools -> ERD Tool та почнемо створювати таблиці. Почнемо з користувача, будемо писати все латиницею для зручності в користуванні в подальшому. Тому створюємо:

Сущність «Користувач» було перетворено в таблицю «user», де атрибутами стали: ідентифікатор (id, integer, PK, UNIQUE, NOT NULL), псевдонім (username, var char, PK, UNIQUE, NOT NULL), поштова скринька (email, var char, PK, UNIQUE, NOT NULL), пароль (password, var char, NOT NULL).

Сущність «Запис» було перетворено в таблицю «entry», де атрибутами стали: ідентифікатор (entry_id, integer, PK, UNIQUE, NOT NULL), заголовок (title, var char), текст (text, var char), посилання на ідентифікатор користувача (user_id, integer, FK, NOT NULL).

Сущність «Нагадувальник» було перетворено в таблицю «reminder», де атрибутами стали: ідентифікатор (reminder_id, integer, PK, UNIQUE, NOT NULL), посилання на ідентифікатор запису (entry_id, integer, FK, NOT NULL), час коли потрібно нагадати (remind_at, timestamp with time zone), активний чи деактивований стан (active, Boolean, NOT NULL).



3. Виконати нормалізацію схеми бази даних до третьої нормальної форми (3НФ).

Функціональні залежності:

User: id \rightarrow username, email, password.

id \rightarrow username.

id \rightarrow email.

id \rightarrow password.

Entry: entry_id \rightarrow title, text, user_id.

entry_id \rightarrow title.

entry_id \rightarrow text.

entry_id \rightarrow user_id.

Reminder: reminder_id \rightarrow entry_id, remind_at, active.

reminder_id \rightarrow entry_id.

reminder_id \rightarrow remind_at.

reminder_id \rightarrow active.

Перевірка на 1НФ:

кожне поле містить **атомарні значення** (нема масивів чи списків),

рядки унікальні,

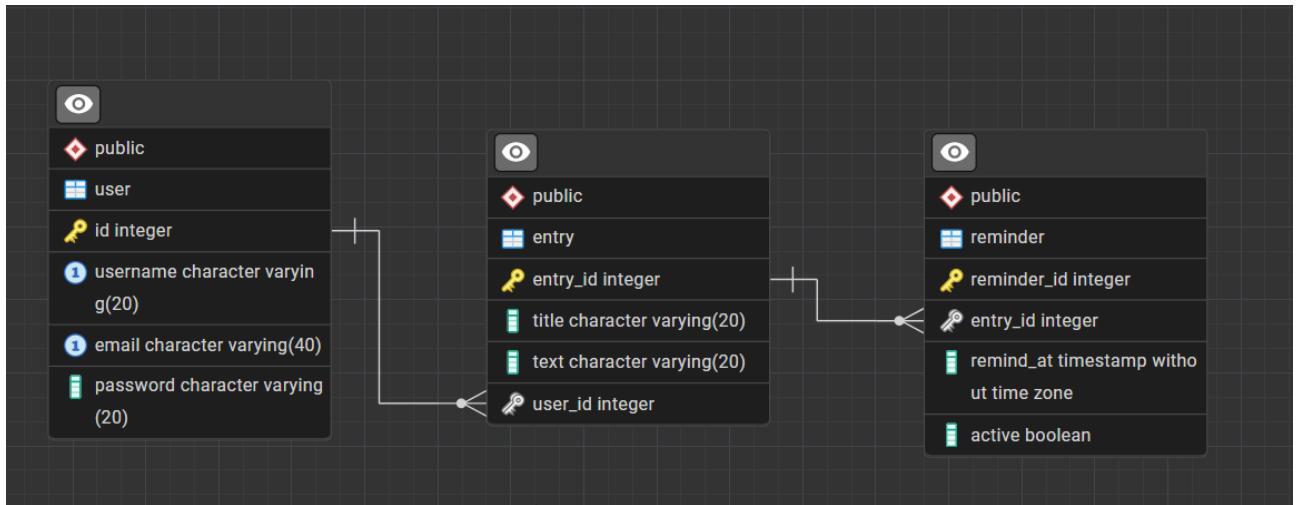
є **первинний ключ**.

В створених таблицях (user, entry, reminder) є прості поля типу varchar, integer, timestamp, boolean, також кожна таблиця має **первинний ключ**, та рядки унікальні, що означає, що 1НФ виконується.

Перевірка на 2НФ:

нема часткової залежності атрибутів від частини складного ключа.

У таблиці user був **PRIMARY KEY (username, email, id)**. Це погано, бо складний ключ створює ризик часткової залежності. Тому переробимо user, знявши галочку на PRIMARY KEY в атрибутах email та username.



Перевірка на ЗНФ:

нема транзитивних залежностей (атрибут залежить від РК через інший атрибут).

Перевіримо:

user: id -> username, email, password. Нормально.

entry: entry_id -> title, text, user_id. Все ок.

reminder: reminder_id -> entry_id, remind_at, active. Все ок.

4. Ознайомитись із інструментарієм PostgreSQL та pgAdmin 4 та внести декілька рядків даних у кожну з таблиць засобами pgAdmin 4.

user

General Columns Advanced Constraints Partitions Parameters Security SQL

Inherited from table(s) Select to inherit from...

Columns

	Name	Data type	Length/Precision	Scale	Not NULL?	Primary key?	Default
	id	integer			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
	username	character varying	20		<input checked="" type="checkbox"/>	<input type="checkbox"/>	
	email	character varying	40		<input checked="" type="checkbox"/>	<input type="checkbox"/>	
	password	character varying	20		<input checked="" type="checkbox"/>	<input type="checkbox"/>	

Close Reset Save

user

General Columns Advanced Constraints Partitions Parameters Security SQL

Primary Key Foreign Key Check Unique Exclude

Constraints

Name	Columns	
	id	id

Close Reset Save

user

General Columns Advanced Constraints Parameters Security SQL

Primary Key Foreign Key Check Unique Exclude

Name	Columns
email	email
username	username

Close Reset Save

entry

General Columns Advanced Constraints Partitions Parameters Security SQL

Inherited from table(s) Select to inherit from...

Name	Data type	Length/Precision	Scale	Not NULL?	Primary key?	Default
entry_id	integer	▾		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
title	character varying	▾ 20		<input type="checkbox"/>	<input type="checkbox"/>	'No title'
text	character varying	▾ 20		<input type="checkbox"/>	<input type="checkbox"/>	'No text'
user_id	integer	▾		<input type="checkbox"/>	<input type="checkbox"/>	

Close Reset Save

entry

General Columns Advanced Constraints Partitions Parameters Security SQL

Primary Key Foreign Key Check Unique Exclude

Name entry_pkey Columns entry_id

Close Reset Save

entry

General Columns Advanced Constraints Partitions Parameters Security SQL

Primary Key Foreign Key Check Unique Exclude

Name entry_user_id_fkey Columns (user_id) -> (id) Referenced Table public.user

Close Reset Save

reminder

General Columns Advanced Constraints Partitions Parameters Security SQL

Inherited from table(s) Select to inherit from...

Columns

Name	Data type	Length/Precision	Scale	Not NULL?	Primary key?	Default
reminder_id	integer	1		On	On	
entry_id	integer	1		On	Off	
remind_at	timestamp without time zone	1		On	Off	
active	boolean	1		On	Off	

i ? X Close ⌂ Reset ⌂ Save

reminder

General Columns Advanced Constraints Partitions Parameters Security SQL

Primary Key Foreign Key Check Unique Exclude

Constraints

Name	Columns
reminder_pkey	reminder_id

i ? X Close ⌂ Reset ⌂ Save

reminder

General Columns Advanced Constraints Partitions Parameters Security SQL

Primary Key Foreign Key Check Unique Exclude

Name	Columns	Referenced Table
reminder_entry_id_fkey	(entry_id) -> (entry_id)	public.entry

i **?** **X Close** **Reset** **Save**

Materialized Views

Operators

Procedures

1..3 Sequences

Tables (3)

- entry**
- reminder**
- user**

Trigger Functions

Data Output **Messages** **Notifications**

	id [PK] integer	username character varying (20)	email character varying (40)	password character varying (20)
1	2	Allan	allan2@gmail.com	3333
2	1	Steve	steve@email.com	1234

Materialized Views

Operators

Procedures

1..3 Sequences

Tables (3)

- entry**
- reminder**
- user**

Data Output **Messages** **Notifications**

	entry_id [PK] integer	title character varying (20)	text character varying (20)	user_id integer
1	3	Allans entry	This is my entry	2
2	1	[default]	[default]	1
3	2	dva	gig	1

Operators

Procedures

1..3 Sequences

Tables (3)

- entry**
- reminder**
- user**

Data Output **Messages** **Notifications**

	reminder_id [PK] integer	entry_id integer	remind_at timestamp without time zone	active boolean
1	1	2	2025-09-18 19:00:00	true
2	2	3	2025-09-19 19:00:00	false

Посилання на репозиторій Github:

<https://github.com/reinlaught/data-bases-kpi>