**Лабораторна робота № 1**

**“Структура пекарні”**

***з дисципліни***

**“БАЗИ ДАНИХ ТА ЗАСОБИ УПРАВЛІННЯ”**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Група: КВ-01**

**Виконала: Бондарчук Марія**

**Лабораторна робота № 1.**

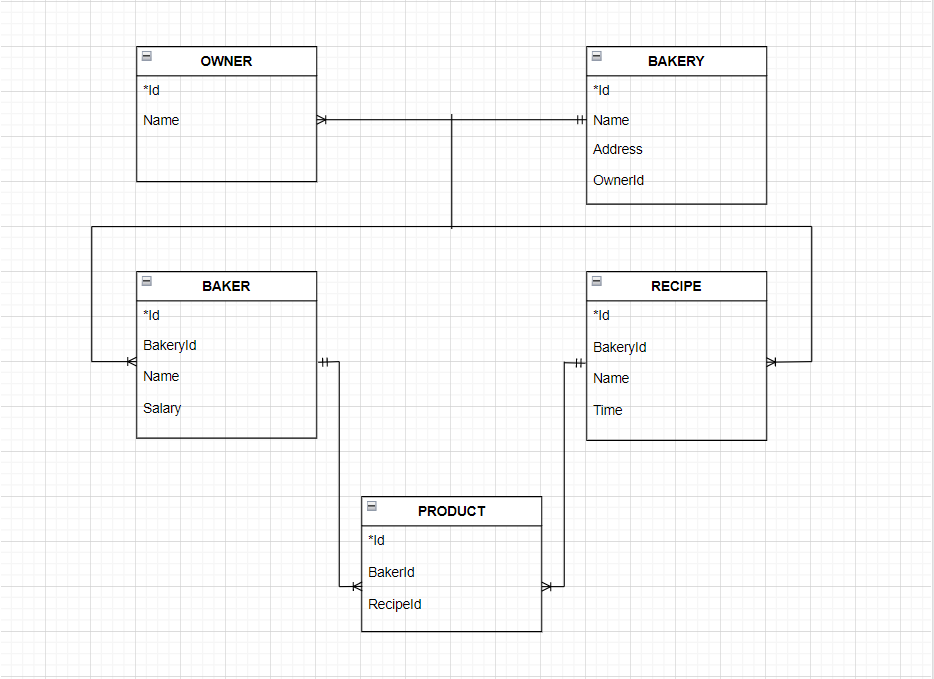
**Проектування бази даних та ознайомлення з базовими операціями СУБД PostgreSQL**

*Метою роботи* є здобуття вмінь проектування бази даних та практичних навичок створення реляційних баз даних за допомогою PostgreSQL.

*Завдання* роботи полягає у наступному:

1. Розробити модель «сутність-зв’язок» предметної галузі, обраної студентом самостійно, відповідно до пункту «Вимоги до ER-моделі».
2. Перетворити розроблену модель у схему бази даних (таблиці) PostgreSQL.
3. Виконати нормалізацію схеми бази даних до третьої нормальної форми (3НФ).
4. Ознайомитись із інструментарієм PostgreSQL та pgAdmin 4 та внести декілька рядків даних у кожну з таблиць засобами pgAdmin 4.

**Пункт 1**

**Модель «сутність-зв’язок»:**   
  


Нотація Пташиної лапки (Crow’s foot)”

План пекарні, згідно якого побудована база даних з сутностями:

1. Власник, з атрибутами:

* Ідентифікаційний номер
* Ім’я

1. Пекарня, з атрибутами:

* Ідентифікаційний номер
* Ідентифікаційний номер власника
* Назва
* Адреса

1. Пекар, з атрибутами:

* Ідентифікаційний номер
* Ідентифікаційний номер пекарні
* Ім’я
* Зарплатня

1. Рецепт, з атрибутами:

* Ідентифікаційний номер
* Ідентифікаційний номер пекарні
* Ім’я
* Інгредієнти

1. Продукт, з атрибутами:

* Ідентифікаційний номер продукту
* Ідентифікаційний номер пекарні
* Ідентифікаційний номер рецепту

**Пункт 2**

**Опис зв’язків:**

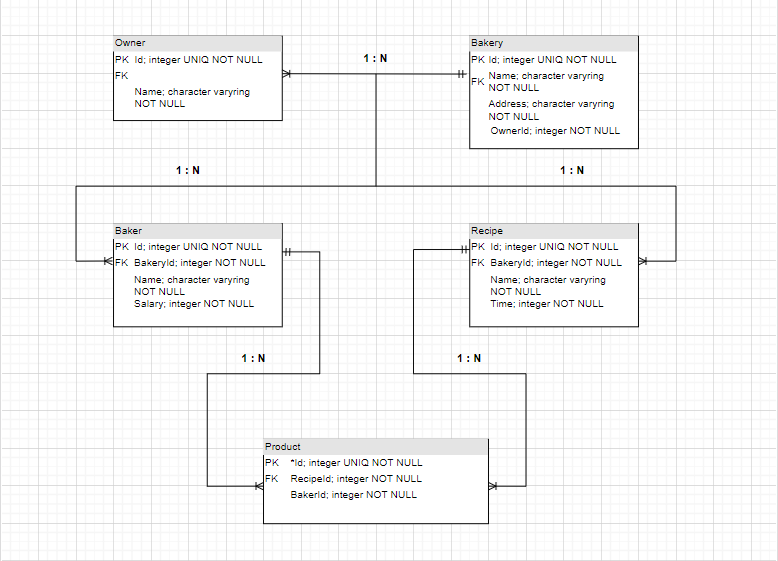
1)Сутність **Bakery** було перетворено в таблицю з назвою **Bakery,** яка має зв’язок 1 : N з **Owner**, а також 1 : N з таблицями **Baker** та **Recipe**, для всіх є головною.

2)Сутність **Owner** було перетворено в таблицю з назвою **Owner,** яка має зв’язок 1 : N з таблицею **Bakery**, є залежною від **Bakery**.

3)Сутність **Recipe** було перетворено в таблицю з назвою **Recipe,** яка має зв’язок 1 : N з таблицею **Bakery**, є залежною від **Bakery** та реалізує зв’язок N : M з таблицею **Baker**, за допомогою таблиці **Product**.

4)Сутність **Baker** було перетворено в таблицю з назвою **Baker,** яка має зв’язок 1 : N з таблицею **Bakery**, є залежною від **Bakery** а також реалізує зв’язок N : M з таблицею **Recipe**, за допомогою таблиці **Product**.

5)Сутність **Product** було перетворено в таблицю з назвою **Product,** яка має зв’язок 1 : N з таблицями **Baker** та **Recipe** і є залежною від них.



|  |  |  |
| --- | --- | --- |
| **СУТНІСТЬ** | **АТРИБУТ** | **ТИП АТРИБУТУ** |
| **OWNER** - містить інформацію про власника | **Id** - унікальний ідентифікатор власника  **Name** – ім’я | **integer**  **character varyring** |
| **BAKERY** - містить інформацію про пекарню | **Id** - унікальний ідентифікатор пекарні  **OwnerId** – ідентифікатор власника  **Name** - назва  **Address** - адреса пекарні | **integer**  **integer**  **character varyring**  **character varyring** |
| **BAKER** - містить інформацію про пекаря | **Id** - унікальний ідентифікатор пекаря  **BakeryId** - ідентифікатор пекарні  **Name** - ім’я  **Salary** - зарплатня пекаря | **integer**  **integer**  **character varyring**  **integer** |
| **RECIPE** - містить інформацію про рецепт | **Id** - унікальний ідентифікатор рецепту  **BakeryId** - ідентифікатор пекарні  **Name** - назва рецепту  **Time** – час на приготування | **integer**  **integer**  **character varyring**  **integer** |
| **PRODUCT** - містить інформацію про готовий продукт | **Id** - унікальний ідентифікатор продукту  **BakerId** - ідентифікатор пекаря  **RecipeId** - ідентифікатор рецепту | **integer**  **integer**  **integer** |

**Пункт 3**  
**Відповідність схеми бази даних нормальним формам:**

**НФ1:**

* кожна «комірка» відношення містить одне значення.
* кожен запис є унікальним.

**НФ2:**

* таблиця в НФ1
* кожен неключовий атрибут функціонально залежати від усього ключа, а не від його частини

**НФ3:**

* таблиця в НФ2
* всі атрибути залежать від первинного ключа, а не від інших атрибутів

**Функціональні залежності:**

**OWNER:**

Id → Name

Id → Name (ім’я власника зв’язане з унікальним ідентифікатором власника)

**BAKERY:**

Id → Name, Address

Id → Name (ім’я пекарні зв’язане з її унікальним ідентифікатором)

Id → Address (адреса пекарні зв’язана з її унікальним ідентифікатором)

Name → Address

Id → Name→ Address (всі атрибути залежать від первинного ключа, а не від інших атрибутів)

Address → Name

Id → Address→ Name (всі атрибути залежать від первинного ключа, а не від інших атрибутів)

**BAKER:**  
Id → Name, Salary

Id → Name (ім’я пекаря зв’язане з його унікальним ідентифікатором)

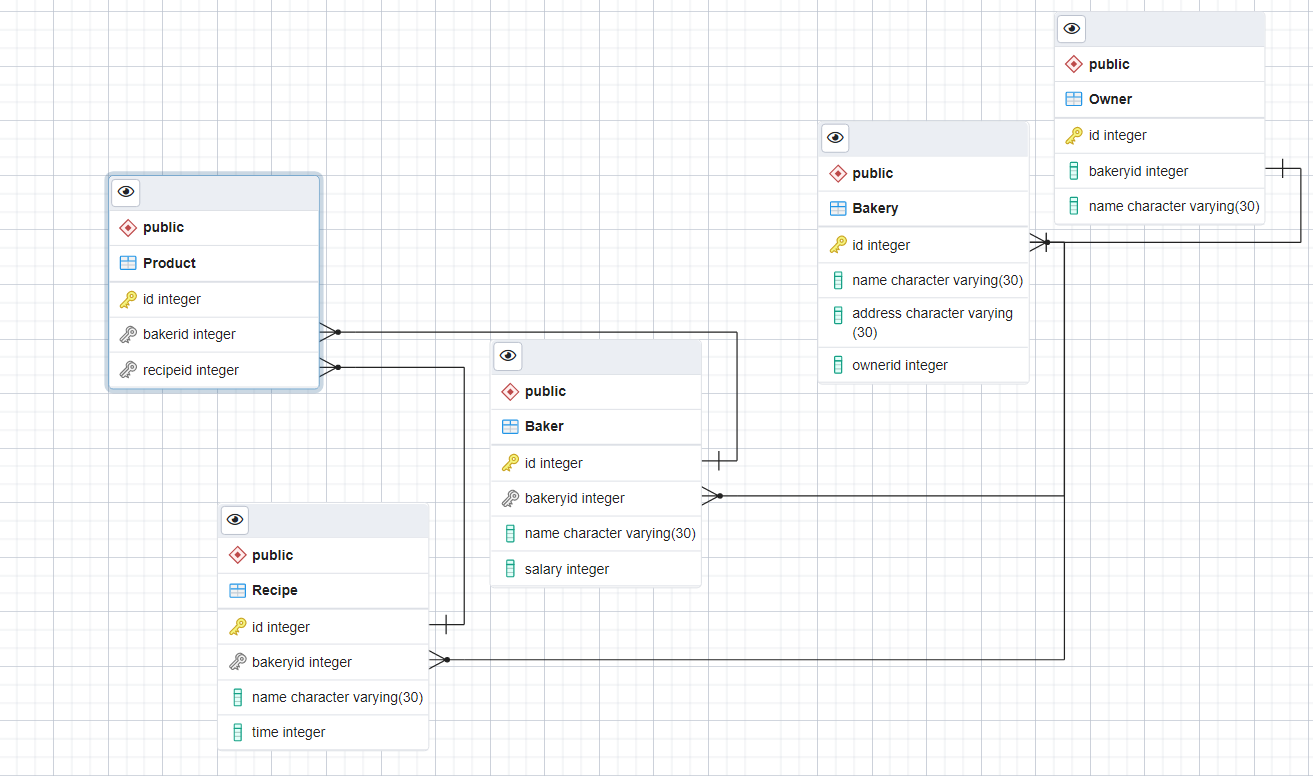
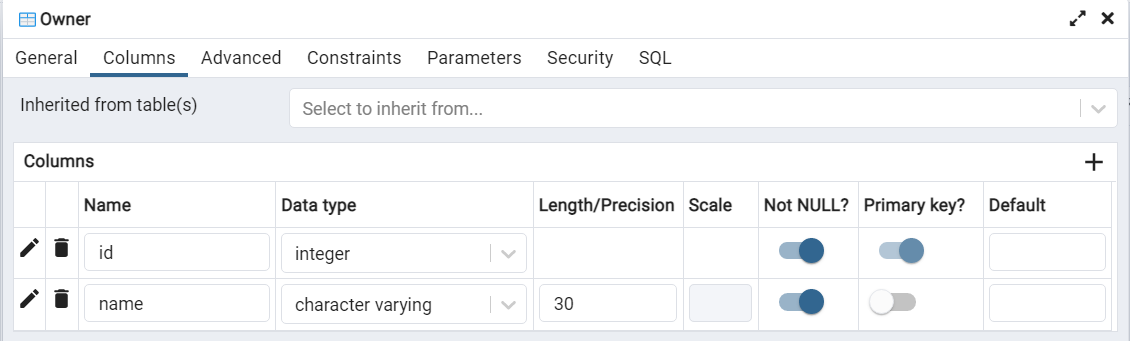
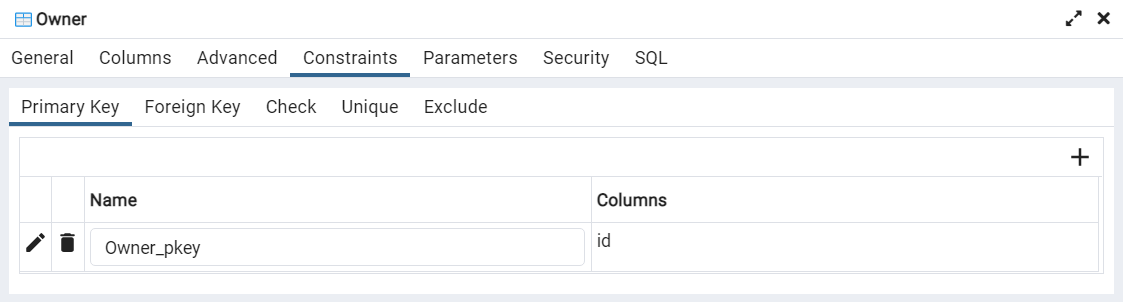
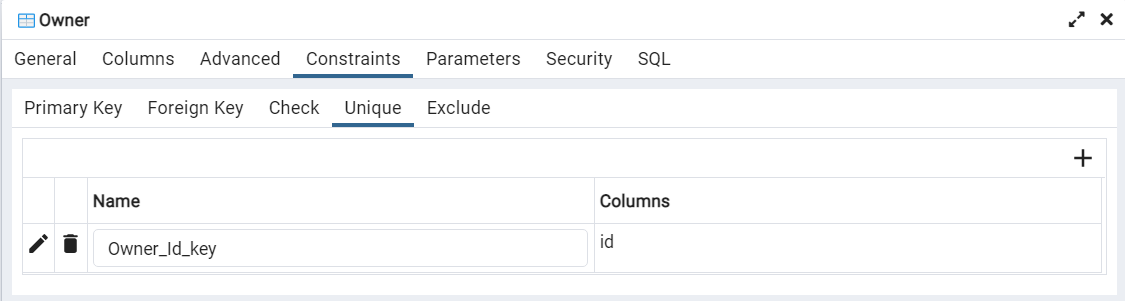
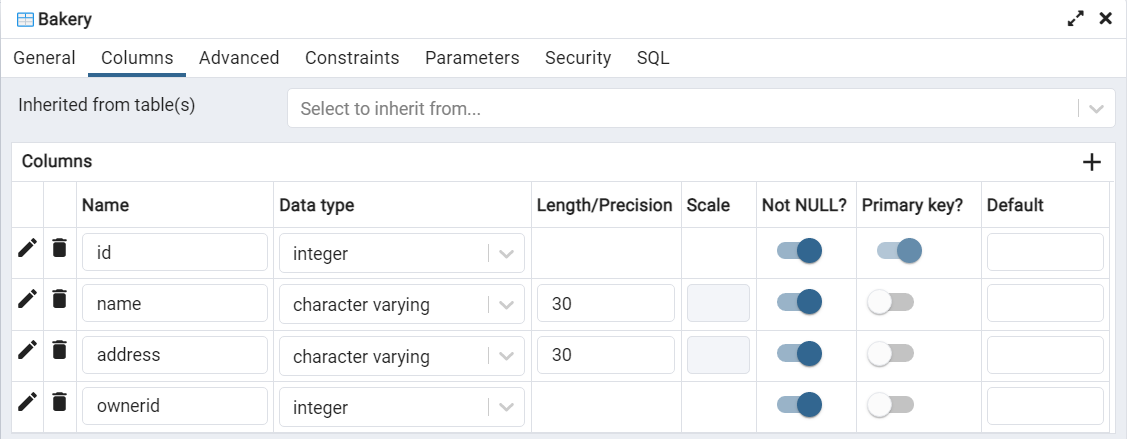
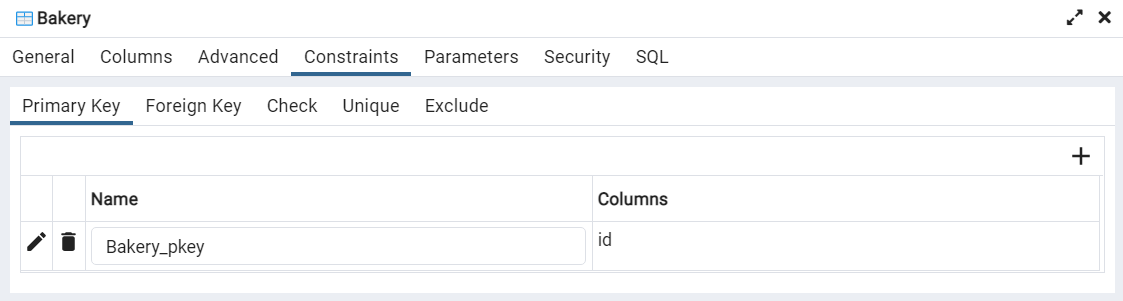
Id → Salary (зарплата пекаря зв’язана з її унікальним ідентифікатором)

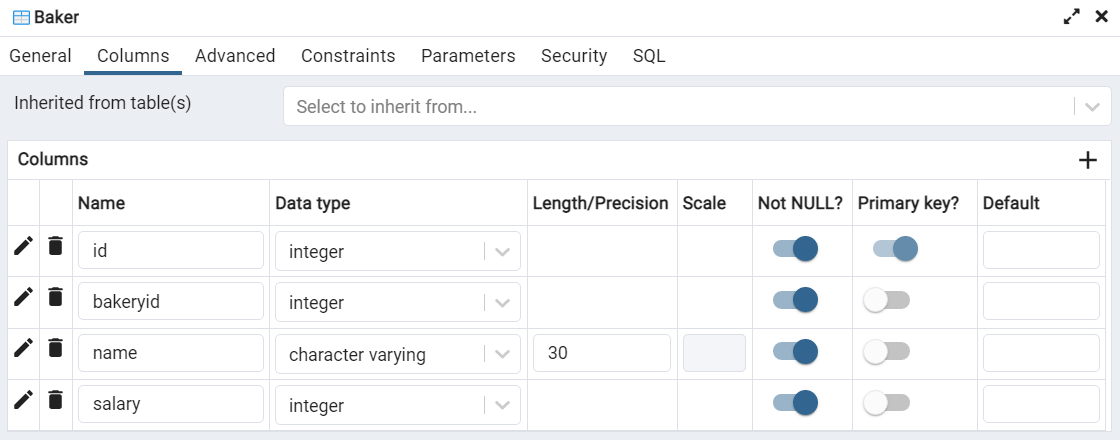
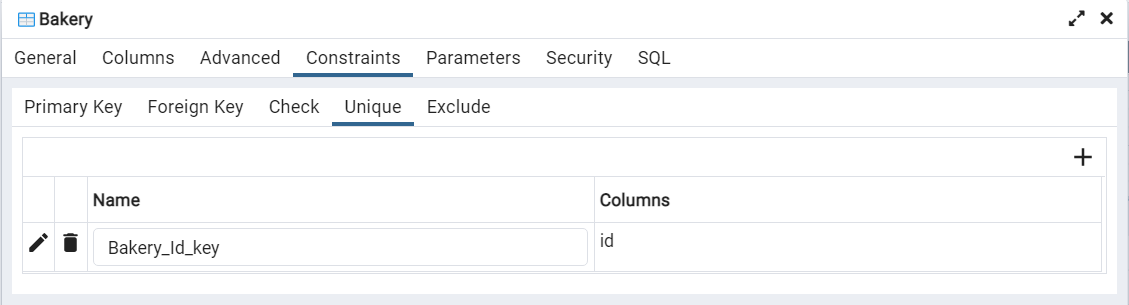
**RECIPE:**

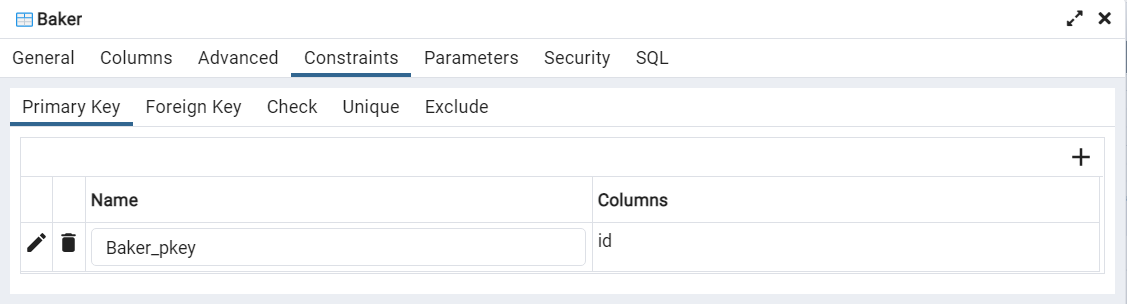
Id → Name, Time

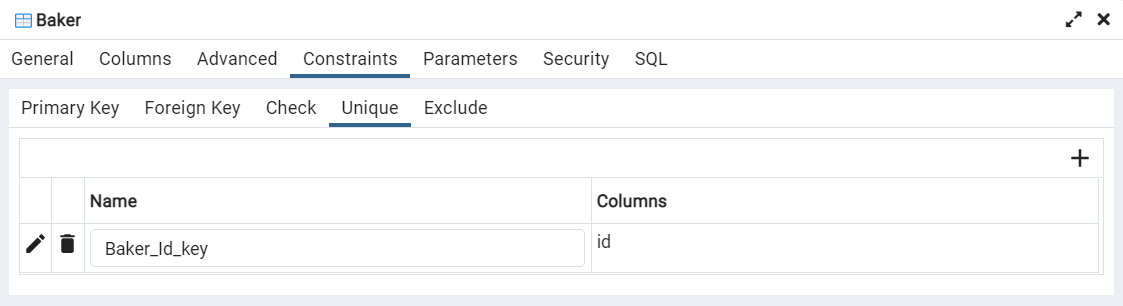
Id → Name (назва рецепту зв’язане з його унікальним ідентифікатором)

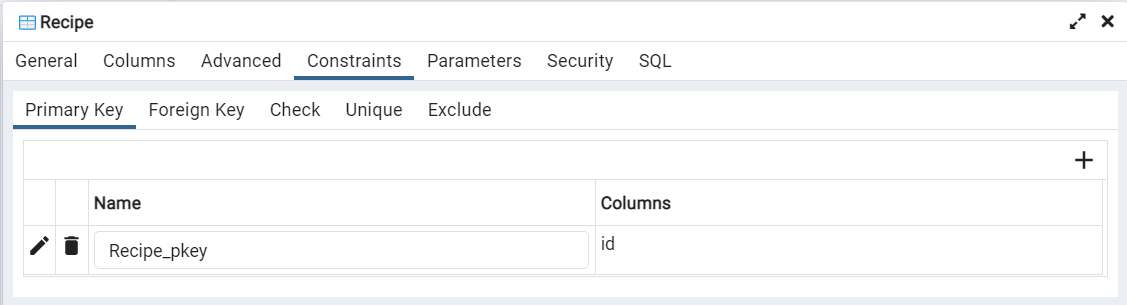
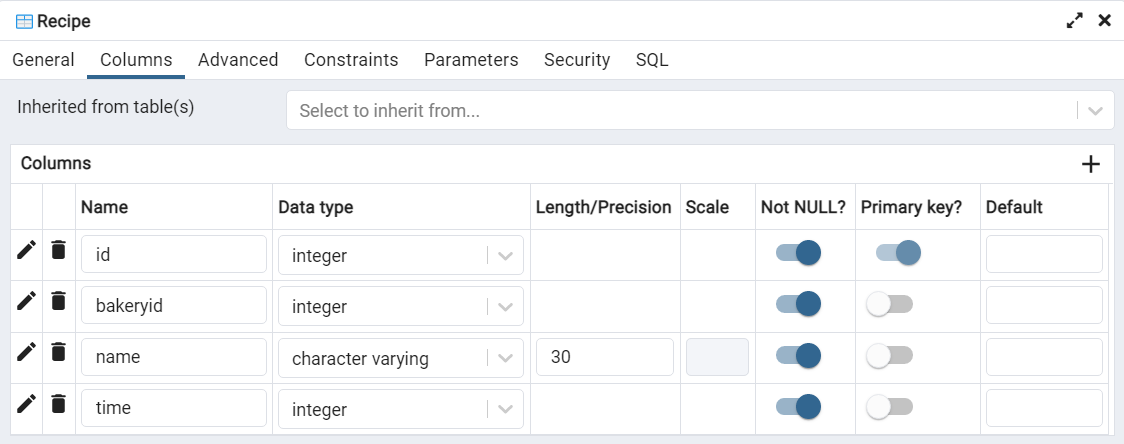
Id → Time (час на виконання рецепту зв’язаний з його унікальним ідентифікатором)

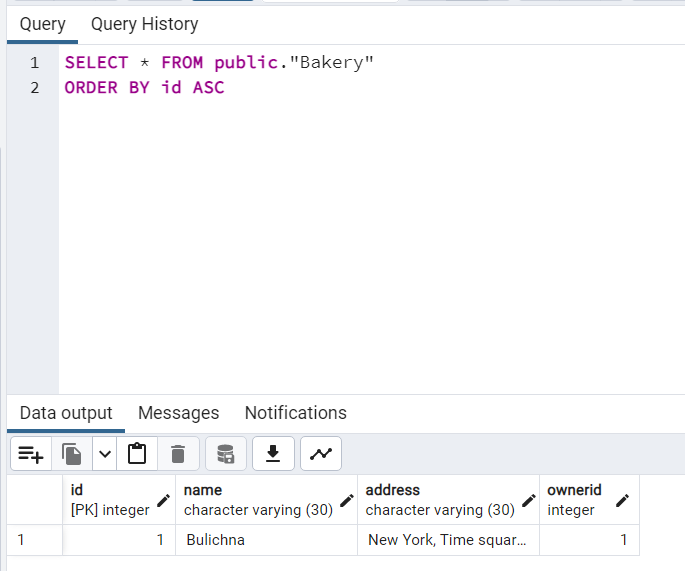
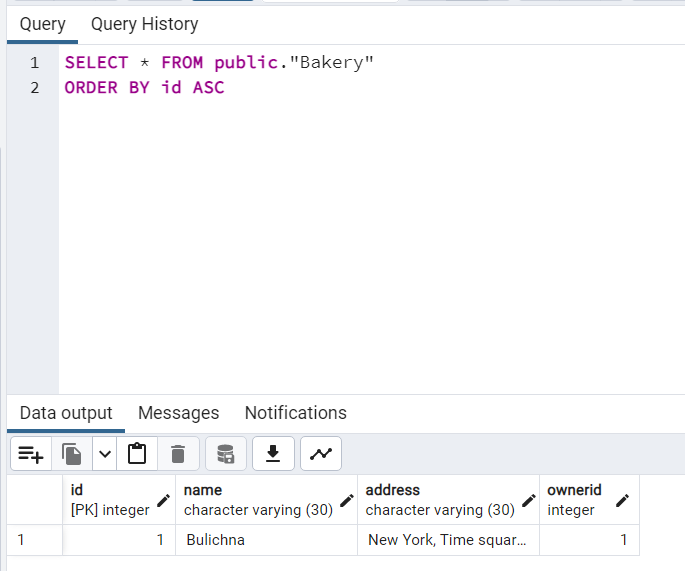
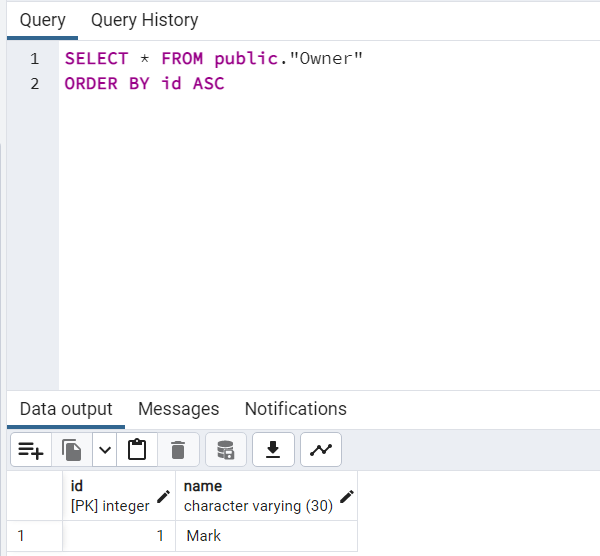
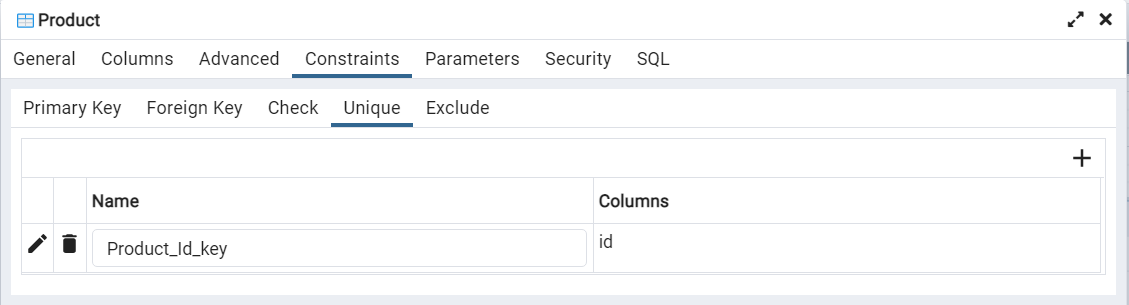
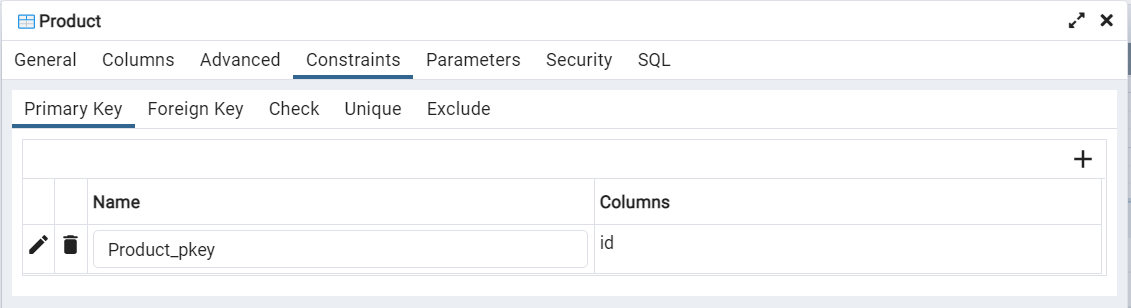
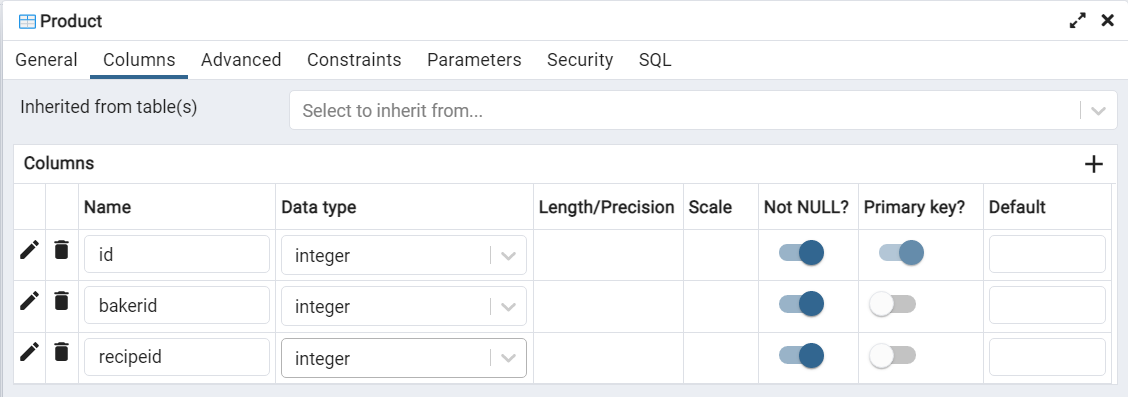
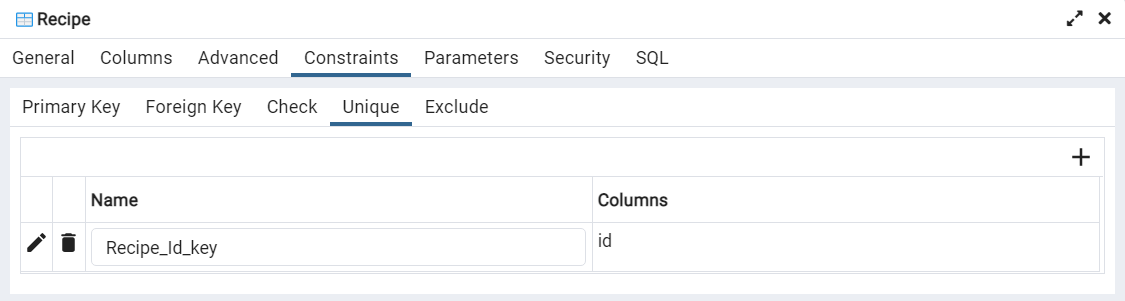
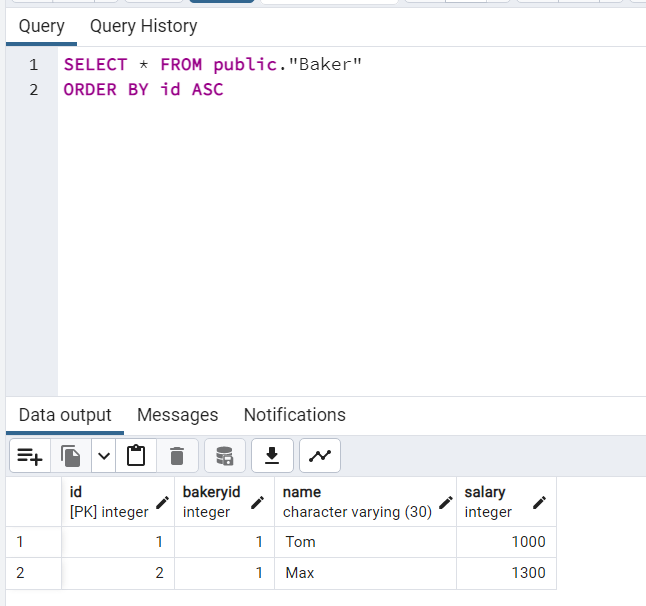
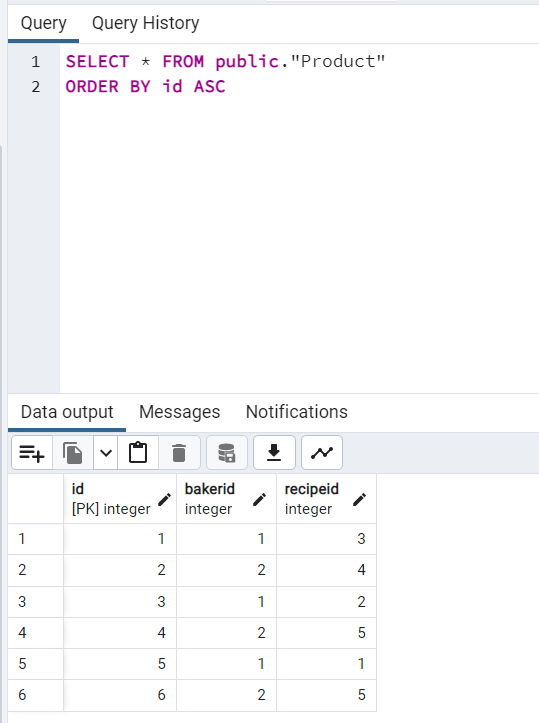
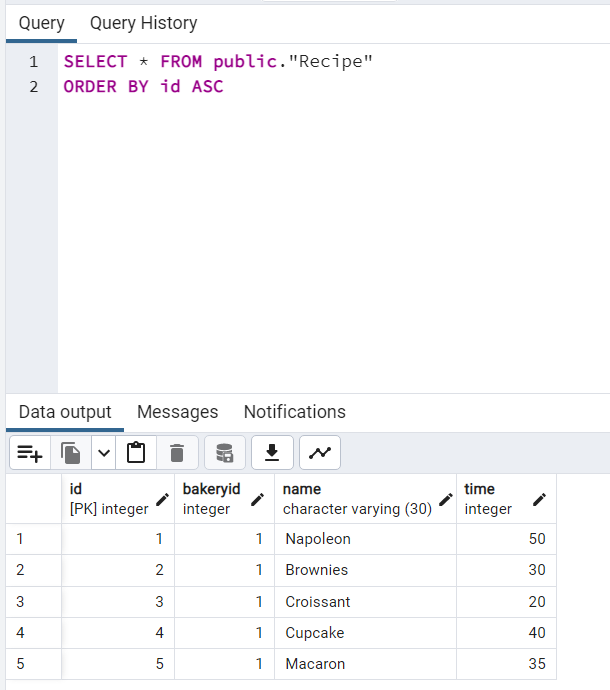
**Пункт 4**  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  










   **Посилання на GitHub:**

[**https://github.com/mariabondarchuk/DB\_Lab\_1**](#_top)