

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**

Виконала студентка групи: КВ-13 ПІБ: Саюн Д.М.

Телеграм: <https://t.me/msdsiu>

**Київ 20**23

*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.

***Завдання 1***

Розроблена модель «сутність-зв’язок» для онлайн-платформи для зберігання та пошуку наукових публікацій.

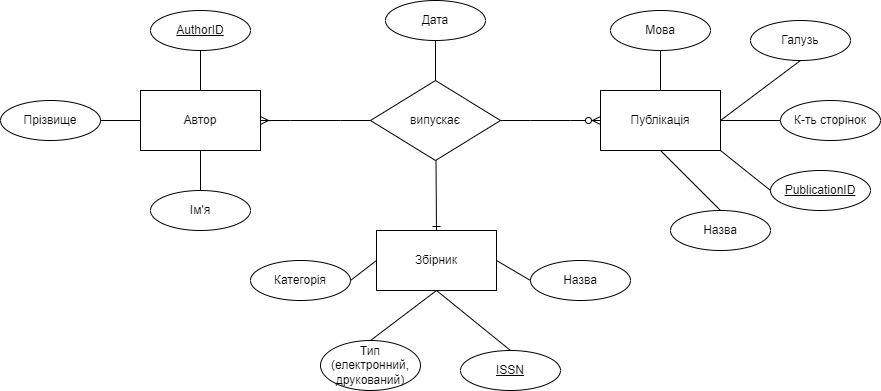


Рис.1 ER-діаграма за нотацією «Crow’s foot»

Модель має такі сутності: *Автор, Публікація, Збірник.*

1. *Автор* має атрибути:

* AuthorID (ідентифікатор автора)
* Ім’я
* Прізвище

Це основні данні, які потрібні від автора для публікацій.

1. *Публікація* має атрибути:

* PublicationID (ідентифікатор публікації)
* Назва
* Галузь
* Мова
* Кількість сторінок

1. *Збірник* потрібний для випуску публікації і містить атрибути:

* ISSN (ідентифікатор збірника)
* Назва
* Тип (у якому форматі був виданий)
* Категорія (міжнародний або всеукраїнський)

Ці три сутності реалізовані у тернарний зв’язок багато до багатьох (N:M). Один автор може написати багато публікацій виданих у збірники, а одну публікацію можуть написати багато авторів в один збірник.

***Завдання 2***

Перетворення розробленої моделі у схему бази даних (таблиці) PostgreSQL

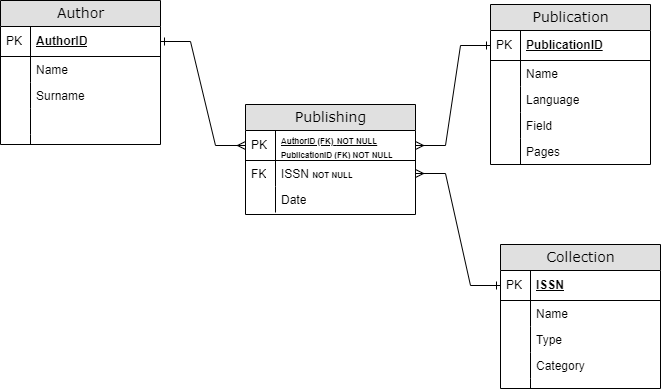


Рис.2 Схема бази даних графічному представленні

1. Сутність *Автор* було перетворено у таблицю Author з первинним ключом AuthorID та атрибутами Name (Ім’я), Surname (Прізвище).
2. Сутність *Публікація* було перетворено у таблицю Publication з первинним ключом PublicationID та атрибутами Name (Назва), Language (Мова), Field (Галузь), Pages (Сторінки) .
3. Сутність *Збірник* було перетворено у таблицю Collection з первинним ключом ISSN та атрибутами Name (Назва), Type (Тип видання), Category (Категорія).
4. Зв’язок «випускає» (M:N) зумовив появу таблиці Publishing (Видання/Випуск) в якому первинний ключ це AuthorID та PublicationID (з таблиць Author та Publication), вторинний ключ ISSN (з таблиці Collection) та атрибут Date (Дата).

У таблиці Publishing: AuthorID, PublicationID та ISSN мають обмеження NOT NULL, адже видання без автора, публікації та збірника не можливе. Первинні ключі обрані саме AuthorID та PublicationID, адже багато авторів можуть написати одну публікацію і один автор може написати багато публікацій, але видається публікація лише в одному збірнику і один раз. Додавання ISSN до первинних ключів дало б змогу випускати одну й ту ж публікацію у різних збірниках.

***Завдання 3***

Нормалізація схеми бази даних до третьої нормальної форми (3НФ)

Функціональні залежності:

***Author (AuthorID, Name, Surname)***

*AuthorID → Name, Surname*

*AuthorID → Name*

*AuthorID → Surname*

***Publication (PublicationID, Name******, Language, Field, Pages)***

*PublicationID → Name, Language, Field, Pages*

*PublicationID → Name*

*PublicationID → Language*

*PublicationID → Field*

*PublicationID → Pages*

***Collection (ISSN, Name, Type, Category)***

*ISSN → Name, Type, Category*

*ISSN → Name*

*ISSN → Type*

*ISSN → Category*

***Publishing (AuthorID, PublicationID, ISSN, Date)***

*AuthorID, PublicationID → ISSN, Date*

*AuthorID, PublicationID → ISSN*

*AuthorID, PublicationID → Date*

*PublicationID → Date (порушує 2NF)*

*PublicationID → ISSN (порушує 2NF)*

Тому таблицю ***Publishing*** перетворюю на дві таблиці:

***Publishing (AuthorID, PublicationID)***, яка має тривіальну функціоналну залежність та

***Publishing\_Pub(PublicationID***, *ISSN, Date)*

*PublicationID → ISSN, Date*

*PublicationID → ISSN*

*PublicationID → Date*

Тому схема баз даних приймає наступний вигляд:

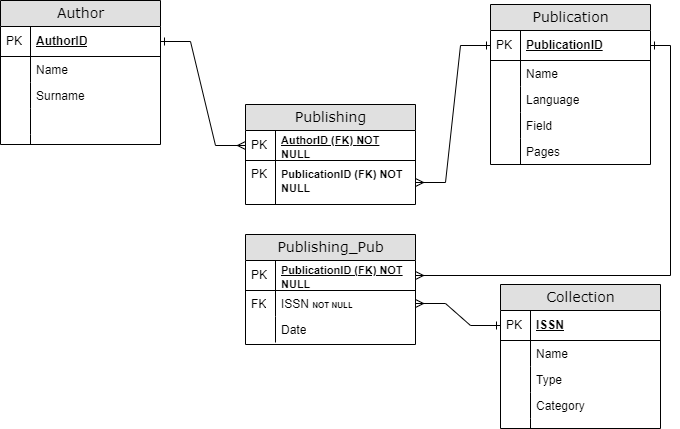


Рис.3 Нормалізований вид схеми баз даних у графічному представленні

Схема бази даних відповідає 1NF, адже комірки атомарні (по одному значенні в комірці.

Також схема відповідає 2NF, адже вона відповідає 1NF та неключові атрибути залежать від повного первинного ключа.

До того ж схема відповідє 3NF, адже вона відповідає 2NF та неключові атрибути не транзитивно залежні від кожного кандидатного ключа.

***Завдання 4***

Ознайомлення із інструментарієм PostgreSQL та pgAdmin 4:

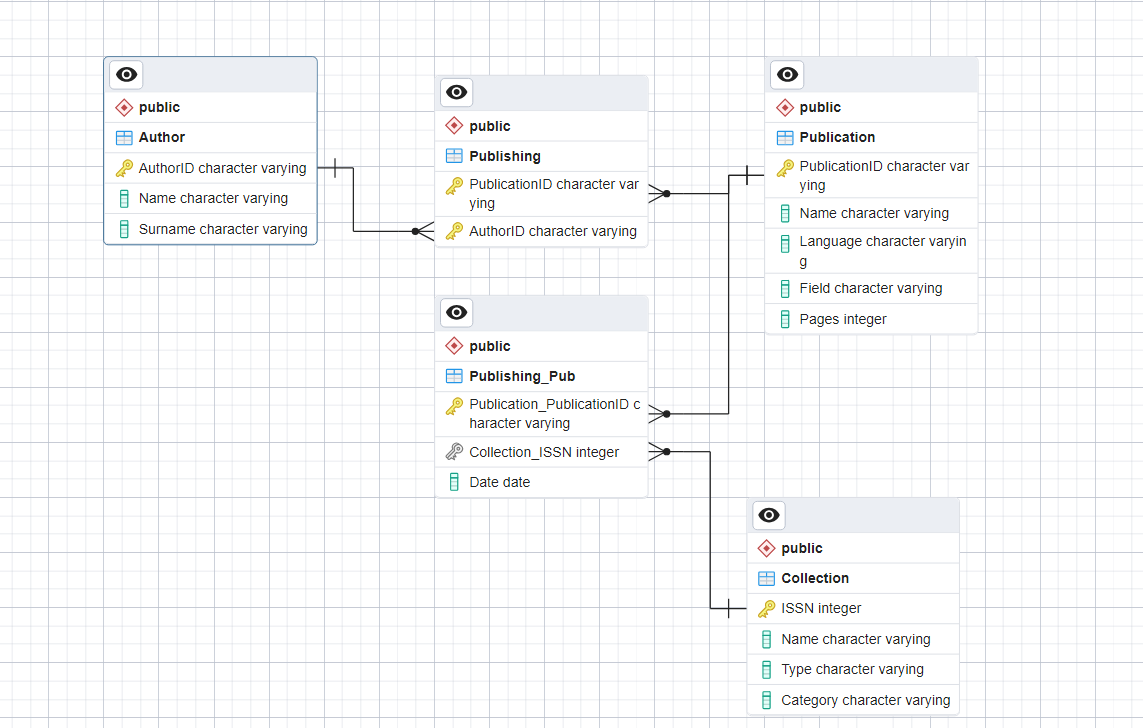
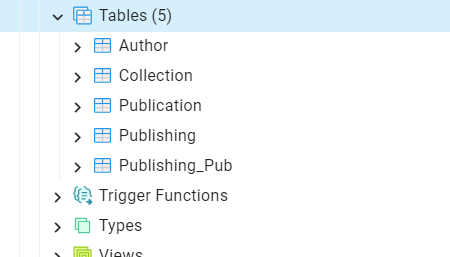
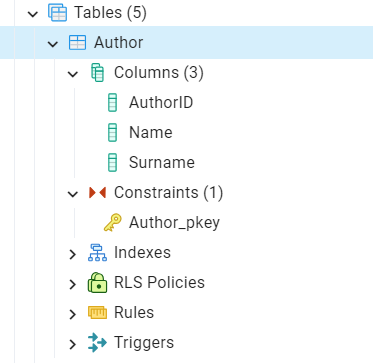


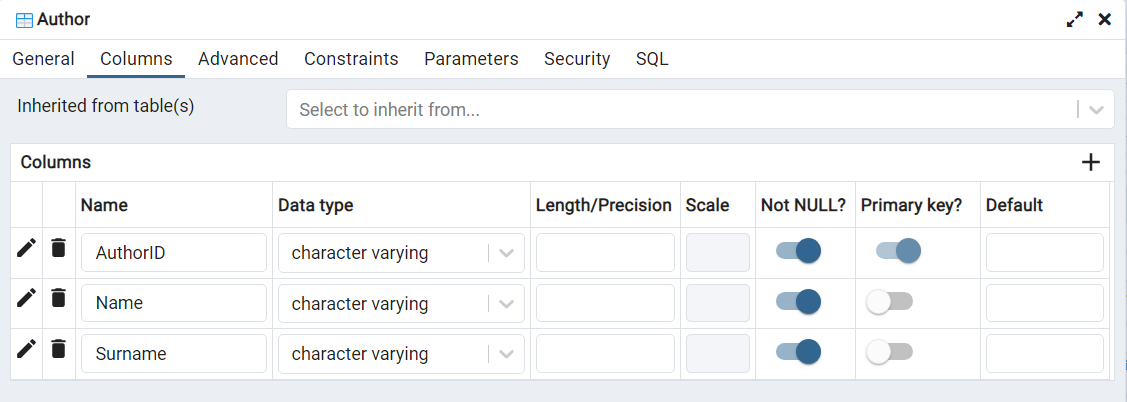
Рис.4 Схема бази даних у pgAdmin4

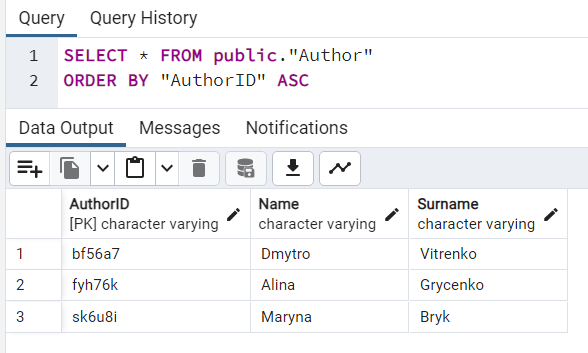
Створені таблиці за допомогою ERD tool:



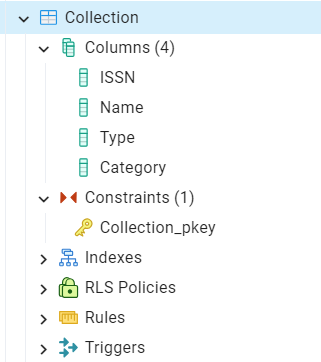
**Author**

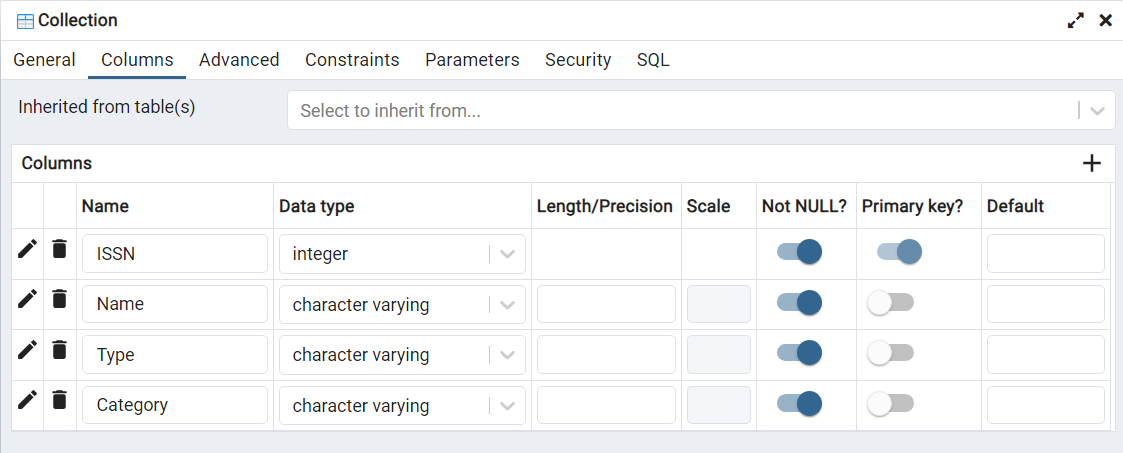


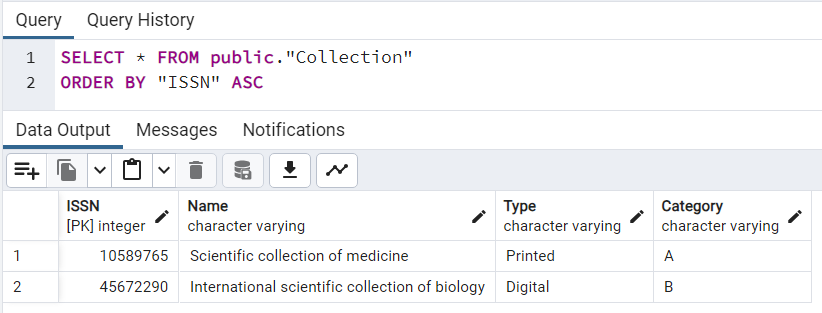




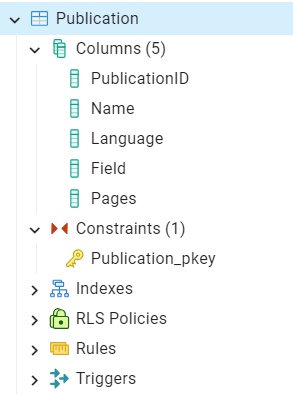
**Collection**

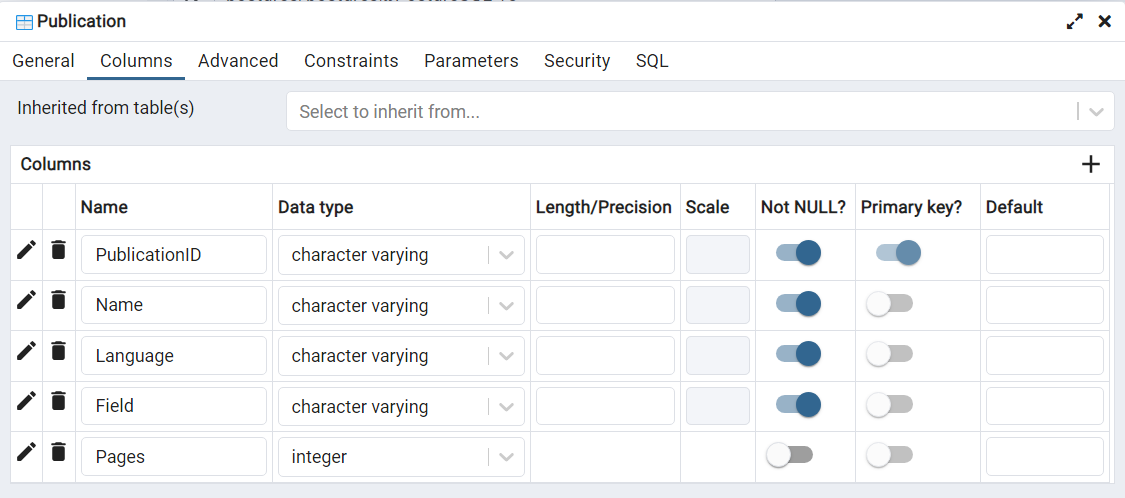


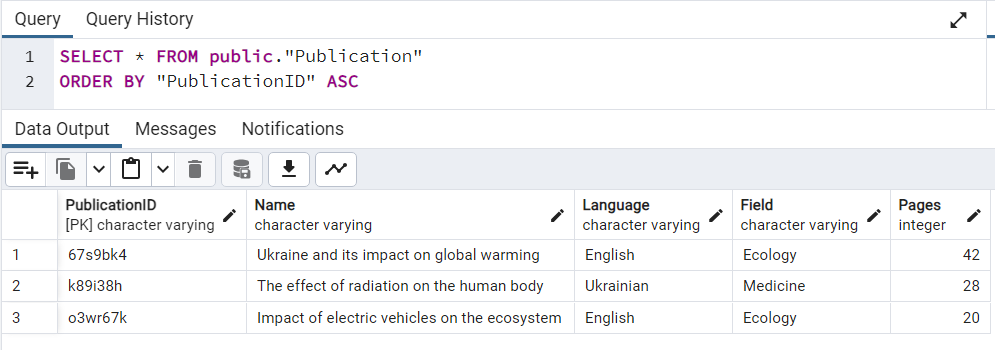




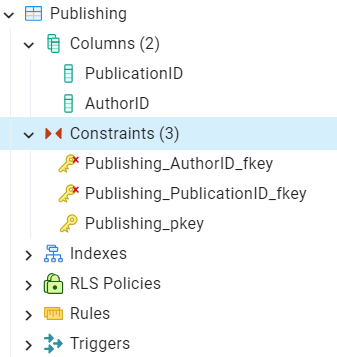
**Publication**

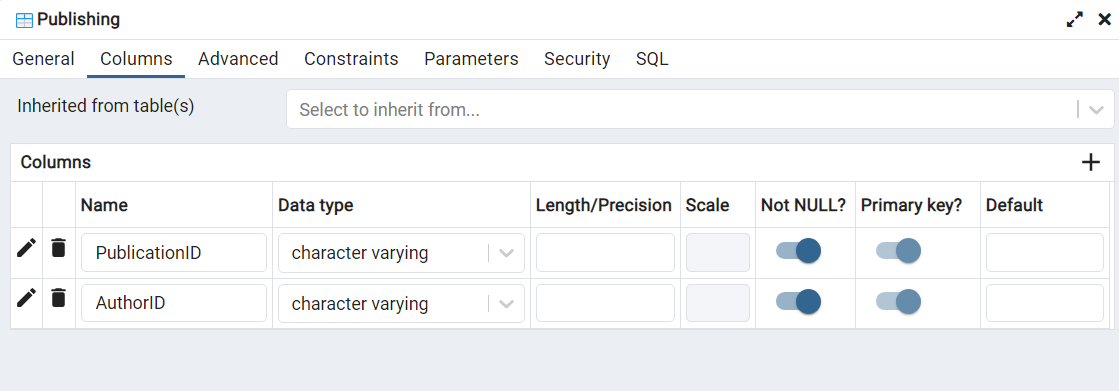


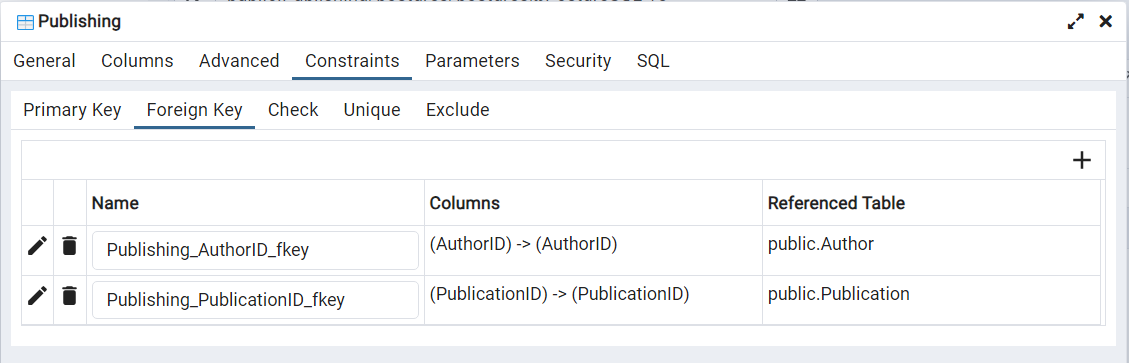


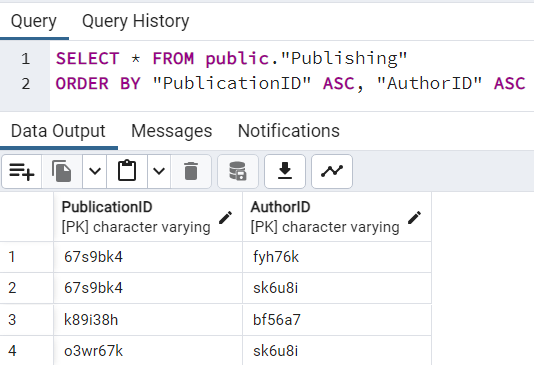


**Publishing**

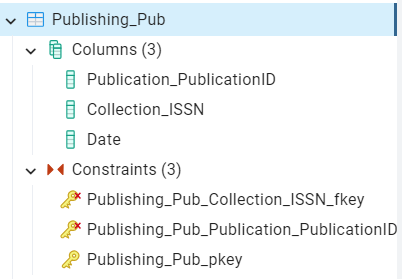


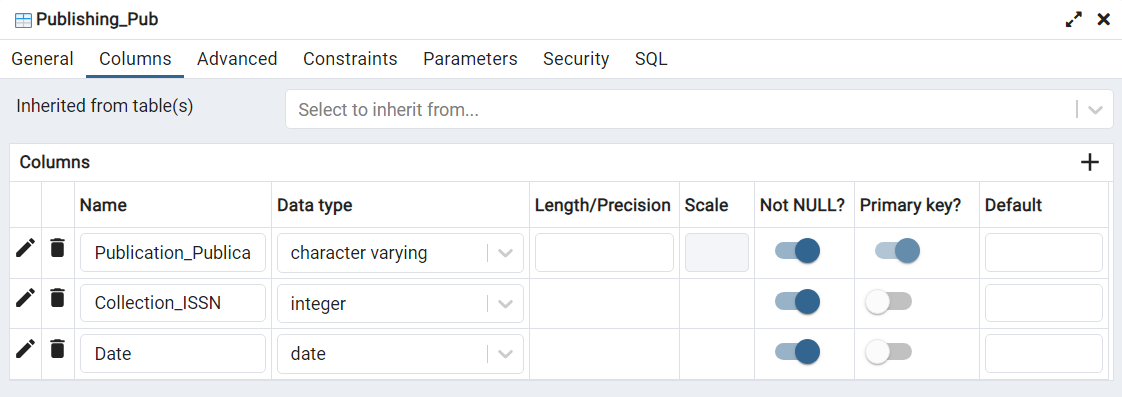




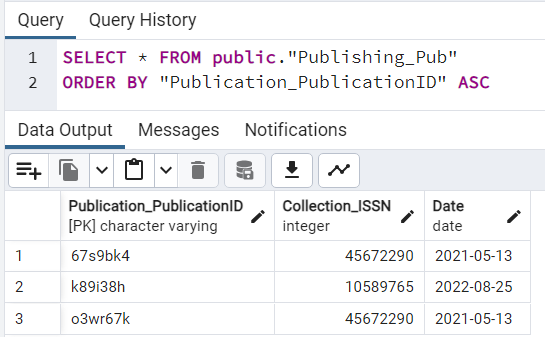


**Publishing\_Pub**









<https://github.com/SaiunD/DB_lab1>