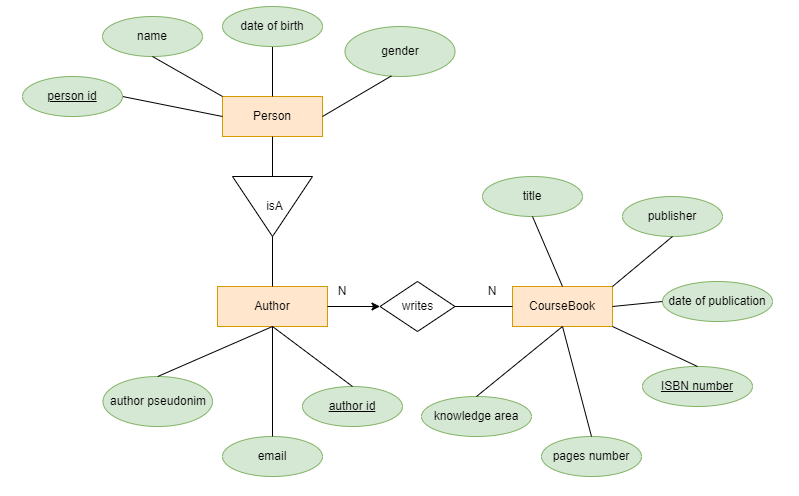
**Основна частина**

* 1. **Концептуальне проектування схеми**

Подія: Людина є автором підручника.

Виділимо 3 сутності: людина, автор та підручник. Людина(Person) має атрибути: ім’я(name), ідентифікаційний номер документу (person\_id), дату народження(date of birth) та стать(gender). Автор(Author) має такі ж властивості, що й людина, тобто Автор є людина, окрім цього Автор має ідентифікаційний номер(author id), можливо псевдонім(author name) та адресу пошти(email). Підручник(Coursebook) має свій ISBN код(ISBN number), назву(title), видавця(publisher), дату видання(date of publication), кількість сторінок(pages number), область знань(knowledge area). Можна встановити зв’язок типу «Є»(isA) між людиною та автором, тобто автор є людиною але не кожна людина є автором. Зв’зок між автором та підручником –можна визначити як багато-до-багатьох, оскільки автор може написати багато підручників і один підручник може бути написаний багатьма авторами.

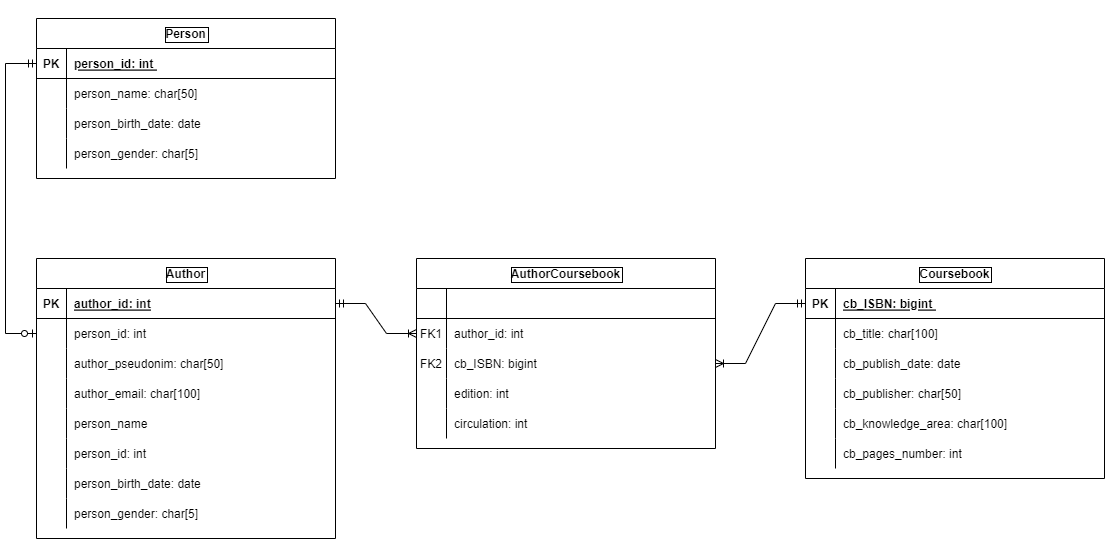
На малюнку 1.1 зображено концептуальне проектування схеми «Людина є автором підручника».

Мал 1.1 – концептуальна схема «Людина є автором підручника»

* 1. **Логічне проектування схеми**

Розглянемо сутності Person та Author. Нехай головним ключем сутності Person – це passport id, а головним ключем сутності Author – author id. Сутність Person є батьківською по відношенню до сутності Author, тому вважатимемо всі атребути Person успадковуються Author. Сутність Coursebook має головний ключ - cb\_ISBN. Розглянемо зв’язок між сутностями Author та Coursebook. Оскільки тип зв’язку мж ними багато-до-багатьох, то необхідно додатково створити сутність-зв’язку Author\_Coursebook. Атрибути Author\_Coursebook: cb\_ISBN (ключ-зв’язка із сутністю Coursebook) , author\_id (ключ-зв’язка із сутністю Author) та особисті атрибути: edition(видання) та circulation(наклад).

На малюнку 1.2 зображено логічне проектування схеми «Людина є автором підручника».



Мал 1.2 – логічна схема «Людина є автором підручника»

* 1. **Фізичне проектування схеми**

Визначимо типи атрибутів для таблиці(сутності) Person. Char(n): person\_name, person\_gender. Int: person\_id. Date: person\_birth\_date. Всі атрибути мають бути заповнені (NOT NULL). Область можливих значень атрибуту person\_gender: ‘woman’, ‘man’.

Всі атребути Person успадковуються таблицею Author. Для цього застосуємо ключове слово inherits, що дозволяє в postresql діалекті створювати ієрархічні зв’язки. Визначимо додаткові атребути. Int: author\_id. Char(n): author\_pseudonim, author\_email. Атрибути NOT NULL: author\_id,.

Визначимо типи атрибутів для таблиці Coursebook. Bigint: cb\_ISBN. Int: cb\_pages\_number. Char(n): cb\_title, cb\_publisher, cb\_knowledge\_area. Date: cb\_publish\_date. Всі атрибути мають бути NOT NULL.

Визначимо типи атрибутів для таблиці Author\_coursebook. Bigint: cb\_ISBN, Int: author\_id, edition, circulation. Атрибути author\_id та cb\_ISBN мають бути NOT NULL та UNIQUE(унікальними).

Визначаючи ключі-зв’язки(FOREIGN KEY), визначимо дії на оновлення та видалення батьківських таблиць (ON UPDATE CASCADE, ON DELETE CASCADE).

Postrgesql-скрипт фізичної схеми знаходится в розділі Скрипти, Скрипт 1.1.

**2.1 Генерація бази даних**

Створено базу даних з книги Б. Форта, на базі даного скрипту та виконано запити(розділ Скрипти, Скрипт 1.2).

**2.2 Виконання запитів**

**-- select**

**-- distinct prod\_name**

**-- from products inner join orderitems on products.prod\_id = orderitems.prod\_id**

**-- where item\_price = (select max(item\_price) from orderitems)**

**-- Як звуть покупця з найдовшим іменем – поле назвати long\_name**

**R1**

**-- select**

**-- cust\_name as long\_name**

**-- from customers**

**-- where length(cust\_name) = (select**

**-- max(length(cust\_name))**

**-- from customers)**

**-- Вивести PROD\_ID товарів та імена постачальників для тих товарів,**

**-- які були продані. Результат вивести у верхньому регістрі, як єдине**

**-- поле products\_sold**

**-- select**

**-- trim(upper(pr\_v.prod\_id || ' - ' || pr\_v.vend\_name)) as products\_sold**

**-- from (select**

**-- pr.prod\_id, v.vend\_name**

**-- from products pr**

**-- inner join vendors v on v.vend\_id = pr.vend\_id) pr\_v**

**-- where pr\_v.prod\_id in (select**

**-- prod\_id**

**-- from orderitems)**

**ВАРІАНТ №4**

**Завдання 1.** Спроектувати базу даних, що дозволить відобразити наступні події (*5 балів*):

**Людина є автором підручника.**

**1a)** Визначити сутності та їх атрибути, встановити зв’язки між сутностями. Побудувати ER-діаграму.

**1b)** Побудувати логічну схему таблиць, використовуючи «crow's foot notation».

**1с)** За допомогою команд мови SQLстворити таблиці в СУБД PostgreSQL. Визначити поля та типи. Первинні та зовнішні ключі створювати окремо від таблиць, використовуючи команду ALTER TABLE.

**Завдання 2.** Згенерувати базу даних з книги Б. Форта та виконати запити (потрібні для виконання завдань файли *create.txt, populate.txt* можна завантажити, наприклад, з <https://github.com/alinbxSorcerer/SQL-in-10-minutes-with-notes.git>) (*6 балів*):

**2a) Яка назва проданого найдорожчого товару?**

**2b) Як звуть покупця з найдовшим іменем – поле назвати long\_name?**

**2c) Вивести PROD\_ID товарів та імена постачальників для тих товарів, які були продані. Результат вивести у верхньому регістрі, як єдине поле products\_sold.**

**Завдання 3.** Виконати запити 2a), 2b) з попереднього завдання, використовуючи операції реляційної алгебри Кодда та агрегатні функції мови SQL (*4 бали*).