Логічне проектування баз даних

План

1. Етапи логічного проектування
2. Спрощення концептуальної моделі
3. Визначення набору відношень
4. Перевірка відношень за допомогою правил нормалізації
5. Перевірка відповідності відношень вимогам транзакцій користувачів
6. Перевірка підтримки цілісності

Етапи логічного проектування

Етапи логічного проектування

- Логічне проектування виконується для певної моделі даних.
- Для реляційної моделі даних логічне проектування полягає у:

- створенні реляційної схеми;
- визначенні числа і структури таблиць;
- формуванні запитів до БД;
- визначенні типів звітних документів;
- розробці алгоритмів обробки інформації;
- створенні форм для вводу і редагування даних в БД
- …

Етапи логічного проектування

- Концептуальні моделі за певними правилами перетворюються в логічні моделі даних.
- Коректність логічних моделей перевіряється за допомогою правил нормалізації, які дозволяють переконатися в структурній узгодженості, логічній цілісності і мінімальній збитковості прийнятої моделі даних.
- Модель також перевіряється з метою виявлення можливостей виконання транзакцій, які будуть задаватися користувачами.
- Проектування являє собою циклічний процес.

Усунення особливостей логічної моделі, які не сумісні з реляційною моделлю:
- вилучення зв'язків M:N
- вилучення рекурсивних зв'язків
- вилучення складних зв'язків
- вилучення багатозначних зв'язків
- вилучення зв'язків з атрибутами
Визначення набору відношень
Перевірка відношень за допомогою правил нормалізації
Перевірка відповідності відношень вимогам транзакцій користувачів
Перевірка підтримки цілісності

Спрощення концептуальної моделі

- **Вилучення двосторонніх зв'язків “багато до багатьох”**

Перетворення зв'язку "багато до багатьох" виконується шляхом введення проміжної сутності із заміною одного зв'язку M:N двома зв'язками 1:N з новою сутністю.

Вилучення двосторонніх зв'язків “багато до багатьох”

а

Вилучення двосторонніх зв'язків “багато до багатьох”

- У результаті перетворення отримана нова сутність, яка є слабкою і залежить від двох інших сутностей. Її первинний ключ складається з первинних ключів двох сутностей, а кожен атрибут окремо є вторинним ключем.

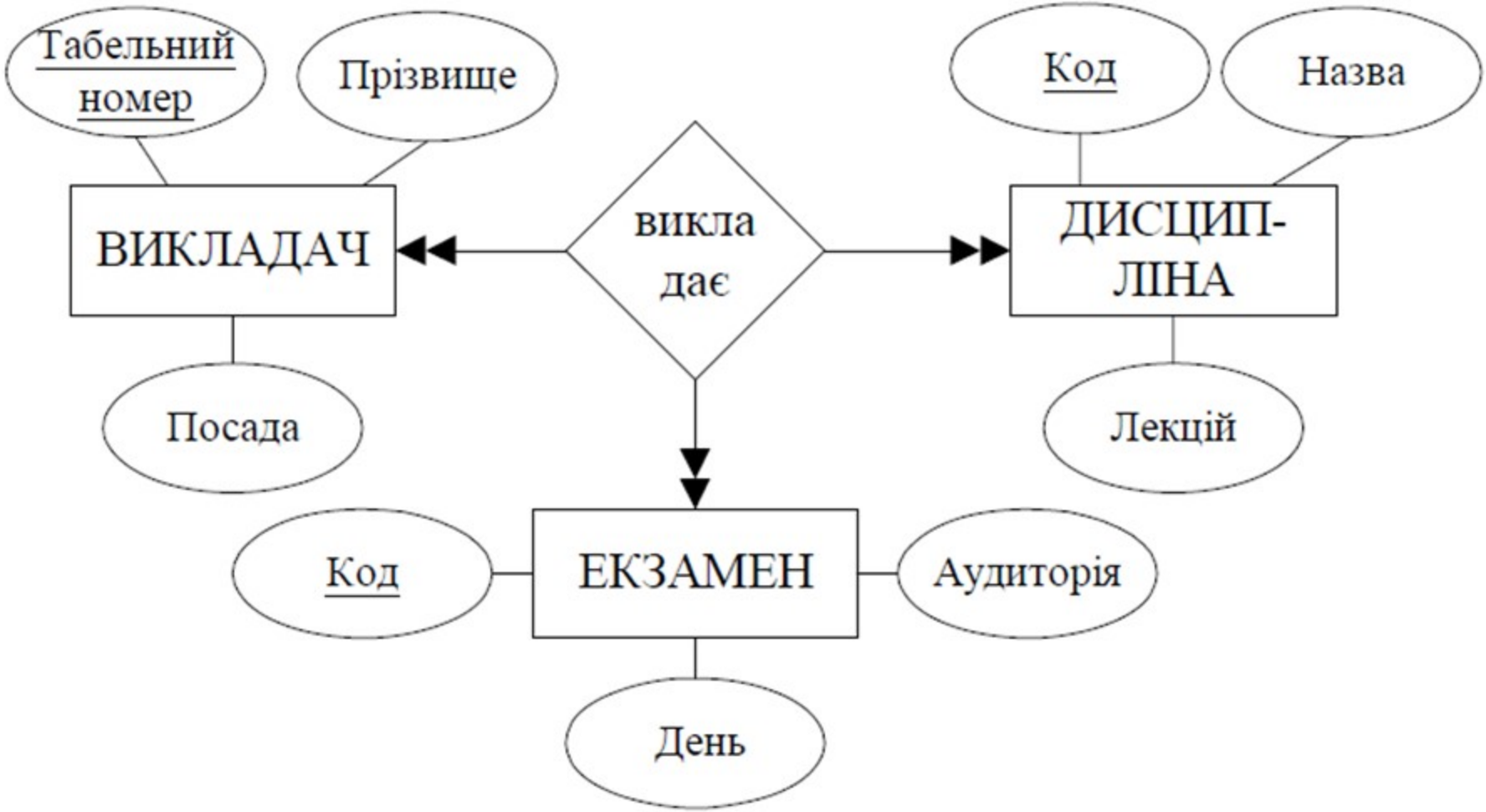
Спрощення концептуальної моделі

- **Вилучення складних зв'язків**

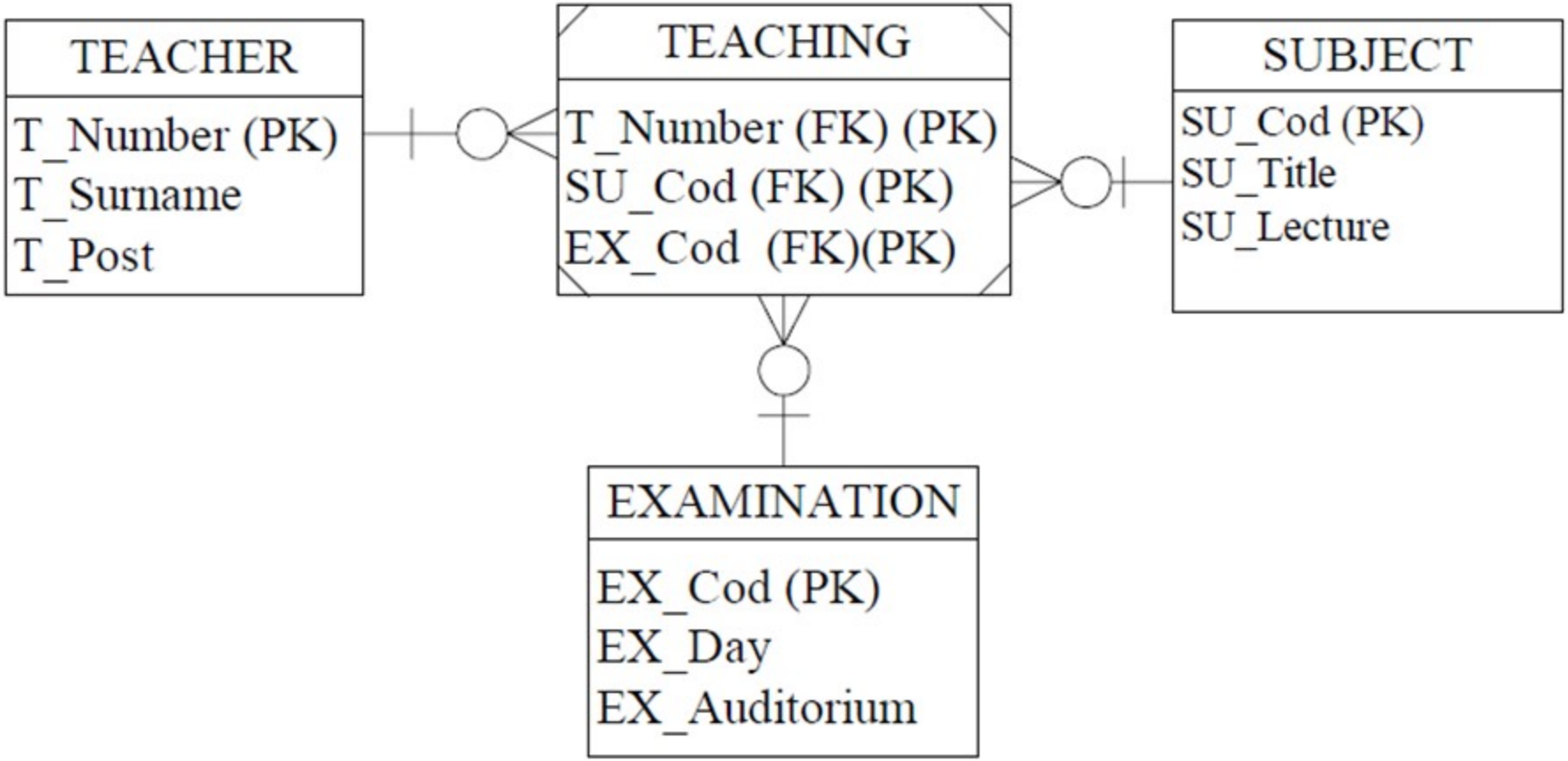
Для вилучення складних зв'язків виконуються такі операції:

- у модель вводиться нова сутність;
- складний зв'язок замінюється бінарними зв'язками "один до багатьох" зі знов створеною сутністю;
- кількість бінарних зв'язків дорівнює ступеню складності зв'язку.

Вилучення складних зв'язків



Вилучення складних зв'язків



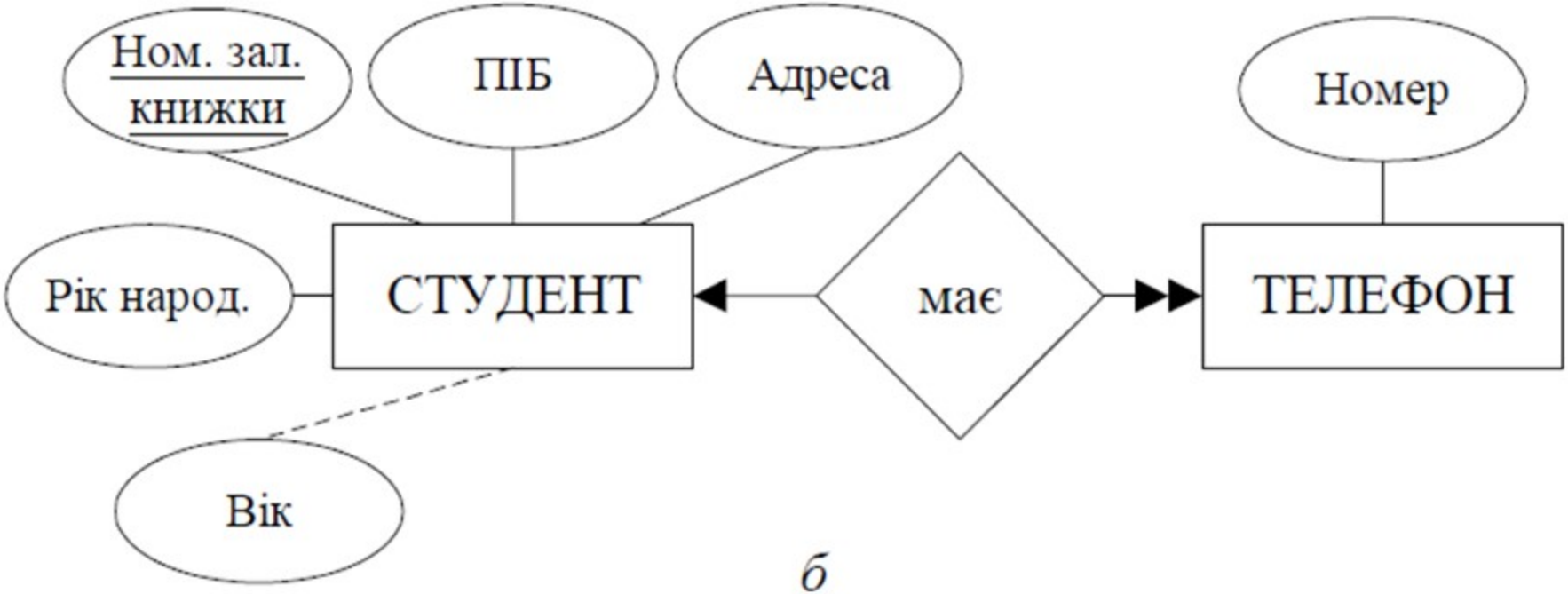
Спрощення концептуальної моделі

- **Вилучення багатозначних атрибутів**

Якщо в концептуальній моделі даних присутній багатозначний атрибут, то потрібно виконати декомпозицію цього атрибуту для визначення деякої сутності.

Вилучення багатозначних атрибутів

Вилучення багатозначних атрибутів



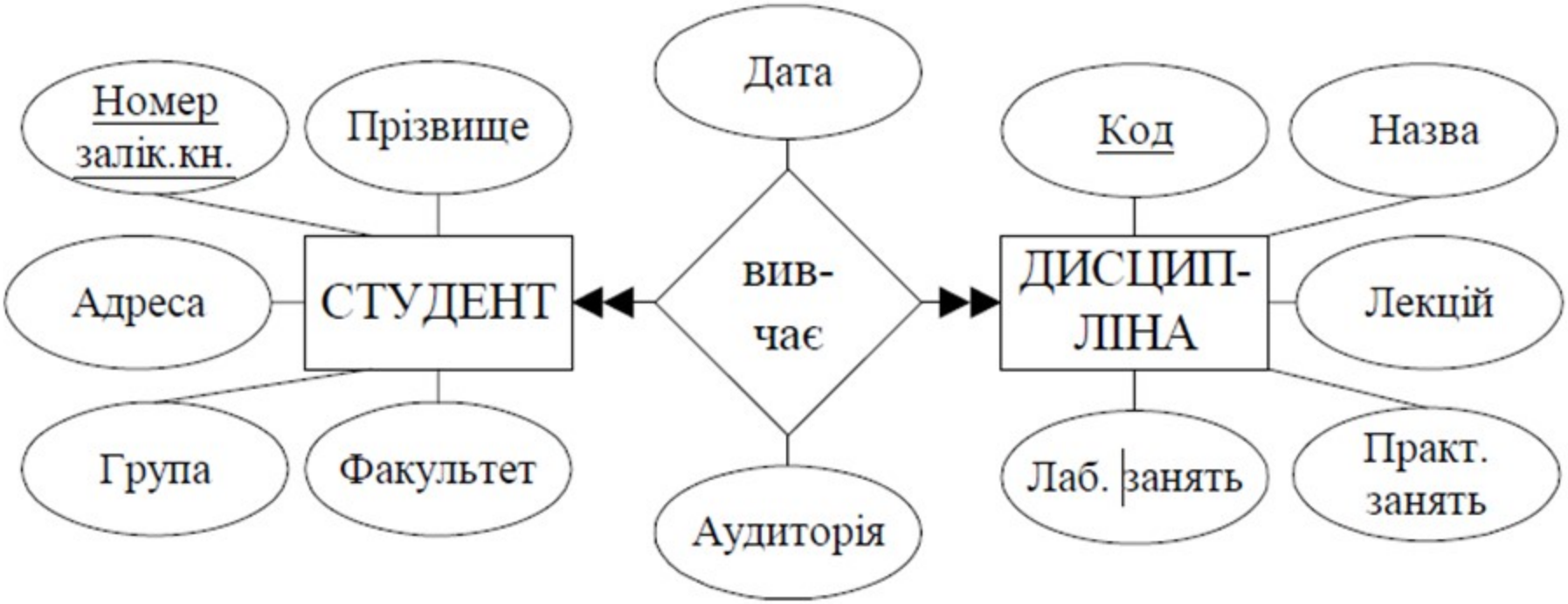
б

Спрощення концептуальної моделі

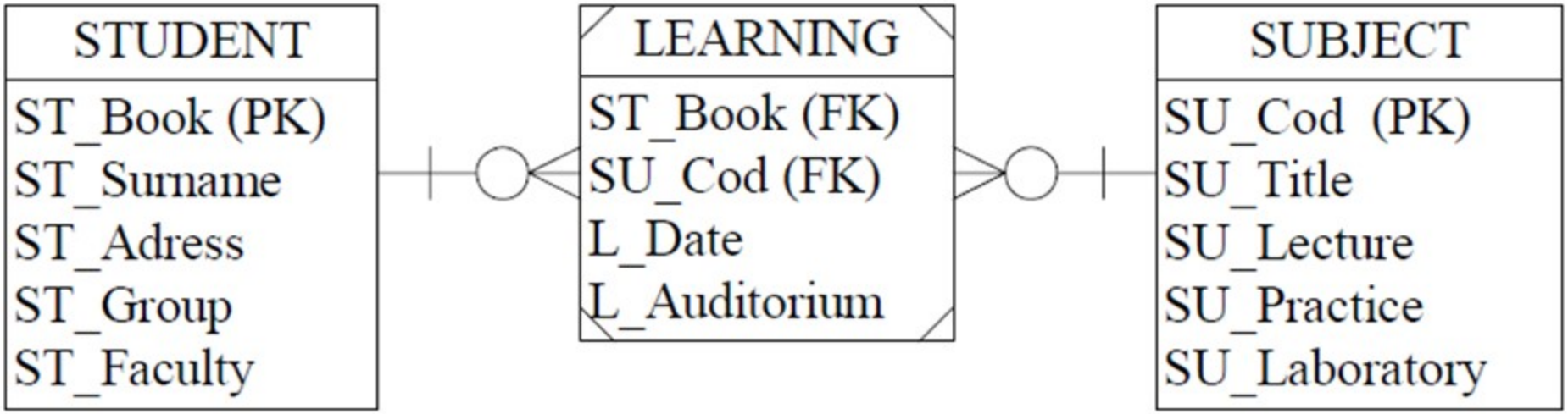
- **Вилучення зв'язків з атрибутами**

Вилучення зв'язків з атрибутами виконується шляхом додавання у модель нової сутності для відношення M:N з атрибутами зв'язку. Для відношення 1:M атрибути зв'язку передаються у сутність "багато" без створення нової сутності.

Вилучення зв'язків з атрибутами



Вилучення зв'язків з атрибутами

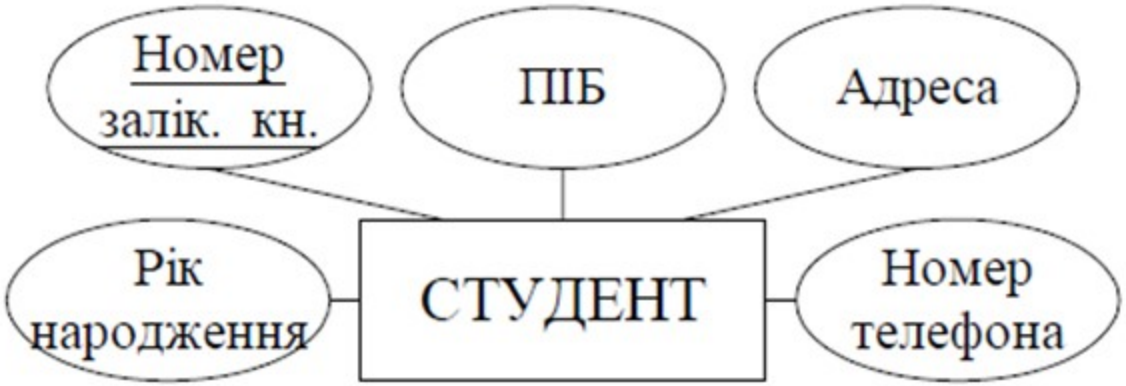


Визначення набору відношень

Сутності і атрибути

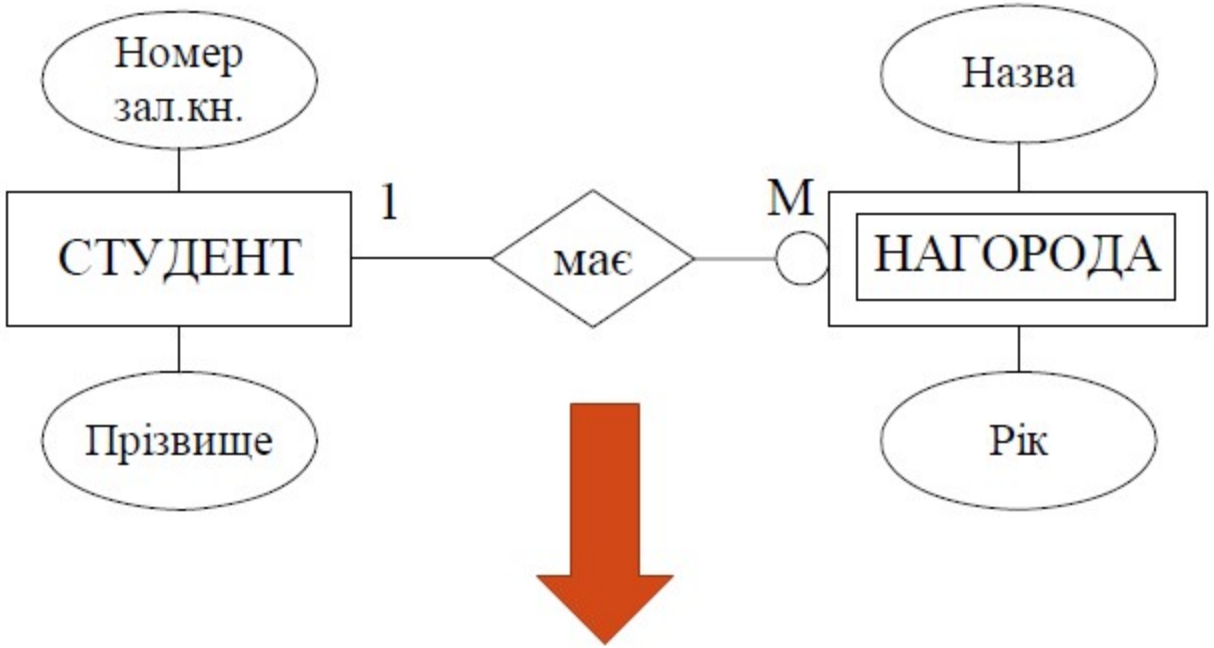
- Для кожної сутності створюється відношення, кожен атрибут сутності стає атрибутом відповідного відношення.
- Для *сильних сутностей* первинний ключ сутності стає PRIMARY KEY (PK) відповідного відношення.

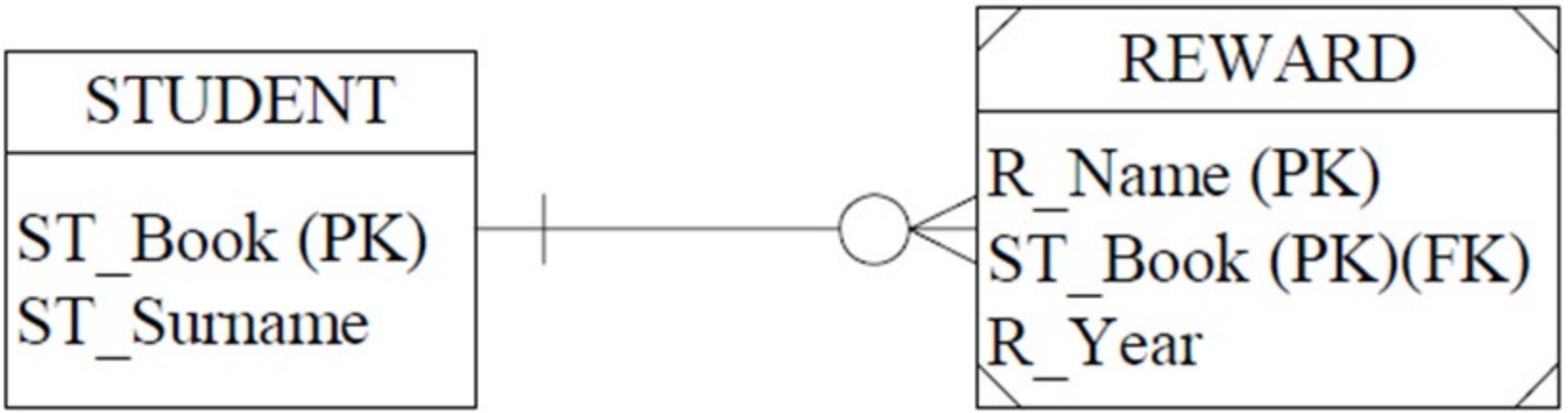
Перетворення сутності *Студент* у відношення *Student*



STUDENT
ST_Book (PK)
ST_Name
ST_Adress
ST_Birth
ST_Tel

Перетворення сильної сутності *Студент* і слабкої сутності *Нагорода*





Зв'язки

Зв'язки

Типи зв'язків:

- "один до одного";
- "один до багатьох";
- рекурсивні зв'язки;

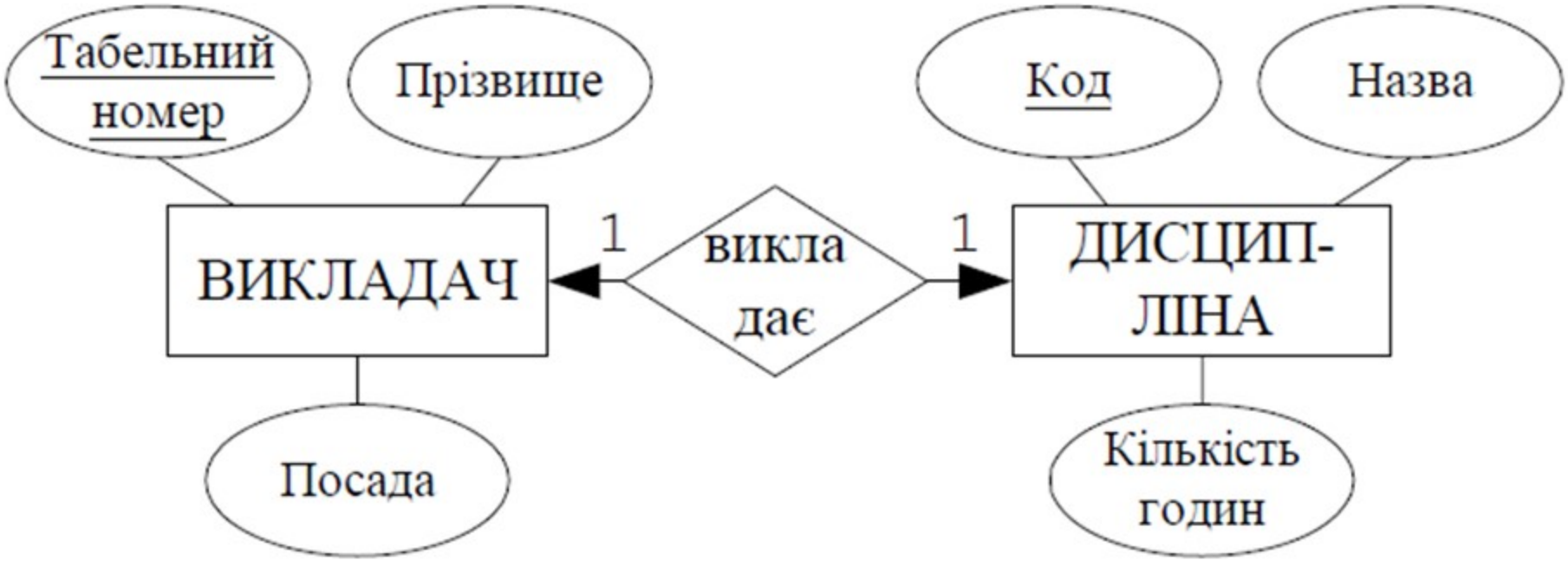
Зв'язки між відношеннями в реляційній моделі реалізуються шляхом використання первинних і зовнішніх ключів.

Зв'язки "один до одного"

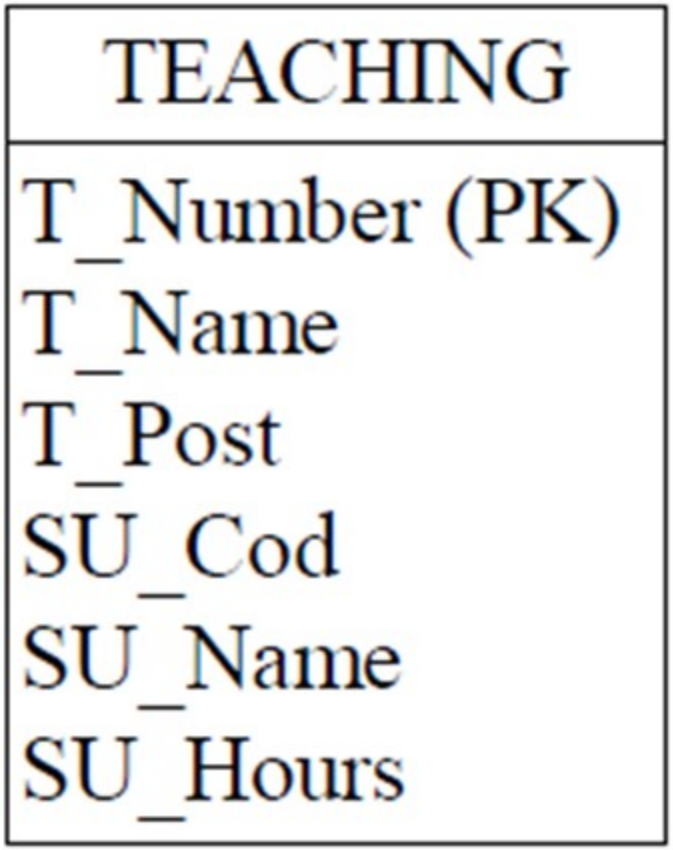
В концептуальних моделях даних визначають такі обмеження ступеня участі сутностей:

- обов'язкова участь для обох сутностей;
- обов'язкова участь для однієї сутності;
- необов'язкова участь для обох сутностей.

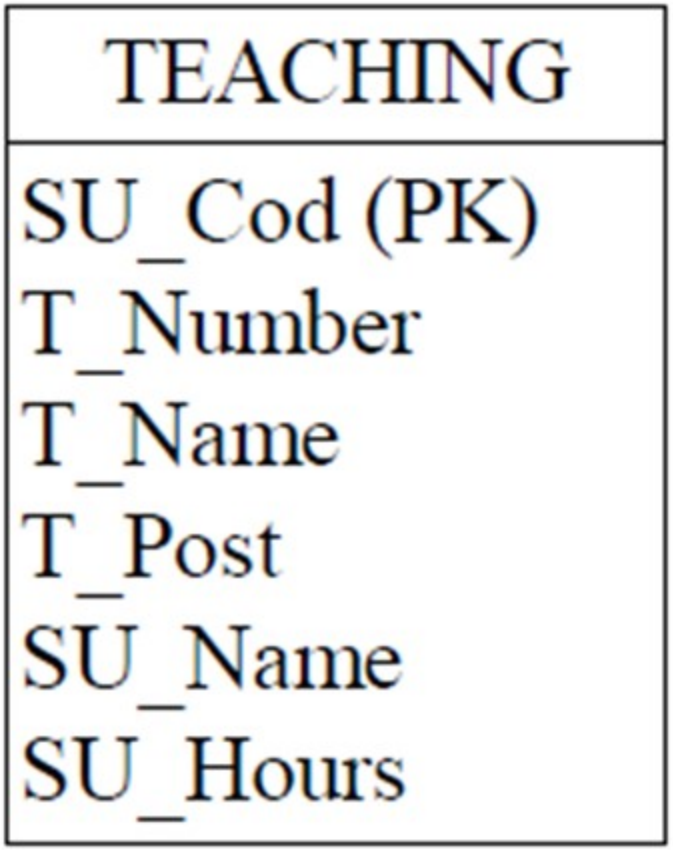
Зв'язки "один до одного"



Обов'язкова участь для обох сутностей

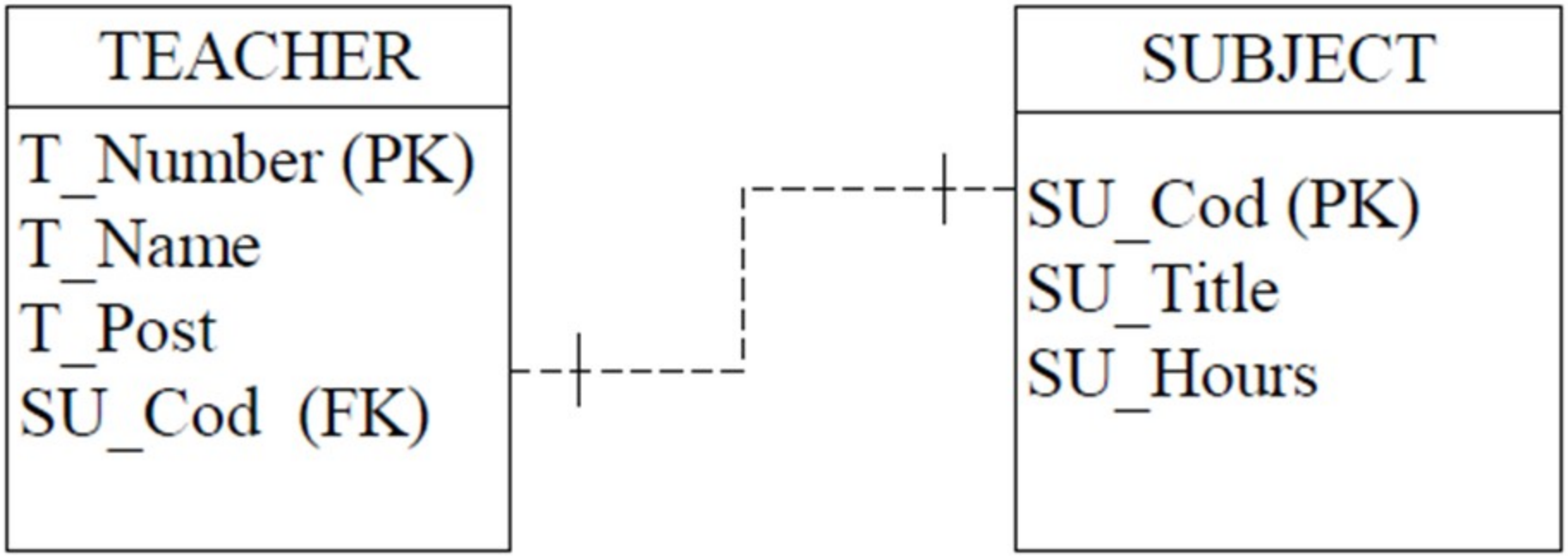


а

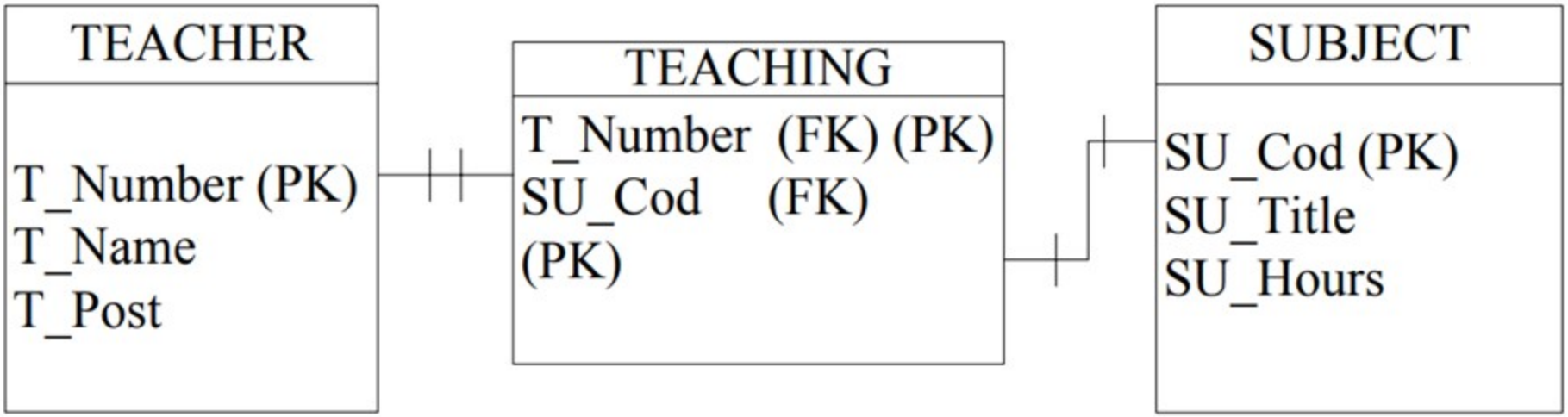


б

Обов'язкова участь для однієї сутності



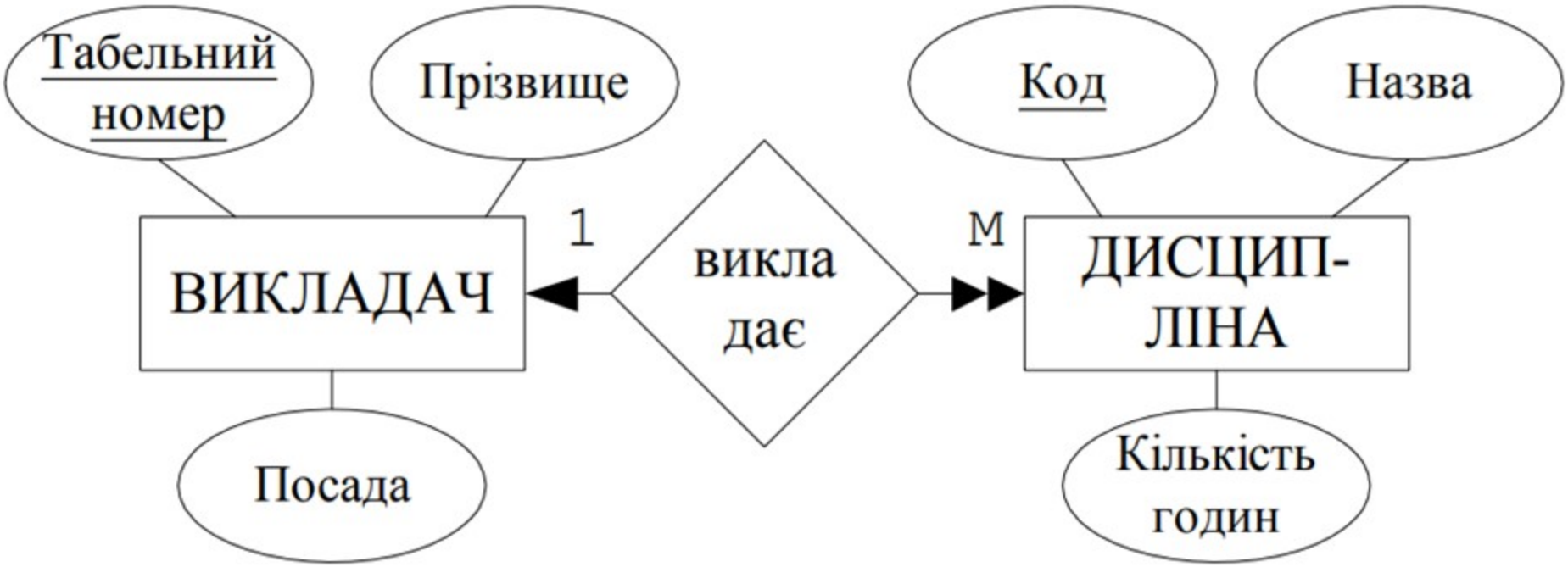
Необов'язкова участь для обох сутностей



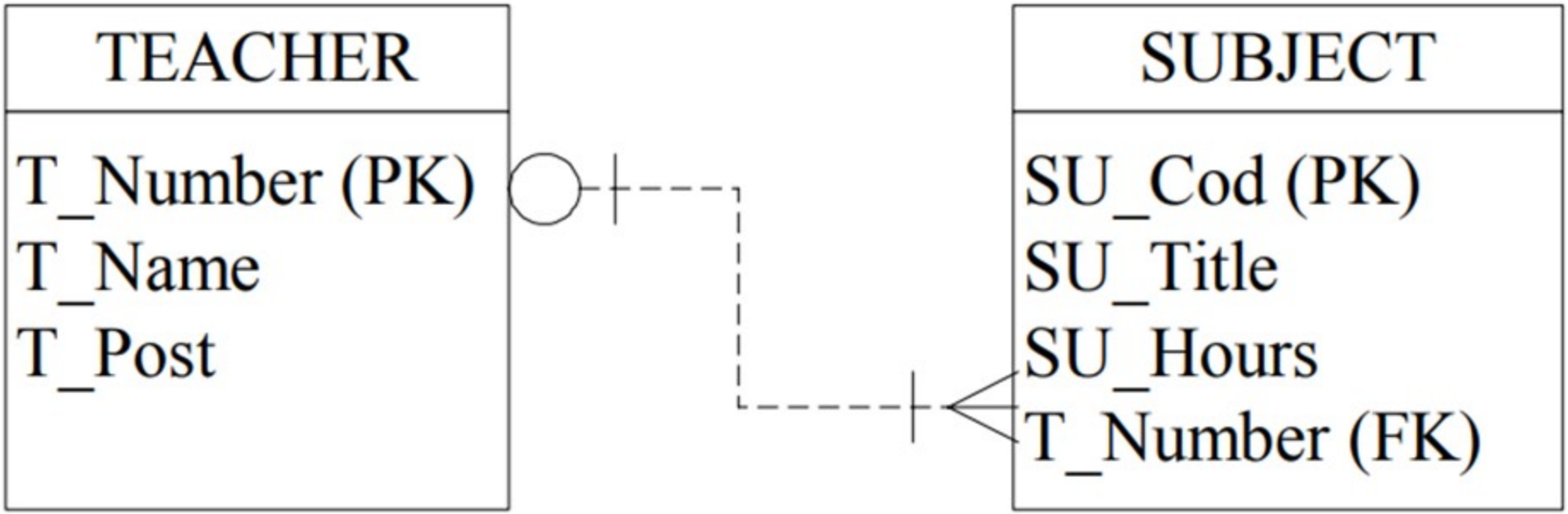
Зв'язки "один до багатьох"

- У кожне відношення, яке відповідає підлеглій (дочірній) сутності, додається набір атрибутів основної (батьківської) сутності, який складає первинний ключ основної сутності.
- У відношенні, що відповідає підлеглій сутності, цей набір атрибутів стає зовнішнім ключем (FOREIGN KEY, FK).

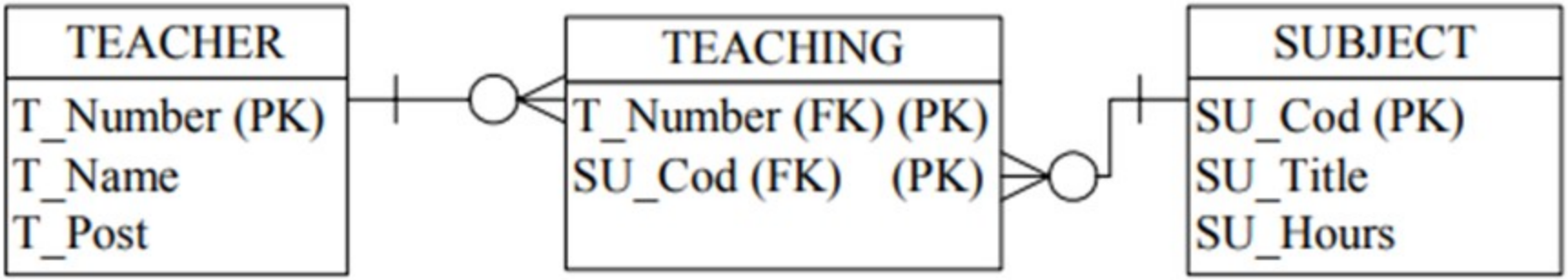
Зв'язок 1:М між сутностями *Викладач і Дисципліна*



Необов'язкова участь сутності *Викладач* і обов'язкова участь сутності *Дисципліна*



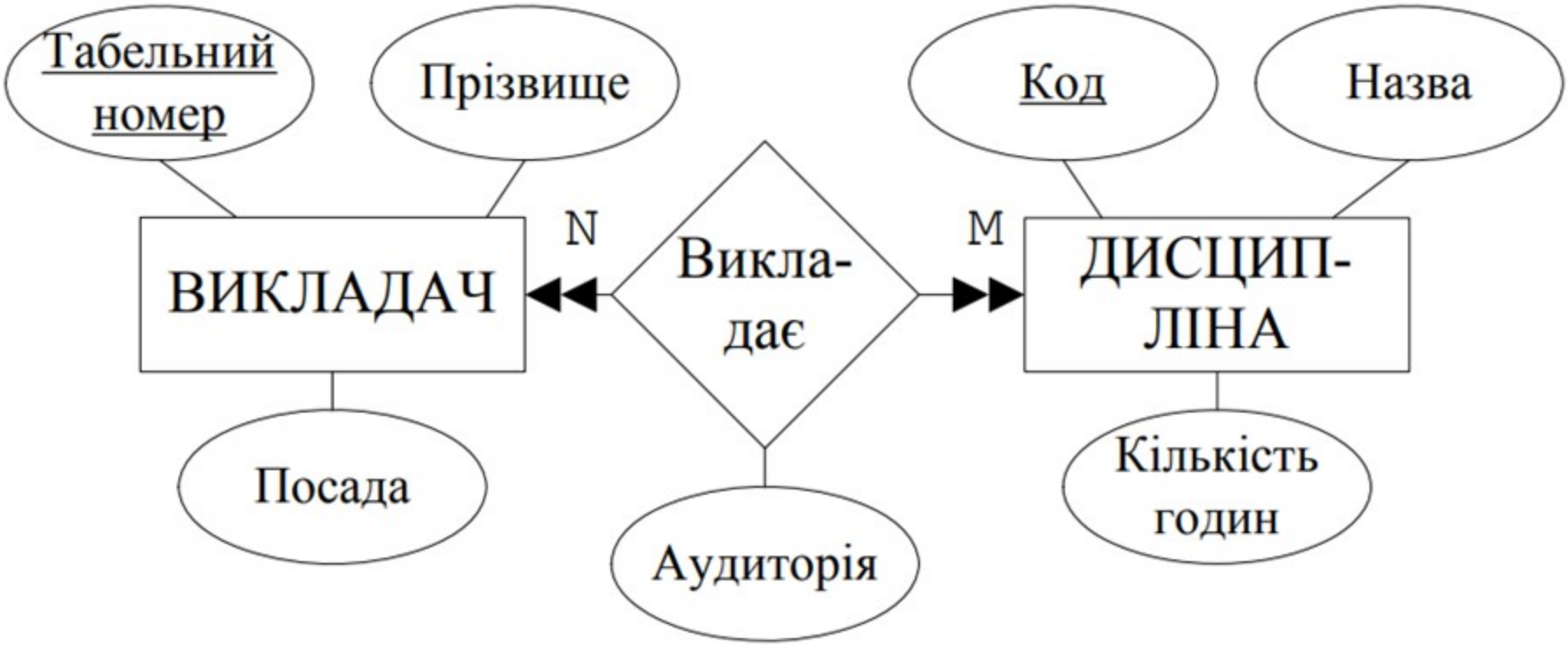
Необов'язкова участь сутності *Викладач* і необов'язкова участь сутності *Дисципліна*



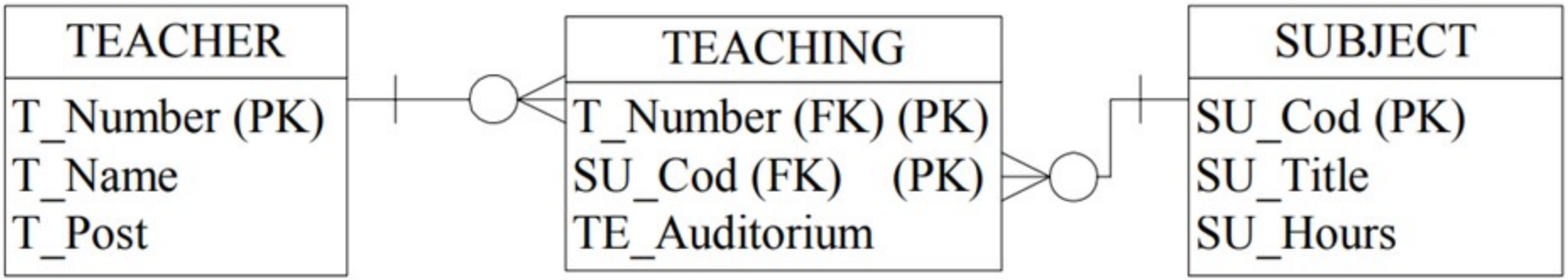
Зв'язки "багато до багатьох "

- Для кожного зв'язку M:N необхідно створювати додаткове відношення, яке представляє цей зв'язок і включати в нього всі атрибути, які входять в склад цього зв'язку.
- Копії атрибутів первинного ключа сутностей, які беруть участь у зв'язку, передаються у нове відношення для використання в якості зовнішніх ключів.
- Ці зовнішні ключі утворюють також первинний ключ нового відношення.

Зв'язок N:M між сутностями *Викладач і Дисципліна*



Перетворення зв'язку N:M у реляційну схему



Перевірка відношень за допомогою правил нормалізації

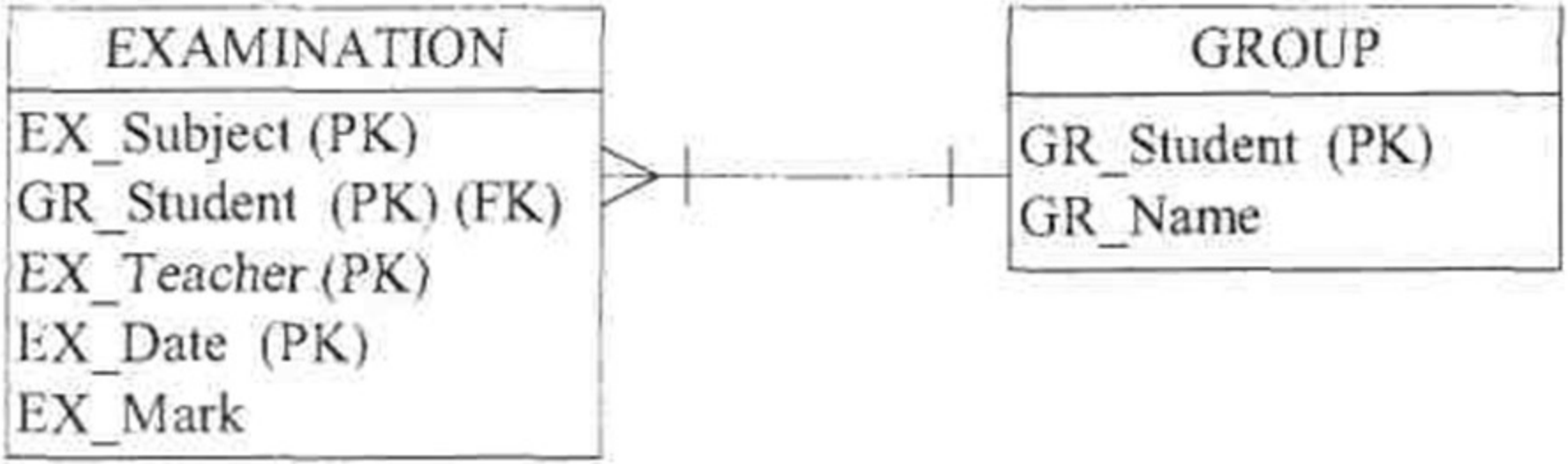
Перетворення сутності *Екзамен (а)* у відношення *Examination (б)*

EXAMINATION
EX_Subject (PK)
EX_Student (PK)
EX_Teacher (PK)
EX_Date (PK)
EX_Group
EX_Mark

Функціональні залежності

- *Дисципліна, Викладач, Студент, Дата —> Оцінка*
- *Студент —> Група*

Реляційна схема, яка відповідає сутності *Екзамен*



Перевірка відповідності відношень вимогам транзакцій користувачів

Транзакції користувачів

- Перевірка полягає в нанесенні безпосередньо на ER-діаграму всіх шляхів, які потрібні для виконання кожної з транзакцій.
- Якщо таким чином вдається виконати всі транзакції, то перевірка на цьому завершується.
- У протилежному випадку необхідно повернутися до попередніх етапів і перевірити, а у разі потреби і змінити ті фрагменти моделі, які не відповідають необхідній роботі транзакцій.
- Якщо в результаті перевірки будуть виявлені області, які не беруть безпосередньої участі у роботі транзакцій, то можливо їх вилучення з моделі.

Перевірка підтримки цілісності

Перевірка підтримки цілісності

Цілісність на стадії проектування полягає у такому:

- наявність обов'язкових і необов'язкових значень даних для атрибутів (NULL, NOT NULL);
- наявність обмежень для доменів атрибутів (визначення області значень або діапазону значень);
- цілісність сутностей (обов'язкова наявність Primary Key в кожному відношенні);
- посилкова цілісність (зв'язування таблиць за допомогою Foreign Key);
- обмеження предметної області (бізнес-правила), які реалізуються як засобами БД, так і на рівні застосувань.