



Бази даних та Інформаційні системи

2. ER-діаграма

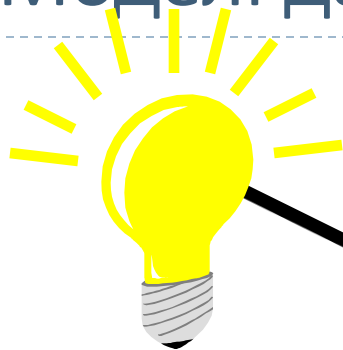
СумДУ, каф. КН
2020

Моделі даних

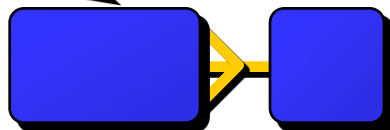
- ▶ **Інформаційна модель** даних відображає потреби системи в даних і зв'язки між даними з точки зору їх споживачів - користувачів.
- ▶ На основі інформаційної моделі в процесі проектування створюються логічна і фізична моделі даних.
 - ▶ **Логічна модель** (Сутнісна) даних є незалежним логічним поданням даних.
 - ▶ **Фізична модель** (Таблична) даних містить визначення всіх реалізованих об'єктів в конкретній базі даних для конкретної СУБД.



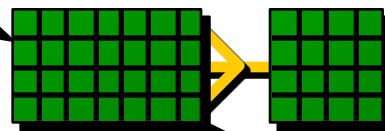
Моделі даних



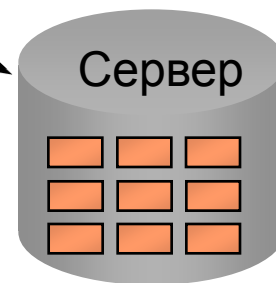
Модель
системи в
голові клієнта



Сутнісна модель
моделі клієнта



Таблична модель
сутнісної моделі



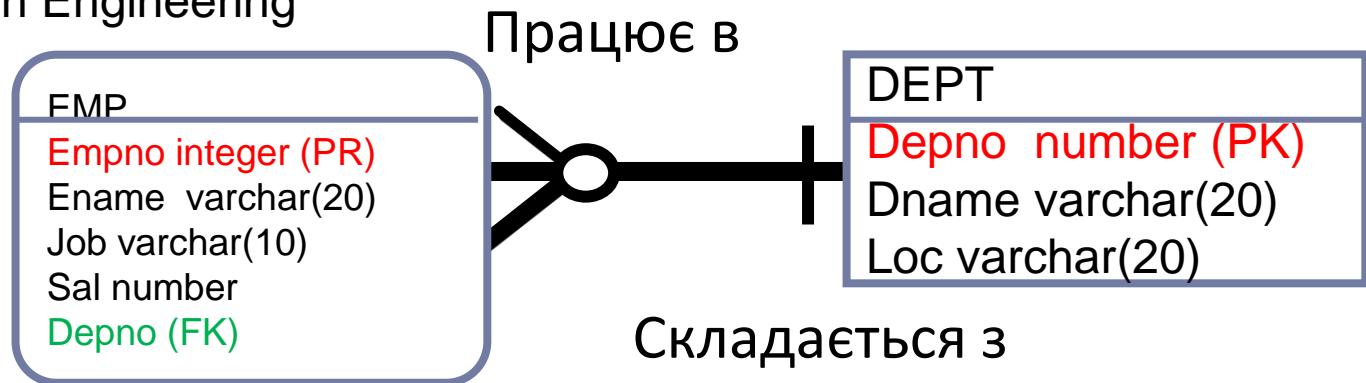
Таблиці на диску



Entity Relationship-модель

ER-діаграма створюється на основі специфікацій або оповідань.

Information Engineering



Сценарій:

- "... Додати одного або більше співробітників в відділ. . . "
- "... Деякі відділи все ще не мають асоційованих співробітників. . . "

Entity Relation Diagram

- ▶ ERD доповнює DFD. DFD орієнтоване на процеси і потоки даних між ними, ERD - на дані і зв'язки між ними.
- ▶ ERD дозволяє представити всю інформацію, яка використовується в системі, в формальному вигляді.
- ▶ ERD це засіб для комунікації між менеджментом (яка інформація потрібна, щоб бізнес працював), адміністраторами (як управляти інформацією) і архітекторами БД (як ефективно організувати інформацію і виключити її надмірність).
- ▶ Використовує 3 компоненти:
 - ▶ Сутність (Entity)
 - ▶ Атрибут (Attribute)
 - ▶ Зв'язок (Відношення) (Relationship)



Сутність

- Сутність - реальні або абстрактний об'єкт, інформація про який повинна бути зібрана або збережена.



Сутність, атрибути

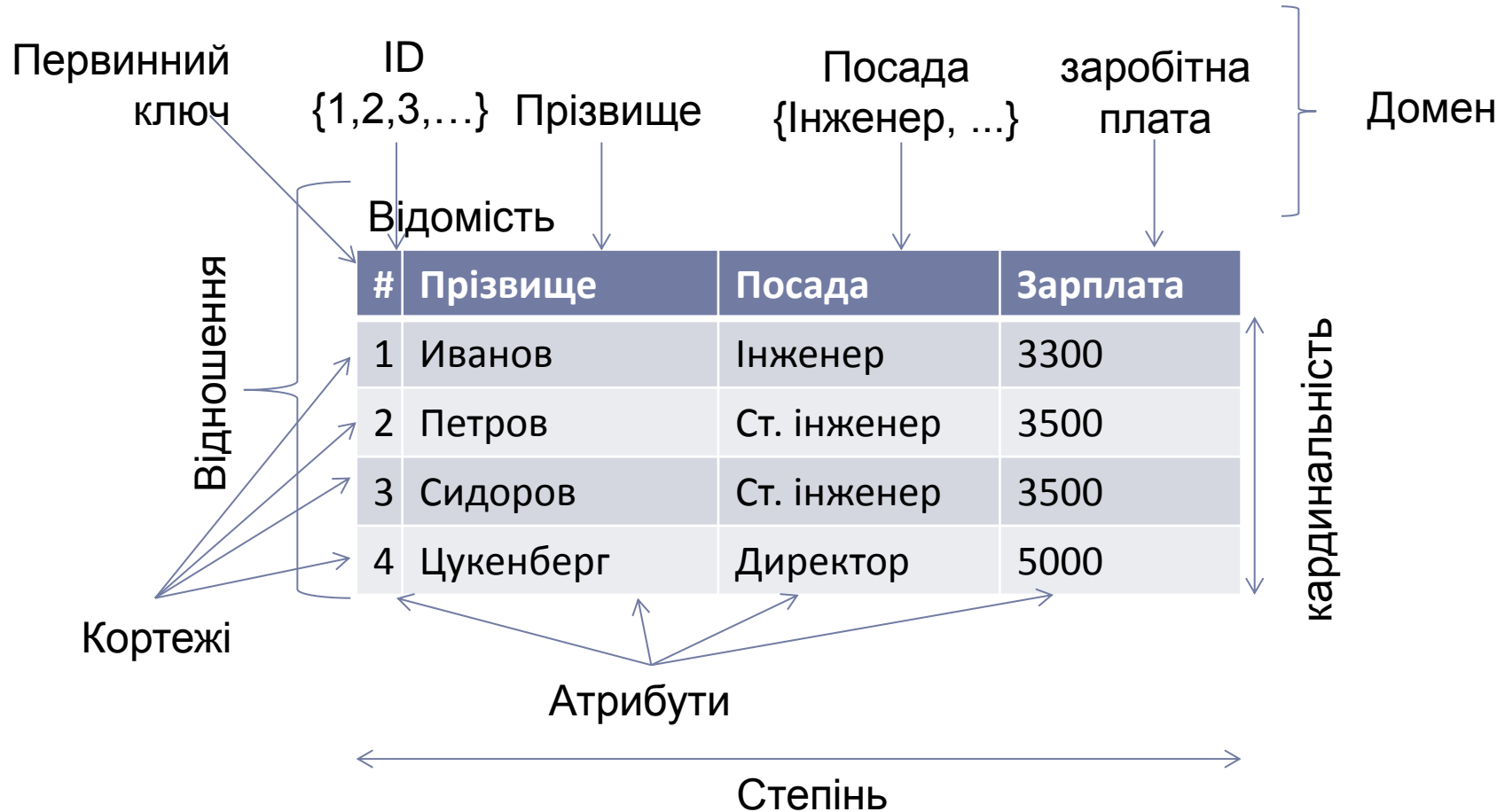
- ▶ Сутність - об'єкт реального світу, який має наступні характеристики:
 - ▶ Унікальний (може бути відділений від усіх інших будь-яким чином);
 - ▶ Грає певну роль в системі, яка моделюється;
 - ▶ Може бути описаний одним або більше елементом інформації (Атрибутом).

Приклад: люди, персонал, події, замовлення, продажі, покупці, постачальники.

- ▶ Атрибут описує деякі властивості сутності.
- ▶ Сутність може мати багато атрибутів, але використовуються тільки ті, які важливі для системи.



Базові терміни



Просто про складне

Формальний термін	Неформальний еквівалент
Відношення	Таблиця
Кортеж	Рядок, запис
Кардинальність	Кількість рядків
Атрибут	Стовпець, поле
Степінь	Кількість стовпців
Первинний ключ	Ідентифікатор
Домен	Область припустимих значень



Ключі

- ▶ Складений ключ, простий ключ;
- ▶ Потенційний ключ;
- ▶ Первинний ключ, альтернативний ключ;
- ▶ Зовнішній ключ.

Потенційний ключ K для відношення R — це підмножина множини атрибутів R , що характеризується такими двома властивостями:

1. Властивість *унікальності*.

Немає двох різних кортежів в R з однаковим значенням K .

2. Властивість *мінімальності* (ненадмірності).

Ніяка з підмножин K не володіє властивістю унікальності.



Primary key

- ▶ Кожне реляційне відношення має тільки 1 первинний ключ, всі інші – альтернативні;
- ▶ Значення всіх атрибутів первинного ключа **не** може бути **не визначено**;
- ▶ Значення первинного ключа не впливає на розташування кортежів в табличному поданні відношення;
- ▶ Первинний ключ не впливає на доступ до атрибутів кортежу.



Зовнішній ключ

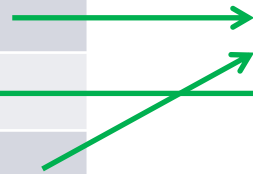
- ▶ Значення зовнішнього ключа завжди посилається на первинний ключ іншого відношення;
- ▶ Значення може бути невизначено і повторюватися в межах одного відношення.

Нерухомість

Адреса	№ Паспорта
Замостянська 5, к301	АА789
Замостянська 5, к301	ББ788
Харківська, 30, к1	АА789
Харківська, 30, к2	
Харківська, 30, к3	ДД786

Власник

№ Паспорта	ПІП
АА789	Іваненко І.І.
ББ788	Петренко П.П.
СС787	Сидоренко С.С.



Зв'язки

- ▶ Встановлюються між сутностями.
- ▶ Характеристики зв'язків:

Сила

- Сильний зв'язок (Identifying Relationship)
- Слабкий зв'язок (Non-identifying Relationship)

Потужність (cardinality)

- 1:1
- 1:N
- M:N

Участь сутності в зв'язку

- Обов'язкова (mandatory)
- Необов'язкова (optional)

Степінь

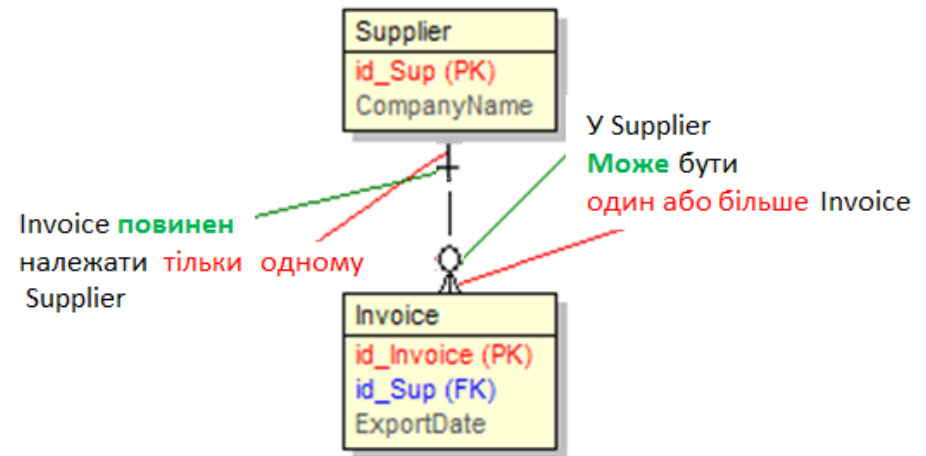
- (кількість асоційованих сутностей)



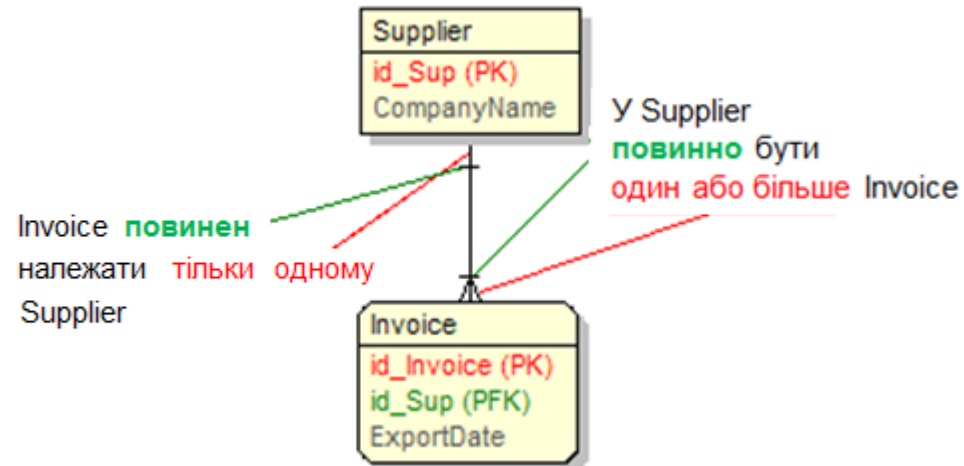
Участь сутності в зв'язку

- Позначається на зв'язку **поперечною лінією** або **окружністю**.
- Поперечна лінія означає **обов'язкову** (mandatory) участь сутності в зв'язку, а **окружність** - **необов'язкове** (optional).
- У разі обов'язкової участі сутності в зв'язку в описі такого зв'язку використовують дієслово **"повинен"**. При необов'язковій участі сутності в зв'язку використовують дієслово **"може"**.

Приклад зв'язку **optional-mandatory**



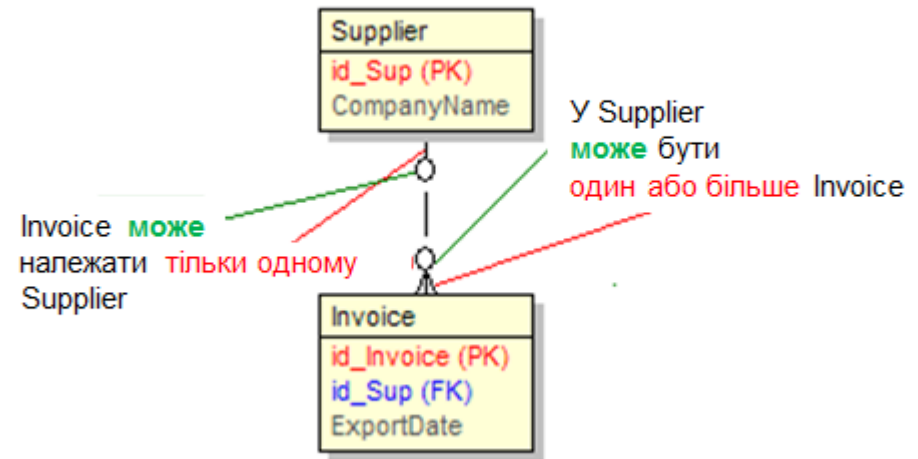
Приклад зв'язку **mandatory-mandatory**



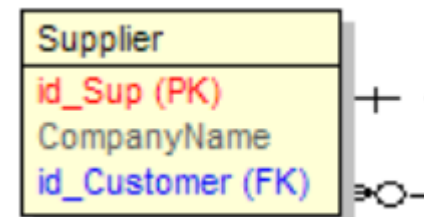
Степінь зв'язку

- ▶ *Степінь зв'язку (relationship degree) вказує на кількість асоційованих сутностей.*
 - ▶ Бінарний зв'язок (binary relationship) описує асоціації двох сутностей.
 - ▶ Тернарний зв'язок (ternary relationship) має місце, коли пов'язуються три сутності.
 - ▶ Унарний зв'язок (unary relationship) описує асоціації всередині однієї сутності.

Приклад зв'язку optional-optional



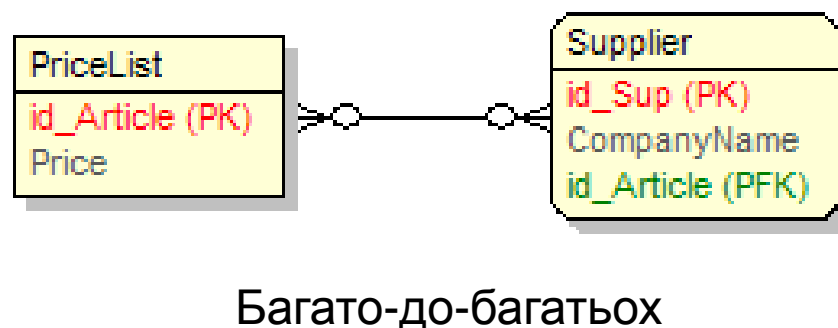
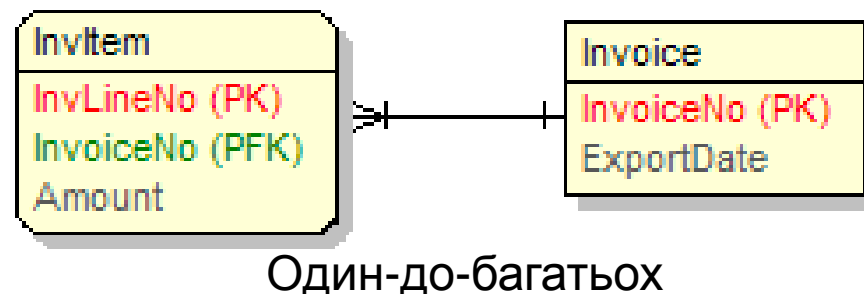
Бінарний зв'язок



Унарний зв'язок

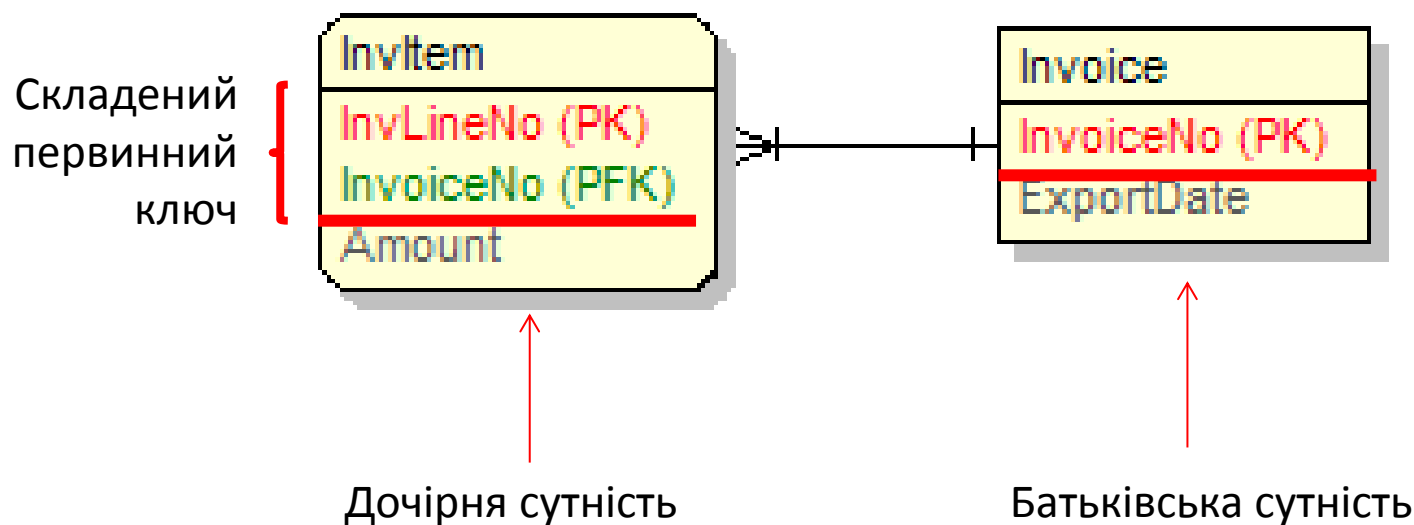
Потужність зв'язку

- ▶ Потужність зв'язку показує, яка кількість екземплярів однієї сутності пов'язана з екземплярами іншої сутності.
- ▶ Потужність може бути:
 - ▶ Один-до-одного (1: 1);
 - ▶ Один-до-багатьох (1: N);
 - ▶ Багато-до-багатьох (M: N).



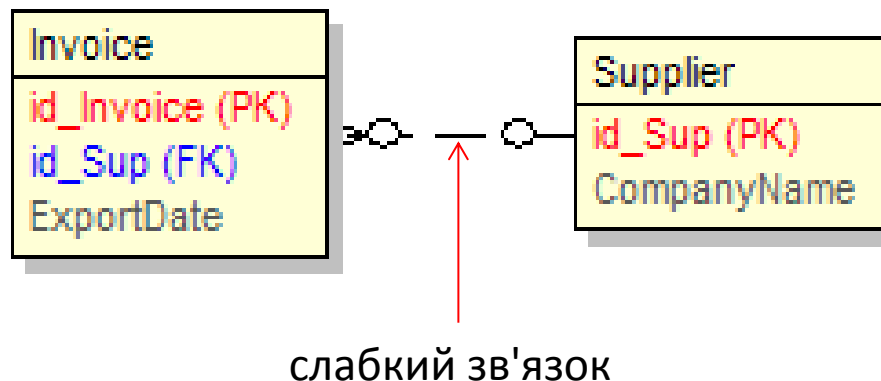
Сильний зв'язок (Identifying Relationship)

- ▶ Дочірня сутність не може існувати без батьківської.
(Не буває відповіді без питання)
- ▶ При цьому **первинний ключ мігрує з батьківської сутності в дочірню, де стає частиною первинного ключа.**



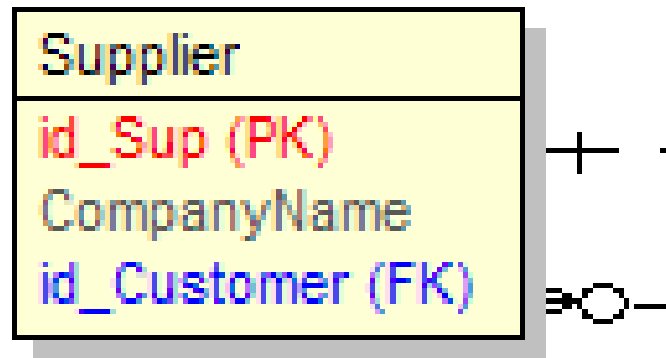
Слабкий зв'язок (Non-identifying Relationship)

- ▶ Батьківська і дочірня сутності пов'язані, але дочірня сутність може бути створена раніше (Вантаж належить відвантаженню, але вантаж може бути на складі, до того як створено замовлення).
- ▶ Первинний ключ мігрує з батьківської сутності в дочірню і **не входить** до складу первинного ключа.



Рекурсивний зв'язок

- ▶ Найчастіше використовується для побудови ієрархій.

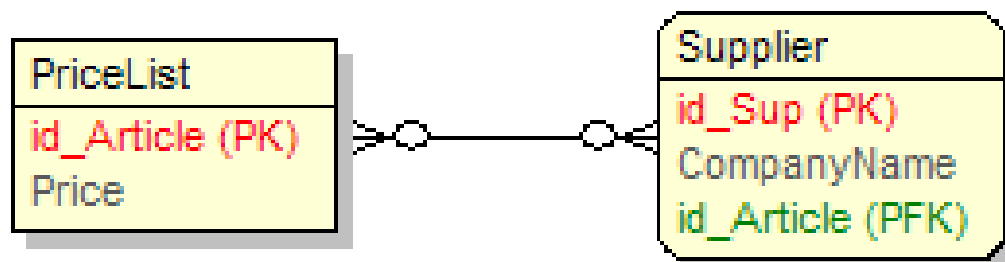


Унарний зв'язок **один-до багатьох**, **mandatory-optional**

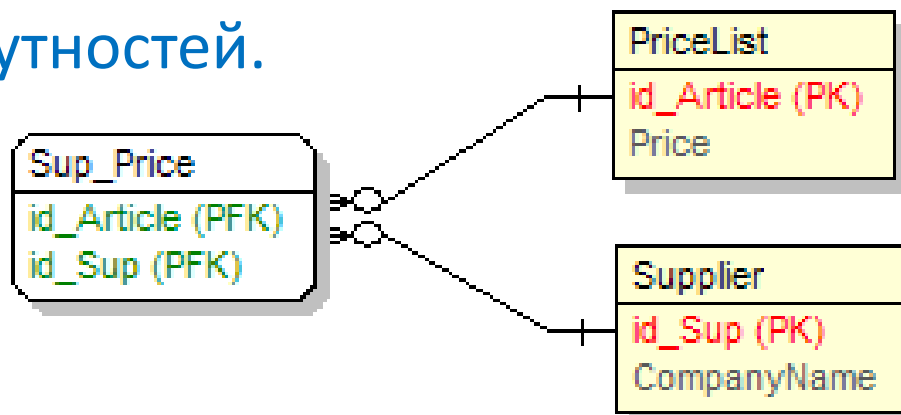
- ▶ Постачальник **може** працювати з **нулем** або **більше** замовників (id_Customer).
- ▶ Замовник **повинен** працювати з одним постачальником (id_Sup).

Зв'язок багато-до-багатьох

- Випадок: постачальники можуть поставляти багато типів товарів. Однакові типи товарів можуть поставляти різні постачальники.



- Неоднозначність зв'язку вирішується введенням **перехідних сутностей**.



Чи всі зв'язки в ERD мають право на життя?

- ▶ Зв'язки один-до одного;
- ▶ Зв'язки один-до-багатьох;
- ▶ рекурсивні зв'язки.

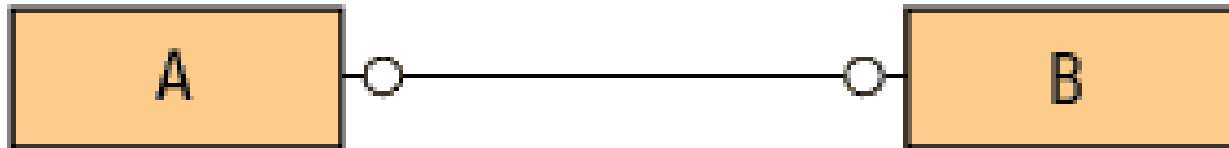
Приклади з <http://ap-che.blogspot.com>



Один-до-одного



Використовується рідко



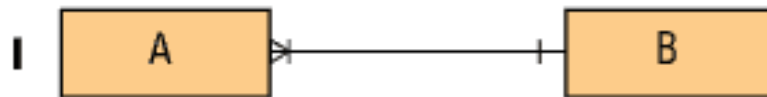
Використовується рідко



Використовується вкрай рідко і
майже завжди помилково



Один-до-багатьох



Багато-до-одного (обов'язкова)



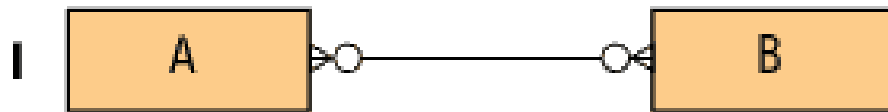
Багато-до-одного (обов'язкова на одному кінці)



Багато-до-одного (необов'язкова)

- ▶ I - досить сильна конструкція, яка передбачає, що входження сутності **B** не може бути створено без одночасного створення щонайменше одного пов'язаного з ним входження сутності **A**.
Найчастіше це невірний зв'язок.
- ▶ II - це форма зв'язку, що зустрічається найчастіше. Вона передбачає, що кожне і будь-яке входження сутності **A** може існувати тільки в контексті одного (і тільки одного) входження сутності **B**. У свою чергу, входження **B** можуть існувати як в зв'язку входженнями **A**, так і без них.
- ▶ III - Як **A**, так і **B** можуть існувати без зв'язку між ними.

Багато-до-багатьох



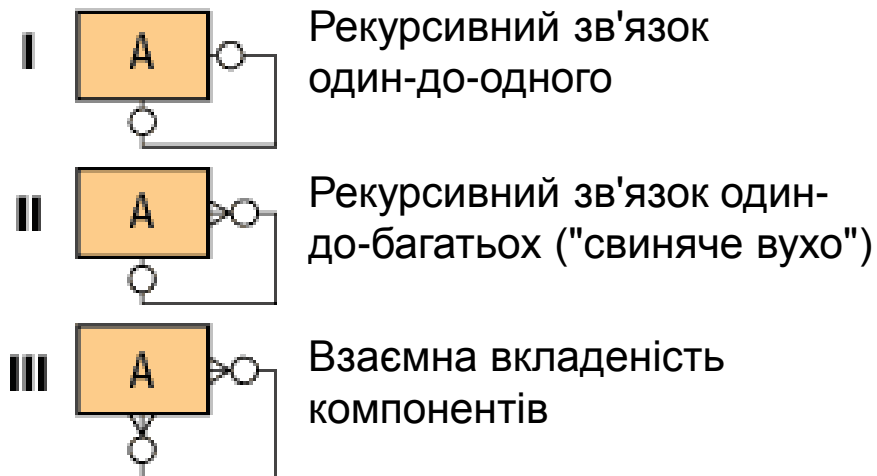
Багато-до-багатьох (необов'язкова)



Багато-до-багатьох (обов'язкова на одному кінці)

- ▶ I - така конструкція частіше буває на початку етапу аналізу і означає зв'язок - або зрозумілий не до кінця і вимагає додаткового вирішення, або відображає просте колективне ставлення - двонаправлений список.
- ▶ II - застосовується рідко. Такі зв'язки завжди підлягають подальшій деталізації.

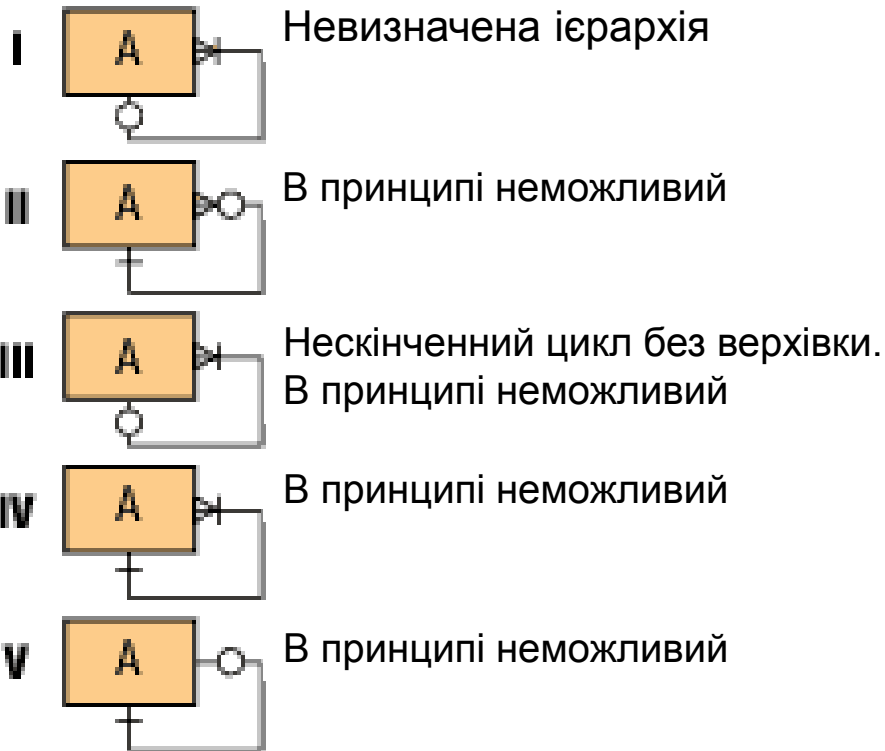
Рекурсивні зв'язки



- ▶ I - рідко, але має місце. Відображає зв'язки альтернативного типу.
- ▶ II - досить часто застосовується для опису ієрархій з будь-якою кількістю рівнів.
- ▶ III - має місце на ранніх етапах. Часто відображає структуру "переліку матеріалів" (взаємна вкладеність компонентів). Приклад: кожен КОМПОНЕНТ може складатися з одного і більше (інших) КОМПОНЕНТІВ і кожен КОМПОНЕНТ може використовуватися в одному і більше (інших) КОМПОНЕНТІВ.



Рекурсивні зв'язки



Домашнє читання

- ▶ Таблиця позначення ключових елементів в різних нотаціях (слайд 28)
- ▶ «ER: діаграми сутність – зв'язь» – Стаття про різні підходи до моделювання систем

Підготовка до практики :

- ▶ Демо-ролик про створення ERD засобами CASE Studio



Notation	Information Engineering	Barker Notation	IDEF1X	UML
Multiplicities:				
- Zero or one				
- One only				
- Zero or more				
- One or more				
- Specific range	N/A	N/A	N/A	
Attributes:				
Names	N/A	Attribute Name: Type	attribute-name: Type	attributeName: Type
Primary key/unique identifier	N/A	# Attribute Name		attributeName <<PK>> {order=#}
Foreign key	N/A	N/A	attribute-name {FK}	attributeName <<FK>> {to=tablename}
Associations:				
Labels				
Entity roles	N/A	N/A	N/A	
Subtyping				
Aggregation				
Composition				
Or Constraint		N/A	N/A	
Exclusive Or (XOR) Constraint			N/A	

Контрольний список питань до сутностей

- ▶ Чи відображає ім'я сутності суть даного об'єкта?
- ▶ Чи немає перетину з іншими сутностями?
- ▶ Чи є хоча б два атрибути?
- ▶ Кількість атрибутів не більше восьми?
- ▶ Чи є синоніми / омоніми даної сутності?
- ▶ Сутність визначена повністю?
- ▶ Чи є унікальний ідентифікатор?
- ▶ Чи є хоча б один зв'язок?
- ▶ Чи існує хоча б одна функція по створенню, пошуку, корегуванню, видаленню, архівуванню та використанню значення сутності?
- ▶ Чи ведеться історія змін?
- ▶ Чи має місце відповідність принципам нормалізації даних?
- ▶ Чи немає такої ж сутності в іншій прикладної системі, можливо, під іншим ім'ям?
- ▶ Чи не має сутність занадто загальний зміст?
- ▶ Чи достатній рівень узагальнення, втілений в ній?



Контрольний список питань до атрибутів

- ▶ Чи є найменування атрибута іменником однини, що відображає суть властивості, що позначається атрибутом?
- ▶ Чи не включає в себе найменування атрибута ім'я сутності (цього бути не повинно)?
- ▶ Чи має атрибут тільки одне значення в кожний момент часу?
- ▶ Чи відсутні значення (або групи), що повторюються?
- ▶ Чи описані формат, довжина, допустимі значення, алгоритм отримання і т.п.?
- ▶ Чи не може цей атрибут бути пропущеною сутністю, яка стала б у пригоді для іншої прикладної системи (що існує або передбачається)?
- ▶ Він не може бути пропущеним зв'язком?
- ▶ Чи немає де-небудь посилання на атрибут як на "особливість проекту", яка при переході на прикладний рівень повинна зникнути?
- ▶ Чи є необхідність в історії змін?
- ▶ Чи залежить його значення тільки від даної суті?
- ▶ Якщо значення атрибута є обов'язковим, чи завжди воно відомо?
- ▶ Чи є необхідність у створенні домену для цього атрибута і йому подібних?
- ▶ Чи залежить його значення тільки від якоїсь частини унікального ідентифікатора?
- ▶ Чи залежить його значення від значень деяких атрибутів, які не включені в унікальний ідентифікатор?



Питання дня

- Опишіть властивості зв'язку:

