



Бази даних та Інформаційні системи

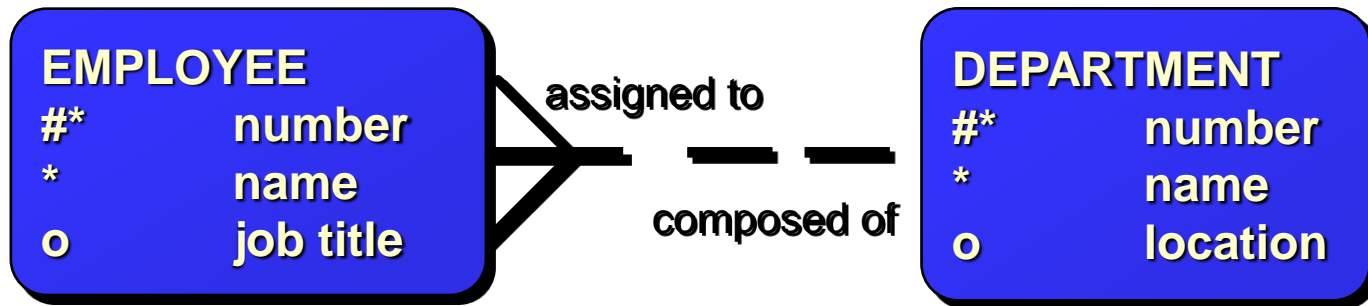
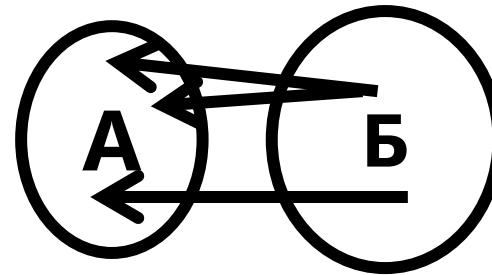
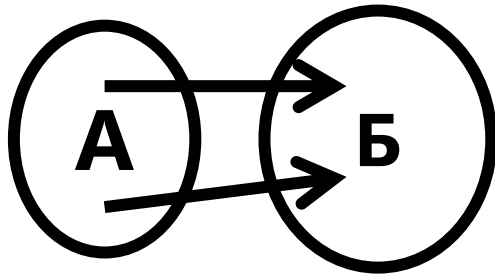
3 Нормалізація

СумДУ, каф. КН
2020

Залежності

- Функціональні

- Багатозначні



Функціональна залежність

- ▶ Найбільше важливі нормальні форми відношень ґрунтуються на фундаментальному в теорії реляційних баз даних понятті функціональної залежності.
- ▶ Нехай задано відношення **R**, яке містить набори атрибутів **A** і **B**.
- ▶ У відношенні **R** набір атрибутів **B** функціонально залежить від **A** і **A** функціонально визначає **B** тоді і тільки тоді, коли в будь-який момент часу кожному значенню проекції **R [A]** відповідає тільки одне значення **R [B]**.
- ▶ Іншими словами, якщо два кортежі збігаються за атрибутами **A**, то вони збігаються і за атрибутами **B**.



Функціональні залежності

- ▶ Функціональна залежність $A \rightarrow B$ називається **повною**, якщо атрибут B не залежить функціонально від будь-якої підмножини A .
- ▶ Функціональна залежність називається **транзитивною**, якщо з функціональних залежностей $A \rightarrow B$ та $B \rightarrow C$ випливає, що $A \rightarrow C$.
- ▶ Атрибут A **багатозначно** визначає атрибут B того ж самого відношення, якщо для кожного значення атрибута A існує добре визначена множина відповідних значень B .



Аномалії

Відношення, які *не перебувають* у відповідній нормальній формі, мають певні небажані властивості, або *аномалії*.

Аномалії - це проблеми, що виникають в даних через дефекти проектування БД.

Аномалії:

- ▶ Додавання;
- ▶ Видалення;
- ▶ Оновлення;



Аномалії додавання(INSERT)

- ▶ *Приклад:* У відношення **співробітник-проект** не можна вставити дані про проект, над яким поки ще не працює жоден співробітник.
- ▶ *Причина аномалії* - зберігання в одному відношенні різнорідної інформації (і про співробітників, і про проекти, і про роботи по проекту).
- ▶ *Висновок:* Логічна модель даних неадекватна моделі предметної галузи. База даних, заснована на такій моделі, працюватиме неправильно.



Аномалії оновлення (UPDATE)

- ▶ *Приклад*: Прізвища співробітників, найменування проектів, номери телефонів повторюються в багатьох кортежах відношення. Тому, якщо співробітник змінює прізвище, або проект змінює найменування, або змінюється номер телефону, то такі зміни необхідно одночасно виконати у всіх місцях, де це прізвище, найменування або номер телефону зустрічаються, інакше відношення стане некоректним (наприклад, один і той же проект в різних кортежах буде називатися по-різному). Таким чином, оновлення бази даних однією дією реалізувати неможливо. Для підтримки відношення в цілісному стані необхідно написати тригер, який при оновленні запису коректно виправляв би дані і в інших місцях.
- ▶ *Причина аномалії*: Надмірність даних через те, що в одному відношенні зберігається різнорідна інформація.
- ▶ *Висновок*: Збільшується складність розробки бази даних. База даних, заснована на такій моделі, працюватиме правильно тільки при наявності додаткового програмного коду у виді тригерів.

Аномалії видалення (DELETE)

- ▶ *Приклад:* При видаленні деяких даних може відбутися втрата іншої інформації. Наприклад, якщо закрити проект "Космос" і видалити всі рядки, в яких він зустрічається, то будуть втрачені всі дані про співробітника Петрова. Якщо видалити співробітника Сидорова, то буде втрачена інформація про те, що у відділі номер 2 знаходиться телефон 33-22-11. Якщо по проекту тимчасово припинені роботи, то при видаленні даних про роботи за цим проектом будуть видалені і дані про сам проект (найменування проекту). При цьому якщо був співробітник, який працював тільки над цим проектом, то будуть втрачені і дані про цього працівника.
- ▶ *Причина аномалії:* зберігання в одному відношенні різнорідної інформації (про співробітників, проекти, роботи по проекту).
- ▶ *Висновок:* логічна модель даних неадекватна моделі предметної галузі. База даних, заснована на такій моделі, працюватиме неправильно.



Нормалізація

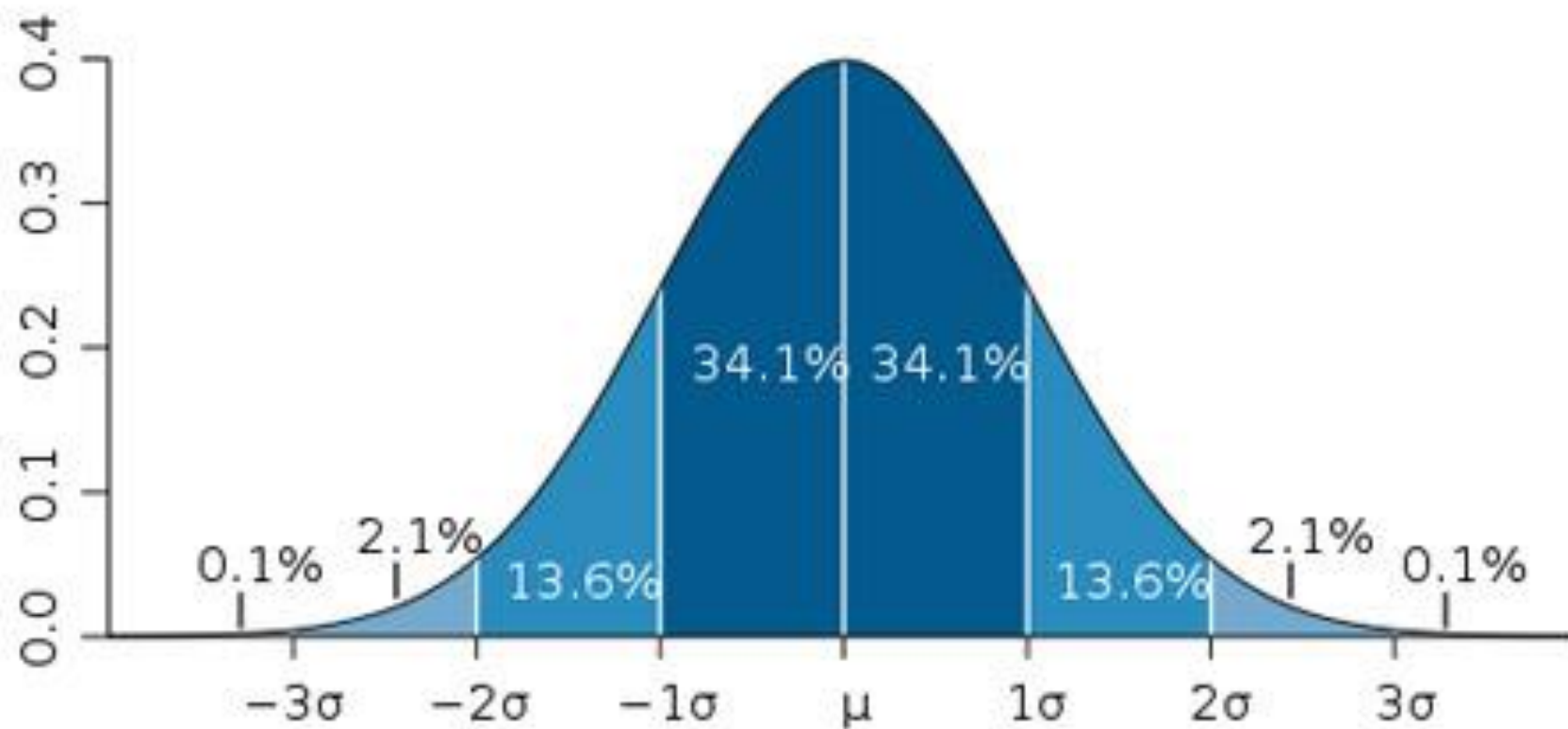
- ▶ Побудова інформаційної моделі предметної галузі і логічної моделі реляційної бази даних - результат рішення комбінаторних задач :
 - ▶ групування атрибутів щодо предметної галузі;
 - ▶ розподіл атрибутів по відношенням бази даних.

Процес усунення потенційної суперечливості і надмірності даних у відношеннях реляційної бази даних називається нормалізацією початкових схем відношень.

Нормалізація — покроковий процес розбиття одного відношення (на практиці таблиці) відповідно до алгоритму нормалізації на декілька відношень на базі функціональних залежностей.



Нормалізація vs Досвід



Перша нормальна форма

- ▶ Відношення знаходиться в першій нормальній формі (1НФ) якщо:
 - ▶ всі атрибути відношення є простими (вимога атомарності атрибутів в реляційної моделі), тобто не мають складових компонентів;
 - ▶ немає груп однотипних атрибутів (категорії даних, що можуть зустрічатись декілька раз).



Перша нормальна форма

Приклад: Приведення відношення до **1НФ**.

Відношення Shipment
(Відвантаження) не є
нормалізованим.
Воно містить групи, що
повторюються і представляють
масив значень: 1st Consignment,
2st Consignment, 3st
Consignment (партії вантажів).

Shipment
<u>Ship Registration Number</u>
<u>Departure Data</u>
Ship Name
Origin
Destination
1st Consignment
2st Consignment
3st Consignment
Customs Value
Ship Capacity
Customs Declaration

Для такого відношення слід ввести бізнес-правило, яке вимагає, щоб вантаж складався не більше ніж з трьох партій!



Перша нормальна форма

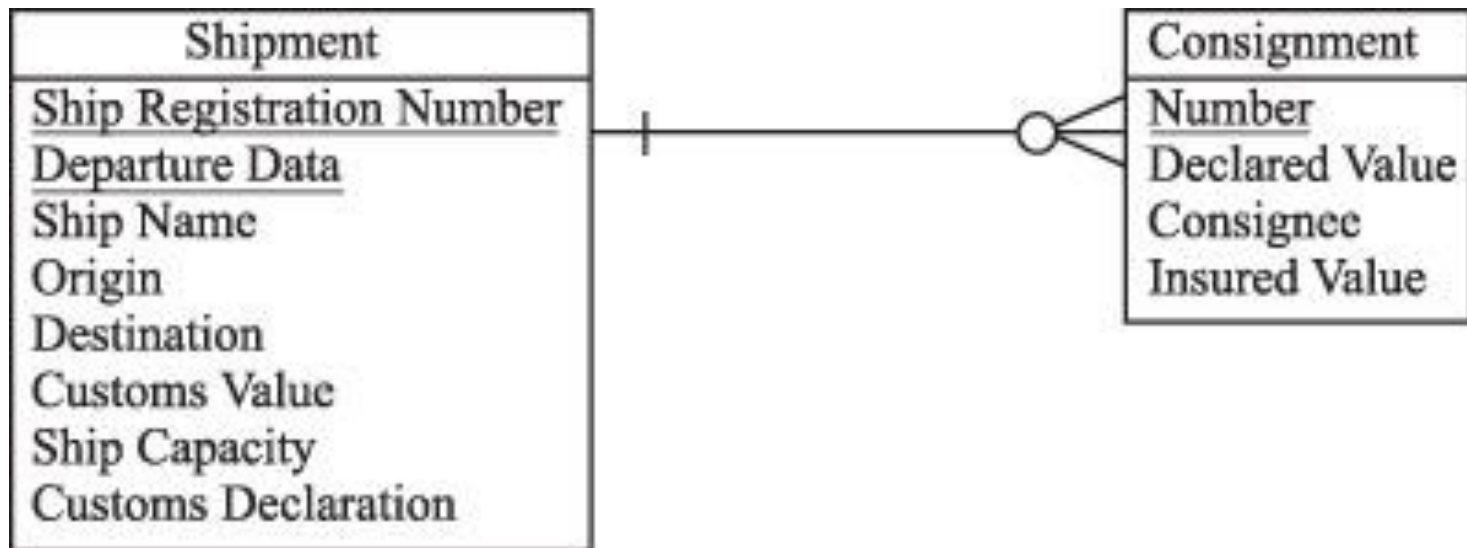
Використання відношення, що представлено не в **1НФ**, може привести до наступних проблем:

- ▶ якщо вантаж анулюється і рядок, пов'язаний з вантажем, видаляється з відношення, то разом з ним видаляється інформація про партії вантажу на борту судна;
- ▶ якщо на склад прибуває нова партія вантажу, і вона ще не включена до складу вантажу, що підлягає відправці, то відомості про партію заносити нікуди;
- ▶ необхідно вводити обмеження: у вантажі не може бути більше трьох партій.

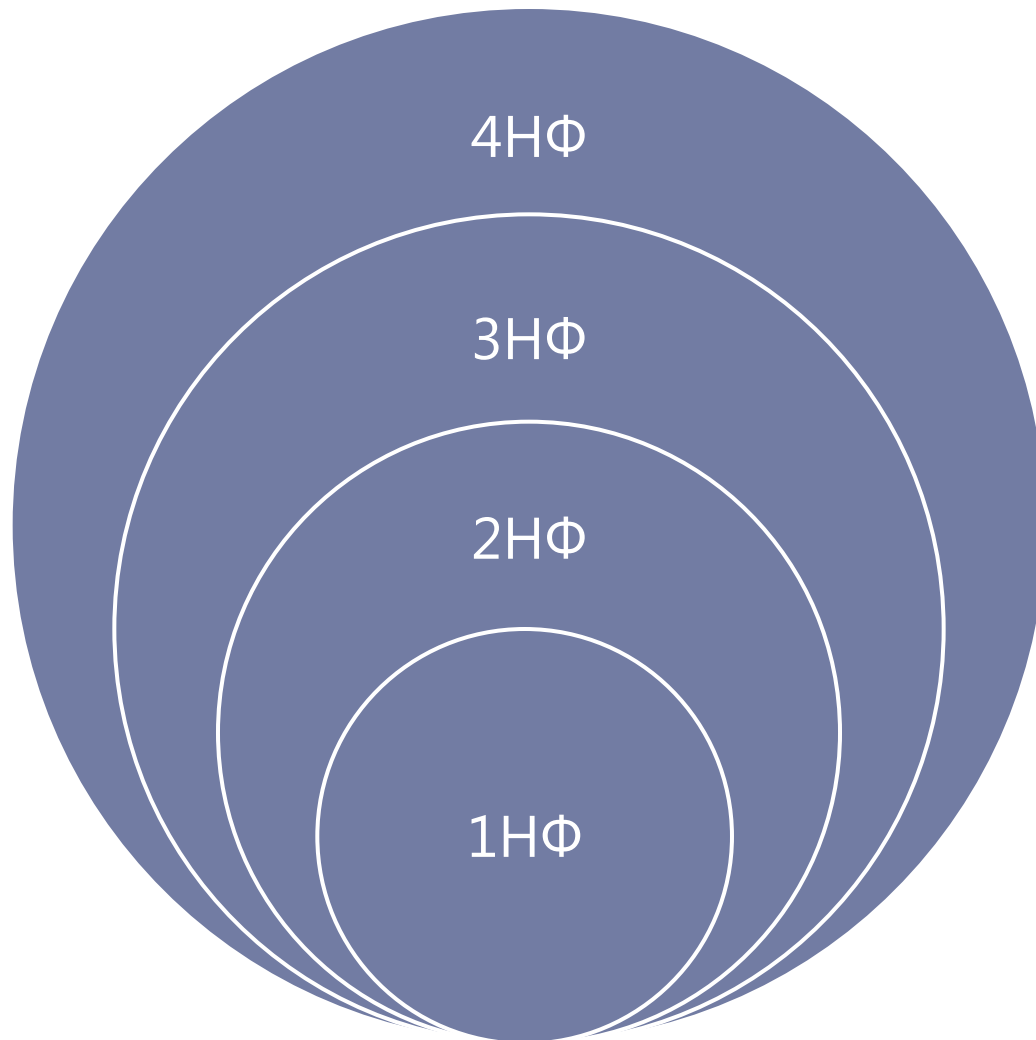
Shipment
<u>Ship Registration Number</u>
<u>Departure Data</u>
Ship Name
Origin
Destination
1st Consignment
2st Consignment
3st Consignment
Customs Value
Ship Capacity
Customs Declaration

Перша нормальна форма

- ▶ Приведення відношення Shipment до **1НФ** полягає у вилученні даних про партії вантажу з відношення Shipment і створення для нього пов'язаного підлеглого відношення Consignment (ПАРТІЯ_ВАНТАЖУ).
- ▶ Результат приведення відношення Shipment до **1НФ** :



Вкладеність нормальних форм



2-а Нормальна форма

- ▶ **Атрибут відношення вважається ключовим, якщо він є елементом будь-якого ключа відношення.** В іншому випадку атрибут буде вважатися неключовим атрибутом.

Наприклад, у відношенні (Місто, Адреса, Поштовий_індекс) всі атрибути є ключовими, оскільки при заданих ФЗ
(місто, адреса) → поштовий_індекс і
поштовий_індекс → місто, ключами є пари (місто, адреса) і (адреса, поштовий_індекс).

- ▶ ***Відношення знаходиться в другій нормальній формі (2НФ), якщо воно знаходиться в 1НФ, і все неключові атрибути відношення функціонально повно залежать від складеного ключа відношення.***



2-а Нормальна форма

Приклад: Приведення відношення до **2НФ**.

Відношення Shipment містить часткову ФЗ:
неключовий атрибут Ship Capacity
(вантажопід'ємність корабля) не залежить від
ключового атрибуту Departure Date (дати
вибуття), а залежить від ключового атрибуту
Ship Registration Number (реєстраційний
номер корабля).

Shipment
<u>Ship Registration Number</u>
<u>Departure Date</u>
Ship Name
Origin
Destination
1st Consignment
2nd Consignment
3rd Consignment
Customs Value
Ship Capacity
Customs Declaration



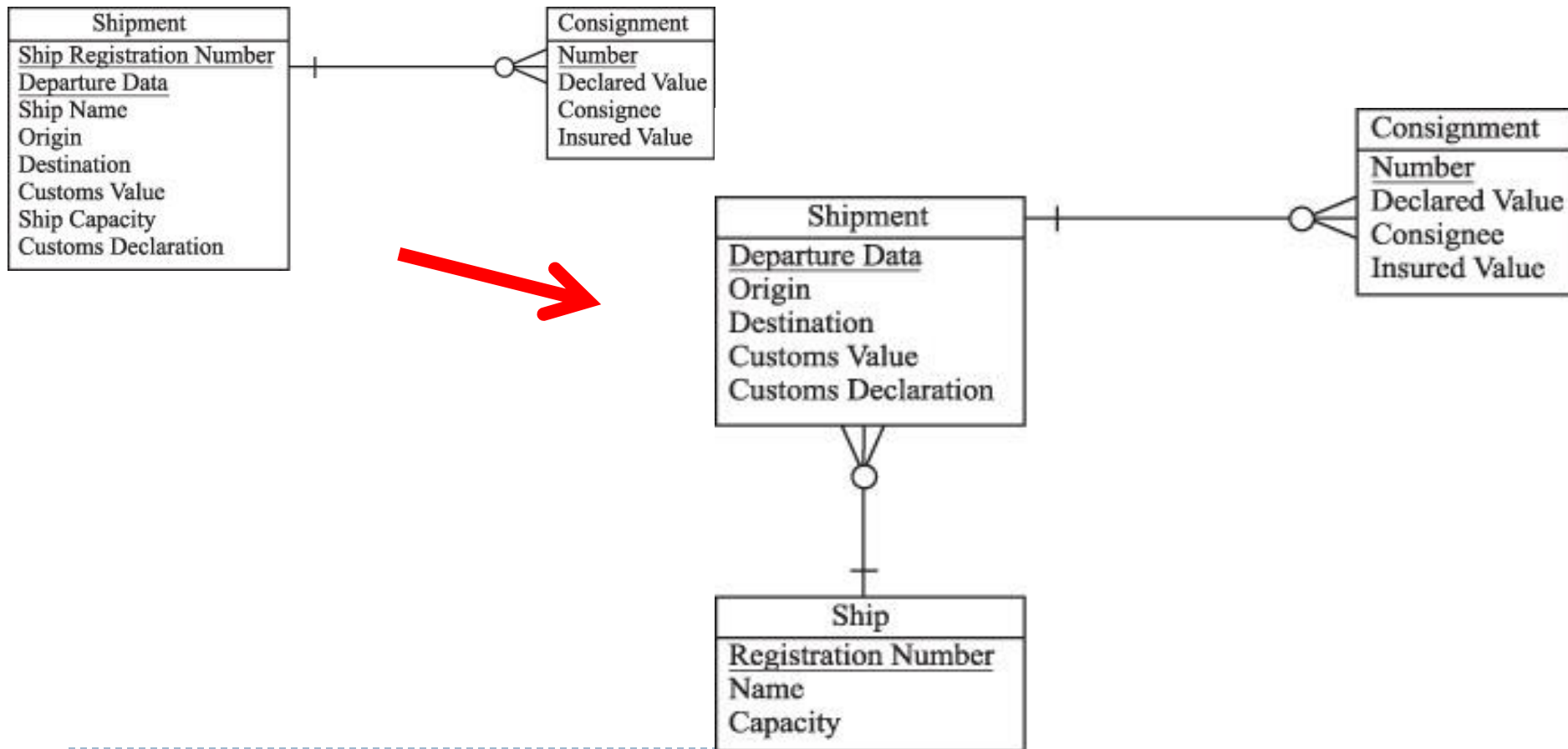
2-а Нормальна форма

- ▶ Використання відношення, яке не приведено до **2НФ**, може привести до наступних проблем:
 - ▶ Неможливо занести в базу даних назву і вантажопід'ємність корабля, який не доставив ще жодного вантажу, - можна тільки ввести для нього фіктивний вантаж;
 - ▶ Якщо видалити кортеж з відношення `Shipment` після відправки вантажу, то втраяться всі дані про кораблі, для яких в даний час немає вантажу;
 - ▶ Неможливо відобразити факт переобладнання корабля і отримання ним нової вантажопід'ємності. Якщо переписати всі попередні кортежі про цей корабель, то вийде, що він в минулому плавав недовантаженим або перевантаженим.



2-а Нормальна форма

- Приведення відношення Shipment до **2НФ** полягає у вилученні атрибутів відношення Shipment і створення для нього пов'язаного підпорядкованого відношення SHIP.



3-я Нормальна форма

Відношення знаходиться в третій нормальній формі (**3НФ**), якщо воно знаходиться у **2НФ**, і всі неключові атрибути відношення залежать тільки від первинного ключа.



3-я Нормальна форма

Можлива наступна проблема:

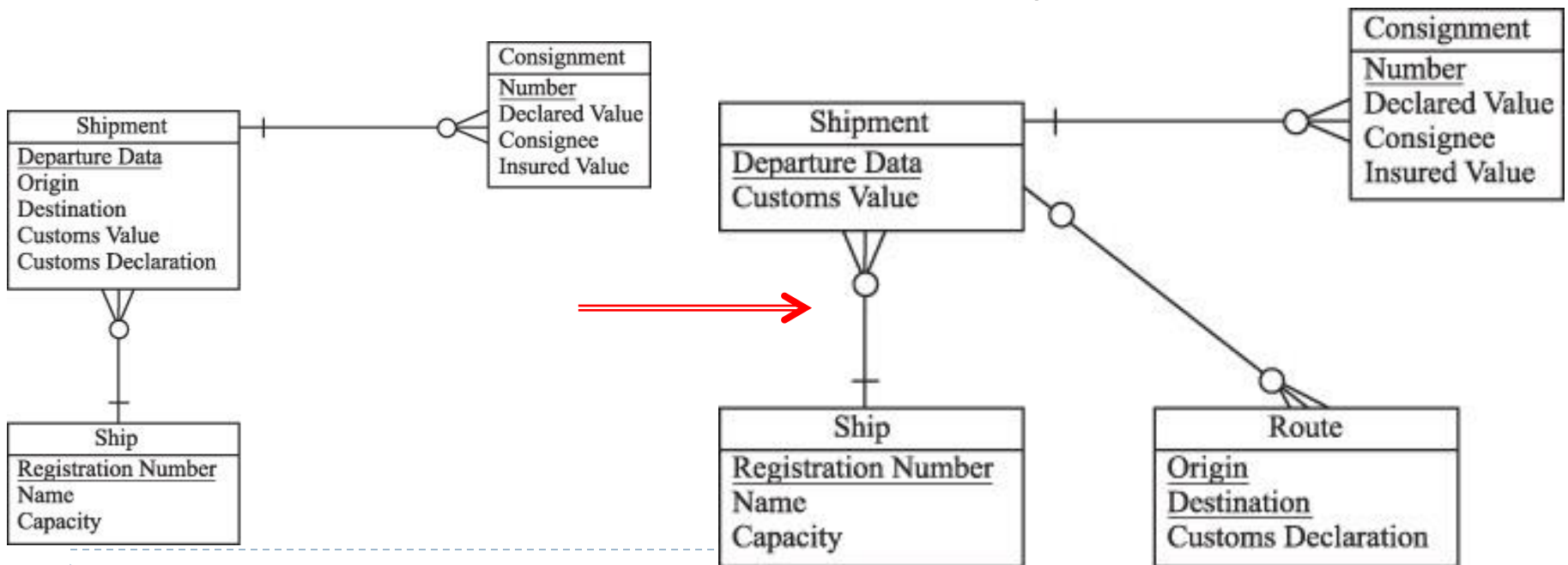
- ▶ Наявність **транзитивної залежності** не дозволяє пов'язати значення **C** і **A**, якщо не існує значення **B**, пов'язаного зі значенням **C**. Це ускладнює додавання і оновлення даних, які необхідно виконати відразу для пари зв'язків, а в разі видалення даних призводить до втрати зв'язку.



3-я Нормальна форма

Приклад: Приведення відношення до 3НФ.

- Відношення `Shipment` містить транзитивну ФЗ: атрибут `Customs Declaration` (митна декларація) є за своєю суттю властивістю атрибутів `Origin` (пункт відправлення) та `Destination` (пункт призначення).



Висновки

(властивості нормальних форм)

- **1НФ** - всі атрибути відношення прості;
- **2НФ** - відношення знаходиться в 1НФ і не містить часткових ФЗ;
- **3НФ** - відношення знаходиться у 2НФ і не містить транзитивних ФЗ від ключа;
- *(Домашнє читання)* **НФБК** - відношення знаходиться в 3НФ і не містить ФЗ ключів від неключових атрибутів;
- *(Домашнє читання)* **4НФ**, застосовується при наявності більш ніж однієї багатозначної ФЗ - відношення знаходиться в НФБК або 3НФ і не містить незалежних багатозначних ФЗ;
- *(Домашнє читання)* **5НФ** - відношення знаходиться в 4НФ і не містить ФЗ по з'єднанню.



Алгоритм нормалізації

Передумова:

- ▶ Всі сутності повинні бути унікально ідентифіковані комбінацією атрибутів і/або зв'язків.

1НФ:

- ▶ Необхідно видалити всі атрибути, які повторюються.

2НФ:

- ▶ Необхідно видалити всі атрибути залежні від частини складеного ідентифікатора.

3НФ:

- ▶ Якщо атрибут залежить від іншого неключового атрибута, їх потрібно перетворити в нову сутність.



Домашнє читання

- ▶ НФБК
- ▶ 4 Нормальна форма
- ▶ 5 Нормальна форма



Нормальна форма Бойса-Кодда

- ▶ **ЗНФ** спрощує вирішення проблем контролю надмірності даних, інтерпретації нуль-значень, контролю за операціями модифікації даних, тільки якщо у відношеннях відсутні будь-які інші ФЗ, зокрема зворотні ФЗ неключового атрибута на один з атрибутів складеного первинного ключа або багатозначні ФЗ.
- ▶ В іншому випадку вище перелічені проблеми залишаються нерозв'язаними. Для усунення таких проблем, пов'язаних з існуванням зворотних ФЗ неключових атрибутів на частину складеного ключа, була запропонована посилена ЗНФ або НФ Бойса-Кодда.



Нормальна форма Бойса-Кодда

Відношення знаходиться в нормальній формі Бойса-Кодда (НФБК), якщо воно знаходиться в 3НФ, і в ньому відсутні залежності ключових атрибутів від неключових атрибутів.

Іншими словами, НФБК допускає наявність лише таких ФЗ, в яких ключ визначає один або більше інших атрибутів.



Нормальна форма Бойса-Кодда

Приклад:

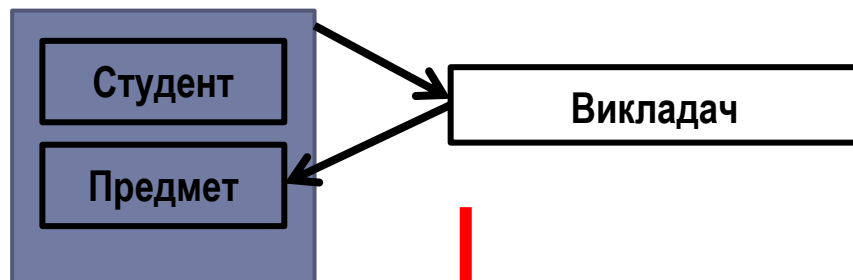
У відношення (Місто, Адреса, Поштовий_індекс), що знаходиться в 3НФ, неможливо записати кортеж для міста з відомим поштовим індексом, якщо не відома адреса з цим поштовим індексом.

Дане відношення не перебуває у НФБК, оскільки має місце ФЗ *Поштовий_індекс* → *Місто*, а атрибут *поштовий_індекс* не є ключем цього відношення.

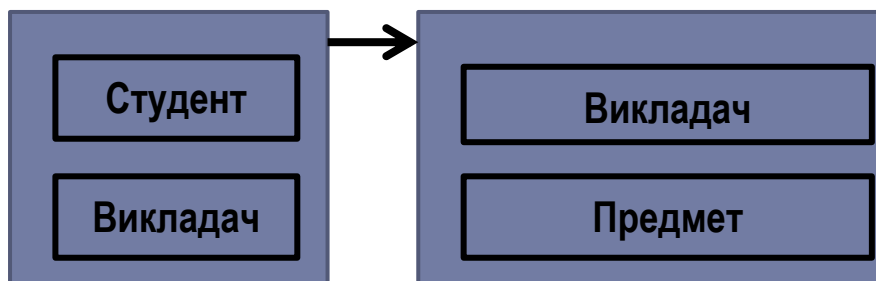


Нормальна форма Бойса-Кодда

Навчання



Викладання



Дисципліна

4-а нормальна форма

- ▶ *Відношення знаходиться в четвертій нормальній формі (4НФ), якщо воно знаходиться в 3НФ або НФБК і всі незалежні багатозначні ФЗ рознесені в окремі відношення з одним і тим самим ключем.*

Іншими словами, **4НФ** застосовується при наявності в відношенні більш ніж однієї багатозначної ФЗ і вимагає, щоб відношення не містило незалежних багатозначних ФЗ.



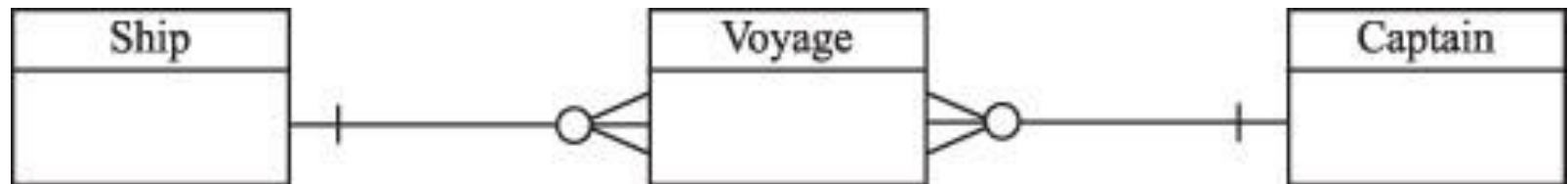
4-а нормальная форма

- ▶ *Приклад:* Приведення до **4НФ**
- ▶ Розглянемо відношення, що містить відомості про кораблі (Ship), рейсах, які вони здійснюють (Voyage) і капітанах (Captain)

Відношення КАПІТАН_КОРАБЕЛЬ_РЕЙС		
Титаник	Іванов	Одеса – Стамбул
Титаник	Петров	Одеса – Стамбул
Титаник	Івлєв	Одеса – Стамбул
Титаник	Прохоров	Одеса – Стамбул
Титаник	Лазарєв	Одеса – Лондон
Титаник	Прохоров	Одеса – Лондон
Потьомкін	Петров	Одеса – Марсель
Потьомкін	Фролов	Одеса – Стокгольм
Потьомкін	Івлєв	Одеса – Стокгольм

Четверта нормальна форма

- ▶ Відношення знаходиться в НФБК і містить тільки багатозначні ФЗ.



- ▶ Однак має місце **аномалія видалення**: якщо капітан Петров піде у відставку і все кортежі про нього будуть видалені, то будуть втрачені відомості про те, що корабель Потьомкін здійснює рейси Одеса - Марсель. Якщо додати новий рейс, то, можливо, доведеться ввести декілька кортежів в відношення.

Четверта нормальна форма

- ▶ Приведення відношення до **4НФ** полягає у виділенні для кожної багатозначною ФЗ свого відношення:



П'ята нормальна форма

- ▶ Нормалізація відношень виконувалася методом розкладання (декомпозиції) схем відношень.
- ▶ Вочевидь, що при такому підході повинен дотримуватися принцип зворотності: з'єднання проєкцій повинно призводити до початкових відношень.
- ▶ Це передбачає відсутність втрати кортежів; появу кортежів, які раніше не існували; збереження ФЗ (семантика взаємозв'язків між даними не повинна порушуватися).



П'ята нормальна форма

- ▶ Декомпозиція схем відношень не завжди гарантує зворотність. Ця обставина пов'язана з існуванням класу ФЗ по з'єднанню.
- ▶ *Якщо відношення задовольняє ФЗ по з'єднанню, то воно може бути відновлено за своїми проекціями.*
- ▶ Відношення, що містять більше трьох багатозначних ФЗ, вимагають особливої уваги при побудові логічної моделі реляційної бази даних. Також 4НФ не усуває надмірність даних повністю, тому потрібно подальша декомпозиція схем відношень.



П'ята нормальна форма

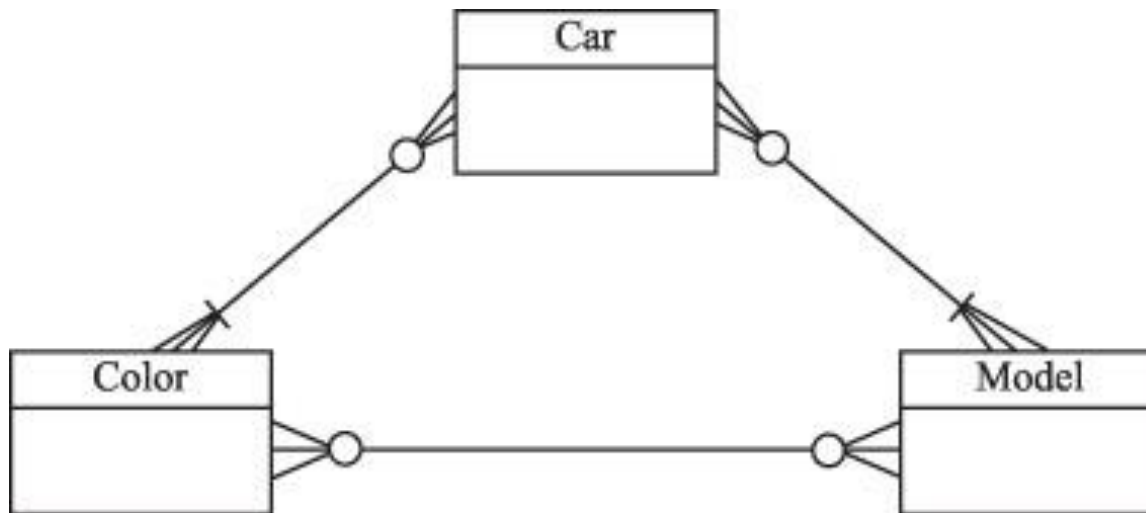
- ▶ *Відношення знаходиться в п'ятій нормальній формі (5НФ), якщо воно знаходиться в 4НФ і задовольняє залежності по з'єднанню щодо своїх проекцій.*
- ▶ **5НФ** називають також нормальною формою з проектуванням з'єднань. Вона використовується для вирішення трьох і більше відношень, які пов'язані більш ніж трьома ФЗ за типом «багато-до-багатьох».



П'ята нормальна форма

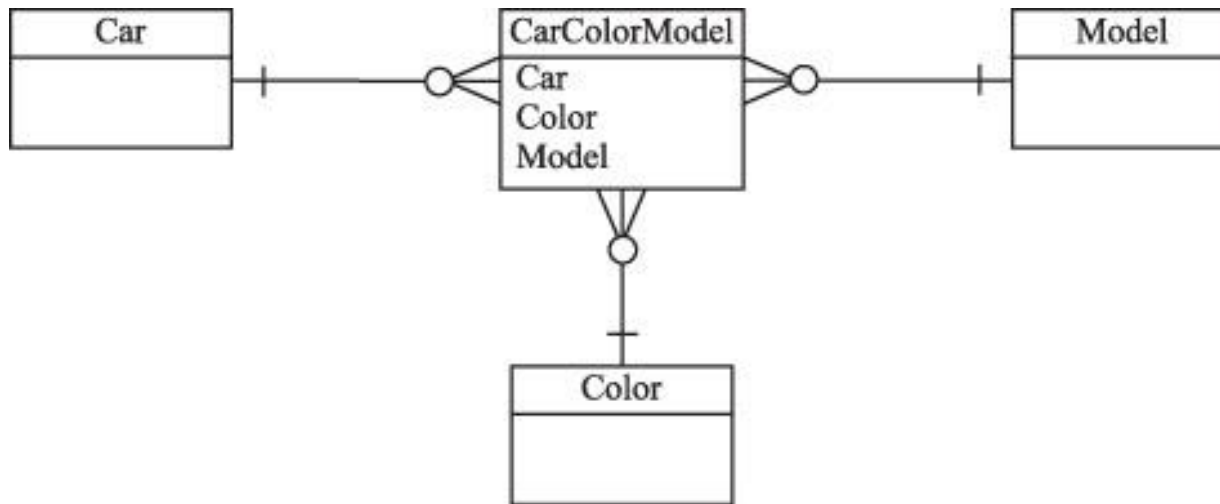
Приклад: приведення до **5НФ**.

Відношення з декількома багатозначними ФЗ :



П'ята нормальна форма

Приведення відношення до **5НФ** полягає у доданні ще одного відношення, що зв'язує три вихідних відношення:



Процедура приведення відношення, що містить багатозначні ФЗ, до **5НФ** полягає в побудові відношення, яке позв'язує і дозволяє виключити появу в з'єднаннях помилкових кортежів.

