



Бази от данни

Релационен модел на данни. Нормализация.

Дизайн на БД. Аномалии

- ▶ Когато създаваме релационен модел на БД, може да се получат следните аномалии:
 - ▶ **Излишества** – когато се повтаря информацията без да е необходимо
 - ▶ **Аномалии при обновяване** – когато направим промени в един кортеж, а забравим да направим същите промени в свързаните с него кортежи
 - ▶ **Аномалии при изтриване** – когато изтрием даден ред и това доведе до загуба на информация
 - ▶ Например ако имаме актьор, който участва във филм и ние искаме да изтрием актьора, а като ефект да получим и изтрита информация за филма, в който е участвал актьора.

StarMovies(title, year, length, filtype, studioName, starName)

Декомпозиране на релации

- ▶ Начин да избегнем излишества е чрез декомпозиране на релации.
- ▶ Декомпозирането на една релация R означава разделяне на атрибутите на релацията R в две нови релации
- ▶ Например: Нека ни е дадена релацията $R(A_1, A_2, \dots, A_n)$, можем да декомпозираме релацията R до две релации $S(B_1, B_2, \dots, B_m)$ и $T(C_1, C_2, \dots, C_k)$, такива че:
 - ▶ $\{A_1, A_2, \dots, A_n\} = \{B_1, B_2, \dots, B_m\} \cup \{C_1, C_2, \dots, C_k\}$
 - ▶ Кортежите в релацията S са проекция по атрибутите B_1, B_2, \dots, B_m на релацията R , а тези на релацията T по атрибутите C_1, C_2, \dots, C_k на релацията R

Декомпозиране на релации

- ▶ Проекция по атрибутите B_1, B_2, \dots, B_m на релацията R означава, че за всеки кортеж от релацията R вземаме само компонентите (стойностите) на кортежите на релацията в атрибутите B_1, B_2, \dots, B_m . Тези компоненти формират кортежите на новата релация S .
- ▶ Понеже релациите са множества е възможно при проекцията на релацията R до релацията S по атрибутите B_1, B_2, \dots, B_m да получим от два различни кортежа за релацията R , два еднакви кортежа в релацията S . Ако се получи така в релацията S попада само един кортеж (защото е множество)
- ▶ Пример: Нека разгледаме релацията
`StarMovies(title, year, length, filtype, studioName, starName)`

Декомпозиране на релации

- ▶ StarMovies(title, year, length, filtype, studioName, starName)
- ▶ Очевидно може да се получат излишества в нея.
- ▶ Ще я декомпозираме до две релации:
 - ▶ Movies1(title, year, length, filtype, studioName)
 - ▶ StarMovieI (title, year, starName)
- ▶ Ако разгледаме ново-получените релации, лесно се установява, че проблемите с излишествата са разрешени.
- ▶ Ако в първия вариант StarMovies са възможни и излишества и аномалии при изтриване и аномалии при обновяване, то при декомпозираните релации това не може да се получи.
- ▶ Дайте пример за аномалии при релацията:
StarMovies(title, year, length, filtype, studioName, starName)?

Декомпозиране на релации - Аномалии

StarMovies(title, year, length, filtype, studioName, starName)

- ▶ **Излишество:** Ако в един филм участват десет актьора, за всеки актьор ще се повтаря данните за филма
- ▶ **Аномалия при обновяване:** Както казахме ако в един филм участват десет актьора, за всеки актьор ще повтаря данните за филма. Ако обновим в един от тези кортежи например дължината на филма, а забравим да го направим в останалите девет това е аномалия!
- ▶ **Аномалия при изтриване:** Аналогично, ако в един филм участва един актьор и ние искаме да изтрием актьора, това ще доведе до загуба за информацията за филма.

Декомпозиране на релации - Аномалии

Да разгледаме декомпозираните релации:

- ▶ **Movies1**(title, year, length, filtype, studioName)
- ▶ **StarMovie I**(title, year, starName)

При тях възможни ли са изброените по-долу аномалии?

- ▶ **Излишество**: Ако в един филм участват десет актьора, за всеки актьор ще повтаря данните за филма (**X**)
- ▶ **Аномалия при обновяване**: Ако обновим в един от повтарящите се за филма кортежи, например дължината на филма, а забравим да го направим в останалите девет това е аномалия (**X**)
- ▶ **Аномалия при изтриване**: Ако в един филм участва един актьор и ние искаме да изтрием актьора, това ще доведе до загуба за информацията за филма (**X**)

Нормална форма на Бойс-Код

- ▶ Целта на декомпозицията е да замени релацията с други, които да не позволяват аномалии. Има просто правило, което гарантира, че ако една релация е подчинена на него то в съответната релация не може да възникнат аномалии.
- ▶ Казваме, че една релация се намира в Нормална форма на Бойс-Код (НФБК), тогава и само тогава когато ако $A_1, A_2, \dots, A_n \rightarrow B$ е нетривиална ФЗ, която е в сила за R , то $\{A_1, A_2, \dots, A_n\}$ е супер-ключ за R
- ▶ Или казано по друг начин лявата страна на всяка нетривиална ФЗ, която е в сила за R трябва да съдържа ключ на релацията R
- ▶ Ако се случи, че има ФЗ, които са в сила за релацията но не отговарят на посоченото по-горе правило (НФБК), тогава трябва да декомпозираме за да приведем релацията до НФБК

Нормална форма на Бойс-Код

- ▶ Пример: Релацията

StarMovies(title, year, length, filtype, studioName, starName)

- ▶ Не е в НФБК

- ▶ За да проверим това, нека първо да определим кой е ключът на релацията

- ▶ Ключът е {title, year, starname}
- ▶ Тогава всяко множество от атрибути, което съдържа тези три атрибута е супер-ключ
- ▶ За релацията е в сила ФЗ:
 - ▶ title, year - \rightarrow length, filmtype, studioName
 - ▶ Също така знаем че title, year не определя функционално starname
 - ▶ Ако релацията беше в НФБК, то в лявата част на ФЗ щеше да бъде супер-ключ
 - ▶ В случая това не е така

Нормална форма на Бойс-Код

- ▶ Пример2: Релацията

`Movies1(title, year, length, filtype, studioName)`

- ▶ Тя е в НФБК

- ▶ За нея е в сила ФЗ

`title, year - > length, filmtypе, studioName`

- ▶ Лявата част съдържа ключа, следователно релацията е в НФБК

Твърдение

- ▶ Всяка релация, която има само два атрибута се намира в НФБК
- ▶ Доказателство:
- ▶ Нека $R(A,B)$
 - ▶ Ако нямаме нетривиални ФЗ, т.е. всички ФЗ които се в сила за релацията са тривиални, тогава релацията се намира в НФБК, тъй като само не тривиалните ФЗ могат да нарушат правилото на НФБК. Единственият ключ за релацията ще бъде $\{A,B\}$
 - ▶ В случай че $A \rightarrow B$, но не е в сила че $B \rightarrow A$, тогава A е единственият ключ за релацията и всички нетривиални ФЗ, които са в сила за релацията го съдържат в лявата си част, т.е. релацията е в НФБК
 - ▶ Аналогично ако $B \rightarrow A$, но не е в сила че $A \rightarrow B$
 - ▶ В случай че $A \rightarrow B$ и $B \rightarrow A$, тогава и атрибута A и атрибута B могат да бъдат ключ за релацията. Всички ФЗ съдържат ключ в лявата си част на правилото, следователно релацията е в НФБК

Твърдение - пояснение

- ▶ Ако има повече от един ключ в релацията, НФБК изисква някакъв ключ да се съдържа в лявата част на правилото на всички ФЗ, които са в сила за релацията (не първичен ключ)
- ▶ Пример за такива атрибути в релация са – ID, EGN, SSN
- ▶ Всяка релационна схема може да бъде сведена до НФБК, чрез подходящи декомпозиции, така че:
 - ▶ Декомпозираните релации да бъдат също в НФБК
 - ▶ Данните в оригиналната релация са представени напълно достоверно в декомпозираните релации, т.е. да можем да „възстановим“ данните в оригиналната релация от данните в декомпозираните релации
- ▶ От правилото за релациите с два атрибути, може едва ли не да заключим, че е напълно достатъчно да сведем всяка релация до множество от релации с по два атрибута. Това не е достатъчно, защото второто условие за възстановяване няма да бъде изпълнено.

Алгоритъм за свеждане към НФБК

- ▶ Първо се търсят нетривиални ФЗ: $A_1, A_2, \dots, A_n \rightarrow B_1, B_2, \dots, B_m$ такива че да нарушават НФБК, т.е. лявата част да не съдържа ключа
- ▶ По правилото за комбиниране добавяме в дясната част всички атрибути, които са функционално определени от A_1, A_2, \dots, A_n
- ▶ Правим декомпозиция на релацията на две релации.
 - ▶ Първата релация съдържа атрибутите от лявата част на функционалната зависимост и тези атрибути, които не са функционално определени от тях
 - ▶ Втората релация съдържа атрибутите от лявата част на ФЗ и атрибутите от дясната част

Алгоритъм за свеждане към НФБК - Пример

StarMovies(title, year, length, filtype, studioName, starName)

- ▶ ФЗ, която нарушава правилото е

title, year - > length, filmtypе, studioName

- ▶ По горе-споменатият алгоритъм разделяме релацията на две релации
 1. Movies1(title, year, length, filtype, studioName)
 2. StarMovies(title, year, starname)

Възстановяване на информацията след декомпозиция

- ▶ Нека разгледаме релацията $R(A, B, C)$ и ФЗ: $B \rightarrow C$, която допускаме, че нарушава НФБК.
- ▶ Има два варианта:
 1. Да съществува ФЗ $A \rightarrow B$, в този случай $\{A\}$ е единственият ключ. Лявата страна на ФЗ $B \rightarrow C$ не е супер ключ, т.е. релацията не е в НФБК.
 2. ФЗ $B \rightarrow C$ да е единствената нетривиална ФЗ. В този случай единственият ключ е $\{A, B\}$ и отново лявата страна на ФЗ $B \rightarrow C$ не е супер-ключ.
- ▶ Декомпозицията базирана на ФЗ $B \rightarrow C$, разделя атрибутите на две схеми $\{A, B\}$ и $\{B, C\}$. Нека t е кортеж от R , така че $t = (a, b, c)$. Като приложим декомпозицията t ще се проектира като (a, b) за схемата $\{A, B\}$ и (b, c) за схемата $\{B, C\}$
- ▶ Възможно е да съединим кортежа от $\{A, B\}$ с кортежа от $\{B, C\}$ по съвпадащите стойности на атрибута B . При съединението на двата кортежа се получава първоначалния кортеж $= (a, b, c)$.

Възстановяване на информацията след декомпозиция

- ▶ Не можем да сме сигурни обаче, че R може да бъде наистина възстановена след декомпозицията. Например нека да вземем кортежите $t=(a, b, c)$ и $v=(d, b, e)$
- ▶ Ако проектираме кортежите по двете схеми получаваме за кортежа t - (a, b) (b, c) , съответно за схемите $\{A, B\}$ и $\{B, C\}$ и за кортежа v (d, b) (b, e) съответно за схемите $\{A, B\}$ и $\{B, C\}$
- ▶ Ако се опитаме да възстановим първоначалната релация обаче и съединим проектираните кортежи по съвпадащите стойности на атрибута B получаваме кортежа (a, b, e) , който реално не е кортеж на релацията R
- ▶ Докато е в сила обаче ФЗ: $B \rightarrow C$, такъв кортеж не можем да получим. Ако кортежите t и v съвпадат в атрибута B , те трябва да съвпадат и в C . Следователно самата релация R не може да съществуват такива кортежи като t и v .

Възстановяване на информацията след декомпозиция

- ▶ **Твърдение:** за всяка ФЗ, която нарушава НФБК, ако декомпозираме релацията по метода за декомпозиция до НФБК, оригиналната релация може да бъде точно възстановена чрез съединение на проектираните кортежи от новите релации по всички възможни начини.
- ▶ Ако направим декомпозиция, без да спазваме метода за декомпозиция, получените проекции на кортежи няма да могат да бъдат възстановени до оригиналната релация.
- ▶ Пример $R(A, B, C)$ и ФЗ $B \rightarrow C$ не е в сила за R . Тогава

(1, 2, 3) (1, 2, 5) (4, 2, 3) (4, 2, 5)

A	B	C
1	2	3
4	2	5

A	B
1	2
4	2

B	C
2	3
2	5

Нормализация

- ▶ **Нормализация** наричаме процеса на декомпозиране на релация на няколко релации с цел отстраняване на аномалии и повторения.
- ▶ **ФЗ:** Нека $R(A, B, C)$ е релация. Функционална зависимост (ФЗ) за релацията R , наричаме твърдение от вида $A, B \rightarrow C$
- ▶ **Пълна ФЗ (ПФЗ)** – Нека $R(A, B, C)$ е релация. ПФЗ за релацията R , наричаме твърдение от вида $A, B \rightarrow C$, при която атрибута C зависи функционално от (A, B) , но не и от под-множество на $\{A, B\}$.
- ▶ **Частична ФЗ (ЧФЗ)** - Нека $R(A, B, C)$ е релация. ЧФЗ за релацията R , наричаме твърдение от вида $A, B \rightarrow C$, при която атрибута C зависи функционално от (A, B) , но и ФЗ от вида $A \rightarrow C$ или $B \rightarrow C$ също са в сила

Нормални форми

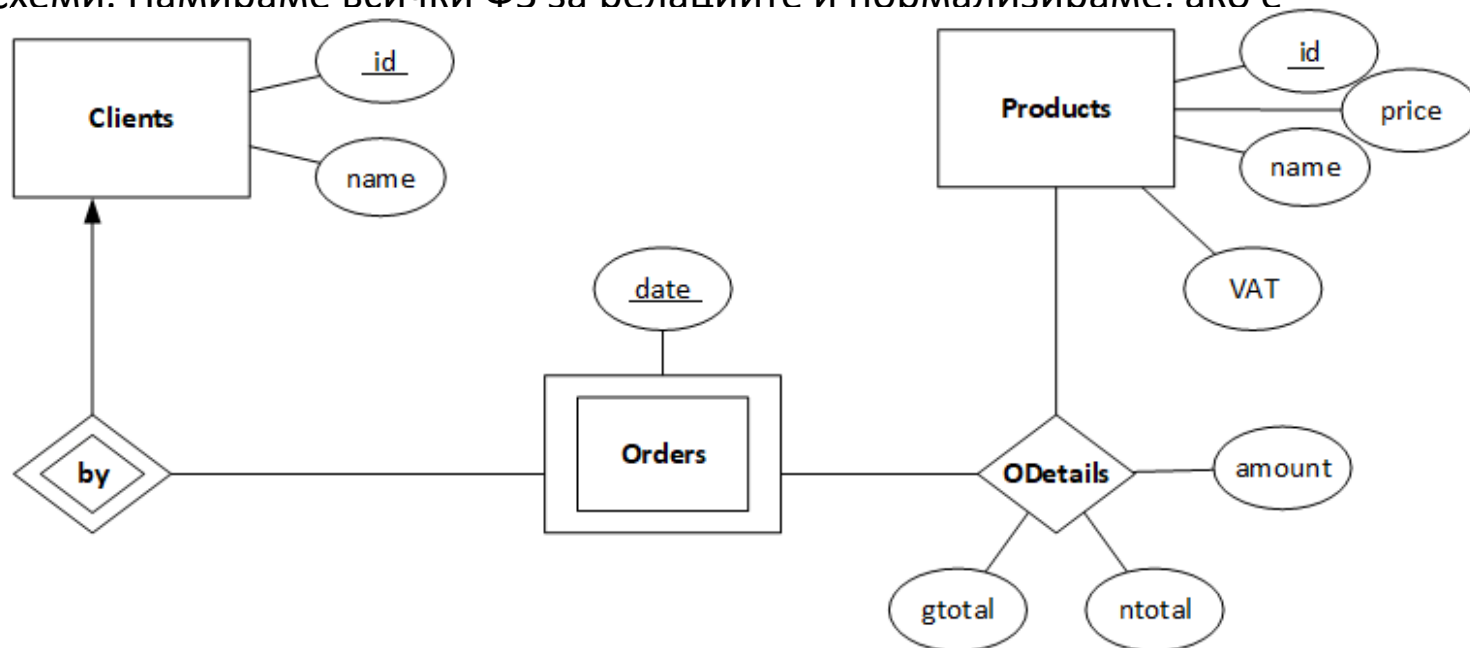
- ▶ **Първа нормална форма (1НФ)** – изисква всеки компонент в кортежите на релацията да има атомарна стройност.
- ▶ **Втора нормална форма (2НФ)** – изисква релацията да е в 1НФ, да няма атрибути, в които да се пазят изчислими стойности, както и всеки атрибут в релацията да е функционално зависим от атрибутите, съставляващи първичния ключ, но да не и от негово подмножество.
- ▶ **Трета нормална форма (3НФ)** – изисква релацията да е в 2НФ и ако $A_1, A_2, \dots, A_n \rightarrow V$ е нетривиална ФЗ която е в сила за R , то или $\{A_1, A_2, \dots, A_n\}$ да е супер-ключ за R или V да е част от ключ.
- ▶ **Нормална форма на Бойс-Код (НФБК)** – изисква релацията да е в 3НФ и само ако $A_1, A_2, \dots, A_n \rightarrow V$ е нетривиална ФЗ, която е в сила за R , то $\{A_1, A_2, \dots, A_n\}$ да е супер-ключ за R .

Задача

- ▶ Да се направи дизайн на БД, съдържаща информация за поръчки.
Известно ни е, че в БД ще се пази информация за дата на поръчката, номер на продукт – уникален за всеки продукт, име на продукт, цена на продукт, ДДС за съответния продукт, номер на клиент – уникален за всеки клиент, име на клиент, количество, което клиентът е поръчал от продукта в съответната поръчка, сума на поръчката без ДДС, сума на поръчката с ДДС.
- ▶ В сила са следните предположения:
 - ▶ Стойността на ДДС за продукта може да варира за различните продукти.
 - ▶ Поръчките на клиентите, направени в един и същ ден (на една и съща дата) се обединяват, т.е. има само една поръчка за клиент за дадена дата.
 - ▶ $\text{сума на поръчката с ДДС} = \text{сума на поръчката без ДДС} + \text{ДДС}$
- ▶ 1. Направете модел на БД
- ▶ 2. Определете ключовете за релациите (релацията).

Решение – Подход 1

Подход 1: Правим E/R модел по описаната информация и преобразуваме до релационни схеми. Намираме всички ФЗ за релациите и нормализираме, ако е необходимо.



Преобразуваме:

- ▶ Clients(cID, cName);
- ▶ Products(pID, pPrice, pName, VAT);
- ▶ Orders(cID, oDate) ✗
- ▶ ODetails(cID, oDate, pID, amount, **ntotal**, **gTotal**) = ODetails(cID, oDate, pID, amount)

Решение – Подход 1

Преобразуван модел в релационни схеми:

- ▶ Clients(cID, cName); **PK - cID**
- ▶ Products(pID, pPrice, pName, VAT); **PK – pID**
- ▶ ODetails(cID, oDate, pID, amount); **PK – cID, pID, oDate**

Намираме всички ФЗ за релациите в модела:

1. cID -> cName;
2. pID -> pName, pPrice, VAT;
3. cID, pID, oDate -> amount;

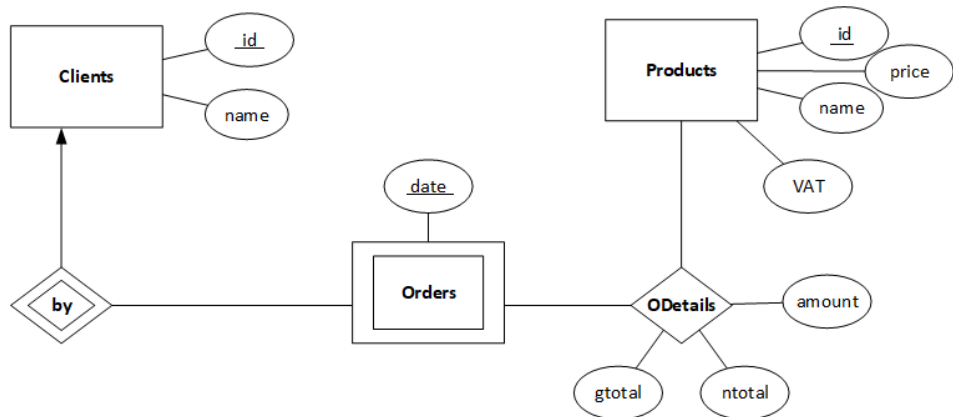
Защо премахнахме атрибутите: nTotal, gTotal ?

Отговор: Защото в тях се пазят изчислими стойности:

- ▶ $nTotal = pPrice * Amount$
- ▶ $gTotal = nTotal * VAT$

Решение – Подход 1

Подход 1: Правим E/R модел по описаната информация и преобразуваме до релационни схеми. Намираме всички ФЗ за релациите и нормализираме, ако е необходимо.



Всички ФЗ:

1. $cID \rightarrow cName$;
2. $pID \rightarrow pName, pPrice, VAT$;
3. $cID, pID, oDate \rightarrow amount$;

Преобразуваме. В коя НФ е БД?

- ▶ **Clients**(cID, cName); [ФЗ-1, НФБК]
- ▶ **Products**(pID, pPrice, pName, VAT); [ФЗ-2, НФБК]
- ▶ **Orders**(cID, pID, oDate, amount) [ФЗ-3, НФБК]

Решение – Подход 2

Подход 2: Поставяме всички атрибути (характеристики) на данните в една таблица. Намираме всички ФЗ за тази релация. Нормализираме.

1. Поставяме всичко в една релация

Orders(pId, pName, pPrice, Amount, cId, cName, oDate, VAT, nTotal, gTotal)

2. Намираме всички ФЗ за релацията Order (от дадената ни информация)

- ▶ pId -> pName, pPrice, VAT (ФЗ-1)
- ▶ cId -> cName (ФЗ-2)
- ▶ pId, cId, oDate -> Amount (ФЗ-3)
- ▶ pPrice, Amount -> nTotal (ФЗ-4)
- ▶ nTotal, VAT -> gTotal (ФЗ-5)

3. Премахваме атрибутите, в които има изчислими стойности от релацията

Решение – Подход 2 (Нормализация)

Orders(pId, pName, pPrice, Amount, cId, cName, oDate, VAT)

- ▶ `pId -> pName, pPrice, VAT (ФЗ-1)`
- ▶ `cId -> cName (ФЗ-2)`
- ▶ **`pId, cId, oDate`** -> `Amount (ФЗ-3)`

4. Намираме ключа за Order: **{pId, cId, oDate}**

5. Декомпозираме до НФБК:

- ▶ Намираме ФЗ, която нарушава НФБК (ФЗ-1, ФЗ-2)
- ▶ Декомпозираме до две релации **Orders1** и **Orders2** (по алгоритъма за привеждане до НФБК)

Решение – Подход 2 (Нормализация)

- ▶ Получаваме от Orders(pId, pName, pPrice, Amount, cId, cName, oDate, VAT) и Ф3-1:
 - ▶ Orders1 (pId, pName, pPrice, VAT) = Products(pId, pName, pPrice, VAT)
 - ▶ Orders2(cId, cName, pId, oDate, Amount)
- ▶ За Orders2 повтаряме стъпка 4 и 5 , докато стигнем до нормализиран модел.
 - ▶ Products(pId, pName, pPrice, VAT)
- ▶ Orders2(cId, cName, pId, oDate, Amount) и Ф3-2 =>
 - ▶ Orders3 (cId, cName) = Customers(cId, cName)
 - ▶ Orders4 (pId, cId, oDate, Amount) = Orders (pId, cId, oDate, Amount)

Решение – Подход 2 (Нормализация)

- ▶ Така от първоначалната релация и ФЗ:

Orders(pId, pName, pPrice, Amount, cId, cName, oDate, VAT)

- ▶ pId → pName, pPrice, VAT (ФЗ-1)
- ▶ cId → cName (ФЗ-2)
- ▶ **pId, cId, oDate** → Amount (ФЗ-3)

- ▶ Получаваме:

- ▶ Products(pId, pName, pPrice, VAT) – НФБК (в сила е ФЗ-1)
- ▶ Customers(cId, cName) – НФБК (в сила е ФЗ-2)
- ▶ Orders(pId, cId, oDate, Amount) – НФБК (в сила е ФЗ-3)

Задача

DreamHome Lease

DreamHome Lease

DreamHome Lease

DreamHome Lease

<p>Client Number <u>CR76</u> (Enter if known)</p> <p>Full Name <u>John Kay</u> (Please print)</p>	<p>Property Number <u>PG4</u></p> <p>Property Address <u>6 Lawrence St, Glasgow</u></p>
<p>Monthly Rent <u>350</u></p> <p>Rent Start <u>01/07/12</u></p> <p>Rent Finish <u>31/08/13</u></p>	<p>Owner Number <u>C040</u> (Enter if known)</p> <p>Full Name <u>Tina Murphy</u> (Please print)</p>

Книга: Database systems.A practical approach to design, implementation and management,
Автори: Thomas Connolly, Carolyn Begg

Property

DreamHome Property Registration Form	
Property Number <u>PG16</u> Type <u>Flat</u> Rooms <u>4</u> Rent <u>450</u> Address <u>5 Novar Drive,</u> <u>Glasgow, G12 9AX</u> <u> </u> <u> </u>	Owner Number <u>C093</u> (If known) Person/Business Name <u>Tony Shaw</u> Address <u>12 Park Pl,</u> <u>Glasgow G4 0QR</u> Tel No <u>0141-225-7025</u>
	Enter details where applicable Type of business _____ Contact Name _____
Managed by staff <u>David Ford</u>	Registered at branch <u>163 Main St, Glasgow</u>

Client

<i>DreamHome</i> Client Registration Form	
<p>Client Number <i>CR74</i> (Enter if known) _____</p> <p>Full Name _____ <i>Mike Ritchie</i></p>	<p>Branch Number <i>B003</i> _____</p> <p>Branch Address _____ <i>163 Main St, Glasgow</i> _____</p>
<p>Enter property requirements</p> <p>Type <i>Flat</i> _____</p> <p>Max Rent <i>750</i> _____</p>	<p>Registered By _____ <i>Ann Beech</i></p> <p>Date Registered <i>16-Nov-11</i> _____</p>

Property-Client

DreamHome Property Viewing Report

Property Number PG4

Property Address

6 Lawrence St, Glasgow

Type Flat

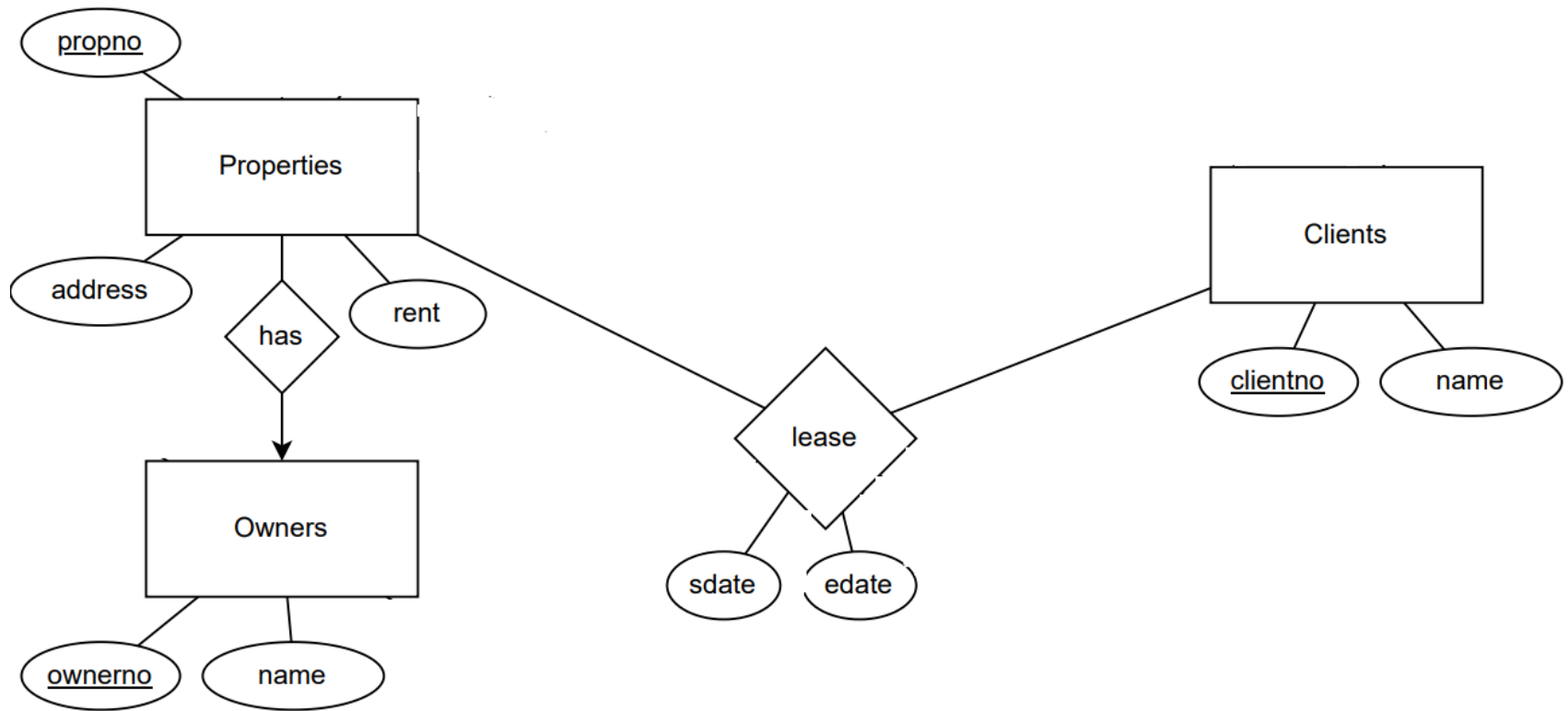
Rent 350

Client No	Name	Date	Comments
CR76	John Kay	20/04/13	Too remote. OK, but needs redecoration throughout.
CR56	Aline Stewart	26/05/13	
CR74	Mike Ritchie	11/11/13	
CR62	Mary Tregear	11/11/13	

Lease

<i>DreamHome</i> Lease Number 00345810	
Client Number <u>CR74</u> (Enter if known) Full Name <u>Mike Ritchie</u> (Please print) Client Signature _____	Property Number <u>PG16</u> Property Address <u>5 Novar Dr, Glasgow</u>
Enter payment details Monthly Rent <u>450</u> Payment Method <u>Cheque</u> Deposit Paid (Y or N) <u>Yes</u>	Rent Start <u>01/06/12</u> Rent Finish <u>31/05/13</u> Duration <u>1 year</u>

Подход 1 – E/R модел



Подход 1 – Релационен модел

- ▶ Properties (propno, paddress, rent, ownerno)
- ▶ Clients (clientno, name)
- ▶ Owners (ownerno, name)
- ▶ Lease (propno, clientno, sdate, edate)

Подход 2 - Не нормализиран модел

ClientRental

clientNo	cName	propertyNo	pAddress	rentStart	rentFinish	rent	ownerNo	oName
CR76	John Kay	PG4	6 Lawrence St, Glasgow	1-Jul-12	31-Aug-13	350	CO40	Tina Murphy
		PG16	5 Novar Dr, Glasgow	1-Sep-13	1-Sep-14	50	CO93	Tony Shaw
CR56	Aline Stewart	PG4	6 Lawrence St, Glasgow	1-Sep-11	10-June-12	350	CO40	Tina Murphy
		PG36	2 Manor Rd, Glasgow	10-Oct-12	1-Dec-13	375	CO93	Tony Shaw
		PG16	5 Novar Dr, Glasgow	1-Nov-14	10-Aug-15	450	CO93	Tony Shaw

Figure 14.10 ClientRental unnormalized table.

1НФ

Първа нормална форма (1НФ) – изисква всеки компонент в кортежите да е атомарна стойност.

ClientRental

clientNo	propertyNo	cName	pAddress	rentStart	rentFinish	rent	ownerNo	oName
CR76	PG4	John Kay	6 Lawrence St, Glasgow	1-Jul-12	31-Aug-13	350	CO40	Tina Murphy
CR76	PG16	John Kay	5 Novar Dr, Glasgow	1-Sep-13	1-Sep-14	450	CO93	Tony Shaw
CR56	PG4	Aline Stewart	6 Lawrence St, Glasgow	1-Sep-11	10-Jun-12	350	CO40	Tina Murphy
CR56	PG36	Aline Stewart	2 Manor Rd, Glasgow	10-Oct-12	1-Dec-13	375	CO93	Tony Shaw
CR56	PG16	Aline Stewart	5 Novar Dr, Glasgow	1-Nov-14	10-Aug-15	450	CO93	Tony Shaw

Figure 14.11 First Normal Form ClientRental relation.

ФЗ за релацията ClientRental

ClientRental

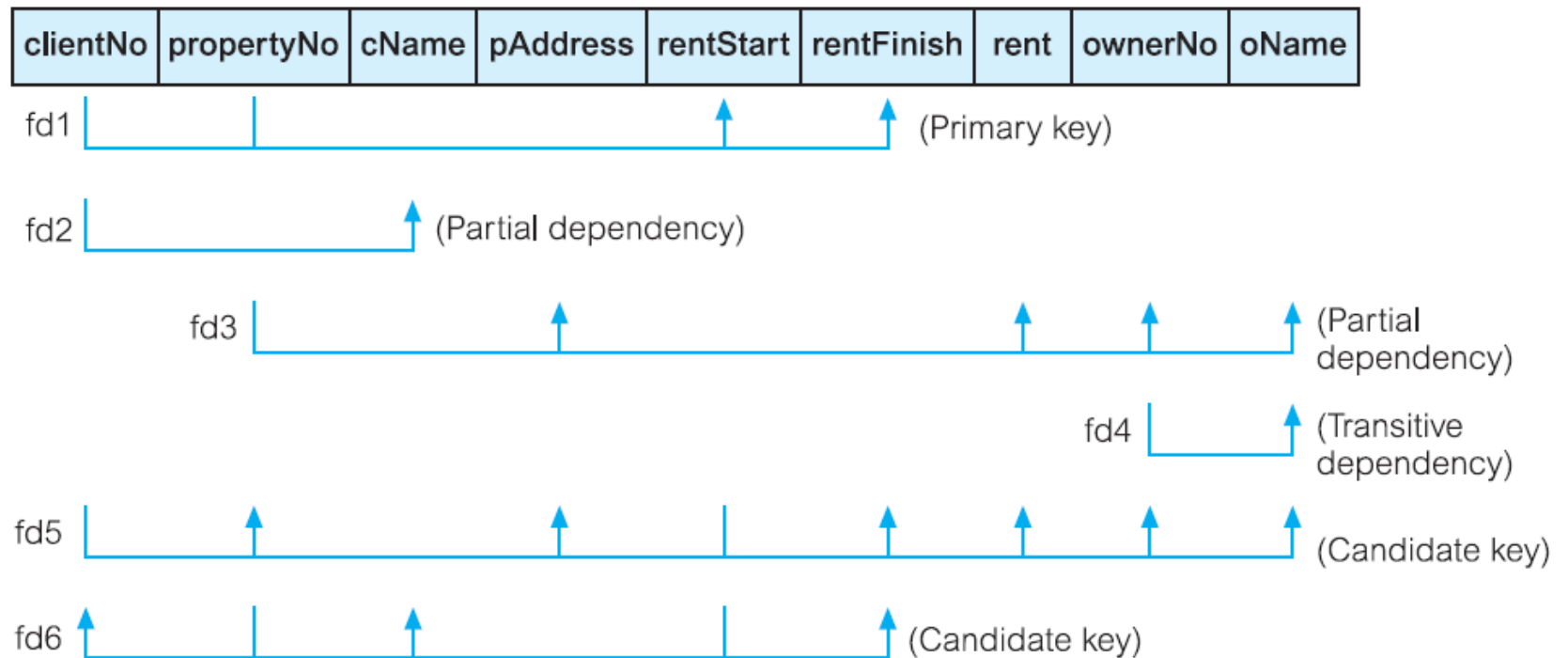


Figure 14.12 Functional dependencies of the ClientRental relation.

Ф3 за релацията ClientRental

ClientRental(clientNo, propertyNo, cName, pAddress, rentStart, rentFinish, rent, ownerNo, oName)

fd1 clientNo, propertyNo → rentStart, rentFinish

fd2 clientNo → cName

fd3 propertyNo → pAddress, rent, ownerNo, oName

fd4 ownerNo → oName

fd5 clientNo, rentStart → propertyNo, pAddress, rentFinish, rent, ownerNo, oName

fd6 propertyNo, rentStart → clientNo, cName, rentFinish

Ключ за релацията ClientRental

- ▶ Има три възможности за ключ на релацията:
 - ▶ 1. (clientNo, propertyNo) - PK
 - ▶ 2. (clientNo, rentStart) – кандидат ключ
 - ▶ 3. (propertyNo, rentStart) – кандидат ключ
- ▶ Доказателство:
 - ▶ Намираме покритието за трите двойки и се уверяваме, че това е така

Нормализираме до НФБК

`ClientRental(clientNo, propertyNo, cName, pAddress, rentStart, rentFinish, rent, ownerNo, oName)`

fd1 `clientNo, propertyNo → rentStart, rentFinish`

fd2 `clientNo → cName`

fd3 `propertyNo → pAddress, rent, ownerNo, oName`

fd4 `ownerNo → oName`

fd5 `clientNo, rentStart → propertyNo, pAddress, rentFinish, rent, ownerNo, oName`

fd6 `propertyNo, rentStart → clientNo, cName, rentFinish`

Функционалните зависимости fd2, fd3 и fd4 нарушават правилото на НФБК. Декомпозираме по алгоритъма ClientRental и получаваме:

- **`Clients(clientNo, cName)` – НФБК (в сила е само fd2)**
- `ClientRental2(clientNo, propertyNo, pAddress, rent, ownerNo, oName, rentStart, rentFinish)`

Нормализираме до НФБК

fd1 clientNo, propertyNo → rentStart, rentFinish
fd2 clientNo → cName
fd3 propertyNo → pAddress, rent, ownerNo, oName
fd4 ownerNo → oName
fd5 clientNo, rentStart → propertyNo, pAddress, rentFinish, rent, ownerNo, oName
fd6 propertyNo, rentStart → clientNo, cName, rentFinish

ClientRental2(clientNo, propertyNo, pAddress, rent, ownerNo, oName, rentStart, rentFinish)

- Ключовете за ClientRental2 са същите
- В сила са всички ФЗ без fd2

Функционалната зависимост fd3 нарушава правилото на НФБК.

Декомпозираме по алгоритъма ClientRental2 и получаваме:

- Properties(propertyNo, pAddress, rent, ownerNo, oName)
- ClientRental3(clientNo, propertyNo, rentStart, rentFinish) – НФБК,
- = Rental (clientNo, propertyNo, rentStart, rentFinish)

Нормализираме до НФБК

fd1 clientNo, propertyNo → rentStart, rentFinish

fd2 clientNo → cName

fd3 propertyNo → pAddress, rent, ownerNo, oName

fd4 ownerNo → oName

fd5 clientNo, rentStart → propertyNo, pAddress, rentFinish, rent, ownerNo, oName

fd6 propertyNo, rentStart → clientNo, cName, rentFinish

- Properties(propertyNo, pAddress, rent, ownerNo, oName)
- Rental(clientNo, propertyNo, rentStart, rentFinish) – НФБК (в сила са fd1, fd5 и fd6 – но без липсващите атрибути)

Функционалната зависимост fd3 и fd4 са в сила за properties, но fd4 нарушава правилото на НФБК. Декомпозираме по алгоритъма Properties и получаваме:

- Properties(propertyNo, pAddress, rent, ownerNo) – НФБК (fd3)
- Owners(ownerNo, oName) – НФБК (в сила е fd4)

ClientRental - НФБК

Client

clientNo	cName
CR76	John Kay
CR56	Aline Stewart

Rental

clientNo	propertyNo	rentStart	rentFinish
CR76	PG4	1-Jul-12	31-Aug-13
CR76	PG16	1-Sep-13	1-Sep-14
CR56	PG4	1-Sep-11	10-Jun-12
CR56	PG36	10-Oct-12	1-Dec-13
CR56	PG16	1-Nov-14	10-Aug-15

PropertyForRent

propertyNo	pAddress	rent	ownerNo
PG4	6 Lawrence St, Glasgow	350	CO40
PG16	5 Novar Dr, Glasgow	450	CO93
PG36	2 Manor Rd, Glasgow	375	CO93

Owner

ownerNo	oName
CO40	Tina Murphy
CO93	Tony Shaw

Задача

JobID	JobDate Time	driverID	driver Name	taxiID	clientID	clientName	jobPickUpAddress
1	25/07/14 10.00	D1	Joe Bull	T1	C1	Anne Woo	1 Storrie Rd, Paisley
2	29/07/14 10.00	D1	Joe Bull	T1	C1	Anne Woo	1 Storrie Rd, Paisley
3	30/07/14 11.00	D2	Tom Win	T2	C1	Anne Woo	3 High Street, Paisley
4	2/08/14 13.00	D3	Jim Jones	T3	C2	Mark Tin	1A Lady Lane, Paisley
5	2/08/14 13.00	D4	Steven Win	T1	C3	John Seal	22 Red Road, Paisley
6	25/08/14 10.00	D2	Tom Win	T2	C4	Karen Bow	17 High Street, Paisley

- ▶ Дадени са следните ФЗ, които са в сила за релацията:
 - ▶ ФЗ в сила за релацията са:
 1. jobID->jobDate Time, jobPickUpAddress
 2. driverID -> driverName, taxiID
 3. clientID -> clientName
- ▶ Нормализирайте до НФБК
- ▶ Решение:
 - ▶ Ключът на релацията е {jobID, driverID, clientID}
 - ▶ Всички ФЗ, нарушават правилото на НФБК => декомпозираме

Нормализация

TaxiJob(jobID, jobDateTime, jobPickUpAddress, driverID, driverName, taxiID, clientID, clientName)

- ▶ ФЗ-1, нарушава правилото за НФБК за TaxiJob, декомпозираме:

1. **Jobs(jobID, jobDateTime, jobPickUpAddress) - НФБК**
2. TaxiJob1(jobID, driverID, driverName, taxiID, clientID, clientName)

- ▶ ФЗ-2, нарушава правилото на НФБК за TaxiJob1, декомпозираме:

1. **Drivers(driverID, driverName, taxiID) - НФБК**
2. TaxiJob2(jobID, driverID, clientID, clientName)

- ▶ ФЗ-3, нарушава правилото на НФБК за TaxiJob2, декомпозираме:

1. **Clients(clientID, clientName) - НФБК**
2. **TaxiJob3(jobID, driverID, clientID) - НФБК**

Как да нормализираме релации?

- ▶ **Задача:** Дадена е релацията R и описание към нея. Нормализирайте до НФБК.
- ▶ **Алгоритъм:**
 1. Намирате всички ФЗ, които са в сила за релацията. Извличате ги или от условието на задачата или от факти от действителността. Например ЕГН-то еднозначно определя всеки живущ в България.
 2. Определяте всички кандидат-ключове за релацията R
 3. Търсите ФЗ, която нарушава правилото на БК (ФЗ, която в лявата си част не съдържа ключ). Например FDI
 4. Декомпозирайте R до две релации R_1 и R_2 , така че всички атрибути от FDI са в R_1 , а лявата част на FDI и всички останали атрибути са в R_2 .
 5. Повтаряме стъпка 2) до 4) за релацията R_2

Задача 1

Student_Id	Student_Address	Lecture	Teaching_Assisant
1234	Rämistrasse 72	Data Modelling and Databases	Bob
1280	Rennweg 19	Concepts of Concurrent Computation	Scott
1234	Rämistrasse 72	Visual Computing	Sarah
1299	Börsenstrasse 42	Concepts of Concurrent Computation	Benjamin
1356	Klusplatz 45	Concepts of Concurrent Computation	Benjamin

- ▶ Известно е че:
 - ▶ Един курс (Lecture) се води само от един Teaching_Assistant
 - ▶ В сила са следните ФЗ:
 - ▶ FD-1: student_id -> student_address
 - ▶ FD-2: lecture -> teaching_assistant
- ▶ Нормализирайте до НФБК
- ▶ Отговор:
 - ▶ Students(Student_Id, Student_Address)
 - ▶ Lectures(Lecture, Teaching_Assistant)
 - ▶ Student_Lectures(Student_Id, Lecture)

Задача 2

leaseNo	bannerID	placeNo	fName	lName	startDate	finishDate	flatNo	flatAddress
10003	B017706	78	Jane	Watt	01/09/2010	30/06/2011	F56	34 High Street, Paisley
10259	B017706	88	Jane	Watt	01/09/2011	30/06/2012	F78	111 Storrie Road, Paisley
10364	B013399	89	Tom	Jones	01/09/2011	30/06/2012	F78	111 Storrie Road, Paisley
10566	B012124	102	Karen	Black	01/09/2011	30/06/2012	F79	120 Lady Lane, Paisley
11067	B034511	88	Steven	Smith	01/09/2012	30/06/2013	F78	111 Storrie Road, Paisley
11169	B013399	78	Tom	Jones	01/09/2012	30/06/2013	F56	34 High Street, Paisley

- ▶ Известно е че в релацията се съхранява информация за студенти (fName, lName) и в коя квартира са настанени в общежитието (flatNo, flatAddress) и на кое място (placeNo). Всеки студент еднозначно се определя от (bannerID).
- ▶ В сила са следните ФЗ:
 - ▶ FD-1: bannerID -> fName, lName
 - ▶ FD-2: flatNo -> flatAddress
 - ▶ FD-3: placeNo -> flatNo
 - ▶ FD-4: leaseNo, bannerID, placeNo -> startDate, finishDate
- ▶ Нормализирайте до НФБК

Задача 2 - Решение

- ▶ StudFlat(leaseNo, bannerID, placeNo, fName, lName, startDate, finishDate, flatNo, flatAddress)

- ▶ FD-1: bannerID → fName, lName
- ▶ FD-2: flatNo → flatAddress
- ▶ FD-3: placeNo → flatNo
- ▶ FD-4: leaseNo, bannerID, placeNo → startDate, finishDate

- ▶ {leaseNo, bannerID, placeNo}+
- ▶ 1) X={leaseNo, bannerID, placeNo}
- ▶ 2) X={leaseNo, bannerID, placeNo, startDate, finishDate, fName, lName, flatNo, flatAddress}

- ▶ PK:{leaseNo, bannerID, placeNo}

- ▶ BCNF:
 - ▶ Students(bannerID, fName, lName)
 - ▶ Flats(flatNo, flatAddress)
 - ▶ Places(placeNo, flatNo)
 - ▶ Leases(leaseNo, bannerID, placeNo, startDate, finishDate)

Задача 3

- Дадена е релацията. Нормализирайте до НФБК

staffNo	dentistName	patientNo	patientName	appointment date time	surgeryNo
S1011	Tony Smith	P100	Gillian White	12-Aug-03 10.00	S10
S1011	Tony Smith	P105	Jill Bell	13-Aug-03 12.00	S15
S1024	Helen Pearson	P108	Ian MacKay	12-Sept-03 10.00	S10
S1024	Helen Pearson	P108	Ian MacKay	14-Sept-03 10.00	S10
S1032	Robin Plevin	P105	Jill Bell	14-Oct-03 16.30	S15
S1032	Robin Plevin	P110	John Walker	15-Oct-03 18.00	S13

- В сила са следните ФЗ:
 - FD1: staffNo, apptDate, appTime -> patientNo, patientName
 - FD2: staffNo -> dentistName
 - FD3: patientNo -> patientName
 - FD4: staffNo, appDate -> surgeryNo
 - FD5: appDate, appTime, patientNo -> staffNo, dentistName

Задача 3

▶ Ф3:

- ▶ FD1: staffNo, apptDate, appTime -> patientNo, patientName
- ▶ FD2: staffNo -> dentistName
- ▶ FD3: patientNo -> patientName
- ▶ FD4: staffNo, appDate -> surgeryNo
- ▶ FD5: appDate, appTime, patientNo -> staffNo, dentistName

▶ Ключ:

- ▶ {staffNo, apptDate, appTime}
- ▶ {appDate, appTime, patientNo}

▶ ИФБК

- ▶ Dentists(staffNo, dentistName)

Задача 3

▶ Ф3:

- ▶ FD1: staffNo, apptDate, appTime -> patientNo, patientName
- ▶ FD2: staffNo -> dentistName
- ▶ FD3: patientNo -> patientName
- ▶ FD4: staffNo, appDate -> surgeryNo
- ▶ FD5: appDate, appTime, patientNo -> staffNo, dentistName

▶ НФБК

- ▶ Dentists(staffNo, dentistName)
- ▶ Patient (patientNo, patientName)
- ▶ Surgery (staffNo, appDate, surgeryNo)
- ▶ Appointment (patientNo, appDate, appTime, staffNo)

Задача 4

- ▶ Дадена е релацията. В сила са следните ФЗ.
Нормализирайте до НФБК

NIN	contractNo	hoursPerWeek	eName	hotelNo	hotelLocation
113567WD	C1024	16	John Smith	H25	Edinburgh
234111XA	C1024	24	Diane Hocine	H25	Edinburgh
712670YD	C1025	28	Sarah White	H4	Glasgow
113567WD	C1025	16	John Smith	H4	Glasgow

- ▶ FD1: NIN, contractNo -> hours
- ▶ FD2: NIN -> eName
- ▶ FD3: contractNo -> hotelNo, hotelLocation
- ▶ FD4: hotelNo -> hotelLocation

Задача 4 - НФБК

▶ Ф3:

- ▶ FD1: NIN, contractNo -> hours
- ▶ FD2: NIN -> eName
- ▶ FD3: contractNo -> hotelNo, hotelLocation
- ▶ FD4: hotelNo -> hotelLocation

▶ Ключ = {NIN, contractNo}

▶ НФБК

- ▶ Staff(NIN, eName)
- ▶ Hotel(hotelNo, hotelLocation)
- ▶ Contract(contractNo, hotelNo)
- ▶ WorkHours(NIN, contractNo, hours)

Задача 5

- ▶ Дадена е релацията. В сила са следните ФЗ.
Нормализирайте до НФБК

<u>StdSSN</u>	<u>StdCity</u>	<u>StdClass</u>	<u>OfferNo</u>	<u>OffTerm</u>	<u>OffYear</u>	<u>EnrGrade</u>	<u>CourseNo</u>	<u>CrsDesc</u>
S1	SEATTLE	JUN	O1	FALL	2006	3.5	C1	DB
S1	SEATTLE	JUN	O2	FALL	2006	3.3	C2	VB
S2	BOTHELL	JUN	O3	SPRING	2007	3.1	C3	OO
S2	BOTHELL	JUN	O2	FALL	2006	3.4	C2	VB

- ▶ ФЗ:
 - ▶ FD-1: StdSSN -> StdCity, StdClass
 - ▶ FD-2: OfferNo -> OffTerm, OffYear, CourseNo, CrsDesc
 - ▶ FD-3: CourseNo -> CrsDesc
 - ▶ FD-4: StdSSN, OfferNo -> EnrGrade

Задача 5 - Решение

- ▶ StdClass(StdSSN, OfferNo, StdCity, StdClass, OffTerm, OffYear, CourseNo, CrsDesc, EnrGrade)
 - ▶ FD-1: StdSSN -> StdCity, StdClass
 - ▶ FD-2: OfferNo -> OffTerm, OffYear, CourseNo, CrsDesc
 - ▶ FD-3: CourseNo -> CrsDesc
 - ▶ FD-4: StdSSN, OfferNo -> EnrGrade
- ▶ {StdSSN, OfferNo}⁺ =
- ▶ {StdSSN, OfferNo, StdCity, StdClass, OffTerm, OffYear, CourseNo, CrsDesc, EnrGrade}
- ▶ PK: {StdSSN, OfferNo}

Задача 5 - Решение

StdClass(StdSSN, OfferNo, StdCity, StdClass, OffTerm, OffYear, CourseNo, CrsDesc, EnrGrade)

- ▶ FD-1: StdSSN → StdCity, StdClass
- ▶ FD-2: OfferNo → OffTerm, OffYear, CourseNo, CrsDesc
- ▶ FD-3: CourseNo → CrsDesc
- ▶ FD-4: StdSSN, OfferNo → EnrGrade

- ▶ PK: {StdSSN, OfferNo}

- ▶ BCNF:
 - ▶ Students(StdSSN, StdCity, StdClass) – FD 1
 - ▶ Courses(CourseNo, CrsDesc) – FD3
 - ▶ Offers(OfferNo, OffTerm, OffYear, CourseNo) – FD2
 - ▶ Grades(StdSSN, OfferNo, EnrGrade) – FD4

Задача 6

- Дадена е релацията StarWars

<u>CharId</u>	<u>CharName</u>	<u>ActorName</u>	<u>FigId</u>	<u>OwnerId</u>	Pseudonym
Vader	Darth Vader	David Prowse	f14	Bill	Jabba the Hoot
Vader	Darth Vader	David Prowse	f22	Amy	Don't Blame Me
Yoda	Yoda	Frank Oz	f16	Lucy	<u>Xena</u> Warrior
Leia	Princess Leia	Carrie Fisher	f45	Bill	Jabba the Hoot
Leia	Princess Leia	Carrie Fisher	f99	Amy	Don't Blame Me

StarWars (charid, charname, actorname, figid, ownerid, pseudonym)

Задача

StarWars (charid, charname, actorname, figid, ownerid, pseudonym)

- ▶ Определете ФЗ за релацията. Намерете ключа за релацията.
- ▶ Решение:
- ▶ В сила са следните ФЗ за релацията StarWars :
 - ▶ figid -> ownerid, pseudonym
 - ▶ charid -> charname, actorname
- ▶ Ключът за релацията е {figid, charid}
- ▶ Нормализация:
 - ▶ Figures (figid, ownerid, pseudonym)
 - ▶ Chars (charid, charname, actorname)
 - ▶ FigChar (figid, charid)