

# РЕЛАЦИОННИ БАЗИ ДАННИ

[dimitrova@tu-sofia.bg](mailto:dimitrova@tu-sofia.bg)

[pct.tu-sofia.bg/dd/pik3](http://pct.tu-sofia.bg/dd/pik3)



# Релационни бази от данни

- Релационен модел на данните
- Основен обект: таблица
- Математически апарат:
  - релационна алгебра
  - теория на множествата



Релация -> таблица

$$R_1 = \{ (A, 1), (A, 3), (B, 1), (B, 2) \}$$

$d_1$	$d_2$
A	1
A	3
B	1
B	2

атрибут

кортеж



## Релационна алгебра

Релация:

крайно под-множество  $R$  на Декартовото произведение на множество домейни  $M = \{ D_1 \times D_2 \times D_3 \times \dots, \times D_n \}$

Домейн:

множество от възможни стойности  $D_i = \{ V_1, V_2, V_3, \dots, V_k \}$

Пример:

$$D_1 = \{ A, B \} \quad D_2 = \{ 1, 2, 3 \}$$

$$M = D_1 \times D_2 \Rightarrow \{ (A, 1), (A, 2), (A, 3), (B, 1), (B, 2), (B, 3) \}$$

$$R_1 = \{ (A, 1), (A, 3), (B, 1), (B, 2) \}$$

$$R_2 = \{ (A, 1), (A, 2), (B, 3), (B, 2), (B, 3) \}$$



## Таблицы

- една таблица – една група обекти, клас
- един ред – един екземпляр от класа
- една колона – един атрибут на този клас обекти
- една клетка – една стойност на атрибута за екземпляра
- всеки екземпляр е уникален, т.е. няма повтарящи се редове
- не подредена (подредбата на редовете няма значение)



Таблица

## Personal

Атрибути

**EGN**

**ime**

**address**

Екземпляри

4

## Personal (EGN, ime, address)

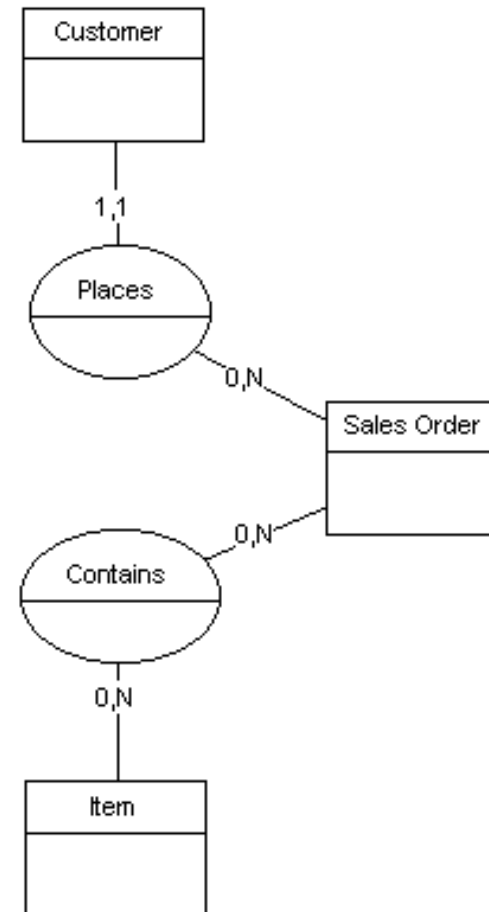
<b>EGN</b>	<b>ime</b>	<b>address</b>
1234	Боян	Слатина
1244	Антон	Младост
1108	Светла	Люлин
8543	Мила	Дианабад



# Entity Relationship Model

класове обекти  
отношения между тях

## ER



- Осигуряване на съответствие между данните в таблиците
- Постига се посредством
  - ключове
  - отношения между таблиците
  - нормализация





# Ключ

- Идентификатор на екземпляра
- Има уникална стойност за всеки екземпляр
- Осигурява уникалност на реда в таблицата
- Нарича се **първичен ключ** (*Primary Key, PK*)



Множество от атрибути  $S$  на класа, за които са изпълнени условията:

1. Няма два екземпляра, за които е (теоретично) възможно да имат еднакви стойности на тези атрибути.
2. Няма под-множество на  $S$ , за което да е валидно условието 1.



## Personal (EGN, ime, address)

ime = “Боян”

address = “Младост”

EGN – не може да се повтаря

EGN – първичен ключ

EGN	ime	address
1234	Боян	Слатина
6543	Еми	Младост
1244	Антон	Младост
1108	Светла	Люлин
7856	Боян	Обеля
8543	Мила	Дианабад



Когато няма един подходящ атрибут, се търси множество от повече атрибути, които заедно идентифицират екземплярите еднозначно

EGN	date	workhours
1236	16/02/04	8
1236	17/02/04	4
1238	16/02/04	8

**Work(EGN,date, workhours)**



Когато няма един подходящ атрибут, алтернативно: добавя се атрибут – идентификатор на екземпляра

фамилия	име	адрес	телефон
Крайников	Йосиф А.	Х. Димитър 66	454529
Крайниковски	Асен И.	Ивац Войвода 20	449949
Крайнин	Христо К.	Червена роза 31	668819
Крайничанец	Драган Д.	В. Левски 52	539282



## ID - ключ

ID	фамилия	име	адрес	телефон
1	Крайников	Йосиф А.	Х. Димитър 66	454529
2	Крайниковски	Асен И.	Ивац Войвода 20	449949
3	Крайнин	Христо К.	Червена роза 31	668819
4	Крайничанец	Драган Д.	В. Левски 52	539282



# Отношения

- установяване на зависимостите между класовете (таблиците)
- едно отношение (**Relationship**) е отношение между два класа
- кардиналност на отношението
  - свойство, което показва колко екземпляра от единия клас имат пряка връзка с екземпляри от другия клас

**1:1      1:N      N:1      N:M**



# Отношения

● Представят се чрез **таблици, атрибути** или и **двете**

- атрибут **външен ключ** (**1:N, N:1, 1:1**)
- **таблица**, чиито атрибути са ключове на асоциирани таблици (**N:M**)
- **сливащи се таблици** (**1:1**)

● **Външен ключ**: Множество от атрибути на релация, които се свързват с ключа на друга релация

● **Referential Integrity**: Ограничение на външния ключ: всяка стойност на атрибута **външен ключ** се свързва с обект от външната таблица





1:1

EGN	specialnost
1234	архитект
1244	архитект
1108	техник
8543	инженер

EGN	ime	address
1234	Боян	Слатина
1244	Антон	Младост
1108	Светла	Люлин
8543	Мила	Дианабад



1:N

cod	specialnost
1	архитект
2	инженер
3	техник

EGN	ime	address	spec
1234	Боян	Слатина	1
1244	Антон	Младост	1
1108	Светла	Люлин	3
8543	Мила	Дианабад	2





Ако  $d_1$  и  $d_2$  са атрибути, то

$$d_1 \rightarrow d_2$$

атрибут  $d_2$  е функционално зависим от атрибут  $d_1$ , ако стойността на  $d_1$  еднозначно определя стойността  $d_2$ .

Примери:

Personal(EGN, ime, address, pcode)

Personal(EGN, ime, grad, ulica, pcode)



# Нормализация

- Процес на осигуряване на оптимална структура на базата от данни
- Основава се на взаимозависимостта на данните
- Трансформиране на структурата на таблиците, с цел оптимизиране на функционалните зависимости на атрибутите им
- Предимства
  - икономия на памет, посредством премахване на излишеството - отстраняване на повтарящи се данни
  - бързодействие, посредством ускоряване на търсенето и намирането на данни



## Нормални форми

1NF: всеки атрибут има по една стойност за екземпляр (атомарни данни)

2NF: всеки атрибут е функционално зависим от целия първичен ключ, но не от негови отделни елементи

3NF: един атрибут не може да бъде функционално зависим от друг (неключов) атрибут

4NF: относно агрегиране на функционални зависимости

5NF: относно зависимости между свързани обекти



1NF:

EGN	name	phone
1236	Симеон	<del>9651833, 08881122</del>

EGN	date	workhours	name	salary
1236	16/02/04	8	Симеон	160



2NF: Work(EGN,date, workhours, name, salary)

3NF: Work(EGN,date, workhours, name, salary)



## Обобщение: елементи на БД

Релация	-	Таблица	-	Файл
Кортеж	-	Ред	-	Запис
Атрибут	-	Колона	-	Поле

Първичен ключ

Външни ключове

Отношения

Изгледи

Процедури





## ● Система за управление на база от данни

## ● Функции

## ● Инструменти

### ■ Речник на данните

- Пълен списък на всички атрибути на всички таблици

- За всеки атрибут:

име

тип

домейн

правила

### ■ Файл на транзакциите

- Регистрация на всички операции, извършени с данните от таблици

