

# Базы данных

---

Лектор: Булова Юлия Олеговна

# Лекция 1. Введение в базы данных.

---

Вопросы:

- Понятие «база данных»;
- Основные требования к базе данных;
- Основные принципы построения (проектирования) базы данных;
- Основные модели данных;
- Основы теории реляционных баз данных.

**База данных** - хранилище систематизированных, динамически обновляемых данных.

**Компьютерные базы данных** - базы данных, использующие электронные носители для хранения данных и специальные программные средства для доступа к данным (СУБД).

**Данные:** текстовые, числовые, графические, мультимедийные (видео, звук), бинарные (исполняемый код) и пр.

**Требования к информации в базе данных:**

1. полезность;
2. полнота информации;
3. точность;
4. достоверность;
5. непротиворечивость;
6. актуальность.

**Мера информации в базе данных:**

1. синтаксическая (в символах, в В, КВ, МВ, GB, ТВ);
2. семантическая мера информации (количество информации на символ);
3. прагматическая мера информации (полезность для управления).

## Построение (проектирование базы данных):

1. определение границ исследуемой области — **предметной области** (часть реального мира, подлежащая изучению, с целью описания и управления);
2. **системный анализ**: определение объектов и связей между ними;
3. построение **логической схемы** базы данных в соответствии с определенными правилами - моделью данных (структурированное представление данных и связей между ними);
4. реализация базы данных (описание ее в терминах некоторой СУБД).

**База данных** представляет собой модель данных, описывающих предметную область и сами данные.

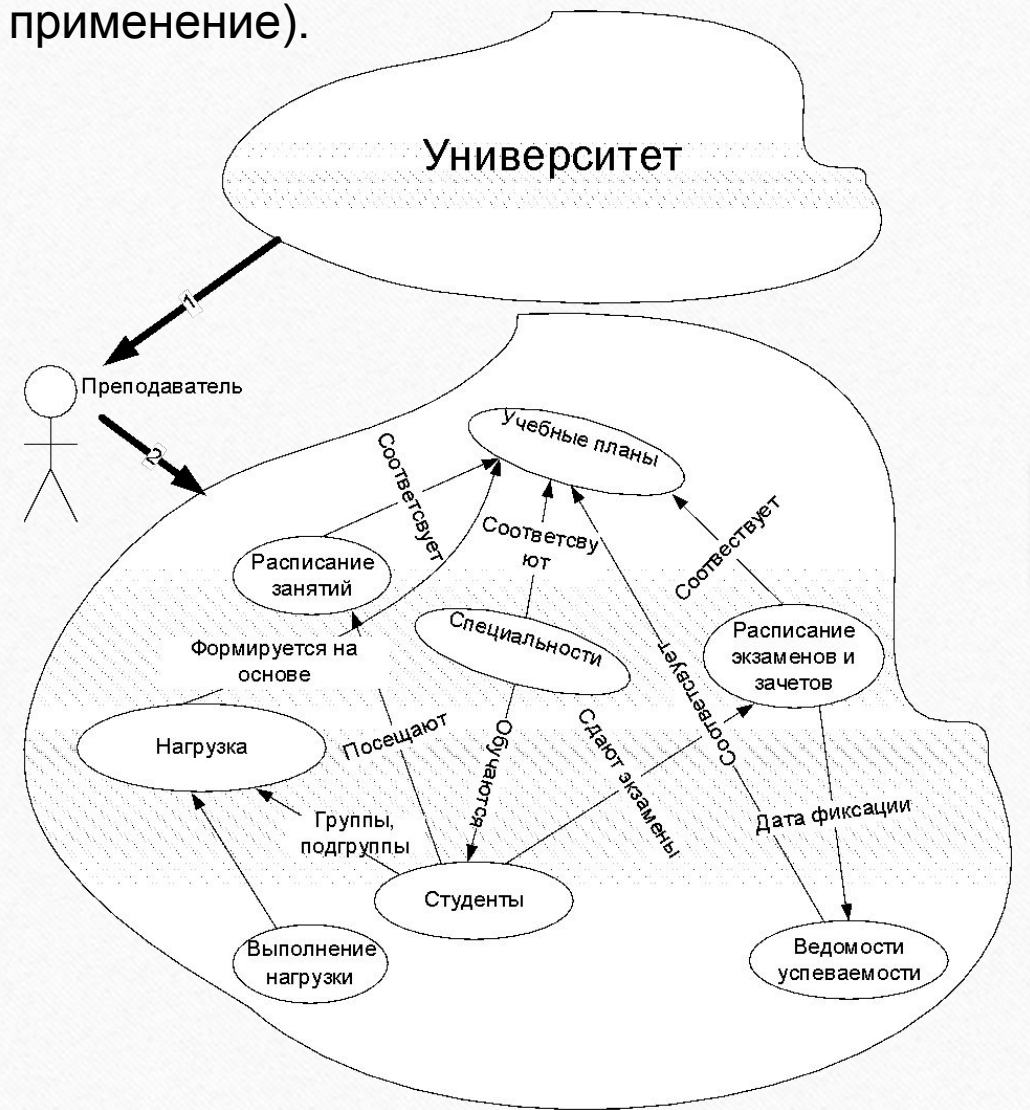
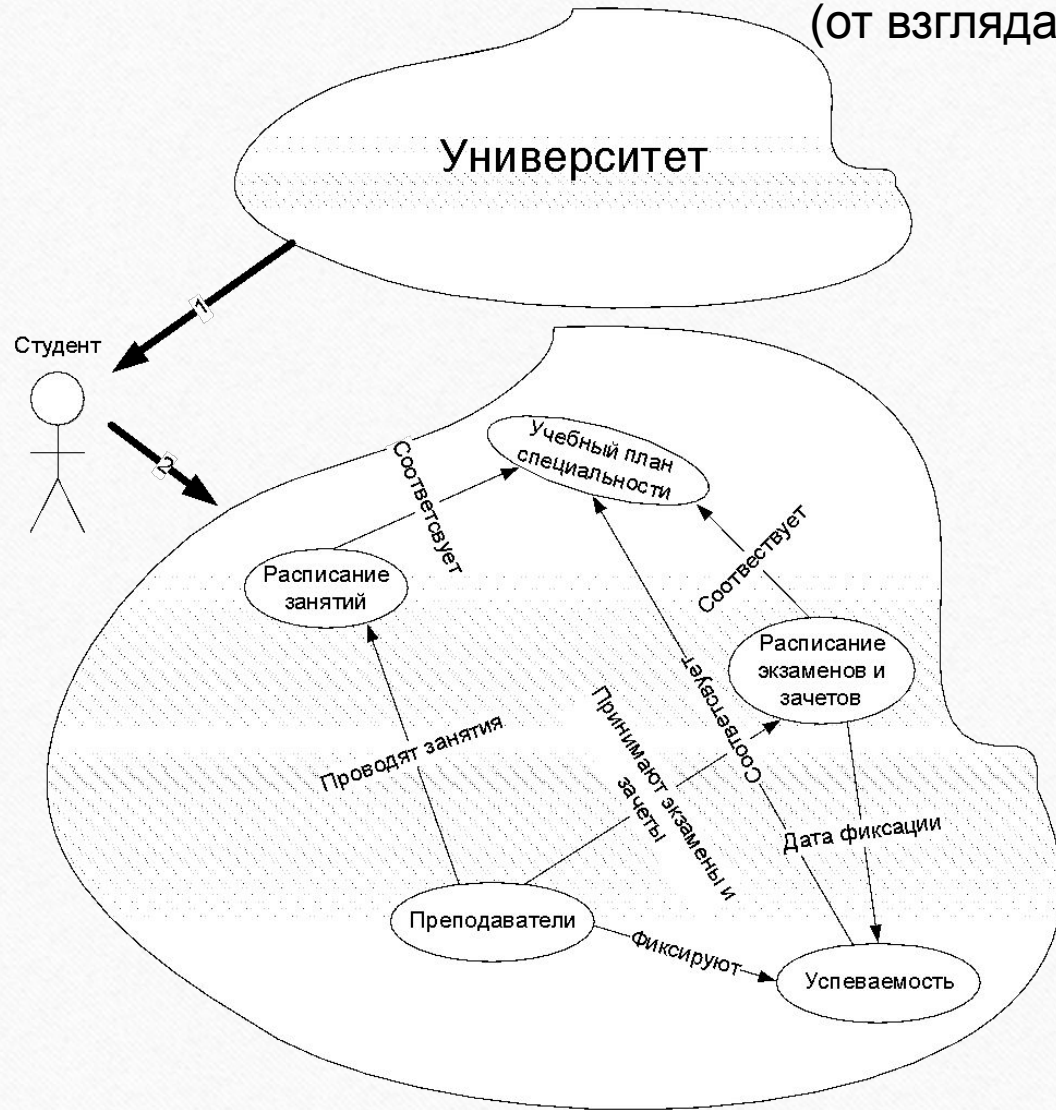
**Информация** в базе данных двух видов:

- 1) метаданные (свойства данных, схема, модель и пр.);
- 2) собственно данные.

**База данных** должна иметь языковые средства для описания свойств данных (в реляционных базах данных SQL) и манипулирования данными.



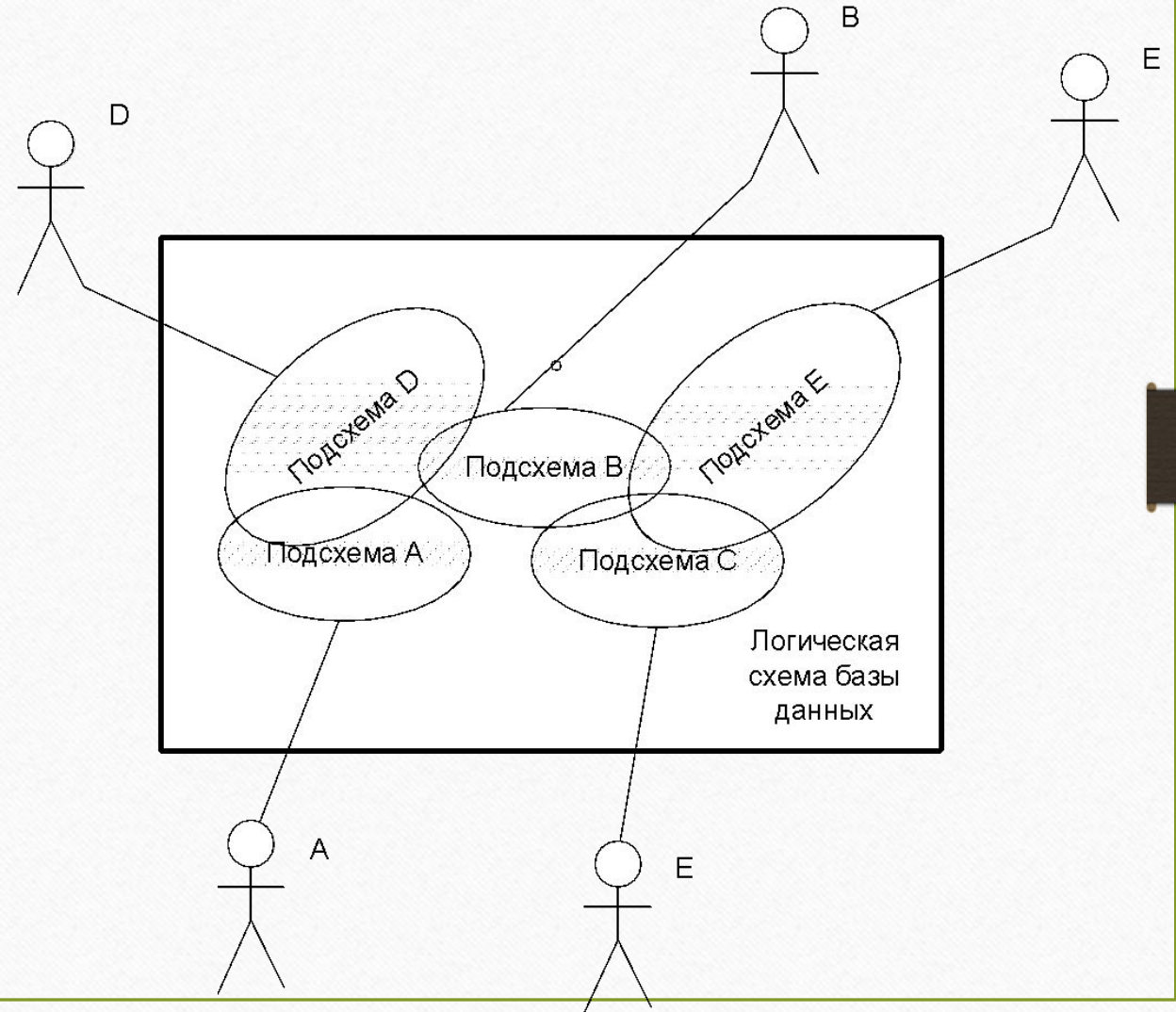
Состав объектов системы и структура зависит от назначения и цели создания базы данных (от взгляда на ее применение).



База данных, как правило, создается для многих пользователей. Каждый пользователь имеет свое представление о базе данных. Совокупность всех представлений – это **логическая схема данных**.

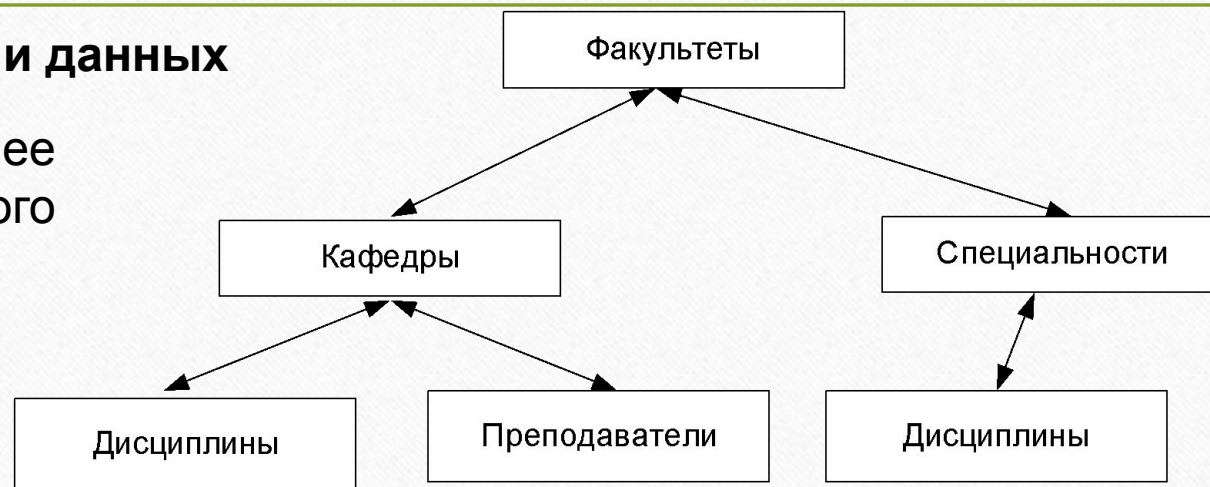
**Итоги:**

1. *База данных* - хранилище динамически обновляемой информации;
2. Информация отражает состояние некоторой предметной области (объекта) и должна быть *полезной, точной, актуальной и непротиворечивой*;
3. Информация представлена в виде метаданных (описание модели данных) и данных;
4. Каждый пользователь базы данных знает только о существовании данных, необходимых для решения его задач;
5. Совокупность всех представлений - это *логическая схема данных*.
6. Система управления базами данных - программная реализация *технологии* хранения, извлечения, обновления и обработки данных в базе данных.

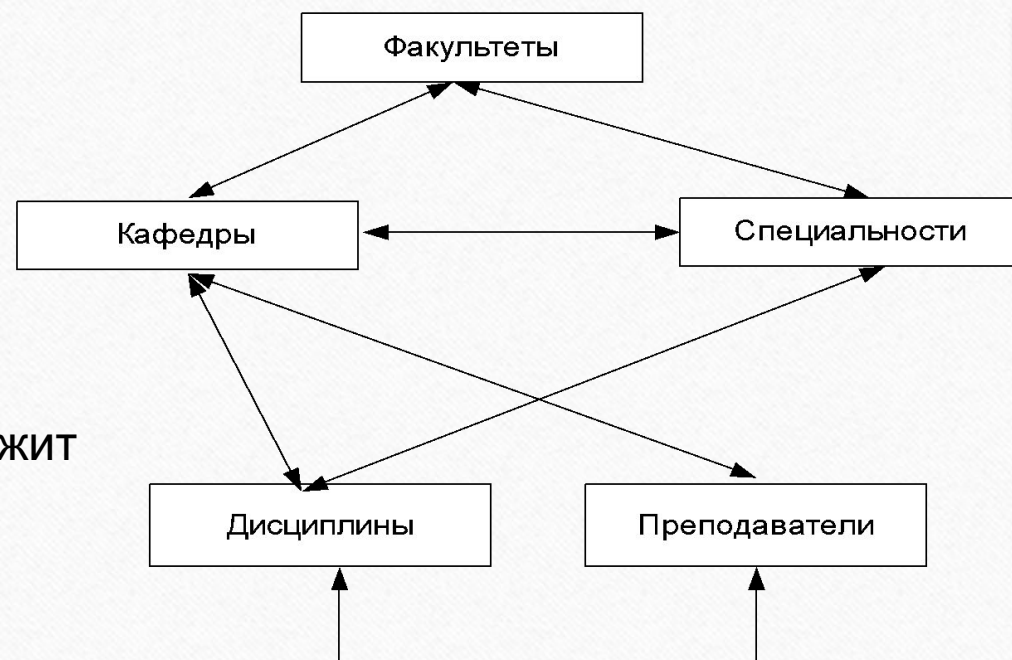


## Модели данных

1. **Иерархическая модель данных:** наиболее понятная и естественная для человеческого сознания (IBM IMS).



2. **Сетевая модель данных:** произвольные связи между данными (SOFTWARE AG ADABAS).



3. **Реляционная модель:** теоретическим базисом реляционного подхода к организации баз данных служит простой и мощный математический аппарат теории множеств и математической логики (Oracle (50-60%), Microsoft SQL Server (15-20%), IBM DB2 (5-10%)).

## Основы теории множеств

Теория множеств появилась на свет 7 декабря 1873 года. Основатель этой теории немецкий математик и философ **Георг Кантор** (1845–1918).

**Множество  $S$**  есть любое собрание определенных и различных между собой объектов нашей интуиции или интеллекта, мыслимое как единое целое. Эти объекты называются **элементами** множества  **$S$** .

**Пустое множество:**  $\emptyset$

**Мощность** множества – число элементов в конечном множестве  $|A|, |\emptyset| = 0$

**Конечные** (число элементов множества конечно  $A = \{a_1, a_2, a_3, \dots, a_n\}$ ) и **бесконечные** (содержит бесконечное число элементов  $B = \{b_1, b_2, b_3, \dots\}$ ) множества.

**Счетные** (все элементы которого могут быть занумерованы в последовательность) и **несчетные** множества.

Подмножества:  $A \subset B$

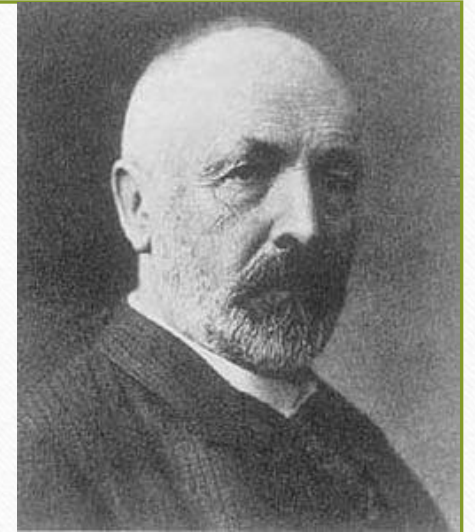
Равенство множеств:  $A = B \Leftrightarrow A \subset B \wedge B \subset A$

Операция пересечения множеств:  $C = A \cap B$

Операция объединения множеств:  $C = A \cup B$

Операция разности множеств:  $C = A \setminus B$

Операция дополнения множества:  $C = \bar{A}$





## Основные свойства операций над множествами:

1) свойства операции пересечения:

$$A \cap B = B \cap A \text{ (коммутативность);}$$

$$A \cap A = A \text{ (идемпотентность);}$$

$$(A \cap B) \cap C = A \cap (B \cap C) \text{ (ассоциативность);}$$

2) свойства операции объединения:

$$A \cup B = B \cup A \text{ (коммутативность);}$$

$$A \cup A = A \text{ (идемпотентность);}$$

$$(A \cup B) \cup C = A \cup (B \cup C) \text{ (ассоциативность);}$$

3) Совместные свойства операций объединения и пересечения:

$$A \cup (B \cap C) = (A \cup B) \cap (A \cup C) \text{ (дистрибутивность);}$$

$$A \cap (B \cup C) = (A \cap B) \cup (A \cap C) \text{ (дистрибутивность);}$$

4) свойства операции дополнения:

$$A \cup \bar{A} = U \quad A \cap \bar{A} = \emptyset$$

$$\overline{\bar{A}} = A \text{ (закон инволюции);}$$

$$\overline{A \cup B} = \bar{A} \cap \bar{B} \quad \overline{A \cap B} = \bar{A} \cup \bar{B} \text{ (закон де Моргана);}$$

5) свойства операции разности:

$$A \setminus A = \emptyset$$

$$A \setminus \emptyset = A$$

# Основы теории отношений

**Унарное отношение**  $R$  на множестве  $A$  - это любое подмножество:

$$R = \{\langle x \rangle \mid x \in A\}$$

**Декартово произведение множеств:**  $A = \{a_1, a_2\}$ ,  $B = \{b_1, b_2, b_3\}$ ,

$$A \times B = \{\langle a_1, b_1 \rangle, \langle a_1, b_2 \rangle, \langle a_1, b_3 \rangle, \langle a_2, b_1 \rangle, \langle a_2, b_2 \rangle, \langle a_2, b_3 \rangle\}$$

$$|A \times B| = |A| \times |B|, \quad A \times B = \{\langle x, y \rangle \mid x \in A, y \in B\}$$

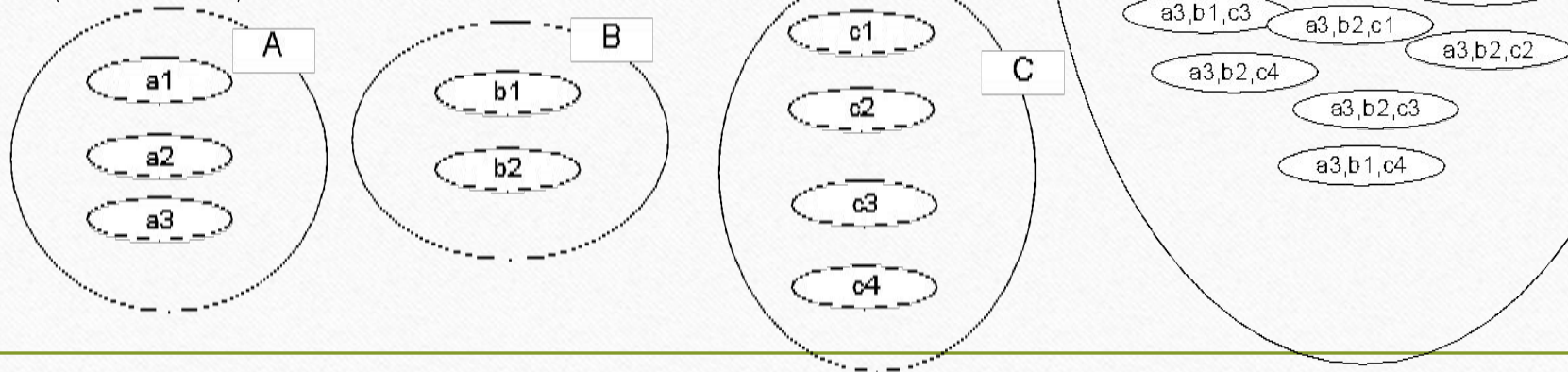
**Бинарное отношение:**  $R \subset A \times B$

**Тернарное отношение:**  $R \subset A \times B \times C$ ,

$$A \times B \times C = \{\langle x, y, z \rangle \mid x \in A, y \in B, z \in C\}$$

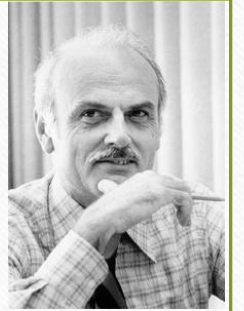
**n-арное отношение:**  $R \subset A_1 \times A_2 \times \dots \times A_n$ ,

$$A_1 \times A_2 \times \dots \times A_n = \{\langle x_1, x_2, \dots, x_n \rangle \mid x_i \in A_i, i = \overline{1, n}\}$$



## Реляционная алгебра (алгебра отношений) Кодда

Эдгар Франк Кодд (1923-2003). Ввел понятия: реляционная база данных, OLAP. Сформулировал основные 12 принципов реляционных СУБД.



1. **Данные представляются в виде таблиц.** Реляционная БД – это набор взаимосвязанных таблиц. Каждая строка таблицы (запись) содержит информацию об одном каком-то объекте, а все характеристики объектов записаны в столбцах (полях).
2. **Данные доступны логически.** Доступ к данным осуществляется не по номерам строк и столбцов, а только через идентификаторы таблицы. Идентификатором строки является первичный ключ (значения одной или нескольких колонок, однозначно идентифицирующих строки). Причем каждое значение первичного ключа в пределах таблицы должно быть уникальным. Ключ называется составным, если идентификация ряда осуществляется на основании значений нескольких колонок.
3. **NULL трактуется как неизвестное значение.**
4. **БД должна включать в себя метаданные.** БД хранит два вида таблиц: пользовательские и системные. В пользовательских таблицах хранятся данные, введенные пользователем. В системных таблицах хранятся метаданные: описание таблиц (название, типы и размеры колонок), индексы, хранимые процедуры и др.
5. **Должен использоваться единый язык для взаимодействия с СУБД (SQL).**
6. **СУБД должна обеспечивать альтернативный вид отображения данных.** Возможность строить виртуальные таблицы – представления (View) - динамические объединения нескольких таблиц. Изменения данных в представлении должны автоматически переноситься на исходные таблицы.

7. **Должны поддерживаться операции реляционной алгебры.** Записи реляционной БД рассматриваются как элементы множества, на котором определены операции реляционной алгебры.
8. **Должна обеспечиваться независимость от физической организации данных.** Приложения, оперирующие с данными реляционных БД, не должны зависеть от физического хранения данных (от способа хранения, формата хранения и др.).
9. **Должна обеспечиваться независимость от логической организации данных.** Приложения, оперирующие с данными реляционных БД, не должны зависеть от организации связей между таблицами. При изменении связей между таблицами не должны меняться ни сами таблицы, ни запросы к ним.
10. **За целостность данных отвечает СУБД.** Целостность данных – это готовность БД к работе. Говорят о физической целостности (сохранность информации на носителях и корректность форматов хранения данных) и логической целостности (непротиворечивости и актуальности данных).
11. **Целостность данных не может быть нарушена.** СУБД должна обеспечивать целостность данных при любых манипуляциях с данными.
12. **Должны поддерживаться распределенные операции.** Реляционная БД может располагаться на одном или многих компьютерах. При этом целостность данных должна обеспечиваться независимо от мест хранения данных.

## Определения, используемые Коддом

**Домен** – это семантическое понятие - множество всех возможных значений конкретного атрибута. Например: домен описывающий фамилии преподавателей – множество строк длиной от 1 до 50, содержащей буквы русского языка, дефис (Дунин-Мартинкевич), апостроф (Д'Артаньян).

**Таблица** - отношение;

**Атрибут** - имя столбца таблицы (имя атрибута обычно совпадает с именем домена);

**Заголовок таблицы** - множества всех атрибутов;

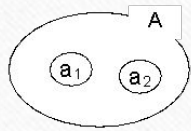
**Кортеж** - элемент отношения или строка таблицы;

**Строка таблицы** - кортеж.

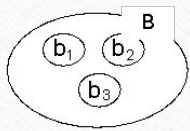
Реляционная алгебра представляет собой набор таких операций над отношениями, что результат каждой из операций также является отношением. Это свойство алгебры называется **замкнутостью**.

Операции над одним отношением называются *унарными*, над двумя отношениями — *бинарными*, над тремя — *тернарными* (таковые практически неизвестны).

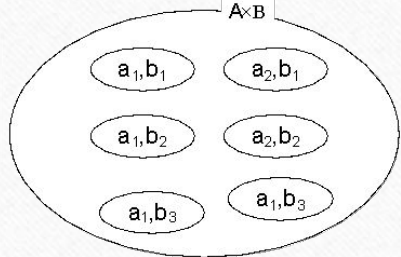
множество



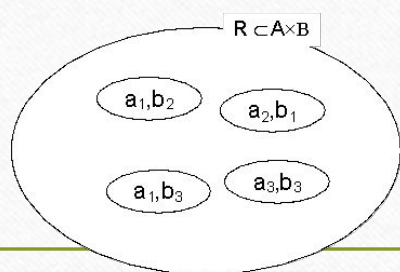
множество



декартово произведение

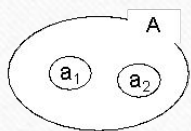


отношение



таблица

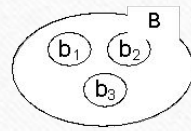
домен



A
a1
a2

таблица

домен



B
b1
b2
b3

таблица

атрибуты	
A	B
a1	b1
a1	b2
a1	b3
a2	b1
a2	b2
a2	b3

заголовок

таблица

атрибуты	
A	B
a1	b2
a1	b3
a2	b1
a2	b3

заголовок

Домен: Преподаватели



ПРЕПОДАВАТЕЛЬ

Преподаватель
Пацей Н.В.
Дятко А.А.
Акунович С.И.
Смелов В.В.

Домен: Дисциплины



ДИСЦИПЛИНА

Дисциплина
Моделирование
СУБД
Базы данных
Компьютерная графика
Объектно-ориентированное программирование
Распределенные системы

Унарные отношения

ЗАКРЕПЛЕНИЕ\_ДИСЦИПЛИНЫ\_ЗА\_ПРЕПОДАВАТЕМ

Преподаватель	Дисциплина
Пацей Н.В.	Объектно-ориентированное программирование
Дятко А.А.	Компьютерная графика
Дятко А.А.	Объектно-ориентированное программирование
Акунович С.И.	Моделирование
Смелов В.В.	Базы данных
Смелов В.В.	СУБД

ОТСУТСТВИЕ\_РЕЗЕРВА

Дисциплина
Компьютерная графика
Базы данных
СУБД
Моделирование

## Операции реляционной алгебры (алгебра Кодда):

1. **UNION** (объединение);
2. **INTERSECT** (пересечение);
3. **MINUS** (разность);
4. **TIMES** (декартово произведение);
5. **WHERE** (ограничение);
6. **PROJECT** (проекция);
7. **JOIN** (соединение);
8. **DIVIDE BY** (реляционное деление);
9. **RENAME** (переименование);
10. **:=** (присваивание).

## Операция UNION

Таблица А

Дисциплина	Семестр	Лектор
ОАиП	1	Пацей
ОАиП	2	Пацей
ДМ	1	Лащенко
МП	1	Бракович

Таблица В

Дисциплина	Семестр	Лектор
БД	3	Смелов
СУБД	4	Смелов
СУБД	5	СМелов

Таблица ( В UNION В )

Дисциплина	Семестр	Лектор
ОАиП	1	Пацей
ОАиП	2	Пацей
ДМ	1	Лащенко
МП	1	Бракович
БД	3	Смелов
СУБД	4	Смелов
СУБД	5	СМелов

## Операция INTERSECT

Таблица А

Дисциплина	Семестр	Лектор
ОАиП	1	Пацей
ОАиП	2	Пацей
ДМ	1	Лащенко
СУБД	4	Смелов

Таблица В

Дисциплина	Семестр	Лектор
БД	3	Смелов
СУБД	4	Смелов
СУБД	5	СМелов
МП	1	Бракович
ОАиП	2	Пацей

Таблица ( А INTERSECT В )

Дисциплина	Семестр	Лектор
ОАиП	2	Пацей
СУБД	4	Смелов

## Операция MINUS

Таблица А

Дисциплина	Семестр	Лектор
ОАиП	1	Пацей
ОАиП	2	Пацей
ДМ	1	Лащенко
СУБД	4	Смелов

Таблица В

Дисциплина	Семестр	Лектор
БД	3	Смелов
СУБД	4	Смелов
СУБД	5	СМелов
МП	1	Бракович
ОАиП	2	Пацей

Таблица ( А MINUS В )

Дисциплина	Семестр	Лектор
ОАиП	1	Пацей
ДМ	1	Лащенко

$$A \text{ INTERSECT } B = A \text{ MINUS } (A \text{ MINUS } B)$$



## Операция TIMES

Таблица А

Дисциплина	Семестр
ОАиП	1
ОАиП	2
ДМ	1
БД	3

Таблица В

Дисциплина	Лектор
БД	Смелов
МП	Бракович
ОАиП	Пацей

Таблица ( A TIMES B )

А.Дисциплина	Семестр	В.Дисциплина	Лектор
ОАиП	1	БД	Смелов
ОАиП	1	МП	Бракович
ОАиП	1	ОАиП	Пацей
ОАиП	2	БД	Смелов
ОАиП	2	МП	Бракович
ОАиП	2	ОАиП	Пацей
ДМ	1	БД	Смелов
ДМ	1	МП	Бракович
ДМ	1	ОАиП	Пацей
БД	3	БД	Смелов
БД	3	МП	Бракович
БД	3	ОАиП	Пацей

## Операция WHERE

Таблица С

А.Дисциплина	Семестр	В.Дисциплина	Лектор
ОАиП	1	БД	Смелов
ОАиП	1	МП	Бракович
ОАиП	1	ОАиП	Пацей
ОАиП	2	БД	Смелов
ОАиП	2	МП	Бракович
ОАиП	2	ОАиП	Пацей
ДМ	1	БД	Смелов
ДМ	1	МП	Бракович
ДМ	1	ОАиП	Пацей
БД	3	БД	Смелов
БД	3	МП	Бракович
БД	3	ОАиП	Пацей

Таблица ( C WHERE A.Дисциплина = В.Дисциплина )

А.Дисциплина	Семестр	В.Дисциплина	Лектор
ОАиП	1	ОАиП	Пацей
ОАиП	2	ОАиП	Пацей
БД	3	БД	Смелов

## Операция PROJECT

Таблица D

А.Дисциплина	Семестр	В.Дисциплина	Лектор
ОАиП	1	ОАиП	Пацей
ОАиП	2	ОАиП	Пацей
БД	3	БД	Смелов

Таблица PROJECT (А.Дисциплина, Семестр, Лектор)

А.Дисциплина	Семестр	Лектор
ОАиП	1	Пацей
ОАиП	2	Пацей
БД	3	Смелов

## Операция JOIN

Таблица А

Дисциплина	Семестр
ОАиП	1
ОАиП	2
ДМ	1
БД	3
СУБД	4
СУБД	5

Таблица В

Дисциплина	Лектор
БД	Смелов
МП	Бракович
ОАиП	Пацей
МС	Гурин
БД	Горбунова

Таблица ( А JOIN В )

Дисциплина	Семестр	Лектор
ОАиП	1	Пацей
ОАиП	2	Пацей
БД	3	Смелов
БД	3	Горбунова

$A \text{ JOIN } B = (A \text{ TIMES } B) \text{ WHERE } A.\text{Дисциплина} = B.\text{Дисциплина}$

## Операция DIVIDE

Таблица А

Дисциплина	Семестр	Лектор
ОАиП	1	Пацей
ОАиП	2	Пацей
ДМ	1	Лащенко
МП	1	Бракович
БД	3	Смелов
СУБД	4	Смелов
СУБД	5	Смелов
МС	4	Гурин
МС	5	Гурин

Таблица В

Семестр	Лектор
4	Гурин
3	Пацей
1	Пацей
1	Бракович

Таблица С

Лектор
Гурин
Лащенко
Пацей
Бракович

Таблица А DIVIDE В

Дисциплина
МС
ОАиП
МП

(A JOIN B) PROJECT (Дисциплина)

Таблица А DIVIDE С

Дисциплина	Семестр
ОАиП	1
ОАиП	2
ДМ	1
МП	1
МС	4
МС	5

(A JOIN C) PROJECT (Дисциплина, Семестр)

**Литература:** К. Дж. Дейт. Введение в системы баз данных. – М: Вильямс 2006 г.

### Итог лекции:

- 1. компьютерная база данных:** хранилище систематизированных и динамически обновляемых данных; данные хранятся на электронных носителях; доступ к данным обеспечивается СУБД; два типа информации: метаданные и сами данные;
- 2. свойство данных:** независимость от их применения, полезность, полнота, точность, достоверность, непротиворечивость, актуальность;
- 3. проектирование:** предметная область, системный анализ, логическая схема, реализация;
- 4. модели данных:** иерархическая, сетевая, реляционная;
- 5. реляционная алгебра Кодда:** домен, атрибут, таблица, заголовок таблицы, кортеж, **UNION**, **INTERSECT**, **MINUS**, **TIMES**, **WHERE**, **PROJECT**, **JOIN**, **DIVIDE**, **RENAME**, :=.