

# Практическое занятие №1

# Терминология

- **База данных** – коллекция логически связанных данных и их описаний, предназначенных для удовлетворения информационных потребностей предприятия (организации).
- **Модель данных** — интегрированная совокупность концепций для описания данных (и манипулирования ими), взаимосвязей между данными и ограничений на данные, имеющих место в организации.

# Стадии жизненного цикла разработки системы с БД

- Планирование БД
- Определение системы
- **Сбор и анализ требований**
- Проектирование БД
  - Концептуальное
  - Логическое (выбор СУБД )
  - Физическое
- Прототипирование (и новая итерация проектирования БД)
- Проектирование приложений (параллельно с проектированием БД)
- Реализация
- Преобразование и загрузка данных
- Тестирование
- Функционирование

# Подготовка технических требований для приложения БД

Включает проведение интервью с сотрудниками с целью сбора информации по итогам которой могут быть сформулированы.

Примерами типичных вопросов могут быть следующие:

- Ваши должностные обязанности?
- Какого вида задачи выполняются Вами повседневно?
- С данными какого рода Вы обычно работаете?
- Какого типа отчеты Вы обычно используете?
- Дела какого типа Вам необходимо отслеживать?
- Какие услуги предоставляет Ваша компания своим Заказчикам?

# Пример технических требований может быть такой:

- Обработка (ввод, модификация и удаление) данных по отделениям.
- Обработка (ввод, модификация и удаление) данных по персоналу.
- Обработка (ввод, модификация и удаление) данных по объектам недвижимости, сдаваемым в аренду.
- Выполнение поиска по отделениям.
- Выполнение поиска по персоналу.
- Выполнение поиска по объектам недвижимости, сдаваемым *в аренду*.
- Формирование отчетов по отделениям.
- Формирование отчетов по персоналу.
- Формирование отчетов по объектам недвижимости, сдаваемым в аренду.

# Модель данных «сущность–связь»

- Чтобы добиться полного понимания характера данных и способов их использования в организации, необходимо применять в процессе обмена информацией между специалистами общую модель, которая не усложнена техническими подробностями и не допускает двойных толкований.
- Структура (схема) БД может быть описана с использованием различных языков и нотаций.
- Наиболее распространенным средством абстрактного представления структур баз данных является ER-модель, или модель «сущность–связь» (entity–relationship model).

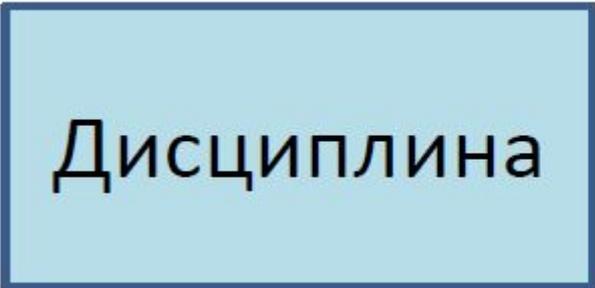
# Основные элементы ER- модели

- Типы сущностей
- Типы связей
- Свойства (атрибуты)

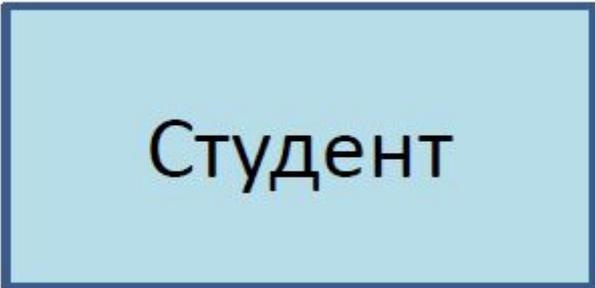
- **Сущность** – любой объект, информация о котором может (должна) быть представлена в базе данных.
- **Связь** – соединяет две или более сущностей и указывает вид взаимодействия между ними. Связи также должны быть представлены в базе данных. Связь можно понимать как сущность особого типа.
- **Свойства (атрибуты)** – описывают (детализируют) сущности и связи.

# Типы сущностей

- Каждый тип сущности изображается в виде прямоугольника с именем сущности внутри него.
- В качестве имени обычно применяется существительное в единственном числе.



Дисциплина



Студент

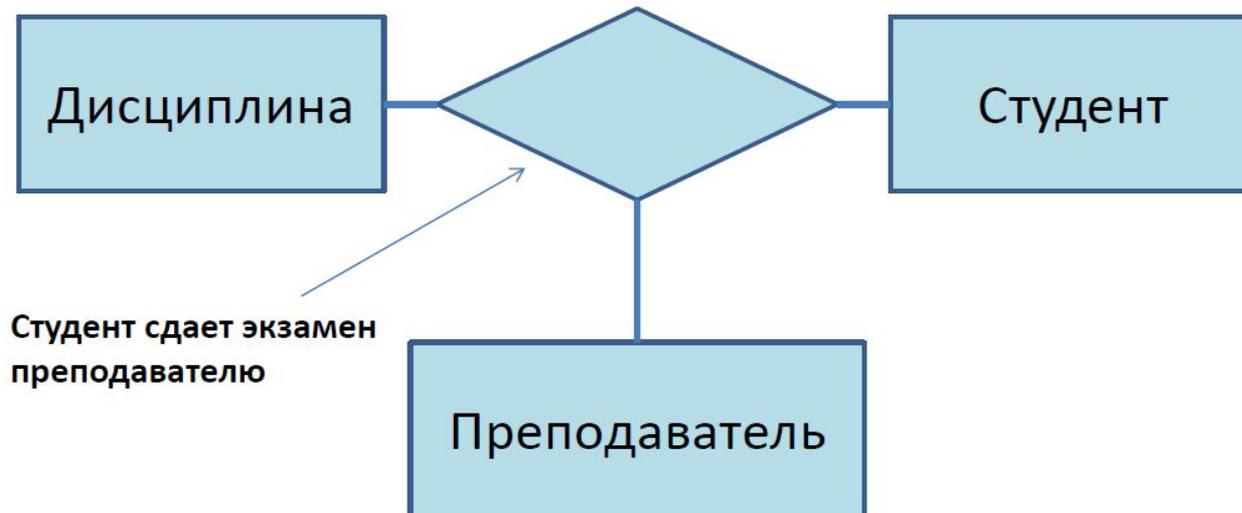
# Типы связей

- **Тип связей** (relationship type) – набор значимых ассоциаций между разными типами сущностей.
- Имя связи должно являться глаголом. По возможности в каждой конкретной ER-модели все имена связей должны быть уникальными.
- Связи должны иметь *направление*.



# Степень типа связи

- **Степень типа связи** (Degree of a relationship type) – количество типов сущностей, которые охвачены данной связью.
- Если в связи участвует два типа сущностей, то связь называется бинарной, если три типа сущностей – трехсторонней (тернарной).
- Для описания связей со степенью больше двух принято применять термин *сложная связь*.



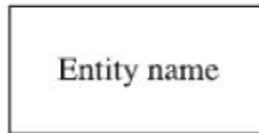
Наиболее распространенной степенью связи является двухсторонняя. Двухсторонние связи обычно обозначаются как связи

- «один к одному» (1 : 1)
- «один ко многим» (1 : M)
- «многие ко многим» (M : N).

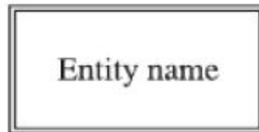
Для построения схем используют *нотацию* (систему обозначений), основанную на языке *UML* (Unified Modeling Language – единый язык моделирования).

Существуют также и другие нотации, например, нотация П. Чена, нотация «вороньи лапки» (Crow's Feet) и т.д.

# Нотация П.Чена



Сильная  
сущность



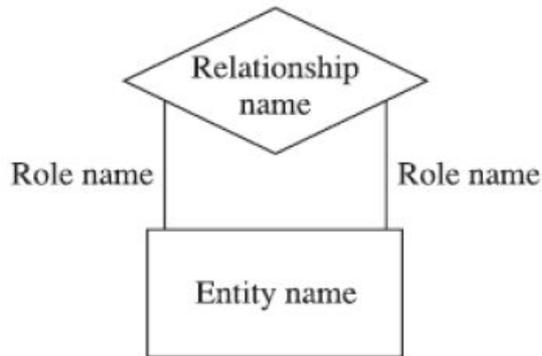
Слабая  
сущность



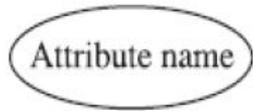
Связь



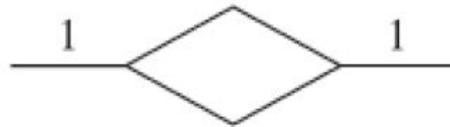
Связь,  
ассоциирова  
нная со  
слабой  
сущностью



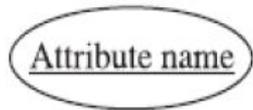
Рекурсивная связь с именами  
ролей, указывающими, какие  
функции выполняет  
сущность в связи



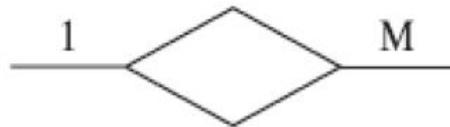
Атрибут



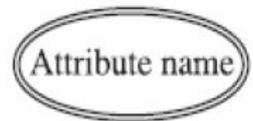
Связь «один к одному» (1:1)



Атрибут  
первичного  
ключа



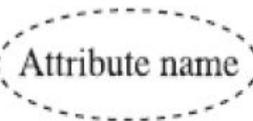
Связь «один ко многим» (1:M)



Многозначный  
атрибут



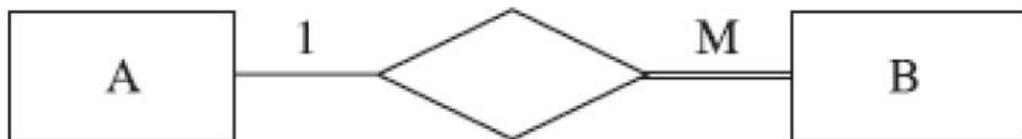
Связь «многие ко многим» (M:N)



Производный  
атрибут



Связь «один ко многим»  
с обязательным  
участием сущностей A и B

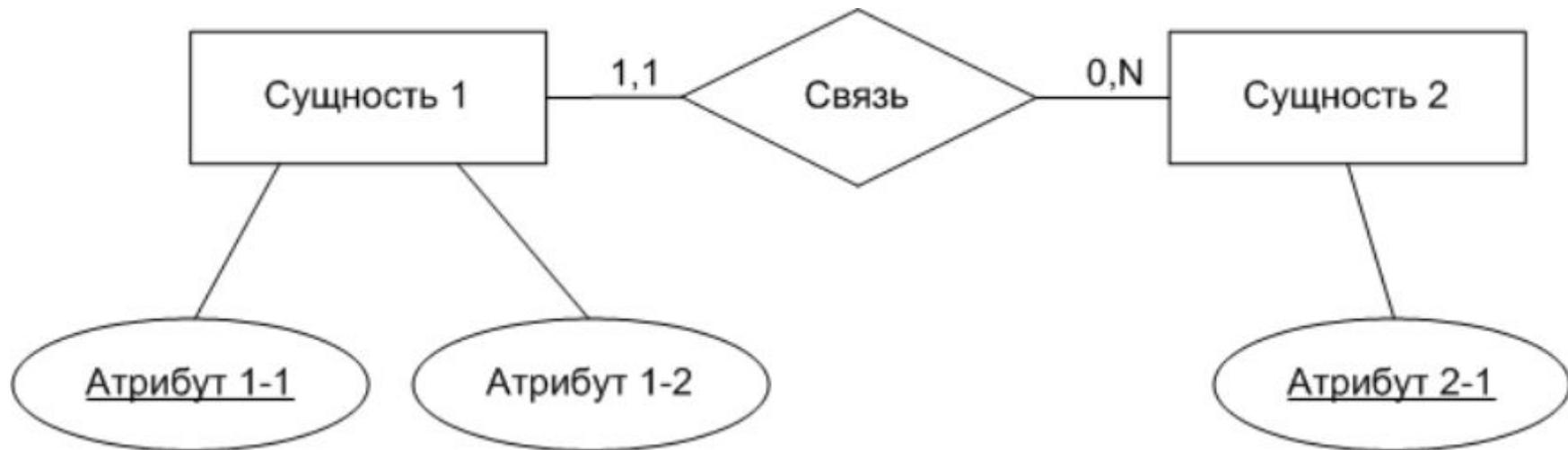


Связь «один ко многим»  
с необязательным участием  
сущности A и обязательным  
участием сущности B

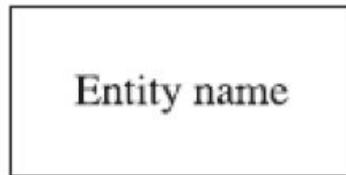


Связь «один ко многим»  
с необязательным участием  
сущностей A и B

# Пример нотации П. Чена



# Нотация «вороньи лапки»

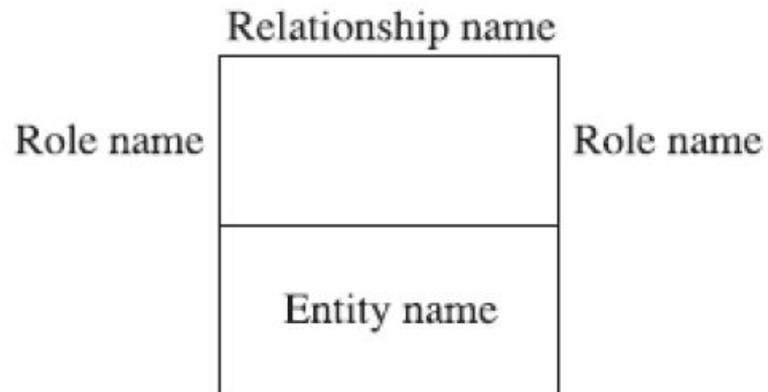


Сущность

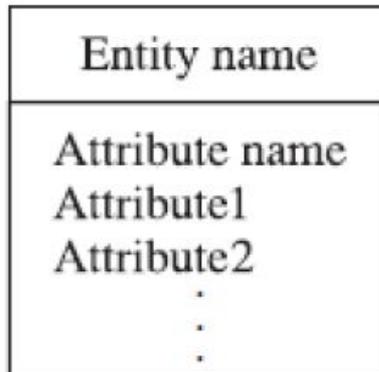
Relationship name

---

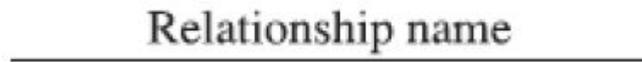
Связь



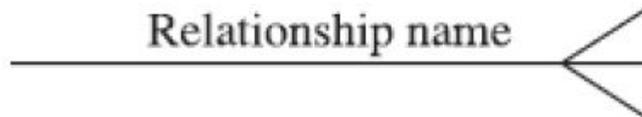
Рекурсивная связь с именами ролей, указывающими, какие функции выполняет сущность в связи



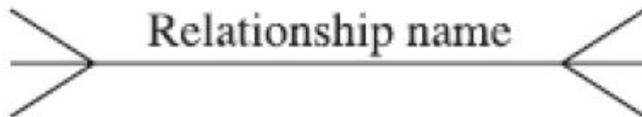
Атрибуты перечисляются в нижней части обозначения сущности. Атрибуты первичного ключа подчеркиваются. Многозначный атрибут помещается в фигурные скобки ({}).



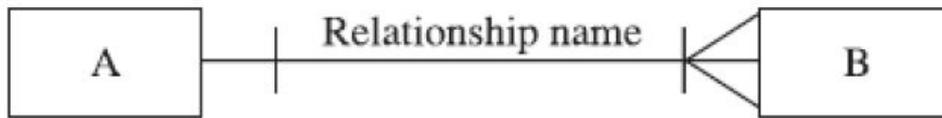
Связь «один к одному»



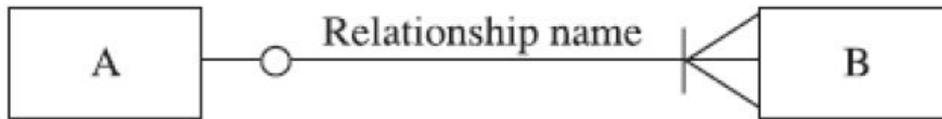
Связь «один ко многим»



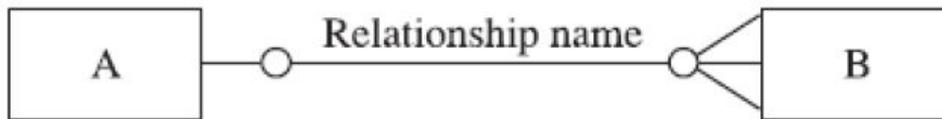
Связь «многие ко многим»



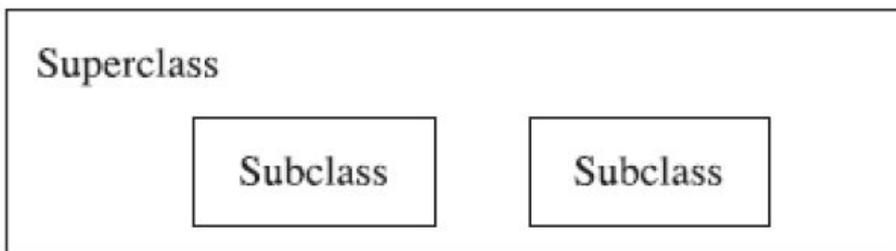
Связь «один ко многим» с обязательным участием сущностей A и B



Связь «один ко многим» с необязательным участием сущности A и обязательным участием сущности B

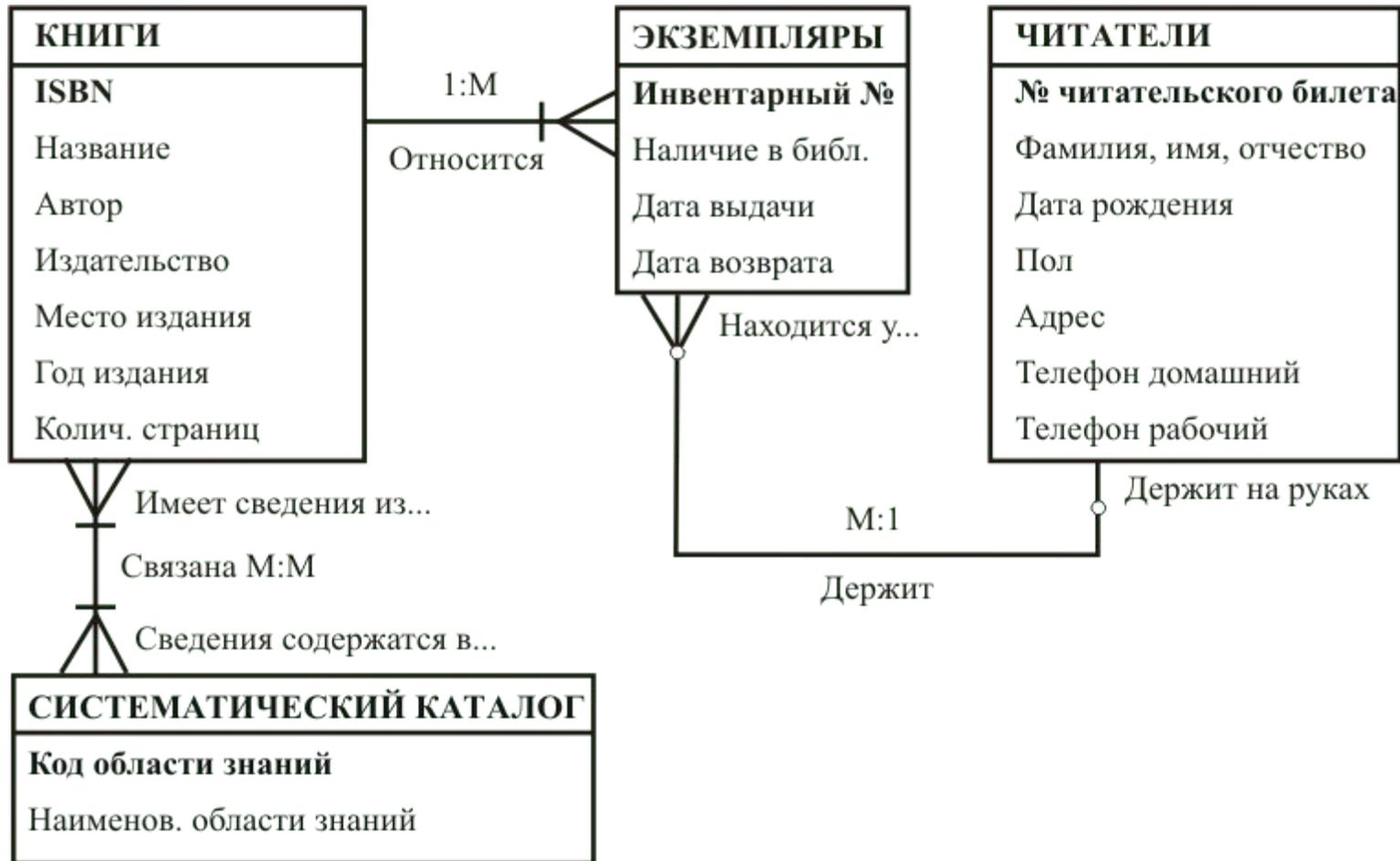


Связь «один ко многим» с необязательным участием сущностей A и B



Для представления иерархии уточнения/обобщения используются прямоугольники, заключенные в прямоугольник

# Пример нотации «вороньи лапки»



- Каждая *книга* может присутствовать в библиотеке в нескольких *экземплярах*, поэтому - связь 1:М. Если книга описана в сущности «Книги», то по крайней мере один экземпляр этой книги присутствует в библиотеке. Это означает, что со стороны книги связь обязательная.
- Каждый *читатель* может держать на руках несколько *экземпляров* книг. Между сущностями «Читатели» и «Экземпляры» установлена связь 1:М, и при этом она не обязательная с двух сторон.
- Каждая *книга* может содержать сведения из нескольких *областей знаний*, а с другой стороны, в библиотеке может присутствовать множество *книг*, относящихся к одной и той же *области знаний*, поэтому необходимо установить между сущностями «Системный каталог» и «Книги» связь М:М, обязательную с двух сторон.

# Задание

Разработайте диаграмму «сущность-связь» (рекомендуется использовать нотацию Чена) или диаграмму классов UML предметной области.

Примеры бесплатных утилит для моделирования в нотациях ER и UML:

- - ArgoUML (только UML);
- - Dia (только UML);
- - Software Ideas Modeler (UML и ER);
- - Umbrello (UML и ER).