

## Модель «сущность – связь» (ER – модель)

*Entity*-сущность, *Relationship*- связь

### **ER-модель:**

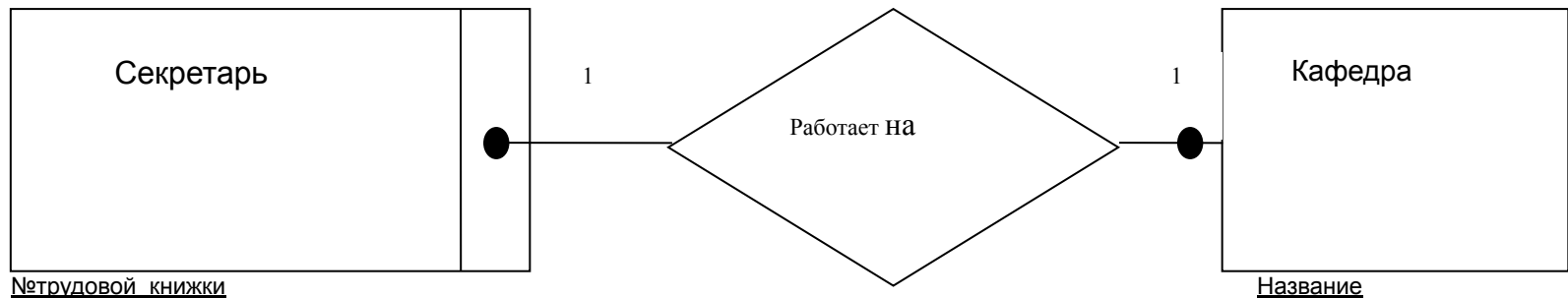
- Является **семантической**, а не реляционной, моделью. Моделирование предметной области базируется на использовании **графических диаграмм**, включающих небольшое число разнородных компонентов (ER-диаграмм).
- Предложена в 1976 году Питером Пин-Шен Ченом.
- Используется не только при ручном проектировании БД методом «сущность – связь», но и во всех **CASE** – средствах проектирования БД.

# Основные понятия ER – модели

- **Сущность** - реальный или представляемый объект, имеющий значение при решении задачи, информация о котором должна храниться и быть доступной.
  - Именуется , как правило, существительным,
  - Имеет экземпляры, каждый из которых должен быть однозначно идентифицируем,
  - На диаграммах изображается прямоугольником с ее именем.
- **Атрибут сущности** – свойство сущности, важное для решения задачи.
  - На диаграмме перечисляются внутри сущности (нотация Чена и Баркера) или под прямоугольником, изображающим сущность (нотация Хау).
  - Допустимо изображение на диаграммах не всех атрибутов сущности, а только ее идентификатора.
- **Связь** – ассоциация (зависимость) устанавливаемая между двумя или более сущностями.
  - именуется обычно глаголом.
  - чаще всего используются **бинарные** связи, то есть связи между двумя разными сущностями или сущностью и ею же самой. В последнем случае связь называется **рекурсивной**.
  - в связи выделяются два конца, для каждого конца указывается:
    - **имя**
    - **степень**, то есть сколько экземпляров сущности участвует в связи
    - **класс принадлежности сущности связи** (обязательная или необязательная).
  - На диаграммах связи изображают по-разному в разных нотациях.

# Изображение связи (нотация Хау)

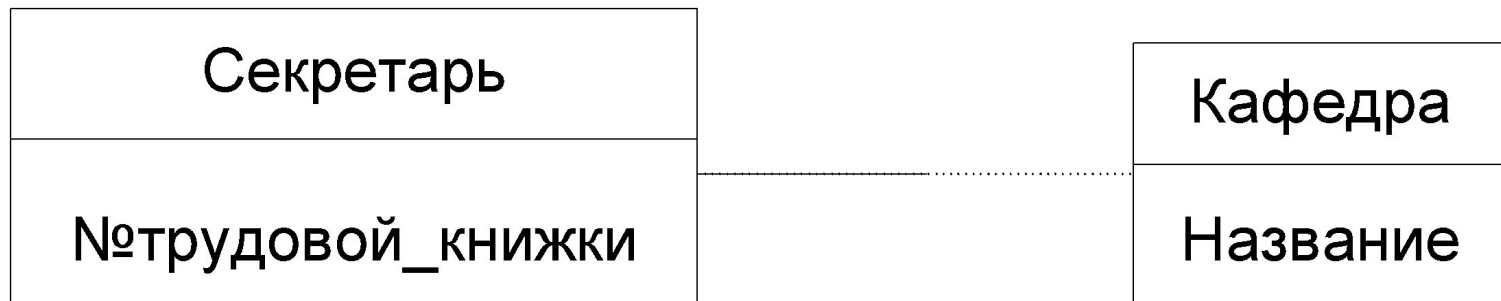
- Связь изображается ромбом.
- Имя связи помещается внутри ромба.
- Степень связи изображается цифрой 1, если в связи участвует один экземпляр сущности, и буквами, например: m, n, p, если в связи участвуют несколько экземпляров сущности.
- Обязательность – необязательность связи обозначается точкой на линии связи внутри сущности или снаружи соответственно.



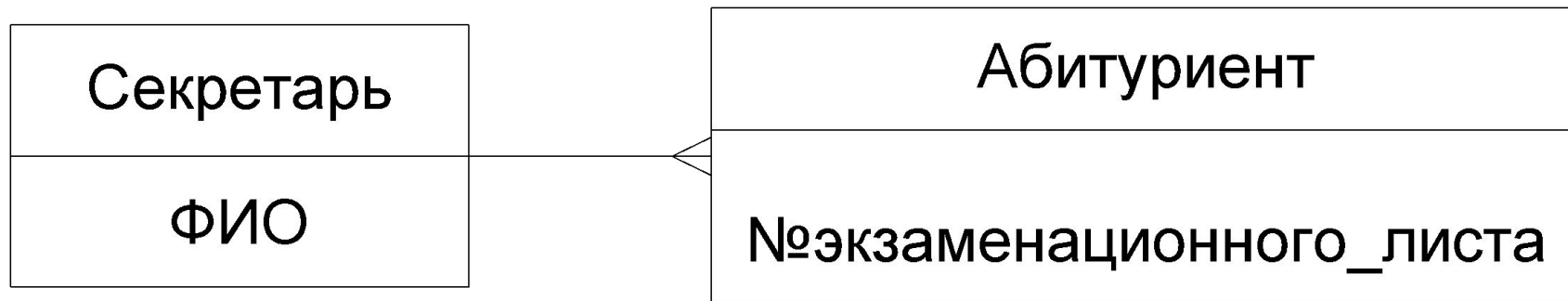
Связь между секретарем кафедры и кафедрой, является связью 1:1. Эта связь необязательна со стороны сущности «кафедра» и обязательна со стороны сущности «секретарь», т.к. кафедра может иметь только 1 секретаря, а временно может и не иметь секретаря, но секретарь кафедры обязательно должен работать на какой-то, причем только на одной, кафедре.

## Изображение связи (нотация Баркера)

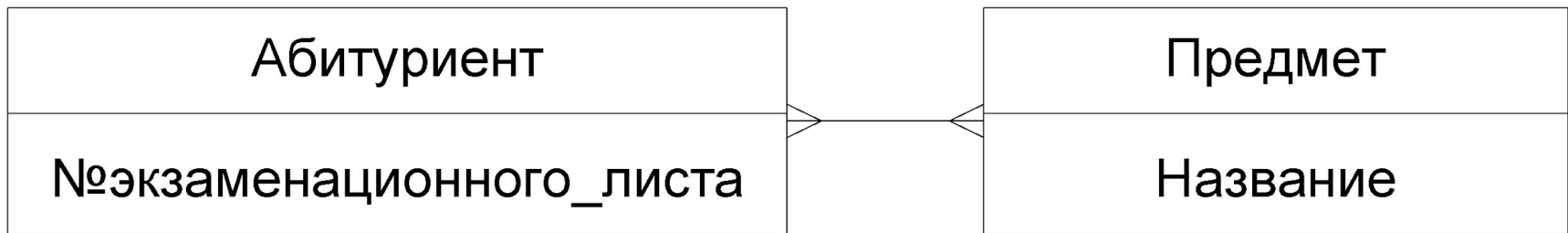
- Связь изображается линией, сплошной или пунктирной до середины в зависимости от того, является она обязательной или необязательной.
  - сплошная линия означает обязательность участия в связи каждого экземпляра сущности, со стороны которой эта линия проведена,
  - пунктирная – необязательность.
- Множественная связь имеет в нотации Баркера вид «вороньей лапы».



# Пример связи 1:n, обязательной с обеих сторон (нотация Баркера)



## Пример связи n:m, обязательной с обеих сторон (нотация Баркера)



Связь является обязательной с обеих сторон, поскольку каждый абитуриент сдает экзамен хотя бы по одному предмету, а каждый предмет сдает хотя бы один человек.

# Пример рекурсивной связи



Связь 1:n, поскольку у каждого человека может быть несколько детей, но отец у каждого один. Связь обязательна со стороны n и необязательна со стороны 1, так как человек не обязательно является отцом, но отец есть у каждого человека.

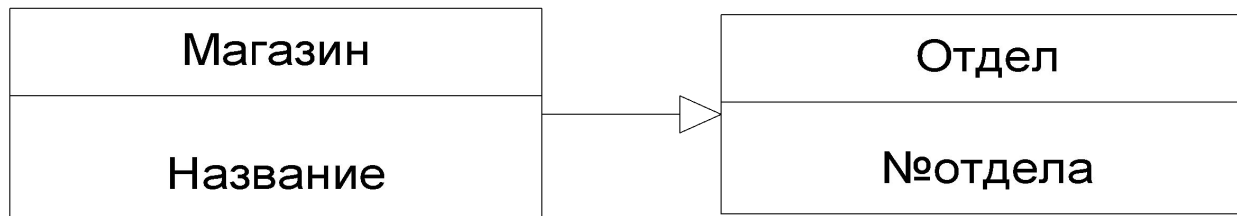
# Зависимая сущность

Зависимая сущность не может существовать сама по себе, а только относительно сущности, от которой она зависит.

Зависимая линия связи представляется стрелкой, указывающей в сторону зависимой сущности (нотация Чена).

**Пример1.** Сущность «задача» зависит от сущности «проект», если каждая задача на предприятии решается в рамках какого-нибудь проекта

**Пример2.** Сущность «отдел в магазине» зависит от сущности «магазин», поскольку вне магазина отдел существовать не может.



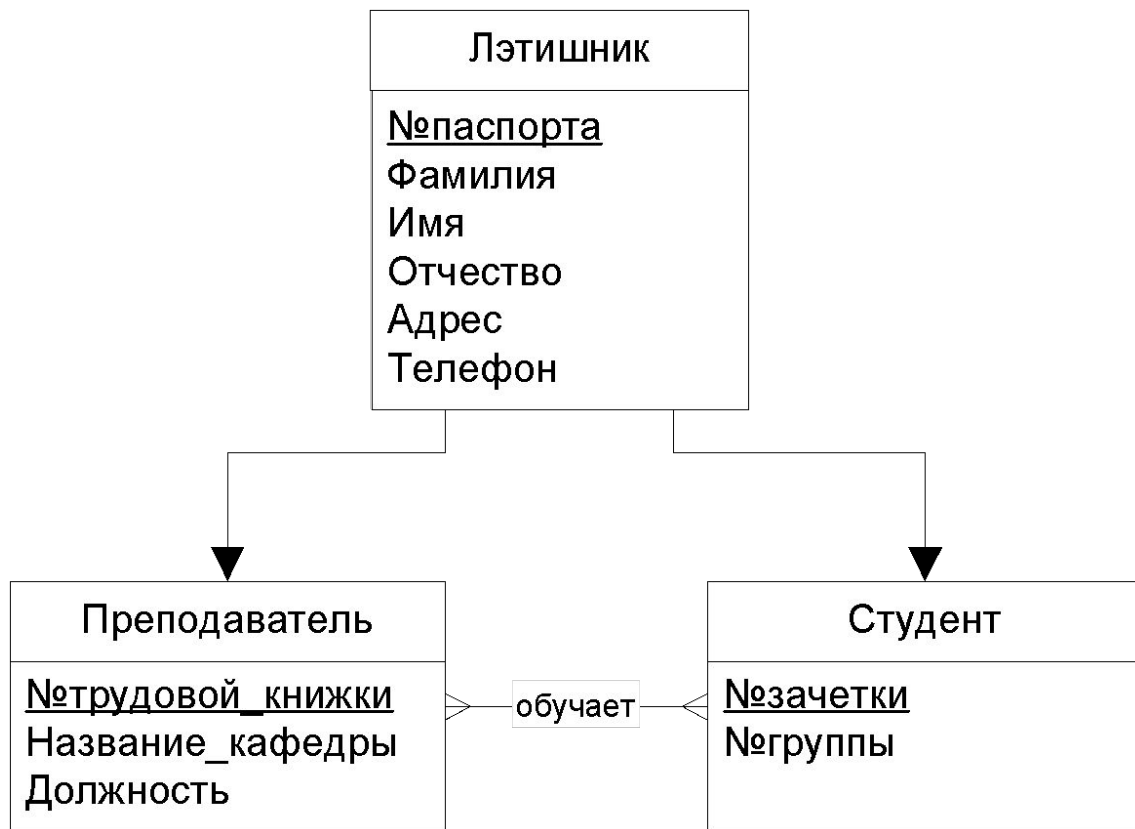


# Общая сущность и ее категории

## (понятия модели Чена)

- ❑ У Баркера понятию общей сущности соответствует **супертип**, а категории - **подтип**, у Хау - соответственно **сущность** и **роль**.
- ❑ **Пример.**  
Общая сущность – «Лэтишник».  
Категории – «Студент» и «Преподаватель».  
Сущности «Студент» и «Преподаватель»
  - a) имеют некоторые общие атрибуты, например, **номер паспорта, фамилию, имя, отчество, адрес и телефон,**
  - b) чем-то отличаются друг от друга.
    - ❑ сущность «Студент», например, имеет атрибуты: «номер зачетки», «номер группы»,
    - ❑ сущность «Преподаватель» имеет атрибуты «номер трудовой книжки», «название кафедры», на которой он работает, и «должность».
- ❑ Общая сущность обязательно имеет однозначный идентификатор. Категории могут не иметь дополнительного идентификатора, отличного от идентификатора общей сущности.
- ❑ Между категориями может существовать связь.

Пример использования в ER - модели категорий сущности, имеющих специфические по сравнению с общей сущностью атрибуты и связанных друг с другом



# Пример связи третьего порядка

Проводник обслуживает несколько озер, каждое озеро обслуживается несколькими проводниками, в каждом озере водится несколько видов рыбы, и каждый вид рыбы может водиться в разных озерах, каждый проводник предпочитает ловить определенные виды рыбы.

В ситуации, когда проводнику безразлично, в каком озере ловить тот или иной вид рыбы, достаточно трех бинарных связей, но если проводник предпочитает ловить рыбу вида 1 в озере А, а рыбу видов 2 и 3 – в озере В, то для описания такой ситуации потребуется трехсторонняя связь.



Если в связи одновременно участвуют три и более сущностей, такая связь называется ***n* – сторонней**, ***n*** – количество участвующих в связи сущностей.

# Правила генерации отношений из ER-диаграмм (правило 1)

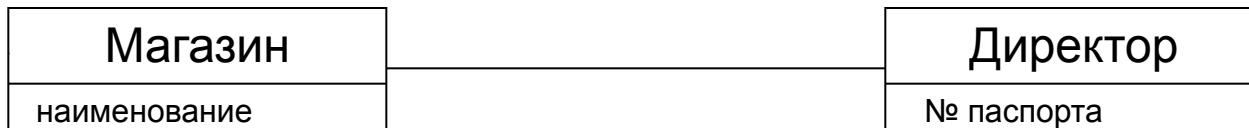
Если :

- степень бинарной связи равна 1:1 ,
- класс принадлежности обеих сущностей является обязательным,

То:

- требуется только 1 отношение.
- его первичным ключом может быть ключ любой из двух сущностей.

Например, имеем две сущности: сущность «Магазин» с атрибутами наименование, факс, телефон, адрес, специализация и сущность «Директор» с атрибутами № паспорта, ФИО. Каждый магазин обязательно имеет директора, а человек, являющийся директором, может быть директором только одного магазина.



Получим одно отношение **Магазин** (наименование, факс, телефон, адрес, специализация, № паспорта директора, ФИО директора)

# Правила генерации отношений из ER-диаграмм (правило 2)

Если:

- степень бинарной связи 1:1,
- класс принадлежности одной сущности является обязательным, а другой необязательным,

То:

- требуется 2 отношения (по одному на каждую сущность),
- ключ сущности является ключом соответствующего отношения,
- ключ сущности, для которой класс принадлежности необязателен, добавляется в качестве атрибута в другое отношение.

Например, имеем сущность «Человек» с атрибутами № паспорта, ФИО, год рождения, адрес и сущность «Зачетка» с атрибутами № зачетки, учебное заведение, факультет, когда выдана. Не каждый человек, а только студент, имеет зачетку, но зачетка обязательно кому-то принадлежит.



Получим два отношения: **Человек** ( № паспорта, ФИО, год рождения, адрес) и **Зачетка** ( № зачетки, учебное заведение, факультет, когда выдана, № паспорта).

# Правила генерации отношений из ER-диаграмм

## (правило 3)

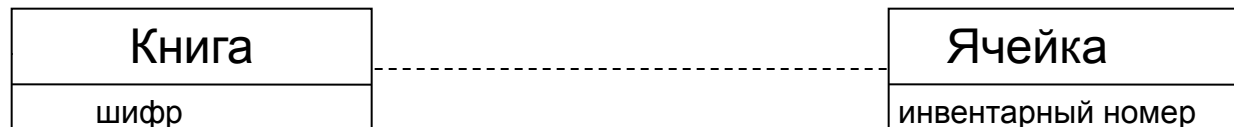
Если:

- степень бинарной связи равна 1:1,
- класс принадлежности ни одной из сущностей не является обязательным,

То:

- требуется 3 отношения (по одному для каждой сущности – объектные, плюс еще одно – связное),
- ключами объектных отношений являются ключи соответствующих сущностей,
- связное отношение содержит ключи объектных отношений в качестве своих атрибутов.

Например, имеем сущность «**Книга в библиотеке**» с атрибутами шифр, название, автор, год издания. Книга может находиться на полке в зале подручного фонда, а может и в запасниках. Если книга находится в зале, она занимает одну определенную ячейку на полке, в этой ячейке может быть несколько экземпляров одной книги, но не может быть другой книги, при этом ячейка может быть и пуста. Сущность «**Ячейка**» имеет атрибуты инвентарный номер, № зала, № стеллажа, № полки.



Получим три отношения: **Книга**(шифр, название, автор, год издания), **Ячейка** (инвентарный номер, № зала, № стеллажа, № полки), **Подручный\_Фонд** (шифр, инвентарный номер, количество экземпляров).

# Правила генерации отношений из ER-диаграмм

## (правило 4)

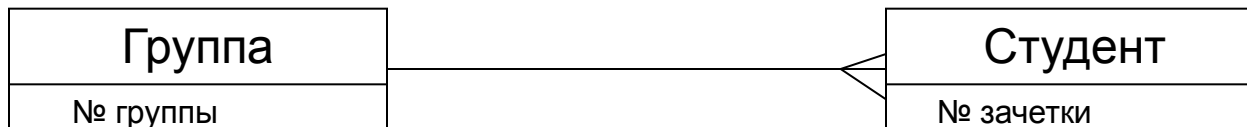
Если:

- степень бинарной связи равна 1:n,
- класс принадлежности n-связной сущности является обязательным,

То:

- требуются 2 отношения (по одному для каждой сущности),
- ключ сущности является ключом соответствующего отношения,
- ключ односвязной сущности добавляется как атрибут в отношение, соответствующее n-связной сущности.

Например, сущность «Группа» имеет атрибуты № группы, факультет, специальность, курс. Группа состоит из студентов. Сущность «Студент» имеет атрибуты № зачетки, ФИО, адрес, год рождения. В группе обычно несколько студентов. Каждый студент обязательно относится к какой-нибудь группе.



Получим два отношения: **Группа**( № группы, факультет, специальность, курс) и **Студент**( № зачетки, ФИО, адрес, год рождения, № группы).

# Правила генерации отношений из ER-диаграмм

## (правило 5)

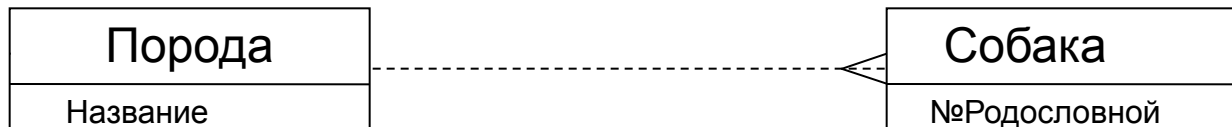
Если:

- степень бинарной связи равна 1:n,
- класс принадлежности n-связной сущности является необязательным,

То:

- требуются 3 отношения (по одному для каждой сущности и одно отношение связи)
- ключами объектных отношений являются ключи соответствующих сущностей,
- связное отношение содержит ключи объектных отношений в качестве своих атрибутов.

Например, в клубе собаководства есть перечень всех пород собак, которые будут представлены на очередной выставке, и список клубных собак. Сущность «**Порода**» имеет атрибуты: название породы, средний рост, средний вес, назначение, а сущность «**Собака**» имеет атрибуты: № родословной, кличка, возраст. Связь между сущностями «Порода» и «Собака» – 1:n, причем необязательная с обеих сторон, поскольку собак какой-то породы в данном клубе может и не быть, но могут быть собаки пород, не представленных на очередной выставке.



Получим три отношения: **Порода**(НазваниеПороды, СреднийРост, СреднийВес, Назначение), **Собака**( №Родословной, Кличка, Возраст) и **УчастникиВыставки**(НазваниеПороды, №Родословной)



# Правила генерации отношений из ER-диаграмм (правило 6)

**Если:**

- **степень бинарной связи равна  $m:n$ ,**

**То:**

- **независимо от класса принадлежности требуются 3 отношения (два объектных и одно связное).**

**Например,** связь между сущностью **Книга** с атрибутами Шифр, Название, Автор, ГодИздания и сущностью **Читатель** с атрибутами Билет, ФИО, Телефон является связью  $n:m$ .



Получим три отношения: **Книга**(Шифр, Название, Автор, ГодИздания), **Читатель**(Билет, ФИО, Телефон) и **Взял**(Шифр, Билет, Дата).

# Правила генерации отношений из ER-диаграмм (правило 7)

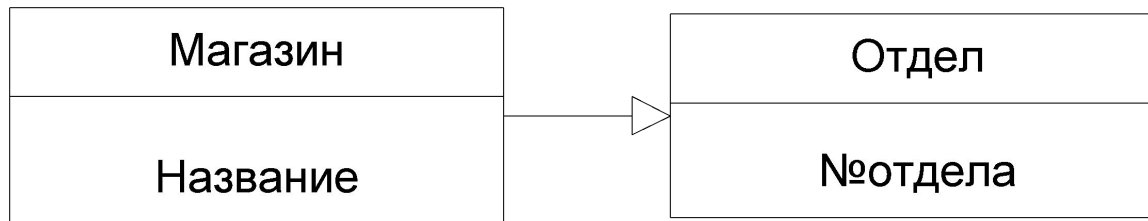
Если:

- одна сущность является зависимой по отношению к другой сущности, т.е. не может существовать без нее,

То:

- будут образованы два отношения – по одному для каждой сущности со своими ключами,
- в отношении, соответствующее зависимой сущности, добавится ключ отношения, соответствующего сущности, от которой она зависит,
- таким образом, ключом отношения, соответствующего зависимой сущности, будет составной ключ, включающий ключи обеих сущностей.

**Например**, сущность «отдел в магазине» зависит от сущности «магазин», поскольку вне магазина отдел существовать не может.



Получим два отношения: **Магазин**(Название, ...) и **Отдел** (Название, №Отдела, ..., ).

# Правила генерации отношений из ER-диаграмм

## (правило 8)

Если одна сущность является общей сущностью, а другие сущности - ее категориями, то возможны **два варианта** генерации отношений:

### Вариант 1:

- общая сущность служит источником для генерации отношения, ключом которого является ключ этой сущности.
- генерируются отношения, соответствующие каждой из категорий
- ключ общей сущности добавляется как атрибут в отношения, соответствующие категориям.

### Вариант 2:

- генерируются отношения, соответствующие каждой из категорий
- общая сущность не порождает отдельного отношения, а все ее атрибуты становятся атрибутами объектных отношений, порожденных категориями.

**Выбор варианта** генерации зависит от количества общих атрибутов:

- Если количество атрибутов общей сущности больше, чем количество атрибутов ее категорий, то обычно выбирают **вариант 1**.
- Если количество общих атрибутов не превышает 2-3, то смысла в создании самостоятельного отношения для общей сущности нет и выбирают **вариант 2**.

Если категории имеют связи друг с другом, то порождается дополнительно такое число отношений, которое определяется правилами 1-6.

# Пример использования правила 8



**Вариант 1 даст 4 отношения:**

**Лэтишник** (№паспорта, Фамилия, Имя, Отчество, Адрес, Телефон),

**Студент** (№зачетки, №паспорта, №группы),

**Преподаватель** (№трудовой книжки, №паспорта, Название кафедры, Должность),

**Обучается** (№зачетки, №трудовой книжки, ...).

**Вариант 2 даст 3 отношения:**

**Студент** (№зачетки, №паспорта, Фамилия, Имя, Отчество, Адрес, Телефон, №группы),

**Преподаватель** (№трудовой книжки, №паспорта, Фамилия, Имя, Отчество, Адрес, Телефон, Название кафедры, Должность),

**Обучается** (№Зачетки, №трудовой книжки, ...).

# Правила генерации отношений из ER-диаграмм (правило 9)

Если имеет место **n-сторонняя связь между сущностями,**

**То:**

- генерируется **n+1** отношение - по одному для каждой сущности со своим первичным ключом и одно для связи,
- связное отношение имеет среди своих атрибутов ключи каждой участвующей в связи сущности.

Например, имеем 3-х стороннюю связь между сущностями **Проводник, Рыба и Озеро**



Получим 4 отношения:

Проводник (ИмяПроводника, ...),

Рыба (Вид, ...),

Озеро (НазваниеОзера, ...),

ПредпочитаетЛовить (ИмяПроводника, Вид, НазваниеОзера).

# Основные этапы проектирования БД методом

## «СУЩНОСТЬ-СВЯЗЬ»

После того, как, используя перечисленные **9 правил**, проектировщик получил **предварительные отношения**, он должен сделать следующее:

- Подготовить список значимых для задачи атрибутов, если это еще не было сделано, и распределить их по предварительным отношениям.
- Определить функциональные зависимости (ФЗ) между атрибутами каждого отношения.
- Проверить соответствие каждого отношения нормальной форме Бойса - Кодда (НФБК).
- Если некоторым атрибутам не нашлось места в предварительных отношениях, или есть отношения, которые не находятся в НФБК, следует перестроить ER- диаграмму.

Данная последовательность действий представляет собой метод проектирования баз данных, который носит название **ER-метода**.

---

## Пример проектирования БД методом «сущность-связь». Постановка задачи

Пусть требуется создать базу данных для работника деканата.

В БД должны храниться **сведения о группах и студентах:**

ФИО студента, № зачетной книжки, № группы, кафедра, направление, год образования группы, адрес постоянной прописки студента, отметки о сданных зачетах и оценки, полученные студентом за курсовые работы и на экзамене по каждой из дисциплин в текущей сессии.

---

# Пример проектирования БД методом «сущность-связь».

## Выделение сущностей.

В предметной области можно выделить следующие сущности:

- **Студент** с атрибутами:
  - №\_зачетки,
  - ФИО,
  - Адрес.
- **Группа** с атрибутами:
  - №\_группы,
  - *Направление*,
  - Кафедра,
  - Год\_образования.
- **Предмет** с атрибутом:
  - Название\_предмета.

Поскольку направление характеризуется не только номером, но и **названием**, а также **сроком подготовки** и присваиваемой **квалификацией**, вместо атрибута «направление» добавим в предметную область **сущность**

- **Направление** с атрибутами:
  - №\_направления,
  - Название\_направления,
  - Срок\_подготовки,
  - Квалификация.



# Пример проектирования БД методом «сущность-связь». Добавление атрибутов

Студенты могут иметь в сессию по одному и тому же предмету до трех записей в зачетной книжке, если сдают и зачет, и экзамен, и курсовую работу. В ведомостях может быть и большее количество записей, если студент сделал несколько попыток сдачи экзамена или зачета. **Введем следующие атрибуты:**

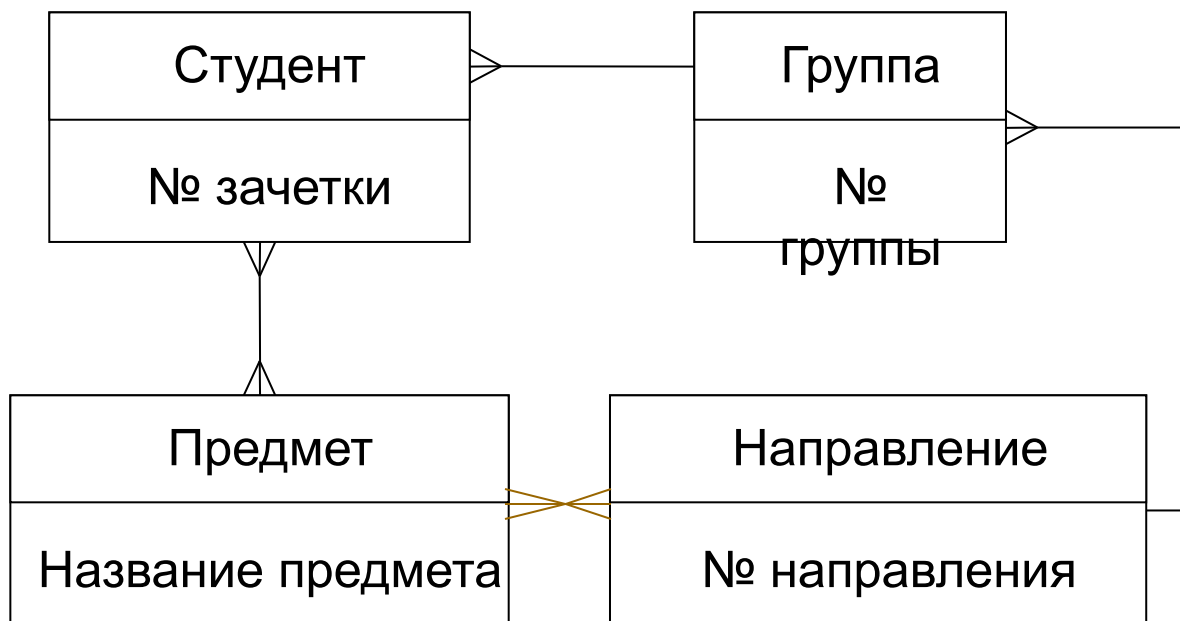
## Вариант1

- **Вид\_отчетности** с возможными значениями: **зачет, экзамен, курсовая работа**
- **Отметка** с возможными значениями: **отлично, хорошо, удовлетворительно, неудовлетворительно, неявка, зачет, незачет**
- **Дата** - ( дата сдачи студентом зачета, экзамена или курсовой работы по предмету )

## Вариант2

- **Отметка зачета** со значениями: **зачет, незачет, неявка**
- **Отметка экзамена** со значениями: **отлично, хорошо, удовлетворительно, неудовлетворительно, неявка**
- **Отметка курсовой работы** со значениями: **отлично, хорошо, удовлетворительно, неудовлетворительно, неявка**
- **Дата зачета**
- **Дата экзамена**
- **Дата курсовой работы**

# Пример проектирования БД методом «сущность-связь». ER – модель предметной области “Деканат”



**Студент** (№ зачетки, №\_группы)

**Группа** (№ группы, №\_напр-я)

**Специальность** (№ напр-я)

**Предмет** (Название предмета)

**Сдал** (№\_зачетки, Название\_предмета)

---

# Пример проектирования БД методом «сущность-связь».

## Набор предварительных отношений

- **Студент** (№\_зачетки, №\_группы, )
- **Группа** (№\_группы, №\_напр-я, )
- **Направление** (№\_напр-я, )
- **Предмет** (Название\_предмета)
- **Сдал** (№\_зачетки, Название\_предмета, )

### Атрибуты, еще не распределенные по отношениям:

ФИО\_студента, Адрес

Кафедра, Год\_образования

Название\_напр-я, Квалификация, Срок\_подготовки

Вид\_отчетности, Дата, Отметка

---

# Пример проектирования БД методом «сущность-связь».

## ФЗ между атрибутами предварительных отношений.

### Проверка на соответствие НФБК

ФЗ отношения **Студент** :

- №\_зачетки → №\_группы, ФИО\_студента, Адрес.

Других зависимостей нет, так как ФИО может повторяться и в пределах группы, адреса студентов, живущих в общежитии, также могут совпадать. Отношение **Студент** по определению находится в НФБК.

ФЗ отношения **Группа**:

- №\_группы → №\_напр-я, Кафедра, Год\_образования

Обратная зависимость неверна, т.к. в один год на кафедре может быть сформировано несколько групп одного направления. Отношение **Группа** находится в НФБК.

ФЗ отношения **Направление**:

- №\_напр-я → Название\_напр-я, Квалификация, Срок\_подготовки
- Название\_напр-я → №\_напр-я, Квалификация, Срок\_подготовки.

Как атрибут №\_напр-я, так и атрибут Название\_напр-я являются возможными ключами и детерминантами отношения. Отношение **Направление** находится в НФБК.

ФЗ отношения **Сдал**:

- №\_зачетки, Название\_предмета, Вид\_отчетности, Дата → Отметка.

Других зависимостей нет, следовательно, отношение **Сдал** находится в НФБК.

# Пример проектирования БД методом «сущность-связь».

## Реляционная модель предметной области “Деканат”

