

Базы данных

ЛЕКЦИЯ 2

**Проектирование баз данных.
Обобщенная
концептуальная модель.**

Алматы, 2023

Этапы проектирования базы данных

- 1. Анализ предметной области
- 2. Концептуальное проектирование
- 3. Логическое проектирование
- 4. Физическое проектирование

Концептуальное проектирование

- **Концептуальное проектирование** - построение семантической модели предметной области, то есть информационной модели наиболее высокого уровня абстракции. Такая модель создаётся без ориентации на какую-либо конкретную СУБД и модель данных. Термины «семантическая модель», «концептуальная модель» и «инфологическая модель» являются синонимами.
- Кроме того, в этом контексте равноправно могут использоваться слова «модель базы данных» и «модель предметной области» (например, «концептуальная модель базы данных» и «концептуальная модель предметной области»), поскольку такая модель является как образом реальности, так и образом проектируемой базы данных для этой реальности

Концептуальное моделирование

- Модель ER (модель сущность-связь) представляет собой способ графического представления логических отношений сущностей для создания базы данных.
- Модель ER была впервые предложена Питером Чен из Массачусетского технологического института (MIT) в 1970-х годах.

Сущность (Entity)

- Введем основные понятия, с помощью которых описывается предметная область.
- **Сущность (Entity)** или объект – это класс однотипных объектов, информация о которых имеет существенное значение для рассматриваемой предметной области. Сущность представляет собой множество экземпляров реальных или абстрактных объектов (людей, событий, состояний, предметов и т.п.).
- Если в системе обрабатывается информация об абитуриентах, сущностью может являться абитуриент, если обрабатывается информация об экзамене, то сущность – экзамен и т.п.
- Каждая сущность обладает определенным набором свойств (рассматриваем только свойства, представляющие интерес для пользователей в рамках проводимого исследования), которые запоминаются в информационной системе. Так, например, в качестве свойств сущности АБИТУРИЕНТ можно указать фамилию, дату рождения, место рождения, в качестве свойств сущности ЭКЗАМЕН – предмет, дату проведения экзамена, экзаменаторов.

Экземпляр сущности

- *Экземпляр сущности* — это конкретный представитель данной сущности. Каждый экземпляр сущности должен быть отличим от любого другого экземпляра той же сущности.
- Пример класса сущностей АБИТУРИЕНТ и конкретного экземпляра сущности показан на рис.

Класс сущностей

АБИТУРИЕНТ Фамилия Дата рождения Место рождения
--

Экземпляр сущности

Иванов 21.05.87 Нижний Новгород

Атрибут сущности

- Каждая сущность обладает одним или несколькими атрибутами.
- *Атрибут сущности* — это именованная характеристика, являющаяся некоторым свойством сущности.
- *Первичный ключ сущности* предназначен для уникальной идентификации каждого экземпляра сущности, т.е. представляет собой совокупность признаков, позволяющих идентифицировать объект.



Связи (Relationships)

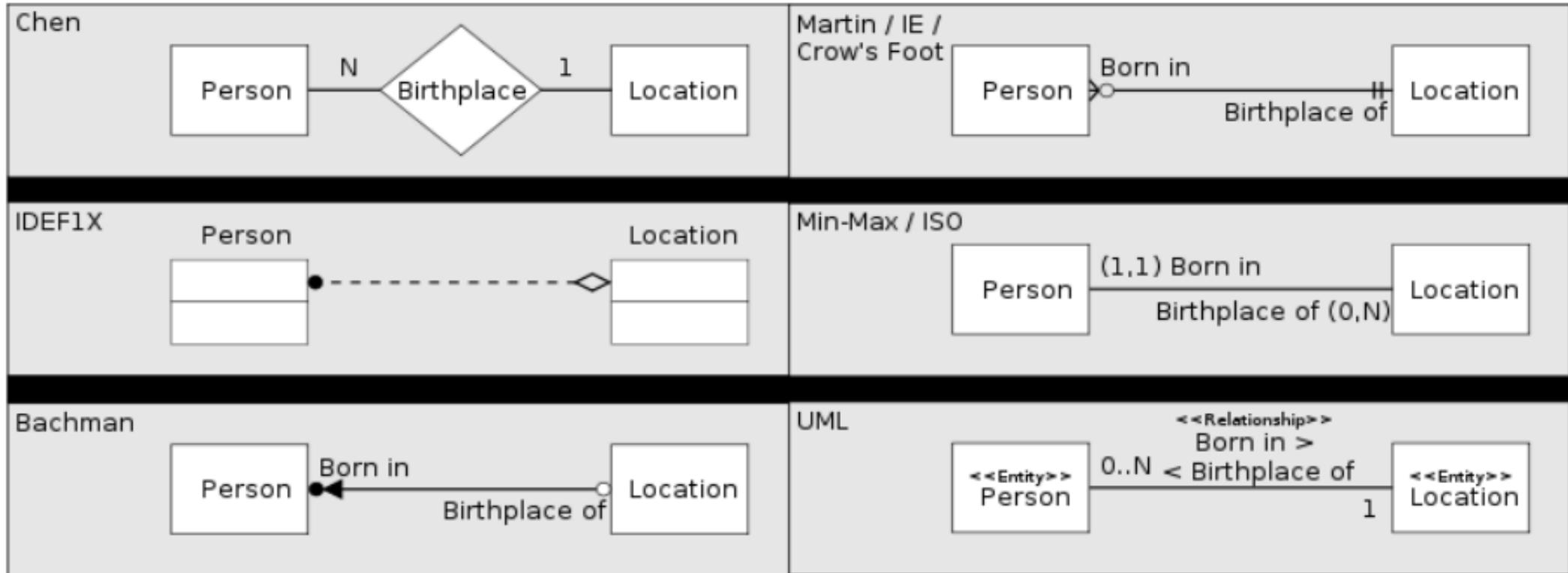
- Взаимоотношения сущностей выражаются связями (**Relationships**).
- Рассмотрим классификацию связей. В зависимости от того, сколько экземпляров сущности одного класса связаны со сколькими экземплярами сущности другого класса, различают следующие типы связей: [?]
- **Связь 1:1.** Один экземпляр сущности связан с одним экземпляром другой сущности. Примером является связь «соответствует» между классами сущностей **ФАКУЛЬТЕТ** и **РАСПИСАНИЕ ЭКЗАМЕНОВ НА ФАКУЛЬТЕТ** (каждому факультету соответствует свое расписание). [?]
- **Связь 1:M.** Единый экземпляр сущности связан со многими экземплярами другой сущности. Примером является связь «зачисление» между классами сущностей **ФАКУЛЬТЕТ** и **АБИТУРИЕНТ** (на один факультет зачисляется много абитуриентов). [?]
- **Связь M:N.** Несколько экземпляров сущности связаны с несколькими экземплярами другой сущности. Примером является связь «сдают» между классами сущностей **АБИТУРИЕНТ** и **ЭКЗАМЕН** (каждый абитуриент сдает несколько экзаменов, и каждый экзамен сдают много абитуриентов).
- Числа, описывающие типы бинарных связей (1:1, 1:M, M:N), обозначают максимальное количество сущностей на каждой стороне связи. Эти числа называются максимальными кардинальными числами, а соответствующая пара чисел называется **максимальной кардинальностью**.

Различные нотации ERD (типы обозначений)

- Многими авторами были разработаны свои варианты моделей ERD. Но все варианты моделей «сущность-связь» исходят из одной идеи — графическое изображение нагляднее текстового описания. В настоящее время ER-модель стала фактически стандартом для семантической структуризации предметной области — стандартом при концептуальном моделировании баз данных.
- Нотация Чена
- Нотация Бахмана
- Нотация IDEF1X
- Нотация Мартина (Crow's foot)
- Нотация min, max
- Нотация Диаграмма классов UML

Различные нотации ERD (типы обозначений)

- Пример различных способы представления одного и того же отношения один ко многим. В каждом случае на диаграмме показана зависимость между лицом и местом рождения: каждое лицо должно иметь одно место рождения, но в каждом месте может родиться ноль или более человек.



Пример: БД университета

- **Сущности (Entities):**

- Students
- Teachers
- Subjects

- **Атрибуты (Attributes):**

- Students (stud_id, name, email, group)
- Teachers (teach_id, name, email, department)
- Subjects (subject_id, name, credits)

Первое обозначение - нотация Чена

Сущность (прямоугольная форма)



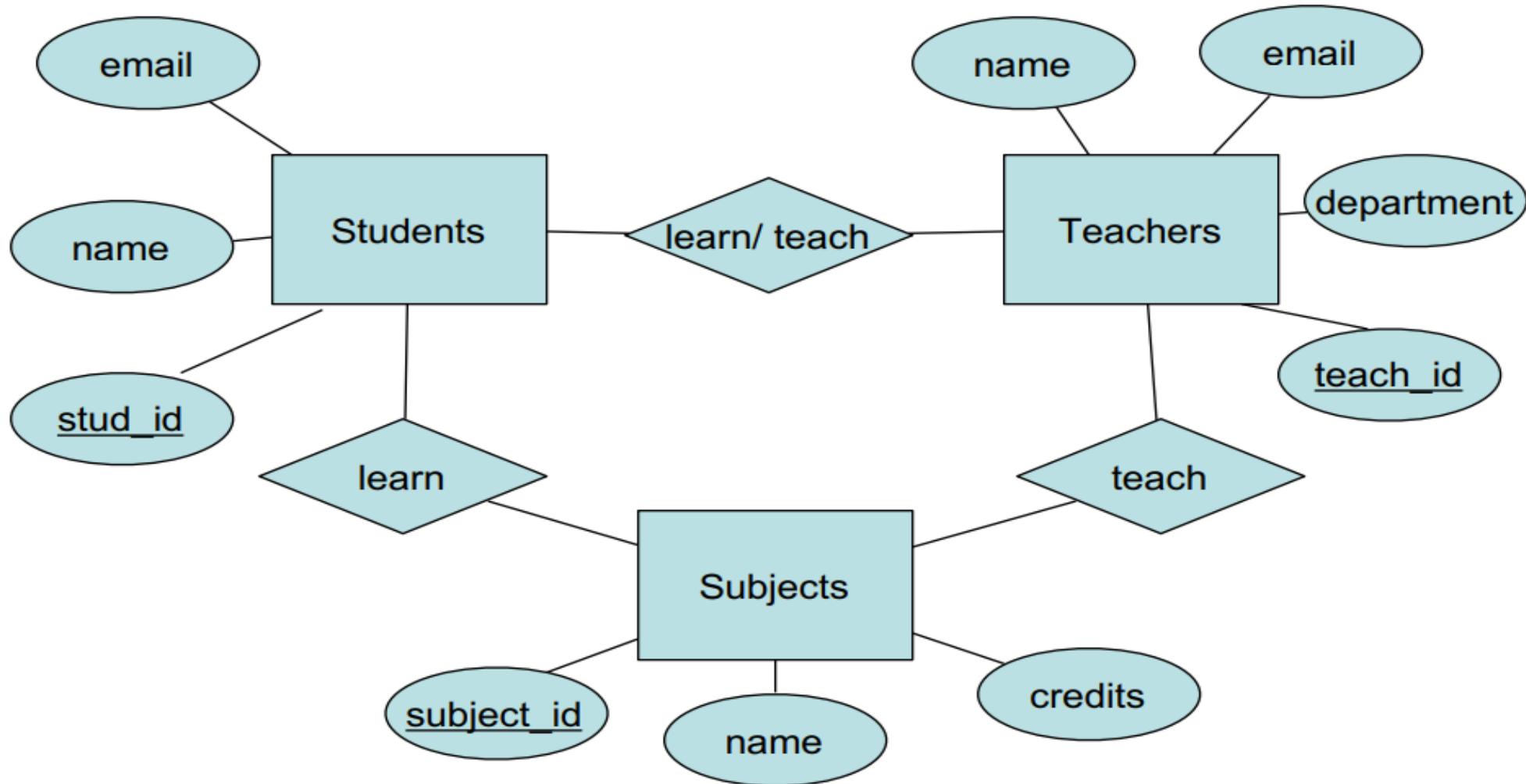
Атрибут (овальная форма)



Отношения (форма ромба)



ER-диаграмма с нотацией Чена



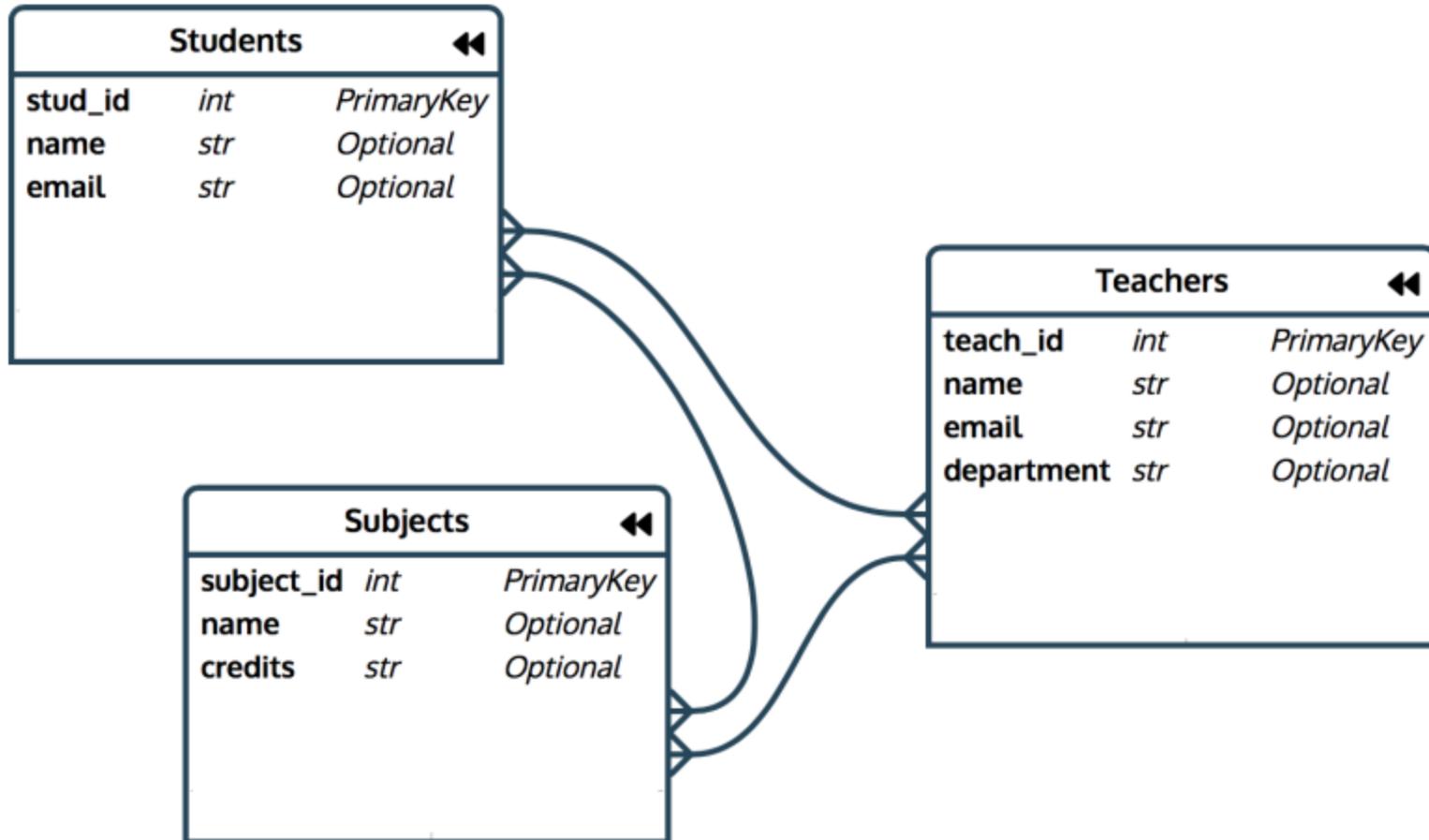
Crow's foot нотация

Crow's foot нотация представляет:

- объекты в виде таблиц (блоков);
- отношения в виде линий между блоками;
- разные фигуры на концах этих линий обозначают относительную кардинальность отношения (тире означает "один", гусиная лапка представляет «много» или «бесконечно»)

	Table_name
PK	<u>attribute_name</u>
	attribute_name

ER-диаграмма в нотации Crow's foot



Внешний ключ

Внешний ключ — это ключ, используемый для связи двух таблиц.

Внешний ключ — это атрибут в одной таблице, который ссылается на первичный ключ в другой таблице.

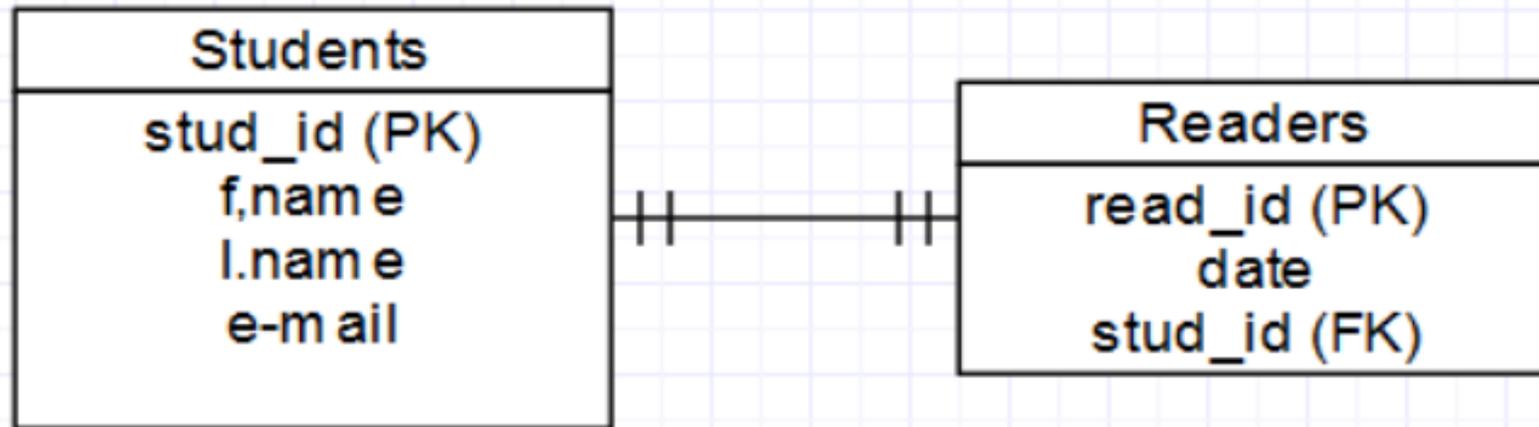
Таблица, содержащая внешний ключ, называется дочерняя таблица и таблица, содержащая первичный ключ называется сылочной или родительской таблицей.

Связь Один к одному

Один к одному - Один экземпляр объекта (А) связанс одним другим экземпляром другого объекта (В)



Пример связи Один к одному

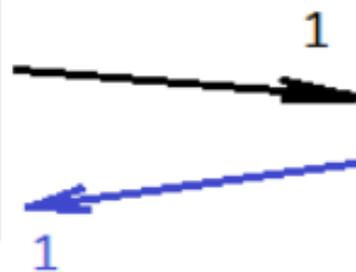


Students

<u>stud_id</u>	f.name	l.name	e-mail
001@gmail.com
002@gmail.com
003@gmail.com

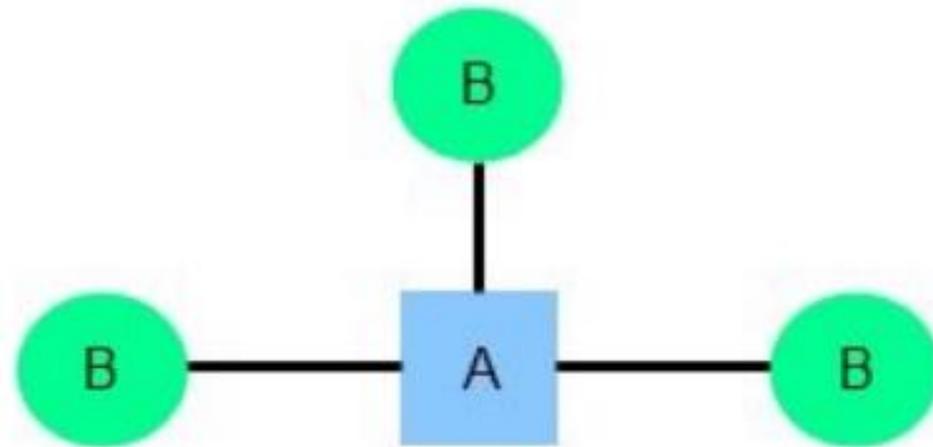
Readers

<u>read_id</u>	date	<u>stud_id</u>
001	31.05.2020	001
002	31.05.2020	003

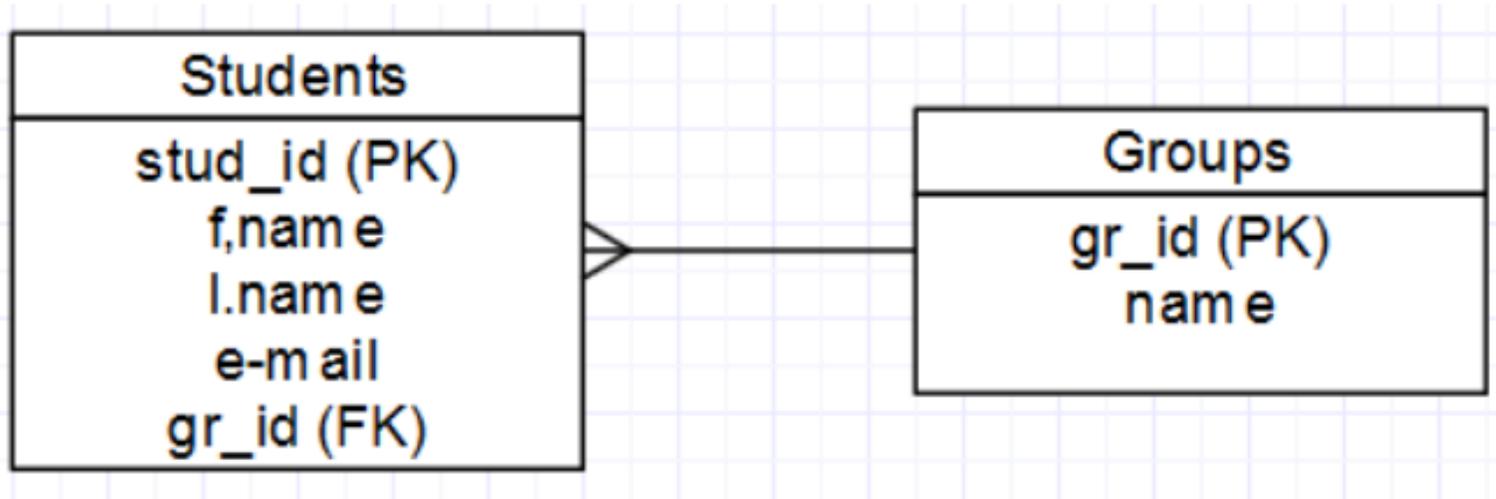


Связь Один ко многим

Один ко многим - Один экземпляр объекта (А) связан с одним или несколькими экземплярами другого объекта (В), но для одного экземпляра объекта В существует только один экземпляр объекта А.



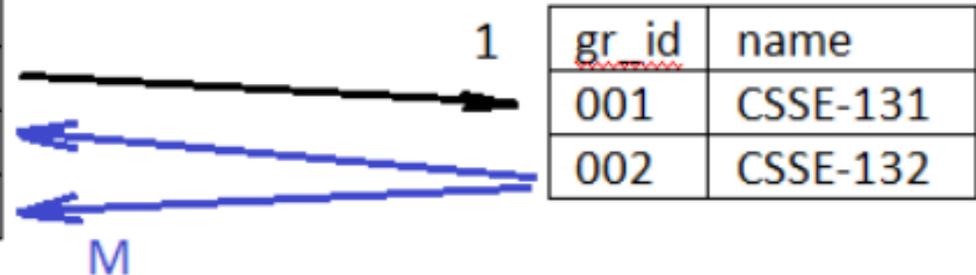
Пример связи Один ко многим



Students

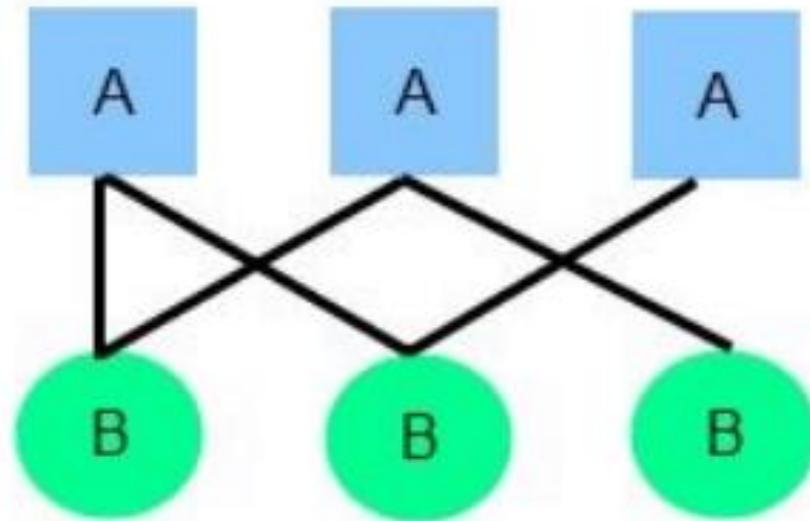
<u>stud_id</u>	f.name	l.name	e-mail	<u>gr_id</u>
001@gmail.com	001
002@gmail.com	002
003@gmail.com	002

Groups

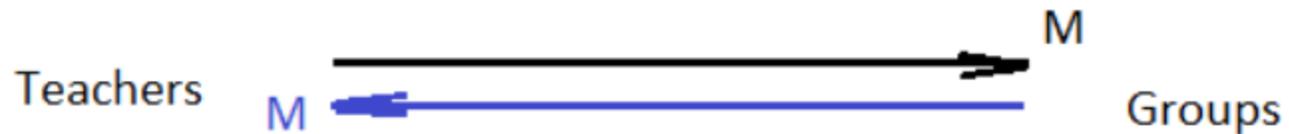
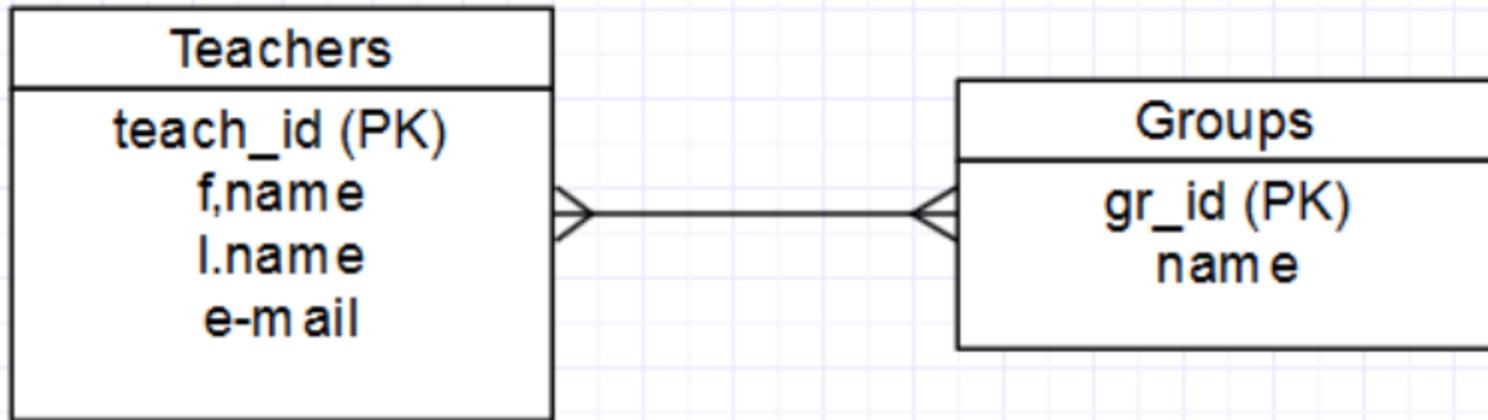


Связь Многие ко многим

Многие ко многим - Один экземпляр объекта (А) связан с одним или несколькими экземплярами другого объекта(В), и один экземпляр объекта В связан с одним или несколькими экземплярами объекта А.



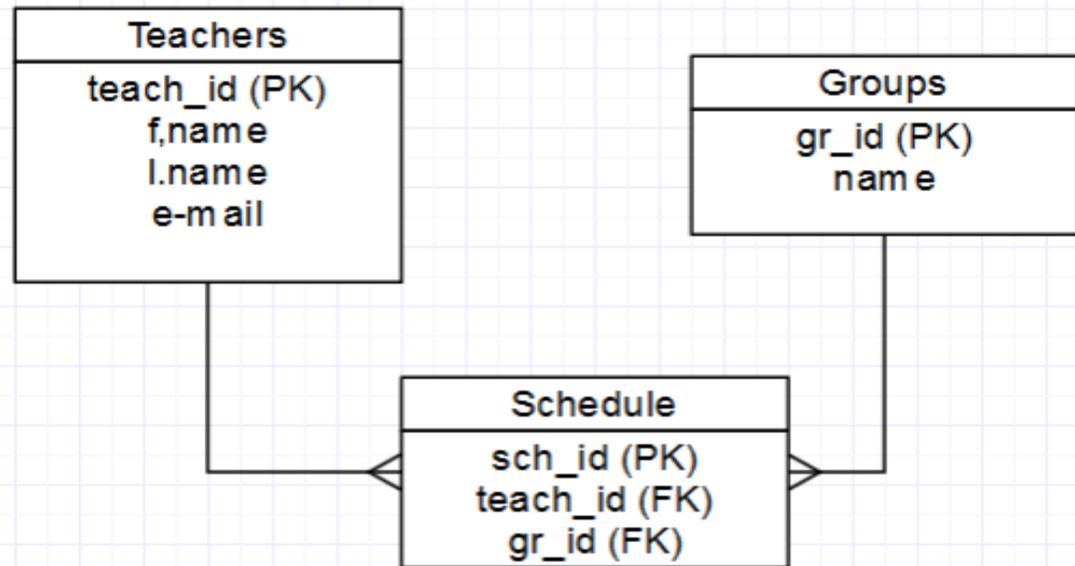
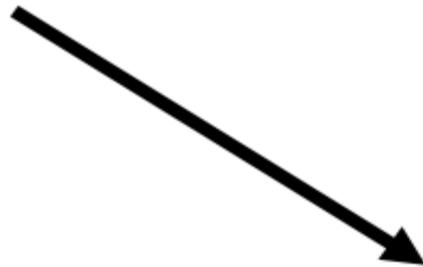
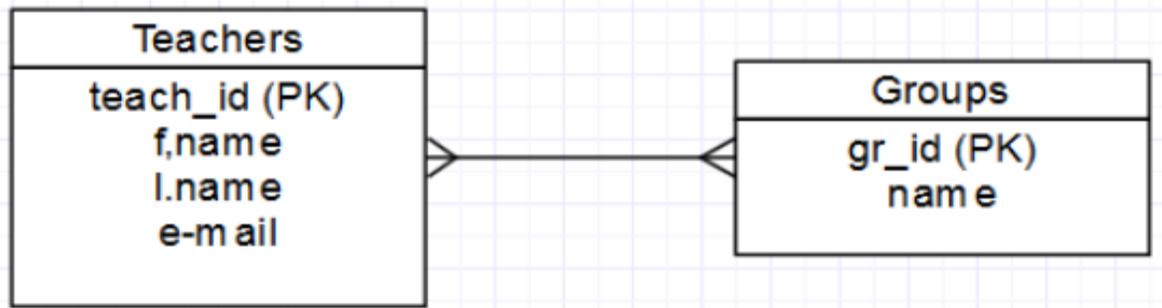
Пример связи Многие ко многим



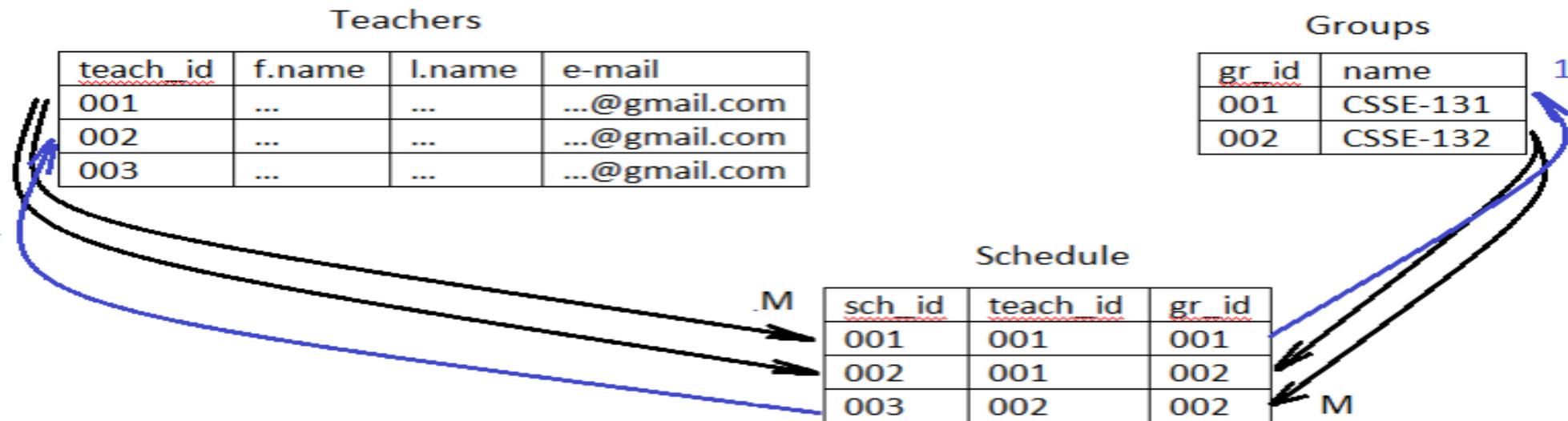
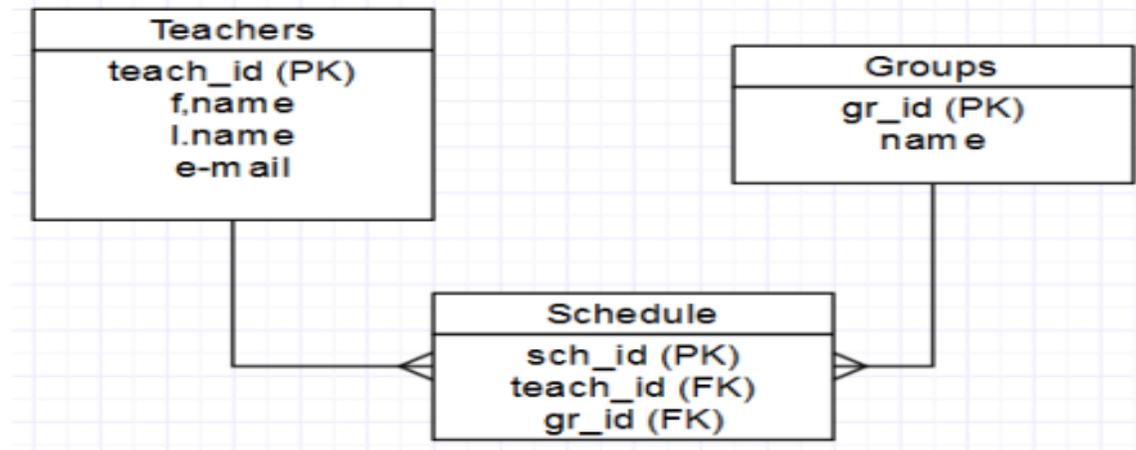
<u>teach_id</u>	f.name	l.name	e-mail
001@gmail.com
002@gmail.com
003@gmail.com

<u>gr_id</u>	name
001	CSSE-131
002	CSSE-132

Пример связи Многие ко многим



Пример связи Многие ко многим



Обозначение отношения в нотации Crow's foot

Символы используются для представления количества элементов:

- кольцо представляет «ноль»
- тире означает «один»
- гусиная лапка представляет «много» или «бесконечность»

Иногда эти символы используются парами. Внутренний компонент обозначения представляет минимум, а внешний компонент представляет максимум.

- кольцо и тире → минимум ноль, максимум один (необязательно)
- тире и тире → минимум один, максимум один (обязательно)
- кольцо и гусиная лапка → минимум ноль, максимум много (необязательно)
- тире и гусиная лапка → минимум один, максимум много(обязательный)

Инструменты

- [Gliffy.com](https://gliffy.com)
- [Lucidchart.com](https://lucidchart.com)
- [Creately.com](https://creately.com)
- [Draw.io](https://draw.io)
- MS Visio
- Erwin
- etc.

Спасибо за внимание!