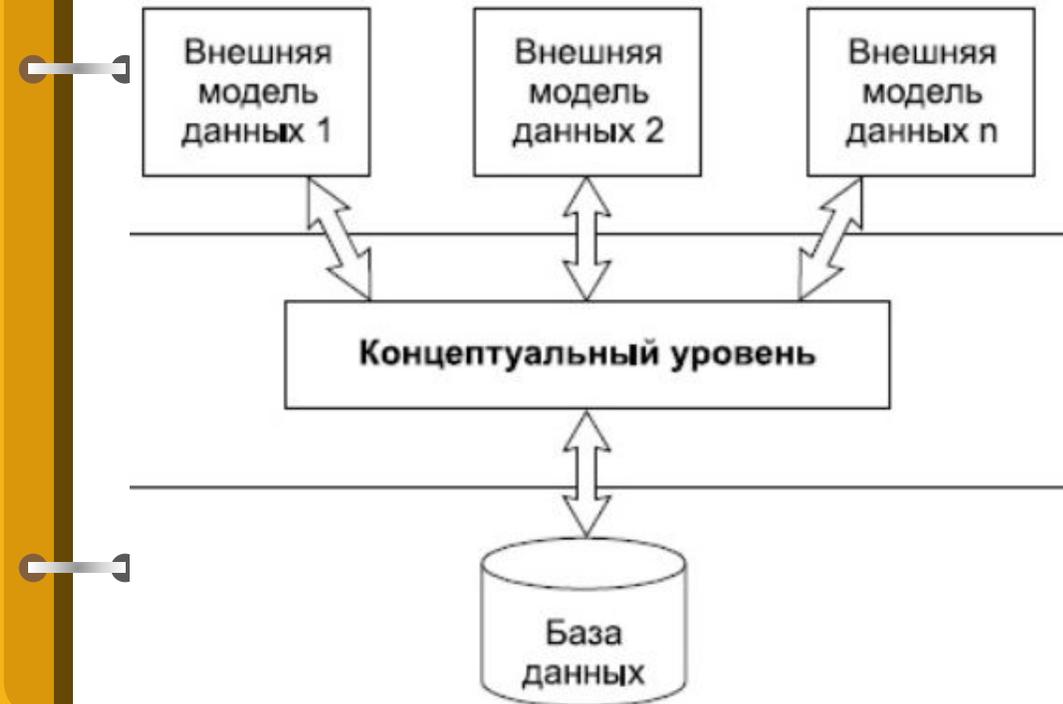


# Основы проектирования

## Баз данных

Тема 2. Взаимосвязи в моделях и  
реляционный подход к построению моделей

Американским комитетом по стандартизации ANSI (American National Standards Institute) была предложена трехуровневая система организации БД.





**Логическая независимость** - возможность изменения одного приложения без корректировки других приложений, работающих с этой же базой данных



**Физическая независимость** - возможность переноса хранимой информации с одних носителей на другие при сохранении работоспособности всех приложений, работающих с данной базой данных

Уровень внешних моделей - уровень на котором определяется видение БД отдельных приложений. Каждое приложение видит и обрабатывает только те данные, которые необходимы именно этому приложению.

Концептуальный уровень — центральное управляющее звено, здесь база данных представлена в наиболее общем виде, который объединяет данные, используемые всеми приложениями, работающими с данной базой данных. Фактически концептуальный уровень отражает обобщенную модель предметной области (объектов реального мира), для которой создавалась база данных. Как любая модель, концептуальная модель отражает только существенные, с точки зрения обработки, особенности объектов реального мира.

Физический уровень — собственно данные, расположенные в файлах или в страничных структурах, расположенных на внешних носителях информации.

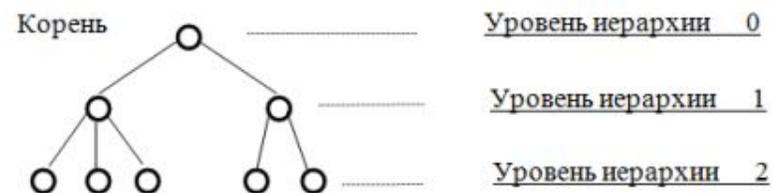
## Типы моделей данных

**Модель данных (МД)** – совокупность правил порождения структур данных в базе данных, операций над ними, а также ограничений целостности, определяющих допустимые связи и значения данных, последовательность их изменения.



Самой первой МД, которая использовалась для построения концептуальных схем была иерархическая модель. Вслед за ней появились сетевые модели. Потом ER-модели, и, как итог развития моделей, возникли реляционные и постреляционные модели. Каждая из перечисленных моделей имеет свои достоинства и недостатки.

**Иерархическая модель данных (ИМД)** – это модель, в которой абстрактные понятия находятся в отношении предшествования таким образом, что каждому понятию соответствует только один предшественник (родитель). Только одна часть, называемая корнем модели, не имеет предшественника.

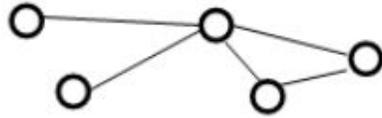


Иерархическая БД состоит из упорядоченного набора деревьев; или, более точно, из упорядоченного набора нескольких экземпляров одного типа

деревья. Примерами типичных операторов манипулирования иерархически организованными данными могут быть следующие:

- найти указанное дерево;
- перейти от одного дерева к другому;
- перейти от одной записи к другой внутри дерева;
- перейти от одной записи к другой в порядке обхода иерархии;
- вставить новую запись в указанную позицию;
- удалить текущую запись.

**Достоинство ИМД:** данные, отражающие общие свойства совокупности конкретных данных не дублируются. **Недостаток ИМД** связан с дублированием данных в случае их однотипности.



Если в иерархических структурах запись-потомок должна иметь ровно одного предка, то в сетевых структурах данных потомок может иметь любое число предков. Связи между записями в **сетевой МД** выполняются в виде указателей, то есть каждая запись хранит ссылку на другую однотипную запись (или признак конца списка) и ссылки на списки подчиненных записей.

**Достоинство сетевых МД:** данные имеют четкую структуру.

**Недостатком сетевых МД** является большое количество дополнительной информации о связях.

**Реляционная модель данных (РМД)** – это модель, в которой данные можно представить в виде отношений, изменяющихся во времени. Традиционно в реляционных системах отношением называют таблицу, кортежем – строку таблицы, а атрибутом – столбец. При этом атрибуты имеют уникальные имена в рамках одного отношения.

**Достоинство реляционной модели** заключается в простоте для понимания, наглядности и удобстве физической реализации на ЭВМ.

**Недостатки реляционной модели данных:** модель не допускает представления объектов со сложной структурой, поскольку в ее рамках возможно моделирование лишь с помощью двумерных таблиц. Данные об объектах содержатся, как правило, во многих таблицах, что значительно замедляет обработку данных

а)  
ГРУППА\_СТАРОСТА

Код группы	ФИО куратора
01	Антонов
02	Фурсов
03	Иванов

ГРУППА

Код группы	ФИО студента
01	Павлова
01	Семенов
02	Терехова
02	Якимова
03	Лунев
03	Мельников
03	Окунев

б)  
ГРУППА

Код группы	ФИО куратора	ФИО студента
01	Антонов	Павлова
		Семенов
02	Фурсов	Терехова
		Якимова
03	Иванов	Лунев
		Мельников
		Окунев

а - реляционная модель;

б - постреляционная модель

**Постреляционная модель данных** в общем случае представляет собой расширенную реляционную модель, снимающую ограничение неделимости значений полей. То есть, допускаются многозначные поля, значения которых состоят из подзначений. Набор значений многозначных полей считается самостоятельной таблицей, встроенной в основную.

**Достоинство постреляционной модели данных:** возможность представления связанных реляционных таблиц одной постреляционной таблицей.

**Недостаток постреляционной модели:** сложность в обеспечении целостности данных.

**Объектно-ориентированная модель данных (ООМД)** представляет структуру, которую можно изобразить графически в виде дерева, узлами которого являются объекты.



Каждый объект характеризуется уникальным идентификатором, состоянием и поведением. Состояние объекта определяется множеством значений его свойств (атрибутов). Поведение объекта описывают методы, называемые процедурами. То есть, составной частью описания объекта являются процедуры, способные производить действия над атрибутами объекта в случае наступления тех или иных событий. Объекты могут объединяться в классы. Экземпляры одного класса отличаются лишь значениями своих свойств, но не своими

**Основным достоинством ООМД** является способность отображать информацию о сложных объектах. Эта модель обычно применяется для сложных предметных областей, при моделировании которых не хватает функциональности реляционной модели. **Недостаток ООМД:** неудобство обработки больших массивов данных.

**Объектно-реляционная модель данных (ОРМД)** является гибридной моделью, сочетающей возможности реляционной модели с объективными свойствами данных. В ОРМД используются такие объектно-ориентированные компоненты, как инкапсуляция, полиморфизм, наследование и т.п. Отличительная особенность объектно-реляционной модели от ООМД состоит в том, что она основана на стратегии реляционной модели.

**Многомерная модель** предназначена для аналитической обработки информации. В данной модели используются такие понятия, как агрегируемость, историчность, прогнозируемость данных. Агрегируемость данных - возможность их рассмотрения с различным уровнем обобщения.

Историчность обеспечивает высокий уровень статичности (неизменяемости) данных и их взаимосвязей, а также, в обязательном порядке, привязку данных к временным точкам. Прогнозируемость данных - задание функций прогнозирования и применение их к различным интервалам времени.

Основные понятия в многомерной модели – **измерение и ячейка**. Измерение многомерной модели – это множество однотипных данных, образующих одну из граней многомерного гиперкуба. Примеры наиболее часто используемых временных измерений – дни, месяцы, кварталы и годы. Ячейка многомерной модели – это поле, значение которого однозначно определяется фиксированным набором измерений. Тип поля чаще всего определен цифрой.

Недостаток этой модели состоит в громоздкости в случае ее использования для решения стандартных задач оперативной обработки.

## Реляционная модель данных

# ОСНОВНЫЕ ПОНЯТИЯ РЕЛЯЦИОННЫХ БАЗ ДАННЫХ

The diagram shows a table with the following structure and labels:

- Заголовок отношения** (Header of the relation) points to the first row of the table.
- Наименование атрибута** (Attribute name) points to the column headers.
- Ключ** (Key) points to the 'Код' column.
- Кортеж** (Tuple) points to a row of data.
- Отношение** (Relation) points to the entire table.
- Значение атрибута** (Attribute value) points to a cell in the table.
- Атрибут** (Attribute) points to a column in the table.

Код	Фамилия	Имя	ДатаРождения	ДатаНайма
1	Белова	Мария	08-дек-1968	01-май-1992
2	Новиков	Павел	19-фев-1952	14-авг-1992
3	Бабкина	Ольга	30-авг-1963	01-апр-1992
4	Воронова	Дарья	19-сен-1958	03-май-1993
5	Кротов	Андрей	04-мар-1955	17-окт-1993
6	Акбаев	Иван	02-июл-1963	17-окт-1993
7	Кралев	Петр	29-май-1960	02-январ-1994
8	Крылова	Анна	09-январ-1958	05-мар-1994
9	Ясенева	Инна	02-июл-1969	15-ноя-1994

# ОСНОВНЫЕ ПОНЯТИЯ РЕЛЯЦИОННЫХ БАЗ ДАННЫХ

тип данных

домен

атрибут

кортеж

первичный ключ

отношение

# ДОМЕН

- Домен – это семантическое понятие. Домен можно рассматривать как подмножество значений некоторого типа данных имеющих определенный смысл. Домен характеризуется следующими свойствами:
- домен имеет уникальное имя (в пределах базы данных);
- домен определен на некотором простом типе данных или на другом домене;
- домен может иметь некоторое логическое условие, позволяющее описать подмножество данных, допустимых для данного домена;
- домен несет определенную смысловую нагрузку.

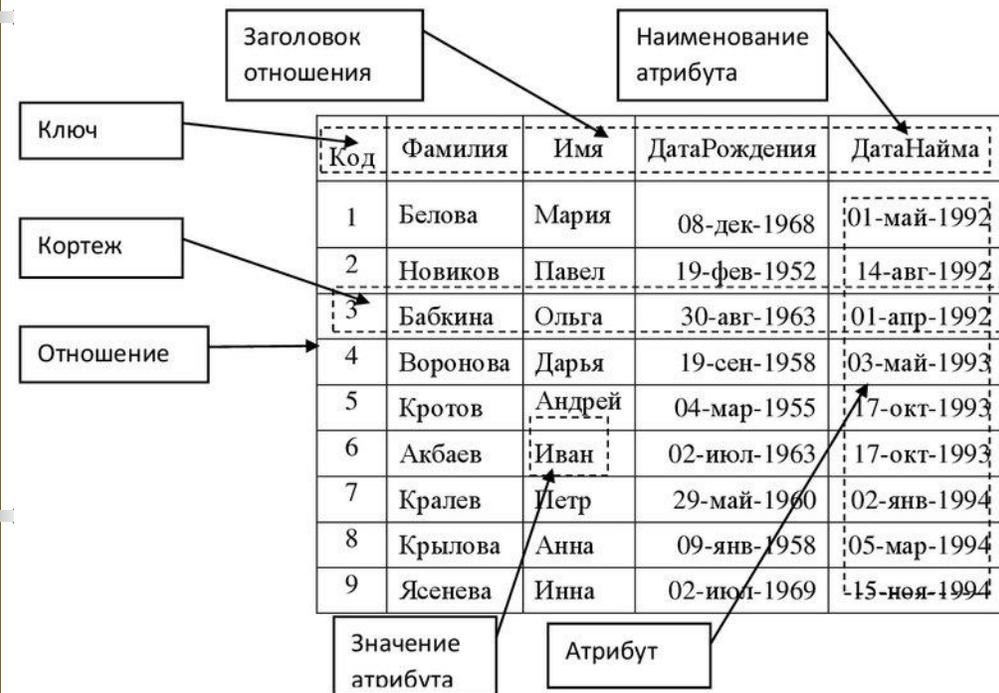
$$D = \{n \in N : (n \geq 18 \text{ and } n \leq 60)\}$$

# КОРТЕЖ

Кортеж отношения представляет собой множество пар вида **<Имя\_атрибута : Значение\_атрибута>**:

(**<A<sub>1</sub> : Val<sub>1</sub>>**, **<A<sub>2</sub> : Val<sub>2</sub>>**, ..., **<A<sub>n</sub> : Val<sub>n</sub>>**), таких, что значение Val<sub>i</sub> атрибута A<sub>i</sub> принадлежит домену D<sub>i</sub>

# ОСНОВНЫЕ ПОНЯТИЯ РЕЛЯЦИОННЫХ БАЗ ДАННЫХ



## ТИП ДАННЫХ

**Тип данных** — множество значений и операций на этих значениях (IEEE Std 1320.2-1998)

- простые типы данных;
- структурированные типы данных;
- ссылочные типы данных.

## ОТНОШЕНИЕ

- **Отношение  $R$** , определенное на множестве доменов  $D_1, D_2, \dots, D_n$  (не обязательно различных), содержит две части: **заголовок** и **тело**.
- **Заголовок отношения** содержит фиксированное количество атрибутов отношения:  $\langle A_1 : D_1 \rangle, \langle A_2 : D_2 \rangle, \dots, \langle A_n : D_n \rangle$ .
- **Тело отношения** содержит множество кортежей отношения.
- Отношение обычно записывается в виде:  $R(\langle A_1 : D_1 \rangle, \langle A_2 : D_2 \rangle, \dots, \langle A_n : D_n \rangle)$ , или короче  $R(A_1, A_2, \dots, A_n)$ , или просто  $R$ .

## ПРИМЕР ОТНОШЕНИЯ

- СОТРУДНИКИ(Код сотрудника, Фамилия, Зарплата, Номер отдела)  
{ (1, Иванов, 1000, 1),  
(2, Петров, 2000, 2),  
(3, Сидоров, 3000, 1) }

Код сотрудника	Фамилия	Зарплата	Номер отдела
1	Иванов	1000	1
2	Петров	2000	2
3	Сидоров	3000	1

## АТТРИБУТ

- **Атрибут отношения  $A$**  есть пара вида
- $\langle \text{Имя\_атрибута} : \text{Имя\_домена} \rangle$

## СООТВЕТСТВИЕ РЕЛЯЦИОННЫХ И «ТАБЛИЧНЫХ» ТЕРМИНОВ

Реляционный термин	Соответствующий «табличный» термин
База данных	Набор таблиц
Схема базы данных	Набор заголовков таблиц
Отношение	Таблица
Заголовок отношения	Заголовок таблицы
Тело отношения	Тело таблицы
Атрибут отношения	Наименование столбца таблицы
Кортеж отношения	Строка таблицы
Степень (-арность) отношения	Количество столбцов таблицы
Мощность отношения	Количество строк таблицы
Домены и типы данных	Типы данные в ячейках таблицы
Первичный ключ	Одно или несколько наименований столбцов таблицы

## ФУНДАМЕНТАЛЬНЫЕ СВОЙСТВА ОТНОШЕНИЙ

В отношении нет одинаковых кортежей

Кортежи не упорядочены (сверху вниз)

Атрибуты не упорядочены (слева направо)

Все значения атрибутов содержат только одно значение из соответствующего домена

```
СТУДЕНТ(< NumRecordBook:CHAR(7)>, <LastName: NVARCHAR(30)> ,
      <FistName : NVARCHAR(30)> , <Patronymic: NVARCHAR(30)> )
```

```
CREATE TABLE Studet
(
  NumRecordBook CHAR(7) PRIMARY KEY,
  LastName NVARCHAR(30) NOT NULL,
  FistName NVARCHAR(30) NULL,
  Patronymic NVARCHAR(30) NULL
);
```

NumRecordBook	LastName	FistName	Patronymic
1234567	Игнатов	Никита	Александрович
2345678	Джумаев	Ровшен	

## Реляционная алгебра

Основы реляционной алгебры были заложены Э. Коддом. Реляционная алгебра задает набор операторов для выполнения операций над реляционными отношениями. Операции реляционной алгебры Кодда можно разделить на две группы: базовые теоретико-множественные и специальные реляционные.

Объединение (**UNION**) двух совместимых отношений R1 и R2 одинаковой размерности – это отношение R, содержащее все элементы исходных отношений. Повторяющиеся кортежи не включаются.



Пересечение (**INTERSECT**) двух совместимых по типу отношений R1 и R2 одинаковой размерности – это отношение R с телом, включающим в себя кортежи, одновременно принадлежащие обоим отношениям



Вычитание (**MINUS**) двух совместимых по типу отношений R1 и R2 одинаковой размерности – это отношение R с телом, состоящим из множества всех кортежей, принадлежащих отношению R1 и не принадлежащих отношению R2.



Декартово произведение отношений (**TIMES**) – это бинарный оператор, предназначенный для комбинирования двух отношений. Пусть R1- отношение степени k1 , а R2- отношение степени k2. R = R1 x R2 будет представлять собой множество кортежей таких, что первые k1 элементов кортежа отношения R составляют кортежи отношения R1, а последующие k2 элементов кортежа из R составляются кортежами из R2. Сочетание кортежей из R1 и R2 при образовании кортежа R осуществляется по правилам декартова произведения. Здесь R1 и R2 могут иметь разные схемы. Степень отношения R равна сумме степеней отношений операндов R1 и R2. Мощность отношения R

R1		R2	
Код студента	Фамилия	Код дисциплины	Название
С3	Титов	Д1	Информатика
С4	Павлов	Д2	Базы данных
		Д3	Физика

**R (R1xR2)**

Код студента	Фамилия	Код дисциплины	Название
С3	Титов	Д1	Информатика
С3	Титов	Д2	Базы данных
С3	Титов	Д3	Физика
С4	Павлов	Д1	Информатика
С4	Павлов	Д2	Базы данных
С4	Павлов	Д3	Физика

## Специальные реляционные операторы

Выборка (**R WHERE f**) отношения R по формуле f представляет собой новое отношение, заголовок которого совпадает с заголовком отношения R, а тело содержит множество кортежей, являющихся подмножеством множества кортежей отношения R, для которых формула f принимает значение истина. Здесь f – логическая формула (предикат), принимающая значение «истина» или «ложь», образованная операндами, представляющими собой имена атрибутов отношений R или литералы, а также скалярными операторами сравнения (=, <>, >, <, >=, <=) и

логические операторы AND, OR, NOT.  
 1) R WHERE Курс = 1; /нарная  
 2) R WHERE Факультет = »Информатики» AND Курс = 3.

**R**

Код студента	Фамилия	Факультет	Курс
001017	Васильев	Информатики	1
002016	Перов	Математики	2
009017	Титов	Электроники	1
087015	Павлов	Информатики	3
005017	Афанасьев	Математики	1

**R WHERE Курс = 1**

Код студента	Фамилия	Факультет	Курс
001017	Васильев	Информатики	1
009017	Титов	Электроники	1
005017	Афанасьев	Математики	1

**R WHERE Факультет = "Информатики" AND Курс = 3**

Код студента	Фамилия	Факультет	Курс
087015	Павлов	Информатики	3

Проекция отношения R по атрибутам X, Y, ..., Z обозначается R [X, Y, ..., Z], где каждый из указанных атрибутов принадлежит отношению R, – это отношение с заголовком { X, Y, ..., Z } и телом, содержащим кортежи отношения R, за исключением повторяющихся кортежей. Повторение одинаковых атрибутов в списке X, Y, ..., Z запрещается.

С помощью оператора проекции получается «вертикальное» подмножество атрибутов отношения R с последующим исключением дублирующих кортежей (если они возникают).

R[Фамилия, Факультет]

Фамилия	Факультет
Васильев	Информатики
Перов	Математики
Титов	Электроники
Павлов	Информатики
Афанасьев	Математики

Соединение – операция реляционной алгебры, связывающая таблицы. Операция соединения используется для связывания данных между таблицами. Это, возможно, наиболее важная функция любого языка баз данных. ТЭТА–соединение отношений. Пусть отношения R1 и R2 не имеют атрибутов с одинаковыми именами, а символ  $\Theta$  (греческая буква тэта) обозначает один из операторов сравнения: =, <>, >, <, <=, >=. Тогда  $\Theta$ –соединением отношения R1 по атрибуту X с отношением R2 по атрибуту Y является отношение, кортежи которого получаются в результате декартова произведения отношений R1 и R2 и для которых условие  $X \Theta Y$

R1 (Пилоты)

Пилот	Статус_П (X)
Иванов	4
Федоров	3
Игнатьев	2
Осипов	1

R2 (Самолеты)

Самолет	Статус_С (Y)
Boeing 737-100	3
ТУ-204	2
Ан-140	1
А380	4

В результате получим отношение R:

R

Пилот	Статус_П	Самолет	Статус_С
Иванов	4	Boeing 737-100	3
Иванов	4	ТУ-204	2
Иванов	4	Ан-140	1
Иванов	4	А380	4
Федоров	3	Boeing 737-100	3
Федоров	3	ТУ-204	2
Федоров	3	Ан-140	1
Игнатьев	2	ТУ-204	2
Игнатьев	2	Ан-140	1
Осипов	1	Ан-140	1

R1  
TIMES  
R2)  
WHERE  
 $X \geq Y$

Частным случаем соединения являются **эквисоединение и естественное соединение**. Операция эквисоединения характеризуется тем, что оператор сравнения в этом соединении – это знак равенства. Операция естественного соединения (операция  $R1 \text{ JOIN } R2$ ) **применяется к двум отношениям, имеющим общий атрибут** (простой или составной). Этот атрибут в отношениях имеет одно и то же имя (совокупность имен) и определен в одном и том

Деление отношения. Пусть  $R1$  и  $R2$  – два отношения. Тогда отношение  $R := R1/R2$  может быть получено следующим образом: 1. Результирующее отношение  $R$  будет содержать только те атрибуты делимого  $R1$ , которых нет в делителе  $R2$ . 2. В результирующее отношение  $R$  включаются только те недублирующиеся кортежи из  $R1$ , декартово произведение которых с делителем  $R2$  содержится в делимом (является

ПОДМНОЖ

R1		R2		R (R1 DIVIDE BY R2)	
Студент	Дисциплина	Дисциплина		Студент	
Иванов	Информатика	Информатика		Иванов	
Иванов	Физика	Физика		Зубков	
Иванов	Математика				
Иванов	Электротехника				
Иванов	История				
Федоров	Информатика				
Федоров	История				
Федоров	Математика				
Орлов	История				
Орлов	Математика				
Орлов	Физика				
Орлов	БД				
Зубков	Информатика				
Зубков	Математика				
Зубков	Физика				

Существует 3 вида взаимосвязей : ассоциация, обобщение и агрегация.

Ассоциация — обозначает наличие логической связи между объектами. С каждой ассоциацией связано понятие мощности, которое может принимать одно из следующих значений: 1:1, 1:M и M:N. Мощность обозначает количество объектов определенного типа, которые будут участвовать в связи.

## Типы связей в моделях реляционного типа

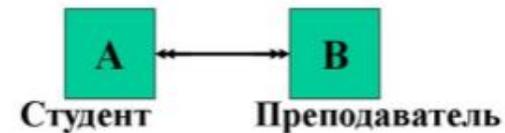
1 : 1



1 : M



M : M



**связь «один к одному»** устанавливается в случаях, когда конкретная строка главной таблицы в любой момент времени связана только с одной строкой подчиненной таблицы;

**связь «один ко многим»** устанавливается в случаях, когда конкретная строка главной таблицы в любой момент времени связана с несколькими строками подчиненной таблицы; при этом любая строка подчиненной таблицы связана только с одной строкой главной таблицы;

**связь «многие ко многим»** устанавливается в случаях, когда конкретная строка главной таблицы в любой момент времени связана с несколькими строками подчиненной таблицы и в то же время одна строка подчиненной таблицы связана с несколькими строками.

### Типы связей между таблицами

**Связь 1:1** – с одной записью в первой таблице связана ровно одна запись во второй таблице.

Сотрудники				Секретно	
Код	Фамилия	Имя	Отчество	Код	Зарплата
1	Иванов	Петр	Сидорович	1	20 000 р.
2	Петров	Сидор	Иванович	2	30 000 р.
3	Сидоров	Иван	Петрович	3	40 000 р.

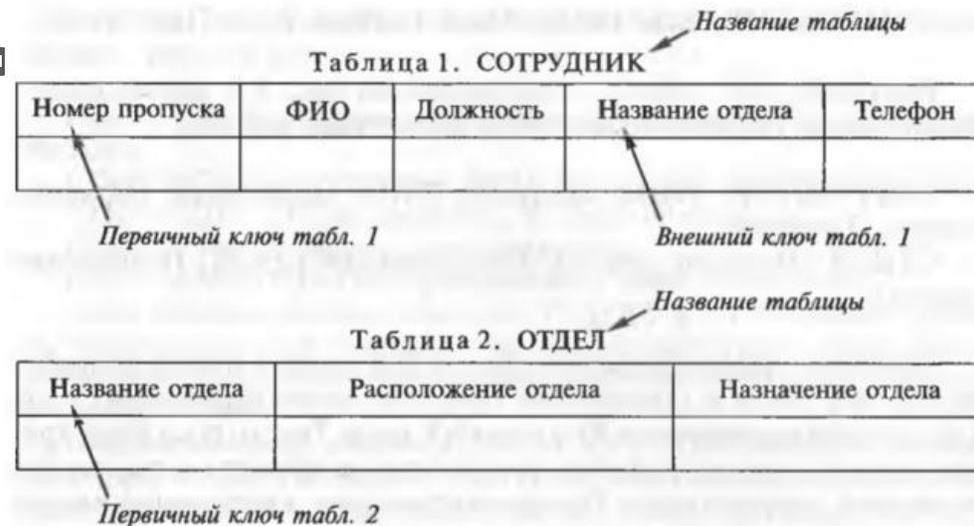
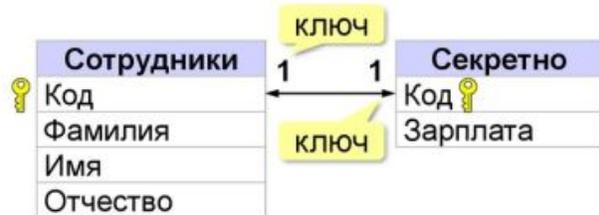


Рис. 1.4. Организация ссылки от одной таблицы к другой

### Иерархическая модель данных



Пользователь выбирает комплектующие для компьютера и база данных выдаёт производителя, стоимость и адреса магазинов.

