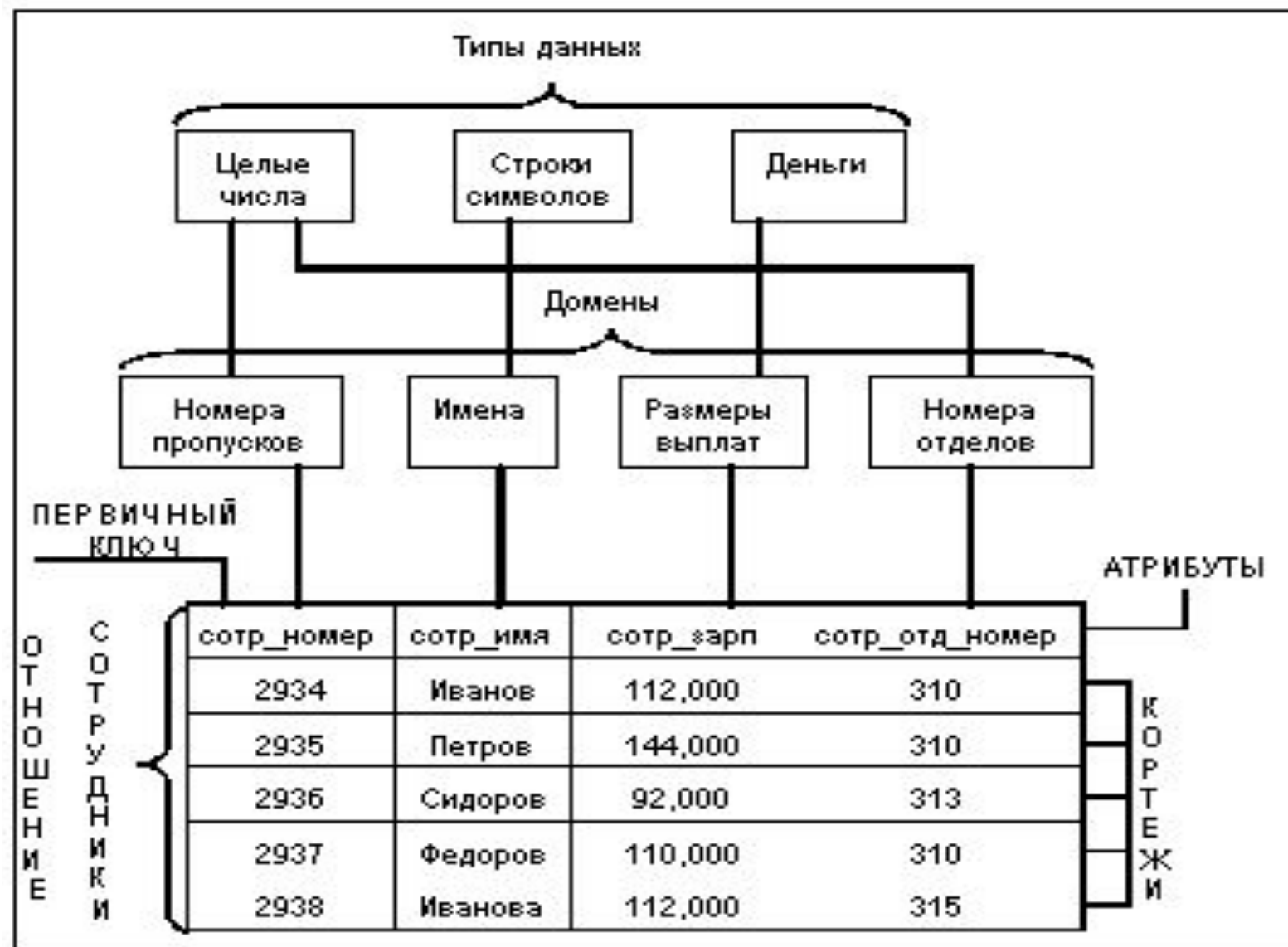


Базовые понятия реляционных баз данных

Пчелинцева Е.Г.

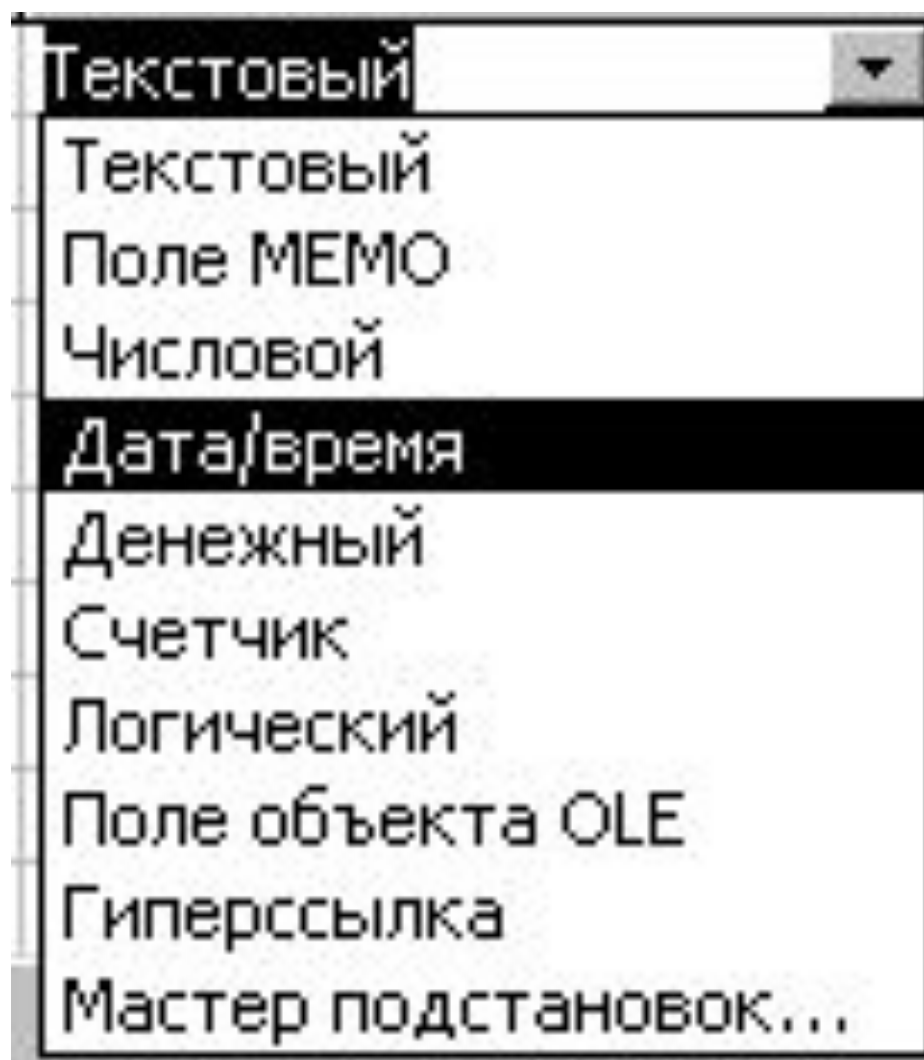
Основными понятиями реляционных баз данных являются тип данных, домен, атрибут, кортеж, первичный ключ и отношение.

Для начала покажем смысл этих понятий на примере отношения СОТРУДНИКИ, содержащего информацию о сотрудниках некоторой организации:



Понятие *тип данных* в реляционной модели данных полностью адекватно понятию типа данных в языках программирования.

Обычно в современных реляционных БД допускается хранение символьных, числовых данных, битовых строк, специализированных числовых данных (таких как "деньги"), а также специальных "темпоральных" данных (дата, время, временной интервал). Достаточно активно развивается подход к расширению возможностей реляционных систем абстрактными типами данных (соответствующими возможностями обладают, например, системы семейства Ingres/Postgres). В нашем примере мы имеем дело с данными трех типов: строки символов, целые числа и "деньги".



Типы данных в Access

Тип данных	Описание
<i>Текстовый</i>	Алфавитно-цифровые данные, длина до 255 символов.
<i>Мемо</i>	Алфавитно-цифровые данные – предложения, абзацы, тексты. Длина до 64 000 байтов.
<i>Числовой</i>	Числовые данные. 1, 2, 4 или 8 байтов.
<i>Дата/Время</i>	Дата и время. Имеется возможность выбора формата отображения на экране и в документах.
<i>Денежный</i>	Данные о денежных суммах. Хранится как число с 4 знаками после запятой.
<i>Счетчик</i>	Уникальное длинное целое, автоматически генерируется Access при создании каждой новой записи. 4 байта. Используется в качестве ключевого поля.
<i>Логический</i>	Логические данные. Длина 1 бит.
<i>Объект OLE</i>	Картинки, диаграммы и другие объекты OLE из приложений Windows. объем до 1 гигабайта.
<i>Мастер подстановок</i>	Это не тип данных. При выборе этой строки в списке типов вызывается мастер подстановок позволяющий задать подстановку для значений описываемого поля. Подробнее о подстановках говорится в следующем разделе.

Таблица1 : таблица

Имя поля	Тип данных	Описание
ДатаРождения	Текстовый	

Свойства поля

Общие

Подстановка

Размер поля	50
Формат поля	
Маска ввода	
Подпись	
Значение по умолчанию	
Условие на значение	
Сообщение об ошибке	
Обязательное поле	Нет
Пустые строки	Нет
Индексированное поле	Да (Допускаются совпадения)
Сжатие Юникод	Да

Необязательный параметр.
 Выводится в строку состояния
 при выборе поля в форме. Для
 справки по описанию поля
 нажмите клавишу F1.

Свойства поля

Access по умолчанию назначает свойства полям создаваемой таблицы. Набор свойств поля зависит от типа данных поля. При желании (или необходимости) разработчик может изменить эти назначения. Для этого служит нижняя часть окна конструктора таблиц. Здесь находится две вкладки – **Общие** и **Подстановка**.

Вкладка **Общие**.

- **Размер поля.** Это свойство задает максимальный размер данных в описываемом поле. Данные текстового типа могут иметь длину от 1 до 255 символов; по умолчанию устанавливается 50 символов. Не изменяйте это значение, если хранимые строки имеют длину не более 50 символов.
- Размеры данных других типов можно посмотреть в справке Access.
- **Формат поля.** Это свойство позволяет задать формат представления данных при выводе на экран или печать. Для всех типов данных, кроме текстового, Мемо и OLE в Access имеются стандартные наборы форматов. Пользователь имеет возможность выбрать желаемый формат из раскрывающегося списка.

- **Маска ввода.** Для типов данных *Текстовый*, *Числовой*, *Денежный* и *Дата/Время* можно задать маску (шаблон) ввода, которую пользователь увидит при вводе данных в это поле. Маска задается путем записи в позиции вводимой строки специальных символов. Символы, используемые в маске ввода, приведены в таблице 4.2. Термин “может быть” в графе **Назначение** означает не обязательный символ. Термин “должна быть” означает обязательный символ; если не ввести никакого символа, то Access вставит что-то по умолчанию или выдаст сообщение об ошибке.

Символы, используемые в маске ввода

Символ	Назначение
9	В данную позицию может быть введена цифра или пробел.
0	В данную позицию должна быть введена только цифра.
#	В данную позицию должна быть введена цифра, пробел, + или -.
L	В данную позицию должна быть введена произвольная буква.
?	В данную позицию может быть введена буква.
A	В данную позицию должна быть введена буква или цифра.
a	В данную позицию может быть введена буква или цифра.
&	В данную позицию должен быть введен произвольный символ или пробел.
C	В данную позицию может быть введен любой символ или пробел.
.	Место для десятичной точки (запятой).
,	Разделитель тысяч
: ; - /	Разделители, используемые при записи даты и времени.

- **Подпись поля.** Это более описательное имя поля, которое Access будет использовать при выводе данных на экран. Если подпись поля не задана, то Access для таких целей будет использовать **Имя Поля**.
- **Значение по умолчанию.** Можно определить значение по умолчанию для всех типов данных, кроме *Счетчика*, *Мемо* и *Объекта OLE*. Например, для даты и времени можно в качестве такого значения можно указать системные дату и время.
- **Условие на значение.** Можно задать выражение, которое при вводе или редактировании должно быть всегда истинным. Например, <100 означает, что значение этого поля должно быть меньше 100. В качестве условия можно ввести список допустимых значений или записать довольно сложные выражения, в том числе и такие в которых используются значения как вводимого поля, так и других полей таблицы.

- **Сообщение об ошибке.** Можно ввести текст, который Access будет выводить на экран, когда вводимое значение не удовлетворяет условию на значение.
- **Обязательное поле.** Если установлено это свойство, то Access не позволит при вводе оставить это поле пустым.
- **Пустые строки.** Для текстовых и Мемо полей можно разрешить ввод пустых строк.
- **Индексированное поле.** Для некоторых типов полей, чтобы ускорить доступ к данным, можно задать построение индекса. Можно также указать, чтобы значения в индексированном поле не повторялись.

- **Ключевое поле** – это поле, значения которого однозначно определяют (идентифицируют) запись в таблице. Например, шифр зачетной книжки студента однозначно определяет студента в списке студентов.

В качестве ключа может выступать не одно поле, а совокупность нескольких полей.

- Ключевых полей может быть несколько. Тогда в качестве первичного ключа следует выбрать какое-то одно.

Домен

- Понятие *домена* более специфично для баз данных, хотя и имеет некоторые аналогии с подтипами в некоторых языках программирования. В самом общем виде домен определяется заданием некоторого базового типа данных, к которому относятся элементы домена, и произвольного логического выражения, применяемого к элементу типа данных. Если вычисление этого логического выражения дает результат "истина", то элемент данных является элементом домена.
- Наиболее правильной интуитивной трактовкой понятия домена является понимание домена как допустимого потенциального множества значений данного типа. Например, домен "Имена" в нашем примере определен на базовом типе строк символов, но в число его значений могут входить только те строки, которые могут изображать имя (в частности, такие строки не могут начинаться с

Схема отношения, схема базы данных

- Схема отношения - это именованное множество пар {имя атрибута, имя домена (или типа, если понятие домена не поддерживается)}. Степень или "арность" схемы отношения - мощность этого множества. Степень отношения СОТРУДНИКИ равна четырем, то есть оно является 4-арным. Если все атрибуты одного отношения определены на разных доменах, осмысленно использовать для именованния атрибутов имена соответствующих доменов (не забывая, конечно, о том, что это является всего лишь удобным способом именованния и не устраняет различия между понятиями домена и атрибута).
- Схема БД (в структурном смысле) - это набор именованных схем отношений

Кортеж, отношение

Кортеж, соответствующий данной схеме отношения, - это множество пар {имя атрибута, значение}, которое содержит одно вхождение каждого имени атрибута, принадлежащего схеме отношения. "Значение" является допустимым значением домена данного атрибута (или типа данных, если понятие домена не поддерживается). Тем самым, степень или "арность" кортежа, т.е. число элементов в нем, совпадает с "арностью" соответствующей схемы отношения. Попросту говоря, кортеж - это набор именованных значений заданного типа.

Отношение - это множество кортежей, соответствующих одной схеме отношения. Иногда, чтобы не путаться, говорят "отношение-схема" и "отношение-экземпляр", иногда схему отношения называют заголовком отношения, а отношение как набор кортежей - телом отношения. На самом деле, понятие схемы отношения ближе всего к понятию структурного типа данных в языках программирования. Было бы вполне логично разрешать отдельно определять схему отношения, а затем одно или несколько отношений с данной схемой.

- Однако в реляционных базах данных это не принято. Имя схемы отношения в таких базах данных всегда совпадает с именем соответствующего отношения-экземпляра. В классических реляционных базах данных после определения схемы базы данных изменяются только отношения-экземпляры. В них могут появляться новые и удаляться или модифицироваться существующие кортежи. Однако во многих реализациях допускается и изменение схемы базы данных: определение новых и изменение существующих схем отношения. Это принято называть **эволюцией схемы базы данных**.

Обычным житейским представлением отношения является таблица, заголовком которой является схема отношения, а строками - кортежи отношения-экземпляра; в этом случае имена атрибутов именуют столбцы этой таблицы. Поэтому иногда говорят "столбец таблицы", имея в виду "атрибут отношения".

Реляционная база данных - это набор отношений, имена которых совпадают с именами схем отношений в схеме БД.

Благодарю за внимание!