**5. Реляционная модель данных. Базовые понятия. Фундаментальные свойства**

**Отношений**

Согласно Дейту, реляционная модель состоит из трех частей:

-Структурной части.

-Целостной части.

-Манипуляционной части.

***Структурная часть*** описывает, какие объекты рассматриваются реляционной моделью

***Целостная часть*** описывает ограничения специального вида, которые должны выполняться для любых отношений в любых реляционных базах данных. Это ***целостность сущностей*** и ***целостность внешних ключей***.

***Манипуляционная часть*** описывает два эквивалентных способа манипулирования реляционными данными - ***реляционную алгебру*** и ***реляционное исчисление***.

## Базовые понятия реляционных баз данных

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

Т.к. теория БД разрабатывалась на много раньше, нежели сами СУБД, то существует некоторая разница в терминологии. т.е. разными названиями могут обозначаться одни и те же вещи. В таблице 2 приведены термины, используемые в различных ситуациях, но имеющие общий смысл.

|  |  |  |
| --- | --- | --- |
| Теория БД | Практика | SQL Server |
| Отношение (Relation) | Таблица (Table) | Таблица (Table) |
| Кортеж (Tuple) | Запись (Record) | Строка (Row) |
| Атрибут (Attribute) | Поле (Field) | Столбец (Column) |
| Домен (Domain) | Общая совокупность допустимых значений | Количество столбцов |
| Степень отношения | Кардинальное число отношения | Количество строк |

**Тип данных** - адекватно понятию типа данных в языках программирования.

В нашем примере мы имеем дело с данными трех типов: строки символов, целые числа и «деньги».

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. Если вычисление логического выражения относительно элемента данных дает результат "истина", то элемент является принадлежит домену.

Например, домен "ФИО" 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 мягкого знака).

**Отношения** – это именованное множество пар {имя атрибута, имя домена (или типа, если понятие домена не поддерживается)}.

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". Когда мы перейдем к рассмотрению практических вопросов организации реляционных баз данных и средств управления, мы будем использовать эту житейскую терминологию. Этой терминологии придерживаются в большинстве коммерческих реляционных СУБД.

**Степень или "арность" схемы отношения** - мощность этого множества.

Степень отношения СТУДЕНТЫ равна четырем, то есть оно является 4-арным. Если все атрибуты одного отношения определены на разных доменах, осмысленно использовать для именования атрибутов имена соответствующих доменов (не забывая, конечно, о том, что это является всего лишь удобным способом именования и не устраняет различия между понятиями домена и атрибута).

**Кортеж**, соответствующий данной схеме отношения, - это множество пар {имя атрибута, значение}, которое содержит одно вхождение каждого имени атрибута, принадлежащего схеме отношения.

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.

## Фундаментальные свойства отношений

***1. В отношении нет одинаковых кортежей***

Это следует из того факта, что тело отношения это математическое множество (множество кортежей). Множества же по определению не содержат повторяющихся элементов.

***2. Кортежи не упорядочены сверху вниз***

Это следует из того свойства, что тело отношения – это математическое множество. Простые множества являются неупорядоченными. Поэтому если в каком-либо отношении кортежи расположить в другом порядке, отношение все равно останется тем же самым. Следовательно, относительно отношения нельзя сказать, что существует понятие, скажем второго или пятого кортежей.

***3. Атрибуты не упорядочены слева направо***

Поэтому с точки зрения реляционной теории отношение, в котором атрибуты представлены в таком порядке: студ\_билет, студ\_ФИО, студ\_Возраст, студ\_стип и отношение, у которого атрибуты представлены в порядке студ\_билет, студ\_стип, студ\_Возраст, студ\_ФИО, будут одним и тем же отношением. Т.о. не существует первого, второго, четвертого атрибута. Атрибут всегда определяется по имени, а не по расположению. Это как раз и помогает избежать путаницы при написании программ.

***4. Все значения атрибутов отношения атомарные***

Применительно к отображению отношений в виде таблиц это свойство говорит о том, что в каждой позиции на пересечении столбца и строки расположено только одно значение, а не несколько. Таблицы, удовлетворяющие этому условию, находятся в так называемой первой нормальной форме.

**Замечание!** Но в реальной жизни бывают ситуации, когда значение атрибута не может быть представлена атомарно. Например, часто в исторических записях дата рождения иногда указывается «Не известна», или например в графе адрес прописки – «БОМЖ». Как же тогда решить проблему отсутствия информации? Кодд предложил использовать в этом случае специальные значения, которые называются NULL. Т.е. если в кортеже, в какой либо позиции атрибута встречается значение NULL, то считается, что значение атрибута отсутствует