

Лекция №2

Нормализация

- **Процесс проектирования представляет собой процесс нормализации схем отношений, причем каждая следующая нормальная форма обладает свойствами лучшими, чем предыдущая. Каждой нормальной форме соответствует некоторый определенный набор ограничений, и отношение находится в некоторой нормальной форме, если удовлетворяет свойственному ей набору ограничений.**

Примером набора ограничений является ограничение первой нормальной формы - значения всех атрибутов отношения атомарны.

Нормализация

В теории реляционных баз данных обычно выделяется следующая последовательность нормальных форм:

- первая нормальная форма (1NF);
- вторая нормальная форма (2NF);
- третья нормальная форма (3NF);
- нормальная форма Бойса-Кодда (BCNF);
- четвертая нормальная форма (4NF);
- пятая нормальная форма, или нормальная форма проекции-соединения (5NF или PJ/NF).

Нормализация

Основные свойства нормальных форм:

- каждая следующая нормальная форма в некотором смысле лучше предыдущей;
- при переходе к следующей нормальной форме свойства предыдущих нормальных свойств сохраняются.

В основе процесса проектирования лежит метод нормализации, декомпозиция отношения, находящегося в предыдущей нормальной форме, в два или более отношения, удовлетворяющих требованиям следующей нормальной формы.

Наиболее важные на практике нормальные формы отношений основываются на фундаментальном в теории реляционных баз данных понятии функциональной зависимости. Для дальнейшего изложения нам потребуются несколько определений

Нормализация (определения)

Определение 1. Функциональная зависимость

В отношении R атрибут Y функционально зависит от атрибута X (X и Y могут быть составными) в том и только в том случае, если каждому значению X соответствует в точности одно значение Y : $R.X(r) R.Y$.

Определение 2. Полная функциональная зависимость

Функциональная зависимость $R.X(r) R.Y$ называется полной, если атрибут Y не зависит функционально от любого точного подмножества X .

Определение 3. Транзитивная функциональная зависимость

Функциональная зависимость $R.X \rightarrow R.Y$ называется транзитивной, если существует такой атрибут Z , что имеются функциональные зависимости $R.X \rightarrow R.Z$ и $R.Z \rightarrow R.Y$ и отсутствует функциональная зависимость $R.Z \rightarrow R.X$. (При отсутствии последнего требования мы имели бы "неинтересные" транзитивные зависимости в любом отношении, обладающем несколькими ключами.)

Определение 4. Неключевой атрибут

Неключевым атрибутом называется любой атрибут отношения, не входящий в состав первичного ключа (в частности, первичного).

Определение 5. Взаимно независимые атрибуты

Два или более атрибута взаимно независимы, если ни один из этих атрибутов не является функционально зависимым от других.

Нормализация (2 форма)

Определение 6.

Вторая нормальная форма (в этом определении предполагается, что единственным ключом отношения является первичный ключ)

Отношение R находится во второй нормальной форме (2NF) в том и только в том случае, когда находится в 1NF, и каждый неключевой атрибут полностью зависит от первичного ключа

Определение 6~

Отношение R находится во второй нормальной форме (2NF) в том и только в том случае, когда оно находится в 1NF, и каждый неключевой атрибут полностью зависит от каждого ключа R.

Нормализация (3 форма)

Определение 7.

Третья нормальная форма. (Снова определение дается в предположении существования единственного ключа.)

Отношение R находится в третьей нормальной форме (3NF) в том и только в том случае, если находится в 2NF и каждый неключевой атрибут нетранзитивно зависит от первичного ключа.

Определение 7~

Отношение R находится в третьей нормальной форме (3NF) в том и только в том случае, если находится в 1NF, и каждый неключевой атрибут не является транзитивно зависимым от какого-либо ключа R.

На практике третья нормальная форма схем отношений достаточна в большинстве случаев, и приведением к третьей нормальной форме процесс проектирования реляционной базы данных обычно заканчивается. Однако иногда полезно продолжить процесс нормализации.

Получение реляционной схемы из ER-схемы

Получение реляционной схемы из ER-схемы

Шаг 1. Каждая простая сущность превращается в таблицу. Простая сущность - сущность, не являющаяся подтипом и не имеющая подтипов. Имя сущности становится именем таблицы.

Шаг 2. Каждый атрибут становится возможным столбцом с тем же именем; может выбираться более точный формат. Столбцы, соответствующие необязательным атрибутам, могут содержать неопределенные значения; столбцы, соответствующие обязательным атрибутам, - не могут.

Шаг 3. Компоненты уникального идентификатора сущности превращаются в первичный ключ таблицы. Если имеется несколько возможных уникальных идентификатора, выбирается наиболее используемый. Если в состав уникального идентификатора входят связи, к числу столбцов первичного ключа добавляется копия уникального идентификатора сущности, находящейся на дальнем конце связи (этот процесс может продолжаться рекурсивно). Для именования этих столбцов используются имена концов связей и/или имена сущностей.

Шаг 4. Связи многие-к-одному (и один-к-одному) становятся внешними ключами. Т.е. делается копия уникального идентификатора с конца связи "один", и соответствующие столбцы составляют внешний ключ. Необязательные связи соответствуют столбцам, допускающим неопределенные значения; обязательные связи - столбцам, не допускающим неопределенные значения.

Шаг 5. Индексы создаются для первичного ключа (уникальный индекс), внешних ключей и тех атрибутов, на которых предполагается в основном базировать запросы.

Целостность реляционных данных

Во второй части реляционной модели данных определяются два ограничения, которые должны выполняться в любой реляционной базе данных. Это:

- Целостность сущностей.
- Целостность внешних ключей.

Целостность реляционных данных

Для того чтобы обойти проблему неполных или неизвестных данных, в базах данных могут использоваться типы данных, пополненные так называемым null-значением. Null-значение - это, собственно, не значение, а некий маркер, показывающий, что значение неизвестно.

Таким образом, в ситуации, когда возможно появление неизвестных или неполных данных, разработчик имеет на выбор два варианта.

Первый вариант состоит в том, чтобы ограничиться использованием обычных типов данных и не использовать null-значения, а вместо неизвестных данных вводить либо нулевые значения, либо значения специального вида - например, договориться, что строка "АДРЕС НЕИЗВЕСТЕН" и есть те данные, которые нужно вводить вместо неизвестного адреса.

Второй вариант состоит в использовании null-значений вместо неизвестных данных.

Целостность реляционных данных

По определению, тело отношения есть множество кортежей, поэтому отношения не могут содержать одинаковые кортежи. Это значит, что каждый кортеж должен обладать свойством уникальности. На самом деле, свойством уникальности в пределах отношения могут обладать отдельные атрибуты кортежей или группы атрибутов. Такие уникальные атрибуты удобно использовать для идентификации кортежей.

Определение 1. Пусть дано отношение R . Подмножество атрибутов отношения будем называть потенциальным ключом, если оно обладает следующими свойствами:

1. Свойством уникальности - в отношении не может быть двух различных кортежей, с одинаковым значением.
2. Свойством избыточности - никакое подмножество K не обладает свойством уникальности.

Любое отношение имеет по крайней мере один потенциальный ключ. Действительно, если никакой атрибут или группа атрибутов не является потенциальным ключом, то, в силу уникальности кортежей, все атрибуты вместе образуют потенциальный ключ.

Потенциальный ключ, состоящий из одного атрибута, называется простым. Потенциальный ключ, состоящий из нескольких атрибутов, называется составным.

Проектирование базы данных

| | |
|------------------------------|------------------------------------|
| 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 данных в ячейках таблицы |

2 ЛАБ. РАБОТА

1. Создать форматы данных
2. Дополнить таблицу полем с фото(объект OLE)
3. Создать запрос :
 - a. Перекрестный
 - b. Группировка с агрегатной функцией
 - c. С функцией IIF
 - d. Удалить с условием
 - e. Обновить
4. Создать подчиненную форму
5. В форме создать вычисляемое поле
6. Кнопочную форму
7. Создать отчет с вычислением и группированием
8. Создать документ в Word
9. Построить сводную таблицу в Excel по данным таблиц
10. Писать макросы
11. Нормализация БД

СПАСИБО ЗА ВНИМАНИЕ!

