

Лекция 7.

Базы данных

Данные – информация, зафиксированная в определённой форме, пригодной для обработки, хранения, передачи.

База данных (БД) – совокупность определенным образом связанных данных, описывающая некоторую предметную область (часть реального мира, представляющую интерес для исследования и использования). База данных – современная форма хранения и доступа к информации. Базы данных предназначены для централизованного накопления и коллективного многоцелевого использования информации. Их использование позволяет ускорить процесс поиска и обработки информации, существенно уменьшить документооборот.

Основные требования, предъявляемые к базам данных:

- 1) полнота;
- 2) непротиворечивость;
- 3) отсутствие дублирования;
- 4) актуальность информации;
- 5) защищённость от разрушения;
- 6) возможность быстрого и полного восстановления.

Данные хранятся в БД в соответствии с моделью данных. Существуют различные типы моделей данных, например: сетевая, иерархическая, реляционная. Наибольшее распространение получила реляционная модель, в которой данные хранятся в виде двумерных таблиц.

Объекты предметной области и связи между ними

При разработке базы данных сначала исследуется предметная область (например, «Университет»). В ней выделяются основные объекты. Они могут быть реальными («Студент») или абстрактными («Дисциплина»). Каждый объект характеризуется набором свойств – *атрибутов объекта (поля данных)*. Для каждого объекта атрибуты заполняются определенными значениями. Атрибуты могут быть простыми и ключевыми.

Ключевой атрибут (ключ) – это отдельные элементы данных, по которым можно определить все остальные элементы данных («Номер зачетной книжки»). Ключ может быть простым или составным («Фамилия», «Имя», «Отчество»).

После определения основных объектов предметной области с помощью их ключевых атрибутов устанавливаются связи между этими объектами:

- а) 1:1 («один к одному») – каждому экземпляру объекта А соответствует только один экземпляр объекта В и наоборот (рисунок 17).

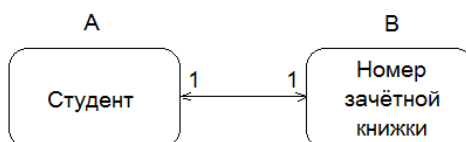


Рисунок 17 – Связь «один к одному»

- б) 1:M («один ко многим») – каждому экземпляру объекта А может соответствовать 0, 1 или несколько экземпляров объекта В, однако каждому экземпляру объекта В соответствует только 1 экземпляр объекта А (рисунок 18).

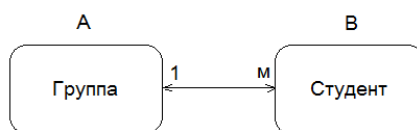


Рисунок 18 – Связь «один ко многим»

- с) M:M («многие ко многим») – каждому экземпляру объекта А соответствует 0, 1 или несколько экземпляров объекта В и наоборот (рисунок 19).

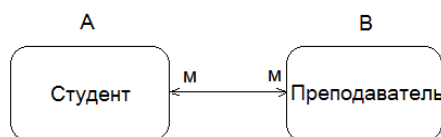


Рисунок 19 - Связь «многие ко многим»

Выделенные основные объекты предметной области с установленными связями между ними представляют собой *инфологическую модель*.

Отношения

Объект предметной области может быть представлен в виде таблицы-отношения – таблицы особого рода, у которой:

- каждая строка содержит информацию об одном экземпляре объекта (строка отношения - *кортеж*);
- все столбцы однородные, то есть все элементы в столбце имеют одинаковый тип и длину, имеют имя и содержат информацию об отдельном атрибуте объекта;
- каждый элемент представляет собой один элемент данных об объекте;
- все строки и столбцы уникальны (нет повторений);
- в таблицах нет пустых ячеек.

Базы данных, основанные на таблицах-отношениях, называются *реляционными* (*relation* - *отношение*). Набор отношений (таблиц) используется в БД для хранения информации об объектах реального мира и моделирования связей между ними. Например, для хранения объекта «студент» используют отношение **СТУДЕНТ**, в котором свойства объекта располагаются в столбцах таблицы, являющихся атрибутами объекта (таблица 8):

Таблица 8 – Отношение **СТУДЕНТ**

Фамилия	Возраст	Группа
Петров	17	МТЗ-12
Иванова	16	МТЗ-12
Сидоров	17	МТ10-12

Список имен атрибутов отношения называется *схемой отношения*. Схему отношения **СТУДЕНТ** можно записать так: **СТУДЕНТ = (Фамилия, Возраст, Группа)**.

Реляционная БД – набор взаимосвязанных отношений. Каждое отношение (таблица) в ЭВМ представляется в виде файла записей.

Над таблицами - отношениями можно выполнять восемь различных операций теории множеств и реляционной алгебры (объединение, выборка, проекция, пересечение, сложение, умножение, разность, деление). Вследствие этого из введенных (базовых) отношений можно получать много новых (вычисляемых) таблиц - отношений (отчетов, выборок, запросов и т.п.).

Благодаря тому, что информация в базах данных представлена в двух видах – хранимая информация (исходные, введенные таблицы) и вычисляемая информация (таблицы, полученные на основании исходных), можно существенно экономить память и ускорить процесс обработки этой информации.

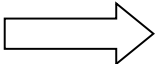
Для создания простой и надёжной базы данных необходимо нормализовать отношения. *Нормализация отношений* – пошаговый процесс разложения отношений на

более мелкие и простые. Не смотря на увеличение при этом количества отношений, операции доступа к данным существенно ускоряются благодаря улучшению корректности, устранению дублирования и обеспечению непротиворечивости данных в базе.

Существует несколько *нормальных форм*:

1-я нормальная форма. Отношение считается находящимся в первой нормальной форме, если все его атрибуты – неделимые (простые). К примеру, приведенное ниже на рисунке 20 отношение не нормализовано, поскольку содержит сложный атрибут **Спорт**. Чтобы привести это отношение к нормализованному виду, нужно избавиться от этого сложного атрибута.

Фамилия	Группа	Спорт	
		Вид	Разряд
Иванов	MT3-12	Шахматы	МС
		Хоккей	1 разряд
Петров	MT10-12	Футбол	2 разряд
		Шашки	КМС

Фамилия	Вид спорта	Группа	Разряд
Иванов	Хоккей	MT3-12	1 разряд
Иванов	Шахматы	MT3-12	МС
Петров	Футбол	MT10-12	2 разряд
Петров	Шашки	MT10-12	КМС

Рисунок 20 – Приведение к первой нормальной форме

В полученном отношении ключ является составным, состоящим из атрибутов **Фамилия** и **Вид спорта**.

2-я нормальная форма. Отношение считается находящимся во второй нормальной форме, если все его атрибуты зависят от составного ключа в целом, а не от его частей. Следовательно, если отношение находится в первой нормальной форме и имеет простой, а не составной ключ, то оно автоматически находится и в первой, и во второй нормальной форме.

Например, в отношении **ВЕДОМОСТЬ** (рисунок 21), имеющем составной ключ «**Студент, Дисциплина**», атрибут **Лектор** зависит только от **Дисциплины**, а не от всего

ключа. Это отношение можно нормализовать, «разбив» его на два отношения **УСПЕВАЕМОСТЬ** и **ПРЕПОДАВАТЕЛЬ**:

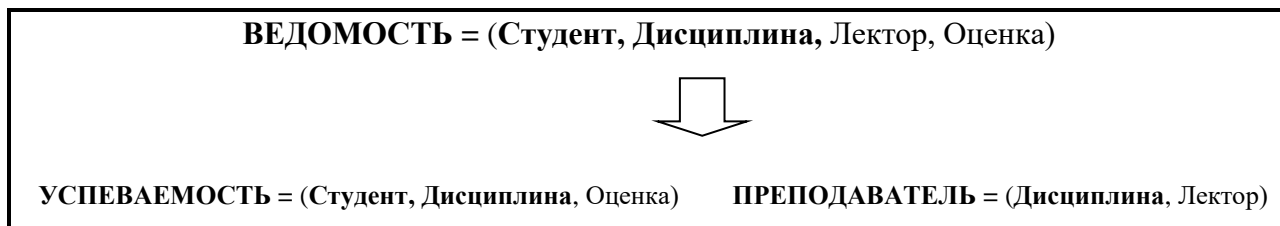


Рисунок 21 – Приведение ко второй нормальной форме

3-я нормальная форма. Отношение считается находящимся в третьей нормальной форме, если устранены зависимости между не ключевыми атрибутами (транзитивные зависимости). Например, в отношении **ПРЕДМЕТ = (Название, Лектор, Кафедра, Телефон)** не ключевой атрибут **Телефон** зависит от не ключевого атрибута **Кафедра**.

Для устранения транзитивной зависимости необходимо «расщепить» исходное отношение на два **ДИСЦИПЛИНА = (Название, Лектор, Кафедра)** и **ДАННЫЕ КАФЕДРЫ = (Кафедра, Телефон)**.

Дальнейшее упрощение таблиц связано с дальнейшим ограничением типов зависимости между атрибутами отношений.

После нормализации отношений и установления связей между ними формируется инфологическая модель предметной области. Ниже (на рисунке 22) представлен пример инфологической модели фирмы, оформляющей сделки с заказчиками через своих сотрудников-менеджеров:

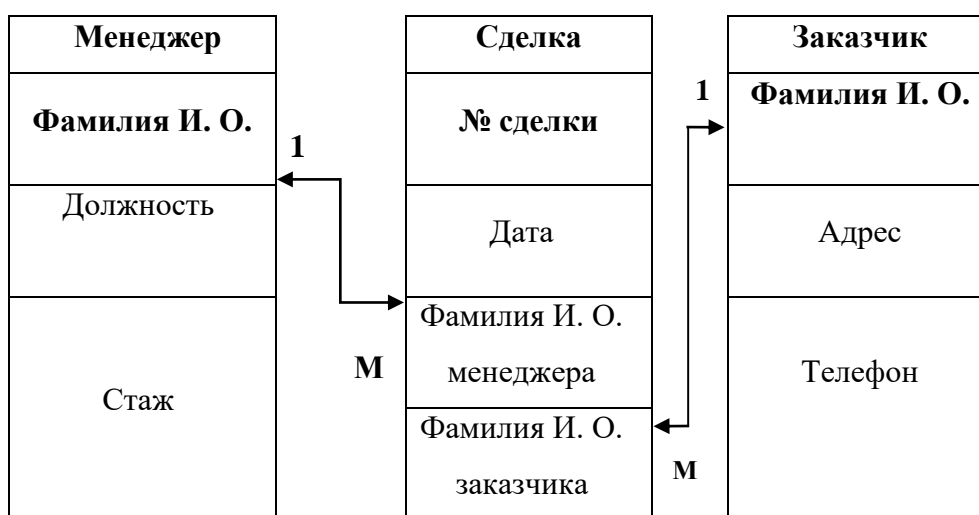


Рисунок 22 – Модель фирмы

На основании инфологической модели разрабатывается модель данных, которая дает описание логической структуры базы данных на языке описания данных (ЯОД), – *даталогическая модель (ДМ)*.

Для привязки ДМ к среде хранения используется модель данных физического уровня – *физическая модель (ФМ)*. На этом этапе физического проектирования базы данных осуществляется выбор типа носителя, разрабатывается формат хранимых записей и проектируются методы доступа к данным.

СУБД

После этого уже возможно формирование (заполнение) базы данных и непосредственно работа с ней. Работа с базами данных сводится к выполнению следующих операций:

- 1) запись (заполнение базы данных);
- 2) просмотр;
- 3) редактирование (добавление, удаление, исправление);
- 4) выборка (запросы, отчеты).

Эти операции накопления и манипулирования данными выполняет специальная программа – *система управления базами данных (СУБД)*.

По технологии решения задач, выполняемых СУБД, базы данных можно разделить на два вида:

- централизованная БД (хранится целиком на ВЗУ одной вычислительной системы и, если система входит в состав сети, то возможен доступ к этой БД других систем);
- распределенная БД (состоит из нескольких, иногда пересекающихся или дублирующих друг друга БД, хранящихся на ВЗУ разных узлов сети).

СУБД предоставляет доступ к данным БД двумя способами:

- локальный доступ (предполагает, что СУБД обрабатывает БД, которая хранится на ВЗУ того же компьютера);
- удаленный доступ (это обращение к БД, которая хранится на одном из узлов сети).

Удаленный доступ может быть выполнен по технологии файл-сервер или клиент-сервер. Технология файл-сервер предполагает выделение одной из вычислительных систем, называемой сервером, для хранения БД. Все остальные компьютеры сети (клиенты) исполняют роль рабочих станций, которые копируют требуемую часть централизованной БД в свою память, где и происходит обработка. Технология клиент-

сервер предполагает, что сервер, выделенный для хранения централизованной БД, дополнительно производит обработку запросов клиентских рабочих станций. Клиент посылает запрос серверу. Сервер пересылает клиенту данные, являющиеся результатом поиска в БД по ее запросу.

Система управления базами данных – совокупность программных и языковых средств.

Программные средства обеспечивают организацию ввода, обработки и хранения данных, а также обеспечивают взаимодействие всех частей системы при её функционировании (настройка, тестирование, восстановление).

Языковые средства обеспечивают взаимодействие пользователя с базой данных.

К ним относятся:

- языки манипулирования данными (ЯМД) – языки запросов к БД, представляющие собой систему команд для работы с данными (выборка, запрос, вставка, удаление и т.п.);
- языки определения данных (ЯОД) – языки, предназначенные для создания схемы базы данных (описания типов данных, структуры базы, взаимодействия и связей между элементами).

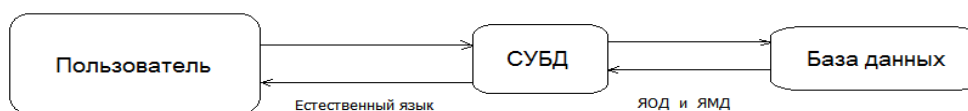


Рисунок 22 - Схема взаимодействия пользователя с базой данных

Современная СУБД – прикладная программа, которая предназначена для облегчения работы неквалифицированного пользователя с БД. Он работает с ней на естественном языке без знания языка манипулирования данными и языка определения данных (рисунок 22). Одним из примеров такой СУБД является широко известный продукт фирмы Microsoft – СУБД Access.