**Урок на тему «База данных. Системы управления базами данных»**

Цель урока: Изучение баз данных, табличных баз данных, системы управления базами данных.

Задачи:

*Образовательная:*

познакомить учащихся с понятиями: типами баз данных, системой управления базой данных, объекты СУБД; дать первоначальные знания по работе с базой данных Microsoft Access.

*Развивающая*: развивать алгоритмическое мышление и творческие способности учащихся.

*Воспитательная:* воспитывать познавательный интерес к предмету информатика с помощью практических заданий; воспитывать у учащихся умение самостоятельно решать вопросы, а также работать в группе.

**Ход урока**

**База данных (БД)** – совокупность определенным образом организованной информации на какую-то тему (в рамках некоторой предметной области).

Например:

* база данных книжного фонда библиотеки;
* база данных кадрового состава учреждения;
* база данных законодательных актов в области уголовного права;
* база данных современной эстрадной песни.

Конечно, вся эта информация может храниться и на бумаге (например, книжный каталог библиотеки). Но современным средством хранения и обработки баз данных является, безусловно, компьютер. В дальнейшем мы будем иметь в виду только компьютерные БД.

 Базы данных бывают **фактографическими и документальными**.

В фактографических БД содержатся краткие сведения об описываемых объектах, представленные в строго определенном формате. Из приведенных выше примеров две первые БД скорее всего будут организованы как фактографические. В БД библиотеки о каждой книге хранятся библиографические сведения: год издания, автор, название и пр. Разумеется текст книги в ней содержаться не будет. В БД отдела кадров учреждения хранятся анкетные данные сотрудников: фамилия, имя, отчество; год и место рождения.

Базы данных в третьем и четвертом примерах наверняка будут организованы как документальные. Первая из них будет включать в себя тексты законов; вторая – тексты и ноты песен; биографическую и творческую справочную информацию о композиторах, поэтах, исполнителях; звуковые записи и видеоклипы. Следовательно, документальная БД содержит обширную информацию самого разного типа: текстовую, графическую, звуковую, мультимедийную.

Современные информационные технологии постепенно стирают границу между фактографическими и документальными БД. Существуют средства, позволяющие легко подключать любой документ (текстовый, графический, звуковой) к фактографической базе данных.

Сама по себе база данных не может обслужить запросы пользователя на поиск и обработку информации. БД – это только «информационный склад». Обслуживание пользователя осуществляет информационная система.

**Информационная система** – это совокупность базы данных и всего комплекса аппаратно-программных средств для ее хранения, изменения и поиска информации, для взаимодействия с пользователем.

Примерами информационных систем являются системы продажи билетов на пассажирские поезда и самолеты. WWW – это тоже пример глобальной информационной системы.

Далее будет идти речь будет только о фактографических базах данных. Дадим более строгое определение компьютерной БД, чем то, что приводилось выше.

**База данных** – организованная совокупность данных, предназначенная для длительного хранения во внешней памяти ЭВМ и постоянного применения. Для хранения БД может использоваться как один компьютер, так и множество взаимосвязанных компьютеров.

Если различные части одной базы данных хранятся на множестве компьютеров, объединенных между собой сетью, то такая БД называется **распределенной базой данных**.

  

Очевидно, информацию в Internet, объединенную паутиной WWW, можно рассматривать как распределенную базу данных. Распределенные БД создаются также и в локальных сетях.

Модель данных строится по принципу взаимосвязанных таблиц - **реляционной**.

Один тип объекта является главным, все нижележащие – подчиненными - **иерархической.**

Любой тип данных одновременно может быть главным и подчиненным - **сетевой**.

**Табличные базы данных**

Информация в базах данных может быть организована по разному. Чаще всего используется табличный способ.

Базы данных с табличной формой организации называются реляционными БД.

В чем же их преимущество?

Главное достоинство таблиц – в их понятности. С табличной информацией мы имеем дело практически каждый день. Загляните, например, в свой дневник: расписание занятий там представлено в виде таблицы, ведомость с оценками за четверти имеет табличный вид. Когда мы приходим на вокзал, смотрим расписание электричек. Какой вид оно имеет? Это таблица! А еще есть таблица футбольного чемпионата. И журнал учителя, куда он ставит вам оценки – тоже таблица.

Видите, как много примеров, и их еще можно продолжить. Мы настолько привыкли к таблицам, что обычно не требуется никому объяснять, как ими пользоваться. Ну разве что маленькому ребенку, который только учится читать.

В реляционных БД строка таблицы называется записью, а столбец – полем. В общем виде это выглядит так:

Каждое поле таблицы имеет имя. Например, в таблице «Игрушки» имена полей такие: НАЗВАНИЕ, МАТЕРИАЛ, ЦВЕТ, КОЛИЧЕСТВО.

Одна запись содержит информацию об одном объекте той реальной системы, модель которой представлена в таблице.

Например, одна запись о каком либо объекте – это информация об одной игрушке.



**Поля** – это различные характеристики (иногда говорят – атрибуты) объекта. Значения полей в одной строчке относятся к одному объекту. Разные поля отличаются именами. А чем отличаются друг от друга разные записи? Записи различаются значениями ключей.

Главным **ключом** в базах данных называют поле (или совокупность полей), значение которого не повторяется у разных записей.

В БД «Домашняя библиотека» разные книги могут иметь одного автора, могут совпадать названия книг, год издания, полка. Но инвентарный номер у каждой книги свой (поле НОМЕР). Он-то и является главным ключом для записей в этой базе данных.

Не всегда удается определить одно поле в качестве ключа. Вот, например, база данных, которая хранится в компьютере управления образования области. В ней содержатся сведения о всех средних школах районных центров в виде такой таблицы:

В такой таблице у разных записей не могут совпасть только одновременно два поля ГОРОД и НОМЕР ШКОЛЫ. Эти два поля вместе образуют составной ключ: ГОРОД-НОМЕР ШКОЛЫ. Составной ключ может состоять и более чем из двух полей.

С каждым полем связано еще одно очень важное свойство – **тип поля**.

Тип определяет множество значений, которые может принимать данное поле в различных записях.

 

 В реляционных базах данных используются четыре основных типа полей:

* числовой;
* символьный;
* дата;
* логический.

**Числовой тип** имеют поля, значения которых могут быть только числами. Например, в БД «Погода» три поля числового типа: ТЕМПЕРАТУРА, ДАВЛЕНИЕ, ВЛАЖНОСТЬ.

**Символьный тип** имеют поля, в которых будут храниться символьные последовательности (слова, тексты, коды и т.п.). Примерами символьных полей являются поля АВТОР и НАЗВАНИЕ в БД «Домашняя библиотека»; поле ТЕЛЕФОН в БД «Школы».

**Тип «дата»** имеют поля, содержащие календарные даты в форме «день/месяц/год» (в некоторых случаях используется американская форма: месяц/день/год). Тип «дата» имеет поле ДЕНЬ в БД «Погода».

**Логический тип** соответствует полю, которое может принимать всего два значения: «да» – «нет» или «истина» – «ложь» или (по-английски) «true» – «false». Если двоичную матрицу представить в виде реляционной БД (табл. 6.4, 6.5), то ее полям, принимающим значения «О» или «1», удобно поставить в соответствие логический тип. При этом «1» заменится на значение «истина», «О» – на значение «ложь».

Итак, значения полей – это некоторые величины определенных типов.

От типа величины зависят те действия, которые можно с ней производить.

Например, с числовыми величинами можно выполнять арифметические операции, а с символьными и логическими – нельзя.

Для взаимодействия пользователя с базами данных используют **системы управления данными (СУБД).**

Принципы построения систем управления баз данных следуют из требований, которым должна удовлетворять организация баз данных:

**Производительность и готовность.** Запросы от пользователя базой данных удовлетворяются с такой скоростью, которая требуется для использования данных. Пользователь быстро получает данные всякий раз, когда они ему необходимы.

**Минимальные затраты.** Низкая стоимость хранения и использования данных, минимизация затрат на внесение изменений.

**Простота и легкость использования**. Пользователи могут легко узнать и понять, какие данные имеются в их распоряжении. Доступ к данным должен быть простым, исключающим возможные ошибки со стороны пользователя.

**Простота внесения изменений.** База данных может увеличиваться и изменяться без нарушения имеющихся способов использования данных.

**Возможность поиска**. Пользователь базы данных может обращаться с самыми различными запросами по поводу хранимых в ней данных. Для реализации этого служит так называемый язык запросов.

**Целостность.** Современные базы данных могут содержать данные, используемые многими пользователями. Очень важно, чтобы в процессе работы элементы данных и связи между ними не нарушались. Кроме того, аппаратные ошибки и различного рода случайные сбои не должны приводить к необратимым потерям данных. Значит, система управления данными должна содержать механизм восстановления данных.

**Безопасность и секретность**. Под безопасностью данных понимают защиту данных от случайного или преднамеренного доступа к ним лиц, не имеющих на это права, от неавторизированной модификации (изменения) данных или их разрушения. Секретность определяется как право отдельных лиц или организаций решать, когда, как какое количество информации может быть передано другим лицам или организациям.

Одно из самых важных преимуществ современных СУБД состоит в логической и физической независимости данных. Например, база данных, реализованная в СУБД DBASE, физически содержит, как минимум, три файла. В то же время эту же базу данных можно перенести в СУБД Microsoft Access, где она физически разместится в одном файле. При этом логическая организация данных не изменится.

Развитие аппаратного и программного обеспечения, средств телекоммуникаций привело к тому, что на сегодняшний день наметился переход от традиционных баз данных, хранящих числа и символы объектно-реляционным базам данных, где каждая запись может содержать данные со сложным поведением. Пример тому развитие internet-технологий. Современный настольные компьютеры и программы просмотра Web - браузеры - позволяют осуществлять поиск в глобальной сети и просматривать большую часть мультимедийных данных.

Далее на примере одной из самых распространенных систем управления базами данных - Microsoft Access входит в состав популярного пакета Microsoft Office - мы познакомимся с основными типами данных, способами создания баз данных и с приемами работы с базами данных.

Существует огромное количество СУБД, например: Lotus Approach, Visual FoxPro, Borland Paradox, Borland dBase.

 Используемая литература:

 Бешенков С.А., Ракитина Е.А. Информатика. Систематический курс. Учебник для 10 класса. М.: Лаборатория Базовых Знаний, 2001

Дополнительно

**Кроссворд по теме: «Базы данных с ключевым словом»**

Перенеси кроссворд в тетрадь и разгадай его.



1. Структура БД, в которой любой тип данных одновременно может быть главным и подчиненным
2. Средство извлечения данных из БД
3. Набор программ, с помощью которых производится работа с базой данных
4. Столбец в табличной БД
5. Строка в табличной БД
6. Структура БД, в которой данные строятся по принципу взаимосвязанных таблиц
7. Структура, в которой одно данное является главным, все нижележащие – подчиненными.

Дайте определение получившемуся вертикальному понятию с точки зрения баз данных.

Ответы: сетевая, запрос, СУБД, поле, запись, реляционная, иерархия.

Таблица – основной объект реляционной БД.

Дидактические материалы

**Практическая работа № 1MS Access 2003.**

 Основные приемы работы с данным

**Задание:** Создание базы данных. Знакомство с основными объектами базы данных. Создание и заполнение таблицы. Режимы представления таблицы. Типы данных. Маска ввода. Изменение структуры таблицы в режиме конструктора.

**Создание базы данных**

1. Заранее создайте каталог (папку) с названием BAZA.
2. Запустите MS Access 2003
3. Нажмите кнопку Создать базу данных на панели инструментов или в меню Файл – Создать. Будет открыто окно диалога Создание базы данных, в которое следует ввести имя базы, например Adress и выбрать каталог BAZA. В результате вы увидите окно базы данных.

 

**Создание таблицы с помощью Мастера**

 В окне База данных нажмите кнопку Таблицы, а затем кнопку Создание таблицы с помощью мастера

1. В окне Создание таблицы выберите образец Список рассылки (самый первый в списке). Далее нужно определить перечень полей, которые вы хотите включить в таблицу. Для этого поочередно выделяйте мышью поля в списке Образцов полей и нажимайте кнопку чтобы образец поля попал в список полей создаваемой таблицы. Выберите следующие поля: Код Списка Рассылки, Фамилия, Имя, Адрес, Домашний телефон, Сотовый телефон
2. Нажмите кнопку Создать  базу данных на панели инструментов или в меню Файл – Создать. Будет открыто окно диалога Создание базы данных, в которое следует ввести имя базы, например Adress и выбрать каталог BAZA. В результате вы увидите окно базы данных.



1. Переименуем названия некоторых полей, выбранных из шаблона. Для этого установите курсор в окне Поля новой таблицы на строке Код Списка Рассылки и щелкните кнопку Переименовать поле и нажмите кнопку Далее.
2. До тех пор, пока кнопка Далее активна, пользуйтесь ею, кнопку Готово нажимайте только в последнем случае!!!
3. В следующем окне диалога введите имя таблицы Адреса и разрешите MS Access 2003 самостоятельно определить ключ и нажмите кнопку Далее
4. В последнем окне диалога согласитесь на Непосредственный ввод данных в таблицу. И нажмите кнопку Готово.

Таблица создана. В строке заголовков таблицы можно найти ее название. Все имена полей представлены в виде заголовков таблицы.

 Измените заголовок первого поля Код Списка Рассылки на Адресат.

**Заполнение таблицы**

В процессе создания таблицы, вы встретились с понятием ключа.

**Первичный ключ** (в последующем будем называть просто ключом) – одно или несколько полей, совокупность которых однозначно определяет любую запись таблицы.

В нашем варианте ключевым является поле **Адресат**. В таблице видно, что это поле – счетчик, т.е. нумерует записи в порядке ввода. Заполнять поле Адресат не нужно, счетчик срабатывает автоматически, как только заполнена хоть одна ячейка строки таблицы.

Заполните только поля Фамилия и Имя, остальные поля оставьте незаполненными.

Попробуйте заполнить поле **Телефон.**

 

Любая таблица Microsoft Access 2003 может быть представлена в двух режимах:

1. режиме таблицы, предназначенном для ввода данных, их просмотра и редактирования;
2. режиме конструктора, предназначенном для создания и изменения таблицы.

Если внимательно рассмотреть содержимое ячейки, то можно увидеть некоторую разметку – маску ввода.

**Маска ввода** – это шаблон, позволяющий вводить в поле значения, имеющие одинаковый формат. Маска ввода автоматически изображает в поле постоянные символы. При вводе данных в поле нет необходимости набирать эти постоянные символы, даже если они должны быть включены в значение поля; достаточно заполнить пустые позиции в маске ввода. Кроме того, Microsoft Access 2003 не позволит ввести в поле значения, не вписывающиеся в определенную для этого поля маску ввода.

Для того, чтобы изменить маску ввода для телефона, следует переключиться в режим конструктора таблицы. Это можно сделать, выбрав команду Конструктор меню Вид или при помощи кнопки панели инструментов .

Окно конструктора содержит таблицу, в первом столбце которой перечислены поля, а во втором типы данных (форматы).

1. Зададим маску ввода для полей Домашний Телефон и Сотовый Телефон. Установите текстовый курсор в имя поля Домашний телефон в нижнем поле Общие напротив Маски ввода и создайте шаблон в окне Создание масок ввода щелкните Cписок введите в поле Описание – Номер телефона, Маска ввода – 0 00 00, Образцы данных – 2-66-34 и щелкните по кнопке Вперед на 1 запись . Создайте маску ввода для номера сотового телефона: Описание – Номер телефона, Маска ввода – 000-000-00-00, Образец данных – 902-275-78-33 щелчок по и Закрыть.
2. В окне Создание масок ввода щелкните по полю Номер телефона 2-66-34 – Далее – Готово.
3. Щелкнув по строке Конструктора Сотовый телефон, измените для него маску ввода на десятизначный номер из созданного вами шаблона. Перейдите из режима Конструктора в режим Таблицы, для этого выполните команду Вид – Таблица или нажмите кнопку на панели инструментов . На вопрос о сохранении таблицы отвечайте Да. Далее заполнение номеров телефонов заключается в вводе в каждую ячейку цифр, перемещая курсор из ячейки в ячейку.
4. Вернитесь в таблицу и заполните поле Адрес. Введите данные в поле Адрес. Измените ширину столбцов таблицы по аналогии с электронными таблицами. Приблизительный вид таблица показан на рисунке ниже.