ТЕМА ЗАНЯТИЯ: ***«Понятие баз данных. Понятие СУБД. Объекты базы данных. Типы связей между таблицами».***

Цель занятия: Проверить уровень знаний по теме «Основы работы с табличным процессором». Повысить уровень знаний по теме «Базы данных. Системы управления базами данных».

Задачи:

Образовательная

- формировать представление о базах данных и назначении систем управления базами данных (СУБД);
- формировать представления об основных объектах СУБД;

Развивающая

- развивать память, внимание, алгоритмическое мышление;
- способствовать развитию творческой деятельности учащихся и интереса к предмету информатика;

Воспитательная

- воспитывать и формировать познавательный интерес к информатике;
- воспитывать культуру общения, умение внимательно выслушивать мнение других.

Тип урока: комбинированный

Форма урока: урок изучения нового материала.

Методы обучения: эвристический.

Методическое и программное обеспечение урока: мультимедийный проектор, экран, ПК, ПО Open Office Base, презентация.

План проведения урока:

1. Организационный момент.
2. Контрольная работа.
3. Актуализация знаний. Мотивация учебной деятельности учащихся.
4. Изучение нового материала.
5. Закрепление изученного материала.
6. Подведение итогов урока.

ХОД ЗАНЯТИЯ

1 урок

1. Организационный момент

Проверка готовности учащихся к уроку, настрой на работу, постановка цели и задач урока.

2. Контрольная работа

Учитель раздает обучающимся задания контрольной работы.
Выполнение заданий контрольной работы.

3 урок

1. Организационный момент

Форма работы: фронтальная.

Включение учащихся в деятельность на личностно-значимом уровне, который направлен на формирование положительных эмоций. Проверка готовности учащихся к уроку, настрой на работу, постановка с учащимися плана урока.

3. Актуализация знаний. Мотивация учебной деятельности учащихся

Форма работы: фронтальная.

Формирование УУД: личностные (смыслообразование), регулятивные (целеполагание), познавательные, коммуникативные (умение выражать мысли).

Перед началом объяснения нового материала в форме фронтальной беседы нужно создать «проблемную ситуацию», которая подведет к изучению новой темы: Представьте себе, что вы медицинские работники школы с количество учеников более 1000, где на каждого ученика заведена медицинская карточка, в которой зафиксированы важнейшие данные: ФИО, г/р, пол, домашний адрес, телефон, персональные сведения. Представьте, что все карточки находятся в шкафу в хаотичном беспорядке. Сколько времени вам понадобится чтобы найти вашу?

Но если их разложить по ступеням классов (1…11), затем в алфавитном порядке в каждой ступени (1а, 1б, …, 11а, 11б, …), а внутри каждой ступени конкретного класса тоже в алфавитном порядке, т.е. организовать картотеку, то на поиск карточки понадобиться несколько секунд.

Поиск информации также можно поручить компьютеру. Для этого были созданы программы – информационно-поисковые системы.

4. Изучение нового материала

Форма работы: фронтальная, индивидуальная.

Информационно-поисковая система – это система, где хранится информация, из которой по требованию пользователя выдается нужная информация, поиск которой осуществляется либо в ручную, либо автоматически.

Информационно-поисковая система состоит из двух частей:

- большой, специально организованной совокупности данных (она называется **базой данных**);
- программа, позволяющая оперировать этими данными (**СУБД** – система управления базой данных)

В настоящее время в мире создано сотни тысяч информационно-поисковых систем. А где используют базы данных? (Ответ учащихся: Они применяются в банках, в библиотеках, в больницах, в институтах, в магазинах и т. д.) Некоторые

информационно-поисковые системы объединяются в крупные, централизованные информационно-поисковые системы, и называются **банками данных или базами данных**.

Так вот, классифицированные нами медицинские данные учеников и представляют собой ***базу данных***. А что такое база данных? (Ответы учащихся)

База данных **–** это совокупность специально организованных данных, которые хранятся на каком – либо носителе (бумаге, диске и т.п.)

А что же такое система управления базами данных? (Ответы учащихся)

СУБД **–** система программ, позволяющая создавать БД, обновлять хранимую в ней информацию, обеспечивающая удобный доступ к ней с целью просмотра и поиска.

Современна СУБД должна также обеспечивать возможность работы с БД в глобальных и локальных сетях.

Широко используются такие БД:

- информационные системы по продаже и резервированию авиа – и железнодорожных билетов;
- заменяющие привычный библиотечный каталог;
- электронные энциклопедии со сведениями, например, о музыкальных инструментах, шедеврах Эрмитажа, сотрудниках какого-либо учреждения и т.п.;

Устройства внешней памяти, предназначенные для хранения БД, должны иметь высокую информационную емкость и малое время доступа к хранимой информации.

В работе с СУБД возможны следующие режимы: создание, редактирование, поиск, манипулирование. Под манипулированием понимаются такие действия с БД, как: просмотр, копирование файлов на какой-либо носитель, сортировка данных по заданному признаку и т.д.

Для работы с БД СУБД должна обеспечивать:

- возможность внесения и чтения информации;
- работу с большим объемом данных;
- быстроту поиска данных;
- целостность данных (их непротиворечивость);
- защиту от разрушения, уничтожения (не только при случайных ошибках пользователя), от несанкционированного доступа;
- систему дружественных подсказок (в расчете на пользователя без специальной подготовки)

Данные находящиеся в одной базе, должны быть логически взаимосвязаны.

БД классифицируются по трем признакам:

I. <u>По характеру хранимой информации:</u>

- <u>Фактографические</u> – содержат краткие сведения об описываемых объектах, представленных в строго определенном формате. ***Например:*** *БД книжного фонда библиотеки хранит следующие данные о книге: год*

издания, автор, название. Сам текст книги в БД содержаться не будет. Т.е. фактографические БД – это картотеки;

- Документальные – содержат обширную информацию самого разного типа: текстовую, графическую, звуковую, мультимедийную. Документальные БД – архивы.

II. По способу хранения информации:

- Централизованные – вся информация хранится на одном ПК;
- Распределенные – части информации хранятся на разных ПК, используются в локальных и глобальных компьютерных сетях.

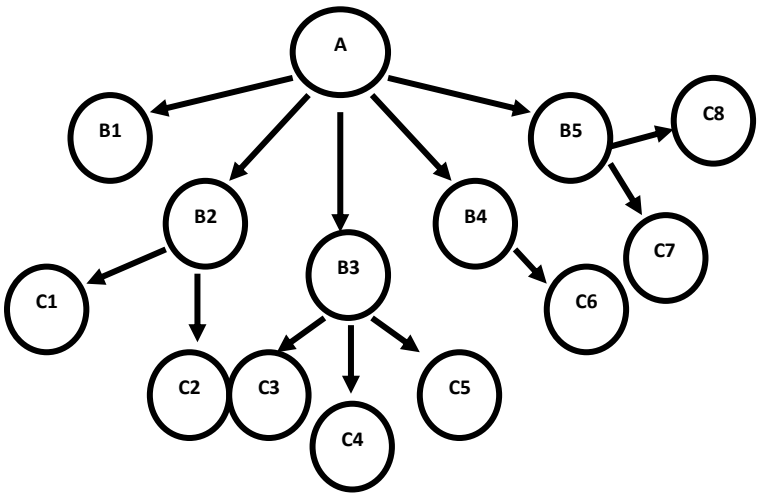
III. По структуре организации:

- Реляционные;
- Иерархические;
- Сетевые.

Ядром любой БД является модель данных. ***Модель данных*** – структура организации данных и способов их обработки.

Иерархическая модель данных – совокупность элементов, связанных между собой по определенным правилам. В ней существует упорядоченность элементов в записи, один элемент считается главным, остальные – подчиненными. Данные в записи упорядочены в определенную последовательность, как ступеньки лестницы, и поиск данных может осуществляться лишь последовательным «спуском» со ступеньки на ступеньку.

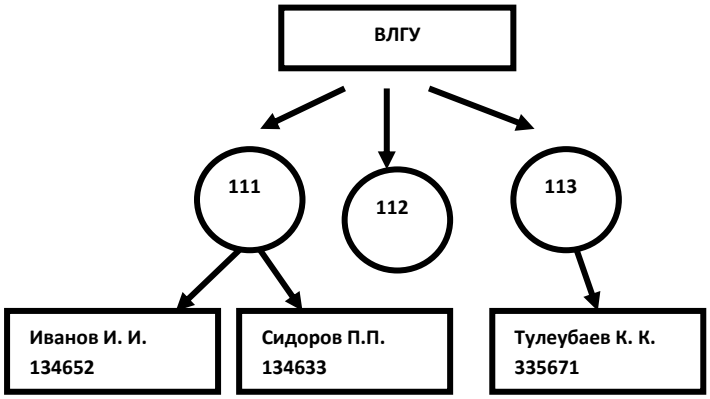
Графически иерархическая модель данных будет иметь вид:



A – уровень 0

B – уровень 1

Пример: *Институт – Группа – Студент*



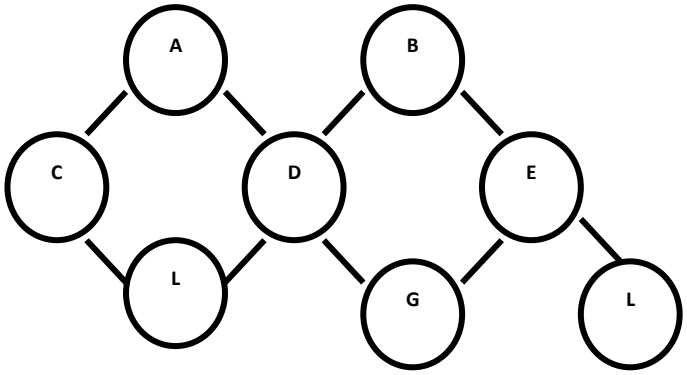
Название института – уровень 0

№ группы – уровень 1

ФИО, № зачетной книжки – уровень 2

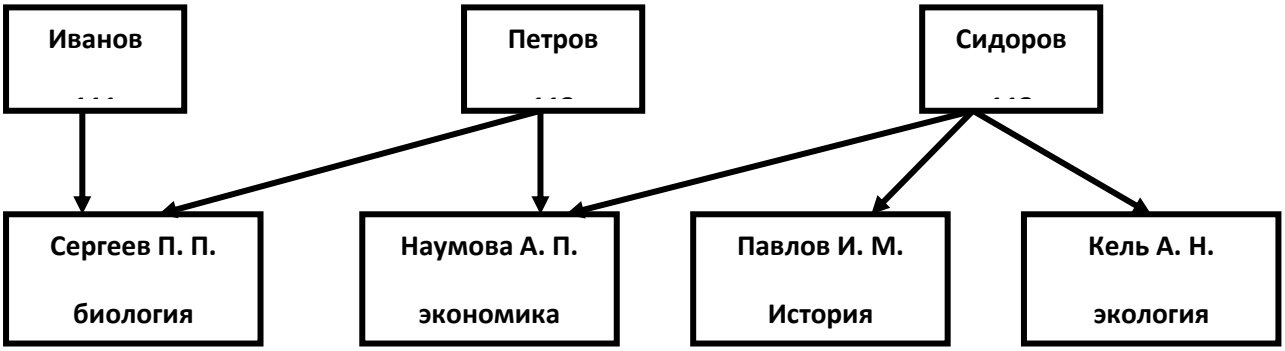
Сетевая модель данных – каждый ее элемент связан с любым другим элементом. Т. е. эта БД отличается большей гибкостью, так кА в ней существует возможность устанавливать дополнительно к вертикальным иерархическим связям горизонтальные связи. Это облегчает процесс поиска требуемых элементов данных, так как не требует обязательного прохождения всех предшествующих ступеней.

Графически сетевая модель данных будет иметь вид:



Пример: *Структура БД, содержащей сведения о студентах, участвовавших в олимпиадах.*

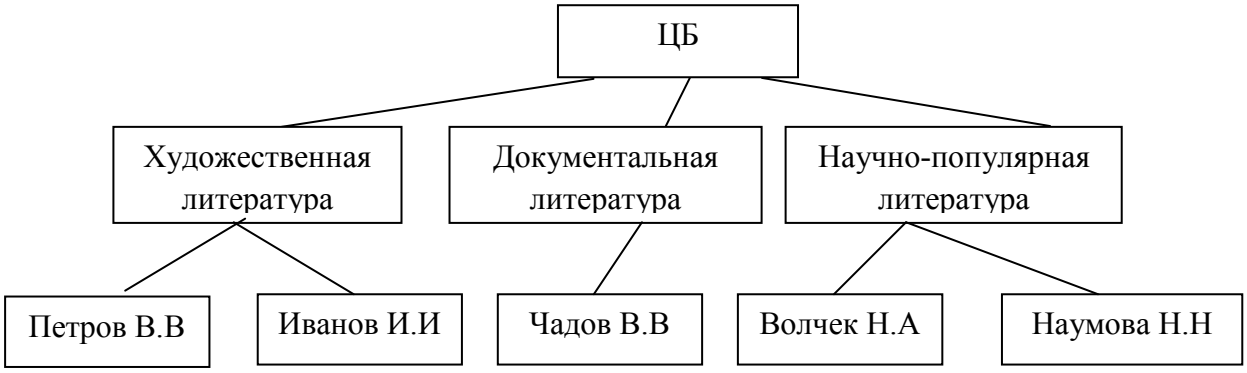
Один студент участвует в нескольких олимпиадах, несколько студентов участвуют одной олимпиаде.



5. Закрепление изученного материала

Предлагаю учащимся выполнить следующие задания.

1. Дана иерархическая модель данных. Определить к какой БД будет относиться данная модель по характеру хранимой информации.



2. Добавьте год издания книг авторов, предложенных в задании 1 и превратите иерархическую БД в сетевую.

4. Изучение нового материала

Основные этапы разработки баз данных

При разработке БД можно выделить следующие этапы работы:

I этап. Постановка задачи.

На этом этапе формируется задание по созданию БД. В нем подробно описывается состав базы, назначение и цели ее создания, а также перечисляется, какие виды работ предполагается осуществлять в этой базе данных (отбор, дополнение, изменение данных, печать или вывод отчета и т. д).

II этап. Анализ объекта.

На этом этапе рассматривается, из каких объектов может состоять БД, каковы свойства этих объектов. После разбиения БД на отдельные объекты необходимо рассмотреть свойства каждого из этих объектов, или, другими словами, установить, какими параметрами описывается каждый объект. Все эти сведения можно располагать в виде отдельных записей и таблиц. Далее необходимо рассмотреть тип данных каждой отдельной единицы записи. Сведения о типах данных также следует занести в составляемую таблицу.

III этап. Синтез модели.

На этом этапе по проведенному выше анализу необходимо выбрать определенную модель БД. Далее рассматриваются достоинства и недостатки каждой модели и сопоставляются с требованиями и задачами создаваемой БД. После такого анализа выбирают ту модель, которая сможет максимально обеспечить реализацию поставленной задачи. После выбора модели необходимо нарисовать ее схему с указанием связей между таблицами или узлами.

IV этап. Выбор способов представления информации и программного инструментария.

После создания модели необходимо, в зависимости от выбранного программного продукта, определить форму представления информации.

В большинстве СУБД данные можно хранить в двух видах:

- с использованием форм;
- без использования форм.

Форма – это созданный пользователем графический интерфейс для ввода данных в базу.

V этап. Синтез компьютерной модели объекта.

В процессе создания компьютерной модели можно выделить некоторые стадии, типичные для любой СУБД.

Стадия 1. Запуск СУБД, создание нового файла базы данных или открытие созданной ранее базы.

Стадия 2. Создание исходной таблицы или таблиц.

Создавая исходную таблицу, необходимо указать имя и тип каждого поля. Имена полей не должны повторяться внутри одной таблицы. В процессе работы с БД можно дополнять таблицу новыми полями. Созданную таблицу необходимо сохранить, дав ей имя, уникальное в пределах создаваемой базы.

При проектировании таблиц, рекомендуется руководствоваться следующими основными принципами:

1. Информация в таблице не должна дублироваться. Не должно быть повторений и между таблицами. Когда определенная информация хранится только в одной таблице, то и изменять ее придется только в одном месте. Это делает работу более эффективной, а также исключает возможность несовпадения информации в разных таблицах. Например, в одной таблице должны содержаться адреса и телефоны клиентов.

2. Каждая таблица должна содержать информацию только на одну тему. Сведения на каждую тему обрабатываются намного легче, если они содержатся в независимых друг от друга таблицах. Например, адреса и заказы клиентов лучше хранить в разных таблицах, с тем, чтобы при удалении заказа информация о клиенте осталась в базе данных.

3. Каждая таблица должна содержать необходимые поля. Каждое поле в таблице должно содержать отдельные сведения по теме таблицы. Например, в таблице с данными о клиенте могут содержаться поля с названием компании, адресом, городом, страной и номером телефона. При разработке полей для каждой таблицы необходимо помнить, что каждое поле должно быть связано с темой таблицы. Не рекомендуется включать в таблицу данные, которые являются результатом выражения. В таблице должна присутствовать вся необходимая информация. Информацию следует разбивать на наименьшие логические единицы (Например, поля "Имя" и "Фамилия", а не общее поле "Имя").

4. База данных должна иметь первичный ключ. Это необходимо для того, чтобы СУБД могла связать данные из разных таблиц, например, данные о клиенте и его заказы.

Стадия 3. Создание форм.

Первоначально необходимо указать таблицу, на базе которой будет создаваться форма. Ее можно создавать при помощи мастера форм, указав, какой вид она должна

иметь, или самостоятельно. При создании формы можно указывать не все поля, которые содержит таблица, а только некоторые из них. Имя формы может совпадать с именем таблицы, на базе которой она создана. На основе одной таблицы можно создать несколько форм, которые могут отличаться видом или количеством используемых из данной таблицы полей. После создания форму необходимо сохранить. Созданную форму можно редактировать, изменяя местоположение, размеры и формат полей.

Стадия 4. Заполнение БД.

Процесс заполнения БД может проводиться в двух видах: в виде таблицы и в виде формы. Числовые и текстовые поля можно заполнять в виде таблицы, а поля типа MEMO и OLE – в виде формы.

VI этап. Работа с созданной базой данных.

Работа с БД включает в себя следующие действия:

- поиск необходимых сведений;
- сортировка данных;
- отбор данных;
- вывод на печать;
- изменение и дополнение данных.

6. Подведение итогов урока

4 урок

4. Изучение нового материала

Табличные базы данных. Типы связей между таблицами.

В реляционной базе данных связи позволяют избежать избыточности данных. Например, в ходе создания базы данных, содержащей сведения о книгах, может появиться таблица под названием "Книги", в которой будут храниться параметры каждой книги, такие как ее название, дата публикации и издатель. Кроме того, есть и дополнительные сведения об издателе, которые может потребоваться сохранить, такие как его телефонный номер, адрес и почтовый индекс. Если хранить их в таблице с книгами, то телефонный номер издателя будет повторяться для каждой опубликованной им книги.

Более правильным вариантом является вынесение сведений об издателях в отдельную таблицу "Издатели". При этом таблица "Книги" будет содержать ссылки на записи таблицы "Издатели".

Чтобы сохранить синхронизацию, следует обеспечить целостность данных между таблицами "Книги" и "Издатели". Связи с обеспечением целостности данных позволяют следить за тем, чтобы данные в одной таблице соответствовали данным в

другой. Например, каждая книга в таблице "Книги" связана с определенным издателем в таблице "Издатели". Добавить в таблицу книгу для издателя, отсутствующего в базе данных, невозможно.

Связь осуществляется путем сопоставления данных в ключевых столбцах; обычно это столбцы, имеющие в обеих таблицах одинаковые названия. В большинстве случаев сопоставляются первичный ключ одной таблицы, содержащий для каждой из строк уникальный идентификатор, и внешний ключ другой таблицы. Например, с каждым из изданий, находящихся в продаже, можно связать объемы его продаж путем создания столбца "ИД_издания" в таблице "Книги" (первичный ключ) и столбца "ИД_издания" в таблице "Продажи" (внешний ключ).

Существует три типа связей между таблицами:

Связь «один-ко-многим»

При такой связи каждой строке таблицы А может соответствовать множество строк таблицы Б, однако каждой строке таблицы Б может соответствовать только одна строка таблицы А.

Рассмотрим базу данных, в которой фиксируются заказы, включающую таблицы «Клиенты» и «Заказы». Клиент может разместить любое количество заказов. Следовательно, у любого клиента, представленного в таблице «Клиенты», может быть много заказов, представленных в таблице «Заказы». Поэтому между таблицами «Клиенты» и «Заказы» существует отношение «один-ко-многим».

Связь «многие-ко-многим»

При такой связи каждой строке таблицы А может соответствовать множество строк таблицы Б и наоборот.

Рассмотрим отношение между таблицей «Продукты» и таблицей «Заказы». Один заказ может включать несколько продуктов. С другой стороны, отдельный продукт может содержаться в нескольких заказах. Следовательно, для каждой записи таблицы «Заказы» может существовать много записей в таблице «Продукты», и наоборот. Поэтому такой тип отношения называется «многие-ко-многим». Обратите внимание, что для обнаружения существующего отношения между таблицами важно рассмотреть обе его стороны.

Связь «один-к-одному»

При такой связи каждой строке таблицы А может соответствовать только одна строка таблицы Б и наоборот. Отношения этого типа используются нечасто, поскольку обычно сведения, связанные таким образом, хранятся в одной таблице. Отношение «один-к-одному» используется для разделения таблицы, содержащей много полей, с целью отделения части таблицы по соображениям безопасности, а также с целью сохранения сведений, относящихся к подмножеству записей в основной таблице. После определения такого отношения у обеих таблиц должно быть общее поле.

6. Подведение итогов урока

Выставление оценок.

2, 3 урок

5. Закрепление изученного материала

Практическая работа: *Создание табличной базы данных. Создание связей.* (Методические рекомендации по выполнению практических работ в программе Microsoft Access / Лабораторная работа №1, №2).

6. Подведение итогов урока

Проверка выполненных заданий. Выставление оценок.