

Содержание

ВВЕДЕНИЕ	3
Система управления базами данных (СУБД)	4
Основные объекты СУБД.....	5
Табличные базы данных.....	5
Создание межтабличных связей.....	7
Формы.....	7
Проектирование форм и работа с ними.....	9
Запросы.....	13
Отчеты.....	18
Список использованных источников.....	25
Приложение.....	26

ВВЕДЕНИЕ

Приложение Microsoft Access – это настольная система управления реляционными базами данных (СУБД), предназначенная для работы на автономном персональном компьютере (ПК) или локальной вычислительной сети под управлением семейства операционных систем Microsoft Windows. СУБД Microsoft Access обладает мощными, удобными и гибкими средствами визуального проектирования объектов с помощью Мастеров, что позволяет пользователю при минимальной предварительной подготовке довольно быстро создать полноценную информационную систему на уровне таблиц, запросов, форм и отчетов

В пособие изложены основные понятия, используемые при работе с базами данных и работа с базами данных в СУБД Access.. Описаны разработка таблиц, создание запросов на выборку и изменение данных, перекрестных запросов. Рассмотрено создание форм, запросов, отчетов.

Приведен пример создания простейшей реляционной базы «Ученик»

Учебно-методическое пособие предназначено для I-II курсов студентов Борского государственного техникума.,

СИСТЕМА УПРАВЛЕНИЯ БАЗАМИ ДАННЫХ (СУБД)

Развитие информационных технологий привело к созданию компьютерных баз данных. Создание баз данных, а так же операции поиска и сортировки данных выполняются специальными программами – *системами управления базами данных(СУБД)*. Таким образом необходимо различать собственно базы данных (БД), которые являются упорядоченным набором данных, и системы управления базами данных (СУБД) – программы, управляющие хранением и обработкой данных.

Система управления базами данных (СУБД) — это программа, позволяющая создавать базы данных, а также обеспечивающая обработку (сортировку) и поиск данных.

В СУБД Microsoft Access используется стандартный для операционных систем много-оконный интерфейс, но в отличие от других приложений, не многодокументный. Единственно может быть открыта только одна база данных, содержащая обязательное окно базы данных и окна для работы с объектами базы данных. В каждый момент времени одно из окон является активным и в нем курсором отмечается активный объект.

Базы данных

Базы данных представляют собой информационные модели, содержащие данные об объектах и их свойствах. Базы данных хранят информацию о группах объектов с одинаковым набором свойств.

Например, база данных «Записная книжка» хранит информацию о людях, каждый из которых имеет фамилию, имя, телефон и т. д. Библиотечный каталог хранит информацию о книгах, каждая из которых имеет название, автора, год издания и т.д

Информация в базах данных хранится в упорядоченном виде. Так, в записной книжке все записи упорядочены по алфавиту, а в библиотечном каталоге либо по алфавиту (алфавитный каталог) или по области знания (предметный каталог).

ОСНОВНЫЕ ОБЪЕКТЫ СУБД

Каждой СУБД могут реализовываться свои типы объектов. К основным объектам СУБД относятся: таблицы, формы, запросы, отчеты

Таблицы

В базах данных вся информация хранится в двумерных таблицах. Это базовый объект базы данных, все остальные объекты создаются на основе существующих таблиц (производные объекты). Каждая строка в таблице – запись базы данных, а столбец — поле. Запись содержит набор данных об одном объекте, а поле — однородные данные обо всех объектах.

Запросы

В СУБД запросы являются важнейшим инструментом. Главное предназначение запросов — это отбор данных на основании заданных условий

Формы

Формы позволяют отображать данные, содержащиеся только в одной записи. При помощи форм можно добавлять в таблицы новые данные, а также редактировать или удалять существующие. Форма может содержать рисунки, графики и другие внедренные объекты

Отчеты

Они предназначены для печати данных, содержащихся в таблицах и запросах, в красиво оформленном виде

Контрольные вопросы:

1. Чем отличаются между собой БД и СУБД?
2. Какую функцию выполняют СУБД?
3. Какие основные объекты входят в СУБД и какие функции они выполняют?

ТАБЛИЧНЫЕ БАЗЫ ДАННЫХ

Таблицы являются основным объектом любой БД, который хранит все имеющиеся в базе данные, а также саму структуру базы (поля, их типы и свойства). Создание таблиц начинают с выбора элемента

Таблицы в окне БД. Таблицу можно создать, используя 2 режима

- с помощью Мастера или
- в режиме Конструктора таблиц (создание таблиц вручную).

При создании таблицы необходимо задавать ключевое поле, что будет способствовать в дальнейшем организации связей между таблицами. Ключевое поле создается с помощью команды Ключевое поле контекстного меню правой кнопки мыши. По окончании создания структуры таблицы при закрытии бланка система выдаст запрос на сохранение таблицы, в ответ на который таблице дают имя, и она становится доступна в основном окне БД.

Заполняют таблицу данными в обычном порядке. Для этого курсор ввода устанавливают в нужной ячейке указателем мыши. Переход к следующей ячейке выполняется нажатием клавиши TAB. Переход к очередной записи осуществляется после внесения данных в последнюю ячейку. Ширину столбцов таблицы можно изменять путем перетаскивания их границ. Удобнее воспользоваться автоматическим форматированием столбцов «по содержанию». После заполнения таблицы данными сохранять их не нужно — все сохранится автоматически. Однако в случае редактирования макета таблицы (к примеру, изменилась ширина столбцов) СУБД выдаст запрос на подтверждение сохранения изменений.

Внесение изменений в структуру таблицы (состава полей или их свойств) выполняется в режиме Конструктор

Элементы таблицы

Столбцы

Базу данных, хранящую данные о группе объектов с одинаковыми свойствами, удобно представлять в виде двумерной таблицы: в каждой ее строке последовательно размещаются значения свойств одного из объектов; каждое значение свойства — в своем столбце, озаглавленном именем свойства

Столбцы

Столбцы такой таблицы называют *полями*; каждое поле характеризуется своим именем (именем соответствующего свойства) и типом данных, представляющих значения данного свойства.

Поле базы данных — это столбец таблицы, содержащий значения определенного свойства.

Строки таблицы являются *записями* об объекте; эти записи разбиты на поля столбцами таблицы, поэтому каждая запись представляет набор значений, содержащихся в полях.

Запись базы данных — это строка таблицы, содержащая набор значений свойств, размещенный в полях базы данных

Каждая таблица должна содержать, по крайней мере, одно *ключевое поле*, содержание которого уникально для каждой записи в этой таблице. Ключевое поле позволяет однозначно идентифицировать запись в таблице.

Ключевое поле — это поле, значения которого однозначно определяют запись в таблице. В качестве ключевого поля чаще всего используют поле, содержащее тип данных счетчик. Однако иногда удобнее в качестве ключевого поля таблицы использовать другие поля, код объекта, например инвентарный номер и т.п.

Тип поля

Тип поля определяется типом данных, которые оно содержит. Поля могут содержать данные следующих основных типов:

- Счетчик. Содержит последовательность целых чисел, которые задаются автоматически при вводе записей. Эти числа не могут быть изменены пользователем
- Текстовый. Содержит символы различных типов

- Числовой. Содержит числа различных типов
- Дата/Время. Содержит даты или время
- Картинка. Содержит изображения
- Логический. Содержит значения Истина (Да) или Ложь (Нет)

Каждый тип поля имеет свой набор свойств. Наиболее важными свойствами полей являются

- Размер поля. Определяет максимальную длину текстового или числового поля
- Формат поля. Устанавливает формат данных
- Обязательное поле. Указывает на то, что данное поле обязательно надо заполнить

Контрольные вопросы:

1. В чем заключается разница между записью и полем в табличной базе данных?
2. Поля каких типов могут присутствовать в базе данных?
3. Чем отличается ключевое поле от остальных полей?

СОЗДАНИЕ МЕЖТАБЛИЧНЫХ СВЯЗЕЙ

При продуманной заранее структуре БД и намеченных связях между таблицами создать реляционные отношения между таблицами можно легко и просто. Вся необходимую работу можно выполнить в специальном окне Схема данных, открываемомся кнопкой панели инструментов или командой Сервис - Схема данных. Образованная межтабличная связь отобразится в окне Схема данных в виде линии, которая соединит 2 поля разных таблиц. Причем в соединении одна из таблиц будет главной, а другая — связанной.

С помощью этой связи можно:

- исключить вероятность удаления или изменения данных ключевого поля главной таблицы, если оно связано с какими-либо полями других таблиц;
- при удалении (или изменении) данных ключевого поля главной таблицы выполнить корректное автоматическое удаление или изменение соответствующих данных в полях связанных с ним таблиц.

ФОРМЫ

Являются средствами для ввода данных, предоставляющими пользователю нужные для заполнения поля. В них размещаются специальные элементы управления (списки, счетчики, флажки, переключатели и пр.) для автоматизации ввода. При выводе данных при помощи форм применяют специальные средства их оформления. Форма, как правило, повторяет внешний вид бланка, что приводит к снижению количества ошибок, упрощает работу

С помощью форм пользователи могут вводить данные в таблицы БД без прямого доступа к самим таблицам.

Различаются два вида формирования структуры форм на основе таблиц и на основе запроса, возможен комбинированный способ.

Различают 2 вида формирования структуры форм: на основе таблицы и на основе запроса, возможен комбинированный способ.

Автоформы

В отличие от таблиц и запросов, формирующихся ручным способом, формы удобнее готовить автоматизировано. Автоматическими в полном объеме являются средства, которые называют автоформами.

Различают три вида автоформ:

- «в столбец» - предназначена для отображения всех полей одной записи, удобна для ввода и редактирования данных;
- «ленточная» - используется для отображения одновременно группы записей, ее удобно использовать при оформлении вывода данных; табличная автоформа внешне схожа с таблицей, на которой она основана.

Автоформа, как правило, основана лишь на 1 объекте. Другие средства создания форм позволяют строить форму на основе нескольких таблиц или запросов. Если форма основана лишь на одном объекте, ее называют простой формой. Если форма основана на полях из нескольких связанных таблиц, то ее называют сложной.

Форма имеет 3 основных раздела, это области заголовка, данных и примечания. Линии между разделами можно перетаскивать по вертикали, используя мышь, что позволит изменить размеры разделов.

Разделы заголовка и примечания используются в основном для оформления. В разделе данных представлены элементы управления, с помощью которых выполняется отображение данных или их ввод. Разработчик формы может размещать в этом разделе дополнительные элементы управления, способствующие автоматизации ввода данных.

Записи базы данных можно просматривать и редактировать в виде *таблицы* или в виде *формы*. Выше мы работали с базой данных, представленной в виде таблицы, когда запись образует строку в этой таблице

Такое представление базы данных позволяет наблюдать несколько записей одновременно, и в этом состоит достоинство табличного представления

Однако часто вид Таблица не позволяет видеть полностью всю информацию на экране. Если база данных содержит достаточно много полей, а значения полей содержат много символов, то не все поля таблицы могут уместиться на экране, а значения полей могут быть видны не полностью

Форма отображает одну запись в удобном для пользователя виде. В процессе создания формы можно указать, какие поля базы данных включить в форму, как расположить поля в окне формы, а также как можно сделать форму визуально привлекательной

Фактически с помощью формы создается графический интерфейс доступа к базе данных, который может содержать различные *элементы управления* (текстовые поля, кнопки, переключатели и т. д.), а также надписи. Обычно на форме размещаются надписи, являющиеся именами полей базы данных, и текстовые поля, содержащие данные из базы данных

Пользователь может изменять *дизайн* формы (размер, цвет и т. д.), элементов управления и надписей

..Примерами форм могут являться Визитка в базе данных «Записная книжка» или Карточка в базе данных «Библиотечный каталог», которые содержат лишь одну запись базы данных, зато представленную в удобном для пользователя виде

Контрольные вопросы

В чем заключается разница между представлением табличной базы данных с помощью таблицы и формы?

ПРОЕКТИРОВАНИЕ ФОРМ И РАБОТА С НИМИ

Access предоставляет возможность вводить данные как непосредственно в таблицу, так и с помощью форм. Форма в БД - это структурированное окно, которое можно представить так, чтобы оно повторяло форму бланка. Формы создаются из набора отдельных элементов управления.

Внешний вид формы выбирается в зависимости от того, с какой целью она создается. Формы Access позволяют выполнять задания, которые нельзя выполнить в режиме таблицы. Формы позволяют вычислять значения и выводить на экран результат. Источником данных для формы являются записи таблицы или запроса.

Форма предоставляет возможности для:

ввода и просмотра информации базы данных;

- изменения данных;
- печати;
- создания сообщений.

Способы создания форм:

1. Конструктор форм (предназначен для создания формы любой сложности).
2. Мастер форм (позволяет создавать формы различные как по стилю, так и по содержанию).
3. Автоформа: в столбец (многостраничная – поля для записи выводятся в один столбец, в форме одновременно отображаются данные для одной записи).
4. Автоформа: ленточная (все поля записи выводятся в одну строку, в форме отображаются все записи).
5. Автоформа: табличная (отображение записей осуществляется в режиме таблица).
6. Автоформа: сводная таблица.
7. Автоформа: сводная диаграмма.
8. Диаграмма (создается форма с диаграммой, построенной Microsoft Graph).
9. Сводная таблица (создается форма Access, отображаемая в режиме сводной таблицы Excel).
- 10.

Алгоритм создания форм следующий:

1. Открыть окно БД.
2. В окне БД выбрать вкладку Формы.

3. Щелкнуть на пиктограмме Создать, расположенной на панели инструментов окна БД.
4. В появившемся диалоговом окне «Новая форма» Выбрать способ создания формы и источник данных.
5. Щелкнуть на кнопке ОК.

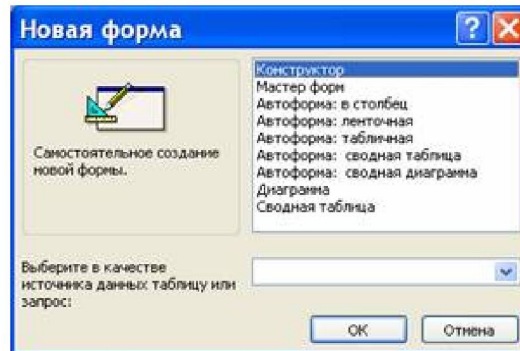


Рис. 1.

Создание формы с помощью Мастера

Вызвать Мастер форм можно несколькими способами. Один из них – выбрать Мастер форм в окне диалога Новая форма и щелкнуть на кнопке ОК. Откроется окно диалога Создание форм, в котором необходимо отвечать на вопросы каждого текущего экрана Мастера и щелкать на кнопке Далее.

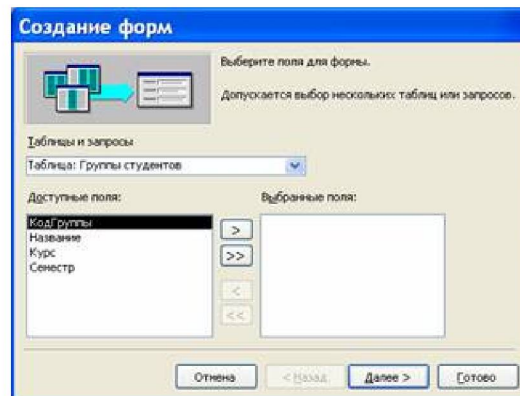


Рис. 2.

В первом окне необходимо выбрать поля из источника данных (таблиц или запросов). Для этого надо открыть список Таблицы и запросы, щелкнув на кнопку, справа. Затем доступные поля требуется перевести в Выбранные поля, выделив их и щелкнув на кнопку >>.

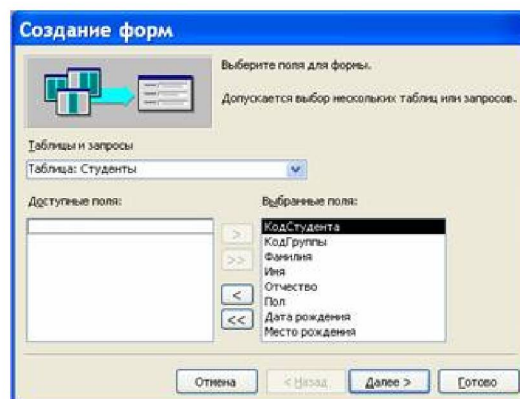


Рис. 3.

Например, выберем источник – таблицу Студенты и все ее поля, а затем необходимо щелкнуть на кнопке Далее.

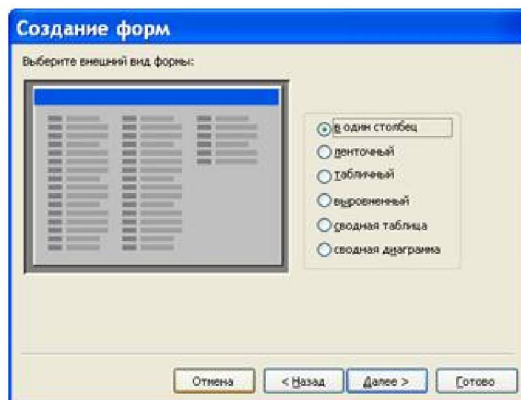


Рис. 4.

В этом окне надо выбрать внешний вид формы, например в один столбец и щелкнуть Далее.

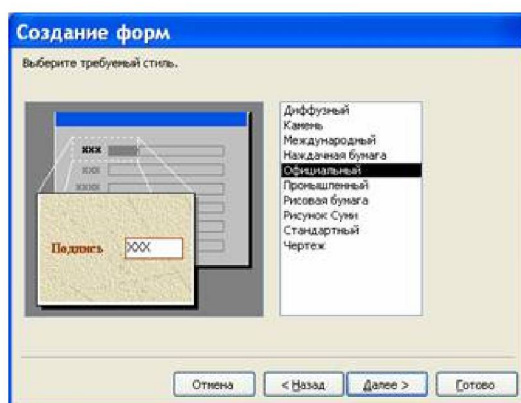


Рис. 5.

После выбора стиля формы (например, официальный), требуется перейти в последнее окно, щелкнув на кнопке Далее. В последнем окне Мастера требуется ввести имя формы и указать дальнейшие действия: Открыть форму для просмотра и ввода данных; Изменить макет формы.

После ввода имени формы (например, Студенты), выбора режима: «Открыть форму для просмотра и ввода данных» и щелчка на кнопке Готово, получим следующую форму для ввода и просмотра записей в таблицу Студенты.

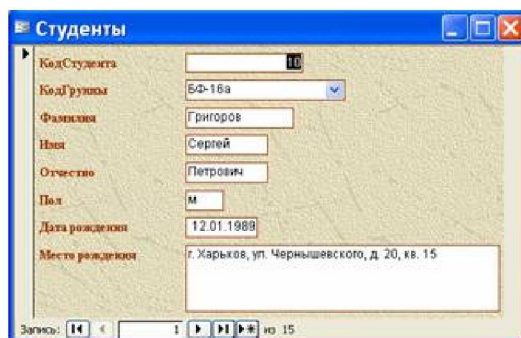


Рис. 6.

Создание формы с помощью Конструктора

Для создания формы Студенты необходимо выполнить следующие действия:

1. Запустить программу Microsoft Access и открыть БД.
2. В окне БД выбрать вкладку Формы. Выполнить щелчок по кнопке Создать. Появится диалоговое окно Новая форма. В этом окне необходимо выбрать из списка пункт Конструктор. Затем в списке "Выберите в качестве источника данных таблицу или запрос" выбрать имя таблицы (например, Студент). Выполнить щелчок по кнопке ОК. На экране появится окно Форма 1.

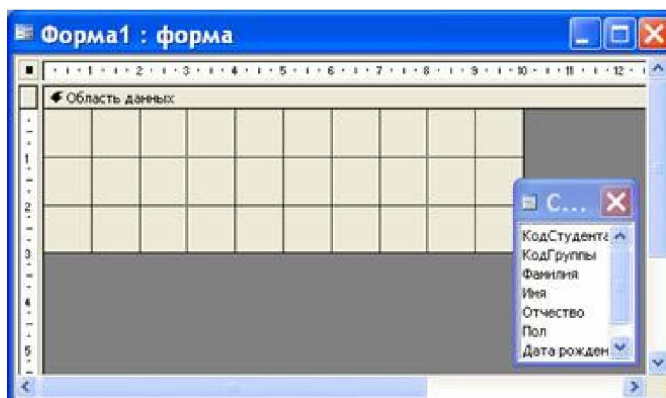


Рис. 7.

3. Если на экране отсутствует список полей выбранной для построения формы таблицы, выбрать пункт меню Вид / Список полей.
4. Поля из списка переместить на форму (по одному или предварительно выделив с использованием клавиши Shift и мыши, для выделения всех полей выполнить двойной щелчок мышью на заголовке окна Список полей).
5. Разместить поля на форме в нужных местах по разработанному образцу.
6. Перемещение полей и их имен по форме производиться следующим образом:
 - выделить объект (поле с именем) щелчком мыши. Вокруг него появятся маркеры перемещения и изменения размеров. Перемещать поле можно вместе с привязанным к нему именем или отдельно от него;
 - для перемещения поместить указатель мыши на квадратик, находящийся в левом верхнем углу элемента. Указатель мыши в виде ладони позволяет перемещать объект вместе с привязанным к нему именем, в виде ладони с вытянутым указательным пальцем - перемещает один объект;
 - нажать кнопку мыши и, удерживая ее, буксировать поле или его имя в нужное место в форме. Затем отпустить кнопку мыши;
 - для изменения надписи, связанной с полем необходимо выполнить на ней двойной щелчок мышью. В открывшемся диалоговом окне Надпись выбрать вкладку Макет и выполнить необходимые изменения. Затем закрыть окно;
 - для изменения размеров поместить курсор на размерные маркеры, при этом курсор примет вид двунаправленной стрелки. Нажать кнопку мыши, буксировать в нужном направлении, затем отпустить кнопку мыши;
 - для удаления поля выделить его, нажать клавишу Delete или выбрать команду Правка / Удалить.
7. Сохранить форму, выбрав из меню Файл команду Сохранить как, и в открывшемся окне выбрать режим сохранения «в текущей базе данных», затем щелчок по кнопке ОК.

8. Просмотреть форму в режиме Конструктора, выполнив щелчок по кнопке Открыть.

Рис. 8.

9. Если вид формы не удовлетворяет, открыть форму в режиме Конструктор и внести необходимые изменения, затем сохранить форму Файл — Сохранить или выполнить щелчок по пиктограмме Сохранить.

Контрольные вопросы:

1. Для чего предназначены формы?
2. Какие методы создания форм вы знаете?
3. Из чего состоят формы?
4. Какие разделы форм вы знаете?
5. Какие элементы управления форм вы знаете?
6. При создании элемента управления формы рядом с ней автоматически образуется элемент управления – присоединенная надпись. Как оторвать эту надпись от элемента управления и поместить в другом месте?
7. Как можно выровнять элементы управления формы, расположенные неаккуратно?
8. Как называются элементы управления формы, позволяющие выбирать данные из заранее подготовленного списка?
9. Можно ли использовать формы для вывода данных? Если да, то на какое устройство ЭВМ выполняется этот вывод?

ЗАПРОСЫ

Используются при извлечении данных из таблиц и предоставлении их в удобном для пользователя виде. С их помощью отбираются данные, сортируются и фильтруются. Кроме того, можно преобразовывать данные по заданному алгоритму, создавать новые таблицы, автоматически заполнять таблицы данными из других источников, выполнять простейшие вычисления в таблицах и многое другое.

Особенностью запросов является то, что они используют данные базовых таблиц и создают на их основе временную результирующую таблицу (моментальный снимок), которая представляет собой образ отобранных из базовых таблиц полей и записей. Работа с образом намного быстрее и эффективнее, чем с таблицами на жестком диске, так как время доступа к БД, хранящейся на винчестере гораздо больше времени

доступа к запросам в оперативной памяти. Обновление БД также можно осуществить с помощью запросов. В базовые таблицы данные вносят в порядке поступления, т.е. они не упорядочены. Однако, используя соответствующий запрос, можно получить отсортированные и отфильтрованные данные. В запрос вносятся лишь требуемые поля из общей таблицы, поэтому просмотр результата значительно упрощается.

Виды запросов

Различают следующие виды запросов:

1. Запрос по образцу. Данный запрос разрабатывается с помощью специального языка SQL (Structured Query Language). Однако большинство операций выполняются щелчком кнопки мыши и путем перетаскивания в бланке.

2. Запрос с параметрами. Специальный тип запросов, позволяющий пользователю самому ввести условие отбора данных на этапе запуска запроса. Этот прием обеспечивает гибкость работы с БД.

3. . Итоговый запрос. Выполняет вычисление по всем записям для какого-либо числового поля. В этом запросе рассчитывается сумма значений или величина среднего значения по всем ячейкам поля, выбирается максимальное или минимальное значение данных в поле, исполняется любая другая итоговая функция. Подготовка итогового запроса, как и запроса на выборку, выполняется с помощью бланка запроса по образцу.

4. Перекрестный запрос. С помощью него создаются результирующие таблицы на основе результатов расчетов, полученных при анализе группы таблиц.

5. Данный запрос отображает результаты расчетов, выполненных по данным из одного поля таблицы. Полученные результаты группируются по двум наборам данных, один из которых расположен в левом столбце таблицы, а второй в верхней строке.

6. Запросы на изменение. Данные запросы за 1 операцию вносят изменения в несколько записей.

1. Запросы на изменения бывают 4 типов:

2. На удаление – предназначен для удаления группы записей из одной или нескольких таблиц. На обновление записей – используется для внесения общих изменений в группу записей одной или нескольких таблиц.

3. На добавление – предназначен для добавления группы записей из одной или нескольких таблиц в конец одной или нескольких других таблиц.

4. На создание таблиц – используется для создания новой таблицы на основе всех или части данных из одной или нескольких таблиц.

Поиск записей с помощью запросов

Запросы осуществляют поиск записей в табличной базе данных так же, как и фильтры. Различие между ними состоит в том, что запрос является самостоятельным объектом базы данных, а фильтр привязан к конкретной таблице

Запрос является производным объектом от таблицы. Однако результатом выполнения запроса является также таблица, т. е. запросы могут использоваться вместо таблиц. Например, форма может быть создана как для таблицы, так и для запроса

Запросы позволяют отобрать те записи, которые удовлетворяют заданным условиям. Запросы, как и фильтры, бывают простые и сложные. *Простой запрос* содержит одно условие, а *сложный запрос* содержит несколько условий для различных полей

В процессе создания запроса можно отбирать не только записи, но и поля, которые будут присутствовать в запросе

Контрольные вопросы:

В чем заключается разница между поиском записей в табличной базе данных с помощью фильтров и запросов?

Создание запроса на выборку с помощью Мастера

При создании запроса необходимо определить:

- поля в базе данных, по которым будет идти поиск информации;
- предмет поиска в базе данных;
- перечень полей в результате выполнения запроса.

В окне база данных выбрать вкладку Запросы и дважды щелкнуть на пиктограмме Создание запроса с помощью мастера, появится окно Создание простых запросов.

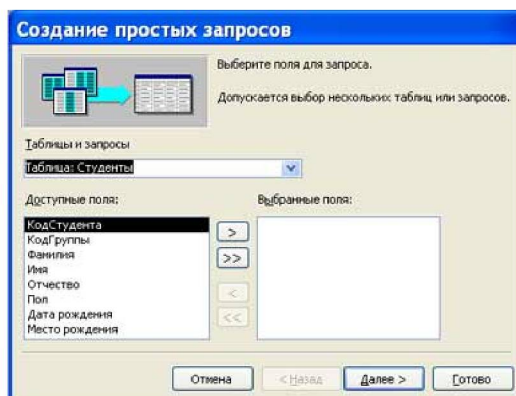


Рис. 1.

В окне мастера выбрать необходимую таблицу (таблицу - источник) из опции Таблицы и запросы и выбрать поля данных. Если формируется на основе нескольких таблиц, необходимо повторить действия для каждой таблицы – источника.

Затем в окне Мастера надо выбрать подробный или итоговый отчет и щелкнуть на кнопке Далее. После этого необходимо задать имя запроса и выбрать один из вариантов дальнейшего действия: Открыть запрос для просмотра данных или Изменить макет запроса и нажать кнопку Готово.

Создание запроса на выборку с помощью Конструктора

С помощью конструктора можно создать следующие виды запросов:

1. Простой.
2. По условию.
3. Параметрические.
4. Итоговые.
5. С вычисляемыми полями.

Чтобы вызвать Конструктор запросов, необходимо перейти в окно базы данных. В окне база данных необходимо выбрать вкладку Запросы и дважды щелкнуть на пиктограмме Создание запроса в режиме конструктора. Появится активное окно Добавление таблицы на фоне неактивного окна «Запрос: запрос на выборку».

В окне Добавление таблицы следует выбрать таблицу – источник или несколько таблиц из представленного списка таблиц, на основе которых будет проводиться выбор данных, и щелкнуть на кнопке Добавить. После этого закрыть окно Добавление таблицы, окно «Запрос: запрос на выборку» станет активным.

Окно Конструктора состоит из двух частей – верхней и нижней. В верхней части окна размещается схема данных запроса, которая содержит список таблиц – источников и отражает связь между ними.

В нижней части окна находится Бланк построения запроса QBE (Query by Example), в котором каждая строка выполняет определенную функцию:

- Поле – указывает имена полей, которые участвуют в запросе.
- Имя таблицы – имя таблицы, с которой выбрано это поле.
- Сортировка – указывает тип сортировки.
- Вывод на экран – устанавливает флажок просмотра поля на экране.
- Условия отбора - задаются критерии поиска.
- Или – задаются дополнительные критерии отбора.

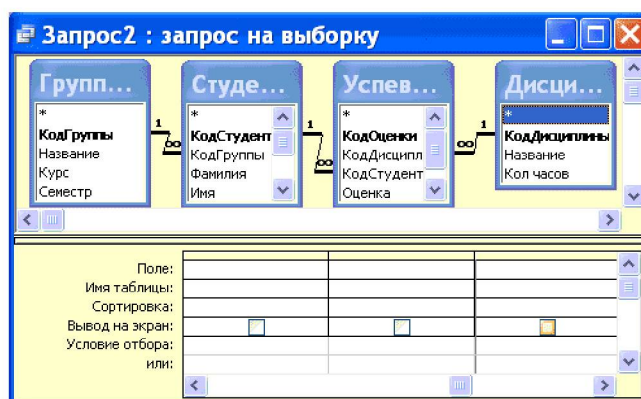


Рис. 2.

Запрос на выборку

В окне «Запрос: запрос на выборку» с помощью инструментов формируем query:

Выбрать таблицу – источник, из которой производится выборка записей.

1. Переместить имена полей с источника в Бланк запроса. Например, из таблицы Группы отбуксировать поле Название в первое поле Бланка запросов, из таблицы Студенты отбуксировать поле Фамилии во второе поле Бланка запросов, а из таблицы Успеваемость отбуксировать поле Оценка в третье поле и из таблицы Дисциплины отбуксировать поле Название в четвертое поле Бланка запросов.

2. Задать принцип сортировки. Курсор мыши переместить в строку Сортировка для любого поля, появится кнопка открытия списка режимов сортировки: по возрастанию и по убыванию. Например, установить в поле Фамилия режим сортировки – по возрастанию.

3. В строке вывод на экран автоматически устанавливается флажок просмотра найденной информации в поле.
4. В строке "Условия" отбора и строке "Или" необходимо ввести условия ограниченного поиска – критерии поиска. Например, в поле Оценка ввести - "отл/А", т.е. отображать все фамилии студентов, которые получили оценки отл/А.
5. После завершения формирования запроса закрыть окно Запрос на выборку. Откроется окно диалога Сохранить – ответить Да (ввести имя созданного запроса, например, Образец запроса в режиме Конструктор) и щелкнуть ОК и вернуться в окно базы данных.

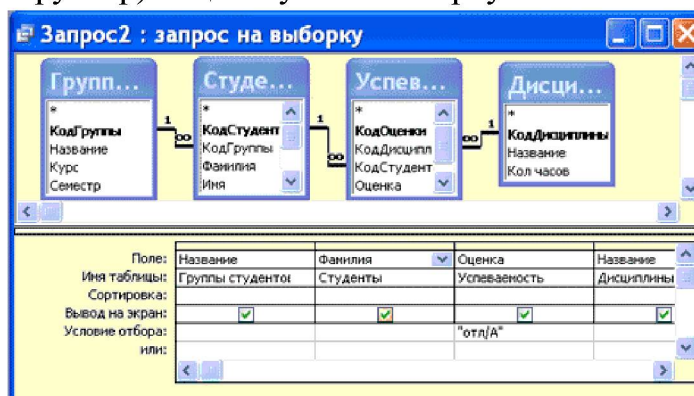


Рис. 3.

Чтобы открыть query из окна базы данных, необходимо выделить имя запроса и щелкнуть кнопку Открыть, на экране появится окно запрос на выборку с требуемым именем.

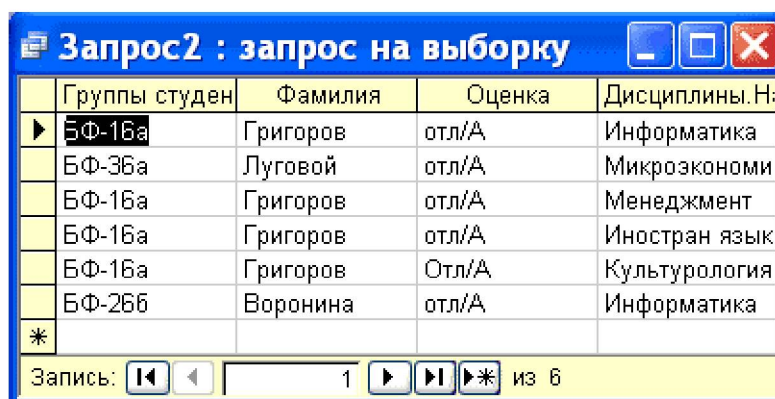


Рис. 4.

Чтобы внести изменения в query его необходимо выбрать щелчком мыши в окне базы данных, выполнить щелчок по кнопке Конструктор, внести изменения. Сохранить запрос, повторить его выполнение.

Параметрические запросы

Запросы, представляющие собой варианты базового запроса и незначительно отличающиеся друг от друга, называются параметрическими. В параметрическом запросе указывается критерий, который может изменяться по заказу пользователя.

Последовательность создания параметрического запроса:

1. Создать query в режиме конструктора или открыть существующий запрос в режиме конструктора, например «Образец запроса в режиме Конструктор».
2. В Бланк запроса в строке Условия отбора ввести условие отбора в виде приглашения в квадратных скобках, например [Введите фамилию].

3. Закрывать окно Запрос на выборку, на вопрос о сохранении изменения ответить – Да. Вернуться в окно базы данных, где созданный query будет выделен.
4. Выполнить query, щелкнув по кнопке: Открыть. В появившемся на экране окне диалога «Введите значение параметра» надо ввести, например фамилию студента, информацию об успеваемости которого необходимо получить, выполнить щелчок по кнопке ОК.

Контрольные вопросы:

1. Для чего предназначены запросы?
2. Запросы баз данных обращены к таблицам. Как называются таблицы, на базе которых основан запрос?
3. Как называется таблица, полученная в результате работы запроса?
4. Как называется категория запросов, предназначенная для выбора данных из таблиц баз данных?
5. Как называется бланк, заполнением которого создается структура запроса?
6. Как называется поле, данные в котором образуются в результате расчета с участием данных, содержащихся в других полях?
7. Как называется категория запросов, выполняющих итоговые вычисления?
8. Какие итоговые функции вы знаете?
9. Как внести в бланк запроса по образцу строку для команды группировки?
10. Как сделать несколько итоговых вычислений по одному полю?
11. Как с помощью запроса провести сортировку записей в таблице по какому-либо полю?

ОТЧЕТЫ

Используются при выводе данных на принтер. В отчетах предусмотрены специальные меры для группировки выводимых данных и для вывода специальных элементов оформления для печатных документов (колоннотитулы, номера страниц и др.). В отличие от других объектов СУБД используются только для вывода данных на печать, а не на экран.

Отчеты многим схожи с формами и страницами доступа к данным, однако имеют другое функциональное назначение — применяются для форматированного вывода данных на печатающее устройство, поэтому они должны учитывать параметры принтера и бумаги.

В отчетах, как и в формах, предусмотрены средства автоматического, автоматизированного и ручного проектирования. Средства автоматического проектирования реализуются с помощью автоотчетов (База данных - Создать - Новый отчет - Автоотчет в столбец). Также существуют «ленточные» автоотчеты.

Автоматизированное создание отчетов осуществляется с помощью Мастера отчетов, который запускается двойным щелчком по значку Создание отчета с помощью мастера в окне База данных. Мастер отчетов выполняет работу в 6 этапов.

С помощью него осуществляются выбор:

1. базовых таблиц или запросов, на которых базируется отчет;
2. полей отчета; полей группировки; полей и методов сортировки;
3. формы печатного макета и стиля оформления.

В структуре готового отчета в отличии от структуры формы присутствует увеличенное количество разделов. Помимо разделов заголовка, примечания и данных в отчете могут содержаться разделы верхнего и нижнего колонтитулов. В отчете объемом более 1 страницы данные разделы необходимы для печати служебной информации (например, номеров страниц). Чем больше по объему отчет, тем важнее роль данных, выводимых на печать в этих разделах. Если для каких-либо полей отчета выполнена группировка, количество разделов отчета увеличится, так как оформление заголовков групп будет выполняться в отдельных разделах.

Редактировать структуру отчета можно в режиме Конструктора. Приемы редактирования такие же, что и у форм. Важная особенность отчетов заключается в наличии средства для вставки в область верхнего или нижнего колонтитула текущего номера страницы и полного количества страниц. Эту операция выполняется в диалоговом окне Номера страниц (Вставка - Номера страниц)

Создание отчета как объекта базы данных

Отчет – это форматированное представление данных, которое выводится на экран, в печать или файл. Они позволяют извлечь из базы нужные сведения и представить их в виде, удобном для восприятия, а также предоставляют широкие возможности для обобщения и анализа данных.

При печати таблиц и запросов информация выдается практически в том виде, в котором хранится. Часто возникает необходимость представить данные в виде отчетов, которые имеют традиционный вид и легко читаются. Подробный отчет включает всю информацию из таблицы или запроса, но содержит заголовки и разбит на страницы с указанием верхних и нижних колонтитулов.

Структура отчета в режиме Конструктора

Microsoft Access отображает в отчете данные из запроса или таблицы, добавляя к ним текстовые элементы, которые упрощают его восприятие.

К числу таких элементов относятся:

- 1. Заголовок.** Этот раздел печатается только в верхней части первой страницы отчета. Используется для вывода данных, таких как текст заголовка отчета, дата или констатирующая часть текста документа, которые следует напечатать один раз в начале отчета. Для добавления или удаления области заголовка отчета необходимо выбрать в меню Вид команду Заголовок/примечание отчета.
- 2. Верхний колонтитул.** Используется для вывода данных, таких как заголовки столбцов, даты или номера страниц, печатающихся сверху на каждой странице отчета. Для добавления или удаления верхнего колонтитула необходимо выбрать в меню Вид команду Колонтитулы. Microsoft Access добавляет верхний и нижний колонтитулы одновременно. Чтобы скрыть один из колонтитулов, нужно задать для его свойства Высота значение 0.
- 3. Область данных,** расположенная между верхним и нижним колонтитулами страницы. Содержит основной текст отчета. В этом разделе появляются данные, распечатываемые для каждой из тех записей в таблице или запросе, на которых основан отчет. Для размещения в области данных элементов управления используют список полей и панель элементов. Чтобы скрыть область данных, нужно задать для свойства раздела Высота значение 0.

4. Нижний колонтитул. Этот раздел появляется в нижней части каждой страницы. Используется для вывода данных, таких как итоговые значения, даты или номера страницы, печатающихся снизу на каждой странице отчета.

5. Примечание. Используется для вывода данных, таких как текст заключения, общие итоговые значения или подпись, которые следует напечатать один раз в конце отчета. Несмотря на то, что в режиме Конструктора раздел "Примечание" отчета находится внизу отчета, он печатается над нижним колонтитулом страницы на последней странице отчета. Для добавления или удаления области примечаний отчета необходимо выбрать в меню Вид команду Заголовок/примечание отчета. Microsoft Access одновременно добавляет и удаляет области заголовка и примечаний отчета.

Способы создания отчета

В Microsoft Access можно создавать отчеты различными способами:

Конструктор.

1. Мастер отчетов.
2. Автоотчет: в столбец.
3. Автоотчет: ленточный.
4. Мастер диаграмм.
5. Почтовые наклейки.

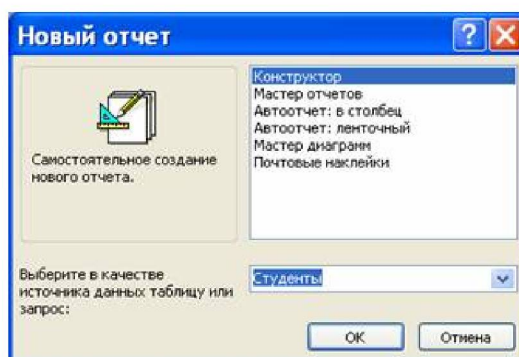


Рис. 1.

Мастер позволяет создавать отчеты с группировкой записей и представляет собой простейший способ создания отчетов. Он помещает выбранные поля в отчет и предлагает шесть стилей его оформления. После завершения работы Мастера полученный отчет можно доработать в режиме Конструктора. Воспользовавшись функцией Автоотчет, можно быстро создавать отчеты, а затем вносить в них некоторые изменения.

Для создания Автоотчета необходимо выполнить следующие действия:

1. В окне базы данных щелкнуть на вкладке Отчеты и затем щелкнуть на кнопке Создать. Появится диалоговое окно Новый отчет.
2. Выделить в списке пункт Автоотчет: в столбец или Автоотчет: ленточный.
3. В поле источника данных щелкнуть на стрелке и выбрать в качестве источника данных таблицу или запрос.
4. Щелкнуть на кнопке ОК.
5. Мастер автоотчета создает автоотчет в столбец или ленточный (по выбору пользователя), и открывает его в режиме Предварительного просмотра, который позволяет увидеть, как будет выглядеть отчет в распечатанном виде.

6. В меню Файл щелкнуть на команде Сохранить. В окне Сохранение в поле Имя отчета указать название отчета и щелкнуть на кнопке ОК.

Изменение масштаба отображения отчета

Для изменения масштаба отображения пользуются указателем — лупой. Чтобы увидеть всю страницу целиком, необходимо щелкнуть в любом месте отчета. На экране отобразится страница отчета в уменьшенном масштабе.

Снова щелкнуть на отчете, чтобы вернуться к увеличенному масштабу отображения. В увеличенном режиме представления отчета, точка, на которой вы щелкнули, окажется в центре экрана. Для пролистывания страниц отчета пользуются кнопками перехода вниз окна.

Печать отчета

Для печати отчета необходимо выполнить следующее:

1. В меню Файл щелкнуть на команде Печать.
2. В области Печатать щелкнуть на варианте Страницы.
3. Чтобы напечатать только первую страницу отчета, введите 1 в поле "с" и 1 в поле "по".
4. Щелкнуть на кнопке ОК.

Прежде чем печатать отчет, целесообразно просмотреть его в режиме Предварительного просмотра, для перехода к которому в меню Вид нужно выбрать Предварительный просмотр.

Если при печати в конце отчета появляется пустая страница, убедитесь, что параметр Высота для примечаний отчета имеет значение 0.

Если при печати пусты промежуточные страницы отчета, убедитесь, что сумма значений ширины формы или отчета и ширины левого и правого полей не превышает ширину листа бумаги, указанную в диалоговом окне Параметры страницы (меню Файл).

При разработке макетов отчета руководствуйтесь следующей формулой: ширина отчета + левое поле + правое поле \leq ширина бумаги.

Для того чтобы подогнать размер отчета, необходимо использовать следующие приемы:

- изменить значение ширины отчета;
- уменьшить ширину полей или изменить ориентацию страницы.

Создание отчета

1. Запустите программу Microsoft Access. Откройте БД (например, учебную базу данных «Деканат»).
2. Создайте Автоотчет: ленточный, используя в качестве источника данных таблицу (например, Студенты). Отчет открывается в режиме Предварительного просмотра, который позволяет увидеть, как будет выглядеть отчет в распечатанном виде

Студенты

КодСтудента	КодГруппы	Фамилия	Имя	Отчество	Пол	Дата рождения	Место рождения
10	БС-10а	Григорян	Саркис	Петрович	м	12.07.1989	г. Харьков
11	БС-10а	Красноба	Ирина	Васильевна	ж	29.09.1989	г. Харьков
12	БС-10б	Ильин	Илья	Ильич	м	13.03.1989	с. Красное Волчанского р-на Харьковской обл.
13	БС-20а	Полтава	Эль	Владимировна	ж	01.05.1989	г. Днепродзержинск
14	БС-20б	Васильева	Светлана	Ивановна	ж	29.09.1989	г. Люботин Харьковской обл.
15	БС-30а	Лукьян	Иван	Федорович	м	17.09.1989	с. Красное Купчанынского р-на Днепродзержинск
16	БС-30б	Семельников	Александр	Владимирович	м	12.02.1989	с. Покровское Богодуховского р-на Харьковской обл.
17	БС-40а	Милышев	Николай	Владимирович	ж	14.06.1989	г. Полтава
18	БС-50а	Саргисян	Василия	Павлович	ж	09.04.1989	г. Харьков
19	БС-40б	Саргисян	Вера	Александровна	ж	30.09.1989	г. Киев
20	БС-20б	Корень	Владимир	Леонович	м	31.03.1989	г. Запорожье
21	БС-30в	Ковалько	Григорий	Григорьевич	м	07.07.1989	г. Ужгород
22	БС-20в	Семельков	Елена	Михайловна	ж	10.10.1989	с. Васильевка Харьковской обл. Харьковской области
23	БС-40в	Присодило	Николай	Михайлович	ж	22.11.1989	г. Харьков
24	БС-10б	Иванова	Иван	Иванович	м	20.09.1989	г. Киев

25 сентября 2009 г.

Страница 2 из 2

Рис. 2.

3. Перейдите в режим Конструктора и выполните редактирование и форматирование отчета. Для перехода из режима предварительного просмотра в режим конструктора необходимо щелкнуть команду Закрывать на панели инструментов окна приложения Access. На экране появится отчет в режиме Конструктора.

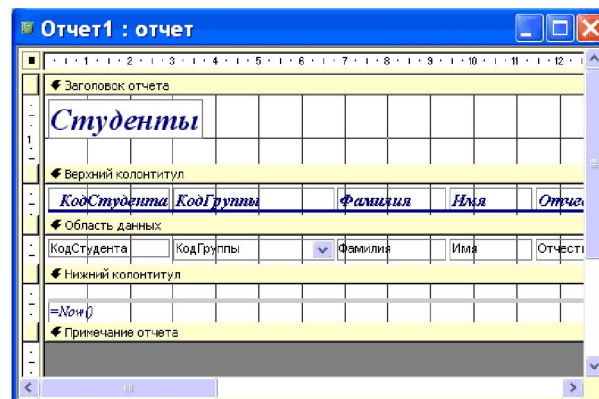


Рис. 3.

Редактирование отчета

Для редактирование отчета необходимо выполнить следующие действия:

1. удалите поля код студента в верхнем колонтитуле и области данных.
2. переместите влево все поля в верхнем колонтитуле и области данных.
3. Измените надпись в заголовке страницы:
 - в разделе Заголовок отчета выделить надпись Студенты;
 - поместите указатель мыши справа от слова Студенты, так чтобы указатель принял форму вертикальной черты (курсора ввода), и щелкните в этой позиции;
 - введите НТУ «ХПИ» и нажмите Enter.
4. Переместите Надпись. В Нижнем колонтитуле выделить поле =Now() и перетащить его в Заголовок отчета под название Студенты. Дата будет отображаться под заголовком.
5. На панели инструментов Конструктор отчетов щелкнуть на кнопке Предварительный просмотр, чтобы просмотреть отчет

Форматирование отчета

Алгоритм форматирования отчета:

1. Выделите заголовок Студенты НТУ «ХПИ».
2. Измените гарнитуру, начертание и цвет шрифта, а также цвет заливки фона.
3. На панели инструментов Конструктор отчетов щелкнуть на кнопке Предварительный просмотр, чтобы просмотреть отчет.

Студенты НТУ "ХПИ"

ИдГруппы	Фамилия	Имя	Отчество	Пол	Дата рождения	Место рождения
БФ-14а	Григорьев	Сергей	Петрович	м	13.01.1989	г. Харьков
БФ-14а	Красовая	Наталья	Васильевна	ж	25.05.1989	г. Харьков
БФ-14б	Ильин	Илья	Ильич	м	13.03.1989	с. Красное Ведмежское р-на Харьковской обл.
БФ-24а	Петрова	Юли	Викторовна	ж	01.01.1989	г. Днепропетровск
БФ-24б	Вороженин	Степан	Николаевич	м	29.04.1989	г. Лубовщина Харьковской обл.
БФ-34а	Луговой	Николай	Федорович	м	17.08.1989	с. Купчино Славянского р-на Донецкой обл.
БФ-34б	Савицкий	Александр	Владимирович	м	12.02.1989	с. Петровское Бельцкого района Харьковской обл.
БФ-44а	Матвеев	Евгений	Борисович	м	14.04.1989	г. Полтава
БФ-44а	Сорокина	Виктория	Павловна	ж	08.04.1989	г. Харьков
БФ-44б	Сорокина	Евгений	Александрович	м	30.05.1989	г. Киев
БФ-24б	Королев	Владимир	Леонидович	м	31.08.1989	г. Чапурово
БФ-34а	Кабачков	Георгий	Георгиевич	м	07.07.1989	г. Ужгород
БФ-34а	Сажин	Евгений	Михайлович	м	10.10.1989	с. Виново Харьковской р-на Харьковской области
БФ-44а	Прохорова	Наталья	Михайловна	ж	22.11.1989	г. Харьков
БФ-14б	Николаев	Николай	Николаевич	м	20.09.1989	г. Киев

Страница 1 из 1

Рис. 4.

Изменение стиля

Для изменения стиля выполните следующее:

1. На панели инструментов Конструктора отчетов щелкнуть на кнопке Автоформат, откроется диалоговое окно Автоформат.
2. В списке Стили объекта "отчет - автоформат" щелкнуть на пункте Строгий и затем щелкнуть на кнопке ОК. Отчет будет отформатирован в стиле Строгий.
3. Переключится в режим Предварительный просмотр. Отчет отобразится в выбранном вами стиле. Впредь все отчеты созданные с помощью функции Автоотчет будут иметь стиль Строгий, пока вы не зададите другой стиль в окне Автоформат.
4. Сохранить и закрыть отчет.

Контрольные вопросы:

1. Для чего предназначены отчеты?
2. На какое устройство происходит вывод данных при работе отчета?
3. Из чего состоят отчеты?
4. Какие разделы отчетов вы знаете?
5. Что общего и в чем различие между разделами отчетов и форм?
6. Какие способы автоматической генерации отчетов вы знаете?
7. На базе чего выполняется отчет? Что задается при создании автоотчета?
8. Что такое экспорт и импорт таблиц?
9. Как произвести экспортирование таблиц базы данных?
10. В какие программы можно переместить таблицы?
11. Из каких программ можно вставить объект в таблицу базы данных?

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Семакин, И. Г. Информатика и ИКТ. Базовый уровень. 10—11 классы: Методическое пособие / И. Г. Семакин, Е. К. Хеннер. — М.: БИНОМ. Лаборатория знаний, 2008, 102с.: ил. ISBN 978-5-94774-743-5
2. Свиридова М. Ю. Система управления базами данных Access : учеб. пособие для нач. проф. образования / М. Ю. Свиридова. — М. : Издательский центр «Академия», 2010. — 192 с. ISBN 978-5-7695-4664-8
3. Проектирование баз данных. СУБД Microsoft Access. Учебное пособие /Н. Н. Гринченко, Е. В. Гусев, Н. П. Макаров. – Горячая Линия – Телеком, 2004, ISBN: 5-93517-193-7

Интернет-ресурсы

http://www.lessons-tva.info/edu/e-inf2/m2t4_9.html

https://spravochnick.ru/bazy_dannyh/sistema_upravleniya_bazami_dannyh_-_subd/osnovnye_obekty_subd_tablicy_formy_zaprosy_otchety/ .

ПРИМЕР РАЗРАБОТКИ БАЗЫ ДАННЫХ «УЧЕНИК»

Разработка БД начнется с проектирования. Так как MS Access использует реляционную модель данных, информацию расположим в трех следующих таблицах — *Классы*, *Классный руководитель*, *Ученики*:

<i>Классы</i>	<i>Классный руководитель</i>	<i>Ученики</i>
Номер класса	Номер классного руководителя	Номер ученика
Название класса	Фамилия	Фамилия
Профиль	Телефон	Возраст
Количество учеников	Название класса	Домашний адрес
	Наличие почетного звания	Название класса

Так как мы рассматриваем небольшой пример, то естественно, что в таблицах отражена только небольшая часть реальной информации. Например, в таблице *Ученики* следовало бы внести не только фамилии учеников, но и имена, и другие личные данные.

Эти три таблицы должны образовать единую систему, поэтому они взаимосвязаны:



Здесь присутствуют два типа связи: “один к одному” и “один ко многим”.

Таблицы *Классы* и *Классный руководитель* объединяет связь типа “один к одному”, она на рисунке обозначена одинарной стрелкой. При этом типе связи две таблицы можно было бы объединить в одну таблицу, так как одной записи в одной таблице соответствует одна запись в другой таблице. Информация разделена на две таблицы, так как одна таблица содержала бы слишком много полей и с ней было бы неудобно работать. Эти таблицы находятся на одном уровне иерархии.

Таблицы *Классы* и *Ученики* объединяет связь типа “один ко многим”, она обозначена двойной стрелкой. Например, в одном классе обучается несколько учеников, но каждый ученик может обучаться только в одном классе. Поэтому одной записи в таблице *Классы* соответствует несколько записей в таблице *Ученики*. Это связь между соседними уровнями иерархической структуры.

Приступим к созданию БД в приложении MS Access. После запуска программы и создания файла новой БД следует построить структуру таблиц и заполнить таблицы данными.

Структуру таблиц можно задать с помощью Конструктора. На рисунке показано окно Конструктора при определении структуры таблиц *Классы*:

Классы : таблица		
	Имя поля	Тип данных
	Номер класса	Числовой
☒	Название класса	Текстовый
	Профиль	Текстовый
	Количество учеников	Числовой

Изображение ключа указывает на ключевое поле. Аналогично определим структуры таблиц *Классный руководитель* и *Ученик*:

Классный руководитель : таблица		
	Имя поля	Тип данных
	Номер классн_рук	Числовой
	Фамилия	Текстовый
	Телефон	Числовой
🔑	Название класса	Текстовый
	Наличие почетного_зван	Логический

Обратим внимание на то, что в таблице *Классный руководитель* поле **Наличие почетного_зван** имеет логический тип.

Ученик : таблица		
	Имя поля	Тип данных
🔑	Номер Ученика	Числовой
	Фамилия	Текстовый
	Возраст	Числовой
	Дом_адрес	Текстовый
	Название класса	Текстовый

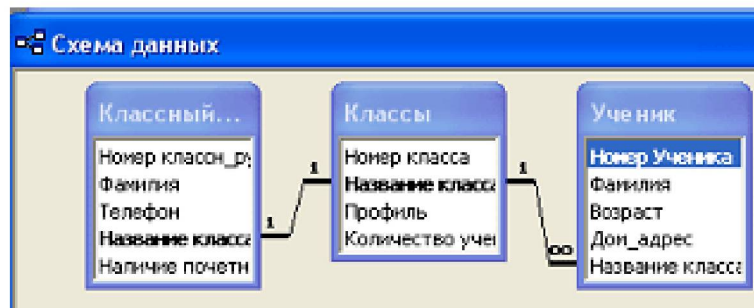
Заполним таблицы данными:

На следующем этапе определим связи между таблицами, используя **Схему данных**:

Классы : таблица				
	Номер класса	Название класса	Профиль	Количество учеников
▶ +	1	1а	гуманитарный	25
+	2	1б	математический	24
+	3	1в	общеобразовательный	25

Классный руководитель : таблица					
	Номер классн_	Фамилия	Телефон	Название клас	Наличие почет
+	1	Иванова	234566	1а	<input checked="" type="checkbox"/>
+	2	Павловская	443218	1б	<input type="checkbox"/>
+	3	Перова	134560	1в	<input checked="" type="checkbox"/>

Ученик : таблица					
	Номер Ученика	Фамилия	Возраст	Дом_адрес	Название клас
	1	Климов		7 ул.Ленина 27,1	1а
	2	Арбузов		7 ул.Юрева 101,	1а
	3	Серов		6 пр.Декабристо	1б
	4	Николаева		7 пр.Комсомольс	1б
	5	Красина		7 ул.Лесная 2	1в
	6	Петров		6 Ул.Лесная 17	1в



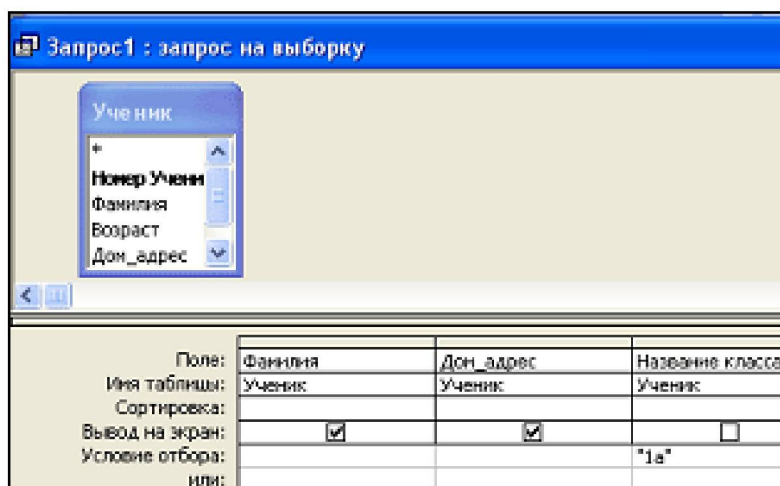
Здесь отражены два типа связи. Связь “один к одному” между таблицами *Классы* и *Классный руководитель* производится через общий главный ключ “Название класса”. Связь “один ко многим” между таблицами *Классы* и *Ученик* создана с помощью главного ключа “Название класса” таблицы *Классы* и одноименного поля таблицы *Ученик*.

Программа MS Access, обеспечивая целостность данных, препятствует удалению и изменению связанных данных. Изменение ключевого поля в первичной таблице приведет к автоматическому изменению ключей в связанных записях. Удаление записей из первичной таблицы вызовет соответствующее удаление записей в связанной таблице. Для этого при создании связи следует установить флажки

Каскадное обновление связанных данных и Каскадное удаление связанных данных.

Для поиска информации в БД создают запросы. Запрос позволяет манипулировать данными БД: выбирать данные по некоторому условию, сортировать, обновлять, удалять и добавлять данные в БД. Простейшим является запрос на выборку. Результатом запроса на выборку будет таблица, которая содержит поля, удовлетворяющие условиям отбора.

В нашем примере Запрос 1 позволяет получить список учеников с домашними адресами, которые обучаются в классе 1а. Создадим этот запрос в конструкторе, в качестве исходной таблицы укажем таблицу *Ученик*, в строке **Условие отбора** в поле **Название класса** зададим условие “1а”:



Далее получим результат выборки в таблице:

Запрос1 : запрос на выборку	
Фамилия	Дом адрес
Климов	ул.Ленина 27,1
Арбузов	ул.Юрева 101,

Второй запрос выводит фамилии классных руководителей, которые не имеют почетного звания. Он основан на таблице *Классный руководитель*. Поле **Наличие почетного зван** имеет логический тип, поэтому в строку **Условие отбора** внесена константа **Нет**:

Поле:	Фамилия	Наличие почетног
Имя таблицы:	Классный руковод	Классный руковод
Сортировка:		
Вывод на экран:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Условие отбора:		Нет
или:		

Результат выборки:

Запрос2 : запрос на выборку	
Фамилия	Наличие почет
Павловская	<input type="checkbox"/>

Запрос может быть многотабличным. Запрос 3 выводит поля **Название класса**, **Профиль** из таблицы *Классы*, и поле **Фамилия** из таблицы *Классный руководитель*, при этом поставлено условие, что класс имеет математический профиль.

Поле:	Название класса	Профиль	Фамилия
Имя таблицы:	Классы	Классы	Классный руковод
Сортировка:			
Вывод на экран:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Условие отбора:		"математический"	

Ответ можно посмотреть в режиме таблицы:

Запрос 3 : запрос на выборку		
Название класса	Профиль	Фамилия
16	математический	Павловская

В заключение нашего примера приготовим отчет. Отчеты предназначены только для вывода информации на печать. Напечатаем фамилии учеников и их домашние адреса, основываясь на таблице **Ученик**. Выберем создание отчета с помощью мастера и, следуя его указаниям, получим следующий отчет:

<i>Ученик-отчет</i>	
<i>Фамилия</i>	<i>Дом_адрес</i>
<i>Арбузов</i>	<i>ул.Юрлова 101,17</i>
<i>Климов</i>	<i>ул.Ленина 27,14</i>
<i>Красина</i>	<i>ул.Лесная 2</i>
<i>Николаева</i>	<i>пр.Комсомольской 23,45</i>
<i>Петров</i>	<i>Ул.Ленина 17</i>
<i>Серов</i>	<i>пр.Декабристов 12,86</i>