

ЛАБОРАТОРНАЯ РАБОТА №1. Системы управления базами данных MS ACCESS: работа с таблицами, создание связей между таблицами.

Цель лабораторной работы: Знакомство с пакетом Microsoft Office Access. Приобретение навыков и умений при работе с таблицами: создавать таблицы с помощью конструктора, задавать тип данных, задавать маску ввода для поля, использовать свойства поля, создавать ключи и индексы для полей таблицы, заполнять таблицы данными.

Методические указания

База данных представляет собой организованную структуру, в которой хранятся разнообразные сведения в какой-либо предметной области. Под предметной областью понимают обычно часть реального мира, подлежащую изучению для автоматизации управления. С базами данных чаще всего работает одновременно большое количество пользователей, имеющих различные права доступа. Комплекс программных средств, предназначенных для создания структуры новой базы, наполнения ее содержимым, редактирования данных, визуализации информации и определения прав пользователей, называется системой управления базой данных. В базе данных можно выполнять расчеты, однако эти вычисления выполняются не так быстро, как, например, в электронных таблицах. Основное назначение баз данных – это хранение и манипулирование данными. Большинство баз данных имеет табличную структуру. В пакет Microsoft Office входит система управления реляционными базами данных Access. Реляционная база данных представляет собой набор из нескольких таблиц, связанных между собой определенным образом. В Access понятие «база данных» относится к единичному файлу, содержащему всю информацию. База данных в Access обычно содержит следующие основные объекты:

1. Таблицы – содержат все хранящиеся в базе сведения, в табличном формате.
2. Запросы – существуют запросы на выборку, которые выбирают данные из одной или нескольких базовых таблиц по заданным условиям, а также запросы действия, которые влияют на содержимое базовых таблиц.
3. Формы – с их помощью можно вводить или отображать данные на основании описанного пользователем формата.
4. Отчеты – создаются на основе одной или нескольких таблиц или запросов и предназначены для вывода данных на печать.

Кроме того, в зависимости от версии Access, в базе могут присутствовать и другие дополнительные объекты. Чтобы связать таблицы между собой, в каждой из них предусматривают так называемое ключевое поле. Такое поле однозначно идентифицирует запись и содержит уникальные значения. Если в таблице нет поля, которое можно использовать в качестве ключевого, его создают искусственно, например, шифр клиента, номер лицевого счета и т.п.

Наиболее распространенные типы связей между таблицами в Access; «один-к-одному» и «один-ко-многим», когда связь осуществляется через общее поле, присутствующее в каждой из связываемых таблиц. Это поле всегда является ключевым на стороне «один». При связи «один-к-одному» каждой записи первой таблице соответствует единственная запись во второй, и наоборот. При связи «один-ко-многим» каждой записи первой таблицы может соответствовать несколько записей во второй, но запись второй таблицы не может иметь более одной соответствующей ей записи в первой. Существует также тип связи «многие-ко-многим», когда каждой записи в первой таблице может соответствовать несколько записей во второй, и наоборот. Такая связь является фактически

двойной связью «один-ко-многим» и организуется с помощью третьей, связующей, таблицы.

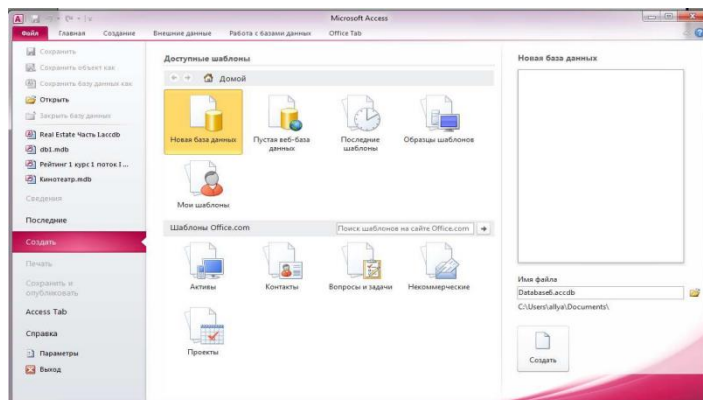


Рис. 1. Внешний вид MS Access

Базы данных Access работают со следующими типами данных. В таблице приведен список предусмотренных в приложении Microsoft Office Access 2010 типов данных, инструкции по их применению и сведения о размере места, необходимого для хранения данных каждого типа. При этом необходимо помнить, что максимальный размер файла базы данных Office Access 2010 составляет 2 гигабайта.

СОЗДАНИЕ БАЗЫ ДАННЫХ

Запустите Microsoft Access из меню Пуск или с помощью ярлыка. Появится представление **Backstage**. Выполните одну из указанных ниже процедур:

1. Создание веб-базы данных. В группе Доступные шаблоны щелкните элемент Пустая веб-база данных. Справа в разделе Пустая веб-база данных в поле Имя файла введите имя файла базы данных или используйте предоставленное имя. Нажмите кнопку Создать. Будет создана новая база данных и открыта новая таблица в режиме таблицы.

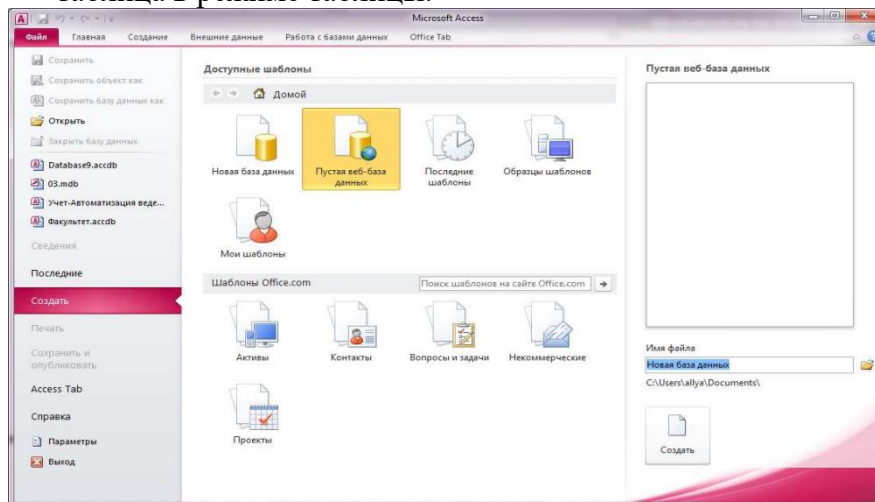


Рис. 2. Создание веб-базы данных

2. Создание базы данных на компьютере В группе Доступные шаблоны щелкните элемент Пустая база данных.

Справа в разделе Пустая база данных в поле Имя файла введите имя файла базы данных или используйте предоставленное имя. Нажмите кнопку Создать. Будет создана новая база данных и открыта новая таблица в режиме таблицы.

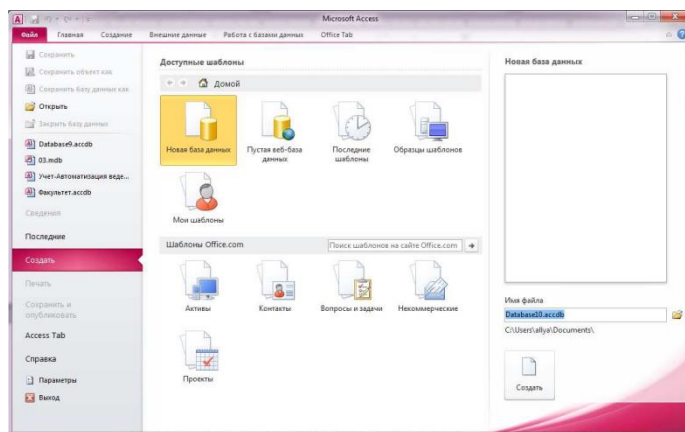


Рис. 3. Создание базы данных на компьютере

Access 2010 содержит ряд шаблонов, а дополнительные шаблоны можно загрузить с веб-сайта Office.com. Шаблон Access представляет собой готовую базу данных с профессионально разработанными таблицами, формами и отчетами. Шаблоны позволяют быстро пройти начальные этапы создания базы данных.

РАБОТА С ТАБЛИЦАМИ

Создание таблиц

При создании базы данных данные сохраняются в таблицах — списках строк (записей) и столбцов (полей), относящихся к конкретной области.

При создании новой базы данных автоматически создается новая таблица. Также таблицу можно создать с помощью добавления новой таблицы в существующую базу данных, импорта или создания ссылки на таблицу из другого источника данных, такого как книга Microsoft Office Excel, текстовый файл или другая база данных.

В режиме таблицы можно сразу же приступить к вводу данных, позволив приложению Access сформировать структуру таблицы автоматически. Имена полей задаются номерами («Поле1», «Поле2» и т. д.), а тип данных поля определяется на основе типа вводимых данных.

1. На вкладке Создание в группе Таблицы нажмите кнопку Таблица.
2. Приложение Access создаст таблицу и поместит курсор в первую пустую ячейку столбца Щелкните, чтобы добавить.
3. Чтобы добавить данные, начните вводить их в первую пустую ячейку или вставьте из другого источника, как будет показано ниже.

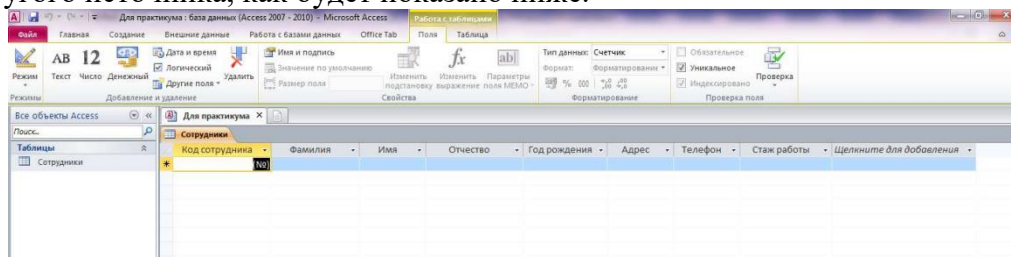


Рис. 4. Создание пустой таблицы в режиме таблицы

- Чтобы переименовать столбец (поле), дважды щелкните заголовок столбца и введите новое имя. Присвойте каждому полю понятное имя, чтобы можно было получить представление о его содержимом без просмотра данных.
- Чтобы переместить столбец, выделите его, щелкнув заголовок, а затем перетащите в новое место. Можно также выделить несколько смежных столбцов и перетащить их в новое место.
- Чтобы добавить в таблицу дополнительные поля, начните вводить данные в столбец Щелкните, чтобы добавить в режиме таблицы либо добавьте поля с помощью команд в группе Добавление и удаление на вкладке Поля.

После этого можно вводить данные прямо в таблицу. При ее сохранении Access анализирует данные и выбирает тот тип для каждого поля, который больше всего соответствует введенным данным.

Создание таблиц в режиме таблицы имеет ограниченное применение. Поля таблицы не имеют содержательных имен, так что почти всегда нужно будет редактировать структуру, чтобы переименовать поля. Кроме того, Access не всегда может правильно определить тип данных. Таблицы, которые созданы в режиме таблицы, не могут включать объекты OLE и Мемо-поля. Если есть желание иметь такие поля, вам нужно изменить структуру таблицы.

В режиме конструктора сначала следует создать структуру новой таблицы. Затем необходимо переключиться в режим таблицы для ввода данных либо ввести данные другим способом, например, с помощью формы. Режим конструктора недоступен для таблиц веб-баз данных.

1. На вкладке Создание в группе Таблицы нажмите кнопку Конструктор таблиц.

2. Для каждого поля в таблице введите имя в столбце Имя поля, а затем в списке Тип данных выберите тип данных.

3. Для каждого поля в столбец Описание можно ввести дополнительные сведения. Это описание отображается в строке состояния, если курсор находится в данном поле, и используется в качестве текста строки состояния всех элементов управления, создаваемых путем перетаскивания поля из области Список полей на форму или отчет, а также всех элементов управления, создаваемых для этого поля с помощью мастера форм или мастера отчетов.

4. Когда все необходимые поля будут добавлены, сохраните таблицу. Для этого выполните указанные ниже действия.

5. Чтобы начать вводить данные в таблицу, можно в любой момент времени переключиться в режим таблицы, щелкнуть первую пустую ячейку и начать ввод. Кроме того, можно вставить данные из другого источника (см. следующий раздел).

В режиме Конструктора таблиц мы видим то, что фактически является графическим бланком для создания и редактирования структуры таблиц. В первом столбце вводят имена полей. Тип для каждого поля выбирают из раскрывающегося списка, открываемого кнопкой выбора типа поля. Нижняя часть бланка содержит список свойств поля, выделенного в верхней части. Некоторые из свойств уже заданы по умолчанию. Свойства полей не являются обязательными. Их можно настраивать по желанию, а можно и не трогать.

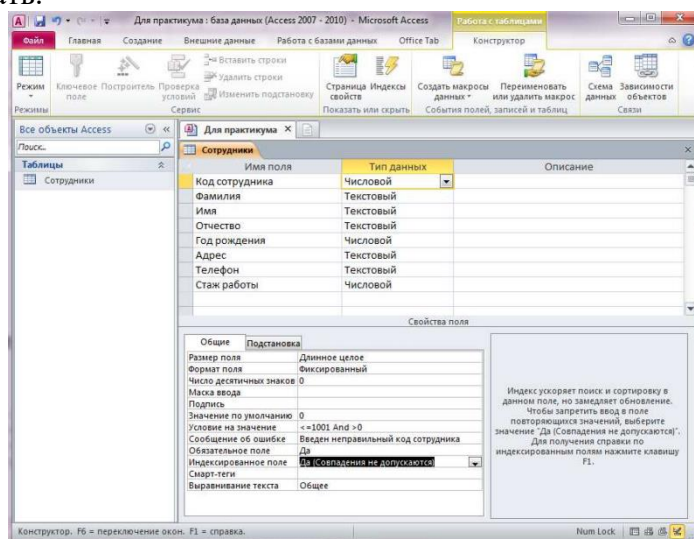


Рис. 5. Форма создания таблицы с помощью Конструктора

Закончив создание структуры таблицы, бланк закрывают (при этом система выдает запрос на сохранение таблицы), после чего дают таблице имя, и с этого момента она доступна в числе прочих таблиц в основном окне База данных.

Ввод данных в таблицу

Созданную таблицу можно открыть в представлении двойным щелчком на ее значке. Новая таблица не имеет записей — только названия столбцов, характеризующие структуру таблицы.

В нижней части таблицы расположена Панель кнопок перехода. Ее элементами управления удобно пользоваться при навигации по таблице, имеющей большое число записей.

Если возникнет необходимость изменить структуру таблицы (состав полей или их свойства), таблицу надо открыть в режиме Конструктора. Для этого ее следует выделить в окне База данных и щелкнуть на кнопке Конструктор.

Установка ключевых полей. Перед началом заполнения таблицы нужно определить ключевое поле, создать индексы и определить связи этой таблицы с другими таблицами базы данных. Ключ состоит из одного или нескольких полей, значения которых однозначно определяют каждую запись в таблице. Ключ необходим для связывания таблиц.

В большинстве случаев связывают ключевое поле одной таблицы с соответствующим ему полем (часто имеющим то же имя), которое называют полем внешнего ключа во второй таблице. Таблица, содержащая ключевое поле, называется главной, а таблица, содержащая внешний ключ — связанной. Хотя связь может быть и не по ключевым полям.

Если таблица никогда не будет использоваться в качестве главной, то ключ для нее определять не обязательно. Однако Access считает таблицы, у которых ключ не определен, подозрительными.

При открытии таких таблиц в режиме конструктора появляется диалоговое окно, сообщающее о том, что ключ таблицы не определен.

В Access можно выделить три типа ключевых полей:

- Ключевые поля счетчика - поле счетчика можно задать таким образом, чтобы при добавлении каждой записи в таблицу в это поле автоматически вносилось порядковое число. Указание такого поля в качестве ключевого является наиболее простым способом создания ключевых полей. Именно этот тип ключевого поля установлен по умолчанию.

- Простой ключ - если поле содержит уникальные значения, такие как коды или инвентарные номера, то это поле можно определить, как ключевое. Если выбранное поле содержит повторяющиеся или пустые значения, то оно не будет определено как ключевое. Для определения записей, содержащих повторяющиеся данные можно выполнить запрос на поиск повторяющихся записей. Если устранить повторы путем изменения значений невозможно, то следует либо добавить в таблицу поле счетчика и сделать его ключевым, либо определить составной ключ.

- Составной ключ. В случаях, когда невозможно гарантировать уникальность значений каждого поля, существует возможность создать ключ, состоящий из нескольких полей. Примером может служить база данных по складским запасам, в которой используются один основной и один или несколько вспомогательных инвентарных номеров.

Примечание. Если определить подходящий набор полей для составного ключа сложно, просто добавьте поле счетчика и сделайте его ключевым. Например, не рекомендуется определять ключ по полям «Имена» и «Фамилии», поскольку нельзя исключить повторения этой пары значений для разных людей.

Ключевые поля определяются на этапе создания таблицы или в режиме Конструктора на этапе редактирования. Для этого надо подвести курсор к заголовку поля, которое будет ключевым, затем выполнить команду Ключевое поле из меню Правка или вызвать правой кнопкой контекстное меню и выполнить ту же команду.

Создание связей между таблицами

Связь между таблицами позволяет:

- либо исключить возможность удаления или изменения данных в ключевом поле главной таблицы, если с этим полем связаны какие-либо поля других таблиц;

- либо сделать так, что при удалении (или изменении) данных в ключевом поле главной таблицы автоматически (и абсолютно корректно) произойдет удаление или изменение соответствующих данных в полях связанных таблиц. Связь между таблицами устанавливает отношения между совпадающими значениями в ключевых полях — обычно между полями разных таблиц.

Существует четыре вида отношений:

- Один к одному. При использовании отношения «один к одному» запись в таблице «А» (главной таблице) может иметь не более одной связанной записи в таблице «В» (связанной таблице) и наоборот. Этот тип связи используют не очень часто, поскольку такие данные могут быть помещены в одну таблицу. Связь с отношением «один к одному» используют для разделения очень широких таблиц или для отделения части таблицы по соображениям защиты. В таблицах связанных отношением «один к одному» ключевые поля должны быть уникальными.

- Один ко многим. Связь с отношением «один ко многим» является наиболее часто используемым типом связи между таблицами. В такой связи каждой записи в таблице «А» могут соответствовать несколько записей в таблице «В», а запись в таблице «В» не может иметь более одной соответствующей ей записи в таблице «А». В таблице «А» ключевое поле должно быть уникальным. Связь с отношением «один ко многим» таблиц «А» и «В» можно рассматривать как связь с отношением «многие к одному» таблиц «В» и «А».

- Многие к одному. При использовании отношения «многие к одному» одной записи в таблице «А» может соответствовать одна запись в таблице «В», а запись в таблице «В» может иметь более одной соответствующей ей записи в таблице «А». В таблице «А» ключевое поле должно быть уникальным. Связь с отношением «один ко многим» таблиц «А» и «В» можно рассматривать как связь с отношением «многие к одному» таблиц «В» и «А».

- Многие ко многим. При использовании отношения «многие ко многим» одной записи в таблице «А» могут соответствовать несколько записей в таблице «В», а одной записи в таблице «В» несколько записей в таблице «А». В этом случае не требуется уникальности ключевых полей таблиц «А» и «В». Этот вид отношения используется для связывания таблиц, имеющих только внешние ключи.

Связанные поля не обязательно должны иметь одинаковые имена, однако, они должны иметь одинаковые типы данных. Кроме того, связываемые поля типа «Числовой» должны иметь одинаковые значения свойства «Размер поля». Исключениями из этого правила являются поля счетчика с последовательной нумерацией, которые могут связываться с числовыми полями размера «Длинное целое».

Существует возможность создать связь между текстовыми полями разной длины, но это может вызвать сложности при создании запросов.

При активном окне открытой Базы данных нажмем на вкладку Работа с базами данных. В открывшемся меню нажмем Схема данных. Нажмем левой кнопкой мыши на свободном участке и выберем Добавить таблицу, выберем вкладку Таблицы и, нажимая кнопку Добавить, разместим в окне Схема данных все ранее созданные таблицы базы данных, которые отображены в окне Добавление таблицы. Затем нажмем кнопку Заккрыть. В результате в окне Схема данных будут представлены все таблицы базы данных со списками своих полей (рис. 6).

Для образования связи достаточно мышкой перетащить ключевое поле главной таблицы на поле внешнего ключа связанной таблицы. При этом появится окно Изменение связей.

Образовавшаяся межтабличная связь отображается в окне Схема данных в виде линии, соединяющей два поля разных таблиц. При этом одна из таблиц считается главной, а другая — связанной.

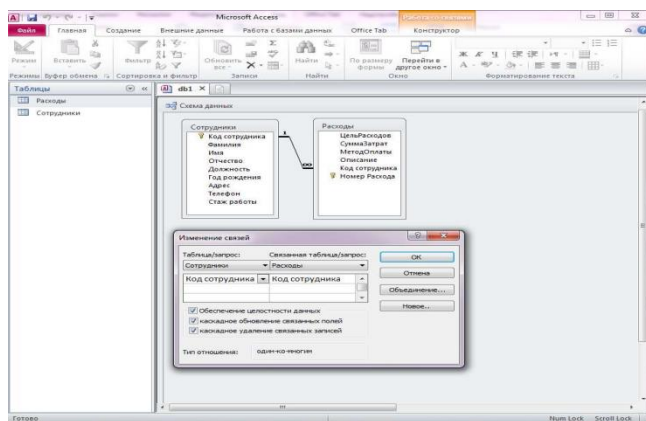


Рис.6. Окно «Схема данных»

Тип связи таблиц зависит от установленного индекса полей, участвующих в связи. Если поле ключа (его индекс всегда Совпадения не допускаются) является связующим полем в обеих таблицах, то тип связи будет один к одному, если в одной таблице, то один ко многим, если ни в одной из таблиц, то многие ко многим. На линии связи тип связи отображается с помощью специальных значков: один обозначается единицей, а многие - знаком бесконечности (∞).

Для настройки свойств связи надо в окне Схема данных выделить линию, соединяющую поля двух таблиц, щелкнуть на ней правой кнопкой мыши и открыть контекстное меню связи, после чего выбрать в нем пункт Изменить связь. В появившемся окне показаны названия связанных таблиц и имена полей, участвующих в связи, (здесь же их можно изменить), а также приведены элементы управления для обеспечения условий целостности данных.

Если установлен только флажок Обеспечение целостности данных, то Access не позволяет добавлять в связанную таблицу записи, для которых нет соответствующих записей в главной таблице, и изменять записи в главной таблице таким образом, что после этого в связанной таблице появятся записи, не имеющие главных записей; а также удалять записи в главной таблице, для которых имеются подчиненные записи в связанной таблице. Если вместе с ним включены флажки Каскадное обновление связанных полей и Каскадное удаление связанных записей, то, соответственно, операции редактирования и удаления данных в ключевом поле главной таблицы разрешены, но сопровождаются автоматическими изменениями в связанной таблице.

Пример 1. Создание однотабличной базы данных

1. Для создания новой базы данных: загрузите MS Access, в появившемся окне выберите пункт **Новая база данных**; Задайте имя вашей базы.

По умолчанию MS Access предлагает вам имя базы *db1* (Access 2007 – Database1.accdb), а тип файла - *Базы данных Access*. Имя задайте *Стоматология*.

2. Для создания таблицы базы данных: в окне базы данных выберите вкладку **Создание**, а затем щелкните по кнопке <Конструктор таблиц> (рис. 1).

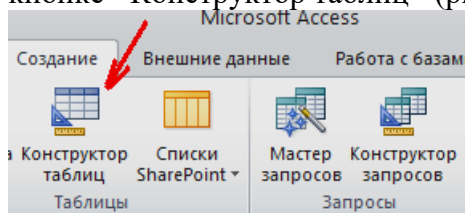


Рис. 1. Создание таблиц в режиме Конструктора

3. Для определения полей таблицы: введите в строку столбца **Имя поля** имя первого поля - *ID врача*. В строке столбца «Тип данных» щелкните по кнопке списка и выберите тип данных *Числовой*. Поля вкладки *Общие* оставьте такими, как предлагает Access (см.рис. 2).

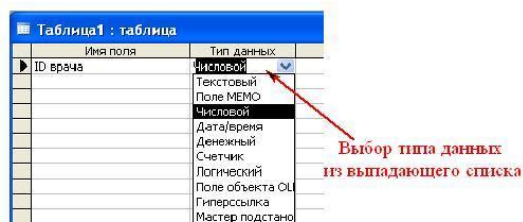


Рис. 2. Выбор типа данных

Для определения всех остальных полей таблицы базы данных *Врачи* выполните действия, аналогичные указанным выше в соответствии с табл. 1.

Таблица 1. Описание свойств полей таблицы *Врачи*

Имя поля	Тип данных	Размер поля
ID врача	Числовой	
ФИО	Текстовый	20
Должность	Текстовый	20
№ кабинета	Числовой	Длинное целое (по умолчанию)
Время приема	Текстовый	12
ВидОказываемыхУслуг	Текстовый	50 (Максимальное значение)

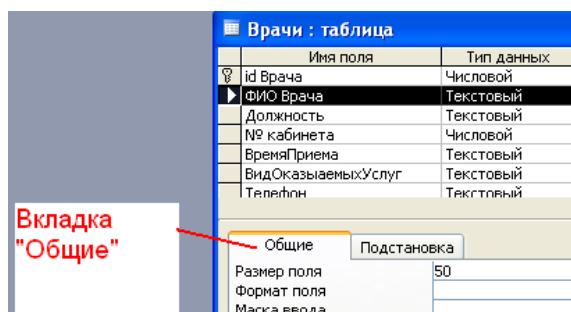
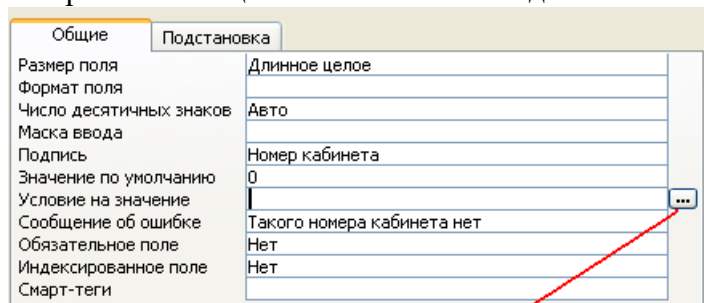


Рис. 3. Использование для поля дополнительных параметров с помощью вкладки «Общие»

4. Задайте условие для поля «№ кабинета».

Пусть по условию в поликлинике 2 этажа и значение номера кабинета находится в пределах от 100 до 130 и от 200 до 230. В строке «Условие на значение» перейдите в режим **Построителя выражений** (рис. 4) и введите условие: $\geq 100 \text{ And } \leq 130 \text{ Or } \geq 200 \text{ And } \leq 230$ (см. рис. 5).

В строке «Сообщение об ошибке» введите: «Такого номера кабинета нет».



Кнопка перехода в режим "Построителя выражений"

Рис. 4. Кнопка перехода в режим «Построителя выражений»

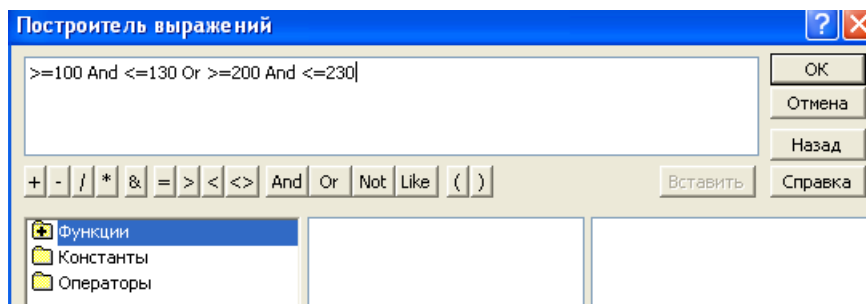


Рис. 5. Окно «Построителя выражений»

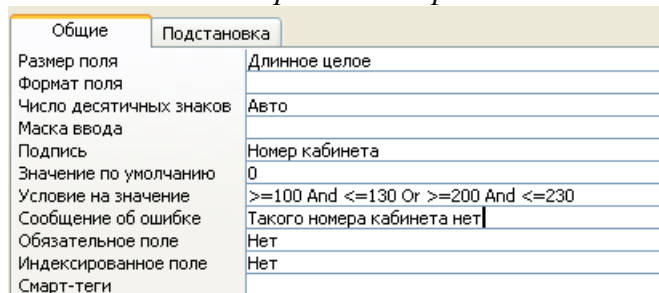


Рис. 6. Вкладка «Общие» с заданными условиями для поля «№ кабинета»

5. Создание ключевых полей.

Первичный ключ - это поле (или ряд полей), которое однозначно идентифицирует каждую запись в таблице. Access не допускает повторных значений в поле первичного ключа.

Сделайте поле **ID врача** ключевым. Для этого откройте таблицу *Врачи* в режиме **Конструктора** и, поместив курсор на имя этого поля, щелкните по кнопке - **Ключевое поле** (рис. 7):

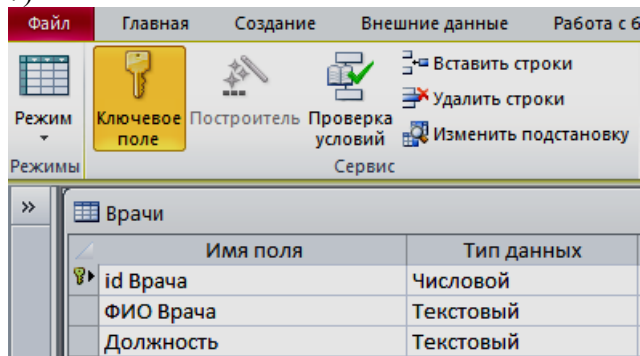


Рис. 7. Назначение поля ключевым

Закройте таблицу. MS Access предложит сохранить таблицу (по умолчанию «Таблица1»). Задайте имя таблицы «Врачи».

Пример 2. Заполнение базы данных.

- 1) Создать таблицы-подстановки.
- 2) Заполнить таблицы данными.
- 3) Создать подстановочное поле.
- 4) Ввести ограничения на данные в поле ID врача; эти данные не должны повторяться.

1. Создание таблиц-подстановок.

Для упрощения ввода данных в таблицу «Врачи» создайте таблицы-подстановки для полей: «ВремяПриема» и «ВидОказываемыхУслуг». Для поля «Должность» данные будут подставляться из таблицы «Штатное расписание».

Подстановка упрощает режим ввода. При этом нет необходимости набирать длинные названия несколько раз. Достаточно будет выбрать необходимую запись из выпадающего списка. Таблица «Смена-подстановка» (используется для поля «ВремяПриема»). Поля таблицы (см.табл.2):

Таблица 2.Описание свойств полей таблицы «Смена_подстановка»

Смена_подстановка : таблица	
Имя поля	Тип данных
НомерСмены	Текстовый
ВремяПриема	Текстовый

Примечание 1. В результате щелчка по кнопке <OK> MS Access предложит вам задать ключевое поле. Для создаваемой таблицы щелкните по кнопке <Нет>. Таблицу «НаименованиеУслуги_подстановка» (используется для поля «Вид оказываемых услуг») создайте в **Excel**, чтобы импортировать её в нашу базу данных. Поля таблицы (см.табл.3):

Таблица 3. Таблица с данными «НаименованиеУслуги_подстановка»

Наименование направления услуги
Терапевтическая
Хирургическая
Ортопедическая

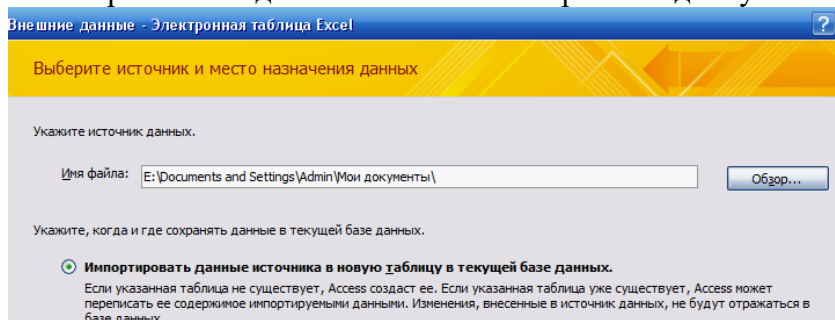
Импорт как самих таблиц, так и полей таблиц можно производить из разных источников и файлов разных форматов.

1. Определите формат ячеек: для поля *НаименованиеУслуги* – текстовый.

2. Сохраните таблицу в той же папке, где находится ваша БД *Стоматология*.

3. В базе данных *Стоматология* на панели инструментов выберите **Внешние данные Excel**.

4. В открывшемся диалоговом окне выберите созданную таблицу:



5. В следующем окне «Импорт электронной таблицы» выберите Лист 1, нажмите <Далее>.

6. Установите флажок «Первая строка содержит заголовки столбцов», далее: «Данные необходимо сохранить в новой таблице», далее подтвердите описание и название полей,

далее выберите «Не создавать ключ», Готово.

Таблица «НаименованиеУслуги_подстановка» будет импортирована в вашу БД. Посмотрите ее в режиме **Конструктора**. Все ли поля соответствуют **типу данных**?

Таблица «Штатное расписание» (данные поля «НаименованиеДолжности» будут использоваться для подстановки в таблицу «Врачи» (см.табл. 4). Таблица 4. Описание свойств полей таблицы «Штатное расписание» Для поля «Оклад» во вкладке **Общие** в строке «Формат поля» выбрать «Денежный». Для поля «Должность» задать значение **Ключевое поле**.

2. Заполнение таблиц данными.

Для заполнения таблиц данными необходимо перейти из режима **Конструктора** в режим **Таблицы** или открыть таблицу двойным щелчком. **Внимание!** (данные в таблицу можно ввести только в режиме **Таблицы**). Заполните таблицу «Смена_подстановка» данными (см.рис. 8).

Смена_подстановка	
НомерСмен	ВремяПрие
1	08.00-12.00
2	13.00-19.00

Рис. 8. Таблица «Смена_подстановка» с данными

Заполните таблицу «ШтатноеРасписание» данными (см.рис. 9).

ШтатноеРасписание : таблица			
	Должность	Оклад	Надбавка
+	врач-стоматолог	7 000,00р.	700,00р.
+	ортопед	14 000,00р.	1 200,00р.
+	хирург-терапевт	10 000,00р.	1 000,00р.

Рис. 9. Таблица «ШтатноеРасписание» с данными

3. Создание подстановочного поля.

Для создания подстановочного поля используется три способа: режим **Конструктора**. Этот способ используется в том случае, если поле для подстановки стоит первым в подстановочной таблице, использование **Мастера подстановок**, или использование языка запросов SQL (с помощью построителя запросов).

1) Откройте таблицу «Врачи» в режиме **Конструктора**. Для поля «Должность» во вкладке **Подстановка** выберите в строке **Тип элемента управления**: «Поле со списком»; в строке **Тип источника строк**: «Таблица или запрос» (задается по умолчанию); в строке **Источник строк** из выпадающего списка выбрать таблицу «Штатное расписание» - то есть таблицу, из которой данные будут подставляться в таблицу «Врачи» (см.рис. 10):

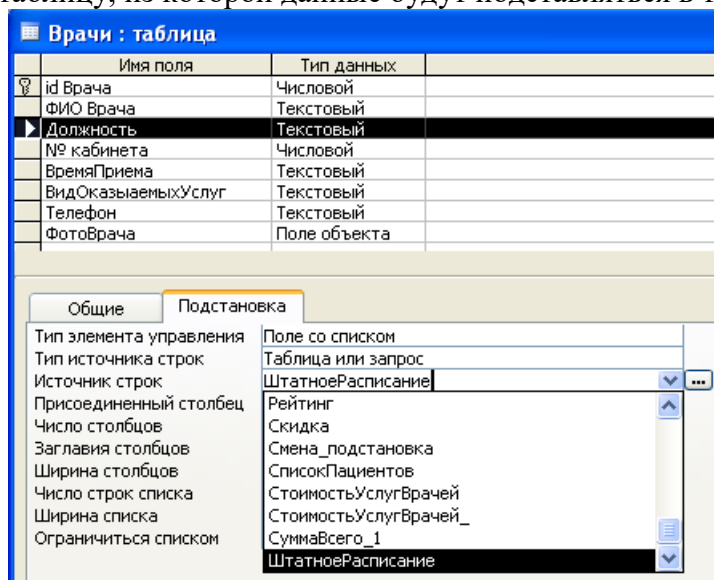


Рис. 10. Подстановка в режиме **Конструктора**

2) Для создания подстановки для поля «ВремяПриема» в столбце **Тип данных** выберите из выпадающего списка «**Мастер подстановки**». В следующем окне нажмите кнопку <Далее>, в открывшемся диалоговом окне выберите «Таблица: Смена_подстановка». Нажмите <Далее>. В следующем окне «Создание подстановки» из списка «Доступные поля» выберите при помощи кнопки поле «ВремяПриема» (данные из этого поля будут подставляться в нашу таблицу), нажмите кнопку <Далее>, затем <Готово> (рис. 11).

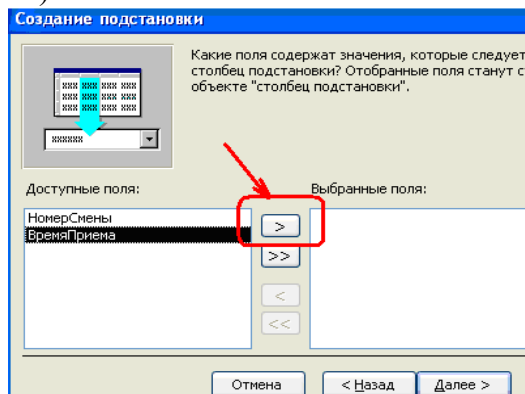


Рис.11. Использование **Мастера подстановки**

3) Использование языка запросов SQL. Для использования построителя запросов для поля «ВремяПриема» выберите вкладку **Подстановка**: в строке **Тип элемента управления** - «Поле со списком», в строке **Тип источника строк** – «Таблица или запрос» в строке **Источник строк** нажмите на кнопку **Построителя выражений** и в открывшемся диалоговом окне **Инструкция SQL: построитель запросов** на панели инструментов выберите: **Вид Режим SQL**.

запрос будет иметь вид:

SELECT Смена_подстановка.ВремяПриема FROM Смена_подстановка;

4. Ввод ограничения на данные в поле «ID врача».

Для поля «ID врача» в таблице «Врачи» выполним условия:

- коды врачей не должны повторяться,
- должна быть обеспечена возможность их изменения (из-за последнего условия в этом поле нельзя использовать тип данных **Счетчик**, в котором данные не повторяются).

Для выполнения этих условий сделайте следующее:

- установите курсор на строке «ID врача»;
- во вкладке **Общие** щелкните по строке параметра **Индексированное поле**;
- выберите в списке пункт «Да (совпадения не допускаются)»;
- перейдите в режим **Таблица**.

Примечание 2. Индекс - это средство MS Access, ускоряющее поиск и сортировку данных в таблице.

5. Заполните таблицу «Врачи» данными, используя подстановку (см.пример: рис. 12):

Использование режима "Подстановка"

Врачи : таблица						
	Табельный ном	ФИО Врача	Должность	Номер кабин	ВремяПриема	ВидОказываемыхУслуг
	100	Валиуллин Д.К.	ортопед	101	08.00-12.00	Ортопедическая
	101	Галимянова О.А.	врач-стоматолог	102	08.00-12.00	Хирургическая
	102	Дюкина Я.Б.	ортопед	203	08.00-12.00	Терапевтическая
	103	Елизарова Е.П.	хирург-терапевт	115	08.00-12.00	Терапевтическая
	104	Жуков А.Е.	врач-стоматолог	120	08.00-12.00	Терапевтическая
	105	Карбышев Г.О.	ортопед	214	08.00-12.00	Ортопедическая
	106	Леонидова М.Р.	врач-стоматолог	224	13.00-19.00	Терапевтическая
	107	Накипов Н.Д.	хирург-терапевт	315	08.00-12.00	Хирургическая
	108	Никоненко П.С.	ортопед	124	13.00-19.00	Ортопедическая

Рис. 12. Пример заполнения таблицы «Врачи» с использованием режима Подстановки

Пример 3. Размещение новых объектов в таблице.

1. Размещение объекта OLE.

Рассмотрим размещение объекта OLE в таблице «Врачи» на примере поля «ФотоВрача». В нашем примере фотографии хранятся в формате графического редактора **Paint** в файлах с расширением *bmp*. (Создайте несколько таких объектов и сохраните их в той же папке, где хранится база данных).

-Откройте таблицу «Врачи» в режиме **Конструктора** и добавьте новое поле «ФотоВрача». Тип данных – **Поле объекта OLE**.

-Перейдите в режим **Таблицы**. В графе «Фото» выполните команду правой кнопкой мыши: **Добавить объект**.

-В окне **Вставка объекта** (рис. 13) отметьте переключатель «Создать из файла». Окно **Вставка объекта** позволяет ввести имя файла с фотографией. Для поиска файла воспользуйтесь кнопкой **Обзор**

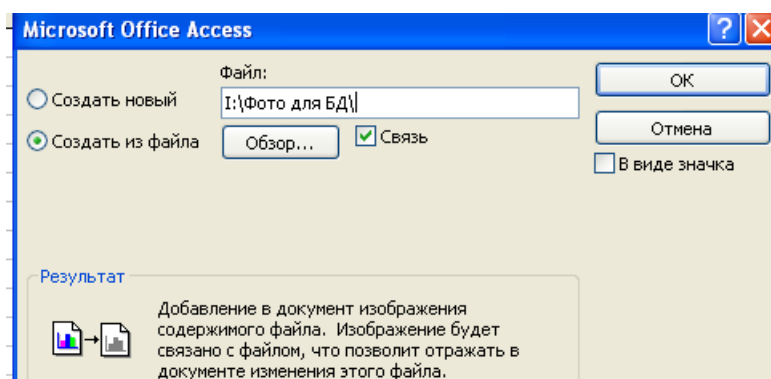


Рис. 13. Добавление объекта.

Для введения в поле связанного объекта надо установить флажок «Связь», что позволит в дальнейшем отражать в документе изменения этого файла

2. Добавление нового поля в таблицу. Создание маски ввода.

1. Откройте таблицу «Врачи» в режиме Конструктора.

2. Добавьте новое поле «Телефон». Тип данных – **Текстовый**.

3. Задайте маску ввода для поля **Телефон** (в виде 22-22-222), для этого:

-Щелкните на вкладке **Общие** в строке *Маска ввода* по кнопке.

-Подтвердите сохранение таблицы.

-В открывшемся диалоговом окне «Создание масок ввода» щелкните по кнопке <Список>. В появившемся диалоговом окне «Настройка масок ввода» выполните действия по образцу (рис. 14):

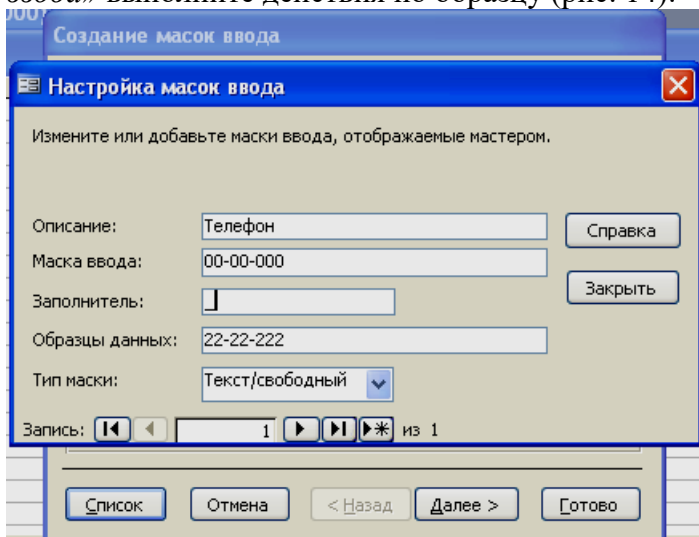


Рис. 14. Создание маски ввода

Нажмите кнопку <Заккрыть> и из появившегося списка выберите созданную маску ввода для поля *Телефон*.

Примечание 3. Маска ввода создается только для поля, имеющего тип данных – **текстовый**!

Таблица 5. Некоторые допустимые знаки масок ввода

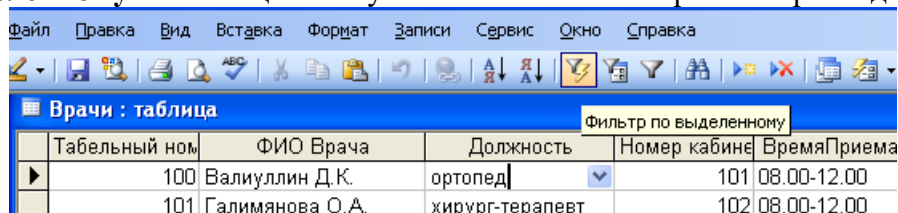
Знак	Описание
0	Цифра (от 0 до 9, ввод обязателен; знаки плюс [+] и минус [-] не допускаются).
9	Цифра или пробел (ввод не обязателен; знаки плюс и минус не допускаются).
#	Цифра или пробел (ввод не обязателен; пустые знаки преобразуются в пробелы, допускаются знаки плюс и минус).
L	Буква (от A до Z или от A до Я, ввод обязателен).

?	Буква (от А до Z или от А до Я, ввод не обязателен).
А	Буква или цифра (ввод обязателен).
&	Любой знак или пробел (ввод обязателен).

3. Фильтрация данных по полям

В режиме **Таблица** в таблице «Врачи» щелкните по записи *Ортопед* поля «Должность»;

1. щелкните по кнопке или выполните команду **Записи, Фильтр, Фильтр по выделенному**. В таблице останутся только записи о врачах - ортопедах;



Файл Правка Вид Вставка Формат Записи Сервис Окно Справка

Врачи : таблица Фильтр по выделенному

Табельный ном	ФИО Врача	Должность	Номер кабин	ВремяПриема
100	Валиуллин Д.К.	ортопед	101	08.00-12.00
101	Галимянова О.А.	хирург-терапевт	102	08.00-12.00

Рис. 15. Фильтр данных

2. для отмены фильтрации щелкните по кнопке на панели инструментов или выполните команду **Записи, Удалить фильтр**. В таблице появятся все данные.

Пример 4. Создание новых таблиц.

1. Создайте таблицу «**СписокПациентов**», которая содержит следующие поля:

1) Для поля «**МедПолис**» создайте маску ввода по образцу: «**АБ-1234**».

2) Для поля «**Телефон**» создайте маску ввода как в задании 3.

3) Для поля «**Пол**» во вкладке **Общие** в строке «**Значение по умолчанию**» введите: *муж*. Это значение будет указано по умолчанию. В случае необходимости его можно заменить.

4) Заполните таблицу данными (не менее 15 записей).

2. Создайте таблицу «**НаименованиеУслуг**», которая содержит следующие поля:

НаименованиеУслуг	
Имя поля	Тип данных
№ Услуги	Числовой
НаименованиеУслуги	Текстовый
ХарактеристикаУслуги	Текстовый
Стоимость	Денежный
Скидка	Числовой

Заполните таблицу данными по примеру:

НаименованиеУслуг				
№ Услуг	НаименованиеУслуги	ХарактеристикаУслуг	Стоимость	Скидка
1	Лечение однокорневого пульпита	Терапевтическая	1 250,00р.	3
2	Лечение двухкорневого пульпита	Терапевтическая	1 650,00р.	1
3	Лечение трехкорневого пульпита	Терапевтическая	2 000,00р.	0
4	Лечение кариеса	Терапевтическая	500,00р.	5
5	Светополимерная композитная пломба (Рос	Терапевтическая	350,00р.	0
6	Лечение периодонтита	Терапевтическая	1 500,00р.	5
7	Удаление зубного камня – (один зуб)	Хирургическая	60,00р.	0
8	Чистка всех зубов одной челюсти (ультразву	Хирургическая	800,00р.	0
9	Профессиональное отбеливание - (один зуб)	Хирургическая	300,00р.	3
10	Удаление зуба	Хирургическая	150,00р.	0

2. Создайте таблицу «**Журнал**», которая содержит следующие поля:

Журнал : таблица		
Имя поля	Тип данных	
ДатаПриема	Дата/время	
НомерКарточки	Числовой	
ФИО Пациента	Текстовый	
ФИО Врача	Текстовый	
ДолжностьВрача	Текстовый	
Услуги	Текстовый	
Количество	Числовой	

Общие	Подстановка
Формат поля	Краткий формат даты

Пример 5. Создание схемы данных.

Одним из способов представления данных, независимо от реализующего его программного обеспечения, является модель «Сущность-связь». Модель сущность-связь (ER-модель) (англ. entity-relationship model, ERM) — модель данных, позволяющая описывать концептуальные схемы предметной области. С её помощью можно выделить ключевые сущности и обозначить связи, которые могут устанавливаться между этими сущностями.

Основными понятиями метода «Сущность-связь» являются: сущность, атрибут сущности, ключ сущности, связь между сущностями, степень связи.

Сущность представляет собой объект, информация о котором хранится в базе данных. Названия сущностей – существительные. Например: *ВРАЧИ, ПАЦИЕНТЫ*.

Атрибут представляет собой свойство сущности. Так, атрибутами сущности ВРАЧИ являются: *табельный номер, ФИО, должность* и т.д.

Ключ сущности – атрибут или набор атрибутов, используемый для идентификации экземпляра сущности.

Связь сущностей – зависимости между атрибутами этих сущностей. Название связи обычно представляется глаголом. Например: *врач ЗАНИМАЕТ должность*.

Важным свойством модели «Сущность-связь» является то, что она может быть представлена в графическом виде, что значительно облегчает анализ предметной области. Будем использовать нотацию Чена-Мартина.

Таблица 6.

Условные обозначения нотации Чена-Мартина

Обозначение	Описание
	Набор независимых сущностей
	Набор зависимых сущностей
	Значение атрибута
	Ключевой атрибут
	Набор связей

В качестве примера построим диаграмму, отражающую связь между пациентами и врачами, которые оказывают стоматологические услуги. (см.рис. 20).

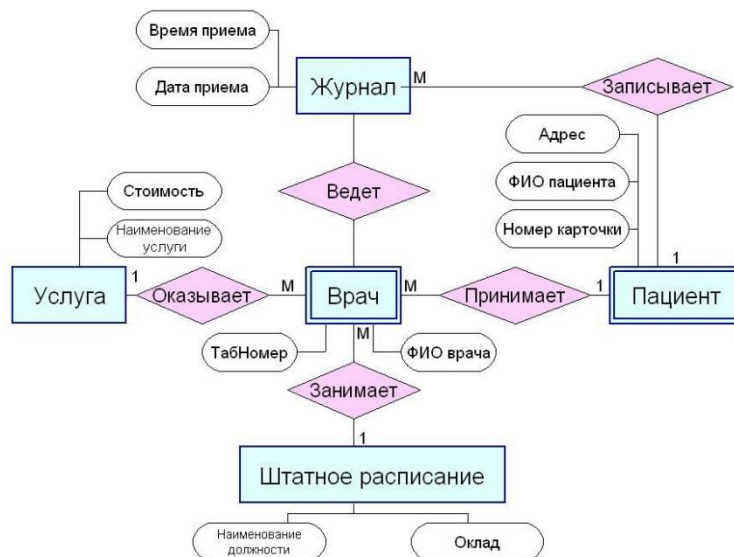
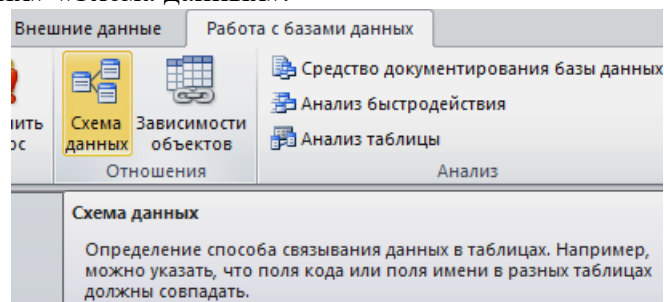


Рис.20. Пример ER-диаграммы

Во время проектирования баз данных происходит преобразование ER- модели в конкретную схему базы. Основными объектами схемы являются таблицы и связи. Рассмотрим создание схемы базы данных.

1. Для создания схемы данных выполните действия: вкладка «Работа с базами данных» «Схема данных».



2. Появится окно **Добавление таблицы**. Выберите вкладку **Таблицы**, выделите таблицы:

«Врачи», «ШтатноеРасписание», «НаименованиеУслуг», «Журнал», «СписокПациентов» и разместите их с помощью кнопки <Добавить>. В окне **Схема данных** появится условный вид этих таблиц (см.рис. 21). Закройте окно.

Примечание 3: Жирным текстом в схеме будут выделены поля, имеющие ключевое значение.

3. Установите связь между таблицами **СписокПациентов** и **Журнал**. Для этого установите курсор на имя поля «**НомерКарточки**» - в этой таблице это поле - ключевое (таблица *СписокПациентов*) и, не отпуская кнопку мышки, перетащите ее на поле «НомерКарточки» - подчиненной таблицы *Журнал*. Отпустите мышку. Появится диалоговое окно **Связи**.

4. Для автоматической поддержки целостности базы данных установите флажок: **Обеспечение целостности данных**. Кроме этого значка в этом окне представлены другие:

-**Каскадное обновление связанных полей**. При включении данного режима изменения, сделанные в связанном поле первой таблицы, автоматически вносятся в поля связанной таблицы, содержащей те же данные.

-**Каскадное удаление связанных записей**. При включении данного режима удаление записей в первой таблице приводит к удалению соответствующих записей связанной таблицы.

Примечание 4. Это возможно сделать только в том случае, если типы и размер полей заданы одинаково.

Щелкните по кнопке <Создать>. Появится связь **один-ко-многим**:

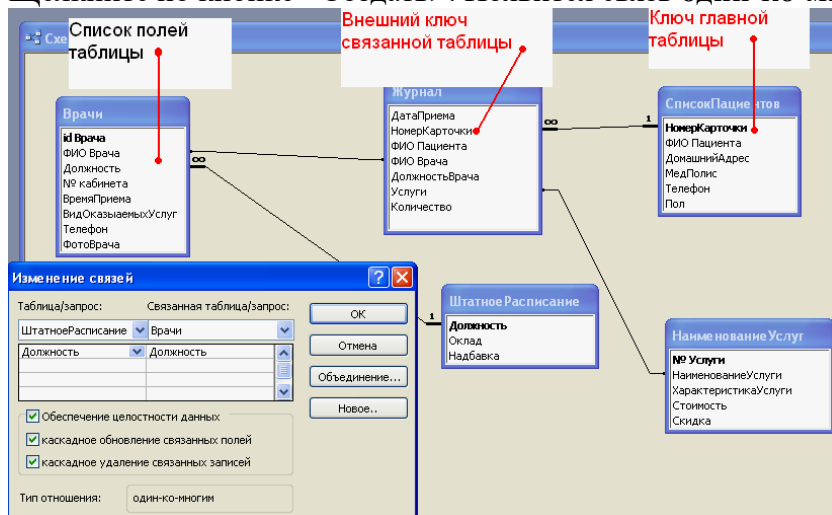


Рис. 21. Создание схемы данных

4. Прodelайте те же шаги для поля **Должность**.

Внимание! Изменить связь и параметры объединения можно, щелкнув правой кнопкой мыши по линии связи, как показано на рис. 22. Откроется окно **Изменение связей**. Выбрав кнопку **Объединение**, можно выбрать другой способ, что позволяет перейти к созданию связи между любыми двумя таблицами базы данных.

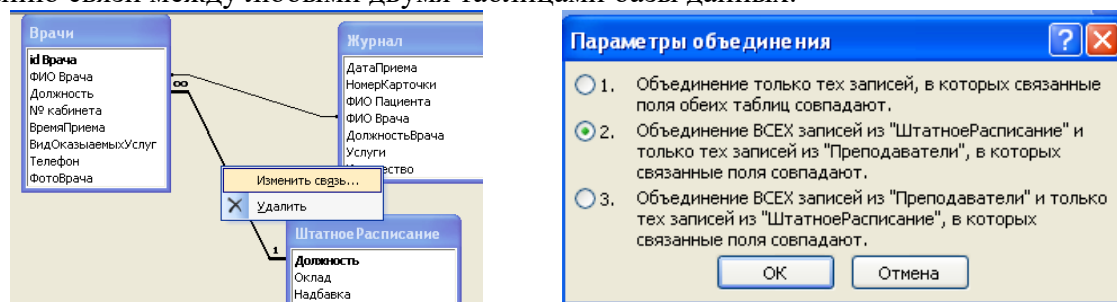


Рис. 22. Изменение или удаление связи

Задания для лабораторной работы

1. Разработать модель «Сущность-связь» по индивидуальному заданию и предварительно определить структуру таблиц базы данных.

2. Разработать структуру (схему данных) базовых таблиц (не менее трех) базы данных (смотри таблицу заданий к работе), удовлетворяющих требованиям целостности, непротиворечивости и не избыточности. В таблицах в соответствии с типом данных, размещенных в каждом поле, определите наиболее подходящий тип для каждого поля.

3. Создать структуры базовых таблиц и наполнить их содержимым, состоящим более чем из 15 записей. При создании структуры таблиц целесообразно задавать ключевые (уникальные) поля. Это поможет в дальнейшем для организации связей между таблицами.

Варианты заданий

1	База данных «Студенческая библиотека». Ориентировочные таблицы: «Книжный каталог», «Категории», «Картотечный номер»
---	---

2	База данных «Страховая фирма». Ориентировочные таблицы: «Виды страховок», «Клиенты\объекты», «Страховая деятельность»
3	База данных «Агентство недвижимости». Ориентировочные таблицы: «Объекты недвижимости», «Продажи», «Покупки».
4	База данных ГИБДД (Государственная инспекция безопасности дорожного движения). Ориентировочные таблицы: «Сотрудники», «Штрафы», «Нарушения»
5	База данных «Деканат ВУЗа». Ориентировочные таблицы: «Список студентов», «Список предметов», «Сессия»
6	База данных отдела кадров производственного предприятия. Ориентировочные таблицы: «Сотрудники», «Штатное расписание», «Отделы», «Цеха».
7	База данных фирмы покупки и продажи автомобилей. Ориентировочные таблицы: «Продажи», «Покупки», «Автомобили»
8	База данных «Гостиница». Ориентировочные таблицы: «Номера», «Счета», «Клиенты»
9	База данных «Расчет квартплаты ТСЖ». Ориентировочные таблицы: «Список жильцов», «Оплаты», «Тарифы».
10	База данных «Железнодорожные кассы». Ориентировочные таблицы: «Продажи», «Посадочные места», «Направления»
11	База данных «Авиапассажирские перевозки». Ориентировочные таблицы: «Рейсы», «Самолеты», «Продажи».
12	База данных «Музей». Ориентировочные таблицы: «Экспонаты», «Авторы», «Экспозиции».
13	База данных «Спортивные комплексы района». Ориентировочные таблицы: «Нормативы», «Спортсмены», «Соревнования»
14	База данных «Экзаменационная сессия». Ориентировочные таблицы: «Предметы», «Оценки», «Студенты»
15	База данных «Турагентство». Ориентировочные таблицы: «Туры», «Продажи», «Клиенты»
16	База данных «Аптека». Ориентировочные таблицы: «Товары», «Поставщики», «Продажи»
17	База данных «Сборка и реализация компьютеров». Ориентировочные таблицы: «Продукция», «Клиенты», «Заказы».
18	База данных «Продуктовый магазин». Ориентировочные таблицы: «Продажи», «Отделы», «Товары».
19	База данных «Картинная галерея». Ориентировочные таблицы: «Выставки», «Картины», «Сотрудники»
20	База данных «Театр». Ориентировочные таблицы: «Артисты», «Театральные постановки», «Продажи билетов»
21	База данных «Ювелирный магазин». Ориентировочные таблицы: «Сотрудники», «Ювелирные изделия», «Журнал покупок»

22	База данных «Парикмахерская». Ориентировочные таблицы: «Клиенты», «Прайс лист», «Сотрудники».
23	База данных «Кондитерский магазин». Ориентировочные таблицы: «Кондитерские изделия», «Поставщики», «Сотрудники».
24	База данных «Автобаза». Ориентировочные таблицы: «Автомашины», «Рейсы», «Сотрудники».
25	База данных «Склад». Ориентировочные таблицы: «Поставщики», «список товаров», «Кладовщики».
26	База данных «Ремонтная мастерская». Ориентировочные таблицы: «Виды работ», «Прайс лист», «Заказы».
27	База данных «Столовая». Ориентировочные таблицы: «Продукты», «Блюда», «Меню».
28	База данных «Фильмотека». Ориентировочные таблицы: «Список фильмов», «Список клиентов», «Журнал выдачи фильмов».
29	База данных «Компьютерные занятия». Ориентировочные таблицы: «Список слушателей курсов», «список предметов», «список преподавателей», «Журнал успеваемости».
30	База данных «Проектные работы». Ориентировочные таблицы: «Список проектов», «Список специалистов», «Список должностей», «Журнал учета работ».

ЛАБОРАТОРНАЯ РАБОТА №2. Системы управления базами данных MS ACCESS: создание запросов.

Цель лабораторной работы: Знакомство с пакетом Microsoft Office Access. Приобретение навыков и умений при работе с базой данных, в процессе создания запросов на выборку, с параметром, перекрестный и сводная таблица.

Методические указания

Запросы являются важным инструментом в любых системах управления базами данных. Они используются для выделения, обновления и добавления новых записей в таблицы. Чаще всего запросы используются для выделения специфических групп записей, чтобы удовлетворить определенному критерию. Кроме того, их можно использовать для получения данных из различных таблиц, обеспечивая единое представление связанных элементов данных.

В Access существует четыре типа запросов для различных целей:

- Запросы на выборку отображают данные из одной или нескольких таблиц в виде таблицы.
- Перекрестные запросы собирают данные из одной или нескольких таблиц в формате, похожем на формат электронной таблицы. Эти запросы используются для анализа данных

и создания диаграмм, основанных на суммарных значениях числовых величин из некоторого множества записей.

- Запросы на изменение используются для создания новых таблиц из результатов запроса и для внесения изменений в данные существующих таблиц. С их помощью можно добавлять или удалять записи из таблицы и изменять записи согласно выражениям, задаваемым в режиме конструктора запроса.

- Запросы с параметрами — это такие запросы, свойства которых изменяются пользователем при каждом запуске. При запуске запроса с параметром появляется диалоговое окно, в котором нужно ввести условие отбора. Этот тип запроса не является обособленным, т. е. параметр можно добавить к запросу любого типа.

Запросы можно создавать с помощью Мастера запросов (Query Wizard) или Конструктора запросов (Query Design).

Создание запросов

В случае создания запроса можно воспользоваться мастером запросов. Для этого нужно открыть окно созданной базы данных перейти на вкладку Создание. Затем можно выбрать режим создания запроса: в режиме Конструктора запросов или с помощью Мастера запросов. Для создания простых запросов обычно используется мастер запросов, который последовательно запрашивает наименования таблиц, используемых в запросе, перечень полей таблиц и некоторые дополнительные параметры.

Для создания более сложных запросов может понадобиться конструктор запросов.

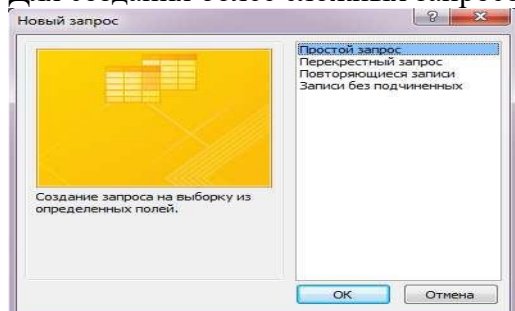



Рис.1. Окно выбора режима создания запроса с помощью Мастера запросов

- Простой запрос - представляет собой запрос на выборку.
- Перекрестный запрос - собирает данные из одной или нескольких таблиц в формате, похожем на формат электронной таблицы.
- Повторяющиеся записи - создание запроса на поиск повторяющихся записей в простой таблице или запросе.
- Записи без подчиненных - создание запроса на поиск записей, которым не соответствует ни одна запись в подчиненной таблице.

Создание запросов с помощью Конструктора

Для создания запроса с помощью Конструктора запросов можно воспользоваться соответствующим пунктом () из вкладки Создание в окне базы данных.

На первом диалоговом окне Конструктор предлагает выбрать те таблицы и запросы, из которых будут использованы поля для формирования нового запроса.

В верхней части окна конструктора запросов появится схема данных запроса.

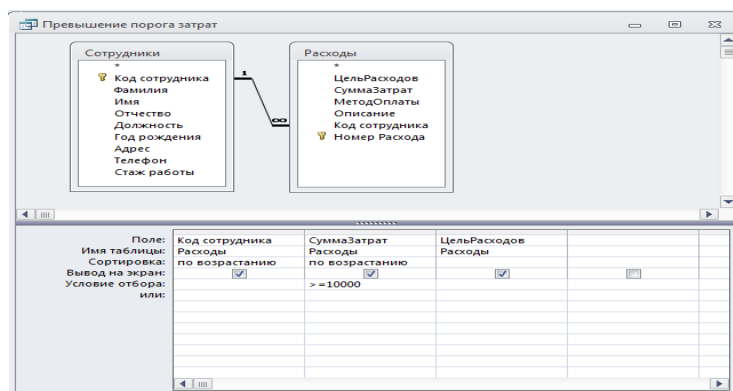


Рис.2. Создание запроса с помощью Конструктора

Эта схема очень сильно напоминает схему базы данных. В отличие от нее, данная схема содержит список таблиц, включенных в запрос, и отображает связи между ними. В нижней части окна располагается бланк запроса. Каждая строка этого бланка выполняет определенную функцию:

- **Поле.** В этой строке помещаются те поля, которые используются для создания запроса, каждое в своей ячейке таблицы.
- **Имя таблицы.** Эта строка показывает, из какой таблицы (или запроса) выбрано данное поле.
- **Сортировка.** В этой строке указывается тип сортировки информации, возвращаемой в запросе, по возрастанию (от А до Я, от большего к меньшему, от более раннего к более позднему и т.д.) или по убыванию (от Я к А и т.д.). Возможна многоуровневая сортировка — сразу по нескольким полям. В этом случае данные сначала сортируются по тому полю, которое в бланке запроса по образцу находится левее, затем по следующему полю, для которого включена сортировка, и так далее слева направо. Соответственно, при формировании запроса надо располагать поля результирующей таблицы не как попало, а с учетом будущей сортировки.
- **Вывод на экран.** По умолчанию предполагается, что все поля, включенные в запрос, должны выводиться на экран, но это не всегда целесообразно. Например, бывают случаи, когда некое поле необходимо включить в запрос только потому, что оно является полем сортировки, но, в то же время, нежелательно, чтобы пользователь базы видел его содержание. В таких случаях отображение содержимого на экране подавляют сбросом флажка Вывод на экран.
- **Условие отбора.** Это наиболее важная часть бланка запроса. В этой строке (и в строке, расположенной ниже ее) вводятся ограничения поиска, задавая определенные условия, которые принято называть критерием поиска.

В качестве критерия поиска можно использовать некоторые постоянные значения. Например, для выбора записей с пустыми значениями надо в строке Условие отбора ввести значение *null*, если наоборот, надо найти записи, не содержащие в определенном поле пустых значений, надо ввести *not null*. Чтобы найти пустые записи в логическом поле надо ввести *No* или *False*, а для выбора только заполненных полей - *Yes* или *True*. Другим примером может служить выбор полей со значениями, лежащими в каком-то диапазоне - для этого можно использовать служебные выражения, например,

between 01.01.14 and 31.01.14.

Результатом поиска будут только те поля, даты в которых соответствуют январю 2014 года. Для формирования критерия поиска также можно воспользоваться сложной выборкой, например, с использованием функций Visual Basic. Например, для выбора только тех значений поля «Телефон», номер которых начинается с 473, можно воспользоваться функцией *Left([Телефон], 3) = «473»*.


Возможно использование нескольких условий в одном запросе с помощью оператора конъюнкции *And*. Также можно использовать оператор дизъюнкции *Or* для выборки

записей, удовлетворяющих либо одному, либо другому условию. При этом условия надо вводить в разных строках бланка запроса, т.е. в строке Условие запроса и в строке или.

Более подробно о возможных критериях поиска можно узнать, воспользовавшись встроенной поисковой системой - для этого введите в поле поиска текст Выражения.

Создание простых запросов с помощью мастера

Как уже говорилось выше, для создания запроса с помощью Мастера запросов можно

воспользоваться соответствующим пунктом () из вкладки Создание в окне базы данных. В результате откроется окно мастера запросов, в котором можно выбрать требуемые поля из одной или нескольких таблиц (рис.3). Если было выбрано более одной таблицы, мастер автоматически связывает таблицы:

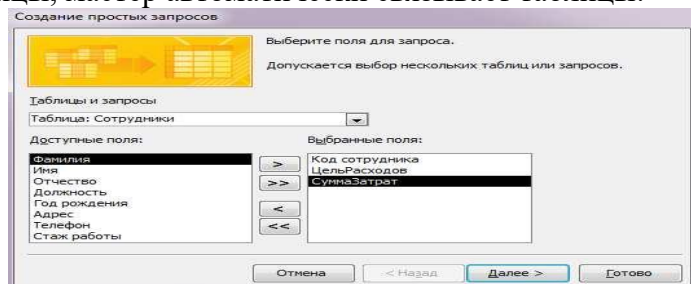
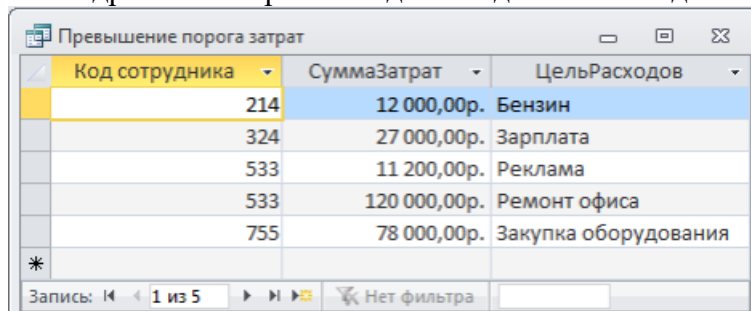


Рис.3. Создание простого запроса

После выбора полей нажмите кнопку Далее и на экране появится следующий диалог Мастера запросов, в котором необходимо выбрать признак подробного или итогового запроса. Подробный запрос выводит каждое поле каждой записи:



Код сотрудника	СуммаЗатрат	ЦельРасходов
214	12 000,00р.	Бензин
324	27 000,00р.	Зарплата
533	11 200,00р.	Реклама
533	120 000,00р.	Ремонт офиса
755	78 000,00р.	Закупка оборудования

Рис.4. Результат выполнения простого запроса с подробным отчетом

Итоговый запрос позволяет рассчитать четыре параметра для каждого цифрового (или денежного) поля, включенного в запрос: расчет суммы, среднего, максимального и минимального значений. Затем мастер предложит ввести название запроса и на этом формирование запроса закончится. При необходимости внести изменения в уже созданный запрос необходимо воспользоваться Конструктором запросов.

Создание перекрестных запросов с помощью мастера

В перекрестном запросе отображаются результаты статистических расчетов (такие как суммы, количество записей и средние значения), выполненных по данным из одного поля таблицы. Эти результаты группируются по двум наборам данных, один из которых расположен в левом столбце таблицы, а второй - в верхней строке.

В первом столбце перекрестной таблицы отображаются значения из полей базовой таблицы или запроса; а количество полей для первого столбца может достигать до трех. Значения этих полей образуют заголовки строк перекрестной таблицы. Из значений другого поля формируются заголовки столбцов. После выбора заголовков мастер предлагает указать поле (только одно) и выбрать тип статистической обработки (максимум, среднее, сумма и т.п.) для данных из этого поля. Результаты этой обработки будут выводиться в перекрестном запросе на пересечении строк и заголовков. На рисунке 5 приведен результат работы перекрестной таблицы, в которой в качестве строк выступает цель расходов; в качестве столбцов - коды сотрудников; на пересечении выведены суммарные расходы затрат по каждой статье для каждого сотрудника.

ЦельРасходов	214	324	533	755
Бензин	12 000,00р.			
Закупка оборудования				78 000,00р.
Зарплата		27 000,00р.		
Командировочные услуг			7 000,00р.	
Коммунальные услуги		2 700,00р.		
Реклама			11 200,00р.	
Ремонт офиса			120 000,00р.	

Записи: 1 из 7 Нет фильтра

Рис.5. Результат выполнения перекрестного запроса

Если необходимо создать перекрестный запрос для данных из связанных таблиц, то сначала необходимо сформулировать базовый простой запрос на выборку, на основе которого потом будет создан перекрестный запрос.

Создание запросов с параметрами

Запрос с параметрами - это запрос, при выполнении которого в его диалоговом окне пользователю выдается приглашение ввести данные, например условие для возвращения записей или значение, которое должно содержаться в поле. Можно создать запрос, в результате которого выводится приглашение на ввод нескольких данных, например, двух дат. В результате будут возвращены все записи, находящиеся между указанными двумя датами.

Для создания запроса с параметрами необходимо ввести текст сообщения пользователю в строке Условие отбора бланка запроса для любого поля (или полей). При этом вводимый текст должен быть заключен в квадратные скобки. Тогда при запуске запроса появится сообщение, предлагающее пользователю ввести условие отбора записей.

На рисунке 6 представлен бланк запроса, причем для поля должность введено условие отбора. После запуска данного запроса появится бланк, предлагающий ввести должность сотрудника (рис.7), например, программист. В результате появится результирующая таблица, содержащая информацию по программисту фирмы (рис. 8).

Сотрудники

- Код сотрудника
- Фамилия
- Имя
- Отчество
- Должность
- Год рождения
- Адрес
- Телефон
- Стаж работы

Расходы

- ЦельРасходов
- СуммаЗатрат
- МетодОплаты
- Описание
- Код сотрудника
- Номер Расхода

Поле: Должность

Имя таблицы: Сотрудники

Сортировка:

Вывод на экран:

Условие отбора: [Введите должность сотрудника]

Должность	Имя	Фамилия	Отчество	СуммаЗатрат
Сотрудники	Сотрудники	Сотрудники	Сотрудники	Сотрудники

Рис.6. Создание запроса с параметрами

Введите значение параметра

Введите должность сотрудника

Программист

OK Отмена

Рис7. Пример сообщения, появляющегося в результате запуска запроса с параметром

Должность	Имя	Фамилия	Отчество	СуммаЗатрат
Программист	Иван	Иванов	Иванович	78 000,00р.
*				

Записи: 1 из 1 Нет фильтра

Рис.8. Результат выполнения запроса с параметрами

Запросы с параметрами удобно использовать в качестве основы для форм и отчетов. Например, на основе запроса с параметрами можно создать месячный отчет о доходах. При выводе данного отчета, на экране появится приглашение ввести месяц, доходы которого интересуют пользователя. После ввода месяца на экране будет представлен требуемый отчет.

Изменение порядка полей

Порядок полей в бланке запроса определяет порядок появления их в результирующей таблице. Для того чтобы изменить расположение поля в этом списке, выполните следующие действия:

1. Установите указатель мыши на область выбора столбца, который располагается прямо над названием поля. Когда указатель изменит вид на стрелку, щелкните по левой кнопке мыши, чтобы выделить столбец.
2. Не перемещая указатель мыши, опять нажмите левую кнопку мыши и удерживайте в этом положении. На конце указателя появится прямоугольник.
3. Перемещайте столбец в требуемом направлении. Толстая вертикальная линия покажет его текущее положение.
4. Отпустите кнопку, когда толстая вертикальная линия окажется в требуемом месте. Поле будет перемещено в новое место. Перемещение полей в режиме таблицы не вызывает изменений в структуре таблицы.

Запросы на изменение и их использование

Запрос на изменение - это запрос, который за одну операцию вносит изменения в несколько записей. Существует четыре типа запросов на изменение: на добавление, на удаление, на обновление записей, а также на создание таблицы.

Запрос на добавление

Добавляет группу записей из одной или нескольких таблиц в конец одной или нескольких таблиц. При этом таблицы не обязательно должны иметь одинаковое число полей или одинаковые имена полей. Запрос на добавление также полезен при выполнении следующих действий:

- Добавление полей на основе условий отбора. Например, необходимо добавить имена и адреса клиентов с очень крупными заказами.
- Добавление записей, если некоторые поля из одной таблицы не существуют в другой. Чтобы создать запрос на добавление надо выполнить следующие действия:
 1. В окне базы данных щелкните на вкладке Создать.
 2. Далее во вкладке выбрать Конструктор запросов.
 3. В появившемся окне Добавление таблицы надо выделить таблицу, из которой будут добавляться поля. Затем щелкнуть по кнопке Добавить, а затем - по кнопке Заккрыть.
 4. На следующем этапе надо выполнить команду Добавление из пункта меню Запрос.
 5. В появившемся диалоговом окне выбрать таблицу, в которую будут добавляться записи.
 6. Перетащить мышкой поля (из которых будут добавляться записи) в верхней половине окна в строку Поле бланка запроса(можно не перетаскивать, а воспользоваться двойным щелчком мыши).

Если значения из добавляемого поля должны быть добавлены в поле с другим именем, то в строке Добавление надо указать поле, в которое они будут добавлены.

Если следует добавлять не все значения из указанного поля, а только отвечающие какому-то условию, то его надо указать в строке Условие отбора.

Запрос на удаление

Удаляет группу записей из одной или нескольких таблиц. Например, запрос на удаление позволяет удалить записи о товарах, поставки которых прекращены или на которые нет заказов. С помощью запроса на удаление можно удалять только всю запись, а не отдельные поля внутри нее.

Последовательность создания запроса следующая:

1. В окне базы данных щелкните на вкладке Создать.
2. В открывшейся вкладке открыть Конструктор запросов.
3. В появившемся окне Добавление таблицы надо выделить таблицу, из которой будут удаляться записи. Затем щелкнуть по кнопке Добавить, а затем - по кнопке Заккрыть.
4. На следующем этапе надо выполнить команду Удаление из пункта меню Запрос.
5. Перетащить мышкой поле (значение которого будет использоваться для выбора удаляемых записей) в верхней половине окна в строку Поле бланка запроса (можно не перетаскивать, а воспользоваться двойным щелчком мыши).
6. В строке Условие отбора указать условие, в соответствии с которым будут отбираться записи на удаление. Запрос на удаление позволяет также удалять записи из связанных таблиц. Последовательность создания такого запроса почти такая же, как и в предыдущем случае, однако на 3 этапе надо добавить все таблицы, из которых будут удаляться связанные записи. Затем на 5 этапе надо перетащить мышкой символ звездочки «*» из списка полей базовой таблицы в верхней части окна в первый столбец бланка запроса. Данное действие указывает Access на то, что в запросе необходимо отобразить все поля таблицы. При этом в строке Удаление появится слово *Из*.

После создания запроса можно запустить его на выполнение, выполнив команду Запуск из пункта меню Запрос.

Запрос на обновление записей

Вносит общие изменения в группу записей одной или нескольких таблиц. Например, на 5 процентов увеличивается зарплата сотрудников определенной категории. Запрос на обновление записей позволяет изменять данные в существующих таблицах. Последовательность создания почти такая же, как и в предыдущем случае, однако на 4 этапе надо выбрать команду Обновление. Самый простой тип запроса на обновление используется для поиска определенного значения и замены его другим значением. При этом в строке Обновление бланка запроса надо указать новое значение, а в строке Условие отбора - старое. Запрос на обновление позволяет изменить формат данных. Возможно также обновление записей в таблице на основании значений полей в связанных записях другой таблицы. Для этого в строке Обновление бланка запроса надо указать поле, значения которого будут использованы для обновления.

Запрос на создание таблицы

Создает новую таблицу на основе всех или части данных из одной или нескольких таблиц. Запрос на создание таблицы полезен для выполнения следующих действий:

- Создание таблицы для экспорта в другую базу данных Microsoft Access.
- Создание отчетов, содержащих данные, соответствующие определенному моменту времени.
- Создание резервной копии таблицы.
- Создание архивной таблицы, содержащей старые записи.
- Повышение быстродействия форм и отчетов, базирующихся на многотабличных запросах или инструкциях SQL. Например, требуется вывести на печать несколько отчетов, базирующихся на запросе, включающем пять таблиц, в котором рассчитываются общие итоги. Чтобы ускорить процесс, разработайте запрос на создание таблицы, извлекающий необходимые записи и сохраняющий их в одной таблице. Затем на базе этой таблицы создайте отчет или укажите ее в инструкции SQL как источник записей для формы или

отчета. Это позволит обойтись без повторного запусков запроса для каждого отчета. Однако следует помнить, что после выполнения запроса на создание таблицы данные в этой таблице не изменяются.

Пример1. Формирование запросов на выборку.

1. На основе таблицы *Врачи* создать простой запрос на выборку, в котором должны отображаться **Табельный номер, ФИО врачей** и их **должность**.

2. Использовать фильтр для отображения некоторых данных, например все **врачи-стоматологи**.

3. Создать запрос на выборку с параметром, в котором должны отображаться ФИО врачей, время приема и номера кабинетов.

4. Создать запрос с помощью построителя выражений.

Ход работы


1. Для создания **простого** запроса:

-в окне базы данных откройте вкладку *Запросы*;

-в открывшемся окне щелкните по кнопке <Создать>;

-из появившихся пунктов окна "Новый запрос" выберите **Простой запрос** и щелкните по кнопке <ОК>;

-в появившемся окне в строке *Таблицы/запросы* выберите таблицу *Врачи*;

-в окне «Доступные поля» переведите выделение на параметр *ID Врача*, щелкните по кнопке . Поле *ID Врача* перейдет в окно «Выбранные поля»;

-аналогично в окно **Выбранные поля** переведите поля: «ФИО врача», «Должность» (порядок важен - в таком порядке данные и будут выводиться);

-щелкните по кнопке. <Далее>;


-в строке параметра «Сохранение запроса» введите новое имя «*Должности врачей*». <Готово>.

2. Для выборки данных: всех врачей, занимающих должность *Врач-стоматолог*, выполните следующие действия:

-В режиме запроса поставьте курсор в строку с записью «*Врач-стоматолог*».

-На панели инструментов выполните следующую команду **Записи Фильтр Фильтр по выделенному**.


-В таблице отобразятся только данные врачей, занимающих должность *стоматолога*.

-Для **отмены** фильтра нажмите кнопку .

3. Для создания запроса на выборку с параметром:


-создайте запрос на выборку для следующих полей таблицы «Врачи»: *ID Врача, ФИО врача, № кабинета, ВремяПриема*; таблицы «Смена_подстановка»: *НомерСмены*.

-задайте имя запросу «*Время приема*»;

-щелкните по кнопке . На экране появится таблица с результатами запроса;

-перейдите в режиме конструктора запроса;

-в строке параметра *Условия отбора* для поля «НомерСмены» введите фразу в квадратных скобках: [*Введите номер смены*];

-выполните запрос .

-в появившемся окне введите «1» и щелкните по кнопке <ОК>. На экране появится таблица с данными всех врачей, работающих в первую смену;

-в режиме **Конструктор** уберите галочку в строке вывода на экран (чтобы в результирующей таблице не выводиться повторяющихся данных с номером смены в каждой строке);

-сохраните запрос и закройте окно.

4. Создание запроса с помощью **Конструктора**.

Во вкладке запроса выполните команду: **Создать Конструктор**. В появившемся окне «Запрос1: Запрос на выборку» выберите таблицы *Врачи* и *НаименованиеУслуг*. Кнопка <Добавить>.

Двойным щелчком мыши выберите поля:

- из таблицы «**Врачи**»: *ID Врача, ФИО Врача*;
- из таблицы «**НаименованиеУслуг**»: *НаименованиеУслуги*.

В столбце: *ФИО Врача* в строке: **Условие отбора** ввести: *M**. Щелкните по кнопке



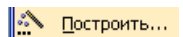
. Посмотрите результат запроса. В результате мы получим информацию о врачах и услугах, фамилии которых начинается на букву **М**.

Закройте запрос и сохраните под названием «*Врачи-выборка*».

5. Создание запроса с помощью построителя выражения.

-- Создайте запрос: расчет зарплаты, добавив таблицы: «**Врачи**»: поля - *ID Врача, ФИО Врача, Должность*; «**Штатное расписание**»: поля – *Зарплата, Надбавка*.

В свободной столбце в строке **Поле**: нажмите правой кнопкой мыши, выберите



В появившемся диалоговом окне **Построитель выражения**:

- выберите из списка таблицу «**Штатное расписание**»,
- из открывшегося списка полей щелкните по полю [Оклад], затем знак «+» и поле [Надбавка], - нажмите <ОК> (см. рис. 1).

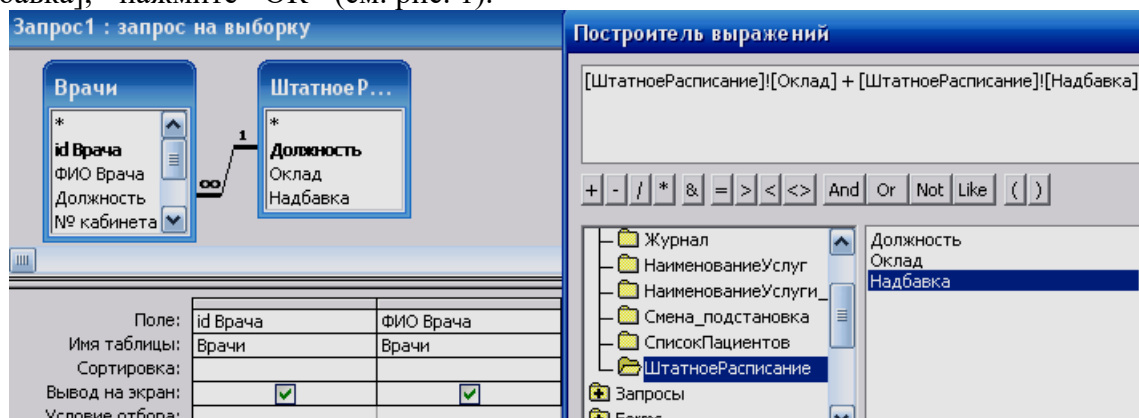


Рис. 1. Создание запроса с вычисляемым полем

- щелкните по кнопке

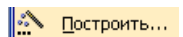
- сохраните запрос «*Итого зарплата*».

- в режиме Конструктора в строке **Поле**: выделите *Выражение1*: и переименуйте на «*Итого:*».

-- Создайте запрос: выборка данных за определенный период, добавив таблицу: «**Журнал**», поля:

*ДатаПриема,
ФИО Врача,
Услуга,
Количество,
ФИО Пациента.*

В свободной столбце в строке **Поле**: нажмите правой кнопкой мыши, выберите



В появившемся диалоговом окне **Построитель выражения** выберите:

Функции Встроенные функции Дата/время. Щелкните два раза кнопкой мыши <**Month**>. Выделите «*number*».

Далее выберите таблицу «**Журнал**» из списка окошка **Построителя** для выборки поля *ДатаПриема*, щелкнув по раскрывающемуся списку:


Таблицы Журнал, ДатаПриема.

В поле построителя должно появиться следующее выражение:

Month([Журнал].[ДатаПриема])

Нажмите <ОК> (см. рис. 2).

В этом же столбце в строке **Условие отбора** ввести: [Введите номер месяца].

Нажмите кнопку .

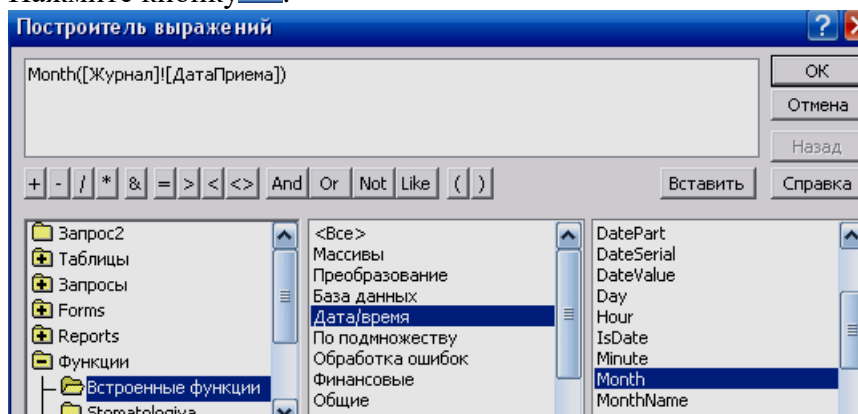


Рис. 2. Создание запроса

Для удаления повторяющихся записей в режиме **Конструктора** уберите галочку в столбце, где записано выражение.

--Создайте запрос на выборку фамилий пациентов, рожденных после 1990 года:
Для этого добавьте поле *ДатаРождения* в таблицу «Пациенты» в формате

Поле	<i>Year([Пациенты]![ДатаРождения])</i>
Условие отбора	<[Год рождения]

--Используя таблицу *Журнал* подсчитайте *Общее количество выполненных услуг* каждого врача.


Для этого выберите поля *ФИО врача* и *Количество*. На панели инструментов нажмите  - **Групповые операции**. В столбце *Количество* в строке «Групповая операция» выбрать из выпадающего списка **Sum**. Просмотреть запрос и проверить правильность, используя таблицу *Журнал* и фильтр записей.

Таблица: Назначение агрегатных функций

Функция	Назначение
Sum	Вычисляет сумму значений, содержащихся в поле
Avg	Вычисляет среднее арифметическое для всех значений поля
Count	Определяет число значений поля, не считая пустых (Null) значений
Min	Находит наименьшее значение поля
Max	Находит наибольшее значение поля
StDev	Определяет среднееквадратичное отклонение от среднего значения
Var	Вычисляет дисперсию значений поля

Примечание: Функция *Ccur*, используемая в выражении, преобразует результаты вычислений в денежный формат.-Функция *Count* вычисляет число записей, содержащих значение в поле.

--Создайте запрос на подсчет количества врачей по должностям.

Для этого в режиме **Конструктора** выберите таблицу *Врачи* (поле: *Должность* добавьте два раза). Выберите групповую операцию. Во втором столбце в строке «Групповая операция» из выпадающего списка выберите функцию **Count**.

В результате запроса должны получиться следующие данные:

Запрос4 : запрос на выборку		
	Должность	Count-Должнос
	врач-стоматолог	8
	ортопед	3
	хирург-терапевт	3

Пример 2. Формирование запросов на обновление и удаление.

1. Запрос на обновление

Создайте таблицы: *Товар* и *Отгрузка*.

Имя поля	Тип данных
Код товара	Числовой
Наименование	Текстовый
Цена	Денежный

Имя поля	Тип данных
НомерНакладной	Числовой
Код товара	Числовой
Отгружено	Числовой
ЦенаЗаЕд	Денежный
Стоимость	Денежный


Заполните поля таблиц. Поля *ЦенаЗаЕд* и *Стоимость* оставьте незаполненными! Они будут заполнены в результате выполнения запроса на обновление, который данные из таблицы «Товар» перенесет в таблицу «Отгрузка».

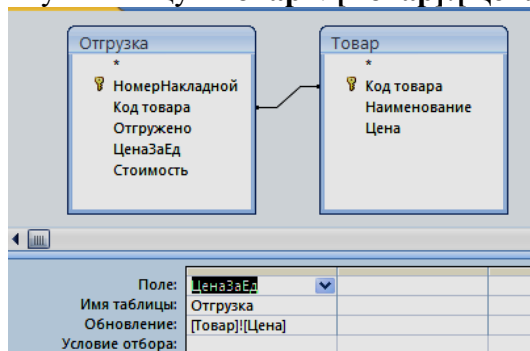
Примерные данные, которыми можно заполнить таблицы:

Товар : таблица		
КодТовара	Наименование	Цена
1	Стол	4 600,00р.
2	Стул	1 500,00р.
3	Кровать	25 000,00р.
4	Диван	42 000,00р.
5	Ковер	300,00р.
6	Кресло	12 000,00р.
7	Комод	7 200,00р.
8	Шкаф	23 000,00р.
0		0,00р.

Отгружено : таблица				
НомерНакладной	КодТовара	Отгружено	ЦенаЗ	
111	1	10		
112	2	1		
113	3	2		
114	4	3		
115	5	5		
116	6	10		
117	7	2		
118	8	1		
*	0	0		

Запрос 1. Создайте запрос на обновление. Для этого:

- 1) Добавьте обе таблицы в запрос.
- 2) Выберите поле *ЦенаЗаЕд* (табл. «Отгрузка»). Щелкните на панели инструментов кнопку  (обновление).
- 3) В строке «Обновление» щелкните правой кнопкой **Построить**.
- 4) В открывшемся диалоговом окне Построителя выражений выберите команду, используя таблицу «Товар»: **[Товар]![Цена]**.



В результате запроса данные поля *Цена* (таблица «Товар») отобразятся в поле *ЦенаЗаЕд* (таблица «Отгрузка»).

Запрос 2. Создайте запрос на обновление поля *Стоимость*, данные для которого являются результатом *Цена*Количество* товара (см.рис. 31):

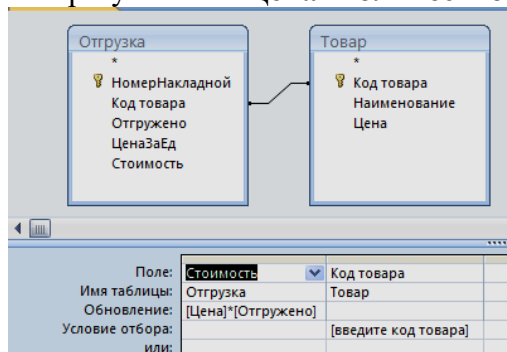



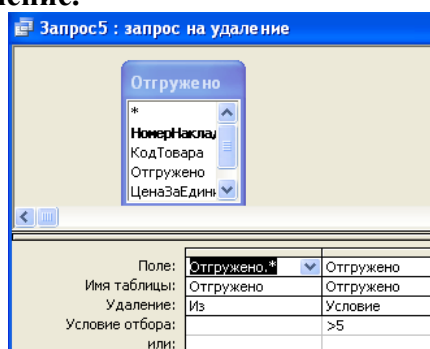
Рис. Окно создания запроса на обновление

2. Запрос на удаление.

Создать запрос на удаление записей из таблицы *Отгрузка*: количество отгруженного товара больше 5. (см. рис.).

Для этого:

- а) добавьте таблицу *Отгружено* в запрос, выберите все поля, щелкнув по * («*» добавляет в запрос все поля таблицы)
- б) добавьте поле **Отгружено**
- с) щелкните на панели инструментов по кнопке . Или выполните команду **Запрос Удаление**.



В результате запроса из таблицы *Отгружено* удалятся записи, соответствующие заданному условию.

Пример 3. Создание перекрестного запроса

Перекрестные запросы служат для более компактного отображения информации. В перекрестном запросе отображаются результаты статистических расчетов (такие, как суммы, количество записей, средние значения), выполненных по данным из одного поля. Эти результаты группируются по двум наборам данных в формате перекрестной таблицы. Первый набор выводится в левом столбце и образует заголовки строк, а второй - выводится в верхней строке и образует заголовки столбцов. Создадим перекрестный запрос, в котором в строках выводятся *фамилии врачей*, в столбцах - *дни приема*, в результирующих ячейках подсчитывается количество приемов в каждый из дней.

- 1) В режиме **Конструктора** создайте новый запрос, добавьте таблицу «**Журнал**».
- 2) Перетащите поля «*Дата приема*», «*ФИО Врача*», «*ФИО Пациента*» в первые три столбца бланка.
- 3) Выберите команду меню **Запрос Перекрестный**. Заголовок окна запроса *Запрос1: на выборку* изменится на *Запрос1: перекрестный запрос*. Кроме того, в бланк запроса будут добавлены строки *Перекрестная таблица* и *Групповая операция*, в которую во всех столбцах автоматически вводится операция **Группировка**.
- 4) Выберите в списке ячейки *Перекрестная таблица* столбца «*Дата приема*» значение **Заголовки столбцов**. Для столбца «*ФИО Врача*» - **Заголовки строк**.
- 5) Выберите в списке ячейки *Групповая операция* столбца «*ФИО Пациента*» значение **Count**. В ячейке *Условие отбора* – **Значение** (Рис.).

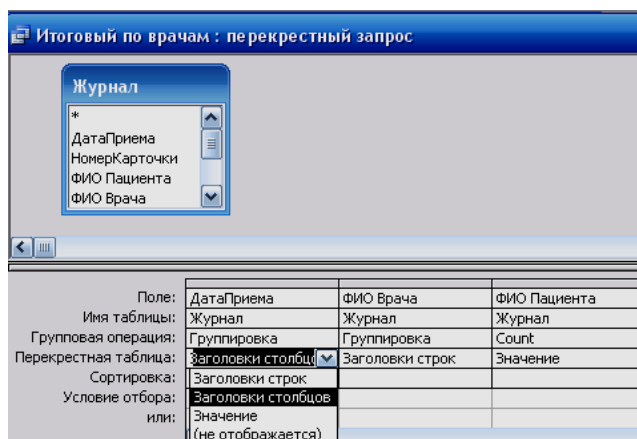


Рис. Перекрестный запрос в режиме **Конструктора**

Пример 4. Создание сводных таблиц и диаграмм.

Сводные таблицы и диаграммы предназначены для интерактивного анализа данных различной сложности и объема. Они позволяют произвести сравнительный анализ данных простыми и удобными средствами. Сводные таблицы являются многомерными и это позволяет представлять данные, сгруппированные по различным координатам.

1. Разработка сводной таблицы для таблицы базы данных. Рассмотрим таблицу «Журнал» в режиме сводной таблицы. Чтобы открыть таблицу в этом режиме, выберите её в окне базы данных, нажмите кнопку «Открыть», а затем для перехода из режима таблицы в режим сводной таблицы выберите на панели **Вид Сводная таблица**. При первом открытии в ее рабочую область не включено не одного поля таблицы (см. рис.):

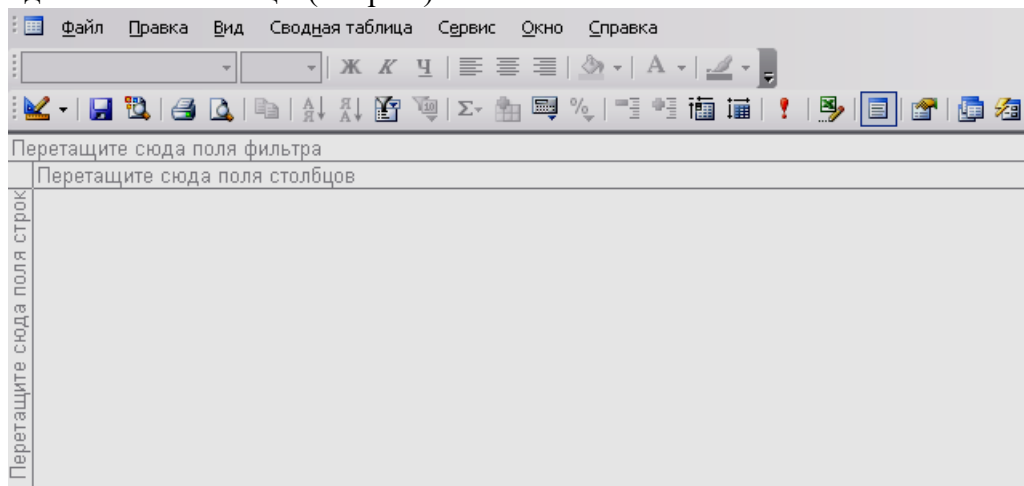


Рис. Пустой макет сводной таблицы *Журнал* и список полей источника записей

Пусть необходимо подсчитать количество выполненных услуг каждым врачом за каждый день (за определенный период). Если список полей этой таблицы не отображается, нажмите на панели инструментов пиктограмму **Список полей**.

- Уникальные значения фамилии врачей составят заголовки строк.

- Перетащите в эту область поле *ФИО Врачей*.

- В область полей столбцов перетащите поле *ДатаПриема*.

- Пусть в нашем примере фильтрация составит по должностям. Перетащите в область фильтра – поле *ДолжностьВрача* (см. рис.).

ДолжностьВрача	ДатаПриема	01.01.2011	02.01.2011	03.01.2011	04.01.2011	06.01.2011	09.01.2011	01.02.2011	09.02.2011	Общие итоги
Валиулина Д.К.	Количество	1	2							
Галимянова О.А.	Количество	5							1	
Дюкина Я.В.	Количество	4								
Елизарова Е.П.	Количество						2	2		
Жуков А.Е.	Количество		2							
Карбышев Г.О.	Количество		2				3	1		
Леонидова М.Р.	Количество				3			2		
Накипов Н.Д.	Количество		2					2		
Никоненко П.С.	Количество					2	1			
Родионова Ф.Я.	Количество			1						
Сидорович М.Л.	Количество					4		1		
	Количество					2				
Халимов И.Ш.	Количество					2				1
Харитоновна Н.Н.	Количество						4			
Шабрнева Ф.Н.	Количество							1		2
Общие итоги	Количество									

Рис. Сводная таблица с фамилиями врачей и датами приема

Для подсчета итоговых значений щелкните в области данных правой кнопкой мыши и выберите **Автовычисления Сумма** (см. рис.).

ФИО Врача	Количество	Количество	Количество	Количество	Количество
Валиуллин Д.К.	1	2			
Галимянова О.А.	1	2			
Дюкина Я.Б.	5				
Елизарова Е.П.	4				
Жуков А.Е.	2				
Карбышев Г.О.	2				
Леонидова М.Р.	3				
Накипов Н.Д.	2				

Рис. Формирование итоговых значений.

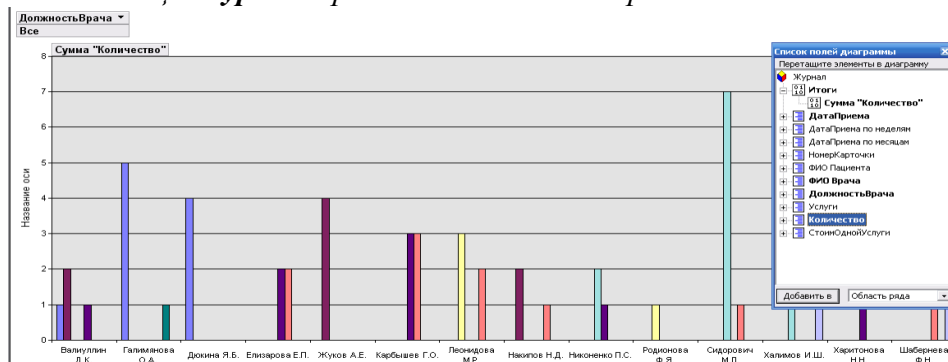
В результате получим таблицу с итоговыми значениями (количество выполненных услуг) и суммарной величиной этих значений. (см. рис.).

ФИО Врача	01.01.2011	02.01.2011	03.01.2011	04.01.2011	06.01.2011	09.01.2011	01.02.2011	09.02.2011	Общие итоги
Валиуллин Д.К.	1	2				1			4
Галимянова О.А.	1	2					1		6
Дюкина Я.Б.	5								4
Елизарова Е.П.	4				2	2			4
Жуков А.Е.		2	2		2	2			4
Карбышев Г.О.		2	2						6
Леонидова М.Р.			3		3	3			5
Накипов Н.Д.		2	3		2	1			3
Никоненко П.С.		2				1			3
Родионова Ф.Я.			1		2	1			1
Сидорович М.Л.			1		4		1		8

Рис. Сводная таблица с вычисляемыми полями.

Также можно получить сводную диаграмму по таблице, выбрав на панели инструментов Вид Сводная диаграмма. (см.рис.).

Рис. Таблица Журнал в режиме сводной диаграммы.



Задание для лабораторной работы

Для базы данных из лабораторной работы №1 создать:

1. простые запросы на выборку без фильтрации;
2. простые запросы с сортировкой данных;
3. простые запросы на выборку с фильтрацией данных;
4. запросы на выборку с параметром;
5. запросы с помощью построителя выражений;
6. запросы с вычисляемыми полями;
7. запросы на обновление записей в таблицах базы данных;
8. перекрестные запросы, применив групповые операции;
9. сводную таблицу с вычисляемыми полями и диаграмму к этой таблице;

ЛАБОРАТОРНАЯ РАБОТА №3. Системы управления базами данных MS

ACCESS: работа с формами.

Цель лабораторной работы: Знакомство с пакетом Microsoft Office Access. Приобретение навыков и умений при работе с базой данных, в процессе создания форм с применением мастера и конструктора форм.

Методические указания

РАБОТА С ФОРМАМИ

MS Access 2010 дает нам возможность ввода и редактирования данных непосредственно в режиме таблицы. Однако для конечного пользователя программного комплекса этого явно недостаточно — из-за его низкой квалификации. Пользователь должен работать с законченным программным продуктом. Его не интересует ни реляционная модель данных, ни то, что информация хранится в таблицах, и что существуют первичные ключи и триггеры. Он желает знать только одно: щелчком по какой кнопке он добьется желаемого результата. Так предоставим же ему эту возможность. Форма MS Access 2010 — это объект базы данных, который можно использовать для ввода, изменения или отображения данных из таблицы или запроса. Форма может использоваться как стартовая точка вашего приложения. Для автоматизации часто выполняемых действий формы содержат так называемые элементы управления, с помощью которых осуществляется доступ к данным. Формы можно рассматривать как окна, через которые пользователи могут просматривать и изменять базу данных. Рационально построенная форма ускоряет работу с базой данных, поскольку пользователям не требуется искать то, что им нужно. Внешне привлекательная форма — достойный элемент интерфейса. Она делает работу с базой данных более приятной и эффективной, кроме того, она может помочь в предотвращении неверного ввода данных. В MS Access 2010 предусмотрен ряд средств, помогающих быстро создавать формы, а также новые типы форм и функциональные возможности, благодаря которым база данных становится более практичной.

Форма — это объект базы данных, который можно использовать для ввода, изменения или отображения данных из таблицы или запроса. Формы применяются для управления доступом к данным, например для определения того, какие поля или строки данных должны отображаться. Эффективная форма ускоряет работу с базой данных, так как пользователям не требуется искать нужные сведения. Привлекательная форма делает работу с базой данных не только эффективной, но и более приятной. Кроме того, формы могут предотвращать ввод неверных данных.

С одной стороны, формы позволяют пользователям вводить данные в таблицы базы без непосредственного доступа к самим таблицам. С другой стороны, они позволяют выводить результаты работы запросов не в виде скупых результирующих таблиц, а в виде красиво оформленных форм.

Форма представляет собой некий электронный бланк, в котором имеются поля для ввода данных. В Форме каждое поле можно разместить в точно заданном месте, выбрать для него цвет и заливку. В Форму можно помещать вычисляемые поля. OLE-объекты можно увидеть только в форме или отчете. В Форме намного проще работать с большими текстами поля типа МЕМО в текстовом окне с полосами прокрутки.

Формы Access являются многофункциональными; они также позволяют выполнять задания, которые нельзя выполнить в режиме таблицы. Например, формы позволяют

производить проверку корректности данных, содержащихся в таблице; позволяют вычислять значения и выводить на экран результат.

Автоматическое создание формы на основе таблицы

В направлении полной автоматизации работы с данными резко улучшает положение дел такая замечательная способность MS Access, как автоматическое создание форм. Перейдите на вторую вкладку ленты Создание в раздел Формы (рис. 1). В нашем распоряжении несколько способов отображения информации из таблиц в формах:

- создание формы с помощью инструмента Форма;
- создание формы при помощи инструмента Пустая форма;
- создание Web-формы, в которой отображаются несколько записей, при помощи инструмента Несколько элементов;
- создание разделенной формы при помощи инструмента Разделенная форма;
- создание формы в виде сводной диаграммы при помощи инструмента Сводная диаграмма;
- создание формы в виде сводной таблицы при помощи инструмента Сводная таблица.

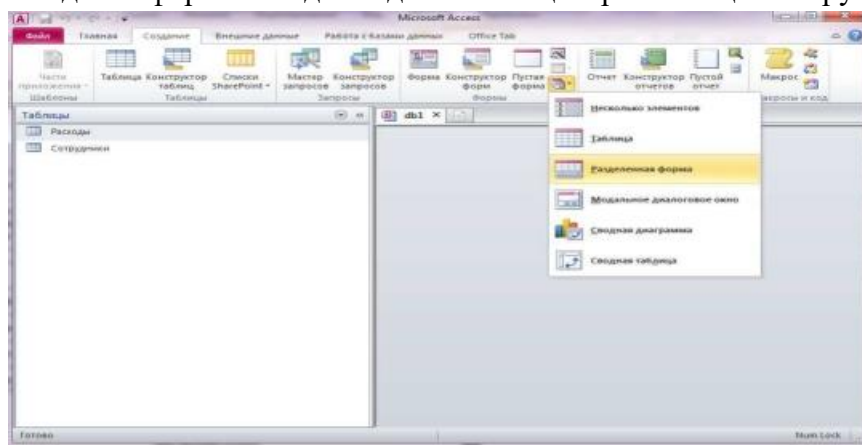


Рис. 1. Раздел Формы вкладки Создание ленты Access 2010

Для создания формы с помощью инструмента Форма в области навигации выберите таблицу с данными, которые должны отображаться в форме, и сделайте щелчок мышью по пиктограмме этого инструмента. MS Access 2010 создаст форму и отобразит ее в режиме макета. В режиме макета можно внести изменения в структуру формы при одновременном отображении данных. При необходимости можно настроить размер полей в соответствии с данными. Созданная форма со стандартной Access-линейкой навигации по записям готова к работе.

Заслуживает особого внимания инструмент Разделенная форма. Созданная им форма одновременно отображает данные в режиме формы и в режиме таблицы.

Эти два отображения связаны с одним и тем же источником данных и всегда синхронизированы друг с другом. При выделении поля в одной части формы выделяется то же поле в другой части. Данные можно добавлять, изменять или удалять в каждой части формы. Работа с разделенной формой дает преимущества обоих типов формы в одной форме. Например, можно воспользоваться нижней (табличной) частью формы, чтобы быстро найти запись, а затем просмотреть или изменить запись в верхней части формы. Для отображения нужных записей можно воспользоваться их сортировкой. Работают и все другие стандартные инструменты MS Access. На рис. 2 показана форма, созданная на основе таблицы.

ЦельРасходов	СуммаЗатрат	МетодОплат	Описание	Код сотруд	Номер Раск
Бензин	12 000,00р.	Безналичный	командировка	214	001
Зарплата	27 000,00р.	Наличный	зароботная плата	324	002
Коммунальные услуги	2 700,00р.	Безналичный	ком. Услуги	324	003
Командировочные услуг	7 000,00р.	Наличный	Командировка	533	004
Реклама	11 200,00р.	Безналичный	Рекламный проект	533	005
Ремонт офиса	120 000,00р.	Безналичный	Ремонт офиса	533	006
Закупка оборудования	78 000,00р.	Безналичный	закупка оборудования	755	007

Рис. 2. Результат работы инструмента Разделенная форма

Применение мастера для создания формы

MS Access 2010 имеет в своем арсенале еще одно средство для быстрого создания формы — мастер форм. С его помощью можно создавать формы как на основе одной таблицы или запроса, так и на основе нескольких связанных таблиц. Освоить работу с мастером — хорошая идея, которая приведет к значительной экономии времени разработчика, но это не значит, что мастера — это всегда лучший способ. Более солидные результаты дает создание формы с помощью мастера с последующим усовершенствованием ее в режиме конструктора.

Мастер форм разбивает процесс создания формы на несколько этапов. На каждом из них выбираются определенные параметры в предложенном диалоговом окне. Если на одном из этапов сделана ошибка и необходимо изменение уже выбранных параметров, то мастер всегда позволяет вернуться к предыдущему шагу.

1. Для запуска мастера форм выберите пункт Мастер форм, расположенный на вкладке ленты Создание в разделе Формы (см. рис. 1).

2. Появится первое окно мастера (рис. 3). Раскройте поле со списком Таблицы и запросы. Выберите в нем таблицу.

Рис. 3. Первый шаг работы мастера форм

3. В поле со списком Доступные поля отображены все поля выбранной таблицы. Выберите только те из них, которые следует отобразить в создаваемой форме. Используйте для этого кнопку. Чтобы добавить все поля из таблицы, примените кнопку.

4. Для перехода ко второму шагу работы мастера форм нажмите кнопку Далее. Появится второе окно мастера (рис. 4).

Существует несколько видов форм в зависимости от представления на них данных. Некоторые из них: в один столбец, ленточный, табличный, выровненный, сводная таблица, сводная диаграмма. Мастер предлагает нам выбрать один, но только из четырех.

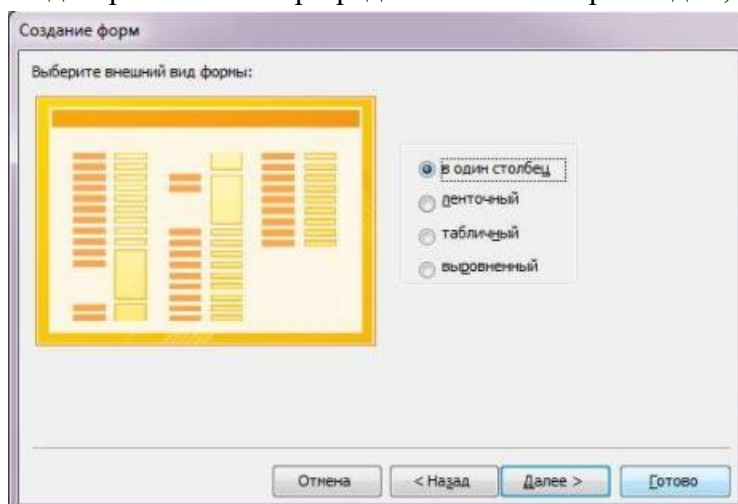


Рис. 4. Выбор внешнего вида формы

5. На третьем шаге требуется указать название формы и выбрать опцию переключателя Дальнейшие действия. Опций две:

- Открыть форму для просмотра и ввода данных;
- Изменить макет формы.

Если вы хотите внести свои изменения в форму, созданную мастером, то выберите Изменить макет формы и нажмите кнопку Готово.

В результате мы получили форму в соответствии с выбранными параметрами. На рис.5 представлена форма в режиме формы. Она готова к использованию, но в таком виде лучше ее не включать в состав программного комплекса. Требуется определенная доработка. Ее можно выполнить в режиме конструктора форм.

Рис. 5. Результат работы мастера форм

Любая форма, так же как и таблица базы данных MS Access 2010, может быть выведена на печать. Работа с формами может вестись в пяти режимах:

- в режиме формы;
- в режиме конструктора;
- в режиме таблицы;
- в режиме сводной таблицы;
- в режиме сводной диаграммы.

Для выбора режима работы с формой найдите ее в области навигации и сделайте двойной щелчок мышью по ее имени. Форма откроется в режиме формы. Этот режим является основным. В нем всегда работают рядовые пользователи программного комплекса (просмотр записей, редактирование, добавление и удаление). Сделайте щелчок по форме правой кнопкой мыши. Появится контекстное меню. В нем вы найдете все вышеперечисленные режимы.

Распечатать форму можно, находясь в любом режиме работы с ней. Внешний вид распечатанной формы всегда соответствует текущему режиму. Распечатка содержит столько «снимков» формы, сколько записей в соответствующей таблице. Для вывода на печать сделайте щелчок по главной кнопке MS Access 2010 — кнопке Файл. В открывшемся меню выберите пункт Печать. Чтобы увидеть, в каком виде форма будет на бумаге, перейдите в режим предварительного просмотра.

Создание простой формы в режиме конструктора

В этом разделе мы создадим форму в режиме конструктора, размещая в ней элементы различных типов. Перед вами самый распространенный вид формы, созданной в MS Access 2010.

Откроем ранее созданную форму в режиме конструктора форм. Для этого в области навигации выберите раздел Формы. Щелкните правой кнопкой мыши по форме, в появившемся контекстном меню выберите пункт Конструктор (рис. 6).

Рис. 6. Форма открыта в режиме конструктора

На экране появятся три дополнительные вкладки главной ленты MS Access 2010: Конструктор, Упорядочить и Формат, а также форма в режиме конструктора. На вкладке

ленты в разделе Элементы управления расположена панель инструментов. Она предназначена для размещения в форме выбранных элементов и содержит их стилизованные изображения. Функции этих кнопок будут рассмотрены в следующих разделах.

В разделе ленты Сервис — пять элементов. Выберите элемент Страница свойств, и окно свойств появится на экране дисплея. Для этих же целей можно применить клавишу **F4**. Каждый объект в MS Access, включая непосредственно базу данных, имеет свойства. Имеются различные категории свойств формы.

В MS Access 2010 они представлены на пяти вкладках:

- Макет — свойства, которые принадлежат способу отображения объекта;
- Данные — свойства, которые принадлежат данным объекта, независимо от того, каким способом они получены;
- События — свойства, которые принадлежат событиям и связанным с ними процедурам;
- Другие — свойства, которые принадлежат характеристикам объекта или его признакам;
- Все — все категории и свойства объекта.

Подготовка к конструированию

Теперь, когда вы в общих чертах знаете о том, с чем придется иметь дело, предлагаю начать конструирование формы с самого начала, останавливаясь подробно на каждом создаваемом элементе.

1. Выберите вторую вкладку ленты MS Access 2010 — Создание.
2. Сделайте щелчок левой кнопкой мыши по значку Конструктор форм. Появится новая пустая форма и три дополнительные вкладки: Конструктор, Упорядочить и Формат. Активной будет вкладка Конструктор.
3. Выберите на этой вкладке значок Страница свойств. Появится Окно свойств. Активной в этом окне должна быть вторая вкладка Данные.
4. Перейдите на первую строчку Источник записей и сделайте щелчок по кнопке .
5. Поле со списком раскроется. В нем будут перечислены все таблицы, входящие в текущую базу данных.
6. Выберите таблицу нужную вам таблицу. Из этой таблицы и будет отображать данные наша первая форма.
7. В окне свойств перейдите на вкладку Макет. Установите свойство Кнопки размеров окна в Отсутствуют. Если пользователь в процессе работы с формой раскроет ее окно на весь экран, то для доказательства вашего профессионального мастерства придется выполнить пересчет координат расположения элементов в форме, а это очень трудоемкое занятие.

На рис. 7 приведен вид окна MS Access 2010 целиком. Система готова к построению формы!

Каждому полю таблицы в форме соответствует определенный элемент.

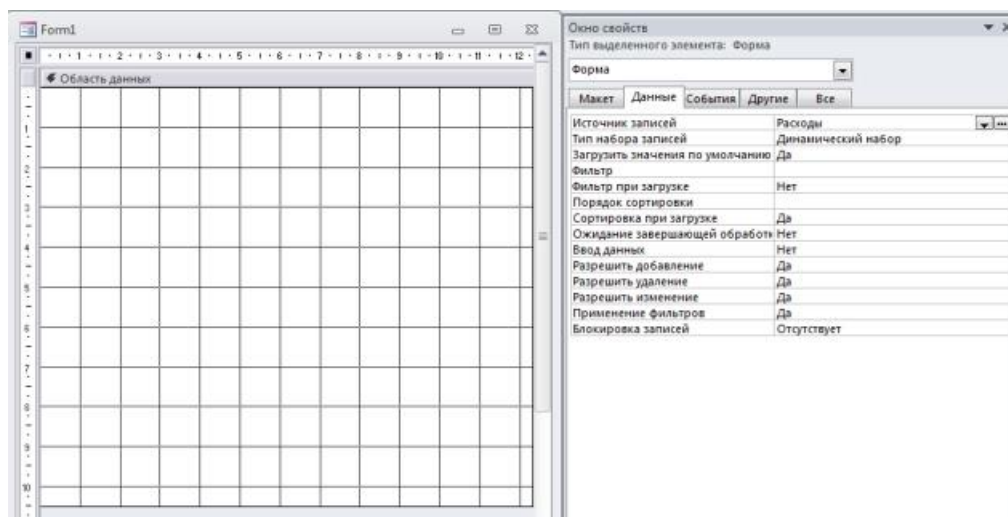


Рис. 7. Создание заготовки для построения формы

Изменение цвета формы

Цвет фона формы является основным параметром, определяющим ее внешний вид. Существует множество причин, в силу которых разработчики изменяют цвет формы. Некоторые меняют цвет, чтобы сделать ее просто более привлекательной. Другие — чтобы выделить группу объектов. Третьи — для условного форматирования данных в полях. При выборе цвета фона всегда нужно помнить о том, для каких целей предназначена создаваемая форма и как она будет использоваться. Не стоит забывать и о том, что программный комплекс будет находиться целый день перед глазами работника. Не следует отдавать предпочтение ярким цветам.

Для изменения цвета фона формы или объекта:

1. Выделите раздел формы, например, Область данных или объект, цвет фона которого нужно изменить, щелкнув левой кнопкой мыши по заголовку раздела или по элементу формы.
2. В окне свойств появятся значения всех свойств этого объекта. Перейдите на вкладку Макет.
3. Щелкните левой кнопкой мыши по свойству Цвет фона. В правой части строки свойства появится кнопка. Нажмите ее. Откроется окно выбора цвета.
4. Если стандартных цветов этого окна недостаточно, сделайте щелчок мышью по строке Другие цвета. Откроется окно Цвета с двумя вкладками: Обычные и Спектр. Перейдите на вторую вкладку и сделайте свой выбор.

Цвет фона раздела формы MS Access 2010 устанавливается независимо от цвета фона других разделов. Чтобы изменить цвет всех разделов, задайте новый цвет для каждого из них отдельно.

Цвет фона формы не влияет на цвет фона элементов управления, размещенных в форме.

Изменение фонового рисунка формы

Если вас не устраивает однородный цвет фона формы, можно в качестве фона использовать рисунок. MS Office Access 2010 «понимает» рисунки, хранящиеся во всех известных на момент выхода в свет этой версии графических форматах.

Чтобы задать фоновый рисунок для формы:

1. В режиме конструктора выделите всю форму. Для этого сделайте щелчок левой кнопкой мыши по квадратику в левом верхнем углу конструктора форм. В центре этого квадрата появится метка

2. Сделайте доступным окно свойств формы. Для этого на вкладке Конструктор главной ленты MS Access 2010 в разделе Сервис выберите значок Страница свойств.

3. Перейдите на первую вкладку Макет.

4. Выберите свойство Рисунок. Во второй колонке этого свойства увидите его значение: Отсутствует. Сделайте щелчок левой кнопкой мыши по этому свойству. Появится кнопка.

5. Активируйте ее щелчком мыши. Откроется окно Выбор рисунка.

6. Сделайте свой выбор и закройте окно щелчком мыши по кнопке **ОК**.

Действие фонового рисунка распространяется, в отличие от цвета фона, на все разделы формы. Для его удаления просто удалите все символы из текстового поля значения свойства.

Используя свойство Масштабы рисунка можно установить его размеры. Для вывода рисунка в исходном виде в раскрывающемся списке этого свойства выберите значение Обрезать. При выборе значения Увеличить произойдет растяжение или сжатие рисунка с сохранением пропорций до максимально возможных размеров, при которых не происходит обрезка рисунка. При выборе значения Растянуть рисунок будет сжат до размеров формы. Изменение пропорций рисунка в этом случае неизбежно.







Панель инструментов Элементы управления вкладки Конструктор















Панель инструментов Элементы управления вкладки Конструктор в MS Access 2010 содержит кнопки, предназначенные для разработки форм (рис. 8). В таблице описано назначение этих кнопок.



Рис. 8. Панель инструментов Элементы управления

Назначение кнопок панели инструментов
Элементы управления

Кнопка	Описание
	Кнопка Выбрать. Выделение прямоугольных областей рукописных штрихов, фигур и текста
	Кнопка Поле. Используется для отображения, ввода и изменения данных в источнике записей формы или отчета для вывода результатов вычислений, а также для приема данных, вводимых пользователем
	Кнопка Надпись. Создает элемент управления, в котором в форме или отчете выводится поясняющий текст. Надписи могут содержать гиперссылки. MS Access автоматически присоединяет подписи к создаваемым элементам управления
	Кнопка Кнопка. Создает элемент управления для вызова другой формы, отчета, макроса, процедуры или функции VBA
	Кнопка Вкладка. Применяется для создания формы с несколькими вкладками. На вкладку можно добавлять другие элементы управления
	Кнопка Гиперссылка. Применяется для создания ссылки на Web-страницу, рисунок или адрес электронной почты

	Кнопка Группа переключателей. Используется для размещения набора флажков, переключателей или выключателей
	Кнопка Вставить разрыв страницы. Применяется для указания начала нового экрана в форме или в отчете
	Кнопка Поле со списком. Создает составной элемент управления, объединяющий поле и раскрывающийся список. Чтобы ввести значение в поле базовой таблицы, можно ввести значение в поле в элементе управления или выбрать значение в списке
	Кнопка Диаграмма. Отображает данные MS Access в форме или отчете в виде диаграммы
	Кнопка Линия. Используется в формах или отчетах для отделения особенно важных разделов формы или отчета
	Кнопка Выключатель. Создает отдельный элемент управления, присоединенный к логическому полю в базе данных MS Access 2010
	Кнопка Список. Создает список, допускающий прокрутку. Если форма открыта в режиме формы, то выбранное в списке значение можно ввести в новую запись или использовать для изменения существующей записи
	Кнопка Прямоугольник. Используется для создания графических объектов для привлечения внимания к важным данным в форме или отчете
	Кнопка Флажок. Создает отдельный элемент управления, присоединенный к логическому полю в базе данных. Флажок в пользовательском окне или входящий в группу параметров является свободным элементом управления
	Кнопка Свободная рамка объекта. Используется для отображения в форме или отчете свободного объекта OLE. Этот объект остается неизменным при переходе от записи к записи
	Кнопка Переключатель (или радиокнопка). Создает отдельный элемент управления, присоединенный к логическому полю в базе данных MS Access 2010
	Кнопка енная форма/отчет. Предназначена для вывода в форме или отчете данных из нескольких таблиц
	Кнопка Присоединенная рамка объекта. Предназначена для отображения в форме или отчете объектов OLE (набор рисунков). При переходе от записи к записи в форме или отчете выводятся разные объекты
	Кнопка Рисунок. Используется для отображения неизменяемого рисунка в форме или отчете. Рисунок не является объектом OLE. После размещения рисунка в форме его изменения не допускаются

В формах MS Access 2010 применяются три типа элементов управления.

- Присоединенные элементы управления, связанные с полем источника данных для формы. Это может быть поле таблицы, запрос и даже значение другого элемента управления текущей или любой другой формы. Присоединенные к таблице элементы отображают и позволяют изменить значение поля, с которым они связаны. Элементы, присоединенные к другим элементам, не могут изменить значения «донора». Самыми распространенными присоединенными элементами являются текстовые поля. Выключатели, переключатели и флажки связывают с логическим полем таблицы. Элемент OLE — с графическим объектом, видео- и звуковым файлом и т. д. Все присоединенные

элементы при «рождении» получают связанные с ними метки. Значение метки представляет собой значение свойства Подпись, относящегося к вкладке Макет. Метку всегда можно удалить.

- Свободные элементы управления не зависят от источника данных формы. Свободные текстовые поля используются для ввода данных, например, для получения значения, которое будет использоваться в выражении. Прямоугольники и линии — для оформления внешнего вида, а OLE — для добавления графики в форму или отчет. Не все свободные элементы имеют метки.

- Вычисляемые элементы управления используют в качестве источника данных в выражении. В выражениях могут использоваться как поля таблиц, так и свободные элементы.

Панель инструментов вкладки Упорядочить Панель инструментов

Упорядочить содержит кнопки и раскрывающиеся меню, которые облегчают выбор параметров форматирования. На рисунке 9 приведена только часть этой панели — Размер и порядок. Почти все пиктограммы и пункты этой панели изначально недоступны и визуально погашены. В нужный момент времени система MS Access обеспечивает к ним доступ. Рис.9 сделан в тот момент, когда в форме была выделена группа элементов. Доступны все пункты всех меню за исключением одного — Разгруппировать.

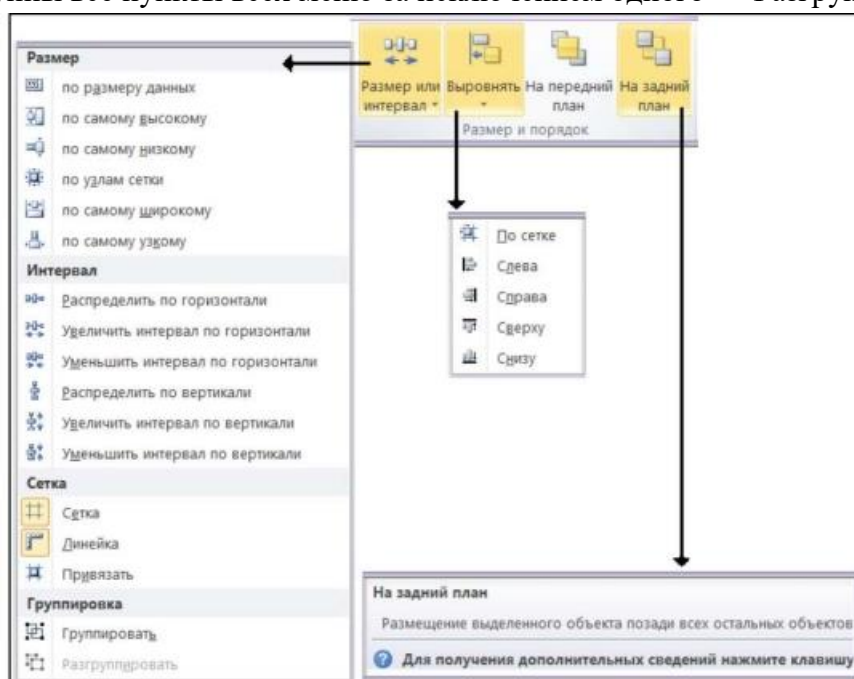


Рис. 9. Панель инструментов вкладки **Упорядочить**

Список свойств элементов управления формы

Каждый элемент управления формы в окне конструктора имеет свой список, который представляет собой окно с вкладками, где перечислены изменяемые свойства элемента управления. Для открытия списка свойств нужно щелкнуть на вкладке Сервис группы Конструктор – Страница свойств. Вкладки, которые появляются в окне свойств, остаются неизменными для любого элемента управления. Они делят свойства на следующие категории:

Макет. На этой вкладке находятся такие свойства, как Подпись, Полосы прокрутки, Тип границы. На этой же вкладке находятся свойства, которые определяют способ отображения дат и чисел с десятичным разделителем.

Данные. На этой вкладке указаны источники данных для формы или для поля, которое выделено в форме. При просмотре свойств всей формы в данной группе среди прочих появляются свойства Фильтр, Порядок сортировки и Применение фильтров. При просмотре свойств поля на данной вкладке среди прочих будут свойства Маска ввода, Значение по умолчанию и Доступ. Задаваемые здесь значения свойств перекрывают значения тех же свойств, установленных в окне конструктора таблицы.

События. На этой вкладке находятся свойства различного рода событий, связанных с выбранным элементом управления: После обновления, Вход, Выход и т. п. В общем случае свойства этой вкладки используются для запуска определенного макроса или выполнения кода на языке Visual Basic при возникновении какого-либо события, например, при щелчке на кнопке.

Другие. Содержит список свойств, которые не вошли в другие категории. Среди них есть такие свойства, как Строка меню, Контекстное меню, Идентификатор справки и т. п.

Все. Включает в себя свойства всех перечисленных выше категорий.

Создание поля со списком

В MS Access грамотно реализован выбор значения из очень длинных списков. Вы можете ввести один или несколько символов названия элемента списка — MS Access сам найдет в списке нужный элемент по первым символам и занесет его в текстовое поле.

1. Убедитесь, что в разделе Элементы управления вкладки Конструктор кнопка

Использовать мастера нажата. Если нет — выделите ее щелчком левой кнопки мыши.

В противном случае вы очень долго будете обвинять во всех грехах свой безупречно работающий компьютер. Построитель этого элемента так и не запустится.

2. Нажмите на панели элементов кнопку Поле со списком. Поместите указатель мыши над активной областью формы. Он превратится в значок поля со списком, снабженный крестиком в левом верхнем углу.

Прицельтесь получше и сделайте щелчок левой кнопкой мыши. Увидите следующую картину (рис. 10).

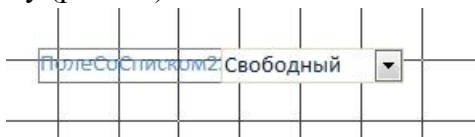


Рис. 10. В области данных формы появился первый элемент

3. Одновременно с этим откроется первое диалоговое окно мастера списков Создание полей со списком. Вам будет предложено выбрать источник значений (рис. 11). Сделайте щелчок левой кнопкой мыши по первой кнопке переключателя Объект «поле со списком»....

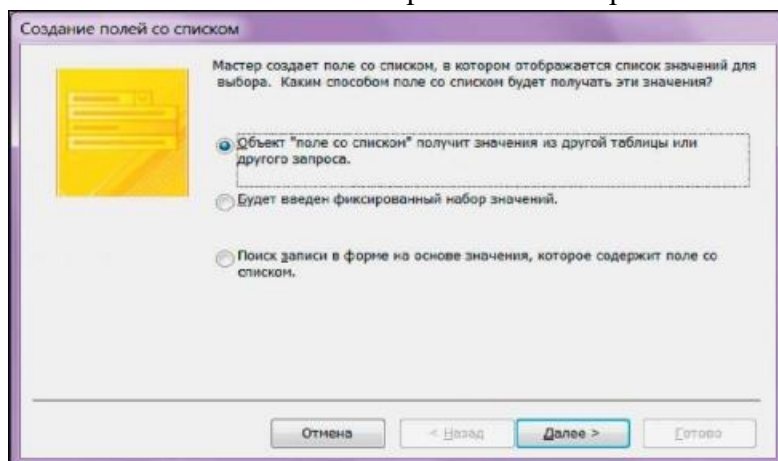


Рис. 11. Первое диалоговое окно мастера построения поля со списком

4. Нажмите кнопку Далее. Появится второе диалоговое окно, в котором отображен список всех таблиц, входящих в базу данных. Выберите таблицу и нажмите кнопку Далее.

5. В появившемся третьем диалоговом окне MS Access предлагает выбрать поля таблицы, значения которых будут отображаться в раскрывающемся поле со списком.

6. Четвертое диалоговое окно (рис. 12.) предлагает определить порядок отображения данных в поле со списком. Если строк в таблице более десятка, то сортировка — обязательный момент. Отсортировать записи можно максимум по четырем полям. Для текстовых полей порядок сортировки — по алфавиту.

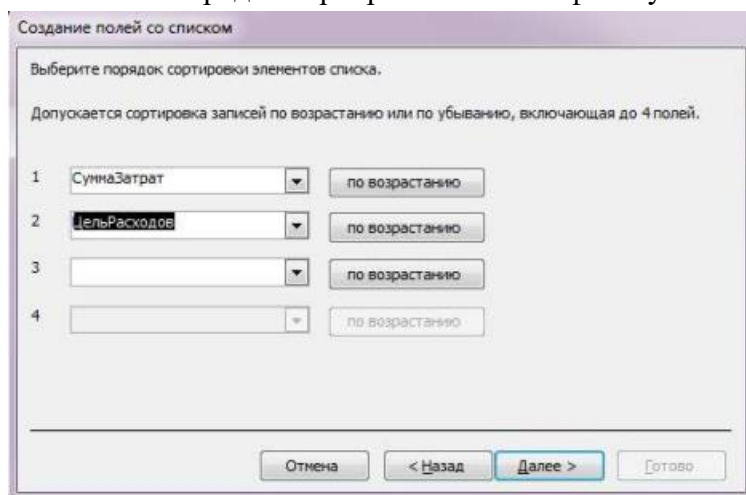


Рис. 12. Четвертое диалоговое окно мастера построения поля со списком

7. В пятом диалоговом окне поставьте флажок Скрыть ключевой столбец (рекомендуется). Пользователю ни к чему видеть, под каким номером в таблице стоит та или иная запись. Здесь же вам предоставлена возможность установить ширину колонок (рис. 13).

8. Нажмите кнопку Далее. Очень ответственный момент. Надо указать поле таблицы, в которое будет записываться ссылка на улицу из другой таблицы.

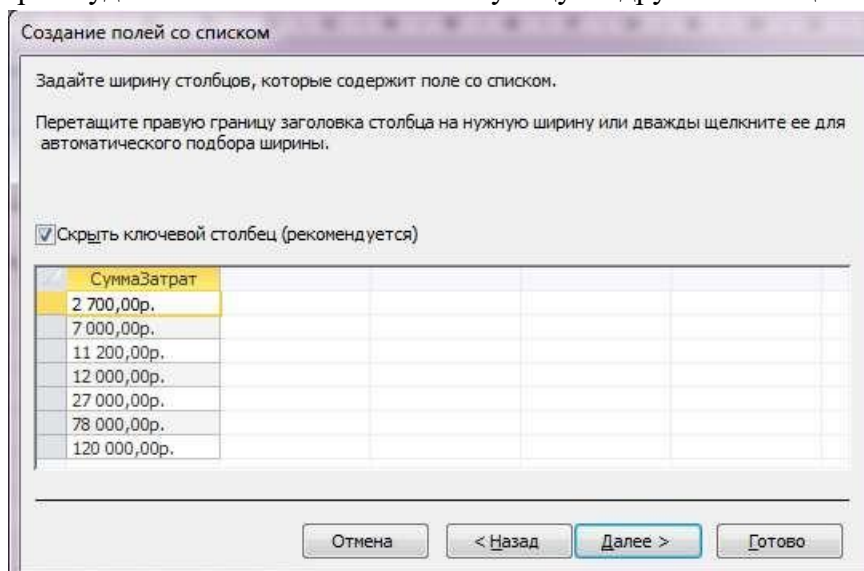


Рис. 13. Выбор ширины колонок раскрывающегося списка

9. В последнем диалоговом окне предлагается ввести надпись, которая будет стоять рядом с раскрывающимся списком. MS Access 2010 предоставляет пользователю массу возможностей для форматирования. Вот только основные из них.

- Во-первых, щелкните правой кнопкой мыши по надписи и выберите в раскрывшемся меню пункт **Размер**, а в появившемся подменю — пункт **По размеру данных**.

- Во-вторых, переместите поле со списком и его метку в нужное место на форме. Для этого поместите указатель мыши в любую точку на границе выделенного элемента, отличную от маркеров изменения размеров. Нажмите левую кнопку мыши и, не отпуская ее, перетащите элемент на новое место.

- В-третьих, более точно «выставить» элемент управления и его метку на форме можно при помощи клавиш-стрелок при нажатой клавише **Ctrl**.

Создание поля типа Флажок

Если в таблице находится поле логического типа, принимающее значение только да или нет, то использование флажка сделает форму более выразительной и удобной. Выделите в списке полей таблицы это поле и перетащите на форму. Остается изменить название метки. Щелкните по ней левой кнопкой мыши. Указатель превратится в текстовый курсор. Используйте клавиатуру для ввода надписи.

Этого же результата можно добиться и другим способом. Сделайте щелчок правой кнопкой мыши по созданной метке. Выберите в открывшемся меню пункт **Свойства**.

В появившемся диалоговом окне выберите вкладку **Макет**. Найдите в списке этой вкладки свойство **Подпись**. Введите текст, который хотите увидеть рядом с флажком.

Создание поля типа Группа переключателей

MS Access 2010 предлагает нам легкий, а с позиции интерфейса и более красивый путь решения задачи, когда значений поля немного и их количество в процессе эксплуатации программного комплекса меняться не будет. Порядок создания группы переключателей следующий.

1. Убедитесь, что кнопка с подсказкой «Использовать мастера» нажата. Если нет — «выделите» ее щелчком левой кнопки мыши. Нам понадобится работа построителя.

2. На этой же панели выберите пиктограмму **Группа переключателей**. Типичная ошибка начинающих — выбор пиктограммы **Переключатель**. Помните! Набором переключателей в MS Access группу переключателей сделать нельзя!

3. Поместите указатель мыши над активной областью формы. Он превратится в значок группы переключателей с крестиком в левом верхнем углу. Нажмите левую кнопку мыши и, удерживая ее в нажатом состоянии, переместите курсор по диагонали так, чтобы получилась рамка требуемого размера. Отпустите левую кнопку мыши. Автоматически запустится построитель группы переключателей.

4. Сделайте подписи у переключателей (рис. 14) и нажмите кнопку **Далее**. Теперь назовем переключатель, используемый по умолчанию (рис. 15).

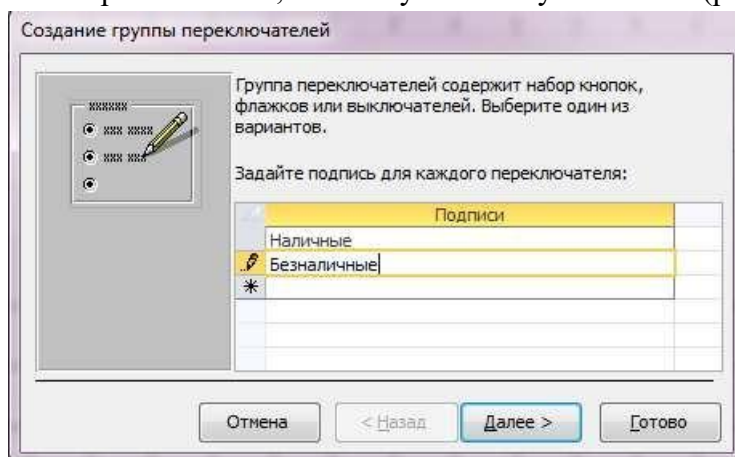


Рис. 14. Создание подписей у группы переключателей

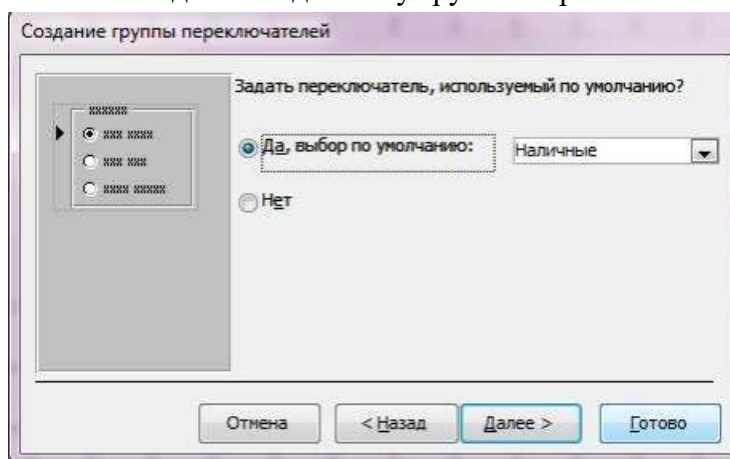


Рис. 15. Выбор значения по умолчанию

5. Третий шаг построителя посвящен привязке подписи к значению (рис.16). Если ничего не менять, то цифра 1 будет соответствовать наличные, 2 — безналичные.

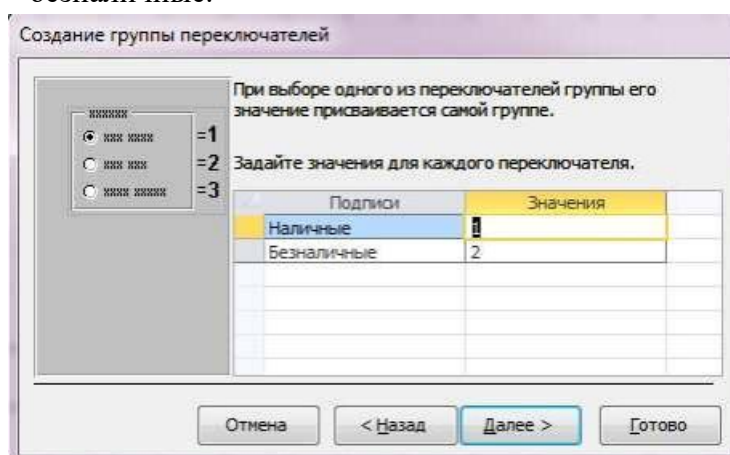


Рис. 16. Задание значений для каждого переключателя

6. На четвертом шаге (рис. 17) необходимо указать поле таблицы, в которое будет занесена выбранная цифра. Сделайте щелчок мышью по второй кнопке и не ошибитесь с выбором.

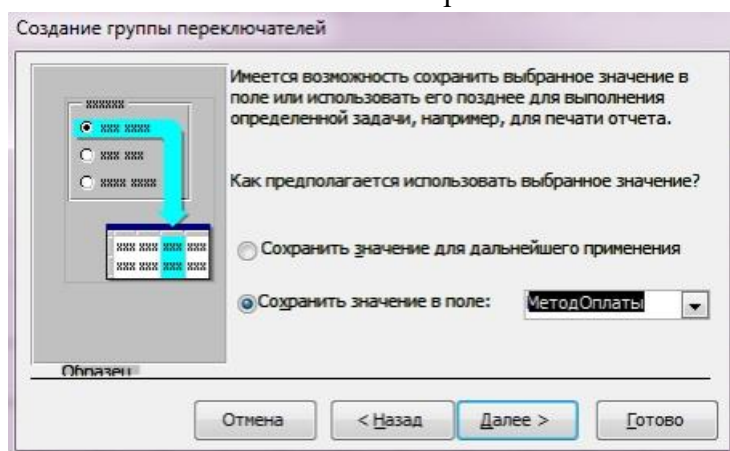


Рис. 17. Четвертый шаг работы построителя группы переключателей

7. Пятый шаг построителя (рис. 18) — оформление внешнего вида группы переключателей. Здесь нам предоставляется возможность увидеть, что будет в форме, если мы сделаем тот или иной выбор. Смелее выбирайте тип оформления и смотрите на левую часть окна (образец).

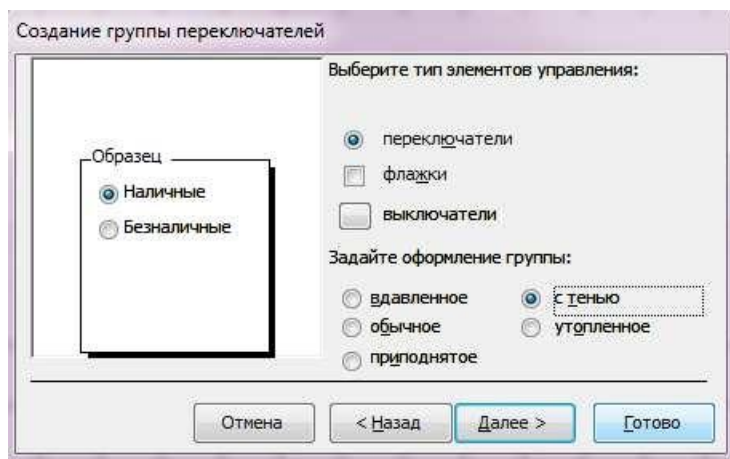


Рис. 18. Оформление внешнего вида группы переключателей

8. На последнем шаге необходимо ввести подпись для созданной группы переключателей. Если подпись в таком виде не нужна, то введите пустое значение, а в форму в нужном месте поместите элемент Надпись.

Использование кнопок

Кнопка является простейшим элементом управления формы и используется для запуска макроса. В Access благодаря макросам, запускаемым с помощью кнопок, можно выполнять самые различные задачи: открыть форму, распечатать отчет и т.п.

Добавление кнопки

При создании кнопки можно воспользоваться Мастером создания кнопок. Для этого необходимо выполнить следующие действия.

1. На панели Элементы управления щелкните на кнопке Кнопка.
2. Установите указатель на то место формы, где должна располагаться кнопка, и нажмите кнопку мыши. Протащите указатель, чтобы задать размеры кнопки. Когда кнопка мыши будет отпущена, откроется первое окно Мастера создания кнопок.
3. Выбранное на первом шаге действие определяет ход дальнейшего диалога с мастером. Например, рассмотрим создание кнопки открывающей форму Ввод даты.
4. На первом шаге Мастер предлагает выбрать то действие, которое предполагается связать с создаваемой кнопкой. В нашем примере – это команда Открыть форму из категории Работа с формами.

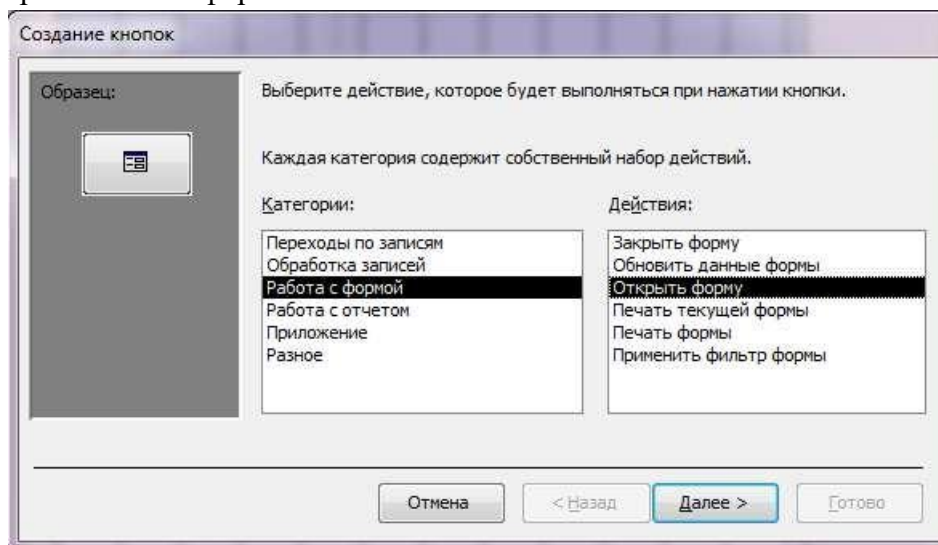


Рис.19. Окно выбора действия при нажатии кнопки

5. На следующем этапе необходимо выбрать тот объект, который должен открываться при нажатии кнопки. В нашем случае –это форма Ввод даты.

6. Очередной шаг мастера предлагает ввести тот текст или рисунок, который будет отображаться на кнопке.

7. На завершающем этапе Мастер предлагает ввести название кнопки. Понятное имя объекта может помочь в дальнейшей работе.

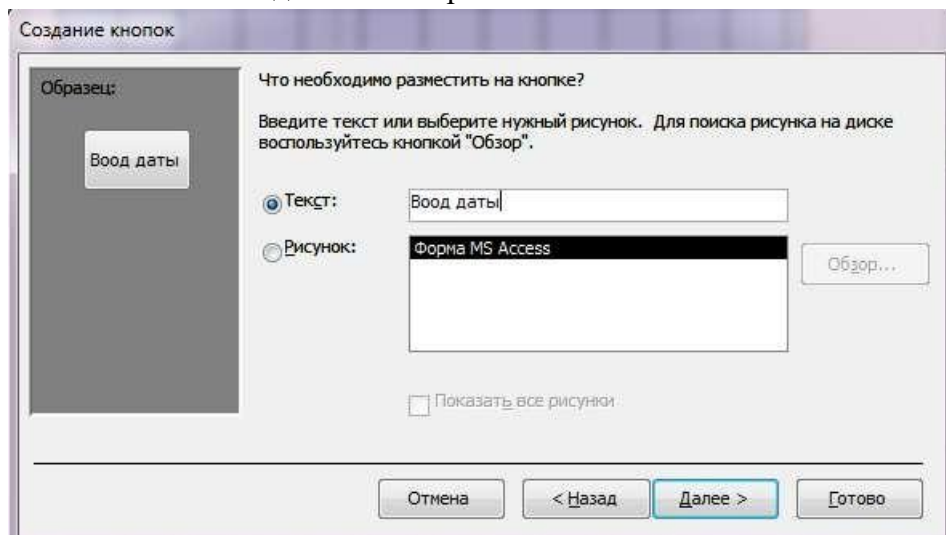


Рис.20. Окно выбора действия при нажатии кнопки

В случае необходимости связать кнопку с действием (макросом или программой), которое не предусмотрено в списках Категории и Действия Мастера создания кнопок, следует создать в форме кнопку и не связывать с ней действия. Для этого существует два способа:

1) запустить Мастер создания кнопок, но на первом шаге мастера щелкнуть на кнопке Отмена;

2) перед тем, как поместить в форму кнопку, убедитесь, что кнопка Мастера на панели инструментов не нажата.

Кнопка, созданная одним из этих методов, будет иметь надпись Кнопка¹¹ или подобную ей (число – это порядковый номер элемента управления в форме). Чтобы такая кнопка выполняла какие-либо действия, необходимо сначала написать для нее макрос или программу, а затем указать имя макроса или программы в строке свойства Нажатие кнопки.

Настройка кнопки

После добавления кнопки в форму иногда бывает необходимо изменить некоторые ее свойства:

- размеры кнопки можно изменять, как размеры любого другого элемента управления в форме. Для этого надо выделить кнопку щелчком, поставить указатель на один из маркеров выделенной кнопки и перетащить границу кнопки;
- для редактирования надписи на кнопке достаточно дважды щелкнуть на ней, внести изменения, а затем щелкнуть за пределами кнопки, чтобы их сохранить;
- чтобы изменить стиль текста надписи на кнопке надо щелчком выделить кнопку, а затем воспользоваться Инструментами конструктора форм Формат, либо внесите изменения в свойства кнопки на вкладке Макет;
- чтобы заменить или изменить макрос, или процедуру, запускающийся путем нажатия по этой кнопке, необходимо открыть вкладку События в списке свойств и изменить

значение в строке указанного свойства. Существует и другой способ: щелкнуть на строке Свойства и открыть раскрывающийся список, чтобы выбрать в нем необходимый макрос.

Изменение свойств кнопки

Каждая кнопка, представленная на панели инструментов, имеет свои свойства. Эти свойства можно использовать для задания внешнего вида кнопки. Для этого надо открыть диалоговое окно Настройка, щелкнуть на вкладке Панели инструментов и установить флажок на нужной панели. Затем щелчком правой кнопки мыши на нужной кнопке вызвать контекстное меню и пользуясь различными командами изменить свойства выбранной кнопки: выбрать (отредактировать) значок для кнопки, отобразить на кнопке текст вместо значка и т.п.

Использование элемента управления Набор вкладок

Список полей, которым мы пользовались, чтобы менять характеристики элементов управления, — это пример элемента управления со вкладками. В такой элемент управления входит несколько вкладок. Щелчком на одной из вкладок осуществляется переход на нее и устанавливаются необходимые для работы параметры. Можно создать форму, включающую в себя несколько вкладок, каждая из которых будет посвящена своему разделу данных.

Добавление в форму набора вкладок

Для добавления вкладок в форму используется следующая последовательность действий.

- Открыть форму в режиме конструктора.
- Освободить на бланке формы место под набор вкладок.
- Щелкнуть на кнопке Вкладка на вкладке Элементы управления.
- Щелкнуть на бланке формы в том месте, где должен появиться набор вкладок.

Добавление элементов управления на вкладку

После создания элемента управления Набор вкладок можно добавлять на каждую из вкладок другие элементы управления формы. Перетаскивать поля из списка полей; при создании новых элементов управления пользоваться инструментами панели элементов; добавлять на вкладку элементы управления посредством вырезания их и последующей вставки на новой вкладке.

Изменение количества вкладок

При добавлении в форму элемента управления Вкладка Access по умолчанию включает в набор две вкладки. Но их количество при необходимости можно изменить. Для этого надо щелкнуть правой кнопкой мыши в любом месте в пределах элемента управления Набор вкладок и выбрать в контекстном меню команду Добавить вкладку. Для удаления вкладки из набора вкладок надо щелкнуть правой кнопкой мыши на вкладке, которую необходимо удалить и выбрать в контекстном меню команду Удалить вкладку.

Добавление других элементов управления

В Access входит еще один набор элементов управления, которые называются элементами ActiveX и обеспечивают выполнение специальных функций, например, таких как отображение в форме данных электронных таблиц. Чтобы сделать эти элементы доступными, надо вывести на экран панель элементов и щелкнуть на кнопке Дополнительные элементы. На экране появится довольно обширный список элементов управления, которые можно включить в форму. За более конкретными сведениями по этим элементам обратитесь к Помощнику Office. Чтобы Помощник выполнил поиск

необходимых разделов Справочной системы, введите текст элемент ActiveX в качестве образца для поиска.

Изменение источника записей формы

Таблица или запрос, откуда поступают данные формы, называется источником записей формы. Очень часто необходимо использовать в форме, первоначально разработанной для одного источника, данные из другой таблицы или запроса. Изменить источник записей для формы можно следующим способом.

- Открыть форму в окне конструктора.
- Щелкнув два раза на области выделения формы, открыть ее список свойств.
- Перейти на вкладку Данные.
- Щелкнуть на строке свойства Источник записей и выбрать в раскрывающемся списке другую таблицу или запрос.
- Выбрать команду Сохранить как из меню Файл и дать измененной форме новое имя, если желательно оставить старую версию.

Если изменить свойство Источник записей и установить в нем ссылку на таблицу или запрос с совершенно иным набором полей, то при просмотре формы будет выдана серия сообщений об ошибке в связи с несоответствием типа элементов управления и полей. Поэтому либо можно удалить из формы ненужные поля, либо изменить у полей свойство Данные.

Создание подчиненной формы

Подчиненная форма — это форма, находящаяся внутри другой формы. Первичная форма называется главной. Подчиненные формы очень удобны для вывода информации из таблиц или запросов, связанных отношением «один-ко-многим». При использовании формы с подчиненной формой для ввода новых записей, текущая запись в главной форме сохраняется при входе в подчиненную форму. Это гарантирует, что записи из таблицы или запроса на стороне «многие» будут иметь связанную запись в таблице или запросе на стороне «один». MS Access автоматически сохраняет каждую запись, добавляемую в подчиненную форму, и никакие специальные приемы типа обработки события До вставки не требуются.

Порядок создания подчиненной формы следующий.

1. Откройте первичную форму в режиме конструктора.
2. Убедитесь, что на вкладке Конструктор главной ленты в разделе Элементы управления кнопка с подсказкой «Использовать мастера» нажата. Если нет — выделите ее щелчком левой кнопки мыши. Нам понадобится работа построителя подчиненных форм.
3. Нажмите на панели элементов кнопку подчиненная форма. Наведите указатель мыши на то место первичной формы, где вы планируете поместить левый верхний угол подчиненной формы. Указатель мыши превратится в значок подчиненной формы с крестиком в левом верхнем углу.
4. Нажмите левую кнопку мыши и, удерживая ее в нажатом состоянии, переместите курсор по диагонали так, чтобы получилась рамка требуемого размера. Отпустите левую кнопку мыши. Автоматически запустится построитель подчиненной формы.
5. Первый шаг работы мастера подчиненных форм — определение данных, которые надо включить в подчиненную форму. Установите переключатель Имеющиеся таблицы и запросы и щелкните по кнопке Далее. Появится окно для выбора таблиц и полей. Выберите все поля из таблицы (рис. 21).
6. Второй шаг — определение полей связи между главной и подчиненной формами.

Мастер подчиненных форм

Какие поля необходимо включить в подчиненную форму или отчет?

Допускается выбор полей из нескольких таблиц и запросов.

Таблицы и запросы
Таблица: Сотрудники

Доступные поля:

Выбранные поля:

Код сотрудника
Фамилия
Имя
Отчество
Должность
Год рождения
Адрес
Телефон

Отмена < Назад Далее > Готово

Рис. 21. Первый шаг работы мастера подчиненных форм

Мастер подчиненных форм

Самостоятельно определите поля связи между главной и подчиненной формами или выберите их из списка.

☒ Выбор из списка ☐ Самостоятельное определение

'Сотрудники' для каждой записи 'Расходы', поле связи 'Код сот' отсутствует

'Сотрудники' для каждой записи 'Расходы', поле связи 'Код сотрудника'

Отмена < Назад Далее > Готово

Рис. 22. Определение полей связи между главной и подчиненной формами

7. Третий шаг — выбор имени для подчиненной формы. Под этим именем она появится в списке форм базы данных. Введите это имя и щелкните по кнопке Готово. После проделанных манипуляций получим форму, изображенную на рис. 23.

Расходы сотрудников

Расходы

ЦельРасхода:

СуммаЗатрат: 12 000,00р.

МетодПлаты: 1

Описание: командировка

Код сотрудника: 214

Номер Расхода: 001

Метод оплаты: ☒ Наличные ☐ Безналичные

Данные о сотрудниках

Код сотрудника	Фамилия	Имя
214	Воробьев	Андрей

Записи: 1 из 1

Рис. 23. Подчиненная форма, созданная при помощи мастера

Задание для лабораторной работы

Для базы данных из лабораторной работы №1:

1. Создать необходимые однотабличные формы.
2. Создать подчиненную или связанную форму для связанных таблиц.
3. Оформить главную кнопочную форму.

При создании форм использовать дополнительные кнопки, вычисляемые поля.

ЛАБОРАТОРНАЯ РАБОТА №4. Системы управления базами данных MS ACCESS: создание отчетов и макросов.

Цель лабораторной работы: Знакомство с пакетом Microsoft Office Access. Приобретение навыков и умений при работе с базой данных, в процессе создания отчетов и макросов к таблицам и запросам.

Методические указания

Конечным продуктом большинства приложений баз данных является отчет. В Access отчет представляет собой специальный тип непрерывных форм, предназначенных для печати. Для создания отчета, который можно распечатать и распределить между потребителями, Access комбинирует данные в таблицах, запросах и даже формах. Распечатанная версия формы может служить отчетом.

Отчеты и формы имеют много общего. И все же между этими двумя типами объектов есть некоторые важные отличия. Одно из них заключается в том, что формы используются преимущественно для редактирования и просмотра данных. Отчеты тоже можно предварительно просмотреть, но их основное назначение – отобразить данные на печатной странице так, чтобы их было удобно читать.

Главное же отличие отчетов от форм в том, что отчеты обладают специальными средствами, позволяющими обобщать данные и проводить итоговые расчеты.

Основная сфера применения форм — обеспечение возможности просмотра отдельных или небольших групп связанных записей. Отчеты же представляют собой наилучшее средство отображения информации из базы данных в виде печатного документа. Разработка отчета очень похожа на разработку формы. Используется та же панель элементов (за исключением двух), тот же список полей и окно свойств. В этом разделе мы построим относительно несложный отчет, пройдя шаг за шагом всю цепочку его создания.

MS Access 2010 предоставляет пользователю возможность создавать множество различных отчетов любой степени сложности. Работа по созданию отчета всегда начинается с выбора источника, из которого будут извлекаться записи отчета. Отчет может представлять собой как простой список, так и подробную сводку данных, представленных в виде официального документа того или иного ведомства.

Однако в любом случае сначала определяют поля, которые должны войти в отчет, и в каких таблицах или запросах находятся эти поля.

Создание простого отчета

Стандартное средство MS Access Отчет — самый быстрый способ создания отчета, потому что с его помощью отчет формируется сразу же, без уточнения дополнительной информации. В отчете будут представлены все записи базовой таблицы или запроса. Для демонстрации этого средства создадим отчет по улицам города. Для этого отображения данных из основных таблиц оно не подходит. Его назначение — информация из таблиц-справочников.

1. В области переходов щелкните таблицу, на основе которой будем создавать отчет.

2. На вкладке Создание ленты MS Access 2010 в разделе Отчеты щелкните пиктограмму - Отчет. MS Access немедленно создаст отчет и отобразит его в режиме макета (рис. 1).

Цель/Расход	Сумма/Затрат	Метод/Платит	Описание	Код сотрудника	Номер Расхода
Бензин	12 000,00р.	1	командировка	214	001
Зарплата	27 000,00р.	2	заработная плата	324	002
Коммунальные услуги	2 700,00р.	1	ком. Услуги	324	003
Командировочные услуги	7 000,00р.	1	Командировка	533	004
Реклама	11 200,00р.	2	Рекламный проект	533	005
Ремонт офиса	120 000,00р.	2	Ремонт офиса	533	006
Закупка оборудования	78 000,00р.	1	закупка оборудования	755	007
	257 900,00р.				

Рис. 1. Создание отчета при помощи стандартного средства Отчет

3. После просмотра отчет можно сохранить, а затем закрыть и его, и источник записей — таблицу или запрос. В следующий раз при его открытии MS Access отобразит в нем самые последние данные из таблицы.

Полученный отчет очень далек от совершенства, он позволяет лишь быстро просмотреть базовые данные.

Вторая возможность — мастер отчетов — средство MS Access 2010, помогающее создать отчет на основании ответов, полученных на заданные пользователю вопросы. Мастер отчетов предоставляет больше возможностей относительно выбора полей для включения в отчет. При этом разработчик может указать способ группировки и сортировки данных, а также включить в отчет поля из нескольких таблиц или запросов, но только в том случае, если отношения между этими таблицами и запросами заданы заранее.

Создание отчета в режиме Мастер отчетов

При создании отчета с помощью Мастера отчетов надо приложить чуть больше усилий, чем при создании простого отчета. Но зато возрастает степень вашего участия в создании отчета. Мастер отчетов позволяет:

- выбрать поля из одной или нескольких таблиц и запросов;
- сгруппировать записи по определенным полям;
- отсортировать записи в отчете по четырем полям;
- вывести при необходимости итоговые величины, а также скрыть основанные на них подчиненные записи;
- выбрать макет отчета (ступенчатый, блок, структура и т.д.) и ориентацию (книжную или альбомную);

Для работы с Мастером отчетов на вкладке главной ленты MS Access 2010 Создание в разделе Отчеты найдите режим Мастер отчетов.

На первом этапе необходимо выбрать таблицы и запросы с указанием полей, которые будут использованы для создания отчета. Затем выбираются поля для группировки (двойным щелчком мыши). На следующем шаге Мастера устанавливается порядок сортировки записей в отчете. Также на этом этапе можно воспользоваться кнопкой Итоги, которая позволяет рассчитать сумму по показателям, максимум, минимум и среднее значения. При этом есть возможность показа итогового отчета как без подробных данных, так и с ними.

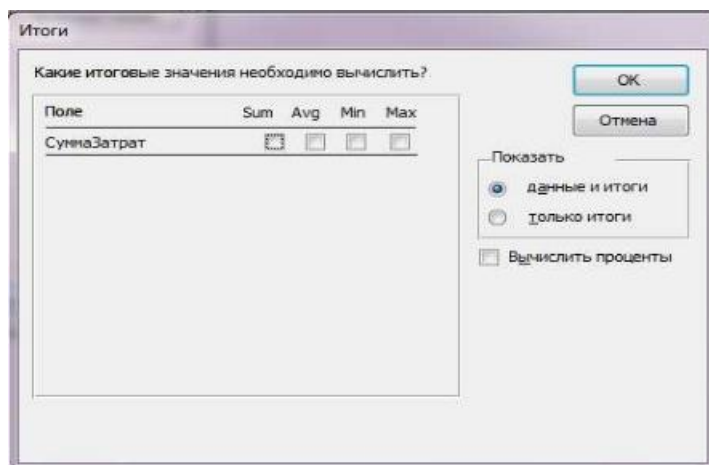


Рис.2. Установка итоговых параметров

Затем можно выбрать макет отчета, его ориентацию, стиль. Когда Мастер закончит свою работу, отчет будет открыт в окне предварительного просмотра.

Если оформление отчета требует некоторой доработки, то на последнем шаге работы Мастера можно выполнить команду Изменить макет отчета, которая запустит Конструктор отчетов, позволяющий внести необходимые доработки.

Создание отчета в режиме конструктора

В MS Access 2010 отчет разбит на разделы. Разделы отчета можно увидеть только в режиме конструктора. Чтобы созданные отчеты правильно работали, необходимо четко представлять назначение каждого раздела. Перечислим типы разделов и укажем назначение каждого из них.

- **Заголовок отчета.** Выводится на печать только один раз в начале отчета. В заголовок включается информация, обычно помещаемая на обложке: название отчета и дата. Заголовок отчета печатается перед верхним колонтитулом.

- **Верхний колонтитул.** Печатается вверху каждой страницы. Верхний колонтитул используется в тех случаях, когда нужно, чтобы название отчета повторялось на каждой странице.

- **Заголовок группы.** Размещается перед каждой новой группой записей. Используется для печати названия группы. Например, если отчет сгруппирован по зданиям, в заголовках групп можно указать их адрес.

- **Область данных.** Этот раздел печатается один раз для каждой строки данных из источника записей. В нем размещаются элементы управления, составляющие основное содержание отчета.

- **Примечание группы.** Печатается в конце каждой группы записей. Примечание группы можно использовать для печати сводной информации по группе.

- **Нижний колонтитул.** Печатается внизу каждой страницы. Используется для нумерации страниц и для печати постраничной информации.

- **Примечание отчета.** Печатается один раз в конце отчета. Примечание отчета можно использовать для печати итогов и другой сводной информации по всему отчету.

Подготовка к конструированию

На вкладке главной ленты MS Access 2010 Создание в разделе Отчеты найдите конструктор отчетов. После его запуска на ленте появятся четыре дополнительные вкладки: Конструктор, Упорядочить, Формат и Параметры страницы, а в центре экрана — окно конструктора отчетов. Новый пустой отчет содержит три раздела: верхний и нижний колонтитулы, между которыми находится область данных. Вы можете изменить размер любого раздела. Линейки с сантиметровыми делениями по верхнему и левому краям отчета помогают расположить данные на странице.

Если линейки отсутствуют, то для их появления на экране выберите вкладку главной ленты Упорядочить, в разделе Размер и порядок щелкните значок Размер или интервал. Появится меню. Найдите в нем в разделе Сетка пункт Линейка.

Верхний и нижний колонтитулы будут напечатаны соответственно сверху и внизу каждой страницы. Их можно убрать совсем с помощью пункта меню Обработка событий..., которое появится на экране, если сделать щелчок правой кнопкой мыши в любом месте отчета (рис. 3). Выберите в нем пункт Колонтитулы страницы. Расположенный чуть ниже пункт Заголовок/примечание отчета даст возможность создать заголовок, который будет напечатан только в начале отчета на первой странице.

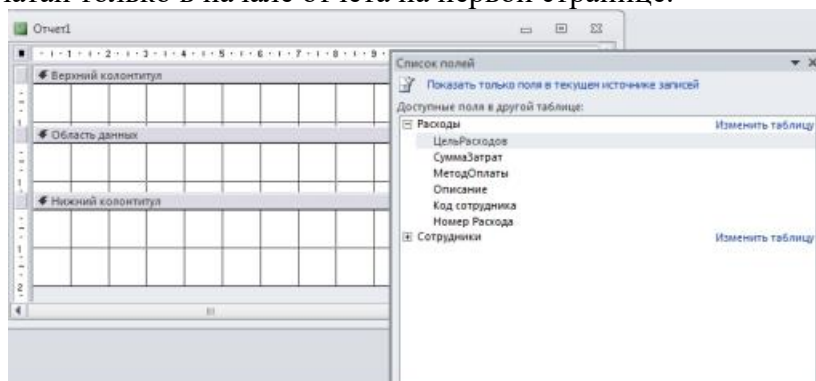


Рис. 3. Конструктор отчетов

Не забудьте про сетку. Она облегчит процесс конструирования. Вид заготовки отчета показан на рис. 3, который представляет собой композицию из фрагмента главной ленты и четырех окон:

- окно *Отчет1* обязано своему появлению значку Конструктор отчетов, расположенному на вкладке Создание ленты MS Access 2010;
- окно Обработка событий — щелчку правой кнопкой мыши в любом месте окна *Отчет1*;
- окно Список полей вызывается при помощи значка Добавить поля из раздела Сервис вкладки Конструктор;
- окно свойств появляется на экране после активации значка Страница свойств из того же раздела.

Включение в отчет даты, времени и номеров страниц

Любой документ, выдаваемый организацией, обязательно должен иметь в своем составе дату, а в некоторых случаях и время выдачи. Это единственный показатель актуальности сведений, которые он содержит. Большие отчеты должны иметь пронумерованные страницы. В MS Access 2010 есть несколько способов добиться желаемого результата. Для добавления номера страницы в область верхнего или нижнего колонтитулов выполните следующие действия. Откройте отчет в режиме конструктора.

1. На вкладке ленты Конструктор выберите значок Номера страниц. Он находится в разделе Колонтитулы. Откроется диалоговое окно Номера страниц (рис. 4).

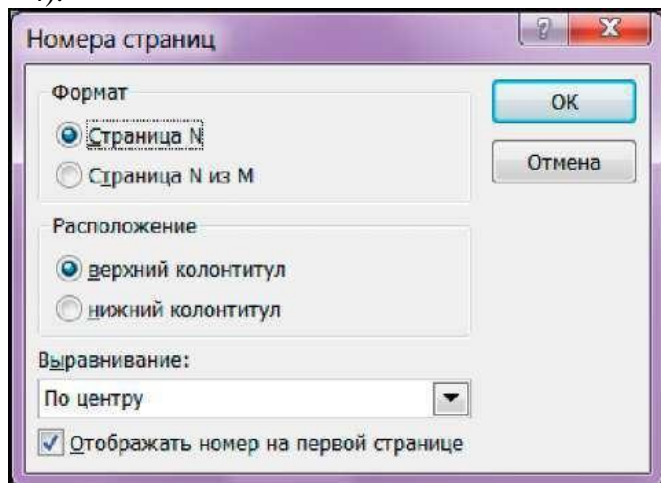


Рис. 4. Вставка номера страницы

2. Выберите формат, расположение и выравнивание для номеров страниц.
3. Снимите флажок Отображать номер на первой странице, если номер на первой странице не нужен.

4. Нажмите кнопку **ОК**. Номера страниц будут добавлены в отчет.

Для добавления номеров страниц в другую область отчета, а также для вставки даты и времени создания документа применяется более общий способ — окно Построитель выражений (рис. 5).

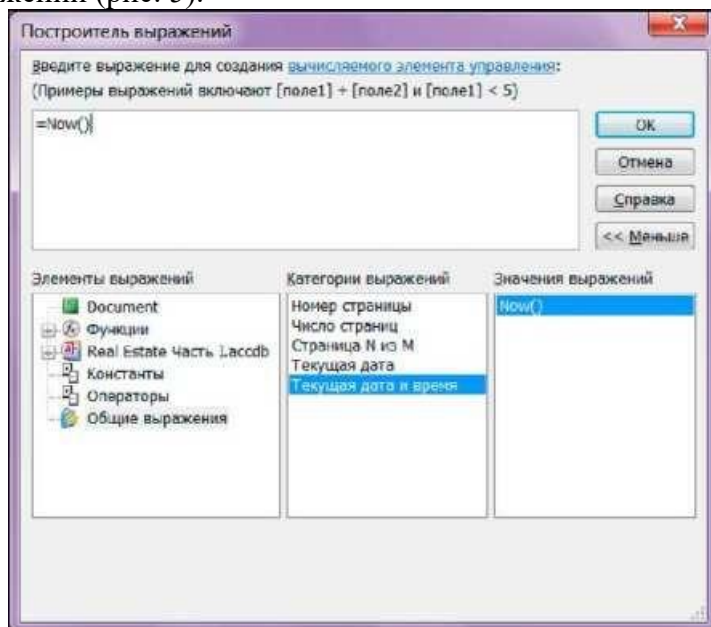


Рис. 5. Добавление в отчет даты и времени создания

Добавим в заголовок отчета надпись: «по состоянию на», после которой поместим дату создания отчета. Для этого:

1. Откройте отчет в режиме конструктора.
2. На вкладке Конструктор ленты MS Access 2010 выберите кнопку Поле. Она находится в разделе Элементы управления. Расположите курсор в области заголовка отчета. Курсор превратится в стилизованное изображение поля таблицы с крестиком в левом верхнем углу.
3. Сделайте щелчок левой кнопкой мыши и, удерживая ее, добейтесь требуемого размера поля. Кнопку отпустите.
4. Откройте окно свойств. Для этого сделайте щелчок по кнопке Страница свойств вкладки Конструктор.
5. В окне свойств выберите вторую вкладку Данные.
6. Перейдите к первой строчке Данные и нажмите кнопку. Появится окно построителя выражений (рис. 3).
7. В левом списке Элементы выражений выберите строчку Общие выражения, а в среднем Категории выражений — Текущая дата и время. В правом списке появится стандартная функция MS Access — Now(). Щелкните по кнопке **ОК**. Эта функция со знаком = появится в поле значений свойства Данные создаваемого элемента.
8. Вместо названия надписи Поле49: напишите: «по состоянию на». Для этого щелчком мыши выделите надпись и внесите изменения прямо на месте.
9. Перейдите на вкладку Формат. В разделе Шрифт установите шрифт *Arial Cyr* размером 10 пунктов или любой другой. Подчеркните текст и сделайте его выделенным. Здесь же можно назначить цвет текста и цвет фона.
10. Перейдите на вкладку Упорядочить. В разделе Размер и порядок щелкните по значку Размер или интервал. Откроется меню. Выберите в нем пункт по размеру данных. Размер элемента управления будет настроен в соответствии с назначенным шрифтом.

11. В этом же меню выберите пункт по узлам сетки. Он предназначен для изменения места расположения одного или нескольких элементов путем выравнивания по узлам сетки. Выполните окончательное форматирование.

Мастер почтовых наклеек

Мастер почтовых наклеек создает отчеты с оформлением, которое используется для печати наклеек стандартных форматов, таких как Avery, EXPE, Herma и Zweckform. С помощью этого мастера можно задать собственные параметры наклеек, если не подходит ни один из стандартных типов.

Работа с Мастером наклеек осуществляется в следующей последовательности.

1. На вкладке Создание ленты MS Access 2010 в разделе Отчеты выберите вкладку Наклейки.
2. Выберите таблицу или запрос для наклеек.
3. Щелкните дважды на пункте Почтовые наклейки, чтобы перейти к первому окну мастера.

Выбор формата и шрифта для наклейки

На первом шаге Мастера почтовых наклеек предлагается выбрать тип наклеек (рис. 6).

1. При необходимости измените систему единиц, установив в группе Система единиц переключатель Метрическая или Британская.

2. В раскрывающемся списке Фильтр по изготовителю выберите тип наклеек, которыми вы пользуетесь (*Avery, EXPE, Herma, Zweckform*). Теперь в списке в верхней части окна выберите по коду товара нужный размер наклейки. Либо, если вы хотите использовать наклейку, которую вы разработали сами и сохранили ранее, щелкните на кнопке Настройка, чтобы открыть диалоговое окно Размеры наклеек. Задайте свои параметры и вернитесь к первому окну мастера. В списке Фильтр по изготовителю появится тип наклейки с вашими параметрами.

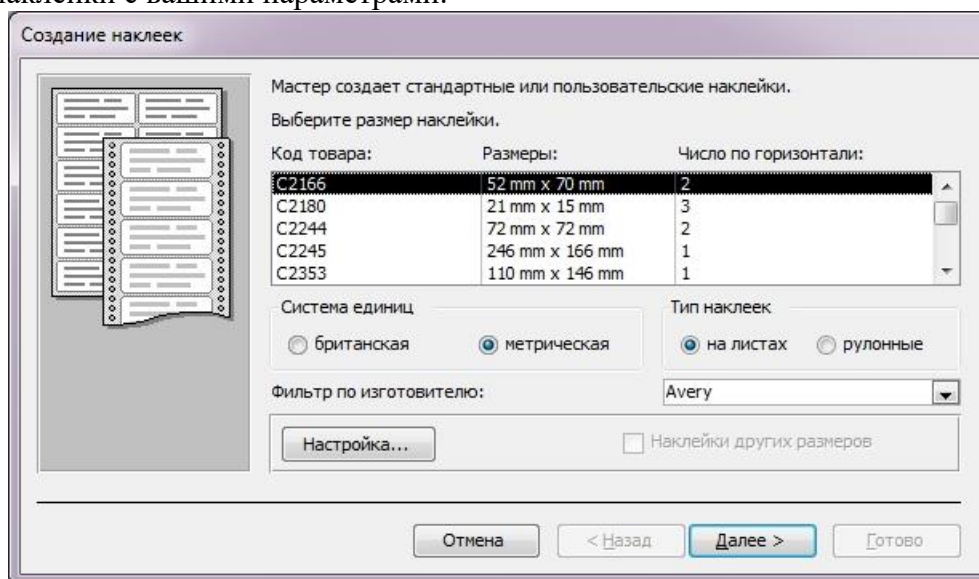


Рис. 6. Создание наклеек с помощью Мастера наклеек

3. Проверьте правильность установки параметра в группе Тип наклеек. Возможные варианты На листах или Рулонные.

4. Щелкните на кнопке Далее.

Создание собственного формата почтовых наклеек

Если необходимо задать собственные параметры наклеек, то выполните следующие действия.

1. На первом шаге Мастера наклеек щелкните на кнопке Настройка, а затем — на кнопке Создать, чтобы открыть окно, показанное на рисунке 7.

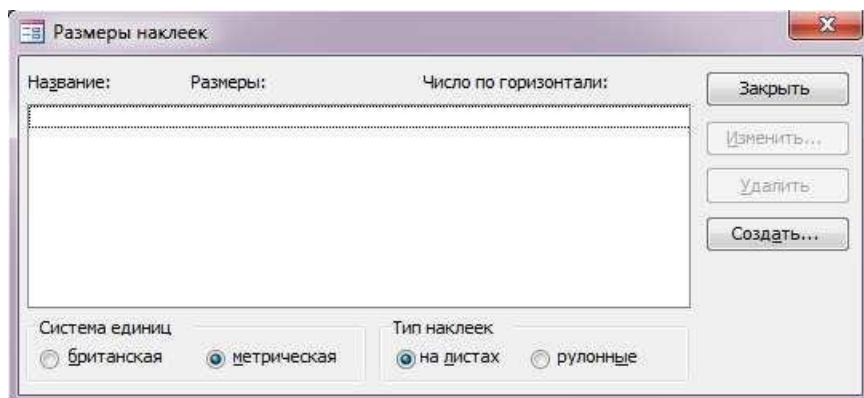


Рис. 7. Создание собственной наклейки

2. В поле Название введите название нового формата.
3. Выберите необходимые параметры в группах Система единиц, Тип наклеек, Ориентация.
4. В полях с нулями в нижней части окна введите размеры наклеек, полей, отступов от края страницы и расстояния между наклейками по вертикали и по горизонтали.
5. В поле Число по горизонтали введите количество столбцов наклеек на странице.
6. Щелкните на кнопке ОК, чтобы вернуться к окну Размеры наклеек. Затем щелкните на кнопке Заккрыть, чтобы вернуться к первому диалоговому окну Мастера наклеек.
7. Щелкните на кнопке Далее, чтобы перейти к следующему шагу мастера.

Выбор стиля текста наклейки

На втором шаге мастера вам предлагается определить шрифт, его размер, цвет и насыщенность, а также его тип (курсив, подчеркнутый или обычный). Действия на данном шаге носят чисто косметический характер, но задаваемые здесь параметры существенны и еще в одном отношении. Следует убедиться, что выбран оптимальный размер шрифта, подходящий для размера наклейки. В диалоговом окне мастера на этом шаге мы видим образец шрифта, отражающий выбранные характеристики.

На следующем шаге определяются поля и заносятся в прототип наклейки. В прототип можно вносить знаки пунктуации, пробелы и текст.

На последнем шаге мастера вы можете отсортировать наклейки по одному или нескольким полям.

После того как Мастер почтовых наклеек закончит свою работу, Access отобразит отчет в специальном окне предварительного просмотра.

Добавление вычисляемых выражений в отчеты

Для выполнения добавления вычисляемых выражений в формы и отчеты:

1. Откройте отчет в режиме Конструктора.
2. Выберите кнопку Поле (группа Элементы управления).
3. Выберите мышью пустое место в любой области отчета (например, область примечаний).
4. Для появившегося нового поля укажите необходимые свойства: откройте окно свойств поля (группа Сервис кнопка Страница свойств); для свойства Данные введите начиная со знака «=» нужное выражение, заключив имена полей БД в квадратные скобки. В качестве выражения может быть использована как встроенная функция (например, «=DATE()» - системная дата), так и любое действие над значениями полей с использованием арифметических или других операций; для свойства Формат поля выберите из списка тип вычисляемых данных. В случае необходимости вставьте рядом с полем элемент Надпись и заполните его нужным текстом.

5. Перейдите в режим Отчета.

Добавление в отчет вычисляемых и итоговых полей

Как уже говорилось, с помощью Конструктора отчетов можно создавать отчеты, содержащие вычисляемые и итоговые данные.

Для этого:

1. Откройте отчет в режиме конструктора.
2. Перетащите вниз нижнюю границу области примечания.
3. На панели Элементы управления в вкладке Конструктор щелкните на кнопке Поле, а затем щелкните в области примечания группы.
4. Щелкните правой кнопкой мыши на новом объекте и выберите в контекстном меню команду Свойства.
5. В списке свойств раскройте вкладку Данные и введите в строку свойства Данные нужное выражение. Например, для расчета суммарной стоимости работ каждого сотрудника достаточно ввести выражение =Sum([Стоимость работ]):

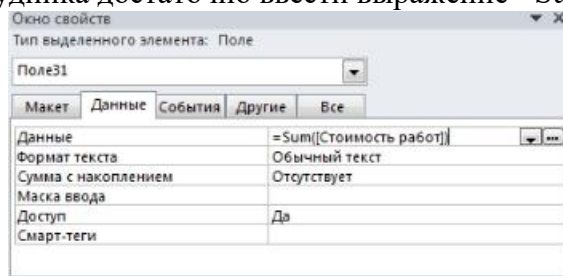


Рис.8. Диалоговое окно Свойства поля

6. Щелкните два раза на надписи нового поля и введите текст. Для создания итогового поля необходимо новое поле поместить в область примечания отчета, а все остальные шаги аналогичны приведенным выше.

Если ваш отчет уже содержит итоговые поля в примечании группы, скрыть область данных можно следующим образом.

1. Откройте отчет в режиме конструктора.
2. Щелкните правой кнопкой мыши на области данных и в контекстном меню выберите команду Свойства.
3. В списке свойств раскройте вкладку Макет.
4. Поменяйте значение свойства Вывод на экран на Нет.
5. После выполнения этих действий область данных отчета в окне предварительного просмотра будет скрыта.

7. Отчет по выборке

Если вы хотите создать отчет, в котором необходимо отобразить не все, а только выбранные на основании некоторого условия записи, то в вашем распоряжении несколько способов.

Если эти условия постоянны, создайте отчет на основе запроса, в котором определены необходимые условия отбора.

Если условия отбора изменяются, создайте запрос с параметрами, а на основании этого запроса — отчет.

Откройте таблицу, примените к ней фильтр и сохраните ее.

Затем создайте для данной таблицы отчет. Однако этот метод имеет свои тонкости: перед созданием отчета требуется активизировать фильтр.

Пример 1. СОЗДАНИЕ ОТЧЕТОВ.

Для создания отчета перейдите во вкладку **Отчеты** **Создание с помощью мастера**.

-Выберите таблицу **«Штатное расписание»**, поля *Должность*, *Оклад*, *Надбавка*.

-Из таблицы **«Врачи»** выберите поля: *ID*, *ФИО Врача*, *ВидОказываемыхУслуг*.

Нажмите кнопку <Далее>.

-В следующем диалоговом окне в строке *«Вид представления данных»* будет выбрана первая таблица, в нашем примере **«Штатное расписание»**.

-Нажмите кнопку <Далее>.

-Добавьте с помощью стрелок в уровень представления данных поле *Должность*.

-В следующих окнах выберите дизайн отчета.

-<Готово>.

Пример готового отчета представлен на рисунке.

Штатное Расписание					
Должность	Оклад	Надбавка	id	ФИО Врача	Оказываемые Услуги
врач-стоматолог	7 000,00р.	2 500,00р.	104	Жуков А.Е.	Терапевтическая
			102	Дюкина Я.Б.	Терапевтическая
			112	Харитонов Н.Н.	Терапевтическая
			109	Родионова Ф.Я.	Терапевтическая
			106	Леонидова М.Р.	Терапевтическая
			103	Елизарова Е.П.	Терапевтическая
			113	Шабарнева Ф.Н.	Терапевтическая
			110	Сидорович М.Л.	Терапевтическая
ортопед	14 000,00р.	3 000,00р.	105	Карбышев Г.О.	Ортопедическая
			100	Валиуллин Д.К.	Ортопедическая
			108	Никоненко П.С.	Ортопедическая

Рис. Отчет

РАБОТА С МАКРОСАМИ

При работе с базой данных часто приходится снова и снова выполнять одни и те же задачи. Чтобы не повторять одни и те же действия, можно использовать макросы. База данных неуклонно разрастается по мере того, как туда вносятся новые записи в формы, создаются запросы и распечатываются новые отчеты. При этом разрабатываются и сохраняются для последующего использования новые объекты. Приходится регулярно выполнять некоторые задачи. Для выполнения таких регулярных задач удобно использовать макросы Access. После создания этих небольших программ можно запускать их автоматически при определенных действиях пользователя (например, при щелчке на кнопке или при открытии определенной формы). Для автоматического запуска макросов в Access используются так называемые события. Макросы Access позволяют автоматически повторять различные действия и тем самым избавляют от написания сложных программ и подпрограмм.

Кроме того, с помощью макроса можно имитировать выбор команды меню или перемещение мыши. Макросы Access обычно не дублируют отдельных нажатий клавиш или перемещений мыши, выполняя особые, заданные пользователем задачи, например, открытие формы или запуск отчета. Использование макросов для выполнения рутинных операций существенно экономит время и силы. Кроме того, поскольку всякий раз при запуске макроса будет осуществляться одна и та же последовательность действий, макрос сделает работу с базой данных более эффективной и аккуратной.

Назначать макросы, зависящие от событий, можно с помощью окна свойств форм и отчетов. Что касается таблиц и запросов, они свойствами событий не обладают. Кроме того, при работе с формами можно запускать макросы с помощью имеющихся в Access панелей инструментов, командных кнопок и меню. Использование этих особенностей позволяет очень легко и эффективно использовать макросы, связанные с событиями. Макросы можно запускать и на уровне отдельных элементов управления, используя для этого соответствующие события. Например, можно проверить правильность ввода сложных данных непосредственно на уровне поля (а не после выхода этой записи), используя для этого свойство данного поля, а не свойства на уровне всей формы.

Любой макрос может состоять из одной или нескольких макрокоманд. После создания макросов можно объединить несколько взаимосвязанных макросов в один.

1. Создание макроса пользовательского интерфейса

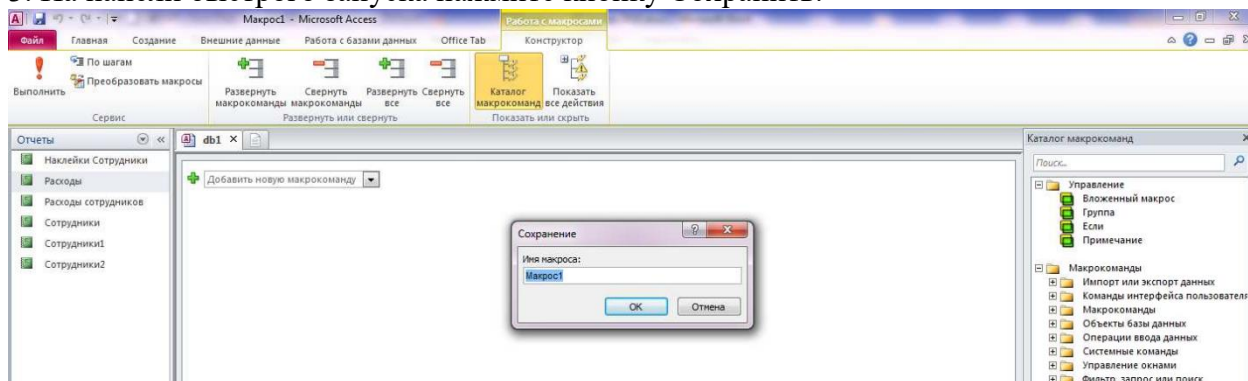
В Microsoft Access 2010 макросы, связанные с объектами пользовательского интерфейса (такими как кнопки, текстовые поля, формы и отчеты), называются макросами пользовательского интерфейса. Этим они отличаются от макросов данных, которые связываются с таблицами. С помощью макросов пользовательского интерфейса можно автоматизировать последовательности действий, таких как открытие другого объекта, применение фильтра, запуск операции экспорта и многие другие задачи. В этой статье описан измененный конструктор макросов и продемонстрировано выполнение основных задач, связанных с созданием макросов пользовательского интерфейса.

Макросы могут содержаться в объектах макроса (иногда их называют изолированными макросами) либо могут быть внедрены в свойства событий форм, отчетов или элементов управления. Внедренные макросы становятся частью объекта или элемента управления. Объекты макроса отображаются в области навигации в группе Макросы; внедренные макросы не отображаются. Каждый макрос состоит из одной или нескольких макрокоманд. В зависимости от текущего контекста некоторые макрокоманды могут быть недоступны. В частности, при работе с веб-базой данных недоступны некоторые макрокоманды, несовместимые с функцией публикации в службах Access.

2. Создание изолированного макроса

В описанной ниже процедуре создается объект изолированного макроса, который будет отображаться в разделе Макросы в области навигации. Изолированные макросы удобно использовать повторно в разных частях приложения. Вызывая тот или иной макрос из других макросов, можно избежать дублирования программного кода в нескольких местах (рисунок).

1. На вкладке Создание в группе Макросы и код нажмите кнопку Макрос.
2. В приложении Access откроется конструктор макросов.
3. На панели быстрого запуска нажмите кнопку Сохранить.



Создание изолированного макроса

4. В диалоговом окне Сохранить как введите имя макроса, а затем нажмите кнопку ОК.
5. Перейдите к разделу Добавление команд в макрос.

3. Создание внедренного макроса

В описанной ниже процедуре создается макрос, который внедряется в свойство события объекта. Такой макрос не отображается в области навигации, однако его можно вызывать из событий, таких как Загрузка или Нажатие кнопки. Поскольку макрос становится частью объекта формы или отчета, внедренные макросы рекомендуется создавать для автоматизации задач, которые специфичны для определенной формы или отчета.

1. В области навигации щелкните правой кнопкой мыши форму или отчет, которые будут содержать макрос, и выберите пункт Режим макета.
2. Если окно свойств не открыто, нажмите клавишу **F4**.
3. Выберите элемент управления или раздел, содержащий свойства события, в который нужно встроить макрос. В верхней части окна свойств из выпадающего списка в

разделе Тип выбора можно выбрать элемент управления или раздел, а также форму или отчет целиком.

4. В области задач Страница свойств откройте вкладку Событие.

Щелкните поле свойства для события, которое должно запускать макрос. Например, если требуется запускать макрос при нажатии кнопки, щелкните поле Нажатие кнопки.

Если поле свойства содержит слова [Внедренный макрос], это означает, что для данного события уже создан макрос. Если нужно отредактировать его, выполните остальные действия процедуры.

Если поле свойства содержит слова [Процедура обработки событий], это означает, что для данного события уже создана процедура Visual Basic для приложений. Прежде чем встраивать в это событие макрос, необходимо удалить процедуру. Чтобы сделать это, удалите слова [Процедура обработки событий], однако сначала следует просмотреть соответствующую процедуру и убедиться, что ее удаление не нарушит функциональность базы данных. Иногда функциональность процедуры VBA можно воссоздать с помощью внедренного макроса.

1. Нажмите кнопку Построение ...

2 В появившемся диалоговом окне Построитель установите флажок Конструктор макросов и нажмите кнопку ОК.

В Access запустится конструктор макросов. Перейдите к следующему разделу, в котором описано добавление команд в макрос.

4. Добавление команд в макрос

Команды представляют собой отдельные действия, составляющие макрос, и имя каждой из них соответствует выполняемому ей действию (например, НайтиЗапись или ЗакрытьБазуДанных). Выбор или поиск макрокоманды. Первым действием при добавлении команды является ее поиск в раскрывающемся списке Добавить новую макрокоманду или в каталоге макрокоманд.

- по умолчанию в раскрывающемся списке Добавить новую макрокоманду и каталоге макрокоманд выводятся только те команды, которые можно выполнить в недоверенных базах данных. Чтобы увидеть полный список команд, выполните указанные ниже действия.

- на вкладке Конструктор в группе Показать или скрыть выберите пункт Показать все действия.

- если каталог макрокоманд не отображается, на вкладке Макет в группе Показать/скрыть нажмите кнопку Каталог макрокоманд.

Чтобы найти макрокоманду, воспользуйтесь одним из указанных ниже способов.

- щелкните стрелку в раскрывающемся списке Добавить новую макрокоманду и прокрутите список до нужной команды. В верхней части списка выводятся элементы программного потока, а затем в алфавитном порядке — макрокоманды.

- перейдите к нужной команде в области каталога макрокоманд. Команды группируются по категориям. Чтобы просмотреть команды в той или иной категории, разверните ее. При выборе команды в нижней части каталога макрокоманд появляется ее краткое описание.

- найдите нужную команду в каталоге макрокоманд, введя запрос в поле поиска в верхней части соответствующей области. По мере ввода список действий фильтруется, при этом отображаются все макросы, содержащие введенный текст. В Access поиск текста выполняется как по именам макросов, так и по их описаниям.

Добавление макрокоманды в макрос

Найдя нужную макрокоманду, добавьте ее в макрос одним из указанных ниже способов.

- выберите команду в списке Добавить новую макрокоманду или просто начните вводить ее имя в поле. Access добавит команду в то место, где отображается список Добавить новую макрокоманду.

- перетащите команду из каталога макрокоманд на область макроса. При этом появится полоса вставки, указывающая, в какое именно место будет вставлена команда, как только будет отпущена кнопка мыши.

- дважды щелкните команду в каталоге макрокоманд.

Если в области макроса выделена команда, Access добавит новую макрокоманду под выделенной. Если в области макроса выделен блок Группа, Если, Иначе если, Иначе или Вложенный макрос, Access добавит новую макрокоманду в соответствующий блок.

Если в области макроса не выделена ни одна команда или блок, Access добавит новую макрокоманду в конец макроса.

Все ранее созданные макросы выводятся в узле В этой базе данных в каталоге макрокоманд.

При перетаскивании изолированного макроса (указанного в разделе Макросы) в область макроса создается команда ЗапускМакроса, которая запускает макрос, в который была перетащена команда. После этого из раскрывающегося списка можно запускать вложенные макросы (если они есть).

Чтобы просто скопировать команды из изолированного макроса в текущий (вместо создания команды ЗапускМакроса), щелкните его в каталоге макрокоманд и выберите команду Добавить копию макроса.

При перетаскивании внедренного макроса (указанного в списке объекта формы или отчета) в область макроса действия из него копируются в текущий макрос.

Можно создать макрокоманду, перетаскив объект базы данных из области навигации в область макроса. При перетаскивании таблицы, запроса, формы, отчета или модуля в область макроса приложение Access добавляет макрокоманду, открывающую таблицу, запрос, форму или отчет. При перетаскивании макроса добавляется макрокоманда, запускающая макрос.

Заполнение аргументов

Большинству макросов необходим как минимум один аргумент. Чтобы просмотреть описания аргументов, выберите макрокоманду и наведите указатель на нужный аргумент. Для многих аргументов значение можно выбрать в раскрывающемся списке. Если аргумент требует ввода выражения, функция *IntelliSense* поможет ввести его, предлагая по мере ввода допустимые значения.

Увидев нужное значение, добавьте его в выражение. Для этого дважды щелкните значение или выделите его с помощью клавиш со стрелками, а затем нажмите клавишу *TAB* или *ENTER*.

Перемещение макрокоманды

Макрокоманды выполняются в порядке сверху вниз по тексту макроса. Чтобы переместить команду в макросе вверх или вниз, воспользуйтесь одним из указанных ниже способов.

- Перетащите команду в нужное место вверх или вниз.

- Выделите команду и нажмите клавиши *CTRL* + СТРЕЛКА ВВЕРХ или *CTRL* + СТРЕЛКА ВНИЗ.

- Выделите команду, а затем щелкните стрелку Вверх или Вниз в правой части области макроса.

Удаление макрокоманды

Чтобы удалить макрокоманду, выполните указанные ниже действия.

- Выделите команду и нажмите клавишу *DELETE*. Кроме того, можно нажать кнопку Удалить (X) в правой части области макроса.

- При удалении блока макрокоманд (например, блока Если или Группа) удаляются все действия в этом блоке.

- Команды Вверх, Вниз и Удалить также доступны в контекстном меню макрокоманды.

5. Запуск макроса

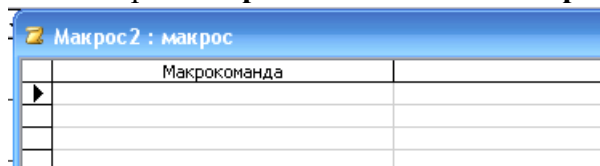
Запустить макрос можно любым из указанных ниже способов.

Дважды щелкнуть макрос в области навигации.

Вызвать макрос с помощью макрокоманды ЗапускМакроса или При- Ошибке.

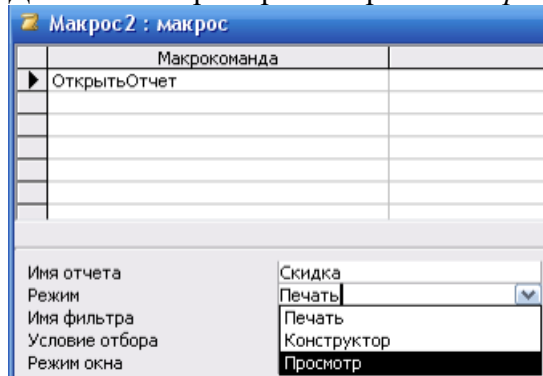
Указать имя макроса в свойстве Событие любого объекта. В результате макрос будет выполнен при возникновении этого события.

Пример 2. Для создания макроса, откройте заглавную форму в режиме конструктора и добавьте кнопку. Пусть с помощью вновь созданной кнопки можно открыть отчеты для просмотра по деятельности стоматологии. Щелкните правой кнопкой мыши по полю кнопки и выберите **Обработка событий Макрос**. Откроется окно конструктора макросов.



В этом окне надо выбрать макрокоманду из выпадающего списка, щелкнув на кнопке открытия списка макрокоманд. В списке насчитывается около различных макрокоманд для выполнения действий.

Для нашего примера выберите: *Открыть отчет*.



Затем выберите имя отчета и режим: *Просмотр*.

Сохраните макрос.

Все созданные макросы будут отображаться в вкладке **Макросы**.

Задания для лабораторной работы

Для базы данных из лабораторной работы №1:

1. Создайте отчеты по выполненным запросам
2. Создайте одну сводную таблицу в базе данных своего варианта.
3. Создайте отчет с рабочими кнопками для печати отчета, открытия закрытия.
4. Создайте все необходимые макросы: для открытия представления (запросов) и просмотра и печати отчетов и другие, необходимые для удобной работы функции.

ЛАБОРАТОРНАЯ РАБОТА №5. Системы управления базами данных MS ACCESS: Язык SQL

Цель лабораторной работы: Знакомство с языком управления данных *SQL* (*Structured Query Language*). Приобретение навыков и умений при работе базой данных с помощью языка SQL.

Методические указания

Язык *SQL (Structured Query Language)* – это специальный язык для управления данными. С его помощью можно полностью контролировать базу данных и выполнять все операции, посылая запросы (команды) в текстовом виде. Таким образом, например, можно управлять базой данных с удаленного компьютера.

Пример 1.

1. Создайте новую пустую базу данных *SQLbase.accdb*.

Создание и заполнение таблиц

2. Создайте запрос в режиме конструктора и перейдите в режим SQL.

Введите следующую команду для создания таблицы:

```
CREATE TABLE [Туры] (  
  
    [Код] INTEGER NOT NULL PRIMARY KEY,  
  
    [Страна] VARCHAR(50) NOT NULL,  
  
    [Транспорт] VARCHAR(20) NOT NULL,  
  
    [Цена] MONEY NOT NULL )
```

В этой команде требуется создать таблицу (**CREATE TABLE**) с именем «Туры». В таблице должно быть 4 поля:

Код – целое число (**INTEGER**), непустое (**NOT NULL**), первичный ключ таблицы (**PRIMARY KEY**)

Страна – строка длиной до 50 символов, непустое

Транспорт – строка длиной до 20 символов, непустое

Цена – поле для хранения денежной суммы (**MONEY**)

Названия таблиц и полей заключаются в квадратные скобки! Если эти названия состоят из одного слова, скобки можно не ставить:

```
CREATE TABLE Туры (  
  
    Код INTEGER NOT NULL PRIMARY KEY,  
  
    Страна VARCHAR(50) NOT NULL,  
  
    Транспорт VARCHAR(20) NOT NULL,  
  
    Цена MONEY NOT NULL)
```

3. Выполните эту команду (вкладка *Конструктор – Выполнить*). Проверьте, что таблица действительно создана.

4. Выполните команду для добавления в базу новой записи:

INSERT INTO Туры

VALUES (1, 'Финляндия', 'автобус', 1200)

Эта команда вставляет (**INSERT**) в таблицу «*Туры*» одну запись. После ключевого слова **VALUES** в скобках перечислены через запятую значения полей в том порядке, в котором они задавались при создании таблицы.

Символьные строки в значениях полей заключаются в апострофы или двойные кавычки!

5. Выполните ещё одно добавление записи:

INSERT INTO Туры

VALUES (1, 'Норвегия', 'самолёт', 15000)

Исправьте ошибку и добавьте новую запись правильно. Аналогично добавьте в таблицу ещё несколько записей:

Страна	Транспорт	Цена
Швеция	паром	9000 р.
Германия	автобус	15700 р.
Греция	самолёт	23000 р.
Норвегия	автобус	8000 р.
Германия	самолёт	19000 р.

Выбор и сортировка данных

6. Выполните запрос на выборку данных:

SELECT * FROM Туры

Посмотрите на результат. Этот оператор выберет все поля (*) всех записей из таблицы «*Туры*».

7. Вместо * можно указать через запятую список нужных полей:

SELECT Страна, Цена FROM Туры

Проверьте результат выполнения этого запроса.

8. Чаще всего нужно выбрать только записи, удовлетворяющие некоторому условию отбора. Для этого используется ключевое слово WHERE, после которого стоит условие:

SELECT * FROM Туры WHERE Страна = 'Норвегия'

Проверьте работу этого оператора.

9. Составьте запрос, который выбирает из таблицы «Туры» значения полей «Страна», «Транспорт» и «Цена» для всех автобусных туров:

10. Составьте запрос, который выбирает из таблицы «Туры» значения всех полей для туров с ценой меньше 10000 руб:

11. Для того, чтобы отсортировать данные по некоторому полю, в запросе после ключевых слов ORDER BY (англ. «упорядочить по») указывают название этого поля:

SELECT * FROM Туры ORDER BY Цена

Проверьте работу этого запроса. Если в конце предыдущего запроса добавить слово DESC (англ. «descending» – нисходящий), сортировка выполняется в обратном порядке.

12. Составьте запрос, который выбирает из таблицы «Туры» значения всех полей для туров с ценой больше 10000 руб. и сортирует результаты по убыванию цены.

13. В запросах можно использовать стандартные функции. Например, функция MIN определяет минимальное значение заданного поля среди всех записей:

SELECT MIN(Цена) FROM Туры

Результат этого запроса – одно число.

14.Составьте запрос, который находит минимальную цену для туров в Норвегию:

15.Результаты запросов можно использовать в других запросах – получается вложенный запрос. Например, запрос вернет данные о самом дешевом туре.

```
SELECT * FROM Туры WHERE Цена =  
(SELECT MIN(Цена) FROM Туры WHERE Страна = 'Норвегия')
```

16.Составьте запрос, который находит тур минимальной цены на самолёте.

Изменение и удаление данных

17.Для изменения записей используется оператор UPDATE. Запрос, приведенный ниже, увеличивает цены всех туров на 10%:

```
UPDATE Туры SET Цена = Цена*1.1
```

Проверьте, что данные в таблице «*Туры*» действительно изменились.

18.Авиакомпании в данный момент представляют скидку на билеты, так что цены всех туров на самолётах составляют 80% от исходных. Составьте и выполните соответствующий запрос. Какая стоимость получилась у тура в Грецию?

19.Скопируйте таблицу «*Туры*», назвав копию «*Туры2*». Удалите все туры в Германию с помощью запроса

```
DELETE FROM Туры2 WHERE Страна = 'Германия'
```


Проверьте, что данные в таблице «*Туры2*» действительно изменились. Удалите таблицу «*Туры2*», которая больше не нужна, с помощью запроса

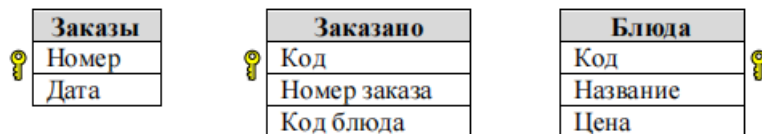
DROP TABLE Туры2

Проверьте, что таблица «Туры2» удалена из списка таблиц.

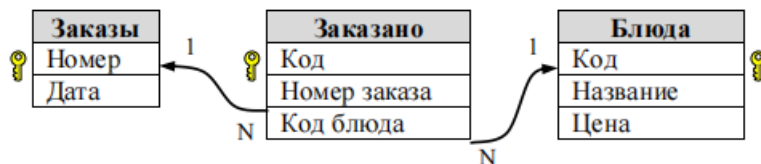
Задания для лабораторной работы:

Задание 1.

1. Используя только SQL-запросы, постройте три таблицы для реляционной базы данных кафе (значок  обозначает первичный ключ таблицы): -



2. Теперь нужно построить связи между таблицами:



Для этой цели используется команда **ALTER TABLE** (англ. изменить таблицу). Построить связь между таблицами – это значит задать ограничение (**CONSTRAINT**), которое связывает первичный ключ одной таблицы с полем соответствующего типа другой. Если связываемое поле второй таблицы – неключевое, то оно называется *внешним ключом* (**FOREIGN KEY**). Например, команда для создания связи 1:N между ключевым полем *Номер* таблицы *Заказы* и неключевым полем *Номер заказа* таблицы *Заказано* выглядит так:

```
ALTER TABLE Заказано
```

```
ADD CONSTRAINT ORDER_NO
```

```
FOREIGN KEY([Номер заказа])
```

```
REFERENCES Заказы(Номер)
```

Такая запись дословно означает:

Изменить таблицу Заказано

добавить связь ORDER_NO

внешний ключ [Номер заказа]

ссылается на поле Номер таблицы Заказы

3. Введите и выполните показанный выше SQL-запрос на добавление связи. Зайдите в меню *Сервис – Связи* и убедитесь, что связь действительно создана.

4. Составьте и выполните SQL-запрос на добавление второй связи.

5. С помощью SQL-запросов заполните базу следующими данными

Заказы	
Номер	Дата
1	11.04.13
2	12.04.13

Заказано		
Код	Номер заказа	Код блюда
1	1	1
2	1	3
3	1	4
4	2	1
5	2	2
6	2	2
7	2	5

Блюда		
Код	Название	Цена
1	борщ	80 р.
2	бифштекс	110 р.
3	гуляш	70 р.
4	чай	10 р.
5	кофе	50 р.

6. Построим в режиме SQL запрос *СоставЗаказа*, который выводит номер заказа и название заказанных блюд. Эти данные находятся в двух таблицах – *Заказано* и *Блюда*, поэтому их нужно как-то объединить. Для этого используется связь 1:N между таблицами, которую недавно установили. Действительно, для каждой записи в таблице *Заказано* нужно выбрать название блюда из таблицы *Блюда*, код которого совпадает с полем *Заказано.Код блюда*.

Это запрос на выборку данных, поэтому используем оператор **SELECT**:

SELECT Заказано.[Номер заказа], Блюда.Название

FROM Заказано, Блюда

WHERE Заказано.[Код блюда] = Блюда.Код

Здесь из таблиц *Заказано* и *Блюда* выбираются поля *Номер заказа* и *Название*; условие в последней строке связывает таблицы. Поскольку названия полей в таблицах, из которых идет выбор, не совпадают, можно было записать запрос в сокращенной форме, указав после оператора **SELECT** только названия нужных полей:

SELECT [Номер заказа], Название

FROM Заказано, Блюда

WHERE Заказано.[Код блюда] = Блюда.Код

7. Теперь добавим в запрос дату заказа. Она находится в таблице *Заказы*, которая пока в запросе не участвует. Таким образом, нам нужно объединить три таблицы. Условие отбора получается сложным, два условия (связи по коду блюда между таблицами *Заказано*

и *Блюда* и по номеру заказа между таблицами *Заказы* и *Заказано*) объединяются с помощью логической операции AND (И):

```
SELECT [Номер заказа], Дата, Название  
FROM Заказано, Блюда, Заказы  
WHERE Заказано.[Код блюда] = Блюда.Код  
AND Заказано.[Номер заказа] = Заказы.Номер
```

Проверьте результат выполнения этого запроса.

8. Измените запрос так, чтобы он выбирал только блюда из состава заказа №1.

9. Построим еще один запрос *Итоги*, в котором для каждого заказа выводится его номер, дата и общая сумма (с помощью функции **SUM**).

```
SELECT [Номер заказа], Дата, SUM(Цена)  
FROM Заказано, Блюда, Заказы  
WHERE Заказано.[Код блюда] = Блюда.Код  
AND Заказано.[Номер заказа] = Заказы.Номер  
GROUP BY [Номер заказа], Дата
```

В последней строке указано, что по полям *Номер заказа* и *Дата* выполняется группировка, то есть сумма цен считается для каждой уникальной пары «*Номер заказа – Дата*».

В таблице с результатами запроса заголовок столбца с суммой выглядит не совсем понятно для пользователя, например:

«Expr1002»

Для того, чтобы сделать у этого столбца заголовок «*Сумма*», нужно добавить в первую строку запроса после **SUM("Цена")** так называемый псевдоним (подпись) с ключевым словом **AS**:

```
SELECT [Номер заказа], Дата, SUM(Цена) AS Сумма
```

...

Проверьте результат выполнения этого запроса. Псевдонимы можно задавать для всех значений, которые выводятся в запросе.

10.Измените запрос так, чтобы заказы были отсортированы в порядке убывания суммы (используйте ключевые слова **ORDER BY**).

11.Построим запрос *МинСумма*, который выводит минимальную сумму заказа. Для этого будем использовать уже готовый запрос *Итоги*. Таким образом, источником данных для запроса *МинСумма* будет не таблица, а другой запрос. Отметим, что предварительно в запросе *Итоги* нужно отменить сортировку. Запрос получается очень простым

```
SELECT MIN(Сумма) AS Сумма FROM Итоги
```

12. Выведем информацию о заказе с минимальной суммой:

```
SELECT [Номер заказа], Дата, Итоги.Сумма
```

```
FROM Итоги, МинСумма
```

```
WHERE Итоги.Сумма = МинСумма.Сумма
```

Обратите внимание, что этот запрос использует результаты выполнения двух ранее построенных запросов – *Итоги* и *МинСумма*. Запрос *МинСумма* можно было и не составлять, а вместо этого использовать вложенный запрос (запрос в запросе):

```
SELECT [Номер заказа], Дата, Сумма
```

```
FROM Итоги
```

```
WHERE Сумма =
```

```
(SELECT MIN(Сумма) AS Сумма FROM Итоги)
```

Заметим, что если в базе данных есть информация о нескольких заказах с такой же (минимальной) суммой, будет показана информация обо всех этих заказах.

13.Измените запрос так, чтобы получить список всех заказов, сумма которых больше средней.

14.

Создание формы с подчинённой

Постройте форму, показанную на рисунке

Заказ

Дата

	Блюдо	Цена
▷	борщ	80 руб.
	гуляш	70 руб.
	чай	10 руб.

Запись 1 из 3

К оплате

Создание отчёта с группировкой

Постройте отчёт с группировкой, показанный на рисунке

Заказы									
Исполнитель: Василий Петров									
Дата: 15.12.12									
Дата 11.12.12									
Заказ 1									
	<table border="1"> <tr> <th>Блюдо</th><th>Цена</th></tr> <tr> <td>гуляш</td><td>40 руб.</td></tr> <tr> <td>борщ</td><td>30 руб.</td></tr> <tr> <td>рассольник</td><td>20 руб.</td></tr> </table>	Блюдо	Цена	гуляш	40 руб.	борщ	30 руб.	рассольник	20 руб.
Блюдо	Цена								
гуляш	40 руб.								
борщ	30 руб.								
рассольник	20 руб.								
Дата 12.12.12									
Заказ 2									
	<table border="1"> <tr> <th>Блюдо</th><th>Цена</th></tr> <tr> <td>компот</td><td>10 руб.</td></tr> </table>	Блюдо	Цена	компот	10 руб.				
Блюдо	Цена								
компот	10 руб.								

Задание 2.

1. Для выполнения задания 2 используйте ранее созданную базу данных MS Access. Например, базу данных созданную в лабораторной работе №1.
2. Дополнить существующую базу данных дополнительной информацией в режиме SQL.
3. Создать запрос на выборку в режиме SQL всех значений нескольких столбцов таблиц.
4. Создать запрос на выборку нескольких полей из нескольких таблиц в режиме SQL.
5. Создать запрос на выборку нескольких полей из таблицы при помощи параметра **WHERE** в режиме SQL.
6. Создать запрос на выборку данных из таблицы при помощи параметра **WHERE** и предиката **BETWEEN A AND B** в режиме SQL.
7. Создать запрос на выборку в режиме SQL, данных из таблицы при помощи параметра **WHERE** и предиката **LIKE**.
8. Сгруппировать данные таблицы при помощи инструкции **GROUP BY**.
9. Подсчитать количество строк таблицы.

10. Создать запрос, в режиме SQL, позволяющий получить информацию о первой записи обрабатываемой таблицы.
11. Создать запрос, в режиме SQL, позволяющий получить информацию о последней записи обрабатываемой таблицы.
12. Создать запрос в режиме SQL удаляющий информацию по заданному критерию.
13. Создать запрос в режиме SQL на обновление данных в любой из таблиц базы данных.
14. Создать запрос в режиме SQL на создание резервной копии одной из таблицы со структурой, аналогичной структуре выбранной таблицы.