

Лабораторная работа № 4

Основные понятия СУБД Access. Создание базы данных

Цель работы: ознакомиться с основными понятиями и объектами СУБД Access, освоить на практике технологию создания таблиц, обеспечения целостности данных.

Задание

1. Выполнить задания, помеченные в методических указаниях по выполнению работы символом **√**.

Методические указания по выполнению работы

В любой информационной системе производится обработка данных об объектах реального мира, при этом объём сохраняемых данных очень велик. Ядро ИС составляет её база данных, в которой сохраняются все данные.

База данных (БД) – упорядоченная совокупность данных, предназначенных для хранения, накопления и обработки с помощью ЭВМ.

Для создания и ведения баз данных (их обновления, обеспечения доступа по запросам и выдачи данных по ним пользователю) используются *системы управления базами данных (СУБД)*.

Система управления базами данных (СУБД) – это комплекс программных и языковых средств, необходимых для создания баз данных, поддержания их в рабочем состоянии и организации доступа пользователя к данным.

СУБД Access, являющаяся частью пакета Microsoft Office, поддерживает реляционную модель данных.

Файл базы данных Access имеет расширение .accdb и содержит таблицы, формы, запросы, отчёты и другие объекты.

Объекты базы данных Access

К объектам базы данных Access относятся:

Таблицы – предназначены для упорядоченного хранения данных.

Запросы – предназначены для поиска, извлечения данных и выполнения вычислений.

Формы – предназначены для удобного просмотра, изменения и добавления данных в таблицах.

Отчеты – используются для анализа и печати данных.

Страницы доступа к данным – предназначены для просмотра, ввода, обновления и анализа данных через сеть или из любого места компьютера.

Макросы – используются для выполнения часто встречающегося набора макрокоманд, осуществляющих обработку данных.

Модули – предназначены для описания инструкций и процедур на языке VBA.

Основным объектом базы данных является таблица. В таблице содержатся сведения о некотором классе объектов выбранной предметной области. Например, таблица может содержать сведения о студентах вуза, при этом каждая строка-запись содержит информацию о конкретном объекте (студенте). Каждый столбец-поле описывает один из атрибутов некоторого объекта, например, фамилию студента или дату рождения. Поэтому все данные одного столбца характеризуются одинаковым типом - множеством допустимых значений и операций над ними. В СУБД Access определены следующие типы данных:

Типы данных	Тип Описание
Текстовый	Используется для хранения символьных или числовых данных, не требующих вычислений. Максимальное количество символов, которые могут содержаться в текстовом поле, – 255
Поле МЕМО	Предназначено для ввода текстовой информации, по объему превышающей 255 символов; может содержать до 65 536 символов
Числовой	Используется для хранения числовых данных, участвующих в математических расчетах
Дата/Время	Используется для представления даты и времени
Денежный	Предназначен для хранения данных, точность представления которых колеблется от 1 до 4 знаков после запятой. Целая часть может содержать до 15 десятичных знаков
Счетчик	Предназначен для автоматической вставки уникальных последовательных (увеличивающихся на 1) чисел в качестве номера новой записи. Номер, присвоенный записи, не может быть удален или изменен. Поля с этим типом данных используются в качестве ключевых полей таблицы
Логический	Предназначен для хранения одного из двух значений, интерпретируемых как «Да / Нет», «Истина / Ложь»
Поле объекта OLE	Содержит данные, созданные в других программах, которые используют протокол OLE. Это могут быть, например, документы Word, электронные таблицы Excel, рисунки, звуковые и видеозаписи и др. Объекты OLE связываются с базой данных Access или внедряются в нее. Сортировать, группировать и индексировать поля объектов OLE нельзя
Гиперссылка	Специальный тип, предназначенный для хранения гиперссылок
Мастер подстановок	С его помощью создаётся поле со списком, из которого можно выбирать данные, содержащиеся в другой таблице или в наборе постоянных значений

Создание базы данных

Создадим базу данных некоторой организации, сотрудники которой работают с клиентами и выполняют их заказы.

Если все сведения поместить в одной таблице, то она станет очень неудобной для работы, так как в ней будут повторяться некоторые данные. Всякий раз, когда сотрудник Иванов будет работать с какой-либо фирмой, придется прописывать данные о сотруднике и этой фирме заново. Чтобы уменьшить избыточность

данных, можно исходную таблицу разбить на несколько таблиц и установить связи между ними.

Реляционная база данных – это совокупность связанных таблиц.

Таким образом, необходимо создать три таблицы: *Сотрудники*, *Клиенты* и *Заказы*

✓ Вызовите СУБД Access. В открывшемся после запуска окне выберите опцию **Новая база данных**, в поле **Имя файла** задайте имя файла, например *Фирма Ирина*, и нажмите кнопку **Создать**.

Активизируется вкладка **Работа с таблицами** и панель **Режим таблицы**. Существует несколько вариантов создания таблицы, но наиболее универсальным является создание таблиц на основе **Конструктора**.

✓ Перейдите в режим **Конструктора**, используя опцию **Режим** панели **Режим таблицы**. В появившемся диалоговом окне введём имя таблицы *Сотрудники* и нажмём ОК.

Эти действия приводят к открытию окна **Конструктора** (рис. 4.1), в котором выполняем описание структуры таблицы.

✓ В описании структуры таблицы *Сотрудники* зададим следующие поля:

Имя поля	Тип данных	Размер	Примечание
Код сотрудника	Счетчик		Ключевое поле
ФИО	Текстовый	17	
Должность	Текстовый	15	
Телефон	Текстовый	11	
Адрес	Текстовый	30	
Дата рождения	Дата/Время	Краткий формат	
Зарботная плата	Денежный		

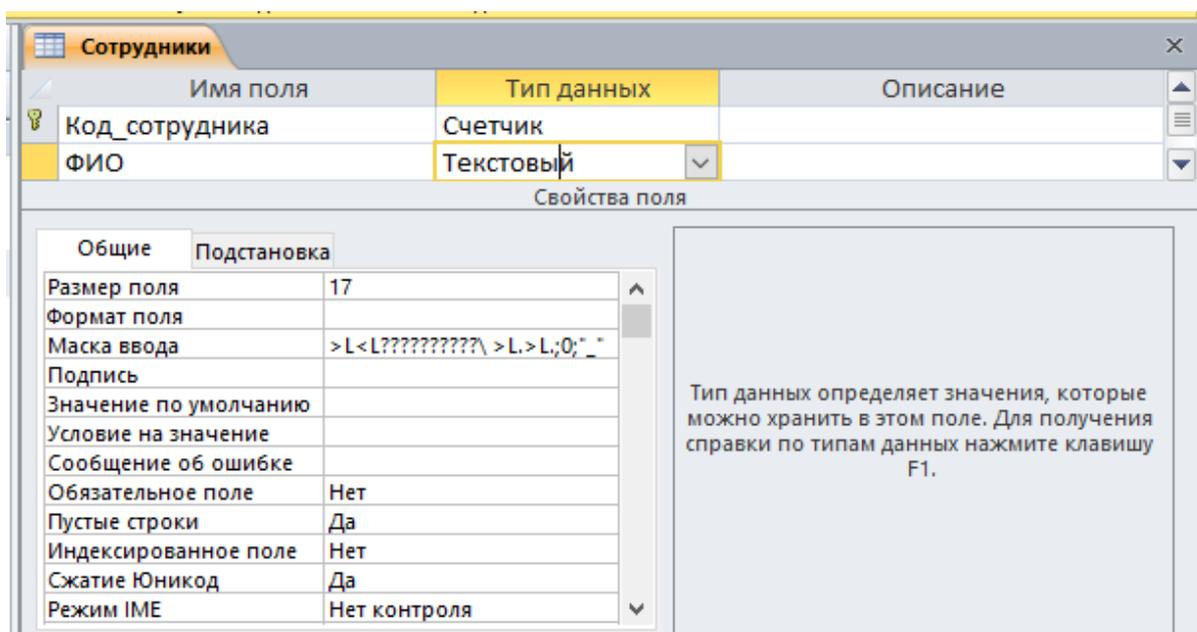


Рис. 4.1. Окно конструктора таблиц

Каждое поле должно иметь имя, длина которого не может превышать 64 символа. В именах полей можно использовать любые комбинации букв, цифр, пробелов и других символов, за исключением ., !, ', [и].

Для каждой реляционной таблицы существует понятие **ключа** - набора полей, которые однозначно идентифицируют каждую запись таблицы. В такой таблице не допускаются две или более записи с одинаковыми значениями ключевых полей. В общем случае ключ может состоять только из одного поля (*простой ключ*). Составной ключ предполагает идентификацию записи на основе комбинации значений нескольких полей.

В столбце **Имя поля** указывается произвольное имя поля. Переход в столбец **Тип данных** приводит к появлению кнопки раскрытия, щелчок по которой отображает список допустимых типов полей. Выбор типа также может производиться щелчком, а затем в строке **Размер поля** производится его конкретизация. Столбец **Описание** является вспомогательным (заполнять не обязательно), в нем могут храниться комментарии к создаваемым полям.

Для задания ключа необходимо выделить требуемое поле и щелкнуть по пиктограмме **Ключевое поле**  в панели инструментов. Если в качестве ключа определяются несколько полей, то они выделяются путем протягивания мыши с нажатой левой клавишей по ячейкам первого столбца у требуемых полей или щелчком у требуемых полей при нажатой клавише **Ctrl**, а затем щелкается пиктограмма **Ключевое поле**.

Контроль достоверности вводимых данных

СУБД автоматически осуществляет проверку достоверности данных, вводимых в поля определенного типа (например, нельзя ввести букву в числовое поле). Уровень контроля может быть повышен пользователем путем задания дополнительных ограничений на значения полей в области конструктора **Свойства поля** из бланка описания структуры таблицы.

Для этого можно использовать следующие свойства:

- **Значения по умолчанию.** Если в каком-либо поле формируемой таблицы будет преобладать одно значение, то целесообразно установить его в качестве значения **по умолчанию**, и оно будет появляться в данном поле автоматически при каждом создании новой записи. Например, если в таблицу заносится город проживания студента и преобладает Калининград, то целесообразно именно его установить в качестве значения по умолчанию. При занесении другого города данное значение необходимо стереть, например, сочетанием клавиш **Ctrl/Backspace**.
- **Обязательное поле.** Установка данного свойства в значение **Да** приводит к тому, что переход к новой записи не будет разрешен, пока пользователь не заполнит обязательное поле.
- **Условие на значение.** После перехода к данному свойству необходимо раскрыть путем нажатия на кнопку  окошко, в котором формируется

само условие с помощью указанных в нём операций. Например, можно установить условие на экзаменационные оценки в виде ограничения >1 And <6 , что позволит вводить только цифры 2, 3, 4 и 5.

- **Маска ввода.** Это свойство используется для установки ввода данных по определенному шаблону. Шаблон состоит из трех частей, разделенных точками с запятой. Первая часть является маской, содержащей служебные символы и символы-заполнители данных, которые нужно ввести в поле. Служебными символами являются:

0 - знакоместо для цифры (от 0 до 9), ввод ее обязателен;

9 - знакоместо для цифры или пробела, ввод не обязателен;

- знакоместо для цифры или пробела, ввод не обязателен, пустые символы преобразуются в пробелы. Разрешены символы + и -;

L - буква, ввод обязателен;

? - буква, ввод не обязателен;

A - буква или цифра, ввод обязателен;

a - буква или цифра, ввод не обязателен;

& - любой символ или пробел, ввод обязателен;

c - любой символ или пробел, ввод не обязателен;

< - принудительный перевод всех последующих символов в нижний регистр;

> - принудительный перевод всех последующих символов в верхний регистр;

! - заполнение маски справа налево (символ ! можно ставить в любую позицию маски);

\ - следующий символ является текстовой константой (отображается как литерал).

Любые символы, не включенные в этот список, отображаются как литералы.

Вторая и третья части шаблона не являются обязательными. Если во второй части указана 1 или ничего не указано, то при заполнении поля сохраняются только введенные символы, тогда как 0 означает, что нужно сохранять как введенные символы, так и символы маски. В третьей части маски указывается символ, который отображается как заполнитель исходно пустых позиций при вводе нового значения. Если этот символ не используется, то по умолчанию используется подчеркивание.

Примеры масок:

00-00-00

>L<??????????

Результат ввода:

27-34-56

17-78-32

Петров

Радиотехника

Достаточно сложно реализовать шаблон для ввода ФИО в виде Фамилия И.О. Если предположить, что сама фамилия содержит не более 12 символов и не короче 2 символов, то маска имеет вид:

>L<L??????????\ >L.>L.;0;"_" и при переходе в данное поле изображается следующей структурой: _____ _._. При ее заполнении

необходимо установить курсор в левую позицию, набрать фамилию в любом регистре (первый символ автоматически станет заглавной буквой, остальные отображаются в прописном варианте), а затем мышью или клавишей → перенести курсор в место ввода инициалов:

Романов 

При переходе в следующее поле введенное значение преобразуется к нормальному виду, т.е. без пропусков, например, Романов П.Р.

Если в шаблоне будет отсутствовать вторая компонента, то точки после инициалов придется вводить самому пользователю, в противном случае ФИО будет отображаться в формате Фамилия ИО.

✓ Задайте маску ввода для поля ФИО таким образом, чтобы первая буква фамилии и инициалы автоматически становились заглавными буквами, точки после инициалов вводились автоматически, количество букв в фамилии – не более 12.

✓ Закроем таблицу *Сотрудники* в *Конструкторе*, сохранив структуру таблицы. При этом появляется окно (рис. 4.2).

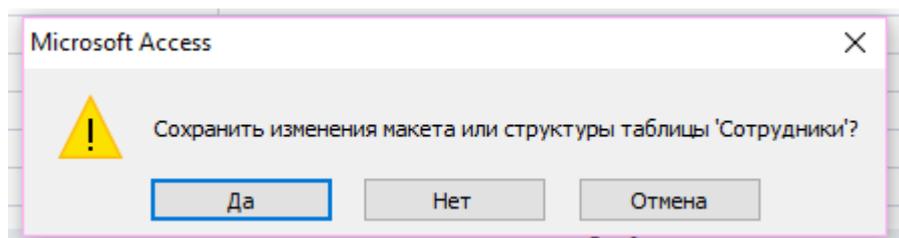


Рис. 4.2. Завершение работы в Конструкторе таблиц

✓ Создадим ещё две таблицы *Клиенты* и *Заказы*, их структура представлена в таблицах.

Клиенты

Имя поля	Тип данных	Размер	Примечание
Код клиента	Счетчик		Ключевое поле
Название компании	Текстовый	20	
Адрес клиента	Текстовый	30	
Телефон клиента	Текстовый	11	

Заказы

Имя поля	Тип данных	Размер	Примечание
Код заказа	Счетчик		Ключевое поле
Код клиента	Числовой		
Код сотрудника	Числовой		

Дата заказа	Дата/ Время	Краткий формат	
Дата исполнения	Дата/ Время	Краткий формат	
Сумма	Денежный		
Отметка о выполнении	Логически й		

✓ Для создания следующей таблицы (*Клиенты*) на вкладке *Создание* вызываем *Конструктор таблиц* и задаём её структуру. При закрытии окна конструктора таблиц в диалоговом окне надо ввести имя создаваемой таблицы.

✓ Создадим таблицу *Заказы*.

Таблица *Заказы* содержит поля *Код сотрудника* и *Код клиента*. При их заполнении могут возникнуть некоторые трудности, так как не всегда удастся запомнить все предприятия, с которыми работает фирма, и всех сотрудников с номером кода.

✓ Для удобства ввода данных создадим раскрывающиеся списки с помощью ***Мастера подстановок***.

- Для поля *Код сотрудника* выберите тип данных *Мастер подстановок*.
- В появившемся окне выберите команду *Объект "столбец подстановки" будет использовать значения из таблицы или запроса* и щелкните на кнопке *Далее*.
- В списке таблиц выберите таблицу *Сотрудники* и щелкните на кнопке *Далее*.
- В списке *Доступные поля* выберите поле *Код сотрудника* и щелкните на кнопке со стрелкой, чтобы ввести поле в список *Выбранные поля*. Таким же образом добавьте поле *ФИО* и щелкните на кнопке *Далее*.
- Выберите порядок сортировки списка по полю *ФИО*.
- В следующем диалоговом окне задайте необходимую ширину столбцов раскрывающегося списка.
- Установите флажок *Скрыть ключевой столбец»* и нажмите кнопку *Далее*.
- На последнем шаге *Мастера подстановок* замените *Надпись для поля подстановок* (зададим *Сотрудник*). Далее необходимо активизировать опцию *Включить проверку целостности данных* и щёлкнуть на кнопке *Готово*.

✓ Аналогичным образом создайте раскрывающийся список для поля *Код клиента*.

После создания ключевых полей можно приступить к созданию связей. Существует несколько типов связей (отношений) между таблицами:

- при отношении «*один-к-одному*» каждой записи ключевого поля в первой таблице соответствует только одна запись в связанном поле другой таблицы, и наоборот. Отношения такого типа используются не очень часто. Иногда их можно использовать для разделения таблиц, содержащих много полей, для отделения части таблицы по соображениям безопасности;
- при отношении «*один-ко-многим*» каждой записи в первой таблице соответствует несколько записей во второй, но запись во второй таблице не может иметь более одной связанной записи в первой таблице;
- при отношении «*многие-ко-многим*» одной записи в первой таблице могут соответствовать несколько записей во второй таблице, а одной записи во второй таблице могут соответствовать несколько записей в первой.

✓ Закройте все открытые таблицы, так как создавать или изменять связи между открытыми таблицами нельзя. Вызовите команду ***Работа с базами данных -Схема данных***.

Если ранее никаких связей между таблицами базы данных не было, то при открытии окна ***Схема данных*** одновременно открывается окно ***Добавление таблицы***, в котором выбирают нужные таблицы.

Чтобы установить связь между таблицами, надо выделить поле в одной таблице и перетащить его на соответствующее поле в другой таблице.

После перетаскивания откроется диалоговое окно *Изменение связей*, в котором надо включить флажок *Обеспечение целостности данных*. Это позволит предотвратить случаи удаления записей из одной таблицы, при которых связанные с ними данные других таблиц останутся без связи.

Поскольку мы предусмотрели поля подстановок для таблицы *Заказы*, то Access автоматически создал связи между таблицами. В нашем случае схема данных выглядит как на рис. 4.3.

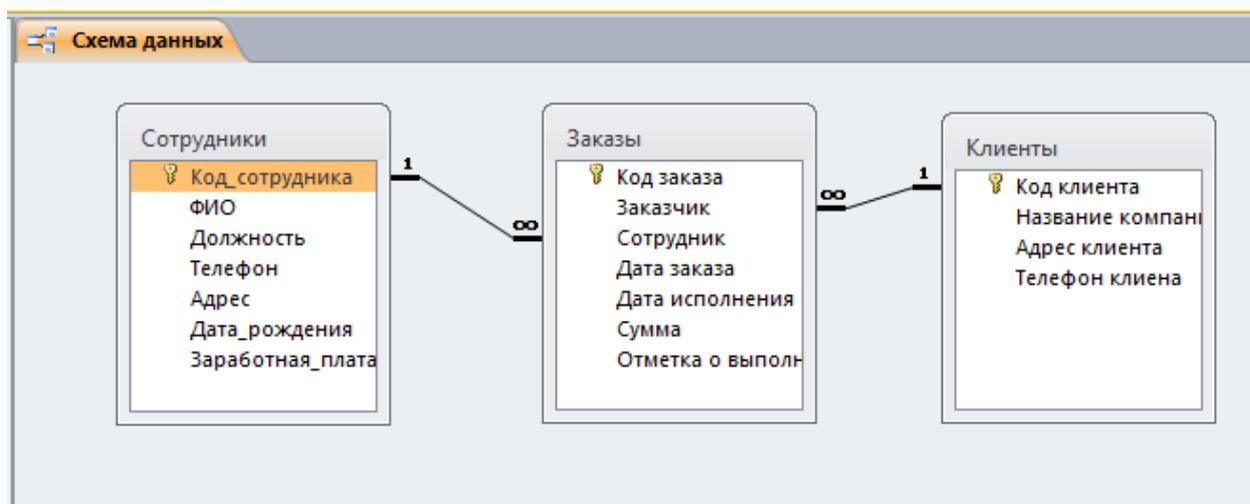


Рис. 4.3. Схема данных

Для изменения связи между таблицами её надо выделить, вызвать контекстное меню щелчком правой клавиши мыши, выбрать команду **Изменить связь**.

✓ Измените связи между таблицами: включите опцию Каскадное обновление связанных полей.

После того, как заданы описания структур таблиц, установлены связи между таблицами, можно приступить к их заполнению. Для этого надо выполнить двойной щелчок на имени таблицы в левой части окна и ввести значения в поля таблицы.

✓ Введите по пять записей в таблицы *Сотрудники* и *Клиенты* и десять записей в таблицу *Заказы*. При вводе данных в таблицу *Заказы* используйте списки, созданные для подстановки.

Навигация по полям таблицы осуществляется клавишами →, ←, ↑, ↓, Tab, Shift/Tab, мышью. Переход к заполнению следующей записи автоматически сохраняет в памяти компьютера предыдущую запись.

В процессе работы с таблицей можно в любой момент вернуться в режим конструктора для изменения её структуры путем нажатия мышью пиктограммы



, которая отображается в панели инструментов в табличном режиме.

Изменению могут быть подвергнуты названия полей, их тип и последовательность. Для удаления некоторого поля таблицы необходимо его выделить щелчком правой клавиши мыши в первом столбе и нажать мышью

пиктограмму  (*Удалить строки*) в панели инструментов.

Пустая строка в описании структуры создается следующим образом:

- курсор устанавливается на строку, перед которой необходимо вставить пустую;
- на панели инструментов нажимается кнопка  (*Вставить строки*).

Перенос строки описания структуры реализуется следующим образом:

- выделяется мышью в первом столбце переносимая строка (строки);
- выделенная строка в первом столбце захватывается мышью и перетаскивается на новую позицию.

✓ Сохраните базу данных на своём носителе данных, поскольку она будет использоваться в последующих работах.

Контрольные вопросы

1. Что называется базой данных (БД)?
2. Что такое система управления базами данных (СУБД)?
3. Чем отличается Microsoft Excel от Microsoft Access?
4. Какие объекты базы данных Microsoft Access вы знаете?
5. Какой объект в базе данных является основным?
6. Что называется полями и записями в БД?
7. Какие типы данных вы знаете?
8. Как можно создать поле с раскрывающимся списком?
9. С каким расширением сохраняется файл БД Access?
10. С помощью каких средств можно создавать таблицы в Access?
11. Что такое ключевое поле?
12. Как установить несколько ключевых полей?
13. Как установить связи между таблицами?
14. Какие типы связей между таблицами вы знаете?

Лабораторная работа № 5

Отбор данных с помощью запросов. Создание отчётов

Цель работы: освоить на практике правила формирования запросов к базе данных, технологии создания запросов и отчётов.

Задание

1. Выполнить задания, помеченные в методических указаниях по выполнению работы символом ✓

Методические указания по выполнению работы

Вопросы, которые формируются средствами СУБД к одной или нескольким таблицам, называются *Запросами*.

Запросы являются основным средством просмотра, отбора, изменения и анализа информации, которая содержится в одной или нескольких таблицах базы данных.

Существует несколько типов запросов:

Запрос на выборку – позволяет отобразить информацию по заданным критериям отбора.

Запрос-создание новой таблицы. Предназначен для сохранения результатов запроса в виде новой таблицы.

Запрос-добавление записей в другую таблицу. Выборку можно добавить к другой таблице, однотипной по структуре или с изменением структуры выборки.

Запрос-удаление. С помощью запросов можно удалить часть или все записи из таблицы.

Запрос-обновление. С помощью запросов можно обновлять значения полей таблицы.

Для создания запросов можно использовать *Мастер запросов* или *Конструктор*.

Самым распространённым типом запросов является *Запрос на выборку*.

✓ С помощью мастера запросов создайте запрос на выборку данных из таблицы *Сотрудники* с именем *Телефоны сотрудников*.

- Откройте свою базу данных, созданную в предыдущей работе. Вызовите команду: *Создание – Запросы – Мастер запросов – Простой запрос*.
- В появившемся окне (рис. 5.1) укажите таблицу *Сотрудники* и выберите требуемые поля. Щёлкните на кнопке *Далее*.

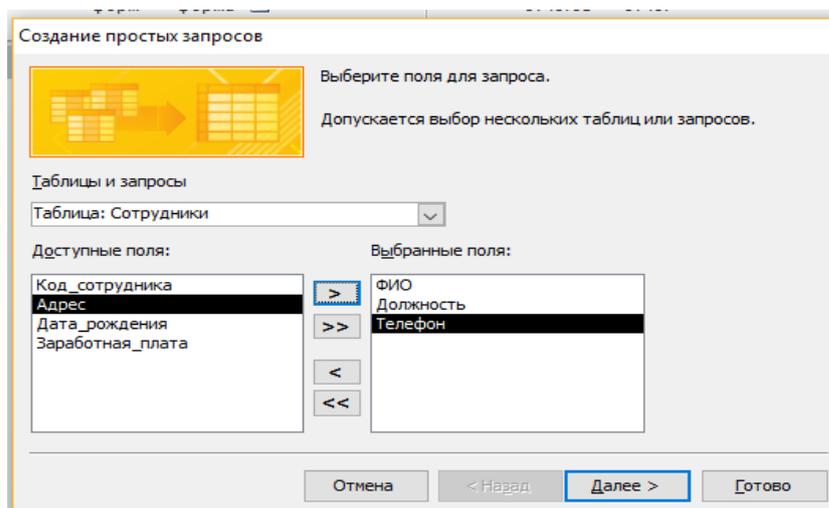


Рис. 5.1. Создание простого запроса

- В следующем окне задайте имя запроса *Телефоны сотрудников* и щёлкните на кнопке *Готово*.
- Появятся результаты выборки по запросу (рис. 5.2).

ФИО	Должность	Телефон
Иванов И.П.	менеджер	89118545478
Сазонов А.Л.	менеджер	89118556780
Петров А.П.	бухгалтер	89062322212
Орлова И.А.	секретарь	89062344532
Воробьев А.А.	директор	89118565644
*		

Рис. 5.2. Результаты выборки по запросу *Телефоны сотрудников*

✓ С помощью **Конструктора** создать запрос с именем *Адреса клиентов*.

- Для этого вызовите команду: *Создание – Запросы – Конструктор запросов* (рис. 5.3)
- В диалоговом окне *Добавление таблиц* выберите таблицу *Клиенты* и щелкните на кнопке *Добавить*, а затем – на кнопке *Закреть*.
- Чтобы перенести нужные поля в бланк запроса, необходимо по ним дважды щелкнуть левой кнопкой мыши (рис. 5.3). Выбираем поля *Название компании*, *Адрес клиента*, *Телефон клиента*.
- Чтобы отсортировать записи в поле *Название компании* в алфавитном порядке, необходимо в раскрывающемся списке строки *Сортировка* выбрать пункт *по возрастанию*.

- Закройте окно конструктора запросов. При попытке закрыть окно конструктора запросов появится окно, в котором надо ввести имя запроса - *Адреса клиентов*.

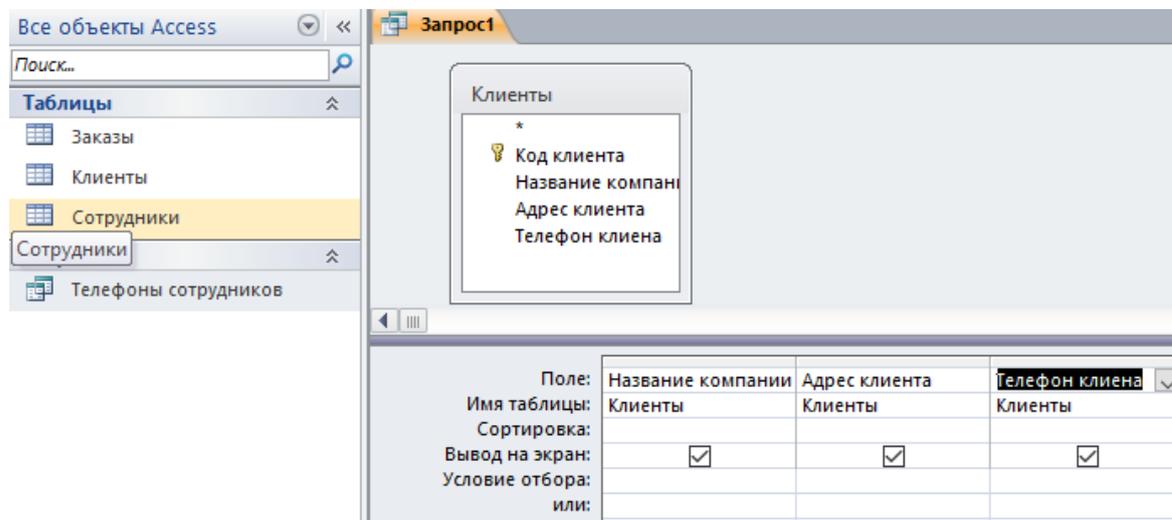


Рис. 5.3. Создание запроса в режиме Конструктора

✓ Самостоятельно создайте запрос *«Дни рождения»*, в котором можно будет просмотреть дни рождения сотрудников.

Самое главное в запросе - критерии выборки, которые вводятся в строку *Условие отбора*. Можно выделить следующие типы запросов на основе критериев:

Выборка по строгому совпадению. В строку *Условие отбора* для определенного поля вводится одно из значений, существующих в таблице. Например, название конкретного товара, должности или название фирмы.

✓ Создайте запрос *Заказы менеджеров*. Для этого вызовите *Мастер запросов-Простой запрос*, из таблицы *Заказы* выберите поля: *Заказчик*, *Сотрудник*, *Дата заказа*, *Дата исполнения*, *Сумма*, *Отметка о выполнении*, из таблицы *Сотрудники* выберите поле *Должность*, на следующих шагах выберите *Подробный* и дайте имя запросу - *Заказы менеджеров* и щёлкните кнопку *Готово*. Появятся результаты выполнения запроса. Далее надо перейти в окно конструктора запросов. Для этого выберите режим *Конструктора* на вкладке *Главная* или в *контекстном меню*, которое появляется, если щёлкнуть правой клавишей на ярлычке запроса (сверху). В строке *Условие отбора* для поля *Должность* наберите *менеджер* (рис. 5.4).

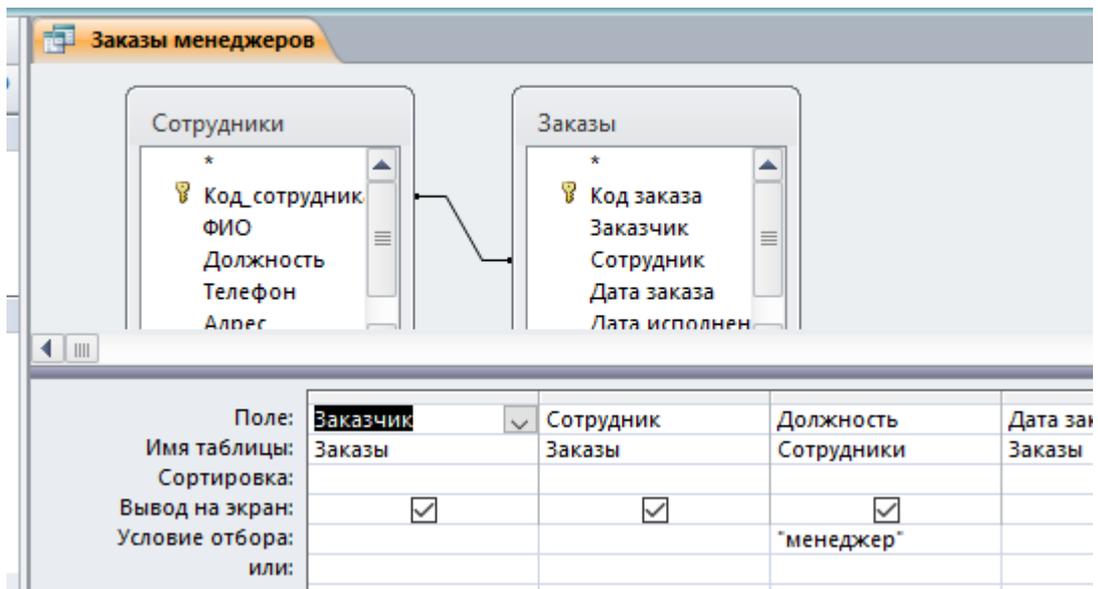


Рис. 5.4. Запрос на выборку по строгому совпадению

✓ Самостоятельно создайте запрос *Выполненные заказы*, содержащий следующие сведения: *ФИО сотрудника, Заказчик*, с которым он работает, *отметка о выполнении и сумма*. Данные запроса возьмите из нескольких таблиц. Для условия отбора по полю *Отметка о выполнении* введите *Да*.

Параметризованные запросы. Это частный случай запросов по строгому совпадению. Для параметризации необходимо в строке *Условие отбора* вместо самого условия ввести *текст приглашения на его ввод* в квадратных скобках, например [Введите условие отбора].

При запуске параметризованного запроса появляется диалоговое окно (рис. 5.5), в котором пользователь должен ввести собственно условие отбора.

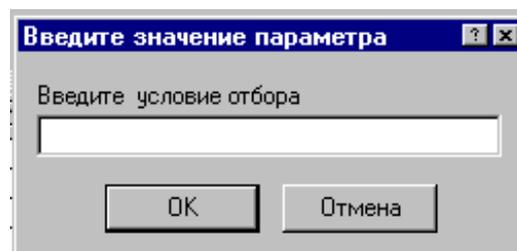


Рис. 5.5. Окно выполнения параметризованного запроса

Выборка по строгому несовпадению. В этом случае в выборку отбираются все записи таблицы, кроме записей, содержащих значение, указанное в строке *Условие отбора*. Для реализации данного запроса перед значением вводится префикс *Not*.

Выборка по неточному совпадению. Для выборки записей в условиях неполноты знаний о требуемых значениях используется оператор *Like*

<условие>. Само <условие> образуется следующими подстановочными символами:

? - любой один символ;

* - любое количество символов (0 - ∞);

- любая одна цифра;

[список_символов] - любой символ из списка;

[!список_символов] - любой символ, не входящий в список;

В списке можно указывать сразу диапазон символов, например, [Г-Л] или [Г-ЛГ-Л].

✓ Измените запрос *Телефоны сотрудников*, предусмотрев для пользователя возможность ввода интересующей его фамилии без инициалов или только первых букв фамилии. Очевидно, что надо создать запрос с параметрами для поля ФИО, по неточному совпадению. Откройте запрос *Телефоны сотрудников* в *Конструкторе*. В строке *Условие отбора* для поля *ФИО* наберите Like [Введите фамилию] (рис. 5.6).

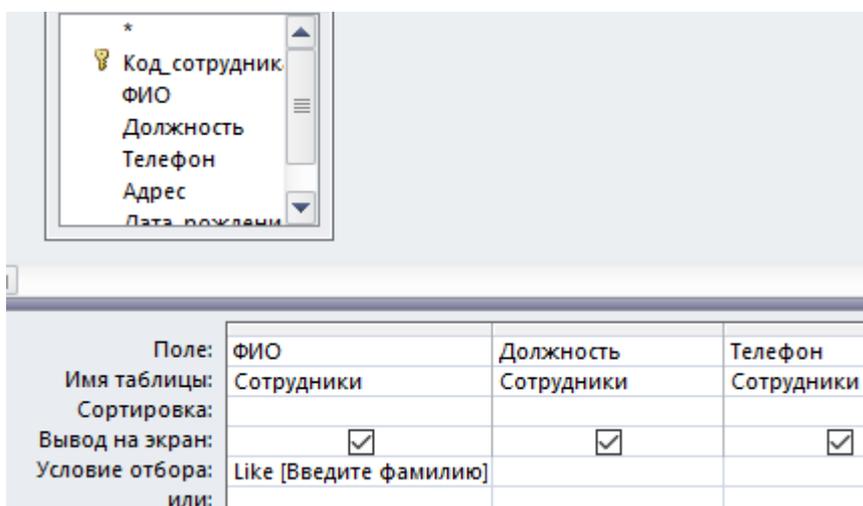


Рис. 5.6. Создание запроса

✓ Для выполнения запроса вызовите команду *Выполнить*, находящуюся на линейке команд *Конструктор*. При выполнении команды появится окно (рис. 5.7). В нем введем: *Иванов**. В результате выполнения запроса будут выведены телефоны всех сотрудников организации с фамилией Иванов.

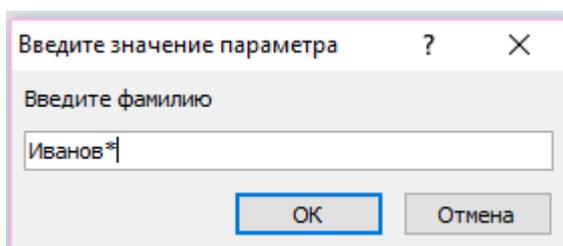


Рис. 5.7. Выполнение запроса

Выборка по диапазону. Для формирования данных условий выбора используются операторы сравнения >, >=, <, <= и <>. Операции сравнения могут связываться логическими операциями And (И) и Or (ИЛИ).

Для этих же целей используется оператор диапазона Between <нижнее_значение> and <верхнее_значение>. Например, выбор книг стоимостью от 100 до 200 руб. может быть реализован через ввод в запросе условия в поле Стоимость в виде >=100 and <=200 или Between 100 and 200.

Перечень значений в условии выборки можно задать и оператором In (значение, значение, ...).

Для выбора записей с пустыми значениями в некотором поле надо в соответствующем поле бланка запроса указать оператор Is Null. Наоборот, записи с непустыми значениями в данном поле выбираются по оператору Is not Null.

Запрос с вычислениями. Такой запрос позволяет получить дополнительную информацию в процессе выборки, например, стоимость всей партии товара при хранимой в таблице информации о количестве товара и стоимости единицы его продукции. Для этого в строку *Поле* пустого столбца заносят выражение для вычисления по следующему формату:

Название_формируемого_поля :выражение.

В *выражении* можно использовать знаки арифметических операций, круглые скобки и имена полей в квадратных скобках. Например, стоимость партии можно вычислить по выражению:

Стоимость партии:[количество товара][стоимость единицы товара]*

ВНИМАНИЕ: имена полей в выражении должны точно совпадать с наименованиями полей в таблицах!

Если используется поле другой таблицы, то в префиксе через ! указывается имя данной таблицы. Например:

Стоимость партии:[Товар]![количество товара][стоимость единицы товара].*

Для задания вычислений в запросах можно использовать *Построитель выражений*. Для этого надо установить курсор в пустом столбце бланка запроса на ячейке *Поле* и вызвать команду *Построитель*.

✓ Подготовьте запрос *Выплата зарплаты*, в котором предусмотрите поля *ФИО*, *Заработная_плата*. Далее в Конструкторе добавьте поля для вычисления налога и суммы к выдаче (рис. 5.8):

*Налог:[Заработная_плата]*0,13*

К выдаче:[Заработная_плата]-[Налог]

ФИО	Заработная_плата	Налог: [Заработная_плата	К выдаче: [Заработная_плата]-[Налог]
Сотрудники	Сотрудники		
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Рис. 5.8. Создание запроса с вычислениями

Запрос с групповыми операциями. Эти запросы анализируют отдельные записи таблицы. При этом СУБД Access позволяет находить интегральные показатели для групп записей в таблице. Каждая такая группа характеризуется одинаковым значением по какому-то полю, например, одинаковой должностью или отметкой о выполнении заказа. Для перехода в данный режим запросов необходимо на панели инструментов нажать клавишу *Итоги* , что приведет к появлению в бланке запроса новой второй строки с названием *Групповая операция*. В ячейках данной строки указывается или режим группировки по некоторому полю (опция *Группировка*), или название групповой операции:

Sum - сумма значений;

Avg - среднее значение по данному полю для всей группы;

Count - число записей в данной группе;

Max - максимальное значение поля в каждой группе;

Min - минимальное значение поля в каждой группе;

First - первое значение данного поля в каждой группе;

Last - последнее значение данного поля в каждой группе и др.

Опции выбора вызываются нажатием кнопки раскрытия в требуемой ячейке.

При запуске такого запроса СУБД разбивает таблицу на группы, число которых равно числу существующих значений в группируемом поле (для него указана опция *Группировка*), и реализует для каждой группы заданную операцию (например, считает средний размер партии товара), т.е. число строк в выборке равно числу групп.

✓ Подготовьте запрос *Заказы клиентов* на основе таблицы *Заказы*. В нём предусмотрите поля *Заказчики* и *Сумма*. Щёлкнув на кнопке *Итоги*, предусмотрите группировку по заказчику (рис. 5.9):

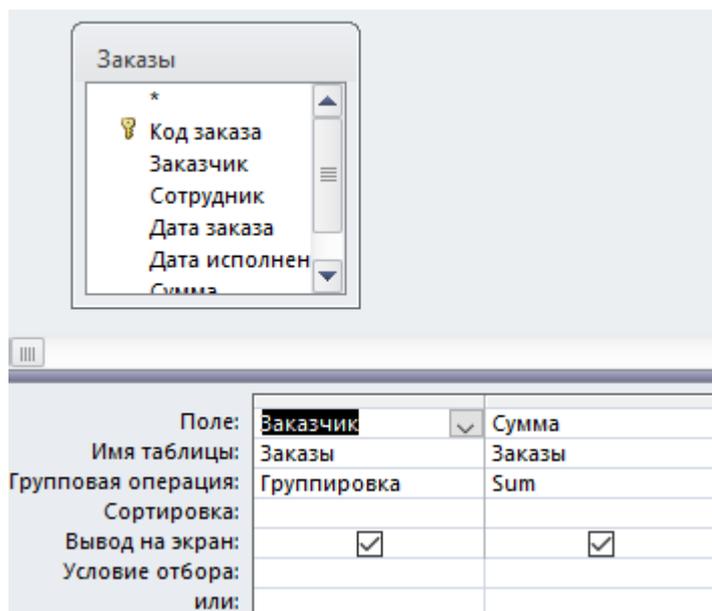


Рис. 5.9. Запрос с групповыми операциями

Результат выполнения запроса представлен на рис. 5.10.

Заказчик	Sum-Сумма
ООО Сервис	200 000,00 Р
ООО Отдых	330 000,00 Р
Бюро Турсервис	630 000,00 Р

Рис. 5.10. Результат выполнения запроса с групповыми операциями

Создание отчётов

Отчеты предназначены для вывода информации на печать. Часто данные в них располагаются в табличной форме. В отличие от распечаток таблиц или запросов отчет дает более широкие возможности сортировки и группировки данных, он предоставляет возможность добавлять итоговые значения, а также поясняющие надписи, колонтитулы, номера страниц, стили и различные графические элементы.

Создавать отчеты в базе данных Access можно с помощью *Мастера отчетов* или в режиме *Конструктора отчетов*.

Отчёты можно создавать на основе таблиц или на основе предварительно подготовленных запросов.

Наиболее универсальным средством создания отчетов является конструктор, в окне которого размещается *Бланк отчёта*.

Бланк отчета имеет структуру:

Заголовок отчёта

Верхний колонтитул

Область данных
Нижний колонтитул
Примечание отчёта

Содержимое *заголовка* и *примечания* отчета выводится (печатается) один раз, поэтому в них целесообразно включать разовую информацию: название отчета, название фирмы, её логотип, дату формирования отчета, итоговые показатели по всему отчету и другую служебную информацию.

В *верхнем* и *нижнем колонтитулах* указывается информация, отображаемая на каждой странице печатаемого отчета: номера страниц. Если отчет имеет табличную структуру, то в верхнем колонтитуле размещают шапку таблицы.

Содержание области данных в отчете отображается для каждой записи источника информации (таблицы или запроса).

✓ Создайте отчет *Ведомость на выдачу зарплаты*. Для этого вызовите команду *Создание-Мастер отчетов*. В появившемся окне в качестве источника данных для отчёта задайте *запрос Выплата зарплаты*, выберите в нём все поля и щёлкните кнопку *Далее*. Уровни группировки в отчёте не предусмотрены, поэтому в следующем окне сразу щёлкните кнопку *Далее*. На следующем шаге надо задать сортировку строк отчёта – по *ФИО*, в следующем окне выбрать *макет отчёта: табличный, книжная ориентация*. На заключительном шаге задайте имя отчёта – *Ведомость начисления заработной платы* и щёлкните кнопку *Готово*. В результате получаем отчет, представленный на рис. 5.11.

Ведомость на выдачу зарплаты			
ФИО	Заработная_плата	Налог	К выдаче
Воробьев А.А.	80 000,00 Р	10 400,00 Р	69 600,00 Р
Иванов И.П.	20 000,00 Р	2 600,00 Р	17 400,00 Р
Орлова И.А.	15 000,00 Р	1 950,00 Р	13 050,00 Р
Петров А.П.	40 000,00 Р	5 200,00 Р	34 800,00 Р
Сазонов А.Л.	20 000,00 Р	2 600,00 Р	17 400,00 Р

Рис. 5.11. Результат построения отчёта

✓ Откорректируйте созданный отчет: установите жирный шрифт для шапки таблицы и предусмотрите вычисление ИТОГО по всему отчету для графы *К выдаче*. Для этого перейдите в режим *Конструктора*, в котором вы увидите макет отчёта (рис. 5.12).

5. Как можно задать вычисления в запросах?
6. Что означает запись в условии запроса «=50»?
7. Можно ли создавать запросы на основе нескольких таблиц?
8. Для чего предназначены отчеты?
9. Назовите все области, входящие в макет отчёта
10. Какие способы создания отчетов вы знаете?
11. Как в отчетах можно посчитать итоговые значения?

Лабораторная работа № 6

Использование форм в базе данных

Цель работы: освоить на практике технологии создания форм базы данных, формирования кнопочной формы.

Задание

1. Выполнить задания, помеченные в методических указаниях по выполнению работы символом ✓

Методические указания по выполнению работы

Формы – это объекты базы данных, предназначенные для просмотра данных из таблиц и запросов, для ввода данных в таблицы, корректировки существующих данных и выполнения заданных действий. Форма может содержать графики, рисунки и другие внедренные объекты. Формы позволяют пользователям вводить данные в таблицы базы данных без непосредственного доступа к самим таблицам.

Можно вносить данные в таблицы и без помощи форм. Но использование форм создаёт ряд преимуществ:

- при работе с формами ограничен доступ к таблицам (самоу ценному в базе данных);
- разные люди могут иметь разные права доступа к информации, хранящейся в базе. Для ввода данных им предоставляются разные формы, хотя данные из форм могут поступать в одну таблицу;
- вводить данные в форму легче, чем в таблицу, и удобнее, так как в окне формы отображается, как правило, одна запись таблицы;
- в большинстве случаев информация для баз данных берется из бумажных бланков (анкет, счетов, накладных, справок и т. д.). Экранные формы можно сделать точной копией бумажных бланков, благодаря этому уменьшается количество ошибок при вводе и снижается утомляемость персонала;
- возможность отображения результатов вычисления по каждой записи;
- возможность скрыть при просмотре части данных БД;
- возможность разрешения на модификацию данных только у части полей;
- возможность размещения в форме управляющих элементов (командных кнопок, кнопок раскрытия и т.п.), что превращает формы в объекты, подобные диалоговым окнам Windows.

При работе с формой для перемещения по всему множеству записей таблицы можно использовать:

- клавиатуру: **PgUp** - переход к следующей записи, **PgDn** - возврат к предыдущей записи, **Home** - переход к 1-й записи таблицы, **End** - переход к последней записи таблицы;
- **кнопки навигации**, расположенные внизу формы, активизируемые левой клавишей мыши (рис. 6.1).

Создавать формы можно на основе нескольких таблиц или запросов.

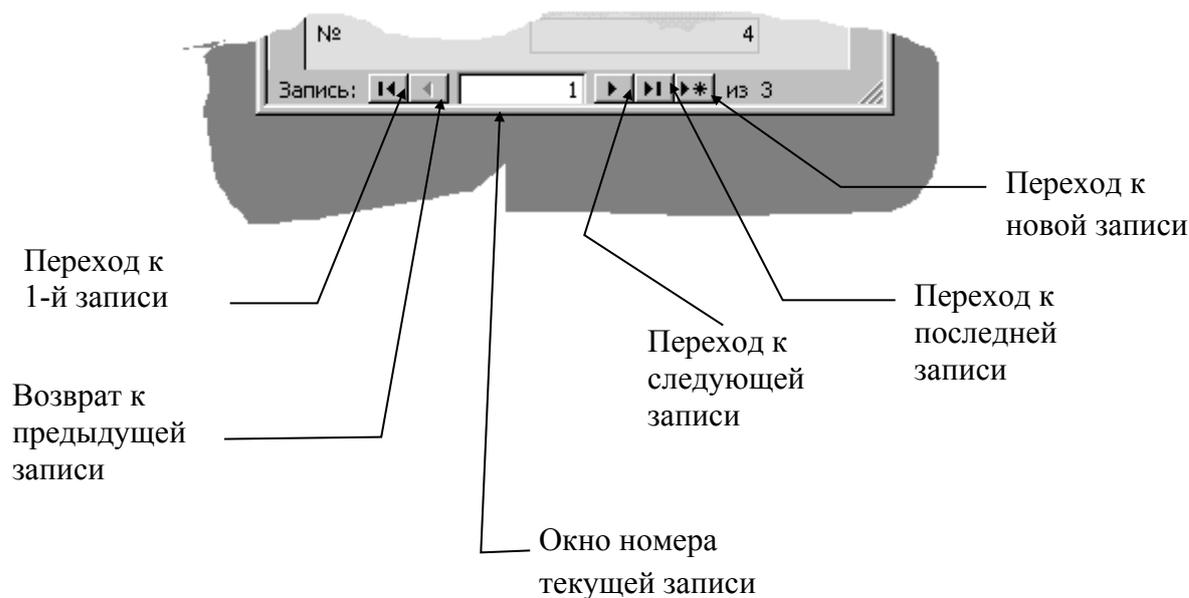


Рис. 6.1. Кнопки навигации в форме

Существует несколько способов создания форм:

- с помощью **Мастера форм**;
- автоматически, используя команду **Создание-Форма**;
- в режиме **Конструктора форм**;
- сохраняя таблицу или запрос как форму.

Рассмотрим некоторые из перечисленных способов.

✓ Создайте форму с именем *Сотрудники ввод* для ввода данных в таблицу *Сотрудники*.

- Вызовите команду **Создание - Мастер форм**.
- В диалоговом окне **Создание форм** выберите таблицу *Сотрудники* и *все поля* для размещения их в форме. Щелкните по кнопке *Далее*.
- В следующих диалоговых окнах мастера выберите внешний вид формы (в один столбец, ленточный, табличный, выровненный), задайте имя формы - *Сотрудники ввод*. Щелкните по кнопке *Готово*.

✓ Аналогично создайте формы *Клиенты ввод*, *Заказы ввод*.

Созданную любым способом форму можно затем изменять в режиме **Конструктора форм**.

Конструктор работает с формой как с совокупностью объектов. Объекты, в свою очередь, характеризуются набором определенных свойств и действий.

Бланк формы включает несколько областей, главной из которых является область данных, содержащая поля таблицы. В заголовке и примечании отображается служебная и/или пояснительная информация.

Размер каждой области бланка может быть изменен путем установки мыши на ее границу, захвата границы левой клавишей мыши и перетаскивания её на новое место.

Размер самого бланка формы также может быть изменен мышью путем перетаскивания его границ.

Набор инструментов, доступных в режиме конструктора и позволяющих формировать собственную структуру формы, расположен в опции *Элементы управления* вкладки *Конструктор* панели *Инструменты конструктора форм*.

Свойства объекта представлены на пяти вкладках:

1. **Макет** - содержит перечень свойств, связанных с оформлением объекта.
2. **Данные** - содержит перечень свойств, связанных с источником данных.
3. **События** - содержит перечень доступных для объекта событий.
4. **Другие** - содержит перечень свойств, не вошедших в три первых категории.
5. **Все** - содержит перечень всех событий и свойств в алфавитном порядке.

Значения свойств можно изменять: путем ввода с клавиатуры новых значений в окно; выбора из списка; настройкой в диалоговом окне.

Все объекты в процессе настройки формы можно перемещать; удалять; менять в размерах.

Одиночный объект выделяется однократным щелчком левой клавишей мыши, а группа объектов – с помощью кнопки **Выбрать** или щелчком левой клавиши мыши при нажатой клавише *Ctrl*.

Размер объекта можно менять путем:

- Выделения объекта, что приводит к появлению по его периметру рамки с маркерами изменения размера. При установке на один из таких маркеров указатель мыши приобретает вид двунаправленной стрелки. Захват мышью маркера и его перемещение приводят к изменению размера объекта;
- Точной установкой размеров объекта в полях **Ширина** и **Высота** окна свойств.

Перемещение объекта реализуется путем его выделения и перетаскивания мышью.

Удаляется объект с бланка формы путем выделения и нажатия клавиши **Delete** клавиатуры.

Примечание: Переход к общим свойствам всей формы реализуется вызовом контекстного меню в окне формы, но вне пределов его поля.

Для *текстовых объектов* формы доступны изменения следующих параметров: размера; цвета текста; цвета фона; толщины; наклона; подчеркивания. Изменение этих свойств выполняется в панели **Инструменты конструктора форм** стандартным образом.

Для *графических объектов* доступны следующие параметры: цвет линии/границы; толщина линии/границы; вид рамки.

✓ В заголовке формы *Сотрудники_ввод* разместите текст *Сотрудники фирмы* (жирным шрифтом).

Для полного устранения поля с бланка формы необходимо его выделить и одним из рассмотренных ранее способов удалить.

Новое поле размещается в бланке формы путем нажатия кнопки *Поле* на панели *Элементы управления* и щелчка левой клавишей мыши в месте его будущего расположения на бланке формы, что приводит к появлению связанного объекта, состоящего из поля ввода и его надписи. Далее вызываются его свойства, и на вкладке *Данные* в одноименном свойстве выбирается связываемое с ним требуемое поле таблицы или задаётся выражение для расчета значений, отображаемых в данном поле. Правила построения выражений рассматривались ранее применительно к формированию запросов. Выражение можно просто внести в окно требуемого свойства в соответствии с синтаксисом языка VisualBasic. Надпись поля, при необходимости, модифицируется.

✓ Создайте Форму *Просмотр сведений о сотрудниках*, разместив в ней поля *ФИО*, *Должность*, *Заработная плата*. Предусмотрите в форме вычисление налога (13% от *Заработной платы*).

Форма также позволяет вводить данные в таблицу на основе фиксированных списков или значений, определяемых связанными таблицами. Для этого в форме должно быть размещено поле со списком.

Специализированные формы

Выше рассматривались универсальные формы, которые можно использовать как для ввода информации в таблицы, так и для просмотра и/или модификации. Однако формы позволяют дифференцировать доступ к данным путем установки следующих режимов:

- запрет на изменение существующих данных;
- запрет на добавление новых данных;
- запрет на изменение значений некоторых полей в форме;
- отображение в форме только части полей исходной таблицы.

Два первых режима устанавливаются путем вызова окна общих свойств формы и задания значений *Нет* для опций *Разрешить добавление*, *Разрешить удаление* и/или *Разрешить изменение* на вкладке *Данные*.

Третий режим предполагает, что вызывается окно свойств конкретного поля и на вкладке *Данные* задается значение *Да* для свойства *Блокировка*. Если ставится задача временного устранения с бланка формы некоторого поля, то на вкладке *Данные* задается значение *Нет* для свойства *Доступ*. При этом само поле деактивировано, но отображается в форме в теновом режиме, т.е. сохраняется возможность изменения его свойств.

✓ Задайте для формы *Просмотр сведений о сотрудниках* свойства таким образом, чтобы она была доступна только для просмотра сведений.

С помощью команды конструктора форм *Эмблема* в форме можно разместить графические изображения, используемые как информационные объекты или элементы оформления бланка. Для этого необходимо после вызова данной команды на панели инструментов выбрать положение будущего изображения на бланке формы щелчком левой клавиши мыши, а далее указать положение графического файла в памяти компьютера.

Если рисунок используется в качестве фона для формы, необходимо его предварительно растянуть по размерам формы и в опции *Положение* вкладки *Упорядочить* панели *Инструменты конструктора форм* установить опцию *На задний план*.

✓ В примечание формы *Просмотр сведений о сотрудниках* добавьте объект *Кнопка* (*Конструктор* - панель инструментов *Элементы управления* - *Кнопка*).

После того как вы «нарисуете» кнопку указателем, на экране появится диалоговое окно *Создание кнопок* (рис. 6.2).

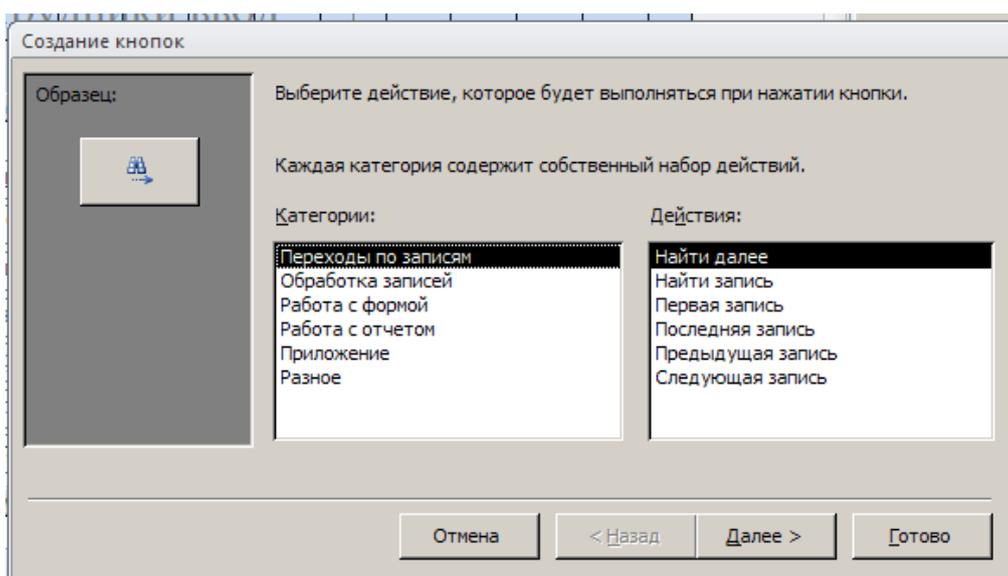


Рис. 6.2. Создание кнопок на форме

- В категории *Работа с формой* выберите действие *Закрытие формы* и нажмите кнопку *Далее*.
 - Выберите рисунок или текст, который будет размещаться на кнопке.
 - В последнем диалоговом окне *Мастера кнопок* задайте имя кнопки и нажмите *Готово*.
 - *Мастер кнопок* создал для данной кнопки процедуру на языке Microsoft Visual Basic. Просмотреть процедуру обработки события можно с помощью команды *Обработка событий* контекстного меню кнопки.
- ✓ Самостоятельно создайте кнопки *Поиск записи*, *Удаление записи*.

Иногда на форме требуется разместить несколько страниц, содержащих данные из различных источников, справочную или вспомогательную информацию. Для этой цели можно использовать набор вкладок.

✓ Создайте форму *Ввод данных*, на которой предусмотрите вкладки с именами *Сотрудники*, *Клиенты*, *Заказы*. На каждой вкладке разместите соответствующую форму.

- Создайте пустую форму.
- Для добавления к форме набора вкладок щелкните по кнопке *Вкладка* на панели инструментов *Элементы управления*. Сначала добавятся только две вкладки с формальными именами *Вкладка 1* и *Вкладка 2*.
- Добавьте еще одну вкладку. Для этого надо вызвать контекстное меню, предварительно щёлкнув на ярлычке вкладки, и вызвать команду *Вставить вкладку*.
- Переименуйте ярлычки вкладок на *Сотрудники*, *Клиенты*, *Заказы*. Для этого измените свойство *Имя* в окне свойств соответствующей вкладки.
- Перейдите на вкладку *Сотрудники* и перетащите на нее мышкой из базы данных форму *Сотрудники ввод*.
- Аналогичным образом поместите формы *Клиенты ввод* и *Заказы ввод* на соответствующие вкладки.
- Данную форму сохраните с именем *Ввод данных*.

В *Microsoft Access* можно создавать кнопочные формы. Они содержат только кнопки и предназначены для выбора основных действий в базе данных.

✓ Для создания кнопочной формы необходимо на вкладке ленты *Работа с базами данных* выбрать команду *Диспетчер кнопочных форм*.

- Если кнопочной формы в базе данных нет, то будет выведен запрос на подтверждение ее создания. Нажмите *Да* в диалоговом окне подтверждения.
- Перед вами появится *Диспетчер кнопочных форм*, в котором щелкните по кнопке *Создать*.
- В диалоговом окне *Создание* (рис. 6.3) введите имя новой кнопочной формы, например, *Фирма Ирина*, и нажмите *ОК*.

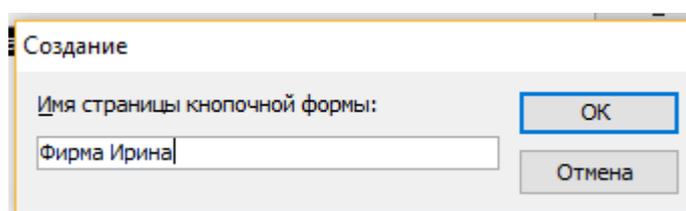


Рис. 6.3. Задание имени кнопочной формы

- Имя новой кнопочной формы добавится в список *Страницы кнопочной формы* окна *Диспетчер кнопочных форм* (рис. 6.4). Выделите имя новой кнопочной формы и щелкните по кнопке *Изменить*.

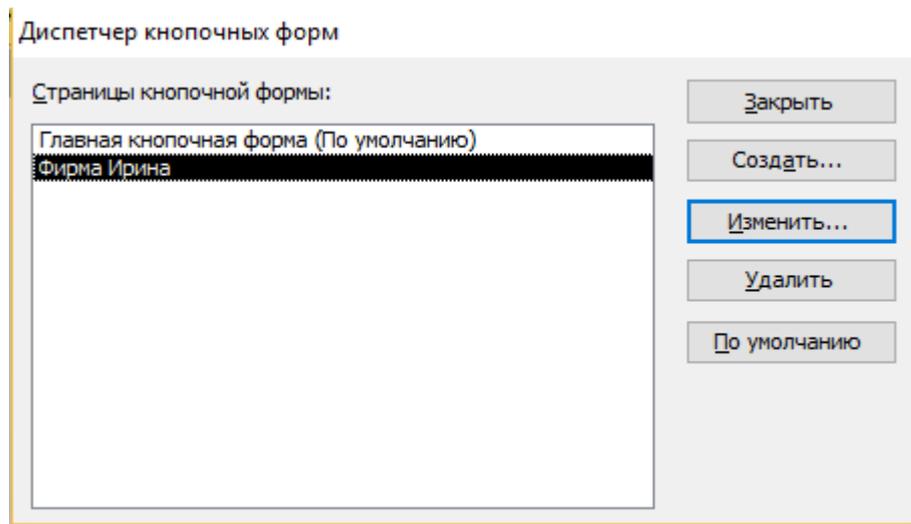


Рис. 6.4. Диспетчер кнопочных форм

- В диалоговом окне *Изменение страницы кнопочной формы* щелкните по кнопке *Создать*. Появится диалоговое окно *Изменение элемента кнопочной формы* (рис. 6.5).

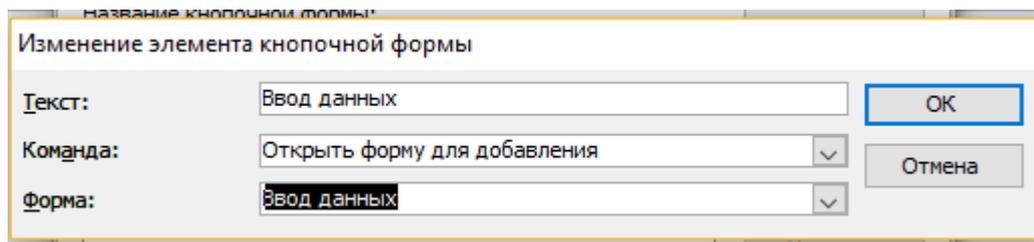


Рис. 6.5. Создание кнопок на форме

- В поле *Текст* введите текст подписи для первой кнопки кнопочной формы, а затем выберите команду из раскрывающегося списка в поле *Команда*. В поле *Форма* выберите форму, для которой будет выполняться данная команда.
- Аналогичным образом добавьте кнопки *Просмотр сведений о сотрудниках*, *Ведомость на выдачу зарплаты*, *Выход*.
- В диалоговом окне *Диспетчер кнопочных форм* выберите имя вашей кнопочной формы и щелкните по кнопке *По умолчанию*. Рядом с названием кнопочной формы появится надпись «(по умолчанию)».
- Чтобы закончить создание кнопочной формы, щелкните по кнопке *Заккрыть*.

✓ Чтобы главная кнопочная форма появлялась на экране при запуске приложения, необходимо в главном меню нажать на кнопку *Параметры*. Для текущей базы данных установите форму просмотра – «кнопочная форма».

Контрольные вопросы

1. Для чего предназначены формы?

2. Какие преимущества при работе с базой данных даёт форма?
3. Какие средства создания форм вы знаете?
4. На основе чего можно создавать формы?
5. Как создать кнопку на форме?
6. Как создать вкладки на форме?
7. Как создать форму, предназначенную только для просмотра данных?
8. Что такое кнопочная форма?
9. Как создать главную кнопочную форму?

Лабораторная работа № 7.

Самостоятельная работа по созданию базы данных на основе СУБД Access

Цель работы: закрепить знания и навыки по созданию и работе с базами данных.

Задание

1. Создать базу данных Access в соответствии с предложенным вариантом.

Вариант 1

1. Разработайте базу данных *Библиотека*, состоящую из трех таблиц со следующей структурой:

Книги – шифр книги, автор, название, год издания, количество экземпляров.

Читатели – читательский билет, фамилия, имя, отчество, адрес.

Выданные книги – шифр книги, читательский билет, дата выдачи, плановая дата возвращения, дата фактического возвращения.

Задайте ключевые поля в таблицах. В структуре таблицы *Выданные книги* предусмотрите поля подстановки.

2. Создайте схему данных, обеспечивающую поддержку ссылочной целостности данных.

3. Подготовьте формы для ввода данных в таблицы.

4. С помощью форм введите данные в таблицы (5-10 строк).

5. Создайте запрос для отбора всех книг, выпущенных с 1990 по 2007 годы.

6. Создайте запрос с параметром для отбора книг определенного автора.

7. Подготовьте отчет – Список книг, находящихся на руках у читателей.

Вариант 2

1. Разработайте базу данных *Продуктовый магазин*, которая состоит из трёх таблиц со следующей структурой:

Поставщики – код поставщика, название поставщика, адрес поставщика, телефон поставщика.

Товары – код товара, код поставщика, наименование товара, цена за единицу товара.

Продажа товаров – код товара, дата продажи, проданное количество товара.

Задайте ключевые поля в таблицах. В структуре таблиц *Продажа товаров* и *Товары* предусмотрите поля подстановки.

2. Создайте схему данных, обеспечивающую поддержку ссылочной целостности данных.

3. Подготовьте формы для ввода данных в таблицы.

4. С помощью форм введите данные в таблицы (5-10 строк).

5. Создайте запрос для отбора товаров, цены которых от 100 до 450 руб.

6. Создайте запрос с параметром для отбора проданных товаров от заданного поставщика.

7. Создайте отчёт *Продажи за день*. Дату должен вводить пользователь.

Вариант 3

1. Разработайте базу данных *Сессия*, состоящую из четырех таблиц со следующей структурой:

Студенты – номер студенческого билета, ФИО, группа.

Дисциплины – шифр дисциплины, название дисциплины, количество часов.

Экзамены – номер студенческого билета, шифр дисциплины, дата, оценка.

Зачеты – номер студенческого билета, дата, шифр дисциплины, зачет.

Задайте ключевые поля в таблицах. В структуре таблиц *Экзамены* и *Зачеты* предусмотрите поля подстановки.

2. Создайте схему данных, обеспечивающую поддержку ссылочной целостности данных.

3. Подготовьте формы для ввода данных в таблицы.

4. С помощью форм введите данные в таблицы (5-10 строк).

5. Создайте запрос с параметром для отбора студентов, сдавших экзамен по заданному предмету на 4 или 5.

6. Создайте запрос с параметром для отбора студентов, не получивших зачет по заданному предмету.

7. Подготовьте отчёт *Ведомость сдачи экзамена по заданному предмету для заданной группы*.

Вариант 4

1. Разработайте базу данных «*Оптовый склад*», состоящую из четырех таблиц со следующей структурой:

Склад – код товара, количество, дата поступления.

Товары – код товара, название товара, единица измерения, цена за единицу, срок хранения.

Заявки – код заявки, название организации, код товара, требуемое количество. *Отпуск товаров* – код заявки, код товара, отпущенное количество, дата отпуска товара.

Задайте ключевые поля в таблицах. В структуре таблиц *Заявки* и *Отпуск товаров* предусмотрите поля подстановки.

2. Создайте схему данных, обеспечивающую поддержку ссылочной целостности данных.

3. Подготовьте формы для ввода данных в таблицы.

4. С помощью форм введите данные в таблицы (5-10 строк).

5. Создайте запрос с параметром для отбора товаров, поступивших на склад какого-либо числа.

6. Создайте запрос с параметром для отбора сведений о наличии заданного товара на складе

7. Подготовьте отчёт *Отпуск товаров за день*. Дату должен вводить пользователь.

Вариант 5

1. Разработайте базу данных Домашняя видеотека, состоящую из трех таблиц со следующей структурой:

Фильмы на CD – шифр фильма, название, режиссёр, год выхода;

Мои знакомые – ФИО, телефон, адрес.

Просмотр – шифр фильма, ФИО, дата выдачи, плановая дата возвращения, дата фактического возвращения.

Задайте ключевые поля в таблицах. В структуре таблицы *Просмотр* предусмотрите поля подстановки.

2. Создайте схему данных, обеспечивающую поддержку ссылочной целостности данных.

3. Подготовьте формы для ввода данных в таблицы.

4. С помощью форм введите данные в таблицы (5-10 строк).

5. Создайте запрос для отбора всех фильмов, выпущенных с 2010 по 2015 годы.

6. Создайте запрос с параметром для отбора фильмов определенного режиссёра.

7. Подготовьте отчёт – Список фильмов, выданных знакомым для просмотра и не возвращённых.

Вариант 6

1. Разработайте базу данных «Транспортные перевозки», состоящую из трех таблиц со следующей структурой:

Транспорт – марка автомобиля, государственный номер, расход топлива.

Заявки – код заявки, дата заявки, название груза, количество груза, пункт отправления, пункт назначения.

Доставка – №п/п, код заявки, дата отправления, дата прибытия, государственный номер автомобиля, пройденное расстояние.

Задайте ключевые поля в таблицах. В структуре таблицы *Доставка* предусмотрите поля подстановки.

2. Создайте схему данных, обеспечивающую поддержку ссылочной целостности данных.

3. Подготовьте формы для ввода данных в таблицы.

4. С помощью форм введите данные в таблицы (5-10 строк).

5. Создайте запрос для отбора всех заявок, выполненных определённым автомобилем.

6. Создайте запрос с параметром для отбора заявок, выполненных в определённый пункт назначения.

7. Подготовьте отчёт – список выполняемых на данный момент заявок.

Вариант № 7

1. Разработайте базу данных «*Прокат спортивного оборудования*», состоящую из трех таблиц со следующей структурой:

Клиенты – код клиента, ФИО, телефон, адрес, паспортные данные, залог.

Склад – код оборудования, название, количество, залоговая стоимость, остаток.

Прокат – №п/п, ФИО, код оборудования, дата выдачи, плановая дата возврата, фактическая дата возврата, оплата проката.

Задайте ключевые поля в таблицах. В структуре таблицы *Прокат* предусмотрите поля подстановки.

2. Создайте схему данных, обеспечивающую поддержку ссылочной целостности данных.

3. Подготовьте формы для ввода данных в таблицы.

4. С помощью форм введите данные в таблицы (5-10 строк).

5. Создайте запрос для отбора оборудования, выданного за определённую дату.

6. Создайте запрос с параметром для отбора клиентов, у которых залог больше 5000 руб..

7. Подготовьте отчёт – список оборудования, находящегося на руках у клиентов

Вариант 8

1. Разработайте базу данных *Фотоконкурс*, состоящую из трех таблиц со следующей структурой:

Авторы – ФИО, адрес, телефон, место проживания, дата рождения.

Работы – код работы, название, ФИО, жанр, тематика, год съёмки, место съёмки, дата начала демонстрации, баллы

Отзывы о работах – код работы, отзыв.

Задайте ключевые поля в таблицах. В структуре таблиц *Работы*, *Рейтинг работ* предусмотрите поля подстановки.

2. Создайте схему данных, обеспечивающую поддержку ссылочной целостности данных.

3. Подготовьте формы для ввода данных в таблицы.

4. С помощью форм введите данные в таблицы (5-10 строк).

5. Создайте запрос для отбора всех работ по заданной тематике.

6. Создайте запрос с параметром для отбора работ определённого автора.

7. Подготовьте отчёт – *Список работ, представленных на выставку в текущем году.*

Вариант 9

1. Разработайте базу данных «*Туристическая фирма*», состоящую из четырех таблиц со следующей структурой:

Клиенты – код клиента, ФИО, телефон, адрес, паспорт.

Сотрудники – код сотрудника, ФИО сотрудника, должность, телефон, адрес, паспортные данные.

Туристические маршруты – код маршрута, название, описание маршрута, страна, стоимость путевки, количество дней, вид транспорта.

Заказы – код заказа, код клиента, код маршрута, код сотрудника (менеджер, оформивший заказ), дата, отметка об оплате.

Задайте ключевые поля в таблицах. В структуре таблиц *Заказы* предусмотрите поля подстановки.

2. Создайте схему данных, обеспечивающую поддержку ссылочной целостности данных.

3. Подготовьте формы для ввода данных в таблицы.

4. С помощью форм введите данные в таблицы (5-10 строк).

5. Создайте запрос для отбора всех маршрутов со стоимостью от 10000 до 20000 руб.

6. Создайте запрос с параметром для отбора клиентов, выбравших маршрут в определённую страну.

7. Подготовьте отчёт – *Заказы сотрудников в текущем году*.

Вариант 10

1. Разработайте базу данных *Автосервис*, состоящую из трех таблиц со следующей структурой:

Клиенты – код клиента, ФИО, адрес, телефон.

Виды работ – код работы, описание работы, стоимость.

Заказы – номер заказа, дата заказа, плановая дата выполнения заказа, фактическая дата выполнения заказа, код работы.

Задайте ключевые поля в таблицах. В структуре таблицы *Заказы* предусмотрите поля подстановки.

2. Создайте схему данных, обеспечивающую поддержку ссылочной целостности данных.

3. Подготовьте формы для ввода данных в таблицы.

4. С помощью форм введите данные в таблицы (5-10 строк).

5. Создайте запрос для отбора всех работ, выполненных до заданной даты.

6. Создайте запрос с параметром для отбора заказов по заданному клиенту.

7. Подготовьте отчёт – *Список заказов, находящихся в работе*.

Методические указания по выполнению работы

При выполнении задания в качестве справочного материала использовать методические указания по выполнению работ 4 – 6.

Список литературы

1. Гаврилов, М. В. Информатика и информационные технологии: учебник/ М. В. Гаврилов; В.А. Климов. - 2-е изд., испр. и доп. - Москва: Юрайт, 2011. – 350с.
2. Гайдюков, А.А. Информатика. Информационные технологии. Решение математических задач в Excel: учеб.-метод. пособие для студ. вузов общетехн. спец./ А. А. Гайдюков. - Калининград: Издательство ФГОУ ВПО "КГТУ", 2010. – 33с.
3. Информатика: учебник/ под ред. В. В. Трофимова. – Москва: Юрайт, 2011. – 911с.
4. Топоркова, О.М. Учебно-методическое пособие по лабораторным работам по использованию табличного процессора MS Excel 2010 для студентов, обучающихся в бакалавриате по направлениям подготовки: «Информатика и вычислительная техника», «Прикладная информатика»/ О.М. Топоркова. – Калининград: Издательство ФГБОУ ВО «КГТУ», 2017. – 36с.
5. Арунянц, Г.Г. Базовые информационные технологии в экономике: учеб. пособие для студ. эконом. спец. вузов/ Г.Г. Арунянц, С.А. Калинина, Г.В. Ломакина, Г.К. Петрикина. – Калининград, Издательство КГТУ, 2008. – 432с.
6. Создание баз данных в Microsoft Access 2007. Метод. указ. по вып. лаб. раб. для студ. вузов./ Федер. агентство по образованию, ГОУ ВПО УГТУ-УПИ; сост. В.М. Паклина, Е.М. Паклина. – Екатеринбург: ГОУ ВПО УГТУ-УПИ, 2009. – 41с.