

**МИНИСТЕРСТВО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ**

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ**

**ВЫСШЕГО ОБРАЗОВАНИЯ**

**«ИВАНОВСКИЙ ГОСУДАРСТВЕННЫЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

**КАФЕДРА ИНФОРМАЦИОННЫХ СИСТЕМ И ТЕХНОЛОГИЙ**

**СОЗДАНИЕ РЕЛЯЦИОННОЙ БАЗЫ ДАННЫХ**

**В ПРИЛОЖЕНИИ MICROSOFT ACCESS**

**МЕТОДИЧЕСКИЕ УКАЗАНИЯ И КОНТРОЛЬНЫЕ ЗАДАНИЯ ПО ИНФОРМАТИКЕ**

**ДЛЯ СТУДЕНТОВ ВСЕХ СПЕЦИАЛЬНОСТЕЙ**

**ИВАНОВО 2016**

УДК 681.3 (076)

Составители: Ахмадулина Ю.С., Слизнева Т.Е.

Создание реляционной базы данных в приложении Microsoft Access: Методические указания и контрольные задания по информатике для студентов всех специальностей / Ивановский государственный политехнический университет; Сост.: Ю. С. Ахмадулина, Т. Е. Слизнева. - Иваново, 2016. - 34 с.

Методические указания содержат информацию для выполнения контрольной работы «СУБД Access». Задания на создание СУБД Access представлены в 10 вариантах. Номер варианта соответствует последней цифре студенческого билета.

В методическом указании приводятся основные сведения о моделях данных, используемых при построении баз данных, рассматриваются примеры создания баз данных в приложении MS Access, даются рекомендации, необходимые для выполнения работы, задаются контрольные вопросы.

Методические указания предназначены для студентов всех специальностей.

Рецензент: доктор технических наук,  
профессор кафедры ВПМСИТ Н. А. Коробов

## Оглавление

Введение .....	1
Порядок выполнения лабораторной работы .....	8
Лабораторная работа № 1 «Создание таблиц» .....	10
Лабораторная работа № 2 «Создание связей между таблицами» .....	12
Лабораторная работа № 3 «Создание запросов» .....	17
Лабораторная работа № 4 «Создание и использование форм» .....	22
Лабораторная работа № 5 «Создание отчетов» .....	27
Варианты заданий .....	29
Библиографический список .....	34

## Введение

Под базой данных понимают совокупность сведений о конкретных объектах реального мира в какой-либо предметной области. Создавая базу данных, пользователь стремится упорядочить информацию по различным признакам и быстро извлекать нужную информацию с произвольным сочетанием признаков. Сделать это возможно, только если данные структурированы.

Структурирование - это введение соглашений о способах представления данных.

**База данных** – это поименованная совокупность структурированных данных, организованных таким образом, что возможен быстрый доступ к информации, представляющей наибольший интерес в настоящий момент.

С понятием базы данных тесно связано понятие системы управления базой данных (СУБД).

**СУБД** - это комплекс программных и языковых средств, предназначенных для создания структуры новой базы, наполнения ее содержимым, редактирования содержимого и организации поиска необходимой информации.

Ядром любой базы данных является модель данных. Модель данных представляет собой множество структур данных, ограничений целостности и операций манипулирования данными. С помощью модели данных могут быть представлены объекты предметной области и взаимосвязи между ними.

**Модель данных** - совокупность структур данных и операций их обработки.

Системы управления базами данных основываются на использовании иерархической, сетевой или реляционной модели, на комбинации этих моделей или на некотором их подмножестве. СУБД Access относится к реляционным базам данных.

### *Реляционная модель данных*

Понятие реляционный (англ. relation - отношение) связано с разработками известного американского специалиста в области систем баз данных Е. Кодда. Эти модели характеризуются простотой структуры данных, удобным для пользователя табличным представлением и возможностью использования формального аппарата алгебры отношений и реляционного исчисления для обработки данных.

**Реляционная модель** ориентирована на организацию данных в виде двумерных таблиц. Каждая реляционная таблица представляет собой двумерный массив и **обладает следующими свойствами:**

- 1) каждый элемент таблицы - один элемент данных;
- 2) все столбцы в таблице однородные, т.е. все элементы в столбце имеют одинаковый тип (числовой, символьный и т.д.) и длину;
- 3) каждый столбец имеет уникальное имя;
- 4) одинаковые строки в таблице отсутствуют;
- 5) порядок следования строк и столбцов может быть произвольным.

### *Реляционные БД*

Базы данных, основанные на использовании реляционных моделей данных, называются реляционными БД. Простейшая база данных имеет хотя бы одну таблицу. Структуру двумерной таблицы образуют столбцы и строки. Их аналогами в базе данных являются поля и записи.

**Поле (столбец)** – набор одной и той же характеристики (например, номер телефона) для нескольких объектов.

**Запись (строка)** - набор нескольких характеристик (например, табельный номер, фамилия, телефон, адрес, должность и др.) для одного и того же объекта.

#### *Разработка структуры реляционной базы данных:*

- 1) Составляют генеральный список полей.
- 2) Определяют наиболее подходящий тип данного для каждого поля.
- 3) Разделяют поля генерального списка по базовым таблицам.

На первом этапе разделение производят по функциональному признаку (цель - обеспечить ввод данных в одну таблицу на одном рабочем месте). На втором этапе приступают к дальнейшему делению таблиц. При этом целью деления является исключение повторяющихся записей в таблице.

- 4) В каждой из таблиц намечают ключевое поле.

**Ключевое поле** – это поле, которое однозначно определяет каждую запись в таблице. Например, для таблицы данных о студентах таким полем может служить индивидуальный шифр студента.

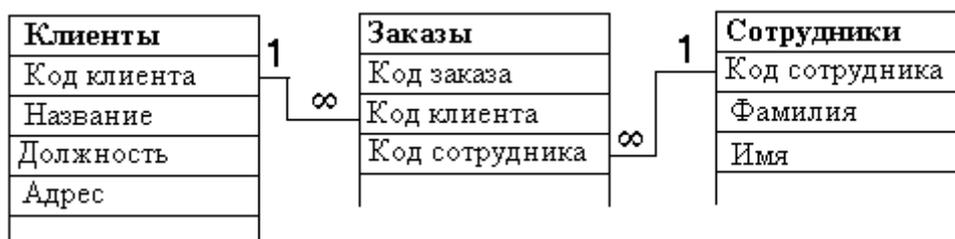
Если такого поля нет, то в качестве ключа можно выбрать комбинацию полей, которая также однозначно определяет каждую запись в таблице. Например, для таблицы, в которой содержатся расписания занятий, в качестве ключа можно выбрать комбинацию полей "Время занятий" и "Номер аудито-

рии". Эта комбинация неповторима (т.к. в одной аудитории в одно и то же время не идут разные занятия).

Если в таблице вообще нет никаких полей, которые можно было бы использовать как ключевые, всегда можно ввести дополнительное поле типа "Счетчик": оно не может содержать повторяющихся данных по определению.

5) Устанавливают связи между таблицами.

Связанные таблицы называют **схемой данных**. Существует несколько типов возможных связей между таблицами. Наиболее распространенными являются связи "**один ко многим**" и "**один к одному**". Связь между таблицами организуется на основе общего поля, причем в одной из таблиц оно обязательно должно быть ключевым, т.е. на стороне "один" должно выступать ключевое поле, содержащее уникальные, неповторяющиеся значения. Значения на стороне "Многие" могут повторяться.



Про подобные таблицы говорят, что они связаны реляционными отношениями. Системы управления, способные работать со связанными таблицами, называют системами управления реляционными базами данных.

Итак, в реляционной базе данных используется несколько разных таблиц, между которыми устанавливаются связи (relation). Они позволяют ввести информацию в одну таблицу и связать ее с записями другой через специальный идентификатор. При этом сокращается общее количество информации, хранимой в базе данных, поскольку в записях повторяются уже не сами данные, а только идентификаторы для связывания.

#### *Основные объекты базы данных*

Основными компонентами (объектами) базы данных являются:

- таблицы,
- запросы,
- формы,
- отчеты,
- макросы и модули.

**Таблица** — фундаментальная структура системы управления реляционными базами данных. В Microsoft Access таблица — это объект, предназначенный для хранения данных в виде записей (строк) и полей (столбцов). Каждое поле содержит отдельную часть записи, например фамилию, должность или инвентарный номер. Обычно каждая таблица используется для хранения сведений по одному конкретному вопросу, например о сотрудниках или заказах.

**Запрос** — вопрос о данных, хранящихся в таблицах, или инструкция на отбор записей, подлежащих изменению. Ниже перечислены типы запросов, которые могут быть созданы с помощью Microsoft Access:

1) запрос-выборка задает вопрос о данных, хранящихся в таблицах, и представляет полученный динамический набор в режиме формы или таблицы без изменения данных. Изменения, внесенные в динамический набор, отражаются в базовых таблицах;

•2) запрос-изменение изменяет или перемещает данные. К этому типу относятся запрос на добавление записей, запрос на удаление записей, запрос на создание таблицы или запрос на обновление;

3) перекрестные запросы предназначены для группирования данных и представления их в компактном виде;

4) запрос с параметрами позволяет определить одно или несколько условий отбора во время выполнения запроса;

5) запросы SQL — это запросы, которые могут быть созданы только с помощью инструкций SQL в режиме SQL: запрос-объединение, запрос к серверу и управляющий запрос. Язык SQL (Structured Query Language) — это язык структурированных запросов, который часто используется при анализе, обновлении и обработке реляционных баз данных (например, Microsoft Access).

**Форма** — это объект Microsoft Access, в котором можно разместить элементы управления, предназначенные для ввода, изображения и изменения данных в полях таблиц.

**Отчет** — это объект Microsoft Access, который позволяет представлять определенную пользователем информацию в определенном виде, просматривать и распечатывать ее.

**Макрос** — одна или несколько макрокоманд, которые можно использовать для автоматизации конкретной задачи. Макрокоманда — основной строительный блок макроса; самостоятельная инструкция, которая может быть объединена с другими макрокомандами, чтобы автоматизировать выполнение задачи.

**Модуль** — набор описаний, инструкций и процедур, сохраненных под одним именем. В Microsoft Access имеется три типа модулей: модуль формы, отчета и общий модуль. Модули форм и отчетов содержат локальную программу для форм или отчетов. Если процедуры общего модуля явным образом не объявлены личными для модуля, в котором они появляются, они распознаются и могут вызываться процедурами из других модулей этой базы данных.

База данных может содержать несколько модулей, в том числе общие модули, модули форм и модули отчетов.

#### *Типы данных СУБД Access*

Для каждого поля таблиц базы данных необходимо указать тип данных. Тип данных определяет вид и диапазон допустимых значений, которые могут быть введены в поле, а также объем памяти, выделяющийся для этого поля. Перечень типов данных полей и описание значений, сохраняемых в таких полях, приведены в следующей таблице:

<b>Тип</b>	<b>Описание</b>
Текстовый	Используется для хранения символьных или числовых данных, не требующих вычислений. В свойстве Размер поля задается максимальное количество символов, которые могут быть введены в данное поле. По умолчанию размер устанавливается в 50 знаков. 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	Предназначен для хранения числовых данных, используемых в математических расчетах. На вкладках Общие и Подстановка можно установить свойства числового поля, среди которых Размер поля, Формат поля, Число десятичных знаков.
Дата/Время	Используется для представления даты и времени. Выбор конкретного формата даты или времени устанавливается в свойстве.
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) или случайных чисел в качестве номера новой записи. Номер, присвоенный записи, не мо-

	жет 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», «Вкл. / Выкл.».
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	Предназначен для автоматического определения поля. С его помощью будет создано поле со списком, из которого можно выбирать данные, содержащиеся в другой таблице или в наборе постоянных значений.

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

На примере СУБД **Microsoft Access**, входящей в состав пакета Microsoft Office, познакомимся с основными типами данных, способами создания баз данных и с приемами работы с базами данных. Основное назначение Access – дать пользователю простое и доступное средство для создания таких баз данных, которые ему нужны. Несмотря на простоту этой СУБД, она позволяет создавать приложения довольно сложной структуры. При желании систему можно развивать и настраивать собственными силами, используя язык программирования Visual Basic. Еще одним достоинством Access является ее интегрированность с Word, Excel и другими программами пакета Microsoft Office. Данные легко импортируются и экспортируются из одного приложения в другое.

Методические указания включают в себя пять лабораторных работ, индивидуальные задания и перечень контрольных вопросов. Индивидуальные задания выполняются студентами самостоятельно и оформляются в виде отчета.

## ***Порядок выполнения лабораторной работы***

Содержание отчета:

### **1. Титульный лист.**

<p>Министерство образования РФ Федеральное государственное бюджетное образовательное учреждение высшего образования Ивановский государственный политехнический университет Кафедра «Информационные системы и технологии»</p> <p><b>ОТЧЕТ ПО ЛАБОРАТОРНЫМ РАБОТАМ</b></p> <p><b>ТЕМА</b> «Разработка базы данных в среде Microsoft Access»</p> <p>Выполнил ст. гр. _____ Ф.И.О. _____ Проверил преп. каф. ИСиТ Ф.И.О. _____</p> <p>Иваново, 20__</p>
---

2. Постановка задачи (формулировка индивидуального задания).
3. Ход разработки базы данных (подробное описание действий, выполненных при разработке базы данных).
4. Заключение (основные итоги и выводы, предложения и рекомендации по использованию полученных результатов в практической деятельности).
5. Библиографический список.
6. Приложения (таблицы, схема данных, формы, запросы и отчеты, созданные при разработке базы данных).
7. Оглавление (автоматическое).

## Лабораторная работа № 1 «Создание таблиц»

Основным объектом базы данных является таблица, которая состоит из записей (строк) и полей (столбцов). На пересечении записи и поля образуется ячейка, в которой содержатся данные.

Каждому полю таблицы присваивается уникальное имя, которое не может содержать более 64 символов. В каждом поле содержатся данные одного типа.

### Создание базы данных

1. Запустите Microsoft Access 2007.



Новая база данных

2. Нажмите на кнопку  .

3. Задайте имя новой базы данных – «Записная книжка.accdb».

4. На вкладке ленты Создание в панели инструментов Таблицы нажмите на кнопку Конструктор таблиц.

5. Введите имена полей и укажите типы данных, к которым они относятся.

Имя поля	Тип данных
№ п/п	Счетчик
Фамилия	Текстовый
Имя	Текстовый
Адрес	Текстовый
Индекс	Числовой
Телефон	Текстовый
Хобби	Текстовый
Эл. почта	Гиперссылка

6. Выйдите из режима Конструктора, предварительно сохранив таблицу под именем «Друзья»; ключевые поля не задавайте.

7. Откройте таблицу «Друзья» двойным щелчком мыши и заполните в ней 10 строк.

8. Добавьте поля «Отчество» и «Дата рождения», для этого:

1) установите курсор на поле, перед которым нужно вставить новый столбец;

2) выполните команду: вкладка ленты Режим таблицы → панель инструментов Поля и столбцы → Вставить;

3) щелкнув два раза на Поле1, переименуйте его в «Отчество», а Поле2 – «Дата рождения».

9. Перейдите в режим Конструктора командой: вкладка ленты Главная → Режимы → Конструктор.

10. Для поля «Дата рождения» установите тип данных Дата / время; в свойствах поля выберите Краткий формат даты.

11. Отформатируйте таблицу следующим образом:

1) цвет сетки – темно-красный;

2) цвет фона – голубой;

3) цвет текста – темно-красный, размер – 12 пт, начертание – курсив.

12. Переименуйте поле «Хобби» в «Увлечения».

13. Удалите запись под номером 8.

14. Измените размер ячеек так, чтобы были видны все данные. Для этого достаточно два раза щелкнуть левой кнопкой мыши на границе полей.

15. Расположите поля в следующем порядке: «№», «Фамилия», «Имя», «Отчество», «Телефон», «Дата рождения», «Увлечения», «Адрес», «Индекс», «Фото», «Эл\_почта».

16. Заполните пустые ячейки таблицы.

17. В режиме Конструктора добавьте поле «Семейное положение», в котором будет содержаться фиксированный набор значений – замужем, не замужем, женат, не женат. Для создания раскрывающегося списка будем использовать Мастер подстановок:

1) установите тип данных Мастер подстановок;

2) в появившемся диалоговом окне выберите строку «Будет введен фиксированный набор значений» и нажмите кнопку Далее;

3) число столбцов – 1;

4) введите данные списка – замужем, не замужем, женат, не женат;

5) нажмите кнопку Готово.

18. С помощью раскрывающегося списка заполните новый столбец. Поскольку таблица получилась широкая, то при заполнении данного столбца возникают некоторые неудобства: не видно фамилии человека, для которого заполняется поле «Семейное положение». Чтобы фамилия была постоянно видна при заполнении таблицы, необходимо воспользоваться командой Закрепить столбцы из контекстного меню поля «Фамилия».

#### *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 переименовать поле?
9. Как можно создать поле с раскрывающимся списком?
10. С каким расширением сохраняется файл БД Access?

### **Лабораторная работа № 2 «Создание связей между таблицами»**

1. Запустите Microsoft Access 2007.

2. Создадим базу данных «Фирма». Сотрудники данной организации работают с клиентами и выполняют их заказы. Если все сведения поместить в одной таблице, то она станет очень неудобной для работы. В ней начнутся повторы данных. Всякий раз, когда сотрудник Иванов будет работать с какой-либо фирмой, придется прописывать данные о сотруднике и клиенте заново, в результате чего можно допустить множество ошибок. Чтобы уменьшить число ошибок, можно исходную таблицу разбить на несколько таблиц и установить связи между ними. Это будет более рационально, чем прежде.

Таким образом, необходимо создать 3 таблицы: Сотрудники, Клиенты и Заказы.

**Сотрудники**

<b>Имя поля</b>	<b>Тип данных</b>
Код сотрудника	Счетчик
Фамилия	Текстовый
Имя	Текстовый
Отчество	Текстовый
Должность	Текстовый
Телефон	Текстовый
Адрес	Текстовый
Дата рождения	Дата/Время
Заработная плата	Денежный
Фото	Объект OLE
Эл_почта	Гиперссылка

### Клиенты

Имя поля	Тип данных
Код клиента	Счетчик
Название компании	Текстовый
Адрес	Текстовый
Номер телефона	Текстовый
Факс	Числовой
Адрес электронной почты	Гиперссылка
Заметки	Поле МЕМО

### Заказы

Имя поля	Тип данных
Код заказа	Счетчик
Код клиента	Числовой
Код сотрудника	Числовой
Дата размещения	Дата/Время
Дата исполнения	Дата/Время
Сумма	Денежный
Отметка о выполнении	Логический

3. Отдельные таблицы, содержащие информацию по определенной теме, необходимо связать в единую структуру базы данных. Для связывания таблиц следует задать ключевые поля. Ключ состоит из одного или нескольких полей, значения которых однозначно определяют каждую запись в таблице. Наиболее подходящим в качестве ключевого поля является «Счетчик», так как значения в данном поле являются уникальными (т. е. исключают повторы).

4. Откройте таблицу Сотрудники в режиме Конструктора.

5. Нажмите правой кнопкой мыши на поле Код сотрудника и в появившемся контекстном меню выберите команду Ключевое поле. Если в таблице необходимо установить несколько ключевых полей, то выделить их можно, удерживая клавишу Ctrl.

6. Для таблицы Клиенты установите ключевое поле Код клиента, а для таблицы Заказы – Код заказа.

7. Таблица Заказы содержит поля Код сотрудника и Код клиента. При их заполнении могут возникнуть некоторые трудности, так как не всегда уда-

ется запомнить все предприятия, с которыми работает фирма, и всех сотрудников с номером кода. Для удобства можно создать раскрывающиеся списки с помощью Мастера подстановок.

8. Откройте таблицу Заказы в режиме Конструктора.

9. Для поля Код сотрудника выберите тип данных Мастер подстановок.

10. В появившемся окне выберите команду «Объект «столбец подстановки» будет использовать значения из таблицы или запроса» и щелкните на кнопке Далее.

11. В списке таблиц выберите таблицу Сотрудники и щелкните на кнопке Далее.

12. 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 поля Фамилия и Имя и щелкните на кнопке Далее.

13. Выберите порядок сортировки списка по полю Фамилия.

14. В следующем диалоговом окне задайте необходимую ширину столбцов раскрывающегося списка.

15. Установите флажок «Скрыть ключевой столбец» и нажмите кнопку Далее.

16. На последнем шаге Мастера подстановок замените при необходимости надпись для поля подстановок и щелкните на кнопке Готово.

17. Аналогичным образом создайте раскрывающийся список для поля Код клиента.

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

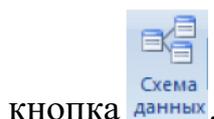
1) при отношении «один-к-одному» каждой записи ключевого поля в первой таблице соответствует только одна запись в связанном поле другой таблицы, и наоборот. Отношения такого типа используются не очень часто. Иногда их можно использовать для разделения таблиц, содержащих много полей, для отделения части таблицы по соображениям безопасности;

2) 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;

3) 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.

19. Закройте все открытые таблицы, так как создавать или изменять связи между открытыми таблицами нельзя.

20. Выполните команду: вкладка ленты Работа с базами данных →



21. Если ранее никаких связей между таблицами базы не было, то при открытии окна Схема данных одновременно открывается окно Добавление таблицы, в котором выберите таблицы Сотрудники, Клиенты и Заказы.

22. Если связи между таблицами уже были заданы, то для добавления в схему данных новой таблицы щелкните правой кнопкой мыши на схеме данных и в контекстном меню выберите пункт Добавить таблицу.

23. Установите связь между таблицами Сотрудники и Заказы, для этого выберите поле Код сотрудника в таблице Сотрудники и перенесите его на соответствующее поле в таблице Заказы.

24. После перетаскивания откроется диалоговое окно Изменение связей (рис. 1), в котором включите флажок Обеспечение условия целостности. Это позволит предотвратить случаи удаления записей из одной таблицы, при которых связанные с ними данные других таблиц останутся без связи.

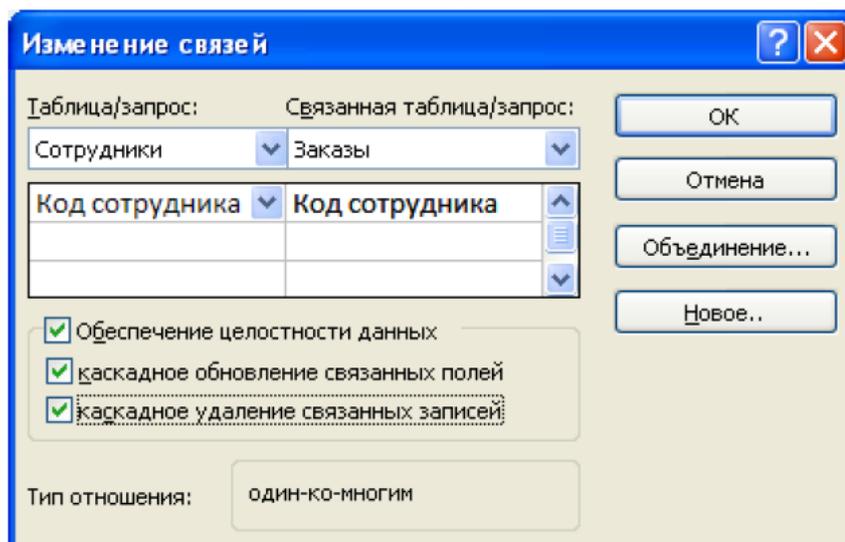


Рис. 1. Создание связи между таблицами

25. Флажки Каскадное обновление связанных полей и Каскадное удаление связанных записей обеспечивают одновременное обновление или удаление данных во всех подчиненных таблицах при их изменении в главной таблице.

26. Параметры связи можно изменить, нажав на кнопку Объединение.

27. После установления всех необходимых параметров нажмите кнопку ОК.

28. Связь между таблицами Клиенты и Заказы установите самостоятельно.

29. В результате должна получиться схема данных, представленная на рис. 2.

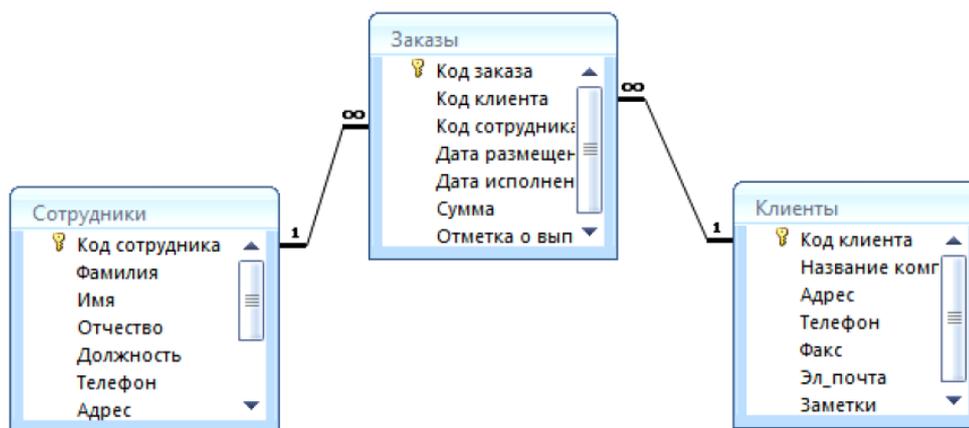


Рис. 2. Схема данных

В приведенном примере используются связи «один-ко-многим». На схеме данных они отображаются в виде соединительных линий со специальными значками около таблиц. Связь «один-ко-многим» помечается «1» вблизи главной таблицы (имеющей первичный ключ) и «∞» вблизи подчиненной таблицы (имеющей внешний ключ). Связь «один-к-одному» помечается двумя «1» (оба поля таблиц имеют первичные ключи). Неопределенная связь не имеет никаких знаков. Если установлено объединение, то его направление отмечается стрелкой на конце соединительной линии (ни одно из объединенных полей не является ключевым и не имеет уникального индекса).

30. В таблицу Сотрудники внесите данные о семи работниках.

31. В таблицу Клиенты внесите данные о десяти предприятиях, с которыми работает данная фирма.

32. В таблице Заказы оформите несколько заявок, поступивших на фирму.

#### Контрольные вопросы

1. С помощью чего можно создавать таблицы?
2. Что такое ключевое поле?
3. Как установить несколько ключевых полей?
4. Как установить связи между таблицами?
5. Какие существуют отношения между таблицами?
6. Что означают на схеме данных «1» и «∞»?
7. Зачем нужен Мастер подстановок?

### Лабораторная работа № 3 «Создание запросов»

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

Существуют различные виды запросов, но наиболее распространёнными являются запросы на выборку, с них и начнем наше знакомство.

1. Откройте базу данных «Фирма», созданную ранее.
2. Выполните команду: вкладка ленты Создание → Мастер запросов → Простой запрос.
3. В появившемся диалоговом окне (рис. 3) укажите таблицу Сотрудники и выберите поля Фамилия, Имя, Телефон. Нажмите кнопку Далее.

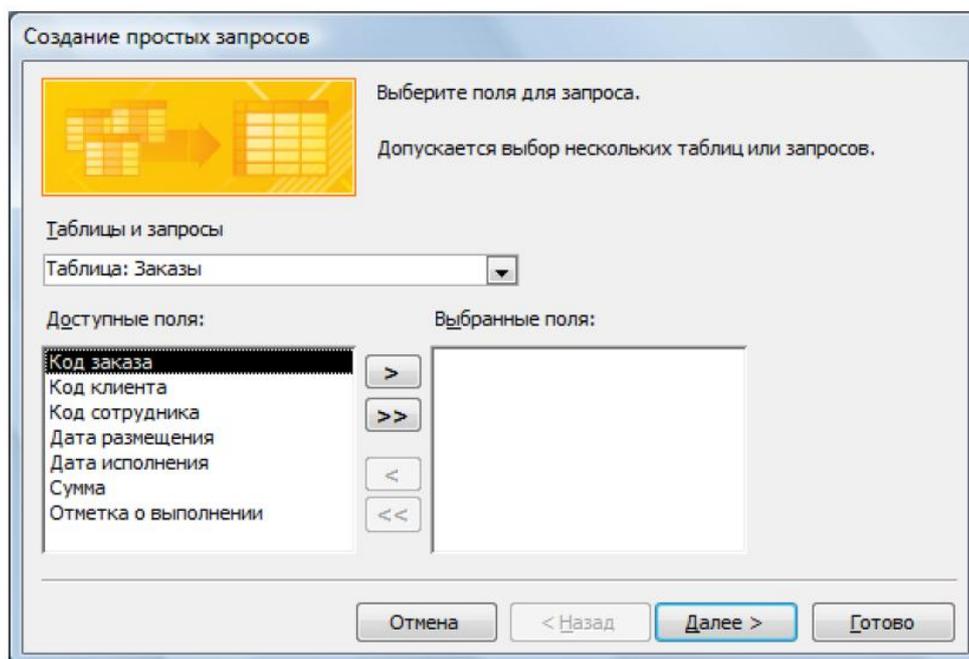


Рис. 3. Создание простого запроса

4. Введите имя запроса – Телефоны – и нажмите кнопку Готово. Перед вами появится запрос, в котором можно просмотреть телефоны сотрудников.
5. Следующий запрос попробуйте создать с помощью Конструктора, для этого выполните команду: вкладка ленты Создание → Конструктор запросов.
6. В диалоговом окне Добавление таблиц выберите таблицу Клиенты и щелкните на кнопке Добавить, а затем – на кнопке Закреть.
7. Чтобы перенести нужные поля в бланк запроса, необходимо по ним дважды щелкнуть левой кнопкой мыши (рис. 4).

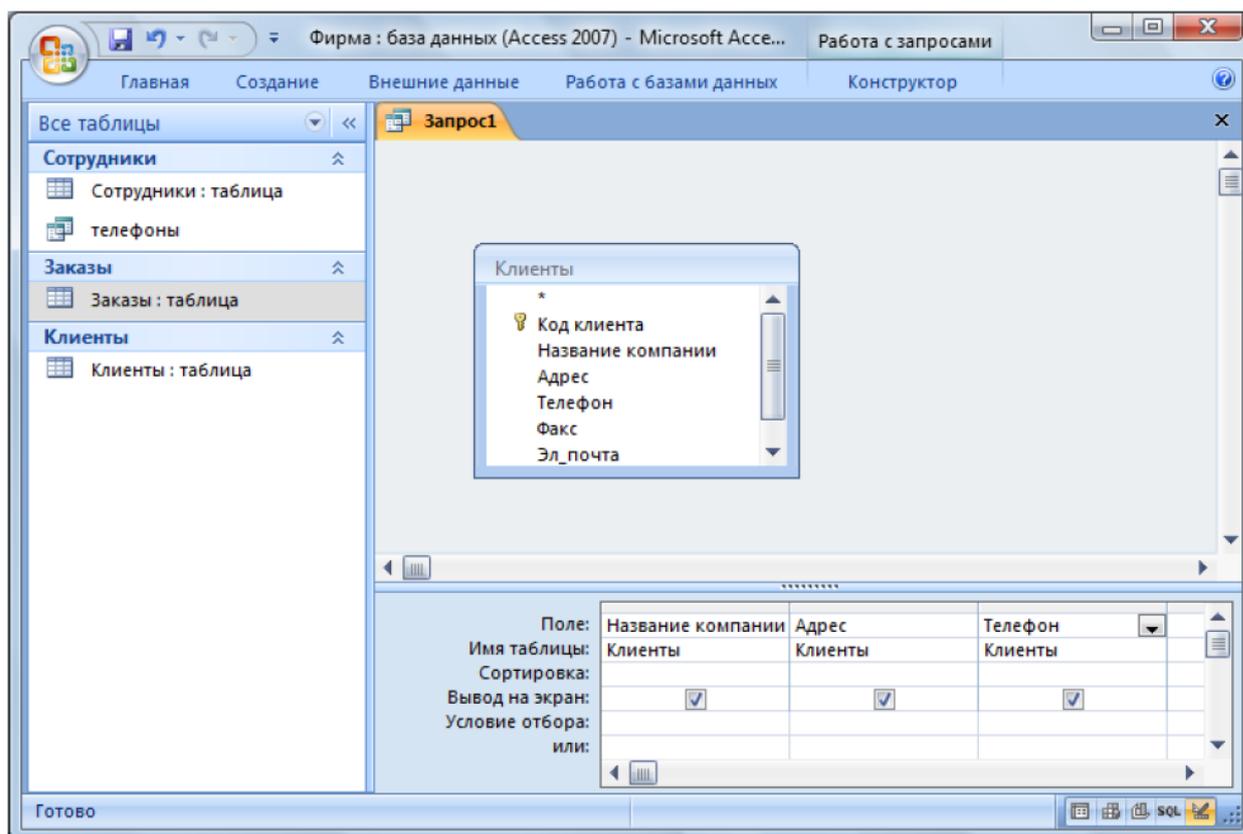


Рис. 4. Создание запроса в режиме Конструктора

8. Чтобы отсортировать записи в поле Название компании в алфавитном порядке, необходимо в раскрывающемся списке строки Сортировка выбрать пункт по возрастанию.

9. Сохраните запрос с именем «Адреса клиентов».

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

11. Допустим, мы хотим узнать, у кого из сотрудников день рождения в текущем месяце, например в апреле. Для этого откройте запрос в режиме Конструктора.

12. В строке Условие отбора для поля «Дата рождения» введите значение \*.04.\*. В данной записи \* означают, что дата и год рождения могут быть любыми, а месяц 4-м (т. е. апрель). После этого окно запроса должно выглядеть так, как оно представлено на рис. 5.

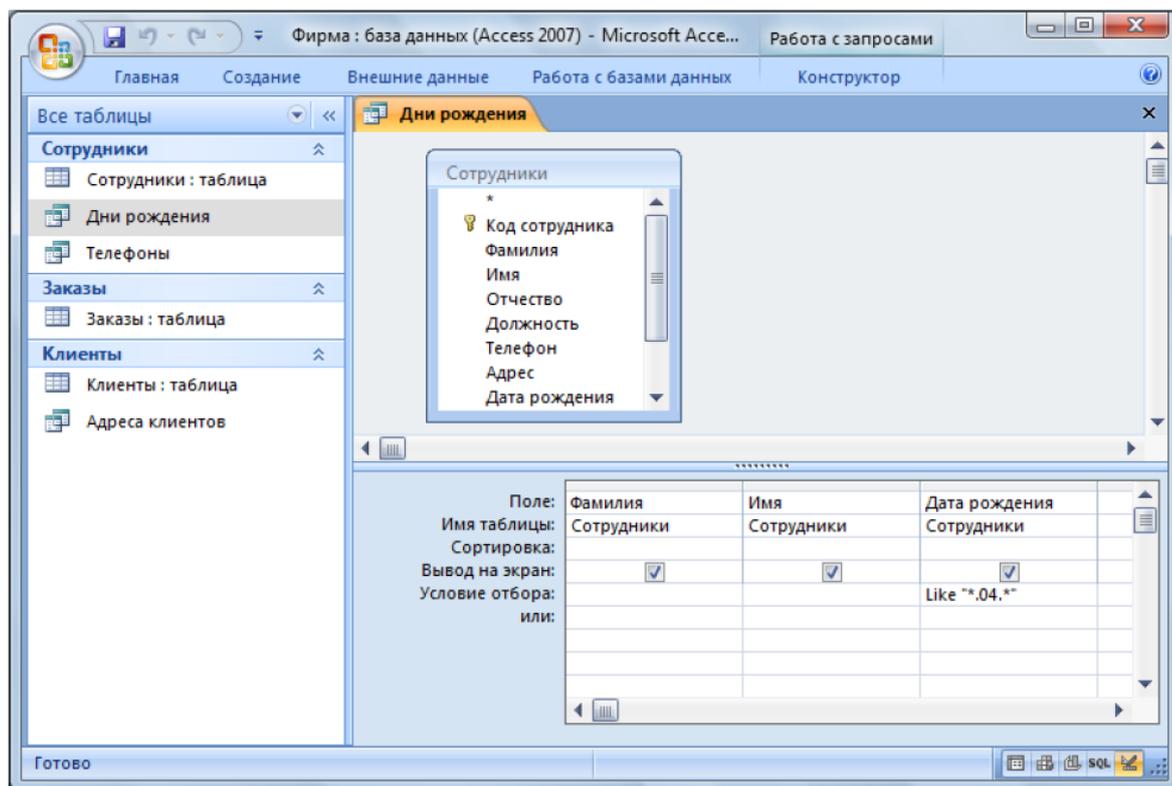


Рис. 5. Создание запроса

13. Закройте Конструктор и просмотрите полученный результат. Если в запросе Дни рождения нет ни одной записи, значит, в таблице Сотрудники нет ни одного человека, родившегося в апреле. Добавьте в таблицу Сотрудники несколько человек, родившихся в апреле, и посмотрите, как изменится запрос. Запросы автоматически обновляются при каждом открытии.

14. Если нам нужно узнать, кто из сотрудников родился в мае, то придется создать новый запрос или изменить условие в существующем запросе Дни рождения. Данная процедура является неудобной и занимает много времени. Если приходится часто выполнять запрос, но каждый раз с новыми значениями условий используют запрос с параметром. При запуске такого запроса на экран выводится диалоговое окно для ввода значения в качестве условия отбора. Чтобы создать запрос с параметром, пользователю необходимо ввести текст сообщения в строке Условие отбора бланка запроса (рис. 6).

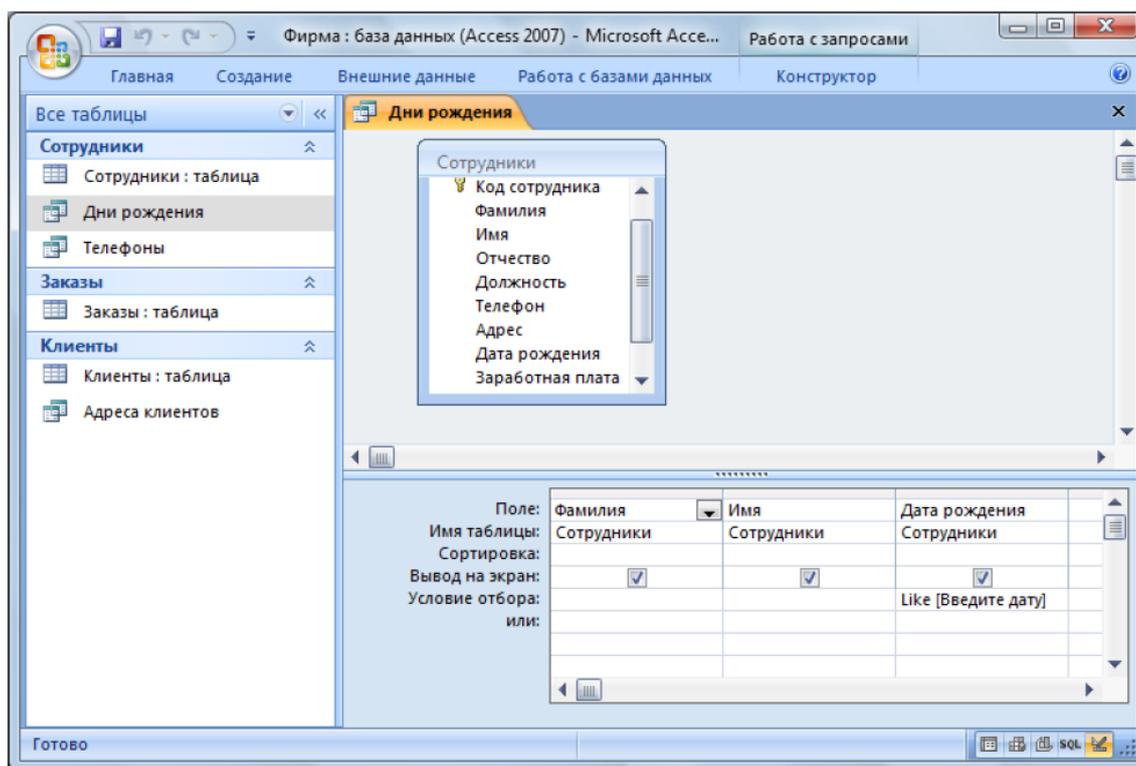


Рис. 6. Создание запроса с параметром

15. Запись Like[Введите дату] означает, что при открытии запроса появится диалоговое окно (рис. 7) с текстом «Введите дату» и полем для ввода условия отбора. Если ввести условие \*.04.\*, то в запросе появится список сотрудников, родившихся в апреле. Запустите запрос еще раз и введите значение \*.05.\*, посмотрите, как изменился запрос.

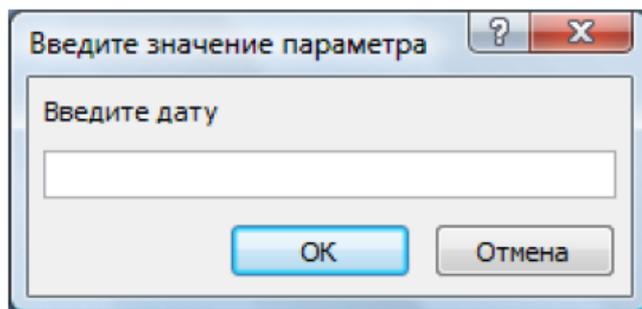


Рис. 7. Окно для ввода условия отбора

16. Измените запрос «Телефоны» так, чтобы при его запуске выводилось диалоговое окно с сообщением «Введите фамилию». Поскольку в запросе нужно вывести конкретную фамилию, в условии отбора слово Like писать не надо.

17. Измените запрос «Телефоны» так, чтобы при его запуске запрашивались не только фамилия, но и имя сотрудника.

18. Самостоятельно создайте запрос «Выполненные заказы», содержащий следующие сведения: фамилия и имя сотрудника, название компании, с

которой он работает, отметка о выполнении и сумма заказа. Данные запроса возьмите из нескольких таблиц.

19. В условии отбора для логического поля Отметка о выполнении введите Да, чтобы в запросе отображались только выполненные заказы.

20. Сделайте так, чтобы столбец Отметка о выполнении не выводился на экран.

21. Создайте запрос Сумма заказа, в котором будут отображаться заказы на сумму более 50 000 руб.

22. Измените запрос, чтобы сумма заказа была от 20 000 до 50 000 руб. Для данных запросов в условии отбора можно использовать операторы сравнения  $>$ ,  $<$ ,  $=$ ,  $>=$ ,  $<=$ ,  $< >$  и логические операторы And, Or, Not и др.

23. Иногда в запросах требуется произвести некоторые вычисления, например посчитать подоходный налог 13 % для каждой сделки. Для этого откройте запрос Сумма заказа в режиме Конструктора.

24. В пустом столбце бланка запроса щелкните правой кнопкой мыши на ячейке Поле и в появившемся контекстном меню выберите команду Построить. Перед вами появится окно Построитель выражений (рис. 8), который состоит из трех областей: поля выражения, кнопок операторов и элементов выражения. Сверху располагается поле выражения, в котором оно и создается. Вводимые в это поле элементы выбираются в двух других областях окна Построителя.

25. В левом списке откройте папку Запросы и выделите запрос Сумма заказа. В среднем списке выделите поле Сумма и нажмите кнопку Вставить. Идентификатор этого поля появится в поле выражения Построителя.

26. Щелкните на кнопке \* и введите 0,13 (см. рис. 8). Таким образом, мы посчитаем подоходный налог 13 %.

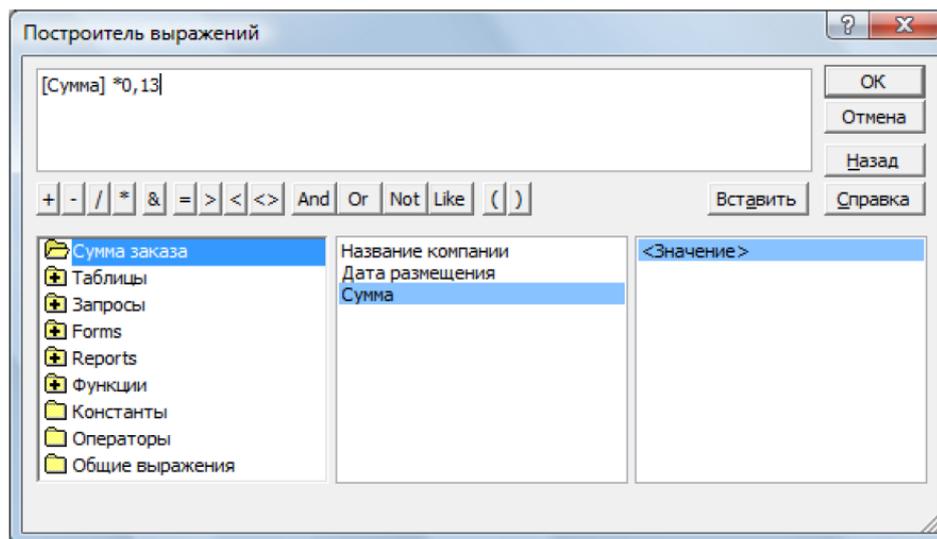


Рис. 8. Построитель выражений

27. Нажмите кнопку ОК, после чего в ячейке свойства Поле появится значение «Выражение1: [Сумма]\*0,13».

28. Замените Выражение1 на Налог и закройте Конструктор.

29. Откройте запрос и посмотрите, что у вас получилось.

30. Используя Построитель выражений, добавьте в запрос Сумма заказа поле Прибыль, в котором будет вычисляться доход от заказа (т. е. сумма минус налог).

31. Создайте запрос Менеджеры, с помощью которого в таблице Сотрудники найдите всех менеджеров фирмы.

#### *Контрольные вопросы*

1. Для чего предназначены запросы?
2. Какие виды запросов вы знаете?
3. С помощью чего можно создавать запросы?
4. Для чего используют запрос с параметром?
5. Как можно сделать вычисления в запросах?
6. Что означает запись в условии запроса «=50»?
7. Можно ли создавать запросы на основе нескольких таблиц?

### **Лабораторная работа № 4 «Создание и использование форм»**

**Формы** – это объекты базы данных, предназначенные для просмотра данных из таблиц и запросов, для ввода данных в базу, корректирования существующих данных и выполнения заданных действий. Форма может содержать графики, рисунки и другие внедренные объекты.

Можно вносить данные в таблицы и без помощи каких-либо форм. Но существует несколько причин, которые делают формы незаменимым средством ввода данных в базу:

- 1) при работе с формами ограничен доступ к таблицам (самому ценному в базе данных);
- 2) 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;
- 3) вводить данные в форму легче, чем в таблицу, и удобнее, так как в окне формы отображается, как правило, одна запись таблицы;
- 4) 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. Создавать формы можно на основе нескольких таблиц или запросов с помощью Мастера, используя средство автоформы, «вручную» в режиме Конструктора, сохраняя таблицу или запрос как форму. Созданную любым способом форму можно затем изменять в режиме Конструктора. Рассмотрим некоторые из перечисленных способов.

1. Выполните команду: вкладка ленты Создание → панель инструментов Формы → Другие формы → Мастер форм.

2. В диалоговом окне Создание форм выберите таблицы (запросы) и поля, которые будут помещены в форму. Щелкните по кнопке Далее.

3. В следующих диалоговых окнах мастера выберите внешний вид формы, стиль, задайте имя формы. Щелкните по кнопке Готово.

4. С помощью Мастера создайте формы Сотрудники, Клиенты, Заказы, Менеджеры.

5. Откройте форму Сотрудники в режиме Конструктора. Этот режим предназначен для создания и редактирования форм.

6. Разместите элементы в удобном для вас порядке, измените размер и цвет текста.

7. В заголовок формы добавьте текст Сотрудники фирмы.

8. В примечание формы добавьте объект Кнопка (вкладка ленты Конструктор → панель инструментов Элементы управления).

9. После того как вы «нарисуете» кнопку указателем, на экране появится диалоговое окно Создание кнопок (рис. 9).

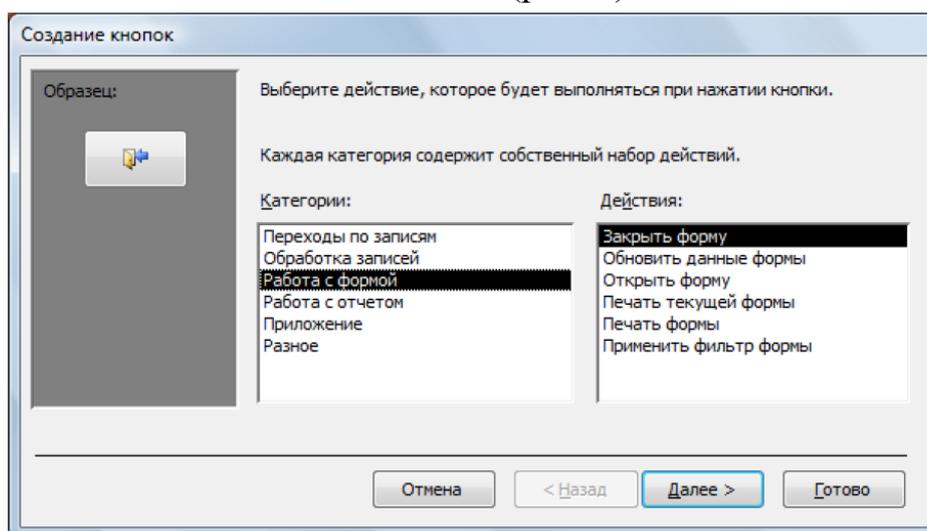


Рис. 9. Создание кнопок на форме

10. В категории Работа с формой выберите действие Закрывать форму и нажмите кнопку Далее.

11. Выберите рисунок или текст, который будет размещаться на кнопке.

12. В последнем диалоговом окне Мастера кнопок задайте имя кнопки и нажмите Готово.

13. Мастер кнопок написал для данной кнопки процедуру на языке Microsoft Visual Basic. Просмотреть процедуру обработки события можно с помощью команды Обработка событий контекстного меню кнопки.

14. Самостоятельно создайте кнопки Выход из приложения, Поиск записи, Удаление записи.

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

16. Создайте пустую форму.

17. 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.

18. Добавьте еще одну вкладку.

19. Переименуйте ярлычки вкладок так, чтобы на них отображались названия данных, которые будут в них располагаться: Сотрудники, Менеджеры, Помощь.

20. Перейдите на вкладку Сотрудники и перетащите на нее мышкой из базы данных форму Сотрудники.

21. Аналогичным образом поместите форму Менеджеры на вкладку Менеджеры.

22. На вкладку Помощь поместите некоторые советы по работе с базой данных.

23. Данную форму сохраните с именем Сотрудники фирмы.

24. 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.

25. Если кнопочной формы в базе данных нет, то будет выведен запрос на подтверждение ее создания. Нажмите Да в диалоговом окне подтверждения.

26. Перед вами появится Диспетчер кнопочных форм, в котором щелкните по кнопке Создать.

27. В диалоговом окне Создание (рис. 10) введите имя новой кнопочной формы и нажмите ОК.

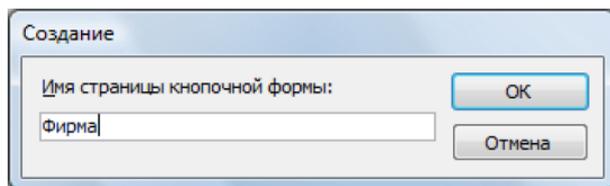


Рис. 10. Задание имени кнопочной формы

28. Имя новой кнопочной формы добавится в список Страницы кнопочной формы окна Диспетчер кнопочных форм (рис. 11). Выделите имя новой кнопочной формы и щелкните по кнопке Изменить.

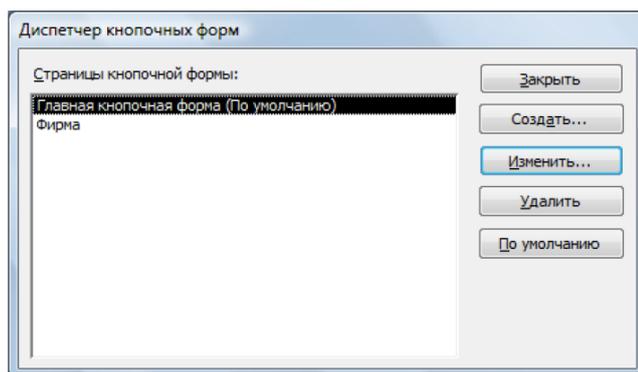


Рис. 11. Диспетчер кнопочных форм

29. В диалоговом окне Изменение страницы кнопочной формы щелкните по кнопке Создать. Появится диалоговое окно Изменение элемента кнопочной формы (рис. 12).

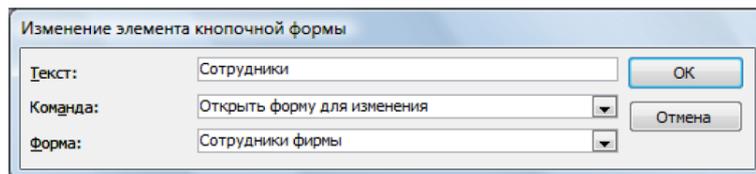


Рис. 12. Создание кнопок на форме

30. 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.

31. Аналогичным образом добавьте кнопки Клиенты, Заказы, Выход.

32. 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)».

33. Чтобы закончить создание кнопочной формы, щелкните по кнопке Закреть.

34. В результате должна получиться форма, представленная на рис. 13.

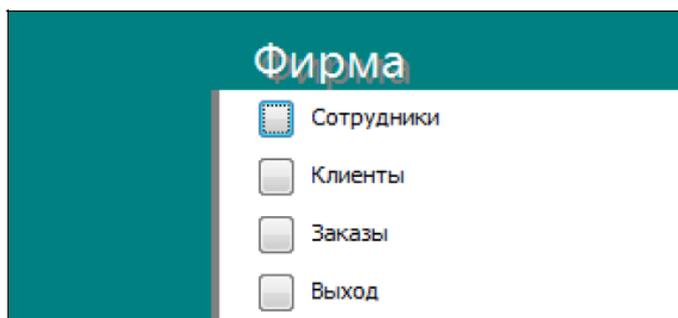


Рис. 13. Главная кнопочная форма

35. Добавьте в форму какой-нибудь рисунок.

36. Для того чтобы главная кнопочная форма появлялась на экране при запуске приложения, необходимо в главном меню нажать на кнопку Параметры Access (рис. 14). Для текущей базы данных установите форму просмотра – «кнопочная форма».

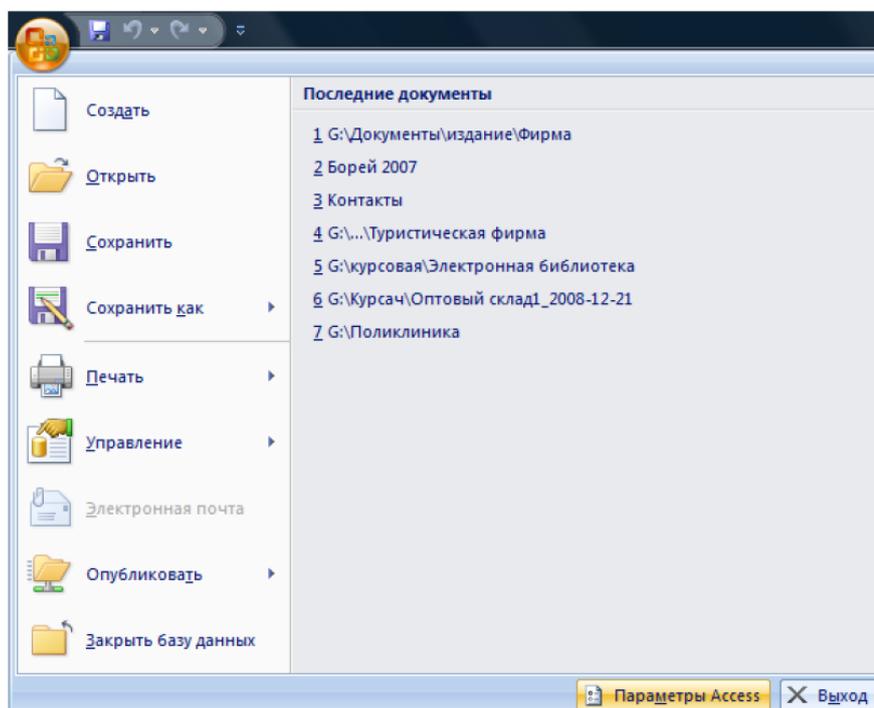


Рис. 14. Задание параметров Access

### *Контрольные вопросы*

1. Для чего предназначены формы?
2. Почему форма является незаменимым средством в БД?
3. С помощью чего можно создавать формы?
4. На основе чего можно создавать формы?
5. Как создать кнопку на форме?
6. Как можно разместить несколько таблиц и запросов на одной форме?
7. Как создать главную кнопочную форму?

## Лабораторная работа № 5 «Создание отчетов»

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 несколькими способами:

- 1) с помощью Мастера отчетов;
- 2) на основе таблиц или запросов;
- 3) в режиме Конструктора.

1. В окне базы данных выполните команду: вкладка ленты Создание → панель инструментов Отчеты → Мастер отчетов.

2. Выберите из списка таблиц (или запрос), которая будет использована как источник данных (например, запрос Адреса клиентов).

3. В появившемся диалоговом окне Создание отчетов (рис. 15) переместите все доступные поля в область «выбранные поля».

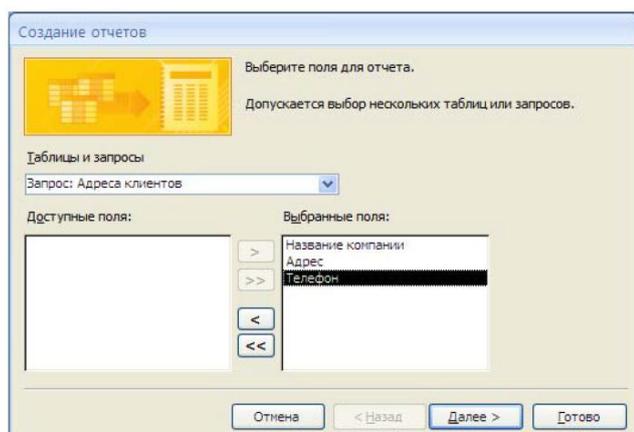


Рис. 15. Мастер отчетов

4. С помощью Мастера отчетов создайте отчет Дни рождения. В качестве источника данных используйте таблицу Сотрудники.

5. Если требуется напечатать почтовые наклейки, Access предоставляет такую возможность. Для этого выделите таблицу Клиенты и выполните команду: вкладка ленты Создание → панель инструментов Отчеты → Наклейки.

6. В появившемся диалоговом окне (рис. 16) укажите размер наклейки, систему единиц, тип наклейки и нажмите кнопку Далее.

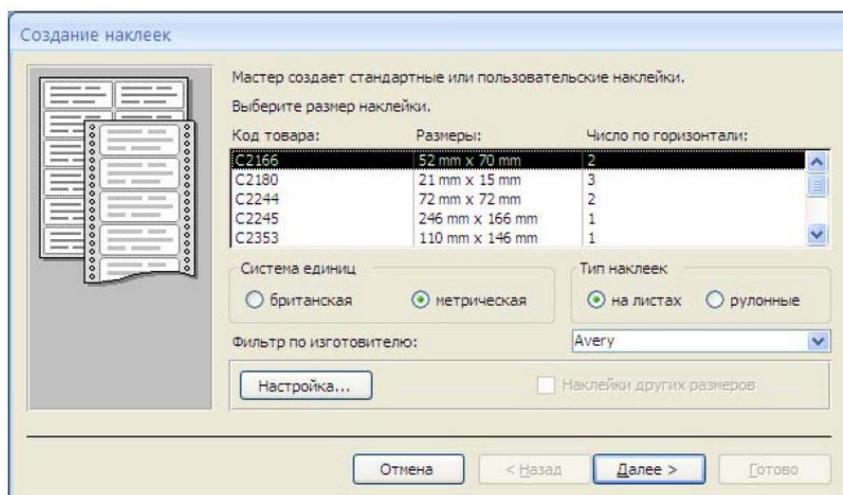


Рис. 16. Диалоговое окно Создание наклеек

7. На следующем шаге создания отчета установите шрифт, размер, цвет текста и начертание. Нажмите кнопку Далее.

8. Выберите поля, которые будут размещаться на наклейке. Например, Название компании, Адрес, Телефон и Факс. Если на каждой наклейке требуется вывести определенный текст, то введите его в прототип наклейки.

9. При необходимости измените название отчета с наклейками и нажмите кнопку Готово.

10. Иногда в отчетах требуется вычислять итоговые значения, среднее, минимальное или максимальное значения, а также проценты. Для этого запустите Мастер отчетов и в качестве источника данных укажите запрос Сумма заказа.

11. В диалоговом окне Мастера, в котором задается порядок сортировки записей, нажмите кнопку Итоги (рис. 17).

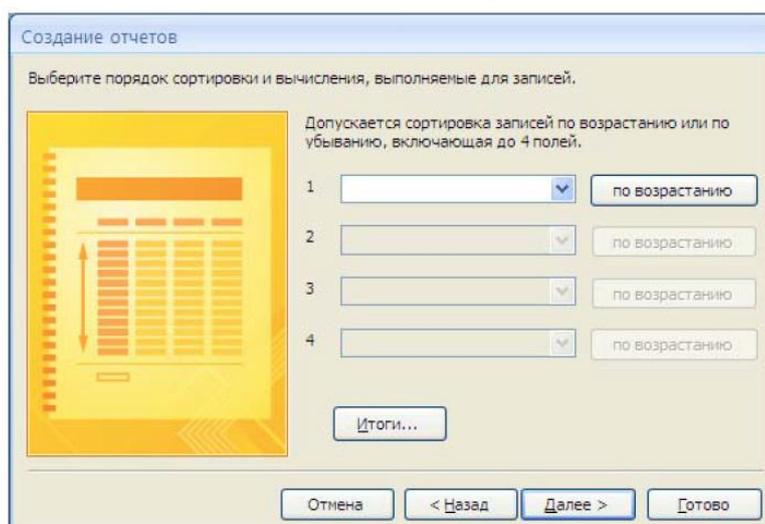


Рис. 17. Вычисление итоговых значений в отчетах

12. В диалоговом окне Итоги (рис. 18) для полей Сумма и Налог установите флажки в столбце sum, чтобы посчитать итоговую сумму.

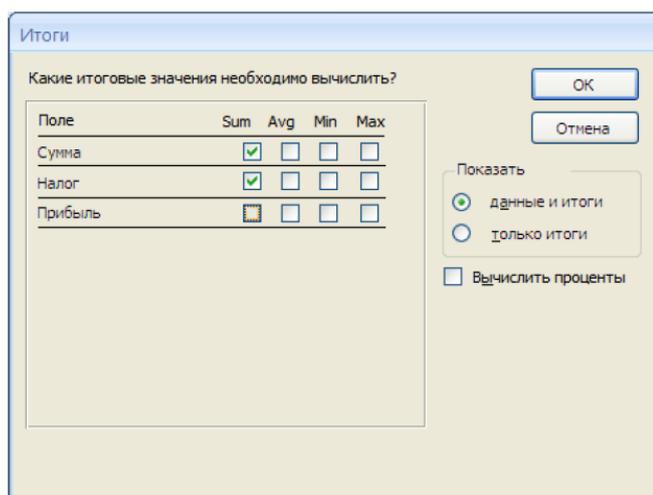


Рис. 18. Вычисление итоговых значений суммы

13. Далее выполните все шаги Мастера и нажмите кнопку Готово.

14. Создайте отчет Дни рождения, используя в качестве источника данных запрос Дни рождения.

15. Составьте отчет Выполненные заказы, в котором будут данные о компании и сумме заказа. Вычислите итоговую сумму, среднее значение (Avg) и максимальную сумму для каждой фирмы.

#### *Контрольные вопросы*

1. Для чего предназначены отчеты?
2. Какие способы создания отчетов вы знаете?
3. Как в отчетах можно посчитать итоговые значения?
4. Какие итоговые значения можно посчитать в отчетах?
5. Как в Access напечатать почтовые наклейки?

## **Варианты заданий**

### **Вариант 1**

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

Книги – шифр книги (ключевое поле), автор, название, год издания, количество экземпляров.

Читатели – читательский билет (ключевое поле), фамилия, имя, отчество, адрес.

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

2. Установите связи между таблицами.
3. С помощью запроса отберите все книги, выпущенные с 1990 по 2007 годы.

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

5. Создайте формы для ввода данных, отчеты и главную кнопочную форму.

### **Вариант 2**

1. Разработайте базу данных «Продуктовый магазин», которая состоит из четырех таблиц со следующей структурой:

Товары – код товара (ключевое поле), наименование товара, количество товара.

Поступление товаров – код товара, дата поступления, цена приобретения товара за единицу, код поставщика.

Продажа товаров – код товара, месяц продажи, проданное количество за месяц, цена продажи товара.

Поставщики – код поставщика (ключевое поле), название поставщика, адрес поставщика, телефон поставщика.

2. Установите связи между таблицами.

3. С помощью запроса отберите товары, цены которых от 100 до 450 руб.

4. Создайте запрос с параметром для отбора товаров, проданных в определенном месяце.

5. Создайте формы для ввода данных, отчеты и главную кнопочную форму.

### **Вариант 3**

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

Студенты – шифр студента (ключевое поле), фамилия, имя, отчество, курс, группа.

Экзамены – шифр студента, дата, шифр дисциплины, оценка.

Зачеты – шифр студента, дата, шифр дисциплины, зачет.

Дисциплины – шифр дисциплины (ключевое поле), название дисциплины, количество часов.

2. Установите связи между таблицами.

3. С помощью запроса отберите студентов, сдавших экзамен на 4 или 5.

4. Создайте запрос с параметром для отбора студентов, получивших или не получивших зачет.

5. Создайте формы для ввода данных, отчеты и главную кнопочную форму.

### **Вариант 4**

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

Склад – код товара, количество, дата поступления.

Товары – код товара (ключевое поле), название товара, срок хранения.

Заявки – код заявки (ключевое поле), название организации, код товара, требуемое количество.

Отпуск товаров – код заявки (ключевое поле), код товара, отпущенное количество, дата отпуска товара.

2. Установите связи между таблицами.

3. С помощью запроса отберите товары, количество которых от 50 до 200 штук.

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

5. Создайте формы для ввода данных, отчеты и главную кнопочную форму.

### **Вариант 5**

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

Анкета – номер абитуриента (ключевое поле), фамилия, имя, отчество, дата рождения, оконченное среднее учебное заведение (название, номер, населенный пункт), дата окончания учебного заведения, наличие красного диплома или золотой / серебряной медали, адрес, телефон, шифр специальности.

Специальности – шифр специальности (ключевое поле), название специальности.

Дисциплины – шифр дисциплины (ключевое поле), название дисциплины.

Вступительные экзамены – номер абитуриента, шифр дисциплины, экзаменационная оценка.

2. Установите связи между таблицами.

3. Составьте запрос для отбора студентов, сдавших экзамены без троек.

4. Создайте запрос с параметром для отбора студентов, поступающих на определенную специальность.

5. Создайте формы для ввода данных, отчеты и главную кнопочную форму.

### **Вариант 6**

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

Транспорт – марка автомобиля, государственный номер (ключевое поле), расход топлива.

Заявки – код заявки (ключевое поле), дата заявки, название груза, количество груза, пункт отправления, пункт назначения.

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

2. Установите связи между таблицами.

3. С помощью запроса отберите заявки с количеством груза от 100 до 500 кг.

4. Создайте запрос с параметром для отбора транспорта по марке автомобиля.

5. Создайте формы для ввода данных, отчеты и главную кнопочную форму.

### **Вариант № 7**

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

Клиенты – код клиента (ключевое поле), фамилия, имя, отчество, телефон, адрес, паспортные данные, залог.

Склад – код оборудования (ключевое поле), название, количество, залоговая стоимость, остаток.

Прокат – № п/п, клиент, оборудование, дата выдачи, срок возврата, отметка о возврате, оплата проката.

2. Установите связи между таблицами.

3. Создайте запрос для отбора оборудования с залоговой стоимостью от 10000 до 50000 руб.

4. Создайте запрос с параметром для отбора клиентов, возвративших оборудование.

5. Создайте формы для ввода данных, отчеты и главную кнопочную форму.

### **Вариант 8**

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

Клиенты – код клиента (ключевое поле), фамилия, имя, отчество, паспорт, телефон, адрес, заработная плата.

Виды кредитов – код кредита (ключевое поле), название кредита, процентная ставка, условия предоставления.

Предоставленные кредиты – № п/п, клиент, кредит, дата предоставления, срок, дата возврата, сумма, отметка о возврате.

2. Установите связи между таблицами.

3. Создайте запрос для отбора клиентов, взявших кредит от 500 000 до 1 000 000 руб.

4. Создайте запрос с параметром для отбора кредитов по процентной ставке.

5. Создайте формы для ввода данных, отчеты и главную кнопочную форму.

### **Вариант 9**

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

Клиенты – код клиента (ключевое поле), фамилия, имя, отчество, телефон, адрес, паспорт.

Сотрудники – код сотрудника (ключевое поле), фамилия, имя, отчество, должность, телефон, адрес, паспортные данные.

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

«Заказы» – код заказа (ключевое поле), клиент, маршрут, сотрудник (менеджер, оформивший заказ), дата, отметка об оплате.

2. Установите связи между таблицами.

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

4. Создайте запрос с параметром для отбора клиентов, выбравших определенный вид маршрута.

5. Создайте формы для ввода данных, отчеты и главную кнопочную форму.

### **Вариант 10**

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

Врачи – код врача (ключевое поле), ФИО, должность, специализация, стаж работы, адрес, телефон.

Болезни – № п/п (ключевое поле), название заболевания, рекомендации по лечению, меры профилактики.

Пациенты – код пациента (ключевое поле), ФИО, адрес, телефон, страховой полис, паспорт.

Диагноз – № п/п (ключевое поле), пациент, заболевание, лечащий врач, дата обращения, дата выздоровления.

2. Установите связи между таблицами.

3. С помощью запроса отберите врачей-стоматологов и ортопедов.

4. Создайте запрос с параметром для отбора пациентов с определенным видом заболевания.

5. Создайте формы для ввода данных, отчеты и главную кнопочную форму.

### **Библиографический список**

1. Кириллов, В.В. Введение в реляционные базы данных / В.В. Кириллов, Г.Ю. Громов. - СПб.: БХВ-Петербург, 2012. - 464 с.

2. Кошелев, В.Е. Базы данных в ACCESS 2007: Эффективное использование / В.Е. Кошелев. - М.: Бином-Пресс, 2009. - 592 с.

3. Агальцов, В.П. Базы данных. В 2-х т. Т. 2. Распределенные и удаленные базы данных: Учебник / В.П. Агальцов. - М.: ИД ФОРУМ, НИЦ ИНФРА-М, 2013. - 272 с.

4. Агальцов, В.П. Базы данных. В 2-х т. Т. 1. Локальные базы данных: Учебник / В.П. Агальцов. - М.: ИД ФОРУМ, НИЦ ИНФРА-М, 2013. - 352 с.

5. Голицына, О.Л. Базы данных: Учебное пособие / О.Л. Голицына, Н.В. Максимов, И.И. Попов. - М.: Форум, 2012. - 400 с.

6. Карпова, И.П. Базы данных: Учебное пособие / И.П. Карпова. - СПб.: Питер, 2013. - 240 с.