

Федеральное агентство по образованию
АМУРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ГОУВПО «АмГУ»

УТВЕРЖДАЮ

Зав. кафедрой КиТО

_____ И.В. Абакумова

« ____ » _____ 2007 г.

ПРИКЛАДНАЯ ИНФОРМАТИКА

УЧЕБНО-МЕТОДИЧЕСКИЙ КОМПЛЕКС ПО ДИСЦИПЛИНЕ

для специальности 260902 – «Конструирование швейных изделий»

Составитель: Т.А. Тибенко

Благовещенск

2007 г.

Печатается по решению
редакционно-издательского совета
факультета прикладных искусств
Амурского государственного
университета

Т.А. Тибенко

Прикладная информатика: Учебно-методический комплекс по дисциплине для специальности 260902 – «Конструирование швейных изделий» – Благовещенск: Амурский гос. ун-т, 2007. – с.

Учебно-методическое пособие предназначено для студентов очной формы обучения специальности 260902 – «Конструирование швейных изделий». Составлено в соответствии с требованиями Государственного образовательного стандарта высшего профессионального образования для специальности 260902 и включает наименование тем лабораторных занятий; методические рекомендации по проведению лабораторных работ; темы для самостоятельной работы; вопросы для итоговой оценки знаний; список рекомендуемой литературы.

ВВЕДЕНИЕ

Целью преподавания дисциплины «Прикладная информатика» является приобретение студентами навыков решения специальных задач швейного производства с использованием ЭВМ.

В любой сфере деятельности часто приходится иметь дело с большими объемами данных. Основными операциями при этом являются сбор информации, ее обработка (поиск требуемых данных, сортировка, фильтрация и т.д.), создание форм, отчетов для просмотра и распечатки данных. Одним из самых популярных программных продуктов, обеспечивающих все эти функции, признана система управления базами данных Microsoft Access. В связи с этим, основными задачами дисциплины является освещение принципов и методов создания баз данных, создаваемых в СУБД Microsoft Access.

В результате изучения дисциплины студент должен уметь профессионально создавать и редактировать базы данных, осуществлять к ним запросы, формировать отчеты и т.д.

Данный курс направлен на закрепление полученных ранее знаний в области работы с компьютером, а также их углубления и расширения в решении специальных практических задач швейного производства.

ПРОГРАММА КУРСА

1.1 Содержание дисциплины

Программа курса «Прикладная информатика» составлена в соответствии с требованиями государственного образовательного стандарта высшего профессионального образования.

1.1.1 Цель изучения дисциплины

Приобретение навыков решения специальных задач легкой промышленности с использованием ЭВМ.

1.1.2 Основная задача курса

Дать студентам общее представление о принципах и методах анализа данных посредством электронных таблиц, создаваемыми в среде табличного процессора Microsoft Excel. Сформировать навыки работы с инструментами доступными в среде табличного процессора Microsoft Excel. Дать студентам общее представление о принципах и методах создания баз данных. Сформировать навыки работы с инструментами доступными в среде СУБД Microsoft Access.

1.1.3 Место курса в профессиональной подготовке выпускника

Принцип построения курса «Информационные технологии в легкой промышленности» состоит в преемственности этого курса по отношению к курсам «Высшая математика», «Информатика», «Английский язык», «Прикладная информатика». Данный курс направлен на закрепление полученных ранее знаний в области работы с компьютером, а также их углубления и расширения в решении специальных практических задач легкой промышленности.

1.1.4 Требования к уровню освоения дисциплины

Основные навыки, приобретаемые студентами при изучении дисциплины: проведение различных вычислений с использованием мощного аппарата функций и формул; исследование влияния различных факторов на данные; решение задач оптимизации; получение выборки данных, удовлетворяющих определенным критериям; построение графиков и диаграмм; статистический анализ данных; создания структуры однотабличной БД; разработки схемы данных и создание структуры реляционной БД; ввода и редактирования данных; разработки пользовательских форм ввода данных в однотабличную и реляционную БД; формирования запросов для поиска и обработки данных; создания отчетов для вывода данных.

1.2 Тематический план лабораторных занятий

Тема занятия, содержание	Объем в часах
1	2
1. <i>Работа с электронной таблицей MS Excel</i> Общие сведения. Запуск MS Excel. Экран MS Excel. Основные элементы таблицы. Ввод данных в электронную таблицу. Основные действия с рабочей книгой.	4
2. <i>Редактирование рабочей книги</i> Шрифты. Выравнивание. Выбор цветов шрифта и фона. Обрамление. Изменение ширины столбцов и высоты строк. Предварительный просмотр. Редактирование содержимого ячейки. Операции со строками, столбцами, блоками. Автоматизация ввода. Работа с листами.	4
3. <i>Вычисления в электронных таблицах</i> Ссылки на ячейки. Копирование формул. Абсолютная и относительная ссылка.	4
4. <i>Применение стандартных функций</i> Понятие формулы. Понятие функции в Excel. Правила синтаксиса при записи функций. Значения ошибок в формулах.	4
5. <i>Диаграммы и графики</i> Создание диаграммы с помощью Мастера диаграмм. Перемещение и изменение размеров диаграммы. Редактирование диаграммы.	4
6. <i>Работа с базой данных в MS Excel</i> Структура базы данных. Создание сводной таблицы.	4
7. <i>Анализ деловых данных</i> Подбор параметра. Поиск решения. Создание сценариев.	4
8. <i>Алгебра матриц с использованием MS Excel</i> Матричная алгебра. Сложение (вычитание) матриц. Транспонирование матрицы. Вычисление матричного выражения.	2
9. <i>Работа с массивами</i>	2
Итого	34

1.3 План самостоятельной работы

Знакомство с научной и технической литературой по исследованию технологических процессов в текстильной и легкой промышленности.

Знакомство с периодическими изданиями по исследованию технологических процессов в текстильной и легкой промышленности.

1.4 Перечень форм контроля знаний

Промежуточный контроль знаний осуществляется при выполнении и сдаче каждого задания лабораторной работы.

В качестве заключительного контроля знаний студентов служит зачет по дисциплине, на который предоставляются все лабораторные работы, оформленные в тетради, а также в электронном виде. Студент должен уметь объяснить, какими средствами он пользовался для выполнения того или иного задания.

2 МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ВЫПОЛНЕНИЮ ЛАБОРАТОРНЫХ РАБОТ

2.1 Лабораторная работа № 1

Тема: Основные принципы работы в MS Access

Содержание:

- 2.1.1 Общая характеристика MS Access
- 2.1.2 Пользовательский интерфейс
- 2.1.3 Объекты СУБД MS Access
- 2.1.4 Типы данных
- 2.1.5 Построитель выражений
- 2.1.6 Задания к лабораторной работе
- 2.1.7 Контрольные вопросы

2.1.1 Общая характеристика MS Access

MS Access – это система управления базами данных (СУБД). Под системой управления понимается комплекс программ, который позволяет не только хранить большие массивы данных в определенном формате, но и обрабатывать их, представляя в удобном для пользователей виде. **MS Access** дает возможность также автоматизировать часто выполняемые операции (например, расчет заработной платы, учет материальных ценностей и т.п.). С помощью **MS Access** можно не только разрабатывать удобные формы ввода и просмотра данных, но и составлять сложные отчеты.

Поскольку и **Windows** и **MS Access** разработаны одной фирмой, они очень хорошо взаимодействуют друг с другом. СУБД **MS Access** работает под управлением **Windows**; таким образом, все преимущества **Windows** доступны в **MS Access**/

MS Access – это реляционная СУБД. С ее помощью можно работать одновременно с несколькими таблицами базы данных. Применение реляционной СУБД помогает упростить структуру данных и таким образом облегчить выполнение работы. Таблицу **MS Access** можно связать с данными, хранящимися на другом компьютере или на сервере, а также использовать таблицу, созданную в СУБД **Paradox** или **Dbase**. Данные **MS Access** очень просто комбинировать с данными **Excel**.

В СУБД **MS Access** предусмотрено много дополнительных сервисных возможностей. **Мастера** помогут создать таблицы, формы или отчеты из имеющихся заготовок, сделав всю черновую работу. Построитель выражений – выражения используются в **MS Access**, например, для проверки допустимости введенного значения. **Макросы** позволяют автоматизировать многие про-

цессы без программирования, тогда как встроенный в MS Access язык VBA (Visual Basic for Applications) – специально разработанный компанией Windows диалект языка Basic для использования в приложениях Windows Office – дает возможность опытному пользователю программировать сложные процедуры обработки данных. Просматривая свою форму или отчет, можно представить, как они будут выглядеть в распечатанном виде. И наконец, используя такие возможности языка программирования C, как функции и обращения к Windows API (Application Programming Interface – интерфейс прикладных программ Windows), можно написать подпрограмму для взаимодействия MS Access с другими приложениями – источниками данных.

Система MS Access содержит набор инструментов для управления базами данных, включающий конструкторы таблиц, форм, запросов и отчетов. Кроме того, MS Access можно рассматривать и как среду для разработки приложений. Используя макросы для автоматизации задач, можно создавать такие же мощные, ориентированные на пользователя приложения, как и приложения, созданные с помощью «полноценных» языков программирования, дополнять их кнопками, меню и диалоговыми окнами. Программируя на VBA, можно создавать программы, по мощности не уступающие самой MS Access. Более того, многие средства MS Access, например *мастера* и *конструкторы*, написаны на VBA. Мощность и гибкость системы MS Access делают ее сегодня одной из лучших программ для управления базами данных.

2.1.2 Пользовательский интерфейс

После запуска MS Access вы увидите пустое окно с панелью задач (это вертикальная полоса у правого края окна). На панели задач отображаются наиболее часто выполняемые операции. В данный момент на ней присутствуют команды открытия файла базы данных. Первый файл – это последний открывавшийся файл (максимальное количество таких файлов от 4 до 9). Следующая строка, содержащая текст **Дополнительно...** позволяет открыть стандартный диалог открытия файла. При помощи этого диалога можно открыть файл базы данных, находящийся в любой доступной папке. И последняя строка **Создать файл...** позволяет создать новую базу данных.

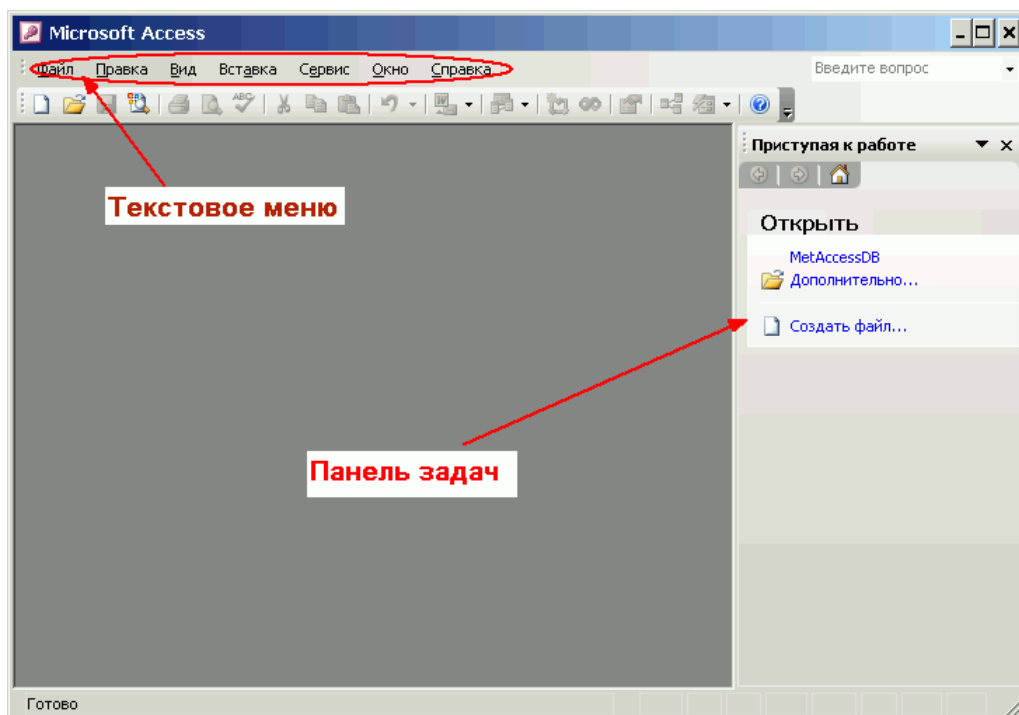


Рисунок 2.1.1 – Интерфейс программы MS Access

После выбора на панели задач пункта **Создать файл...**, содержимое панели задач меняется. На ней появляется список возможных способов создания базы данных (рис. 2.1.2). Существует шесть способов создания новой базы данных.

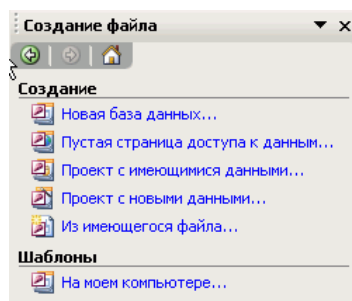


Рисунок 2.1.2 – Создание файла

Новая база данных. После выбора данного пункта появляется окно для задания расположения файла базы данных и имени этого файла. И после нажатия кнопки **Создать** появляется окно пустой базы данных.

Пустая страница доступа к данным. Страницы доступа к данным представляют специальный тип **web**-страниц, предназначенный для просмотра и работы через **Internet** или интрасеть с данными, хранящимися в базах данных **MS Access** или в базах данных **Microsoft SQL Server**. Страница доступа к данным может также включать данные из других источников, таких как **Microsoft Excel**. Для того чтобы иметь возможность создать страницы доступа к данным необходимо установить **web**-компоненты пакета **MS Office**.

Проект с имеющимися данными. Данный пункт позволяет создать новый проект MS Access, для доступа к данным SQL Server. То есть таким образом можно создать клиентское приложение.

Проекты создаются в основном для работы с автономными данными, то есть по определенным запросам данные получаются от сервера, а затем их можно обрабатывать локально. Такие приложения очень удобны для больших предприятий, использующих MS SQL Server. Например, если для какого-либо подразделения имеющего собственную базу данных, необходима часть информации с основного сервера, то при помощи проекта MS Access можно эти данные получить и добавить в локальную базу данных.

Проект с новыми данными. Эта команда похожа на уже описанную с той лишь разницей, что в предыдущем случае проект создавался для доступа к уже существующей базе данных сервера, а в данном случае будет создана новая база данных MS SQL Server и проект MS Access для работы с ней.

Из имеющегося файла. При использовании этой команды новая база данных будет создана на основе имеющейся, то есть новый файл базы данных, являясь полной копией оригинала, будет иметь другое имя.

Шаблоны: на моем компьютере. Последний пункт позволяет создать базу данных при помощи мастеров на основании заранее определенных шаблонов. Причем в этом случае создается полностью работоспособная база данных вместе с таблицами, запросами, формами и отчетами в несколько щелчков. Затем созданную базу данных можно доработать по своему усмотрению.

2.1.3 Объекты СУБД Access

Перед созданием базы данных должна быть определена логическая структура базы – состав таблиц, их структура и межтабличные связи.

Объектами БД помимо таблиц являются также запросы, формы, отчеты, макросы и модули, создание которых существенно упрощает работу пользователя с массивами данных.

Таблицы. Основная информация хранится в таблицах. Таблица – совокупность записей. В БД столбцы называют полями, а строки – записями. Поля образуют структуру БД, записи составляют информацию, которая в ней содержится. Количество записей в таблице ограничивается емкостью жесткого диска. Допустимое количество полей – 255. БД MS Access может содержать до 32768 объектов, причем одновременно могут быть открыты 1024 таблицы (если позволяют системные ресурсы).

Запросы. Это инструмент для анализа выбора и изменения данных, они обеспечивают быстрый и эффективный доступ к данным из одной или нескольких связанных таблиц. Результатом выполнения запроса является та-

блица, которая может быть использована наряду с другими таблицами базы данных при обработке данных. Запросы позволяют исключить несанкционированный доступ к конфиденциальной информации, содержащейся в основных таблицах. При работе с запросами данные можно упорядочивать, фильтровать, объединять и производить с ними необходимые итоговые вычисления. Запрос может формироваться с помощью QBE-запросов по образцу или с помощью инструкции SQL – языка структурированных запросов. Запросы т.ж. используются в качестве источника данных для форм и отчетов.

Формы. Форма представляет собой специальный формат экрана, служит для организации внесения информации в таблицы, их использование позволяет легко автоматизировать ввод данных и исключить ошибки ввода. Она может содержать рисунки, графики и др. внедренные объекты. Формы могут также использоваться для просмотра запросов и таблиц на экране.

Отчеты. Отчет это гибкое и эффективное средство для организации данных. Отчеты формируются для печати данных, содержащихся в таблицах и запросах, в том виде, в котором требуется пользователю.

Макросы. Макрос – набор из одной или более макрокоманд, их использование позволяет автоматизировать повторяющиеся операции.

Модули. Модуль содержит Access VBA код, который используется для написания процедур обработки событий. Например, нажатие кнопки для автоматического выполнения операций, которые не могут быть выполнены с помощью макросов.

2.1.4 Типы данных

При создании таблиц для каждого поля необходимо указать тип хранимой в нем информации. Кроме того, для каждого поля можно определить дополнительные свойства. Тип поля указывается во втором столбце окна конструктора путем выбора из заранее определенного списка (рис. 2.1.3).

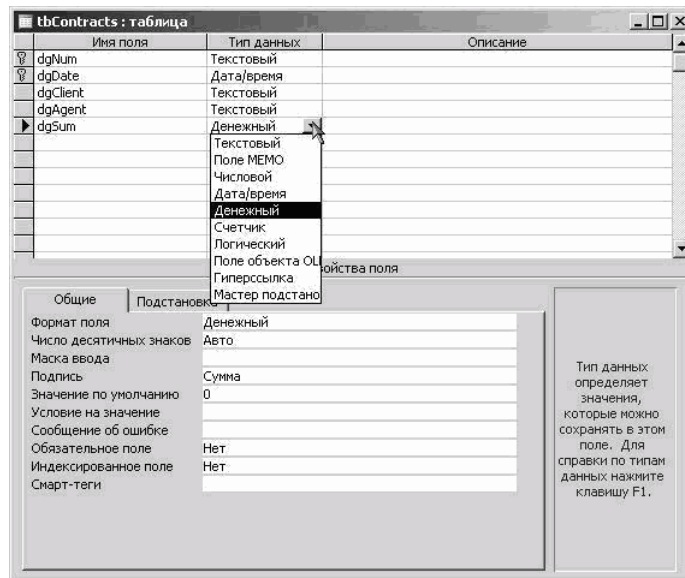


Рисунок 2.1.3 – Определение типов данных

Текстовый тип.

Поля данного типа служат для хранения небольшой текстовой информации такой как имена, фамилии, различные названия и т. д. Размер поля устанавливается пользователем и может быть в пределах от 0 до 255 символов. Поля данного типа используются наиболее часто. Они применяются для хранения такой информации, как номера телефонов с разделителями, имена и фамилии, должности, названия и т. д. На рис. 2.1.4 приведено изображение окна со свойствами текстового поля.

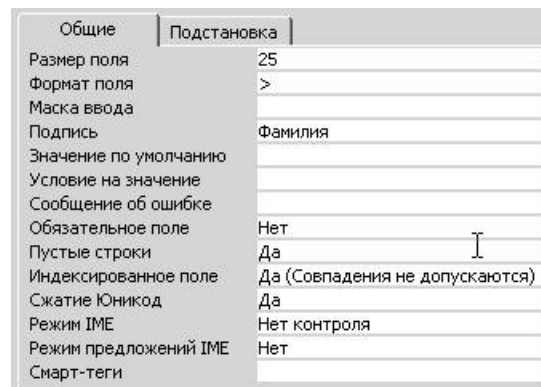


Рисунок 2.1.4 – Свойства поля Текстового типа

Размер поля. В данной строке задается размер поля в символах, то есть максимальное количество символов, которое может храниться в поле.

Формат поля. Задание формата для текстового поля позволяет отображать данные в виде, отличном от вводимого. При этом формат влияет только на то, как данные отображаются в таблице, но в самой БД данные хранятся так, как было введено пользователем.

Примечание. Символы и правила определения формата полей рассмотрены после описания всех типов полей.

Маска ввода. В этой строке можно задать маску ввода. Маска ввода позволяет задать шаблон для вводимого значения поля, а также определить какие символы можно вводить в определенные позиции шаблона. На первый взгляд маска ввода ничем не отличается от формата, но это не так. Увидеть разницу можно уже в момент ввода: у поля с заданным форматированием способ отображения меняется после завершения ввода значения, а при заданной маске шаблон появляется уже в момент начала ввода информации.

Примечание. Символы и правила определения маски ввода рассмотрены после описания всех типов полей.

Подпись. Это текст, который будет отображаться в заголовке поля. Если подпись не задана, то в качестве заголовка поля используется его название. Однако в тех случаях, когда имена полей заданы латинскими буквами или сильно сокращены, пользователю трудно догадаться, что именно необходимо ввести в данное поле. В этом случае можно заполнить свойство поля Подпись. Следует отметить также, что текст Подписи используется в качестве поясняющей надписи визуального элемента, связанного с полем при создании форм и отчетов.

Значение по умолчанию. В этой строке можно ввести текст, который будет помещен в поле каждой новой записи в момент ее создания. Например, пусть имеется поле, в котором хранится информация о статусе компьютера в компьютерном центре. Большинство компьютеров – это пользовательские терминалы. Поэтому можно в данное поле ввести значение по умолчанию: Пользовательский терминал. Таким образом, не нужно будет тратить время на ввод этого стандартного значения, а в редких случаях его легко изменить.

Условие на значение. Не редко встречаются ситуации, когда значение поля должно соответствовать какому-то условию. Это поле позволяет задать его. Для текстовых полей используется функция сравнения строк Like, позволяющая сравнить содержимое поля с заданным шаблоном. При задании шаблона используются символы: «*» любое количество любых символов, «?» – один любой символ. Например, если ввести Like «Про*», то в поле можно будет ввести слова любой длины, начинающиеся с «Про». А если ввести Like «???», то это будет означать, что значение поля может состоять из любых трех символов, не больше и не меньше.

Сообщение об ошибке. В этом свойстве задается текст, который будет выведен в случае нарушения накладываемого пользователем условия на значение поля.

Обязательное поле. Данное свойство может принимать значения только «Да» и «Нет». Если поле обязательно, то ввод записи (транзакция) не будет завершена до тех пор, пока не будет задано значение поля.

Пустые строки. Может принимать только значения «Да» и «Нет». Указывает, допустим, ли ввод в данное поле строк нулевой длины, то есть, можно ли оставить поле пустым.

Индексированное поле. Индексирование поля ускоряет поиск по нему, однако замедляет обновление. Значение данного свойства может принимать только три значения: Нет – поле не индексировано; Да (совпадения допускаются) – по полю строится индекс и не накладывается ограничений; Да (совпадения не допускаются) – по полю строится индекс и запрещается ввод одинаковых значений, т. е. также как и в случае с первичным ключом в таблице не может быть двух записей, содержащих одинаковое значение этого поля.

Сжатие Юникод. Начиная с MS Access 2000, текстовые значения хранятся в юникоде, то есть на каждый символ отведено по два байта, что означает увеличение пространства, занимаемого текстом. Если включено сжатие юникода, то база данных будет занимать меньше места.

Свойства **Режим ИМЕ** и **Режим предложений ИМЕ** необходимы при работе с иероглифами, то есть в тех случаях, когда текстовое поле может содержать иероглифы.

Смарт-теги. В этом свойстве можно указать смарт-тэги, которые будут подключены к полю.

Тип Мемо

Большое текстовое поле, которое может содержать от 0 до 65535 символов. Может использоваться для хранения больших текстовых данных, таких как фрагменты литературных произведений, тексты приказов и распоряжений. Все свойства данного поля отображены на рис. 2.1.5.

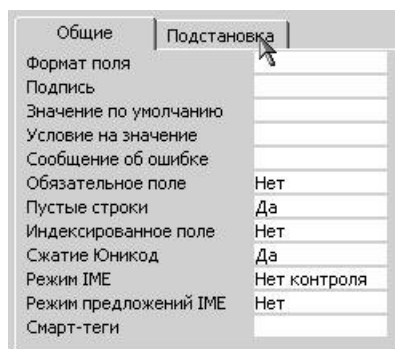
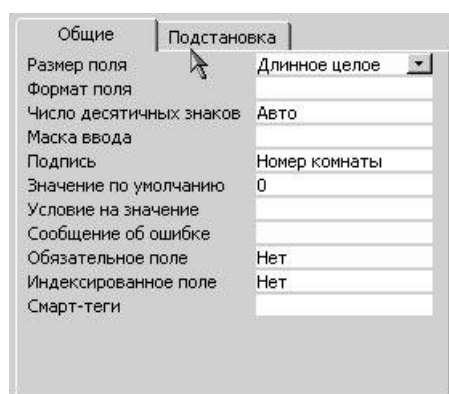


Рисунок 2.1.5 – Свойства поля типа MEMO

Поля данного типа используются для хранения числовой информации (что следует из самого названия типа). Вид числовой информации, хранимой в данном поле представлены в табл.

Тип	Размер, байт	Точность	Описание
Байт (Byte)	1		хранит числа от 0 до 255
Целое (Integer)	2		хранит числа от -32768 до 32767
Длинное целое (Long Integer)	4		хранит числа от -2147483648 до 2147483647
Одинарное с плавающей точкой (Single)	4	7	хранит числа от -3,402823E+38 до -1,401298E-45 и от 1,401298E-45 до 3,402823E+38
Двойное с плавающей точкой (Double)	8	15	хранит числа от -1,79769313486231E+308 до -4,94065645841247E-324 и от 4,94065645841247E-324 до 1,79769313486231E308.
Действительное (Decimal)	12	28	хранит числа от -1038-1 до 1038-1 для .adp-проектов, то есть для файлов серверных проектов; и от -1028-1 до 1028 - 1 для .mdb-проектов, то есть файлов обычных баз данных
Код репликации (Replication ID)	16		16-байтовое поле, используемое для однозначной идентификации записи. Если выбрать этот вид числового поля, то оно будет практически полностью идентично полю типа счетчик.

Вид числовой информации устанавливается в свойстве **Размер поля**. **Формат** числового поля можно как выбрать из списка (рис. 2.1.6), так и задать самостоятельно при помощи специальных символов. Кроме того, для чисел с плавающей точкой можно указать количество цифр после запятой. Остальные свойства полей данного типа уже описаны выше.



а)

Основной	3456,789
Денежный	3 456,79р.
Евро	3 456,79€
Фиксированный	3456,79
С разделителями разрядов	3 456,79
Процентный	123,00%
Экспоненциальный	3,46E+03

б)

Рисунок 2.1.6 – Свойства поля Числового типа

Дата/время

Поля данного типа предназначены для хранения информации, связанной со временем: время суток, даты. Свойство формат позволяет выбрать из списка формат отображения времени или даты: длинный, средний или краткий форматы времени и полный, длинный, средний, краткий форматы даты (рис. 2.1.7). Остальные свойства уже описаны выше.

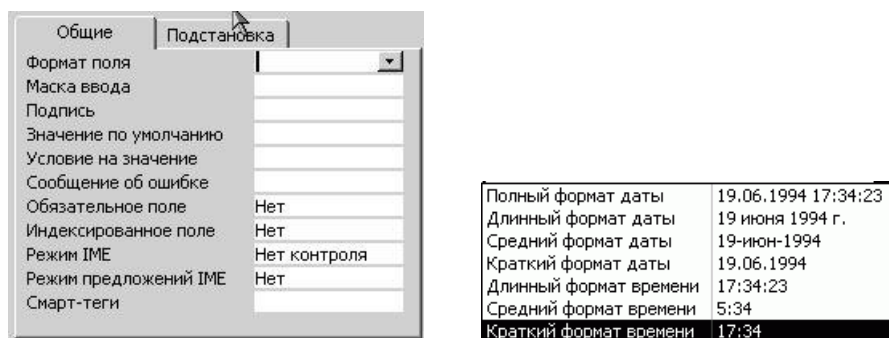


Рисунок 2.1.7 – Свойства поля типа Дата/время

Денежный

Поля данного типа используются для хранения чисел, соответствующих денежным суммам. Это тот же числовой тип данных, но с указанием свойства формат – Денежный. Поэтому разбирать его свойства не имеет смысла, достаточно обратиться к числовому типу данных.

Счетчик

Поля данного типа являются не изменяемыми. Значения данным полям присваиваются автоматически в соответствии с установками свойства Новые значения: Последовательно или Случайно. Свойство Размер поля также может принимать всего два значения: Длинное целое и Код репликации, в последнем случае значение поля состоит из латинских букв и цифр. Поля данного типа используются для однозначного определения записи, например, в тех случаях, когда трудно выявить первичный ключ. Именно так и поступает MS Access, когда самостоятельно определяет ключ таблицы. Следует отметить, что если какая-то запись была удалена из базы данных, то исчезает навсегда и значение поля типа счетчик. То есть если удалена запись, в которой поле типа счетчик имело значение «5», то уже никогда в этой базе данных не будет записи со значением поля-счетчика «5».

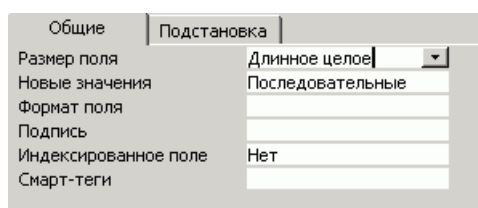


Рисунок 2.1.8 – Свойства поля типа Счетчик

Логический

Поле для хранения логической информации. Например, семейное положение женат/замужем (Да), холост (Нет). Как видно из рис. можно выбрать формат отображения данного поля. В самой же базе данных для положительно результата (Да, Истина, Вкл.) хранится значение – 1, а для отрицательного (Нет, Ложь, Выкл.) значение – 0. Все остальные свойства описаны выше.

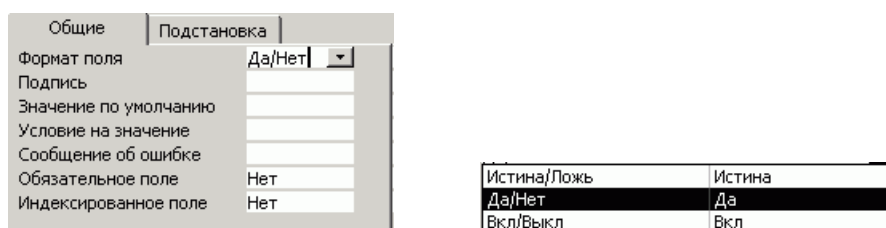


Рисунок 2.1.9 – Свойства поля для хранения Логической информации

Объект OLE

Поля рассматриваемого типа способны хранить любую бинарную информацию (картинки, звуковые и даже видеофайлы) размером до 1Гб. Вся информация, хранимая в поле этого типа должны создаваться в приложениях, способных обмениваться информацией через OLE сервер. Если говорить просто, то OLE сервер это стандартный механизм ОС Windows позволяющий обмениваться данными между приложениями, например при помощи буфера обмена, а также запускать для редактирования/просмотра информации соответствующее приложение. Например, если в это поле вставлен рисунок, то для просмотра рисунка необходимо запустить программу просмотра изображений, она будет приложением-сервером. Часто его используют для хранения именно изображений – например, фотографий работников.

Это поле имеет всего два свойства: позволяющие указать его подпись и является ли оно обязательным.

Гиперссылка

Поля данного типа используются для хранения ссылок на сайты сети ИНТЕРНЕТ, файлы на диске или другие объекты. Поле может хранить комбинацию букв и цифр, при помощи которых записана ссылка на целевой объект. Свойства данного поля весьма похожи на свойства полей текстового типа за тем исключением, что для гиперссылок не задается размер поля.

Задания к лабораторной работе

1. Выполнить индивидуальное задание.

Контрольные вопросы

2. Что называется базой данных?

3. Какую базу данных называют реляционной?
4. Какие основные функции СУБД и что это такое?
5. Какие объекты СУБД Access вы знаете?
6. Какую информацию содержит таблица, в которой нет ни одной записи?
7. Приведите примеры использования различных типов полей в таблицах.
8. Что означает свойство поля «уникальное»?
9. Какой тип поля лучше подходит для ключевого в большинстве таблиц?
10. Назовите свойства поля.

2.2 Лабораторная работа № 2

Тема: Создание и корректировка таблиц

Содержание:

- 2.2.1 Способы создания таблицы
- 2.2.2 Создание таблиц в режиме конструктора
- 2.2.3 Создание таблицы в режиме таблицы
- 2.2.4 Создание таблиц при помощи Мастера таблиц
- 2.2.5 Импорт таблиц
- 2.2.6 Просмотр и редактирование содержания полей
- 2.2.8 Связывание таблиц
- 2.2.9 Задания к лабораторной работе
- 2.2.10 Контрольные вопросы

2.2.1 Способы создания таблицы

При выборе в панели задач пункта Новая база данных будет создан пустой файл базы данных. При этом откроется окно (рис. 2.2.1).

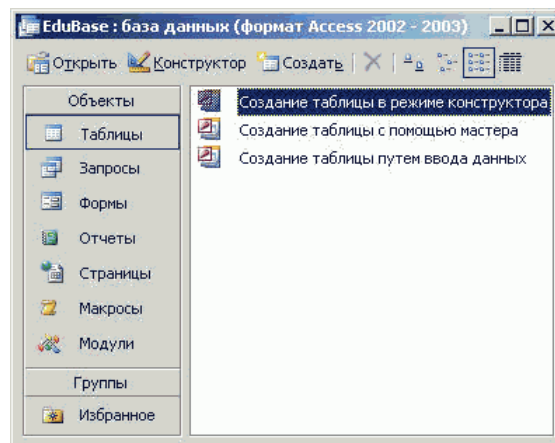


Рисунок 2.2.1 – Создание таблиц

В левой части этого окна приведены кнопки, позволяющие перейти на страницы различных объектов базы данных: Таблицы, Формы, Запросы, Отчеты, Макросы, Модули и т. д. Каждая страница позволяет оперировать (создавать, редактировать и удалять) объекты только одного типа. Начинать разработку базы данных, как уже говорилось необходимо на бумаге, а непосредственное воплощение проекта – с создания таблиц.

Для того чтобы создать таблицу необходимо перейти на страницу Таблиц. Для этого нажимаем на кнопку с надписью [Таблицы]. Затем нажать на панели инструментов кнопку [Создать]. На экране появится окно (рис. 2.2.2), в котором будет предложено выбрать способ создания таблицы.

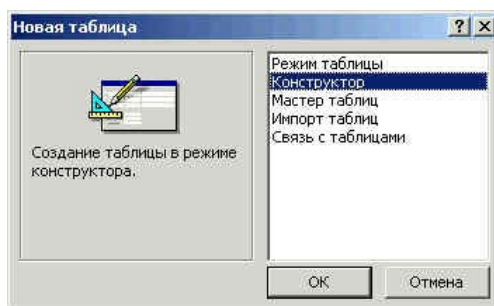


Рисунок 2.2.2 – Создание таблицы

2.2.2 Создание таблиц в режиме конструктора

Режим конструктора предоставляет разработчику баз данных полный спектр возможностей по созданию таблиц. В данном режиме можно настроить достаточно большое количество свойств для каждого из полей, определить названия полей, тип хранимых данных, списки подстановки и так далее. Кроме того, только в режиме конструктора можно создать именно ключ, а не ключевое поле. То есть режим конструктора позволяет создать ключ таблицы, состоящий из нескольких полей.

Для создания таблицы в режиме конструктора следует в окне выбора способа создания таблицы щелкнуть на пункте **Конструктор** и нажать кнопку [Ок]. После этого на экране появится окно конструктора.

Верхняя часть окна конструктора представляет собой таблицу (рис.2.2.3). В первом столбце этой таблицы вводятся имена полей, во втором столбце из списка выбирается тип информации сохраняемой в данном поле, а в третьем столбце можно ввести описание данного поля, то есть краткую подсказку о назначении данного поля. Текст третьего столбца отображается только в режиме конструктора и служит для нужд разработчика.

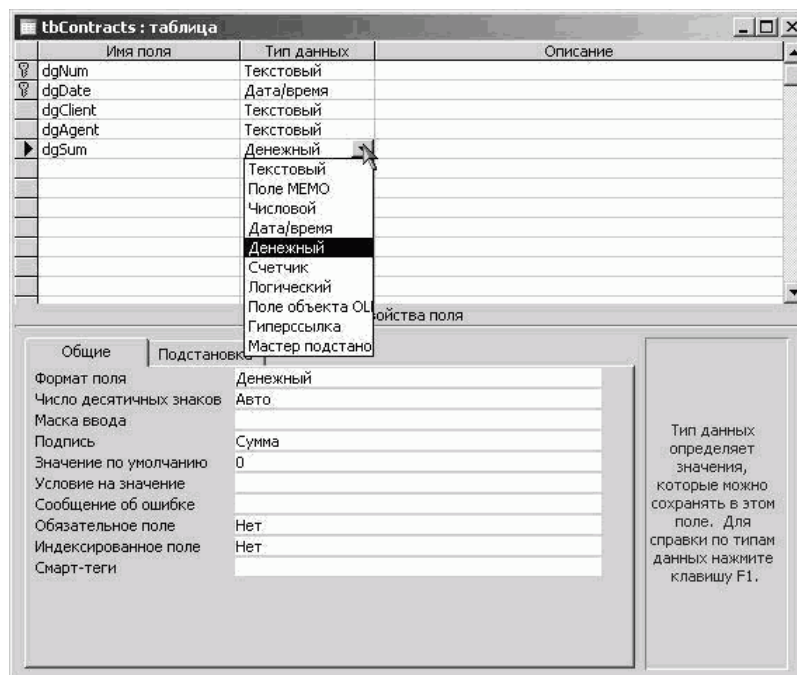


Рисунок 2.2.3 – Создание таблицы в режиме Конструктора

Нижняя часть окна конструктора таблиц состоит из двух вкладок. На первой из них представлены свойства текущего поля. Вторая вкладка служит для создания сложного поля, значения которого выбираются из списка. Список может храниться в другой таблице или его можно ввести вручную. Справа от вкладок находится область, содержащая текст, отображаемый синим цветом. Это окно, в котором появляется подсказка по текущему свойству или типу поля. Содержимое справки зависит от текущих действий разработчика.

На рисунке приведен пример таблицы с именем **Контракты**. Как видно, имена всех полей записаны при помощи латинских букв. Хотя это и не является обязательным правилом, но использование в таблицах имен, состоящих только из латинских букв, цифр и символа подчеркивания повышает переносимость приложения. Конечно, если вы создаете базу данных только для себя, то это не важно, но если продукт разрабатывается для некоторой организации или может быть использован за рубежом (где нет кириллицы), то соблюдение этих правил позволит избежать множества проблем и переделок.

Во всех приложениях существует условные и безусловные правила. Следовать *условным правилам* не обязательно, поскольку они сформулированы разработчиками для удобства. Условные правила прививают хороший стиль, а также позволяют получить проект понятный для любого разработчика. Это своего рода «цеховой устав». Не выполнение *безусловных правил* приводит к ошибкам в самом проекте без устранения, которых проект не будет работать. В качестве примера приведем условные и безусловные правила задания имен объектов MS Access.

Безусловные. Имена объектов Access должны содержать от 1 до 64 символов (не менее и не более); имена объектов Access могут состоять из букв, цифр и специальных символов кроме точки (.), восклицательного знака (!), квадратных скобок ([]), символа апострофа ('), а также комбинаций служебных символов с кодами от 0 до 31; имя не должно начинаться с пробела.

Условные. Имена объектов Access следует начинать с префикса, состоящего из 2-3 символов, указывающего на принадлежность поля какой либо таблице или на вид объекта (tbContracts – таблица «Договора», dgNumb – поле «Номер договора» из таблицы Договора); в именах объектов Access следует использовать только латинские буквы, цифры и символ подчеркивания; имена полей должны быть максимально короткими, всю остальную информацию можно записать в комментарии или в свойстве Подпись. По возможности следует давать полям осмысленные имена.

Не соблюдение безусловного правила приводит к появлению окна ошибки. Так на приведенном ниже рисунке в имени поля использован восклицатель-

ный знак. Пока мы не исправим эту ошибку, MS Access не позволит работать с другими полями и таблицами.

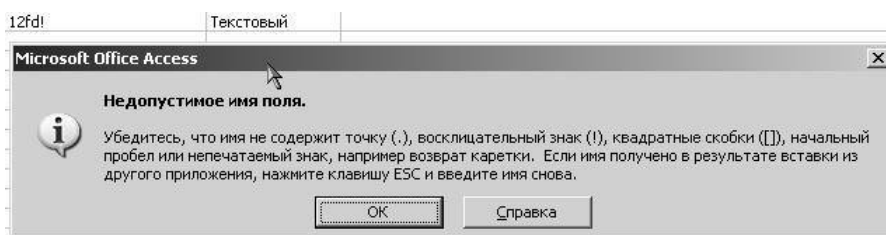


Рисунок 2.2.4 – Сообщение об ошибке в названии поля

Не соблюдение условного правила, касающегося длины имени может привести к тому, что данные нельзя будет считать из старых СУБД, но с другой стороны создаст самодокументированную структуру таблицы. То есть, вместо того, чтобы описывать назначение поля в комментарии достаточно будет дать длинное, но осмысленное имя. Например, вместо поля Dgklschn ввести поле dgNomerSchetaKlienta. К полю с таким именем не нужно писать комментарии.

Мастер подстановок. Последний пункт списка не является типом поля. Выбор этого пункта запускает мастер, позволяющий создать столбец подстановок для данного поля и тип поля поэтому определяется автоматически. В качестве столбца подстановок могут выступать как другие таблицы/запросы, так и просто список значений. Например, для хранения пола человека нет нужды создавать отдельную таблицу, поэтому можно обойтись просто списком, который разработчик может ввести вручную. А если рассматривать поле *Должность*, то здесь уже простым списком не обойтись, поскольку должностей на предприятии может быть достаточно много и они могут изменяться. В последнем случае поле следует связать с таблицей-справочником по должностям.

Рассмотрим как работает мастер подстановок на примере таблицы отдела кадров. Пусть имеется таблица с полями: okFam, okName, okVoзраст, okPol, olDolzhn. Создадим для двух последних полей столбцы подстановки.

Выбираем для поля okPol в столбце Тип пункт **Мастер подстановок** и на экране появляется окно первого шага. Поскольку вводится фиксированный набор значений (перечислены варианты пола), то в этом окне выбирается второй пункт: **Будет введен фиксированный набор значений.**

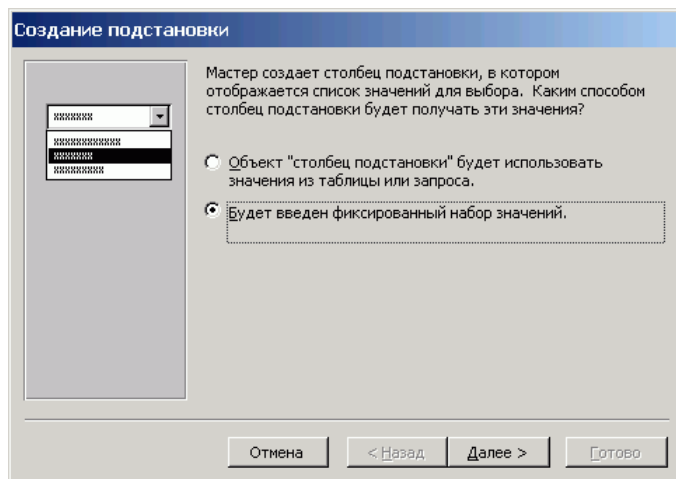


Рисунок 2.2.5 – Диалоговое окно Создание подстановки. Шаг первый

На втором шаге необходимо указать количество столбцов и ввести данные в эти столбцы. После ввода значений списка нажимаем кнопку **Далее**.

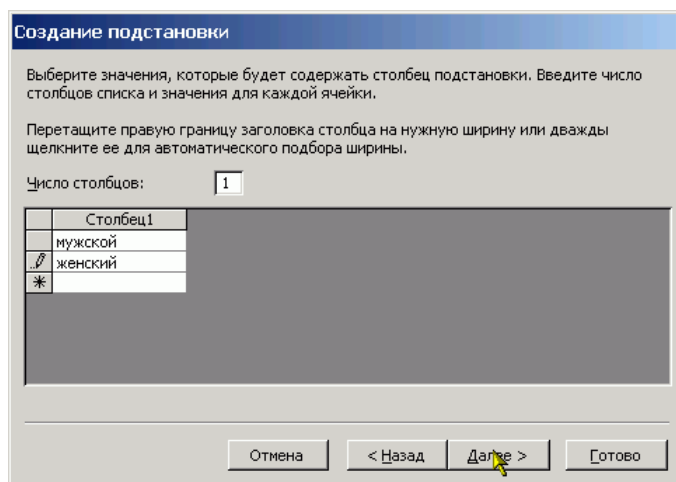


Рисунок 2.2.6 – Диалоговое окно Создание подстановки. Шаг второй

На следующем (завершающем) шаге мастер просит ввести подпись для поля. Однако, на самом деле вводится не подпись а имя поля, это следует помнить.

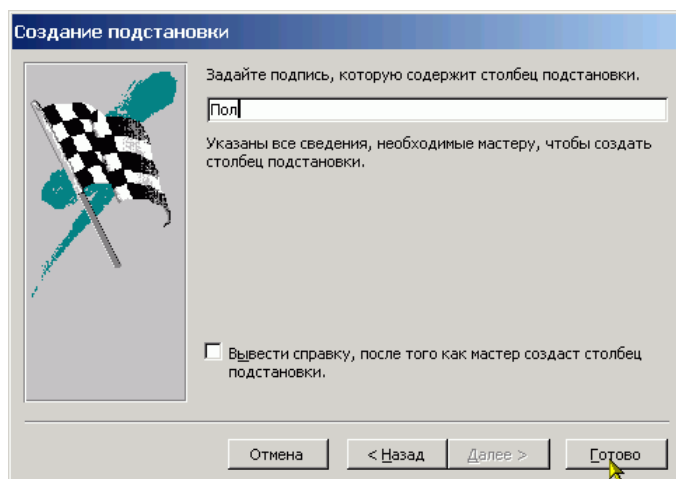


Рисунок 2.2.7 – Диалоговое окно Создание подстановки. Шаг третий

После задания имени нового поля нажимаем кнопку **Готово**, и создание столбца подстановки с фиксированным набором значений закончено.

Теперь рассмотрим создание столбца подстановки с привлечение внешней таблицы. Окно **Мастер подстановок** на первом шаге такое же, как и в предыдущем случае, но выбираем в нем первый пункт.

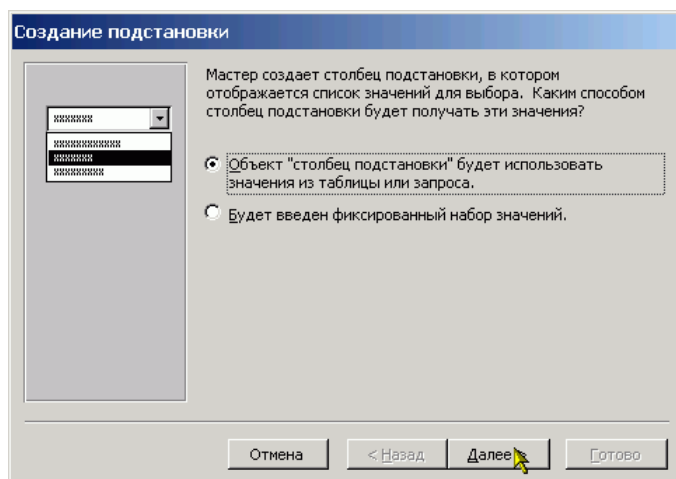


Рисунок 2.2.8 – Диалоговое окно Создание подстановки. Шаг первый

Затем выбираем из приведенного списка таблицу или запрос, источник информации для подстановки. В нашем случае это таблица `tbDolzhnosty`.

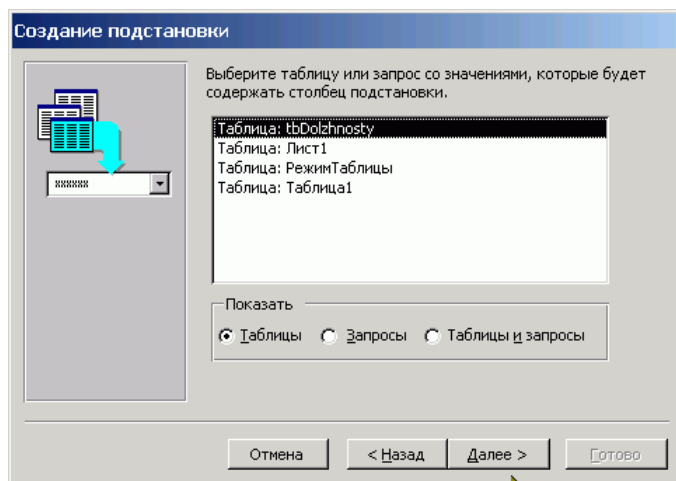


Рисунок 2.2.9 – Диалоговое окно Создание подстановки. Шаг второй

На следующем шаге необходимо указать те поля, которые будут использоваться для подстановки. В нашем случае это всего лишь одно поле. Выделяем его и нажимаем кнопку с изображением знака больше «>>» (рис.). После этого имя выбранного поля появляется в правом окне и можно переходить к следующему шагу.

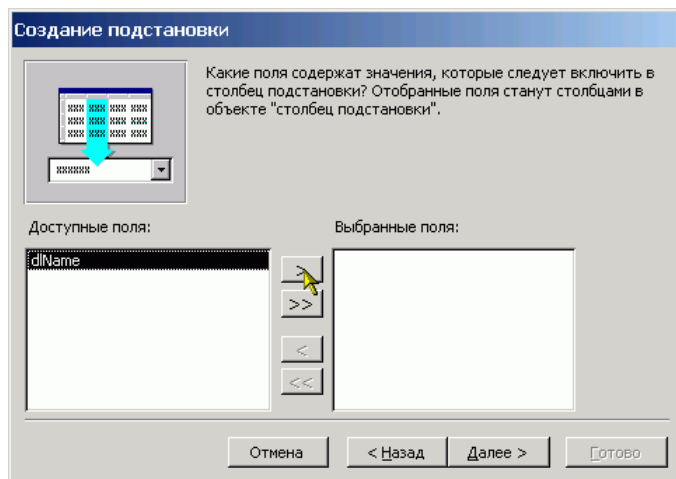


Рисунок 2.2.10 – Диалоговое окно Создание подстановки. Шаг третий

На четвертом шаге мастера подстановок можно определить вид сортировки для значений поля(ей) в выпадающем списке.

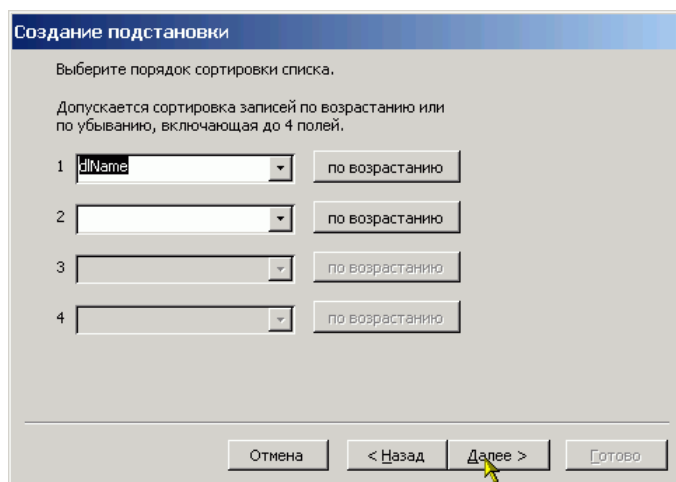


Рисунок 2.2.11 – Диалоговое окно Создание подстановки. Шаг четвертый

На пятом шаге задается ширина столбцов подстановки.

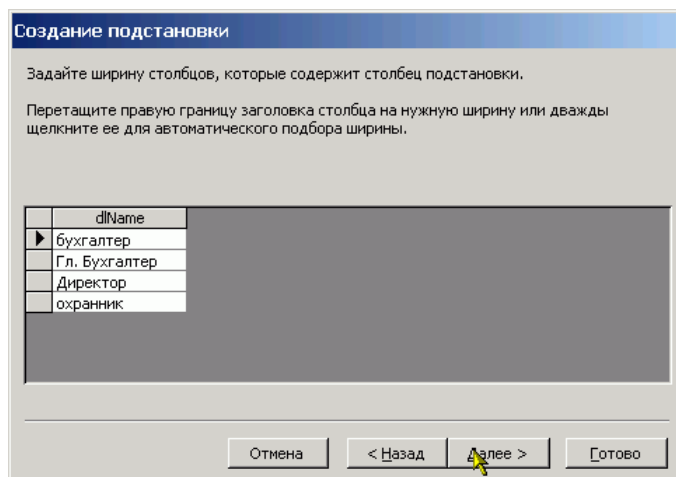


Рисунок 2.2.12 – Диалоговое окно Создание подстановки. Шаг пятый

Шестой шаг является последним и также как и в случае постоянного списка на нем необходимо определить имя созданного поля.

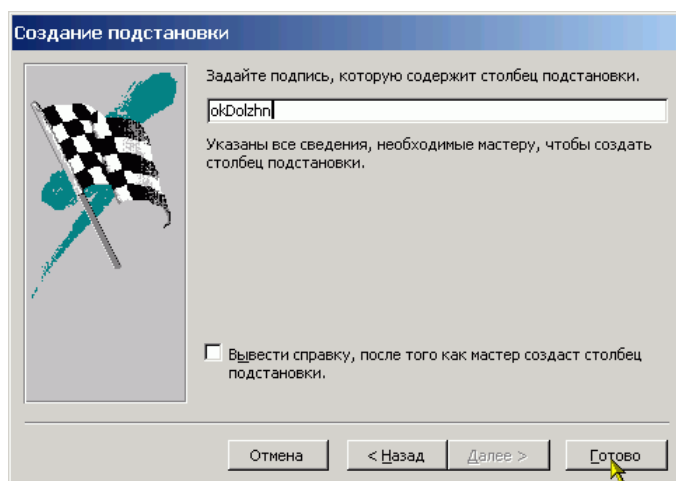


Рисунок 2.2.13 – Диалоговое окно Создание подстановки. Шаг шестой

В результате проведенных действий мы получаем таблицу, в которой при вводе не нужно вручную набирать наименование должности и пола – достаточно выбрать их из списка. На приведенном рисунке показано как это будет выглядеть. Конечно же каждый список появляется отдельно и показанная ситуация никогда не реализуется. Просто в данном случае пришлось совместить два рисунка, чтобы показать все результаты.

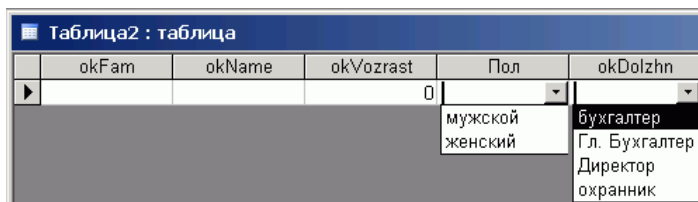


Рисунок 2.2.14 – Создание подстановки. Результат

Формат поля

Свойство формат задает шаблон отображения информации данного поля. То есть вне зависимости от того, как вводится или в какой форме хранится значение поля, отображаться оно будет в соответствии с заданным форматом.

Для задания формата *текстовых* и *Мето*-полей используются следующие символы:

- < отображение текста при помощи строчных букв;
- > отображение текста при помощи прописных букв;
- @ должен быть любой символ;
- & может быть любой символ.

Пример

Пусть для текстового поля задан формат: (&&&&) @@-@@-@@, при этом пользователь набрал: 3472334455, в результате применения формата, число в ячейке будет отображаться в виде: (3472) 33-44-55, но в самой базе данных число будет храниться без скобок, пробелов и дефисов.

Формат поля может состоять из двух частей, разделенных символом «;».

Например, у текстового поля задан формат вида: @@-@@@;«Данные не введены»[Red], если в поле будет введено какой-либо значение, то оно будет отображаться согласно формата, а если в поле ввод еще не производился, то в нем будет отображаться текст красного цвета «Данные не введены».

Для числовых полей свойство формат содержит несколько готовых шаблонов форматирования поля, выпадающий список готовых шаблонов показан выше. Разработчик может выбрать один из них, либо задать свой собственный формат, используя специальные символы форматирования.

Самостоятельно определяемый шаблон может состоять из четырех частей, разделенных символом *точка с запятой (;)*: «формат для положительных чисел»; «формат для отрицательных чисел»; «формат для нуля»; «формат при отсутствии значения». При этом используются следующие специальные символы:

- 0 в позиции отображается цифра, либо ноль;
- # в позиции отображается цифра, либо пусто;
- \$, % вывод в позиции соответствующего знака.

Кроме того, в квадратных скобках может быть указан цвет.

Например, пусть задано следующее форматирование:

##0,00[Green]; # ##0,00[Red]; «Ноль»; «Пусто»

В этом случае положительные числа будут отображаться зеленым цветом, отрицательные – красным, при вводе нулевого значения будет отображаться текст «Ноль», а в еще не заполненном поле будет отображаться текст «Пусто».

Для поля типа **Дата/время** также существует заранее определенный набор готовых шаблонов форматирования. Однако, в редких случаях разработчику требуется создать свой собственный шаблон для вывода даты или времени, используя следующие обозначения:

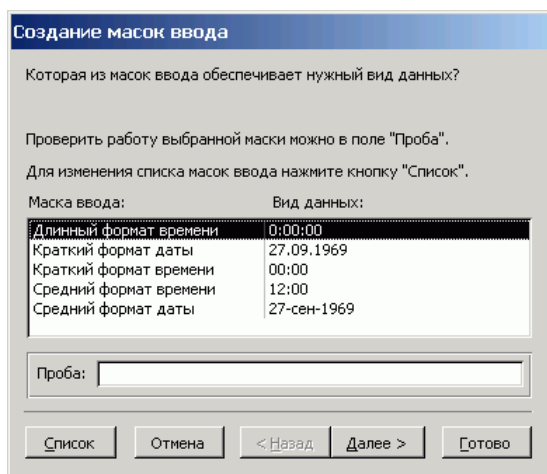
Символ	Значение
:	Разделитель частей времени. Его вид зависит от локальных настроек машины.
/	Разделитель частей даты
c	То же самое, что и полный формат даты
d, dd	День месяца – одна или две цифры

Символ	Значение
ddd	Первые три буквы дня недели, например, <i>Втр</i> или <i>Суб</i> .
dddd	Полное название дня недели
dddddd	То же самое, что короткий формат даты
dddddd	То же самое, что и длинный формат даты
w	Номер дня недели, число от 1 до 7
ww	Номер недели в году, число от 1 до 52
m, mm	Номер месяца, одна или две цифры
mmm	Первые три буквы названия месяца
mmmmm	Полное название месяца
q	Номер квартала года, в котором находится указанная дата
y	Номер дня в году, число от 1 до 366
yy	Последние две цифры года
yyyy	Четыре цифры года
h, hh	Час – одна или две цифры
n, nn	Минуты – одна или две цифры
s, ss	Секунды – одна или две цифры
tttt	То же самое, что длинный формат времени
AM/PM или A/P	Двенадцати часовое представление времени с указанием соответствующих обозначений «до полудня» и «после полудня» прописными буквами.
am/pm или a/p	Двенадцати часовое представление времени с указанием соответствующих обозначений «до полудня» и «после полудня» строчными буквами.
AMPМ	Двенадцатичасовое представление времени в соответствии с локальными региональными настройками

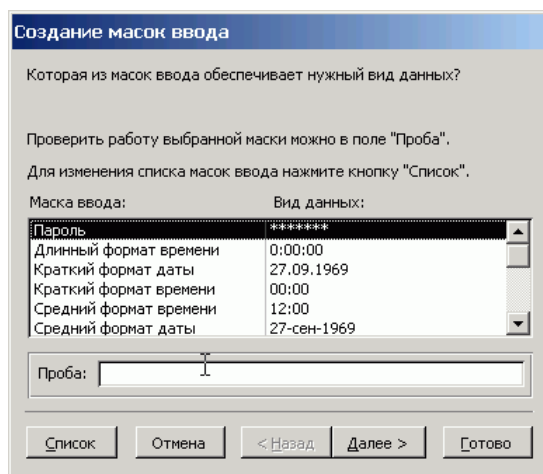
Для *логического* поля также можно не только выбрать готовый формат, но и задать свой. В качестве формата логического поля указываются в двойных кавычках те слова, которое будут отображаться в качестве логических значений. Слова разделяются символом *точка с запятой* (;), и формат также начинается с этого же символа. Например, если необходимо, чтобы в качестве логических значений выступали слова *Открыто* или *Закрыто*, то необходимо в свойстве формат соответствующего логического поля ввести следующий текст: ;«Открыто»;«Закрыто»

Маска ввода

Как указывалось ранее, **Маска ввода** определяет, как данные будут вводиться в поле. Также как и в случае с форматом поля существует заранее определенный набор готовых решений. Готовые шаблоны можно применить только к текстовым и полям типа **Дата/Время**. Для того чтобы вызвать список готовых шаблонов необходимо щелкнуть левой кнопкой мыши в строке свойства, то есть перейти в строку свойства. После этого справа от строки появится кнопка с изображением троеточия. Нажатие этой кнопки и выводит список доступных масок. Примеры этого окна показаны ниже: правое для текстового поля, левое для поля типа **Дата/Время**.



а)



б)

Рисунок 2.2.15 – Диалоговое окно создания масок ввода

Нажатие кнопки [Список] откроет окно, в котором разработчик может создать свою собственную маску и сохранить ее в списке готовых шаблонов для дальнейшего использования.

Однако часто встречаются случаи, когда маску ввода необходимо применить и к числовым полям. В этом случае ее необходимо задать вручную, то есть ввести в свойство **Маска ввода**. И при создании маски вручную и при создании нового шаблона используются одни и те же символы. Ниже приведена таблица, в которой перечислены эти символы.

Символ	Значение
0	Цифра от 0 до 9, ввод обязателен. То есть позиция в любом случае <u>должна</u> быть заполнена. Знаки «+» и «-» не допускаются.
9	Цифра от 0 до 9 или пробел. То есть позицию можно оставить пустой. Знаки «+» и «-» не допускаются.
#	Цифра или пробел. Допусти также ввод знаков «+» и «-».
L	Текстовый символ (буква). Ввод обязателен.
?	Текстовый символ (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.
>	Преобразовать все символы к верхнему регистру
<	Преобразовать все символы к нижнему регистру.
!	Заполнение маски справа налево.
\	Символ, означающий, что следующий за ним литерал является простым символом. Любой символ, при помощи которого определяется Маска ввода , называется литералом. В этой таблице и описываются все литералы. То есть, появление их в тексте маски означает, что их необходимо интерпретировать особым образом, как команду. Однако иногда необходимо в маске использовать эти литералы. Например, символ &, в этом случае перед ним необходимо поставить обратный слэш, то есть

Символ	Значение
	символ "\". То же самое касается и скобок.
. , : ; / -	Различные разделители для числовых данных и данных типа Дата/Время.

В общем случае маска состоит из трех частей, но вторую и третью можно опустить. Каждая часть маски разделяется символом *точка с запятой «;»*:

«непосредственно сама маска»; «0 или 1» «символ»

непосредственно сама маска, то есть тот шаблон, который определяется пользователем.

0 или 1, указывающие как данные хранятся в таблице. Если указывается 0, то это означает, что в таблице хранится вводимое пользователем значение вместе с символами маски, то есть если в маске присутствуют скобки, пробелы, дефисы и т. д., то они хранятся вместе с числом. Если же указать 1 или оставить эту часть не заполненной, то в таблице базы данных будет храниться только то, что ввел пользователь.

символ, который отображается в пустых позициях маски. Если оставить не заполненным, то используется пробел.

2.2.3 Создание таблицы в режиме таблицы

При выборе этого режима откроется окно похожее на рабочий лист Excel. Только в этом окне около двух десятков строк и десять столбцов. Каждый столбец имеет свое имя, состоящее из слова «Поле» и номера столбца. Если дважды щелкнуть на названии столбца, то название можно изменить. Именно так были изменены названия первых двух столбцов и начато изменение названия третьего.

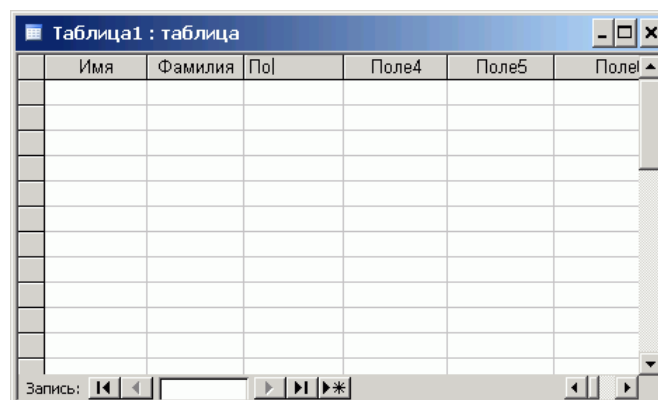


Рисунок 2.2.16 – Изменение названий столбцов

Для добавления новых полей необходимо щелкнуть правой кнопкой мыши на заголовке любого столбца. После этого появится контекстное меню (рис. 2.2.17а).

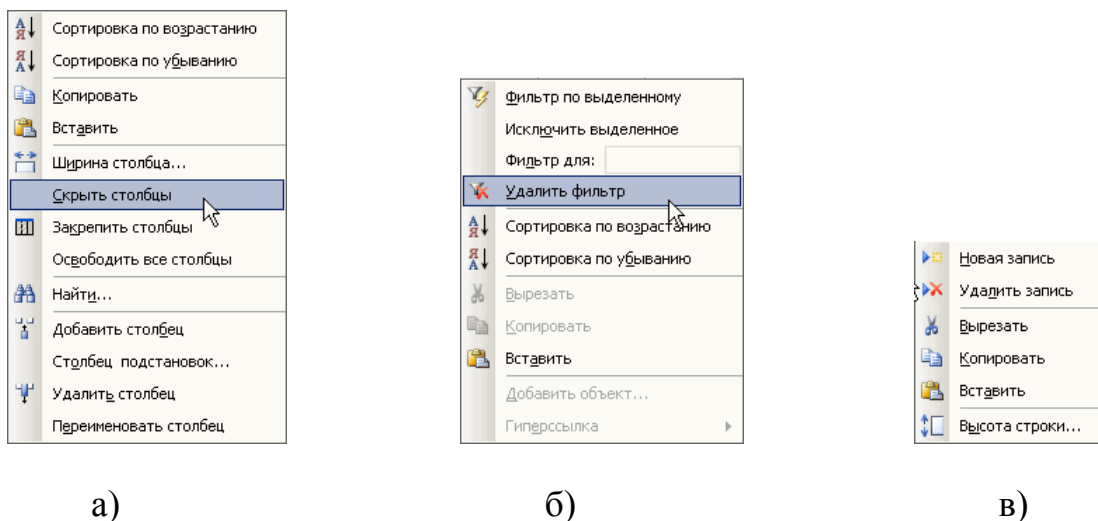


Рисунок 2.2.17 – Контекстно-зависимое меню

При помощи команд данного меню можно и добавить дополнительные столбцы (то есть поля), и удалить лишние. Кроме этого имеющиеся записи можно отсортировать, найти необходимую информацию и произвести некоторые другие действия.

Если выполнить правый щелчок в какой-либо ячейке, то есть вне заголовка столбца, то будет выведено контекстное меню по работе со значениями данного поля (рис 2.2.17б).

Если щелкнуть правой кнопкой мыши в первом столбце, ячейки которого также как и ячейки первой строки закрашены серым цветом, то будет выведено контекстное меню по работе с записями (рис. 2.2.17в).

Таким образом, можно определить поля. Формат каждого поля MS Access определяет сам на основании анализа содержимого полей. Например, если все ячейки некоторого поля содержат только числа, то полю будет дан формат: *Числовой*.

После определения полей таблицы ее можно закрыть. Если таблица еще ни разу не была сохранена, то будет выдано окно с предложением задания имени таблицы, а затем сообщение о том, что в таблице не задан первичный ключ и предложение создать его автоматически. Если вы соглашаетесь на автоматическое создание ключа, то необходимо нажать кнопку Да и в таблицу будет добавлено поле типа *счетчик*. Те же самые запросы выдаются и при сохранении таблицы без ее закрытия. Что касается задания имени таблицы, то рекомендуется перед самым именем ставить префикс tb или tbl, чтобы далее (например, при составлении отчетов или запросов) по названию уже определять, что это таблица.

Режим таблицы может оказаться удобным в том случае, когда необходимо быстро создать простую таблицу, однако я не рекомендую использовать его, так как он не дает полного доступа ко всем параметрам структуры таблицы.

2.2.4 Создание таблиц при помощи Мастера таблиц

Мастер таблиц, также как и мастер создания базы данных, позволяет создать таблицу на основании имеющихся шаблонов в несколько шагов. На первом шаге предлагается определиться с набором полей таблицы.

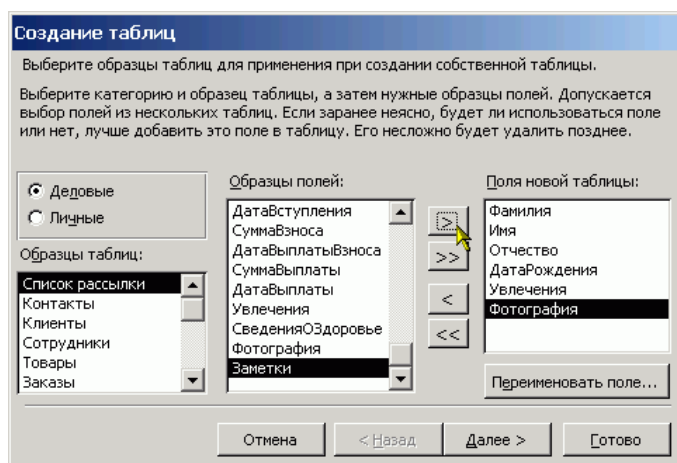


Рисунок 2.2.18 – Диалоговое окно Мастера таблиц. Шаг первый

Набор полей составляется из готовых шаблонов (образцов). Для удобства шаблоны разделены на категории: Деловые и Личные – а также из них составлены стандартные таблицы. Новая таблица составляется путем включения в нее готовых шаблонов полей. Это делается следующим образом:

1. Выбирается категория;
2. В первом столбце выбирается шаблон готовой таблицы. При смене таблицы меняется содержимое второго столбца;
3. Во втором столбце выбирается образец поля. Как правило, названия образцов полей говорят сами за себя и дополнительных пояснений не требуют;
4. Нажимается кнопка с изображением знака больше [$>$]. Образец поля появляется в четвертом столбце. Если стандартное имя поля не нравится, то его можно переименовать, нажав кнопку, расположенную под четвертым столбцом.

Процедура повторяется до тех пор, пока таблица не будет укомплектована в соответствии с желаниями разработчика. Можно также перенести все поля из текущего образца таблицы. Для этого после выбора шаблона таблицы необходимо нажать кнопку с двойным знаком «больше». Если поле было перенесено по ошибке или просто стало не нужным, то его можно удалить из таблицы, для этого необходимо выделить его в четвертом столбце и нажать кнопку

с изображением знака меньше [$<$]. Для того чтобы удалить все поля из таблицы необходимо нажать кнопку с изображением двойного знака «меньше».

После определения набора полей, то есть структуры таблицы следует нажать кнопку [Далее].

На втором шаге мастер таблиц предлагает задать имя новой таблицы, а также выбрать способ создания ключа таблицы. В верхней строке вводится имя таблицы и, если определение ключа разработчик оставляет на усмотрение MS Access, то после нажатия кнопки Далее происходит переход к третьему шагу.

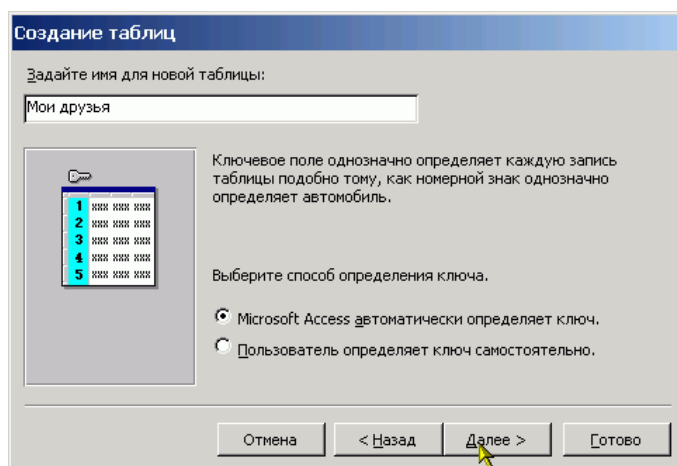


Рисунок 2.2.19 – Диалоговое окно Мастера таблиц. Шаг второй

Если разработчик предпочитает самостоятельно определить ключ таблицы, то ему следует указать это внизу формы. После этого нажимается кнопка [Далее], и открывается окно мастера для задания параметров ключа.

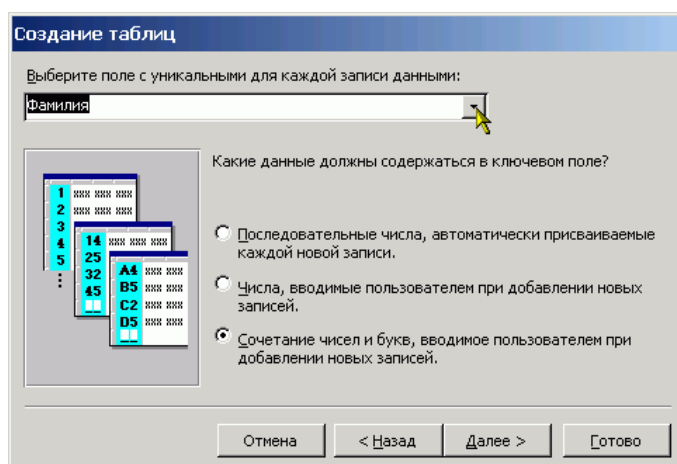


Рисунок 2.2.20 – Диалоговое окно Мастера таблиц. Шаг третий

К сожалению, в данном режиме можно создать только *ключевое поле*, что не всегда возможно. В верхнем поле со списком разработчик должен выбрать

ключевое поле, а затем внизу выбрать один из трех пунктов, характеризующих содержимое ключевого поля. В базе данных не может быть двух записей с одинаковым значением ключа. После настройки всех параметров следует нажать кнопку [Далее].

Если в БД уже есть таблицы, то на третьем шаге разработчику предлагается определить связи между новой и уже имеющимися таблицами. Если таблица является изолированной, то изменения не требуются, тогда следует нажать кнопку [Далее]. В противном случае выбираем из списка строку, содержащую имя таблицы, которая будет связана с новой и нажимаем кнопку [Связи...]

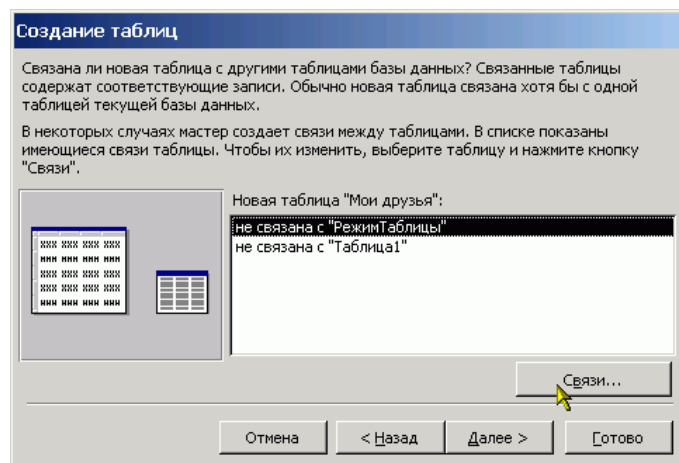


Рисунок 2.2.21 – Диалоговое окно Мастера таблиц. Шаг четвертый

На экране появится окно задания типа связи между таблицами. Следует выбрать любой из пунктов и нажать кнопку [Ok]. В результате на экране появится окно третьего шага, в котором будет указан выбранный тип связи между таблицами.

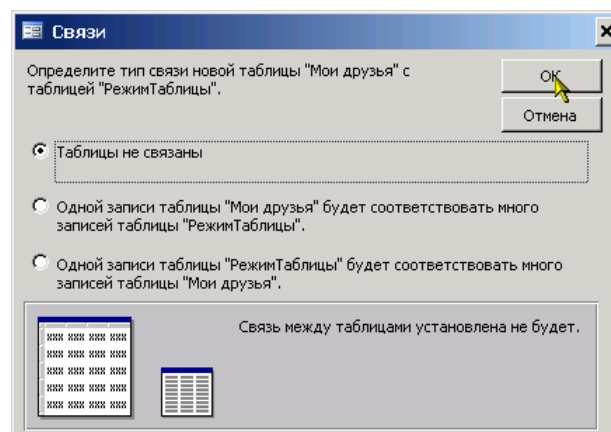


Рисунок 2.2.22 – Диалоговое окно Мастера таблиц. Шаг пятый

После создания связей между таблицами следует нажать кнопку [Далее]. Появится окно последнего четвертого шага, в котором предлагается выбрать

дальнейшее действие. Необходимо выбрать одно из предложенных и нажать кнопку [Готово].

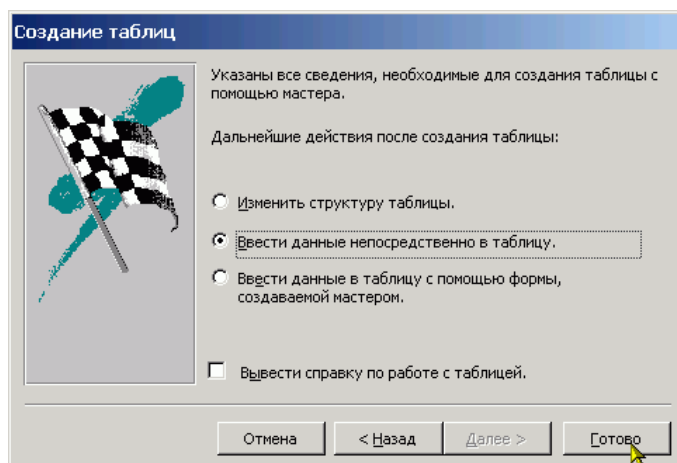


Рисунок 2.2.23 – Диалоговое окно Мастера таблиц. Шаг шестой

Изменить структуру таблицы, то есть произвести «более тонкую» настройку свойств полей таблицы. При выборе этого пункта откроется окно конструктора таблиц.

Ввести данные непосредственно в таблицу. При выборе данного пункта откроется окно с однострочной таблицей. В заголовке столбцов таблицы будут указаны имена полей. В этом режиме можно ввести данные, то есть заполнить таблицу.

Ввести данные в таблицу при помощи формы, создаваемой мастером. При выборе этого пункта откроется простая экранная форма, позволяющая ввести данные в таблицу.

Таким образом, можно создать таблицу, практически полностью отвечающую требованиям разработчика и ввести в нее данные. Однако этот способ создания таблиц хоть и имеет больше возможностей, по сравнению с режимом таблицы, но также ограничен своими рамками и не предоставляет полной свободы.

2.2.5 Импорт таблиц

Часто встречается ситуация, когда уже существует некая таблица с данными, но хранящаяся в формате, отличном от MS Access, например, книга MS Excel или простой текстовый файл. Создание структуры новой таблицы в MS Access и внесение уже имеющихся данных может занять много времени и сил. Гораздо проще импортировать эти данные в базу данных MS Access. При этом будет сохранена структура имеющейся таблицы.

MS Access способен импортировать данные из многих форматов. Специально для данного пособия были созданы файлы в форматах MS Excel, HTML и

dBASE. Первые два формата можно отнести к текстовым, поэтому процедура импорта этих файлов практически идентична. Третий формат является форматом файла базы данных, поэтому его импорт проходит намного проще, однако при этом часто возникают проблемы с кодировкой русского текста.

Импортирование таблицы из рабочей книги MS Excel. Для начала указывается, что необходимо импортировать таблицу из внешнего файла. Затем в появившемся окне перейти в папку с файлом рабочей книги и выбрать в выпадающем списке нижней строки Тип файла – MS Excel (рис.). После этого в списке файлов текущей папки останутся только файлы MS Excel. Далее необходимо указать нужный файл и нажать кнопку [Импорт].

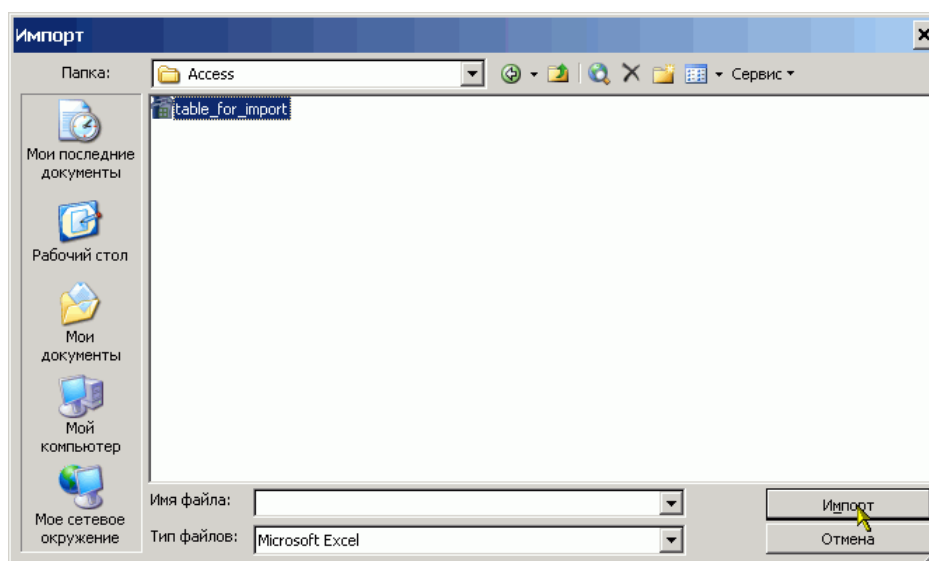


Рисунок 2.2.24 – Импорт таблиц. Выбор импортируемой таблицы

После этого на экране появится первое окно мастера импортирования таблиц. На первом шаге разработчику предлагается выбрать лист рабочей книги, на котором располагается таблица. Для удобства внизу отображается найденная таблица выбранного в данный момент листа.

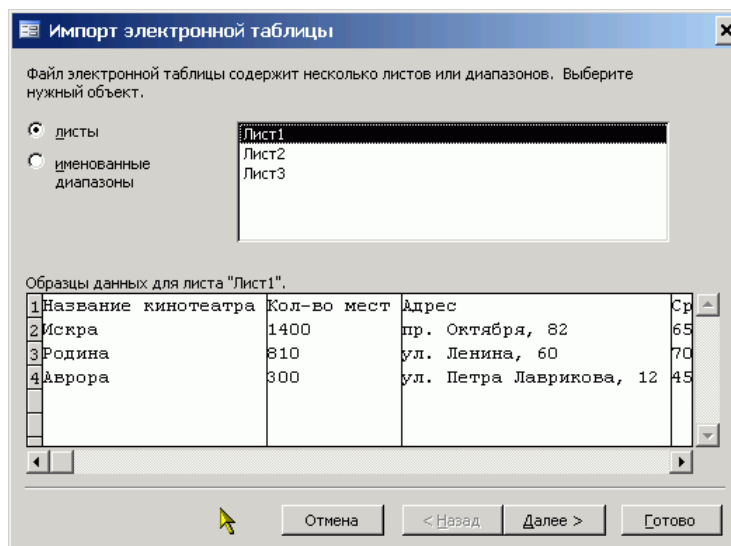


Рисунок 2.2.25 – Импорт таблиц. Шаг первый

После выбора листа и нажатия кнопки [Далее] происходит переход к следующему окну мастера, в котором необходимо указать содержит ли первая строка найденной таблицы заголовки столбцов. Если установить галочку в соответствующей строке, то заголовки столбцов станут именами полей новой таблицы, в противном случае поля таблицы получат имена *Поле1*, *Поле2* и т.д.

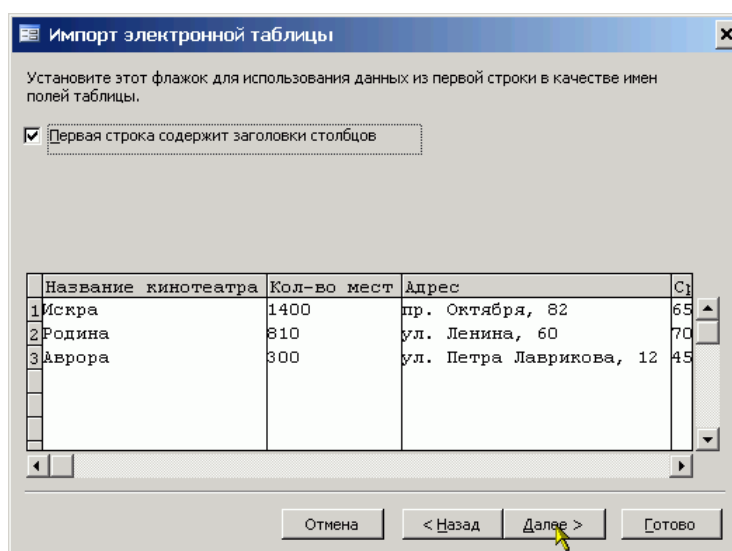


Рисунок 2.2.26 – Импорт таблиц. Шаг второй

На следующем шаге предлагается выбрать назначение импортируемых данных поместить ли их в новую таблицу или добавить к уже имеющейся. В последнем случае таблицу необходимо выбрать из списка.

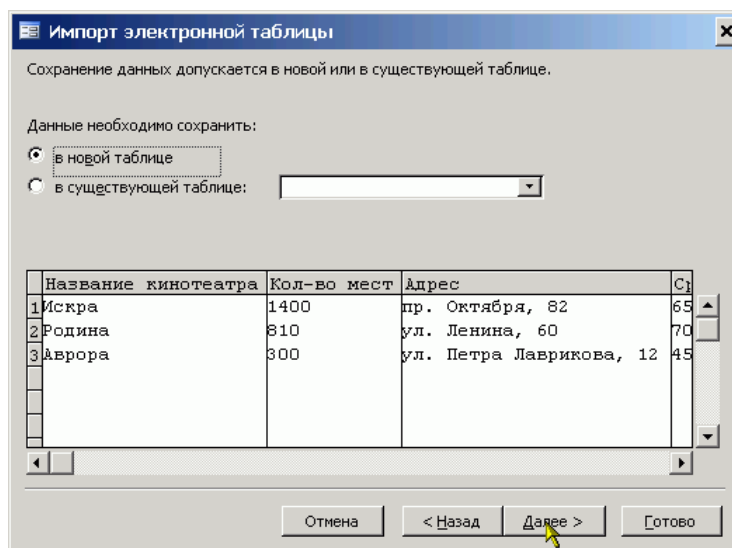


Рисунок 2.2.27 – Импорт таблиц. Шаг третий

На четвертом шаге мастера можно изменить имена полей таблицы, выбрать тип данных, а также указать является ли данное поле индексируемым.

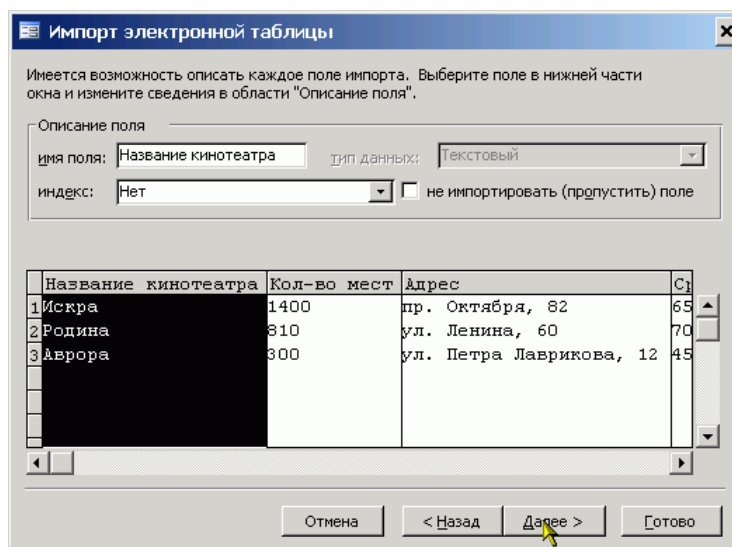


Рисунок 2.2.28 – Импорт таблиц. Шаг четвертый

Пятый шаг позволяет выбрать способ создания ключа таблицы. Разработчик может указать ключевое поле самостоятельно, либо предоставить это MS Access. Но MS Access не определяет, какое поле должно быть ключевым, а просто добавляет к таблице еще одно поле типа Счетчик, которое может принимать только целые значения и они никогда не повторяются. Можно также вообще не создавать ключа в таблице.

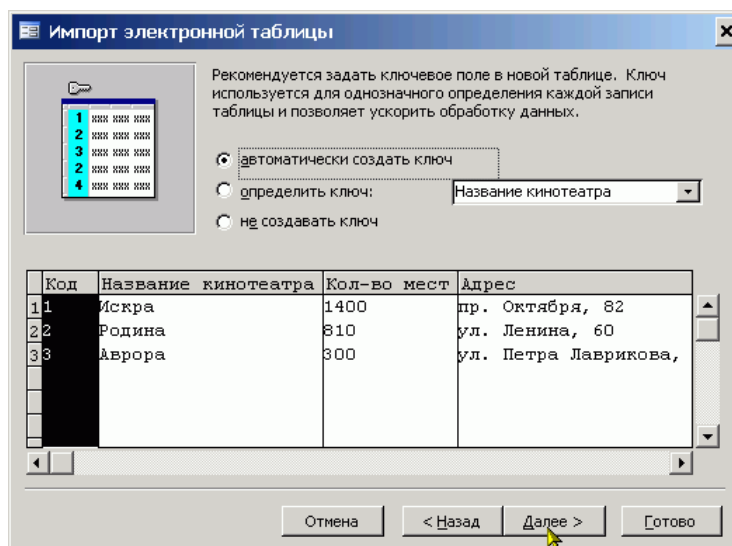


Рисунок 2.2.29 – Импорт таблиц. Шаг пятый

На последнем шаге следует задать имя таблицы. Кроме этого можно проанализировать таблицу средствами MS Access. В некоторых случаях рекомендации СУБД весьма полезны. Так MS Access может предложить разбить таблицу на несколько более простых и самостоятельно определит между ними

связи. Однако рекомендации выдаются только при достаточно большом количестве полей в новой таблице.

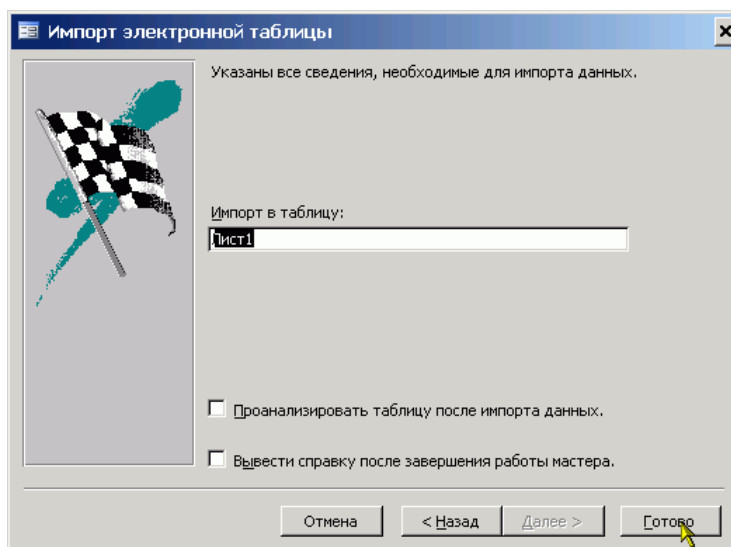


Рисунок 2.2.30 – Импорт таблиц. Шаг шестой

После нажатия кнопки [ГОТОВО], можно открыть таблицу в *режиме таблицы* для ввода записей или в *режиме конструктора* для редактирования ее структуры.

Код	Название кино	Кол-во мест	Адрес	Средняя цена билета
1	Искра	1400	пр. Октября, 82	65
2	Родина	810	ул. Ленина, 60	70
3	Аврора	300	ул. Петра Лаврикова, 12	45
*	Счетчик)			

Рисунок 2.2.31 – Импорт таблиц. Результат

Для того, чтобы импортировать файл dBASE необходимо выбрать способ создания таблицы **Импорт таблиц** и в появившемся окне выбрать тип файлов dBASE IV. Затем указать файл для импорта и нажать кнопку [Импорт]. После этого будет выведено сообщение либо о том, что импорт произведен успешно, либо о том, что импорт не возможен.

Отличие импортирования HTML-файлов или текстовых файлов заключается только в том, что не выбирается лист рабочей книги, а в остальном окна мастера полностью одинаковы.

2.2.6 Просмотр и редактирование содержания полей

В MS Access существует множество способов отобразить только требуемые данные при выполнении поиска конкретного значения, одной записи или группы записей.

Поиск и замена информации. Для поиска записей по нужному значению поля нужно выбрать команду **Найти** в пункте меню **Правка**. С помощью диалогового окна **Поиск** легко найти конкретные записи или определенные значения в полях (курсор целесообразно установить в то поле, где нужно найти информацию). При обнаружении каждого вхождения требуемого элемента выполняется перемещение по записям. Поиск может осуществляться и по всей таблице. Если нужно заменить конкретные обнаруженные при поиске значения, следует воспользоваться диалоговым окном **Замена**.

Фильтры позволяют временно изолировать и просмотреть конкретный набор записей в открытой форме или в открытом объекте в режиме таблицы.

Фильтр – это набор условий, применяемый для отбора подмножества записей из таблицы, формы или запроса. Фильтрация таблицы предусмотрена для просмотра и корректировки записей базы данных, удовлетворяющих указанным пользователем условиям отбора.

Простейшим способом задания условия отбора записей является выделение в таблице или в форме некоторого значения поля или его части.

Например, нужно найти все записи о студентах, фамилии которых начинаются с буквы «С». Для этого в режиме таблицы выделим букву «С» в поле **Фамилия** одной из записей, где фамилия начинается с этой буквы. Выполним команду меню **Фильтр** по выделенному в пункте меню **Записи**. В результате фильтрации в таблице будут отображены только записи, в которых фамилия студента начинается на букву «С». К полученному подмножеству записей всегда можно применить новые фильтры.

Запросы дают возможность работать с конкретным набором записей, которые удовлетворяют условиям, заданным для одной или нескольких таблиц базы данных. При выполнении запроса становится возможной независимая работа с этим набором записей в конкретной форме или в объекте в режиме таблицы.

5.8 Связывание таблиц

На практике встречаются и такие случаи, когда необходимо связать две разные базы данных. Например, базу данных по учету товара (Склад) и базу данных со сведениями о реализации товара (Магазин). По сути дела это две разные базы данных, но они обе используют информацию о товарах. Перенос информации из одной базы данных в другую не решит проблемы. Так как его придется производить каждый раз при изменении информации в исходной базе данных, а это связано с возможностью появления ошибок.

В таких случаях таблицы из одной базы данных могут быть вставлены в другую, более точно вставляются не сами таблицы, а ссылки на них. Для этого выбирается способ создания таблицы **Связь с таблицами**, затем в стандарт-

ном окне открытия файла выбирается файл базы данных таблицы, из которого будут вставлены в текущую базу данных и нажимается кнопка [Связь].

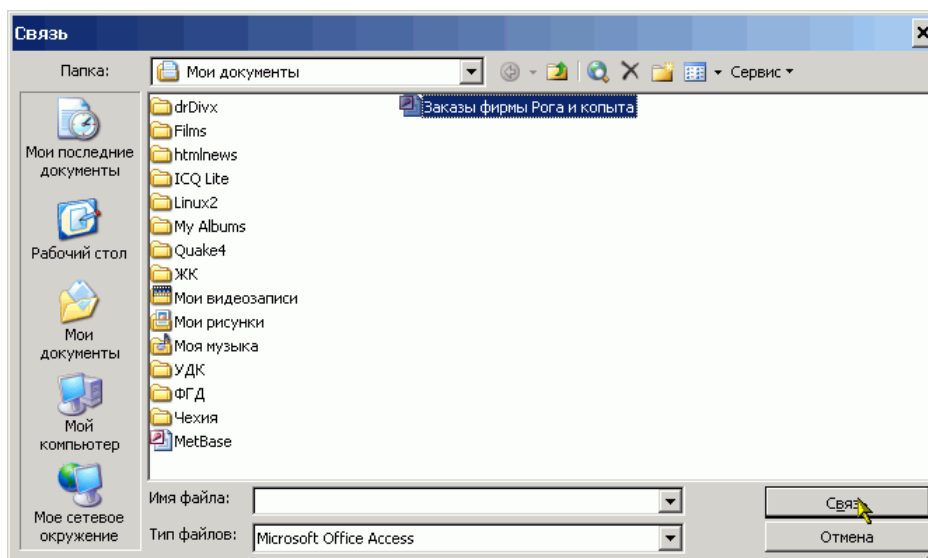


Рисунок 2.2.32 – Создание связей между таблицами. Выбор БД

После этого появляется окно с перечнем таблиц выбранной базы данных. В этом окне необходимо щелкнуть один раз левой кнопкой мыши на тех таблицах, которые будут включены в текущую базу данных и нажать кнопку [Ок].

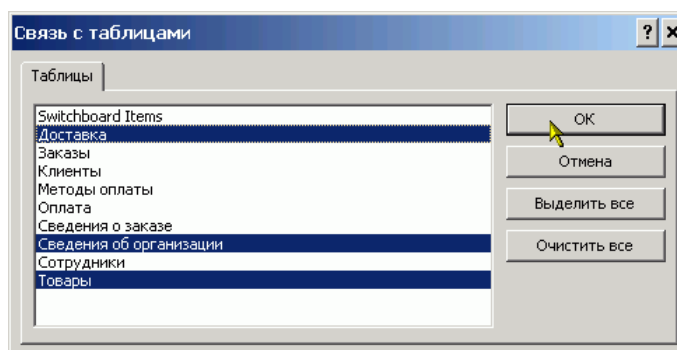


Рисунок 2.2.33 – Создание связей между таблицами. Выбор таблиц

В результате произведенных действий в списке таблиц текущей базы данных появятся новые таблицы. Обозначением того, что они являются внешними, то есть хранятся не в текущей базе данных, служит значок стрелки слева от имени таблицы.

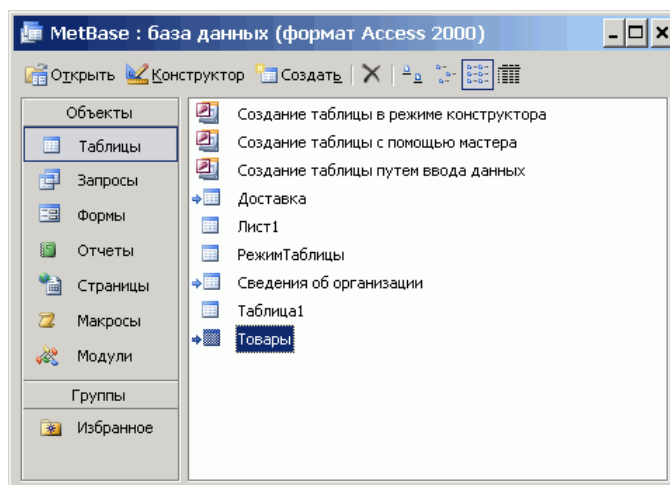


Рисунок 2.2.34 – Создание связей между таблицами. Результат

Следует отметить, что при удалении или перенесении базы данных с исходными таблицами в другое место ссылки на таблицы из текущей базы данных не исчезают, однако таблицы становятся недоступны и при попытке их просмотра будет выдано сообщение, о том, что не найден файл базы данных. То есть **MS Access** не следует за верностью ссылок на внешние таблицы и это ложится на плечи разработчиков. Известны случаи, когда не опытные пользователи, посчитав, что две базы данных с одинаковыми таблицами не нужны, удаляли один из файлов, что приводило к потере значительной части информации. Поэтому при использовании ссылок на таблицы все файлы баз данных, на таблицы которых ссылаются извне, следует защитить от удаления и/или разместить в труднодоступных для обычных пользователей местах.

2.2.9 Задания к лабораторной работе

1. Создать таблицу по индивидуальному заданию.

2.2.10 Контрольные вопросы

1. Для чего создаются межтабличные связи в схеме данных?
2. Какую информацию содержит таблица, в которой нет записей?
3. Что определяет структуру таблицы?
4. На основе, каких полей создаются межтабличные связи?
5. Для чего создаются межтабличные связи при объединении таблиц и создании схем данных?
6. В каком режиме возможно изменение структуры таблицы?
7. В каком режиме производится наполнение таблицы данными?

2.3 Лабораторная работа № 3

Тема: Формирование запросов

Содержание:

- 2.3.1 Понятие запроса к БД
- 2.3.2 Создание запроса в режиме конструктора
- 2.3.3 Технология создания запросов
- 2.3.4 Перекрестные запросы
- 2.3.5 Задания к лабораторной работе
- 2.3.6 Контрольные вопросы

2.3.1 Понятие запроса к БД

После создания таблицы и ввода в них данных, одними из самых важных объектов базы данных становятся «Запросы». С их помощью можно выбрать из базы данных определенную информацию и упорядочить ее для использования в отчете или для просмотра на экране в форме или таблице.

Запрос позволяет выбрать необходимые данные из одной или нескольких взаимосвязанных таблиц, произвести вычисления и получить результат в виде таблицы. Через запрос можно производить обновление данных в таблицах, добавление и удаление записей. С помощью запроса можно выполнить следующие виды обработки данных:

- выбрать записи, удовлетворяющие условиям отбора;
- включить в результирующую таблицу запроса заданные пользователем поля;
- произвести вычисления в каждой из полученных записей;
- сгруппировать записи с одинаковыми значениями в одном или нескольких полях для выполнения над ними групповых функций;
- произвести обновление полей в выбранном подмножестве записей;
- создать новую таблицу базы данных, используя данные из существующих таблиц;
- удалить выбранное подмножество записей из таблицы базы данных;
- добавить выбранное подмножество записей в другую таблицу.

В зависимости от задачи, которую решает запрос, различают: запрос на выборку; запрос на создание таблиц; запросы на обновление, добавление, удаление.

Среди этих трех типов особая роль отводится запросу на выборку, так как на его основе строятся запросы других видов.

После нажатия на кнопку "Создать" в разделе "Запросы" окна базы данных появится несколько возможностей по созданию запросов к базе данных.

2.3.2 Создание запроса в режиме конструктора

После выбора конструктора появляется диалоговое окно **Добавление таблицы**, в котором нужно выбрать те таблицы и запросы, на основе которых будет строиться новый запрос. Выбор осуществляется выделением имени таблицы и нажатием кнопки **Добавить**.

После выбора нужных таблиц, можно перейти к конструированию самого запроса. Рассмотрим, какие возможности для этого имеет **MS Access**.

Окно конструктора запросов разделено на две панели. Верхняя панель содержит схему данных запроса, которая включает выбранные для данного запроса таблицы. Таблицы представлены списками полей. Нижняя панель является бланком запроса по образцу, который нужно заполнить.

Схема данных запроса. В окне запроса отображаются выбранные таблицы и связи между ними, имеющиеся в схеме данных базы. Кроме того, Access устанавливает связи для объединения таблиц, если таблицы имеют поля с одинаковыми полями и типами данных, даже когда они не были определены в схеме данных.

Бланк запроса по образцу. Бланк запроса по образцу представлен в виде таблицы в нижней панели окна запроса. До формирования запроса эта таблица пуста. Каждый столбец относится к одному полю, с которым нужно работать в запросе. Поля могут использоваться для включения их в результат запроса, для задания сортировки по ним, а также для условий отбора записей. При заполнении бланка запроса необходимо:

- в строку **поле** включить имена полей, используемых в запросе;
- в строке **вывод на экран** отметить поля, которые должны быть включены в результирующую таблицу;
- в строке **условие отбора** задать условия отбора записей;
- в строке **сортировка** выбрать порядок сортировки записей результата.

Поля бланка запроса. Для включения нужных полей из таблиц в соответствующие столбцы запроса можно воспользоваться следующими приемами:

В первой строке бланка запроса **Поле**, щелчком мыши вызвать появление кнопки списка и, открыв список, выбрать из него нужное поле. Список содержит все поля таблицы, представленные в бланке запроса. Перетащить нужное поле из списка полей таблицы в схему данных запроса в первую строку бланка запроса.

В списке полей каждой таблицы на первом месте всегда стоит знак звездочки (*), который означает **все поля таблицы**. Этот пункт выбирается, если в запрос включаются все поля.

В MS Access всегда можно добавить нужные или удалить ненужные поля запроса. Для добавления поля в бланк запроса надо перетащить его с помощью мыши из таблицы в схему данных в нужное место бланка. Все столбцы полей справа от него передвинутся на один столбец вправо. Для удаления ненужного столбца выделите его и нажмите на клавишу [Delete]

2.3.3 Технология создания запросов

Условие отбора записей. Условия отбора записей могут задаваться для одного или нескольких полей в соответствующей строке бланка запроса. Условием отбора является выражение, которое состоит из операторов сравнения и операндов, используемых для сравнения. В качестве операндов выражения могут использоваться: литералы и константы.

Литералом называют значение в явном представлении, например, число, строковое значение или дату. Примерами значений в явном представлении могут служить "Москва", 100 и #1-января-94#. Даты необходимо заключать в символы (#), а строковые значения в прямые кавычки (").

Константа представляет не изменяющееся значение. True, False, Истина, Ложь и Null являются примерами констант, автоматически определяемых в Microsoft Access.

В выражении условия отбора допускается использование операторов сравнения и логических операторов: *Between...And*, <, >, <>, <=, >=, *Like*, *In*, *And*, *Or*, *Not*. Разрешено использование подстановочных знаков.

Подстановочными знаками являются: знак звездочки (*), вопросительный знак (?), знак номера (#), восклицательный знак (!), дефис (-) и квадратные скобки ([]). Эти знаки используются в запросах и выражениях, чтобы включить все записи, имена файлов или конкретные названия, начинающиеся с определенных символов или соответствующие определенному образцу.

Оператор *Between...And* определяет принадлежность значения выражения указанному диапазону.

Оператор *Like*, позволяет использовать образцы, использующие символы шаблона. В следующем примере возвращаются данные, начинающиеся с буквы "P", за которой следуют любая буква от "A" до "D" и три цифры:

Like "P[A-D]###"

Оператор *In* проверяет, совпадает ли значение выражения с одним из элементов указанного списка, который задается в круглых скобках.

Оператор *And* требует одновременного выполнения всех выражений, которые он соединяет.

Оператор *OR* требует выполнения хотя бы одного из тех выражений, которые он соединяет.

Оператор *Not* требует невыполнения того выражения, перед которым он стоит.

Еще одним встроенным элементом поиска нужной информации в **Access** являются функции. Функция возвращает значение, которое является результатом расчетов или выполнения других операций. В **Microsoft Access** определен ряд встроенных функций, например:

- функция *Sum* возвращает сумму набора значений поля;
- функция *Avg* вычисляет арифметическое среднее набора чисел, содержащихся в указанном поле запроса;
- функция *Count* вычисляет количество записей, возвращаемых запросом.

Наиболее простой вызов этих функций – это выбор в пункте меню «Вид» команды

«Групповые Операции» или нажатие кнопки с изображением *S* на панели инструментов. После этого в бланке запроса появится еще одна строка «Групповые операции», в которой для каждого поля можно выбрать нужную операцию.

После того как запрос создан, его надо выполнить. Это можно сделать двумя способами:

- в меню «запрос» выбрать пункт «запуск»;
- на панели инструментов нажать кнопку с изображением «!»

Чтобы вернуться обратно в режим конструктора можно воспользоваться пунктом меню «Вид» и выбрать «Конструктор». Точно так же можно вернуться в табличный режим. Если необходимо сохранить запрос в виде таблицы, то надо воспользоваться командой «Сохранить» в пункте меню «Файл».

2.3.4 *Перекрестные запросы*

В *перекрестном запросе* отображаются результаты статистических расчетов (такие, как суммы, количество записей, средние значения), выполненных по данным из одного поля таблицы. Эти результаты группируются по двум наборам данных, один из которых расположен в левом столбце таблицы, а второй - в верхней строке. Например, нам надо узнать средний стаж работы ассистентов, доцентов и профессоров на разных кафедрах (на основе таблицы

Преподаватели). Перекрестный запрос позволит легко решить эту задачу, создав таблицу, в которой заголовками строк будут служить должности, заголовками столбцов – названия кафедр, а в ячейках будут рассчитаны средние значения стажа преподавателей.

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

Запрос на удаление удаляет группу записей, удовлетворяющих заданным условиям, из одной или нескольких таблиц. С помощью запроса на удаление можно удалять только всю запись, а не отдельные поля внутри нее.

Запрос на обновление записей вносит общие изменения в группу записей одной или нескольких таблиц. Например, на 10 процентов увеличилась заработная плата ассистентов, Запрос на обновление позволит быстро внести эти изменения в таблицу *Преподаватели*.

Запрос на добавление добавляет группу записей из одной или нескольких таблиц в конец одной или нескольких таблиц. Например, появилось несколько новых преподавателей, а также база данных, содержащая сведения о них. Чтобы не вводить все данные вручную, их можно добавить в таблицу *Преподаватели*.

Запрос на создание таблицы создает новую таблицу на основе всех или части данных из одной или нескольких таблиц. Например, на основе таблицы *Преподаватели* можно создать новую таблицу, содержащую данные только о профессорах.

Запрос SQL – это запрос, создаваемый при помощи инструкций SQL. Этот тип запросов довольно сложен для начинающих пользователей и используется обычно опытными пользователями, имеющими навыки программирования и общения с серверами баз данных, Из-за сложности и специфики рассматривать запрос SQL в данной главе не будем.

2.3.5 Задания к лабораторной работе (примерные)

1. Разработайте запрос с параметрами о студентах заданной группы, в котором при вводе в окно параметров номера группы (в примере это 151 или 152) на экран должен выводиться состав этой группы.
2. Создайте запрос, в котором выводятся оценки студентов заданной группы по заданной дисциплине.
3. Создайте перекрестный запрос, в результате которого создастся выборка, отражающая средний балл по дисциплинам в группах.
4. Разработайте запрос на увеличение на 10% заработной платы тех преподавателей, кто получает менее 500 руб.

5. Создайте запрос на удаление отчисленных студентов.
6. Разработайте запрос на создание базы данных отличников.
7. Для всех созданных вами запросов разработайте формы.

2.3.6 Контрольные вопросы

1. Назовите три основных свойства запросов, используемых пользователями при работе с большими базами данных?
2. Дайте определение понятию «запрос».
3. Какую роль играют связи между таблицами при выполнении запроса?
4. Поясните технологию выполнения запроса с логической связкой «AND»?
5. Поясните технологию выполнения запроса с логической связкой «OR»?
6. Поясните технологию выполнения запроса с отрицанием «NOT»?
7. Что означает запрос к таблице о клиентах: «Фамилия = *ов»?

2.4 Лабораторная работа № 4

Тема: Проектирование форм и работа с ними

Содержание:

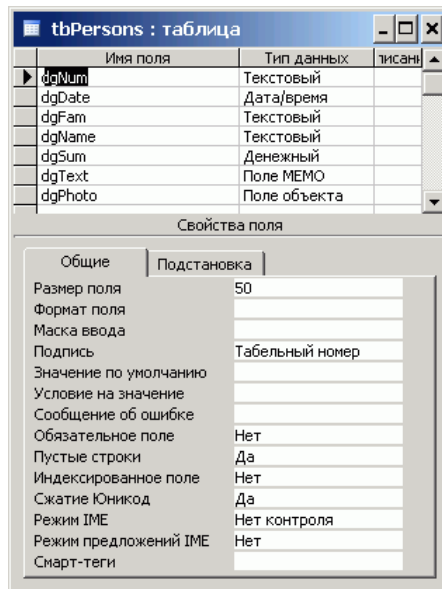
- 2.4.1 Способы создания форм
- 2.4.2 Создание форм с помощью Мастера
- 2.4.4 Автоформы
- 2.4.5 Диаграмма
- 2.4.6 Задания к лабораторной работе
- 2.4.7 Контрольные вопросы

2.4.1 Способы создания форм

После создания таблицы в режиме конструктора, ее тут же можно открыть и начать вводить данные. Однако, табличный вид данных не всегда удобен, например, когда в таблице присутствуют поля типа Поле объекта OLE, то есть графическая, аудио или видео информация. В режиме таблицы в поле отображается только название типа вставленного объекта, например, *Документ Microsoft Office Word*, но не сам объект. Даже при не большом количестве полей трудно добиться того, чтобы все поля таблицы отображались на экране. Кроме того, в режиме таблицы невозможно изменить такие параметры как шрифт, начертание и цвет шрифта. Поэтому разработчиками создаются специальные экранные формы для ввода данных. Формы реализуют интерфейс пользователя базы данных. От их грамотной разработки зависит удобство использования и логичность всего приложения.

Также как и любой объект базы данных **MS Access** форму можно создать автоматически при помощи мастера или вручную в режиме конструктора. А также при создании можно выбрать один из уже готовых шаблонов, которые называются *Автоформы* (см. рисунок). Во всех случаях, перед тем как создавать форму необходимо выбрать таблицу или запрос, поля, которых будут присутствовать на форме. Для этого в окне, приведенном на рисунке в нижней строке нажимаем кнопку (▼) и, в появившемся списке имеющихся в базе данных таблиц и запросов, выбираем объект для создания формы.

Прежде чем перейти к описанию способов создания экранных форм в **MS Access**, представим структуру и вид образца таблицы, на основе которой мы и будем создавать форму. Их можно увидеть на рисунках представленных ниже.



Табельный ном	Дата рождения	Фамилия	Имя	Оклад	Приказы	Фотография
1234	12.01.1965	Иванов	Иван	5 700,00р.		
4321	10.03.1975	Петров	Петр	3 210,00р.		
*				0,00р.		

Запись: 2 из 2

Рисунок 2.4.1 – Исходная таблица

2.4.2 Создание форм с помощью Мастера

Рассмотрим создание формы в режиме мастера. Переходим на вкладку Формы, нажимаем кнопку Создать. В появившемся окне выбираем пункт Мастер форм. Выбираем таблицу, поля которой будут присутствовать на форме (в нашем случае это таблица tbPerson) и нажимаем кнопку <Ok>.

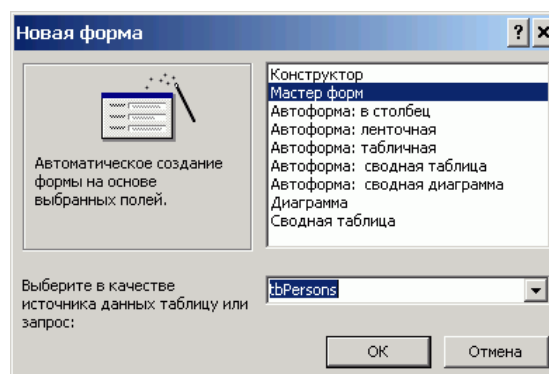


Рисунок 2.4.2 – Создание формы с помощью Мастера. Шаг первый

На экране появляется окно первого шага мастера форм. На этом шаге мы определяем поля, которые будут присутствовать на форме. Для того, экранная форма содержала поле, его необходимо выбрать и нажать кнопку с изображением знака «больше» (>). Если требуется, чтобы все поля

таблицы/запроса отображались на форме, то необходимо нажать кнопку, на которой изображено два знака «больше» (>>). Кнопки с изображением знака «меньше» (< и <<) используются для удаления полей из списка отображаемых на форме.

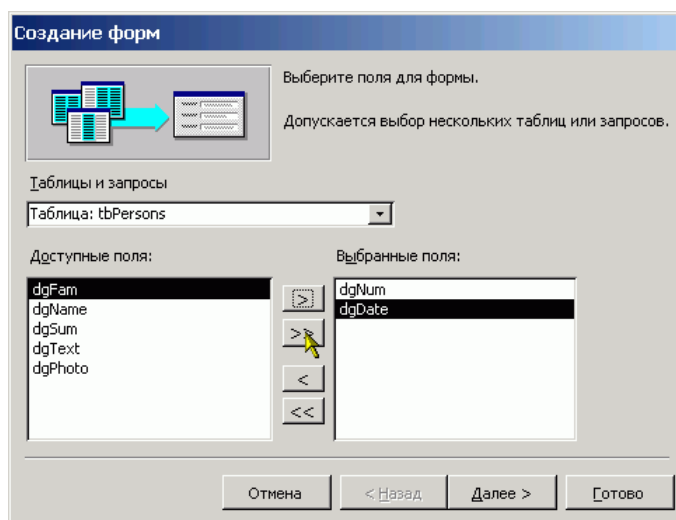


Рисунок 2.4.3 – Создание формы с помощью Мастера. Шаг второй

В тех случаях, когда на форме должны отображаться поля из нескольких таблиц необходимо, выбрав поля для отображения на форме из первой таблицы, в первой строке окна мастера форм выбрать вторую таблицу (или запрос) и из появившегося списка ее полей выбрать те, которые будут отображаться на форме, затем также выбрать третью таблицу и указать ее поля для отображения и т. д. Следует помнить, что поля из разных таблиц могут присутствовать на форме только в том случае, если таблицы являются связанными.

После определения списка полей формы нажимаем кнопку [Далее]. На экране появляется окно второго шага мастера форм. В этом окне мы должны выбрать вид нашей формы. По умолчанию обычно выбран Ленточный вид. Выбираем любой из пунктов и нажимаем кнопку [Далее].

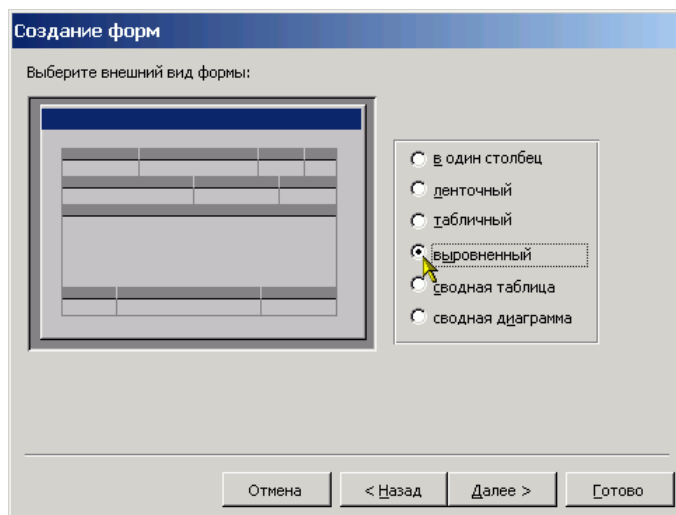


Рисунок 2.4.4 – Создание формы с помощью Мастера. Шаг третий

На третьем шаге необходимо выбрать стиль оформления формы. При выборе любого из перечисленных стилей справа показывается пример оформления. После выбора стиля оформления нажимаем кнопку [Далее].

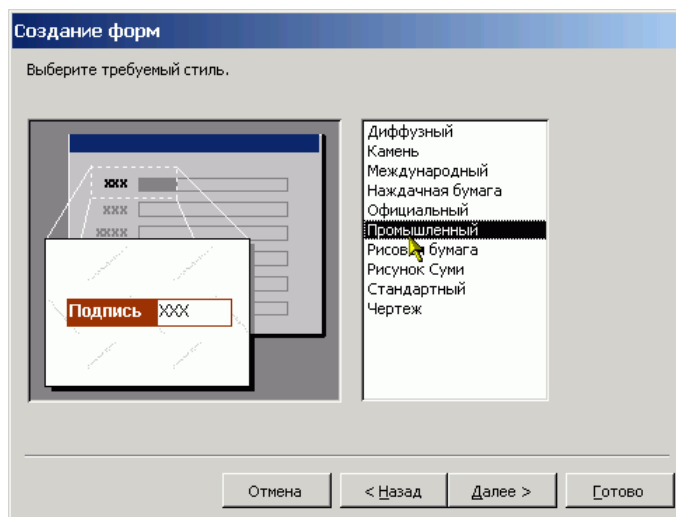


Рисунок 2.4.5 – Создание формы с помощью Мастера. Шаг четвертый

На последнем шаге нам предлагают ввести имя формы, а также указать, что делать с формой далее. Если выбран пункт **Изменить макет формы**, то форма будет открыта в режиме конструктора. После задания имени формы и выбора дальнейшего действия нажимаем кнопку [Готово].

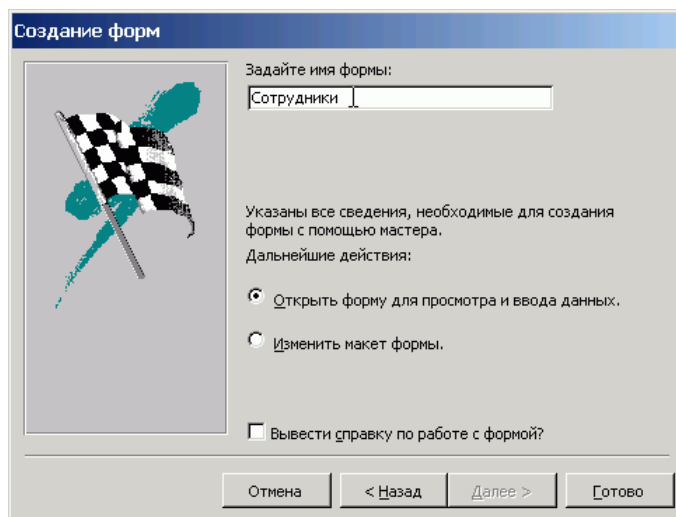


Рисунок 2.4.6 – Создание формы с помощью Мастера. Шаг шестой

На этом работа мастера заканчивается, и получается готовая форма для ввода данных. Как видите, не смотря на то, что имена всех полей таблицы записаны при помощи латинских букв, на форме отображается текст из свойства «Подпись».

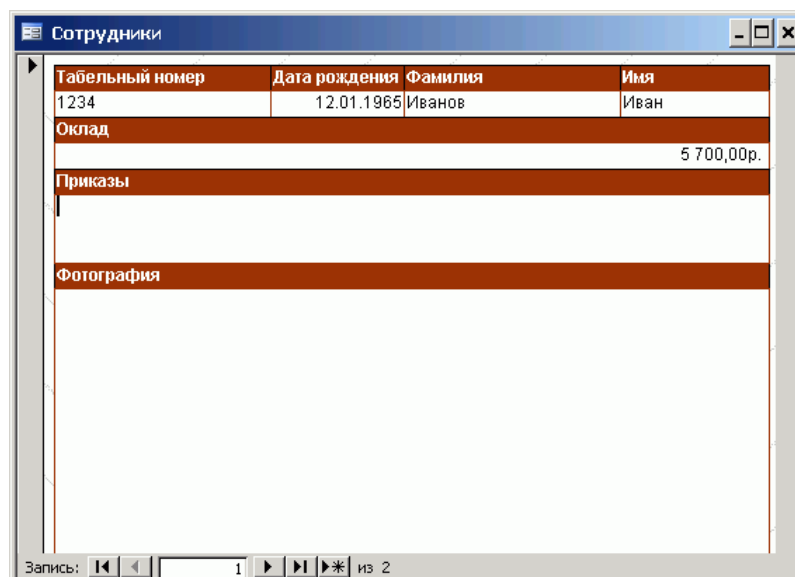


Рисунок 2.4.7 –Создание формы с помощью Мастера. Результат

2.4.4 Автоформы

Как говорилось ранее можно также использовать и автоформы. Автоформы это средство быстрого создания форм без дополнительных вопросов. При их создании используются все поля, имеющиеся в указанной таблице или запросе. И результат будет таким, как если бы при использовании мастера вы выбрали все поля одной таблицы и затем выбрали соответствующий вид:

в один столбец – Автоформа: в столбец;

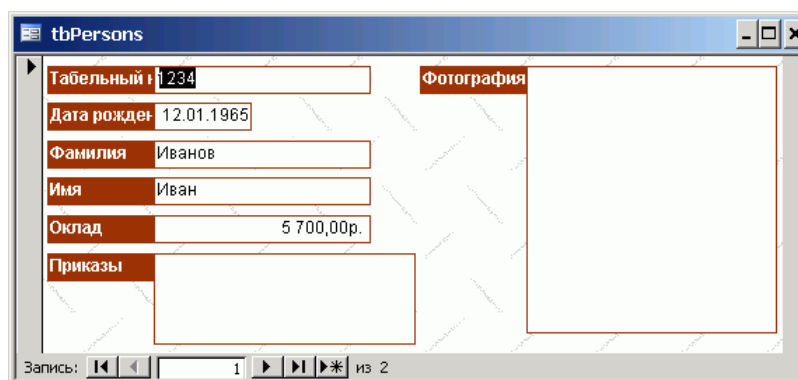


Рисунок 2.4.8 – Автоформа: в столб

ленточный – Автоформа: ленточная

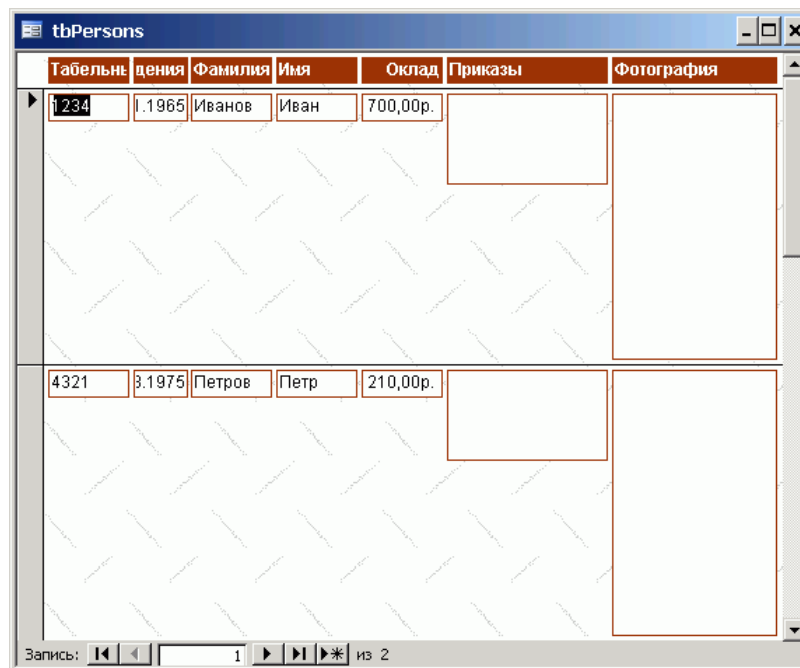


Рисунок 2.4.9 – Автоформа: ленточная

табличный – Автоформа: табличная. Приводить вид этой формы не имеет смысла, так как он идентичен виду обычной таблицы приведенному в начале этого документа.

2.4.5 Диаграмма

Как известно, человек лучше воспринимает визуальную информацию, нежели текстовую. Нет ничего лучше для сравнения, каких либо параметров, чем диаграмма, отображающая значения каждого из них. В Access можно создать форму, на которой данные будут представлены в виде диаграммы. Для этого, необходимо в окне выбора способа создания формы выбрать пункт **Диаграмма**.

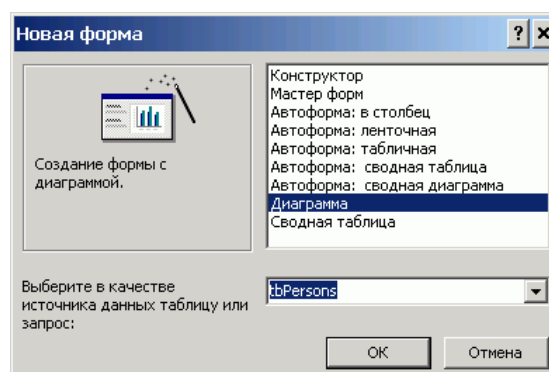


Рисунок 2.4.10 – Создание формы. Диаграмма

После нажатия кнопки **Ок** на экране появляется окно первого шага *мастера диаграмм*. В этом окне необходимо выбрать поля, значения которых будут участвовать в построении диаграммы. Выбор осуществляется стандартно при

помощи кнопочек с изображением знаков «больше» (добавить поле) и меньше» (удалить поле). Кнопки с двойными знаками позволяют добавить/убрать все поля таблицы либо запроса.

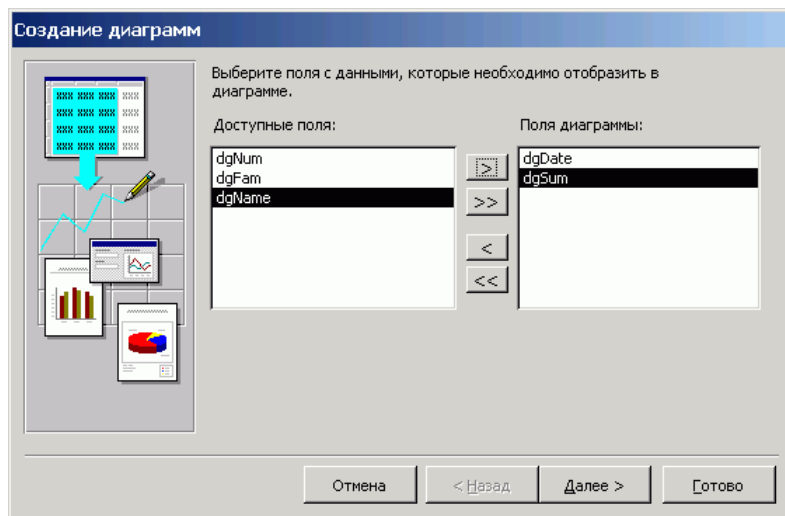


Рисунок 2.4.11 – Создание диаграммы. Шаг первый

На втором шаге мастера диаграмм необходимо выбрать тип диаграммы для отображения данных. Разработчик должен просто выбрать тип, щелкнув на кнопке с его изображением и нажать кнопку Далее.

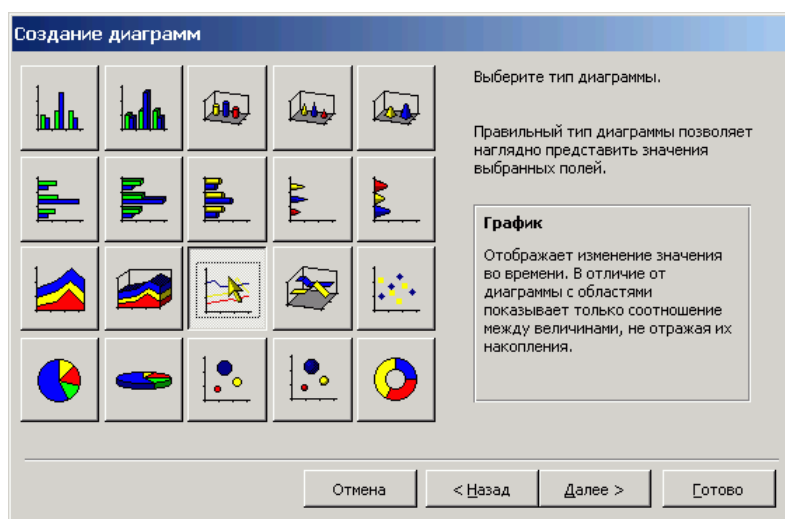


Рисунок 2.4.12 – Создание диаграммы. Шаг второй

На следующем шаге необходимо определить расположение данных. Для этого необходимо перетащить прямоугольники с названиями полей на соответствующие оси и, если необходимо, в область данных. На приведенном рисунке поле, значения поля «Дата рождения» типа Дата/время, отображается по оси категорий (ось X), а значения поля «Оклад» по оси значений (Ось Y). Такое представление данных не имеет никакого смысла. Именно поэтому Диаграммы строятся обычно на основании запросов. Следует все же отметить, что при построении нашей диаграммы суммы окладов людей, родившихся в

одном и том же месяце, то есть оклады (значения), соответствующие одному и тому же значению на оси категорий суммируются.

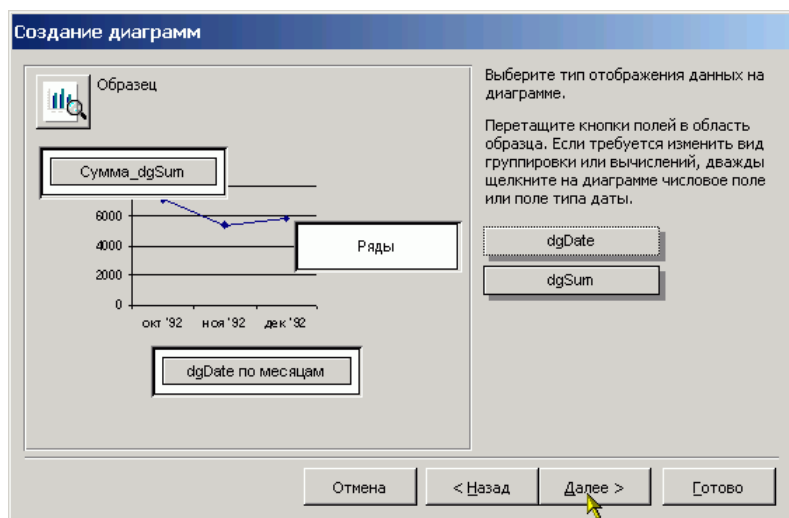


Рисунок 2.4.13 – Создание диаграммы. Шаг третий

На последнем шаге необходимо задать название диаграммы и выбрать вид дальнейших действий.

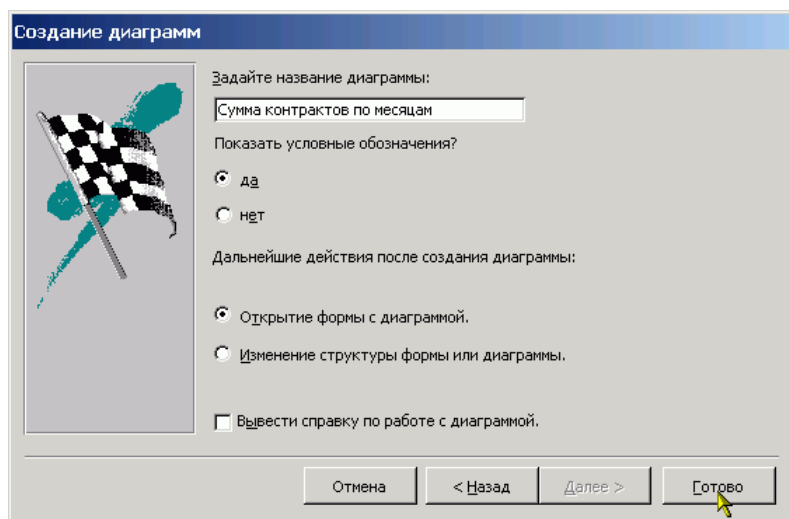


Рисунок 2.4.14 – Создание диаграммы. Шаг четвертый

Если все параметры оставить без изменений, то есть, так как на приведенном выше рисунке, то в результате можно получить следующую диаграмму.

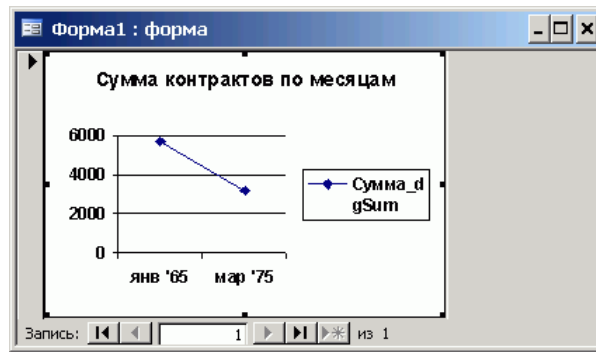


Рисунок 2.4.15 – Создание диаграммы. Шаг пятый

2.4.8 Задания к лабораторной работе

8. Выполнить индивидуальное задание.

2.4.7 Контрольные вопросы

1. Дайте определение понятию «форма».
2. Для каких целей создаются экранные формы?
3. Перечислите способы создания форм. В чем их суть.

2.5 Лабораторная работа № 5

Тема: Отчет как объект БД

Содержание

- 2.5.1 Понятие отчета
- 2.5.2 Создание отчета с помощью мастера
- 2.5.3 Задания к лабораторной работе
- 2.5.4 Контрольные вопросы

Отчеты, это специальным образом оформленные документы для вывода на печать. От форм отчеты отличаются, наверное, только способом отображения данных: формы отображают данные на экране, а отчеты выводятся на принтер. Поэтому и создание отчета во многом напоминает создание формы, но есть и отличия. Формы, как правило, создаются для редактирования и отображения данных таблиц, а отчеты чаще всего создаются на основании запросов. Например, на основании данных о работниках предприятия, занесенных в таблицы базы данных, выполняется запрос на расчет зарплаты и выдается отчет – платежная ведомость. Или на основании данных о закупке печатается заполненный бланк накладной и счета-фактуры.

Поскольку отчет это представление данных для печати на бумаге, то и начинать его разработку лучше всего на бумаге. Вы должны ясно представить шаблон и определить структуру отчета. Проще всего это сделать, если рисовать шаблон отчета карандашом на бумаге. Конечно, если у вас уже есть шаблон, например, образец бланка и т. д., то можно сразу приступить к разработке отчета в **MS Access**, но такое бывает редко. Поэтому, еще раз настоятельно рекомендую составить отчет на бумаге и затем уже переносить его в электронную форму.

Отчеты, как и любой другой элемент базы данных **MS Access** можно создать вручную и автоматически. При автоматическом создании можно воспользоваться автоотчетами или запустить мастер создания отчета. Автоотчет – это самый простой вид отчета, Мастер позволяет получить более сложное форматирование отчета, а ручной способ создания (в режиме конструктора) позволяет получить отчет в том виде, который требуется. Следует помнить о том, что если в операционной системе не установлен ни один принтер, то работа с отчетами не возможна.

На данном уроке рассматривается только использование Автоотчетов и мастеров. Работа в режиме конструктора описана в следующем уроке.

Для того чтобы создать отчет необходимо перейти на вкладку **Отчеты**. После этого необходимо нажать кнопку **Создать**. Появится окно весьма похожее на окно создания форм. В нем перечислены различные способы создания от-

четов. Выбираем **Мастер отчетов**. Затем в нижней строке из списка выберем таблицу или отчет, на основании которых нужно будет создать отчет. Выберем запрос и нажмем кнопку **[Ок]**.

На экране появится окно уже знакомое нам по мастеру форм. В этом окне необходимо выбрать поля, которые будут участвовать в построении отчета. Также, как и при построении форм можно выбирать поля из нескольких таблиц или запросов. Но не следует забывать, что они должны быть связаны между собой.

Выбрав поля и нажав кнопку **[Далее]**. На экране появляется окно, в котором мы выбираем уровни группировки данных. То есть определяем по каким полям и как будет производиться группировка данных. Так на приведенном рисунке группировка осуществляется по одному полю (dgClients). Можно добавить до шести уровней группировки. Однако увлекаться группировкой не следует, так как в случае многих уровней группировки большая часть листа отчета будет пуста.

Определив уровни группировки, нажимаем кнопку **[Далее]**. Во вновь появившемся окне можно настроить сортировку данных отчета. Как видно из приведенного в окне текста допускается задание до четырех критериев сортировки. Если нужна сортировка, то настраиваем критерии, если не нужна, то просто нажимаем кнопку **[Далее]**.

Следующее окно мастера отчетов позволяет задать макет оформления отчета и ориентацию бумаги. При этом в правой части окна показывается пример выбранного оформления. После задания необходимых параметров нажимаем кнопку **[Далее]**.

В следующем окне мастера отчетов предлагается выбрать стиль отчета. В понятие стиля входят настройки шрифта и стилей (заголовков, примечаний и т. д.). В правом окне показывается пример стиля. Выбрав понравившийся стиль и нажимаем кнопку **[Далее]**.

На экране появляется последнее окно мастера отчетов. В нем предлагается дать имя отчету, а также выбрать дальнейшее действие. Это окно также уже знакомо нам по работе с мастером форм. После задания имени отчета и выбора необходимого действия нажимаем кнопку **[Готово]**. Отчет готов.

При использовании автоотчетов готовый отчет формируется сразу же после того, как будет выбрана таблица и нажата кнопка **[Ок]**. Но, как и в случае форм элементы отчета могут быть расположены не совсем удачно. На приведенном рисунке видно, что подписи полей обрезаются, так как поля для вывода информации расположены слишком близко к надписям. Поэтому еще раз повторим прописную истину: наилучший результат получается при ручной разработке приложения баз данных.

2.4.6 Задания к лабораторной работе

1. Выполнить индивидуальное задание.

3.4.7 Контрольные вопросы

1. Дайте определение понятию «отчет».
2. Для каких целей создаются отчеты?
3. Перечислите способы создания отчетов.
4. Редактирование отчетов.

Создание базы данных (по индивидуальным заданиям) – 22 часа.

Первый этап

Принимается решение о необходимости и целесообразности автоматизации обработки информации на объекте управления. Определяются цели создания БД, их функции, информация, которая должна содержаться в них.

На объекте управления, для которого разрабатывается база данных, уточняется и составляется полный список основных задач с позиции системного подхода, а также с учетом перспектив развития объекта управления и появления новых задач. Для каждой задачи заполняется рабочий бланк, в котором ей присваивается номер, даются название, краткое описание и указывается связь с другими задачами.

Второй этап

Последовательность выполнения задачи. Основные задачи (например, учет материалов, учет готовой продукции) объединяются в тематические группы; внутри каждой из них задачи располагаются в порядке их выполнения. Последовательность выполнения задач можно отображать графически. Нужно иметь в виду, что некоторые задачи имеют межгрупповые связи.

Определяется и утверждается количество таблиц, в которых будет содержаться база данных. Рекомендуются предварительно разработать таблицы на бумаге.

Третий этап

Анализ данных. Определение структуры каждой таблицы: перечень полей таблицы и последовательность их размещения. Каждая таблица должна содержать информацию по отдельной теме, а каждое поле – отдельные данные по теме таблицы.

Для каждой задачи составляется подробный перечень всех элементов данных (реквизитов), которые нужны для ее решения. Указывается назначение каждого элемента.

Четвертый этап

Определение структуры данных. Элементы (реквизиты) данных каждой задачи после предварительного анализа нужно упорядочить по объектам и создать эффективный и гибкий способ сохранения информации в базе данных. Это достигается путем нормализации таблицы. Следует задать индивидуальное значение каждому полю в каждой записи таблицы. Каждая таблица должна содержать поле (несколько полей), указывающее индивидуальное значение каждой записи в таблице – *основной ключ*.

Пятый этап

Разработка макета базы данных (задачи) в целом и ее объектов (форм, запросов, отчетов, макросов и др.). Необходимо определить связи между таблицами. После определения структуры таблиц и ключевых полей в них надо установить связи между таблицами, которые находятся в логико-информационных отношениях.

Шестой этап

Создание базы данных (приложения). Для полной автоматизации отношений с базой данных пишутся нужные процедуры, специальные формы связи для перехода между задачами, разрабатываются специальные меню. Непосредственно перед заполнением таблиц информацией необходимо еще раз проверить, проанализировать их структуру и внести необходимые уточнения, изменения – с помощью контрольного примера.

Седьмой этап

Тестирование и усовершенствование созданного. Проверяется функционирование базы данных в возможных режимах, а также работа макросов и процедур, выявляются и исправляются ошибки.

Добавляются и создаются другие объекты базы данных (таблицы, запросы, формы, отчеты, макросы, модули). После утверждения структуры таблиц базы данных и их взаимосвязей можно приступить к заполнению их информацией, на основе которой MS Access создаст любые объекты (запросы, формы, отчеты, макросы, модули).

Эксплуатация и усовершенствование

Руководствуясь итогами функционирования (эксплуатации) БД, производится ее усовершенствование

Перечень примерных тем для создания БД

1. Разработка БД поузловой обработки швейных изделий (по видам)
2. Разработка БД поузловой обработки трикотажных изделий (по видам)
3. Разработка БД трикотажных переплетений
4. Разработка БД вязального оборудования (основовязальные одно- и двухфонтурные; плосковязальные одно- и двухфонтурные; кругловязальные малого и большого диаметра; хлопчатобумажные)
5. Разработка БД швейного оборудования (универсальные; зигзагообразного стежка; обметочные; стачивающе-обметочные; петельные; пуговичные; закрепочные; вышивальные; подшивочные; стегальные; выметочные; полуавтоматы)

4 ФОНД ТЕСТОВЫХ И КОНТРОЛЬНЫХ ЗАДАНИЙ

4.1 Фонд тестовых заданий

Раздел: Базы данных

1. База данных – это:

- 1) совокупность данных, организованных по определенным правилам;
- 2) совокупность программ для хранения и обработки больших массивов информации;
- 3) интерфейс, поддерживающий наполнение и манипулирование данными;
- 4) определенная совокупность информации.

Один правильный ответ

Правильный ответ – а)

2. Наиболее распространенными в практике являются:

- 1) распределенные базы данных;
- 2) иерархические базы данных;
- 3) сетевые базы данных;
- 4) реляционные базы данных.

Один правильный ответ

Правильный ответ – г)

3. Наиболее точным аналогом реляционной базы данных может служить:

- 1) неупорядоченное множество данных;
- 2) вектор;
- 3) генеалогическое дерево;
- 4) двумерная таблица.

Один правильный ответ

Правильный ответ – г)

4. Таблицы в базах данных предназначены:

- 1) для хранения данных базы;
- 2) для отбора и обработки данных базы;
- 3) для ввода данных базы и их просмотра;
- 4) для автоматического выполнения группы команд;
- 5) для выполнения сложных программных действий.

Один правильный ответ

Правильный ответ – а)

5. Что из перечисленного не является объектом MS Access:
- 1) модули;
 - 2) таблицы;
 - 3) макросы;
 - 4) ключи;
 - 5) формы;
 - 6) отчеты;
 - 7) запросы.

Один правильный ответ

Правильный ответ – г)

6. Для чего предназначены запросы:
- 1) для хранения данных базы;
 - 2) для отбора и обработки данных базы;
 - 3) для ввода данных базы и их просмотра;
 - 4) для автоматического выполнения группы команд;
 - 5) для выполнения сложных программных действий;
 - 6) для вывода обработанных данных базы на принтер.

Один правильный ответ

Правильный ответ – б)

7. Для чего предназначены формы:
- 1) для хранения данных базы;
 - 2) для отбора и обработки данных базы;
 - 3) для ввода данных базы и их просмотра;
 - 4) для автоматического выполнения группы команд;
 - 5) для выполнения сложных программных действий.

Один правильный ответ

Правильный ответ – в)

8. Для чего предназначены модули:
- 1) для хранения данных базы;
 - 2) для отбора и обработки данных базы;
 - 3) для ввода данных базы и их просмотра;
 - 4) для автоматического выполнения группы команд;
 - 5) для выполнения сложных программных действий.

Один правильный ответ

Правильный ответ – д)

9. Для чего предназначены макросы:
- 1) для хранения данных базы;
 - 2) для отбора и обработки данных базы;

- 3) для ввода данных базы и их просмотра;
- 4) для автоматического выполнения группы команд;
- 5) для выполнения сложных программных действий.

Один правильный ответ

Правильный ответ – г)

10. В каком режиме работает с базой данных пользователь:
- 1) в проектировочном;
 - 2) в любительском;
 - 3) в заданном;
 - 4) в эксплуатационном?

Один правильный ответ

Правильный ответ – г)

11. В каком диалоговом окне создают связи между полями таблиц базы данных:
- 1) таблица связей;
 - 2) схема связей;
 - 3) схема данных;
 - 4) таблица данных?

Один правильный ответ

Правильный ответ – в)

12. Почему при закрытии таблицы программа Access не предлагает выполнить сохранение внесенных данных:
- 1) недоработка программы;
 - 2) потому что данные сохраняются сразу после ввода в таблицу;
 - 3) потому что данные сохраняются только после закрытия всей базы данных?

Один правильный ответ

Правильный ответ – б)

13. Без каких объектов не может существовать база данных:
- 1) без модулей;
 - 2) без отчетов;
 - 3) без таблиц;
 - 4) без форм;
 - 5) без макросов;
 - 6) без запросов?

Один правильный ответ

Правильный ответ – в)

14. В каких элементах таблицы хранятся данные базы:

- 1) в полях;
- 2) в строках;
- 3) в столбцах;
- 4) в записях;
- 5) в ячейках?

Один правильный ответ

Правильный ответ – д)

15. Содержит ли какую-либо информацию таблица, в которой нет ни одной записи?

- 1) пустая таблица, не содержит ни какой информации;
- 2) пустая таблица содержит информацию о структуре базы данных;
- 3) пустая таблица содержит информацию о будущих записях;
- 4) таблица без записей существовать не может.

Один правильный ответ

Правильный ответ – б)

16. Содержит ли какую-либо информацию таблица, в которой нет полей?

- 1) содержит информацию о структуре базы данных;
- 2) не содержит ни какой информации;
- 3) таблица без полей существовать не может;
- 4) информацию о будущих записях.

Один правильный ответ

Правильный ответ – в)

17. В чем состоит особенность поля «счетчик»?

- 1) служит для ввода числовых данных;
- 2) служит для ввода действительных чисел;
- 3) данные хранятся не в поле, а в другом месте, а в поле хранится только указатель на то, где расположен текст;
- 4) имеет ограниченный размер;
- 5) имеет свойство автоматического наращивания.

Один правильный ответ

Правильный ответ – д)

18. В чем состоит особенность поля «мемо»?

- 1) служит для ввода числовых данных;
- 2) служит для ввода действительных чисел;
- 3) данные хранятся не в поле, а в другом месте, а в поле хранится только указатель на то, где расположен текст;
- 4) имеет ограниченный размер;

- 5) имеет свойство автоматического наращивания.

Один правильный ответ

Правильный ответ – в)

19. Какое поле можно считать уникальным?
- 1) поле, значения в котором не могут повторяться;
 - 2) поле, которое носит уникальное имя;
 - 3) поле, значение которого имеют свойство наращивания.

Один правильный ответ

Правильный ответ – а)

20. Ключами поиска в системах управления базами данных (СУБД) называются:
- 1) диапазон записей файла БД, в котором осуществляется поиск;
 - 2) логические выражения, определяющие условия поиска;
 - 3) поля, по значению которых осуществляется поиск;
 - 4) номера записей, удовлетворяющих условиям поиска;
 - 5) номер первой по порядку записи, удовлетворяющей условиям поиска?

Один правильный ответ

Правильный ответ – в)

21. Транзакция – это ...
- 1) группа операций, которую невозможно прервать. Выполняется всегда полностью и обеспечивает целостность данных;
 - 2) группа операций которая всегда выполняется успешно, вне зависимости от конкурирующих транзакций, но не обеспечивает целостность данных;
 - 3) группа операций, которая всегда может быть выполнена либо полностью успешно, соблюдая целостность данных и независимо от параллельно идущих других транзакций, либо не выполнена вообще;
 - 4) логическая группа операций, выполнение которой может быть прервано администратором после любой операции с сохранением промежуточных результатов.

Один правильный ответ

Правильный ответ – в)

22. Операция приведения данных к единой форме с целью повышения их уровня доступности для обработки – это ...
- 1) фильтрация;
 - 2) формализация (нормализация);

- 3) архивация;
- 4) сортировка.

Один правильный ответ

Правильный ответ – б)

23. Задано условие: «Ни один из ключевых атрибутов отношения не зависит функционально от любого другого неключевого атрибута». Условие выполняется для ...
- 1) ни для одной из вышеперечисленных;
 - 2) 1 нормальной формы;
 - 3) 2 нормальной формы;
 - 4) 3 нормальной формы.

Один правильный ответ

Правильный ответ – а)

24. Небольшая группа связанных между собой элементов данных, хранящихся в информационной системе называется ...
- 1) таблицей;
 - 2) ключом;
 - 3) полем;
 - 4) записью.

Один правильный ответ

Правильный ответ – г)

25. Ключ к базе данных – это ...
- 1) поле, которое однозначно определяет соответствующую запись;
 - 2) простейший объект базы данных для хранения значений одного реального объекта или процесса;
 - 3) совокупность логически связанных полей, характеризующих типичные свойства реального объекта;
 - 4) процесс группировки данных по определенным параметрам.

Один правильный ответ

Правильный ответ – а)

26. Аналогом элемента реляционной базы данных является ...
- 1) файл;
 - 2) папка;
 - 3) вектор;
 - 4) двумерная таблица.

Один правильный ответ

Правильный ответ – г)

27. По способу доступа к БД СУБД бывают ...

- 1) таблично-серверные;
- 2) диск-серверные;
- 3) клиент-серверные;
- 4) серверные.

Один правильный ответ

Правильный ответ – в)

28. В записи таблицы реляционной БД может содержаться ...

- 1) только текстовая информация;
- 2) неоднородная информация (данные разных типов);
- 3) только числовая информация;
- 4) исключительно однородная информация (данные только одного типа).

Один правильный ответ

Правильный ответ – б)

29. СУБД – это ...

- 1) свойства удаленной базы данных;
- 2) система удаления заблокированных данных;
- 3) система управления большими данными;
- 4) система управления базами данных.

Один правильный ответ

Правильный ответ – г)

30. База данных – это:

- 1) интерфейс, поддерживающий наполнение и манипулирование данными;
- 2) специальным образом организованная и хранящаяся на внешнем носителе совокупность взаимосвязанных данных о некотором объекте;
- 3) программа для хранения и обработки больших массивов информации;
- 4) двумерный массив данных.

Один правильный ответ

Правильный ответ – б)

31. Можно ли в форме запретить изменения значений в каком-либо поле?

- 1) да, если в режиме конструктора формы для нужного элемента вызвать окно **Свойств** и установить для свойства **Блокировка значения** – Да;

- 2) да, если в режиме конструктора формы выделить нужный элемент и выполнить команду **Формат** ▶ **На задний план**;
- 3) да, если в режиме конструктора формы выделить нужный элемент и выполнить команду **Вид** ▶ **Последовательность перехода**;
- 4) нет, этого сделать нельзя.

Один правильный ответ

Правильный ответ – а)

32. Вы хотите, чтобы при открытии базы данных на экране автоматически отображалась заданная форма. Можно ли это сделать и как, если можно?
- 1) да, если дать команду **Сервис** ▶ **Параметры**;
 - 2) да, если дать команду **Файл** ▶ **Свойства базы данных**;
 - 3) да, если дать команду **Сервис** ▶ **Параметры запуска**.

Один правильный ответ

Правильный ответ – в)

33. Между двумя таблицами в базе данных установлена связь один-ко-многим, при этом включен режим Обеспечение целостности данных, но выключен режим Каскадное удаление. Что случится, если Вы удаляете из главной таблицы запись, которая имеет связанные записи в подчиненной таблице?
- 1) **MS Access** не позволит выполнить удаление;
 - 2) **MS Access** попросит пользователя подтвердить удаление;
 - 3) **MS Access** удалит записи только в подчиненной таблице;
 - 4) **MS Access** удалит связь между таблицами.

Один правильный ответ

Правильный ответ – а)

34. Могут ли дублироваться значения в ключевом поле таблицы?
- 1) да, если в качестве ключевого поля использовать поле с типом данных объекта OLE;
 - 2) нет, никогда;
 - 3) да, если это поле будет использовано для установления связи с другими таблицами;
 - 4) да, если ключевое поле состоит более чем одного поля таблицы.

Один правильный ответ

Правильный ответ – б)

35. Вы поместили в форму командную кнопку для запуска макроса. Как нужно модифицировать свойства кнопки, чтобы она запускала макрос?
- 1) имя кнопки должно быть таким же, как имя макроса;
 - 2) для свойства **До обновления** указать имя макроса;

- 3) для свойства **Текущая запись** указать имя макроса;
- 4) для свойства **Нажатие кнопки** указать имя макроса.

Один правильный ответ

Правильный ответ – г)

36. Может ли пользователь изменять данные в поле таблицы, которое имеет тип Счетчик?
- 1) нет, этого сделать нельзя, так как MS Access автоматически обслуживает содержимое этих полей;
 - 2) да, если дважды щелкнуть ячейку и начать редактирование;
 - 3) да, если дать команду Записи ► Ввод данных;
 - 4) да, если дать команду Правка ► Вставить.

Один правильный ответ

Правильный ответ – г)

37. В отчете, содержащем данные о сотрудниках, записи сгруппированы по полю «КодОтдела». Что нужно сделать, чтобы в отчете после каждой группы указывалось количество сотрудников данного отдела?
- 1) в разделе Примечание группы «КодОтдела» поместить поле с выражением =Count([КодОтдела]);
 - 2) в разделе Примечание группы «КодОтдела» поместить поле с выражением =Sum([КодОтдела]);
 - 3) в разделе Примечание отчета поместить поле с выражением =Count([КодОтдела]);
 - 4) в разделе Примечание отчета поместить поле с выражением =Sum([КодОтдела]).

Один правильный ответ

Правильный ответ – б)

38. В базе данных имеется таблица Отделы и таблица Сотрудники. При условии, что в каждый отдел включено несколько сотрудников, но каждый сотрудник может быть включен только в один отдел, какой тип связи между таблицами нужно установить?
- 1) один-ко-многим от таблицы Отделы к таблице Сотрудники;
 - 2) один-ко-многим от таблицы Сотрудники к таблице Отделы;
 - 3) многие-ко-многим;
 - 4) один-к-одному.

Один правильный ответ

Правильный ответ – а)

39. Можно ли создать запрос, который при выполнении отображает диалоговое окно для ввода параметров условия отбора?

- 1) да, если создать запрос с изменениями;
- 2) да, если создать запрос с итогами;
- 3) да, если создать запрос с параметрами;
- 4) нет, этого сделать нельзя.

Один правильный ответ

Правильный ответ – в)

40. Можно ли изменить шрифт по умолчанию, который используется для отображения всех таблиц в базе данных, и если можно то, как это сделать?
- 1) нет, этого сделать нельзя;
 - 2) дать команду **Файл** ▶ **Свойства базы данных**;
 - 3) дать команду **Вид** ▶ **Объекты базы данных** ▶ **Таблица**;
 - 4) дать команду **Сервис** ▶ **Параметры** и перейти на вкладку **Режим таблицы**.

Один правильный ответ

Правильный ответ – а)

41. Как можно ускорить ввод в таблицу одинаковых данных?
- 1) использовать команды **Копировать** и **Вставить**;
 - 2) воспользоваться окном **Область ввода**;
 - 3) использовать комбинацию клавиш **Ctrl+**;
 - 4) использовать фильтр.

Один правильный ответ

Правильный ответ – г)

42. Вы удалили в таблице запись. Можно ли восстановить удаленную запись в таблице и как?
- 1) нет, этого сделать нельзя;
 - 2) да, если сразу после удаления записи нажать кнопку **Восстановить запись**;
 - 3) да, если сразу после удаления записи нажать клавишу **Esc**;
 - 4) да, если закрыть таблицу без сохранения.

Один правильный ответ

Правильный ответ – а)

43. Можно ли изменить шрифт по умолчанию, который используется для отображения всех таблиц в базе данных, и если можно то, как это сделать?
- 1) нет, этого сделать нельзя;
 - 2) дать команду **Файл** ▶ **Свойства базы данных**;
 - 3) дать команду **Вид** ▶ **Объекты базы данных** ▶ **Таблица**;

- 4) дать команду Сервис ▶ Параметры и перейти на вкладку Режим таблицы.

Один правильный ответ

Правильный ответ – а)

44. Какую команду нужно выбрать, чтобы просмотреть связи между таблицами базы данных?
 - 1) Сервис ▶ Схема данных;
 - 2) Файл ▶ Свойства базы данных;
 - 3) Вид ▶ Объекты базы данных;
 - 4) Сервис ▶ Параметры.

Один правильный ответ

Правильный ответ – а)

45. Вы открыли таблицу и решили изменить шрифт в одном из столбцов таблицы. Для этого Вы выделили нужный столбец, дали команду Формат ▶ Шрифт и выбрали другой шрифт. Что Вы увидите на экране?
 - 1) изменение шрифта произошло в выделенном столбце таблицы;
 - 2) изменение шрифта произошло во всей таблице.

Один правильный ответ

Правильный ответ – б)

46. Вы хотите с помощью операции поиска найти в таблице текстовые строки, которые начинаются на букву «А». Какой образец поиска Вы должны задать?
 - 1) символ вопроса и заданная буква(?А);
 - 2) заданная буква и звездочка(А*).

Один правильный ответ

Правильный ответ – б)

47. Вы открыли таблицу с информацией о сотрудниках, в которой есть поле «Дата рождения». Вам нужно найти запись в таблице, относящуюся к самому молодому сотруднику. Какой из способов является самым быстрым?
 - 1) установить курсор ввода в любую ячейку столбца «Дата рождения» и щелкнуть кнопку **Сортировка по возрастанию** на панели инструментов;
 - 2) установить курсор ввода в любую ячейку столбца «Дата рождения» и щелкнуть кнопку **Сортировка по убыванию** на панели инструментов;
 - 3) использовать команду **Найти**;
 - 4) применить фильтр.

Один правильный ответ

Правильный ответ – а)

48. Что можно использовать в качестве источника данных для формы?
- 1) таблицы;
 - 2) запросы;
 - 3) комбинацию таблиц и запросов;
 - 4) другую форму.

Один правильный ответ

Правильный ответ – в)

49. Какое свойство поля таблицы должно использоваться, чтобы при добавлении новой записи в таблицу в ячейке автоматически устанавливалось заданное значение?
- 1) Подпись;
 - 2) Обязательное поле;
 - 3) Значение по умолчанию;
 - 4) Условие на значение.

Один правильный ответ

Правильный ответ – в)

50. Что можно использовать в качестве источника данных для отчета?
- 1) таблицы;
 - 2) запросы;
 - 3) комбинацию таблиц и запросов;
 - 4) другой отчет.

Один правильный ответ

Правильный ответ – в)

51. Представлена таблица базы данных «Телефонный справочник». После проведения сортировки по полю «Фамилия» в порядке возрастания запись, содержащая номер телефона 384-15-15 переместится на ...

	Фамилия	Имя	Отчество	Телефон
	Иванов	Николай	Васильевич	234-56-98
	Сидоров	Александр	Иванович	235-60-07
	Петров	Иван	Денисович	435-88-78
	Скворцов	Артем	Георгиевич	568-98-00
	Трофимов	Михаил	Валерьевич	384-15-15

- 1) 4 строки вверх;
- 2) 2 строки вверх;
- 3) 3 строки вверх;
- 4) 1 строку вверх;

5) не переместится.

Один правильный ответ

Правильный ответ – д)

52 Представлена таблица базы данных «Литература». Запросу (Серия=«Для чайников») или (Количество страниц>299) в этой базе данных удовлетворяют ...

Автор	Серия	Название	Год	Кол_стр
Уолш Р.	Для начинающих	Windows 95	1996	126
Султанов И.	Для начинающих	Энциклопедия Delphi	1997	300
Кирсанов Д.	Для чайников	Word 7.0	1996	236
Харвей Г.	Для чайников	Excel 2000	2001	382
Сигел Ч.	Изучи сам	Access 97	1998	352
Визо М.	Компьютер для носорогов	Access 2.0	1994	255

- 1) три записи;
- 2) пять записей;
- 3) одна запись;
- 4) ни одной записи;
- 5) четыре записи.

Один правильный ответ

Правильный ответ – д)

53 Эффективное средство представления данных запроса к базе в печатном формате – ...

Школа

Класс	Фамилия	Имя	Отчество	Год рождения
10	Лькова	Ольга	Игоревна	1988
11	Семенов	Олег	Иванович	1987
11	Морозов	Иван	Сергеевич	1987
10	Рыков	Роман	Валерьевич	1988
10	Попов	Сергей	Вячеславович	1988
10	Зайцева	Марина	Адольфовна	1988

- 1) запрос;
- 2) форма;
- 3) структура;
- 4) страница удаленного доступа;
- 5) отчет.

Один правильный ответ

Правильный ответ – д)

54 Представлена таблица базы данных «Кадры». При сортировке по возрастанию по полю «Фамилия» поменяются местами записи с порядковыми номерами

Порядковый номер	Фамилия	Год рождения	Оклад
1	Иванов	1956	2400
2	Сидоров	1957	5300
3	Петров	1956	3600
4	Скворцов	1952	1200
5	Трофимов	1958	4500

- 1) 1-я и 4-я;
- 2) 3-я и 4-я;
- 3) 2-я и 3-я;
- 4) 2-я и 4-я;
- 5) 1-я и 3-я.

Один правильный ответ

Правильный ответ – в)

55 Созданный пользователем графический интерфейс для вывода информации из таблицы базы данных – это ...

- 1)запрос;
- 2)форма;
- 3)структура;
- 4)страница удаленного доступа;
- 5)отчет.

Один правильный ответ
Правильный ответ – б)

56 Таблица, полученная из совокупности связанных таблиц путем выбора строк, удовлетворяющих данным условиям на значения полей – это ...

- 1)схема;
- 2)отчет;
- 3)запрос;
- 4)модуль;
- 5)форма.

Один правильный ответ
Правильный ответ – в)

57 Представлена таблица базы данных «Школа». Запрос для вывода списка учеников 10-х классов, 1988 года рождения, имеющих оценки не ниже 4, содержит выражение:

	Код учащегося	Фамилия	Имя	Отчество	Год рождения	Класс	Оценка
	1	Лыкова	Ольга	Игоревна	1988	10	5
	2	Семенов	Олег	Иванович	1987	11	4
	3	Морозов	Иван	Сергеевич	1987	11	3
	4	Рыков	Роман	Валерьевич	1988	10	5
	5	Попов	Сергей	Вячеславович	1988	10	4
	6	Зайцева	Марина	Адольфовна	1988	10	5

- 1)(Оценка>=4) или (Год_рождения=1988) и (Класс=10);
- 2)(Класс=10) и (Год_рождения=1988) и (Оценка=5) и (Оценка=4);
- 3)(Оценка>=4) и (Год_рождения=1988) и (Класс=10);
- 4)(Класс=10) или (Оценка>=4) и (Год_рождения=1988);
- 5)(Класс=10) или (Оценка>=4) или (Год_рождения=1988).

Один правильный ответ
Правильный ответ – в)

58 Представлена таблица базы данных «Кадры». При поиске по условию (Год_рождения>1956 или Оклад>5000) будут найдены фамилии...

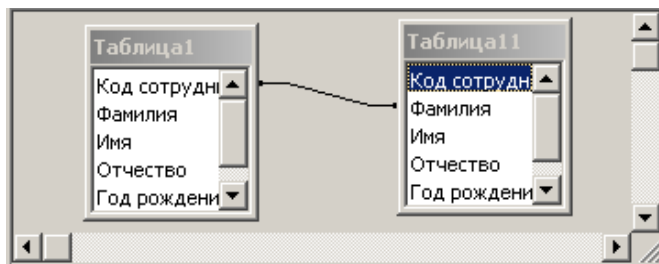
Порядковый номер	Фамилия	Год рождения	Оклад
1	Иванов	1956	2400
2	Сидоров	1957	5300
3	Петров	1956	3600
4	Скворцов	1952	1200
5	Трофимов	1958	4500

- 1) Трофимов, Сидоров;
- 2) Иванов, Петров, Трофимов, Скворцов;
- 3) Трофимов;
- 4) нет записей, удовлетворяющих данному запросу;
- 5) Иванов, Петров, Трофимов.

Один правильный ответ

Правильный ответ – а)

59 Для наглядного отображения связей между таблицами служит ...



- 1) условие на значение;
- 2) схема данных;
- 3) сообщение об ошибке;
- 4) значение по умолчанию;
- 5) список подстановки.

Один правильный ответ

Правильный ответ – б)

5.2 Фонд контрольных заданий

Раздел: Электронные таблицы

Вариант 1

1. В ячейке **F5** записана формула: $=A12*5$. Ее скопировали в ячейку **F16**. Какой вид будет иметь формула в ячейке **F16**?
2. В ячейке **C8** записана формула: $=B9+19$. Ее скопировали в ячейку **C9**. Какой вид будет иметь формула в ячейке **C9**?
3. В ячейке **E7** записана формула: $=C\$10*2$. Ее скопировали в ячейку **E3**. Какой вид будет иметь формула в ячейке **E3**?
4. В ячейке **D10** записана формула: $=\$E9+6$. Ее скопировали в ячейку **D7**. Какой вид будет иметь формула в ячейке **D7**?
5. В ячейке **A7** записана формула: $=B6/\$A\2 . Ее скопировали в ячейку **C7**. Какой вид будет иметь формула в ячейке **C7**?
6. В ячейке **E7** записана формула: $=C\$10*2$. Ее перенесли в ячейку **E3**. Какой вид будет иметь формула в ячейке **E3**?
7. В ячейке **E10** записана формула: $=\$C\$5-F12$. Ее скопировали в ячейку **C10**. Какой вид будет иметь формула в ячейке **C10**?
8. В ячейке **H6** записана формула: $=F6+\$G\6 . Ее скопировали в ячейку **H4**. Какой вид будет иметь формула в ячейке **H4**?
9. В ячейке **D8** записана формула: $=B\$8+F8$. Ее скопировали в ячейку **D10**. Какой вид будет иметь формула в ячейке **D10**?
10. В ячейке **H6** записана формула: $=F\$6+\$G6$. Ее скопировали в ячейку **D9**. Какой вид будет иметь формула в ячейке **D9**?
11. В ячейке **E5** записана формула: $=E\$3+4$. Можно ли ее скопировать в ячейку:
а) **B5**? в) **F3**?
б) **A5**? г) **F1**?
12. В ячейке **D4** записана формула: $=E5+C3$. Можно ли ее скопировать в ячейку:
а) **D6**? в) **H4**? д) **A4**?
б) **D1**? г) **B4**? е) **F7**?
13. В ячейке **C2** записана формула $=E3-1$. Ее скопировали в ячейку **B1**.

	A	B	C	D	E
1		5			
2		10			
3			4		7
			5		2
					12

Какое значение будет выведено в ячейке **B1**?

14. При копировании формулы из ячейки **B3** в ячейку **C4** в последнюю была занесена формула $=D5*5$. Что было записано в ячейке **B3**?

15. При копировании формулы из ячейки **A2** в ячейки **B2** и **A3** в них были занесены формулы **=B1+6** и **=A2+6** соответственно. Что было записано в ячейке **A2**?

Задание 2

В фирме А. каждому сотруднику зарплату за месяц выдают дважды: сначала часть оклада в виде аванса, а по истечении месяца – остальную часть оклада. Произвести расчет выплат, если:

- аванс составляет 40% оклада;
- подоходный налог определяется по формуле $13\%(\text{Оклад} - \text{МРОТ} - \text{Пенсионный налог})$, где МРОТ – минимальный размер оплаты труда; МРОТ = 600 р. (может изменяться)
- профсоюзный взнос и пенсионный налог составляют по 1 % от оклада;
- в колонке Сумма к выдаче должна быть указана сумма денег, получаемых сотрудником по истечении месяца.

ФИО	Оклад	Аванс	Подоходный налог	Пенсионный налог	Профсоюзный взнос	Сумма к выдаче
1.Иванов П.Ю.	2500					
2.Сидоров П.Д.	2600					
3.Миронов А.Л.	3000					
4.Дибров К.И.	1800					
5.Петров В.А.	1500					
6.Данченко А.Г.	3500					
7.Рогов Н.А.	2900					
8.Бендер П.В.	1750					
9.Смирнов Е.П.	2200					
10.Лакин Р.Д.	2500					
ИТОГО:						

Постройте диаграмму, наглядно показывающую распределение заработной платы.

Вариант 2

1. В ячейке **E15** записана формула: $=A2+5$. Ее скопировали в ячейку **E16**. Какой вид будет иметь формула в ячейке **E16**?
2. В ячейке **B8** записана формула: $=D8-8$. Ее скопировали в ячейку **B7**. Какой вид будет иметь формула в ячейке **B7**?
3. В ячейке **E3** записана формула: $=\$C10+3$. Ее скопировали в ячейку **G3**. Какой вид будет иметь формула в ячейке **G3**?
4. В ячейке **B7** записана формула: $=A\$2*2$. Ее скопировали в ячейку **D7**. Какой вид будет иметь формула в ячейке **D7**?
5. В ячейке **B6** записана формула: $=A\$6/\$A2$. Ее скопировали в ячейку **B7**. Какой вид будет иметь формула в ячейке **B7**?
6. В ячейке **E3** записана формула: $=\$C10+3$. Ее перенесли в ячейку **G3**. Какой вид будет иметь формула в ячейке **G3**?
7. В ячейке **E10** записана формула: $=\$B10+F12$. Ее скопировали в ячейку **E6**. Какой вид будет иметь формула в ячейке **E6**?
8. В ячейке **H5** записана формула: $=F\$5-\$G\$5$. Ее скопировали в ячейку **E5**. Какой вид будет иметь формула в ячейке **E5**?
9. В ячейке **D10** записана формула: $=\$E9*10$. Ее скопировали в ячейку **A10**. Какой вид будет иметь формула в ячейке **A10**?
10. В ячейке **A7** записана формула: $=B6/\$A\2 . Ее скопировали в ячейку **C7**. Какой вид будет иметь формула в ячейке **C7**?
11. В ячейке **D4** записана формула: $=\$B\$2+E3$. Можно ли ее скопировать в ячейку:
а) **C3**? в) **A1**? д) **D2**? ж) **C2**?
б) **A2**? г) **B2**? е) **D1**?
12. В ячейке **F5** записана формула: $=F\$3+4$. Можно ли ее скопировать в ячейку:
а) **B5**? в) **F3**?
б) **A5**? г) **F1**?
13. В ячейке **B2** записана формула $=\$A\$2+100$. Ее скопировали в ячейку **C4**.

	A	B	C	D
1	10	6	5	3
2	6		12	4
3		14	2	5

Какое значение будет выведено в ячейке **C4**?

14. При копировании формулы из ячейки **B4** в ячейку **D7** в последнюю была занесена формула $=E7+8$. Что было записано в ячейке **B4**?

15. При копировании формулы из ячейки **B3** в ячейки **D3** и **B2** в них были занесены формулы **=C2+2** и **=A1+2** соответственно. Что было записано в ячейке **B3**?

Задание 2

Граждане А., В. и С. открыли 01.01.2002 счет в банке, вложив соответственно 1000, 2500 и 6000 рублей. Через каждый месяц размер вклада увеличивается на 1,2% от имеющейся суммы. Определите прирост суммы вклада за каждый месяц и за год. Результаты расчета представьте в виде таблицы:

Клиент	январь, 01.01.02	февраль		...	декабрь		год	
		% прироста, руб.	итого, руб		% прироста, руб.	итого, руб	% прироста, руб.	итого, руб
А.	1000							
В.	2500							
С.	6000							

Постройте диаграмму, наглядно показывающую увеличение вкладов в течение года.

Осуществите расчеты, изменив процент увеличения вклада в месяц: 2%, 2,5%, 3%.

Вопросы и задания к зачету

Задание 1 (Лист: Продажа ткани): При покупке больших партий ткани (габардин, твид, драп, бязь, мадаполам, вельвет) фирма предоставляет следующие скидки: 5% – при покупке более двух кусков ткани одного вида; 10% – при покупке более пяти кусков ткани одного вида.

Создайте базу данных, содержащую следующие поля: покупатель; дата поставки, дд/мм/гг.; вид ткани; цена 1 метра ткани, м.; количество метров в куске, м.; количество кусков, шт.; стоимость товара, руб.; скидка, %; скидка, руб.; окончательная стоимость товара, руб.

Создайте диапазон условий, с помощью которого можно было бы извлекать из базы данных сведения о покупателях, приобретенных ими видах тканей, скидках и т.д.

По результатам расчетов произведите **Сортировку данных**, постройте **Сводную таблицу**.

Задание 2 (Лист: Выпуск изделий): Для выпуска двух видов изделий используется три вида ресурсов. Известны запасы ресурсов и прибыль на единицу продукции:

Вид сырья:	C1, C2, C3;
Запасы ресурсов:	20, 40, 30;
Количество ресурсов на единицу продукции 1-го вида:	2, 8, 5;
Количество ресурсов на единицу продукции 2-го вида:	5, 5, 6;
Прибыль от единицы продукции 1-го вида:	50;
Прибыль от единицы продукции 2-го вида:	40.

Необходимо так спланировать производство, чтобы прибыль, получаемая от реализации продукции, была наибольшей.

По результатам расчетов постройте **Сценарий**, покажите графически объем выручки полученной от реализации продукции всех видов.

Задание 3 (Лист: Функции):

1. Рассчитайте таблицу значений функций:

$$f(x) = \cos\left(-3x + \frac{\pi}{2}\right)$$

$$g(x) = \sqrt{\left|\sin\left(\frac{x}{3} + \frac{\pi}{2}\right)\right|}$$

для значений x в пределах от -3 до 1 с шагом $0,2$. Постройте графики этих

функций на одной координатной плоскости и сделайте подписи под осями.

Создайте легенду.

Разместите название кривых возле самих кривых с помощью функции подписи данных, а не легенды.

2. Создайте трехмерный график функции $\sin(xy)$ для x и y , изменяющихся от $-\pi$ до $+\pi$.

3. При **A1 = ИСТИНА** и **B2 = ЛОЖЬ**, каковы будут итоги следующих формул:

a) =И(ИЛИ(И(A1;A1);A1);B1)

b) =ИЛИ(И(A1;A1);И(A1;B1))

Задание 4 (Лист: Ссылки):

16. В ячейке **C5** записана формула: **=B\$5+D3**. Можно ли ее скопировать в ячейку:

a) **A5?** в) **C3?** д) **C2?** ж) **D1?**

б) **A2?** г) **B3?** е) **D4?**

17. В ячейке **D4** записана формула: **=B\$3*3**. Можно ли ее скопировать в ячейку:

a) **D2?** в) **B4?**

б) **D1?** г) **A4?**

18. В ячейке **E2** записана формула **=E1*10**. Ее скопировали в ячейку **F2**. Какое значение будет выведено в ячейке **F2** (значения в ячейках **E1=6**; **F1=5**) ?

19. При копировании формулы из ячейки **E7** в ячейку **B4** в последнюю была занесена формула **=\$F7*8**. Что было записано в ячейке **E7**?

20. При копировании формулы из ячейки **D5** в ячейки **D2** и **F3** в них были занесены формулы **=B\$7*2** и **=D\$7*2** соответственно. Что было записано в ячейке **D5**?

Раздел: Базы данных. MS Access

При выполнении задания необходимо привести баз данных к третьей нормальной форме. В базе данных должна быть *Главная кнопочная форма*, вызываемая при открытии файла базы данных. Пользователь базы данных может работать только с кнопочной формой и элементами, вызываемыми при помощи нажатия кнопок главной формы. В тех запросах, где указано, что данные вводятся пользователем, перед выполнением запроса должно выводиться окно, в котором вводится значение указанного параметра. Для этого используются запросы с параметром. В созданную базу данных должно быть введено не менее 10 записей.

Вариант 1.База данных по учету отгрузки товаров.

- 1.1. База Данных (БД) должна быть приведена к третьей нормальной форме и содержать следующие поля: Номер накладной, Дата, Имя заказчика, Адрес заказчика, телефон заказчика, Расчетный счет заказчика (20 цифр), Товар, кол-во, цена за ед., единицы измерения, сумма за товар.
 - 1.2. Создать формы для ввода информации в каждую из созданных таблиц.
 - 1.3. Создать два запроса:
 - Имя заказчика, телефон и расчетный счет, отсортированные по имени заказчика.
 - Общие суммы заказов для каждого заказчика
 - 1.4. Создать два отчета на основании имеющихся запросов.
 - 1.5. Создать основную кнопочную форму.
-

Вариант 2.База данных по учету успеваемости студентов факультета. (Примечание: считаем, что в группе не может быть полных тезок, то есть людей, у которых Фамилия, Имя и Отчество совпадают полностью.)

- 2.1. База Данных (БД) должна быть приведена к третьей нормальной форме и содержать следующие поля: ФИО, Группа, Предмет, Преподаватель по предмету, количество часов по предмету, зачет/экзамен, оценка, кол-во пропусков, процент посещения занятий.
 - 2.2. Создать формы для ввода информации в каждую из созданных таблиц.
 - 2.3. Создать два запроса:
 - ФИО студента, Предмет, кол-во пропусков. Сортировка по фамилии студента. Вывести все записи, в которых есть хотя бы один пропуск.
 - ФИО студента, Группа, Средний балл.
 - 2.4. Создать два отчета на основании имеющихся запросов.
 - 2.5. Создать основную кнопочную форму.
-

Вариант 3.База данных по учету видеокассет.

- 3.1. База Данных (БД) должна быть приведена к третьей нормальной форме и содержать следующие поля: Номер видеокассеты, название фильма, Режиссер, год выхода фильма, Актер 1, год рождения Актера1, основные фильмы актера1, гл.роль в др. фильмах Актера 1, Актер 2, год рождения Актера1, основные фильмы актера 2, гл.роль в др. фильмах Актера 2, жанр фильма, оригинальный язык фильма, тип перевода.
- 3.2. Создать формы для ввода информации в каждую из созданных таблиц.
- 3.3. Создать два запроса:
 - Название фильма, режиссер, жанр, тип перевода. Сортировка по жанрам.
 - Номер видеокассеты, название фильма, Актер1, Актер2, жанр. Выводятся только фильмы вышедшие в указанном году. Год вводится пользователем.
- 3.4. Создать два отчета на основании имеющихся запросов.
- 3.5. Создать основную кнопочную форму.



Вариант 4.База данных «Библиотека».

- 4.1. База Данных (БД) должна быть приведена к третьей нормальной форме и содержать следующие поля: Код книги, Название, Автор1, Автор2, Автор3, кол-во стр., Цена, Жанр, Издание, Изд-во, Адрес изд-ва, e-mail изд-ва, www-адрес изд-ва.
- 4.2. Создать формы для ввода информации в каждую из созданных таблиц.
- 4.3. Создать два запроса:
 - Название книги, Автор1, кол-во страниц. Сортировка по Коду книги.
 - Название книги, Автор1, Автор2, Автор3, кол-во страниц, жанр. Выводятся все записи, указанного пользователем жанра. Сортировка по авторам.
- 4.4. Создать два отчета на основании имеющихся запросов.
- 4.5. Создать основную кнопочную форму.



Вариант 5.База данных «Отдел кадров».

- 5.1. База Данных (БД) должна быть приведена к третьей нормальной форме и содержать следующие поля: Табельный номер, ФИО, должность, разряд, оклад, Номер цеха, Специализация цеха, начальник цеха, Дата рождения, количество несовершеннолетних детей.
- 5.2. Создать формы для ввода информации в каждую из созданных таблиц.
- 5.3. Создать два запроса:
 - Табельный номер, ФИО, должность, номер цеха. Сортировка по номеру цеха и затем по табельному номеру.

- Табельный номер, ФИО, сумма налоговой льготы. Налоговая льгота рассчитывается по следующему правилу: 400 руб. на работающего + по 600 рублей на каждого несовершеннолетнего ребенка.

5.4. Создать два отчета на основании имеющихся запросов.

5.5. Создать основную кнопочную форму.

Вариант 6.База данных «Учет наличия товаров торговой компании».

6.1. База Данных (БД) должна быть приведена к третьей нормальной форме и содержать следующие поля: Название склада, Адрес склада, ФИО старшего кладовщика, тел. склада, Товар, единицы измерения товара, кол-во на складе, цена за единицу.

6.2. Создать формы для ввода информации в каждую из созданных таблиц.

6.3. Создать два запроса:

- Товар, Название склада, единицы измерения, количество. Выводятся данные только для указанного пользователем товара.
- Номер склада, Общая стоимость всех товаров. Сортировка по номеру склада.

6.4. Создать два отчета на основании имеющихся запросов.

6.5. Создать основную кнопочную форму.

Вариант 7.База данных «Расчет зарплаты»

7.1. База Данных (БД) должна быть приведена к третьей нормальной форме и содержать следующие поля: Табельный номер, ФИО работника, Отдел, Начальник отдела, внутренний телефон отдела, Должность, разряд, количество отработанных дней, оклад, Сумма на руки.

7.2. Создать формы для ввода информации в каждую из созданных таблиц.

7.3. Создать два запроса:

- Табельный номер, ФИО работника, Должность. Выводятся все записи для указанного пользователем отдела.
- Табельный номер, ФИО работника, должность, сумма на руки. Алгоритм расчета заработной платы. Начислено: (Оклад + уральские (оклад*15%)) * Количество отработанных дней/количество дней в месяце; Удержано: Начислено*13%; На руки: Начислено – Удержано. Количество рабочих дней в месяце вводится пользователем при вызове запроса.

7.4. Создать два отчета на основании имеющихся запросов.

7.5. Создать основную кнопочную форму.

Вариант 8.База данных «Подшивки журналов»

8.1. База Данных (БД) должна быть приведена к третьей нормальной форме и содержать следующие поля: Номер подшивки, Название статьи, автор,

журнал, номер, www-адрес журнала, e-mail журнала, кол-во страниц в статье, тематика статьи.

8.2. Создать формы для ввода информации в каждую из созданных таблиц.

8.3. Создать два запроса:

- Автор, Название статьи, количество страниц в статье. Сортировка по автору. Выводятся все статьи указанной пользователем тематики.
- Автор, количество статей данного автора во всех подшивках.

8.4. Создать два отчета на основании имеющихся запросов.

8.5. Создать основную кнопочную форму.

Вариант 9.База данных «Банк»

9.1. База Данных (БД) должна быть приведена к третьей нормальной форме и содержать следующие поля: Номер счета клиента, Наименование клиента, Адрес клиента, ИНН клиента, тел. клиента, сумма на счету, Дата последней операции, Основание операции, Тип платежа.

9.2. Создать формы для ввода информации в каждую из созданных таблиц.

9.3. Создать два запроса:

- Наименование клиента, ИНН Клиента, Сумма на счету. Сортировка по наименованию клиента.
- № счета клиента, Дата последней операции, тип платежа. Вывести записи за последний месяц.

9.4. Создать два отчета на основании имеющихся запросов.

9.5. Создать основную кнопочную форму.

Вариант 10.База данных «Учет больничных листов»

10.1.База Данных (БД) должна быть приведена к третьей нормальной форме и содержать следующие поля: Номер карточки, ФИО больного, Номер участка, ФИО участкового врача, тел. уч. врача, диагноз, кол-во дней, пособие в день, сумма.

10.2.Создать формы для ввода информации в каждую из созданных таблиц.

10.3.Создать два запроса:

- ФИО врача, Номер участка, ФИО больного. Выводятся все записи для указанного пользователем участка.
- ФИО больного, диагноз, кол-во дней, сумма. Сумма рассчитывается по алгоритму: количество дней * пособие в день.

10.4.Создать два отчета на основании имеющихся запросов.

10.5.Создать основную кнопочную форму.

Вариант 11.База данных «Поликлиника»

- 11.1.База Данных (БД) должна быть приведена к третьей нормальной форме и содержать следующие поля: Номер участка, ФИО участкового врача, ФИО больного, Адрес больного, Номер страхового полиса, Дата посещения, Диагноз.
- 11.2.Создать формы для ввода информации в каждую из созданных таблиц.
- 11.3.Создать два запроса:
- Номер участка, ФИО участкового врача, количество принятых больных за текущий месяц.
 - ФИО больного, ФИО участкового врача, Количество посещений за текущий месяц.
- 11.4.Создать два отчета на основании имеющихся запросов.
- 11.5.Создать основную кнопочную форму.
-
-

Вариант 12.База данных «Кинопрокат»

- 12.1.База Данных (БД) должна быть приведена к третьей нормальной форме и содержать следующие поля: Название фильма, Длительность, кинотеатр, Адрес Кинотеатра, Тел. автоответчика, число мест, жанр фильма, Цена билета, Рейтинг фильма.
- 12.2.Создать формы для ввода информации в каждую из созданных таблиц.
- 12.3.Создать два запроса:
- Название фильма, Жанр, средняя по всем кинотеатрам цена билета. Сортировка по Названию фильма.
 - Название фильма, Кинотеатр, Цена билета, число мест. Выводятся все записи, соответствующие указанному пользователем фильму.
- 12.4.Создать два отчета на основании имеющихся запросов.
- 12.5.Создать основную кнопочную форму.
-
-

Вариант 13.База данных «Кафедра»

- 13.1.База Данных (БД) должна быть приведена к третьей нормальной форме и содержать следующие поля: ФИО преподавателя, Должность, Ученое звание, Предмет, Количество часов по предмету, количество учебников по предмету, Вид отчетности (зачет/экзамен).
- 13.2.Создать формы для ввода информации в каждую из созданных таблиц.
- 13.3.Создать два запроса:
- ФИО преподавателя, Должность, Количество часов по всем предметам в течение года. Сортировка по ФИО преподавателя.
 - ФИО преподавателя, Должность, Вид отчетности. Выводятся все записи, для указанного пользователем предмета.
- 13.4.Создать два отчета на основании имеющихся запросов.
- 13.5.Создать основную кнопочную форму.
-
-

Вариант 14.База данных «Компьютерный клуб»

14.1.База Данных (БД) должна быть приведена к третьей нормальной форме и содержать следующие поля: Номер компьютера, Номер зала, ФИО администратора зала, ФИО игрока, возраст игрока, любимые игры игрока, Дата посещения, время, проведенное за компьютером.

14.2.Создать формы для ввода информации в каждую из созданных таблиц.

14.3.Создать два запроса:

- Номер зала, Номер компьютера, Время работы за текущий месяц за компьютером. Выводятся все записи для указанного пользователем игрока.
- Название зала, Количество компьютеров, Среднее время работы всех компьютеров за текущую неделю.

14.4.Создать два отчета на основании имеющихся запросов.

14.5.Создать основную кнопочную форму.

Вариант 15.База данных «Учебные курсы»

15.1.База Данных (БД) должна быть приведена к третьей нормальной форме и содержать следующие поля: Название учебного курса, ФИО преподавателя, Телефон преподавателя, Адрес преподавателя, Стоимость одного часа работы данного преподавателя, Количество часов на курс, Дата разработки, Область знаний курса, стоимость полного курса.

15.2.Создать формы для ввода информации в каждую из созданных таблиц.

15.3.Создать два запроса:

- ФИО преподавателя, Название курса, Количество часов. Сортировка по ФИО преподавателя. Выводятся все записи для указанной пользователем области знаний.
- Название учебного курса, Дата разработки, Стоимость полного курса.

15.4.Создать два отчета на основании имеющихся запросов.

15.5.Создать основную кнопочную форму.

Вариант 16.База данных «Собес»

16.1.База Данных (БД) должна быть приведена к третьей нормальной форме и содержать следующие поля: ФИО состоящего на учете, Статус (инвалид, пенсионер, и т.д.), Адрес, ФИО куратора, Адрес куратора, внутренний телефон куратора, сумма пенсии.

16.2.Создать формы для ввода информации в каждую из созданных таблиц.

16.3.Создать два запроса:

- ФИО состоящего на учете, Адрес, ФИО куратора. Выводятся все записи для введенного пользователем статуса;
- ФИО куратора, количество курируемых. Сортировка по ФИО куратора;

16.4. Создать два отчета на основании имеющихся запросов.

16.5. Создать основную кнопочную форму.

Вариант 17. База данных по учету отгрузки товаров

17.1. База Данных (БД) должна содержать следующие поля: Номер накладной, Дата, Имя заказчика, Адрес заказчика, телефон заказчика, Расчетный счет заказчика (20 цифр), Товар, кол-во, цена за ед., единицы измерения, сумма за товар.

17.2. Создать формы для ввода информации в каждую из созданных таблиц.

17.3. Создать запрос: Имя заказчика, телефон и расчетный счет, отсортированные по имени заказчика.

17.4. Создать отчет на основании запроса.

17.5. Создать основную кнопочную форму.

Вариант 18. База данных по учету успеваемости студентов факультета.

18.1. База Данных (БД) должна содержать следующие поля: ФИО, Группа, Предмет, Преподаватель по предмету, количество часов по предмету, зачет/экзамен, оценка, кол-во пропусков, процент посещения занятий.

18.2. Создать формы для ввода информации в каждую из созданных таблиц.

18.3. Создать запрос: ФИО студента, Предмет, кол-во пропусков. Сортировка по фамилии студента. Вывести все записи, в которых есть хотя бы один пропуск.

18.4. Создать отчет на основании запроса.

18.5. Создать основную кнопочную форму.

Примечание: считаем, что в группе не может студентов у которых Фамилия, Имя и Отчество совпадают полностью.

Вариант 19. База данных по учету видеокассет.

19.1. База Данных (БД) должна содержать следующие поля: Номер видеокассеты, название фильма, Режиссер, год выхода фильма, Актер 1, год рождения Актера1, основные фильмы актера1, гл. роль в др. фильмах Актера 1, Актер 2, год рождения Актера1, основные фильмы актера 2, гл. роль в др. фильмах Актера 2, жанр фильма, оригинальный язык фильма, тип перевода.

19.2. Создать формы для ввода информации в каждую из созданных таблиц.

19.3. Создать запрос: Название фильма, режиссер, жанр, тип перевода. Сортировка по жанрам.

19.4. Создать отчет на основании запроса.

19.5. Создать основную кнопочную форму.

Вариант 20. Вариант 20. База данных «Библиотека».

20.1. База Данных (БД) должна содержать следующие поля: Код книги, Название, Автор1, Автор2, Автор3, кол-во стр., Цена, Жанр, Издание, Изд-во, Адрес изд-ва, e-mail изд-ва, www-адрес изд-ва.

20.2. Создать формы для ввода информации в каждую из созданных таблиц.

20.3. Создать запрос: Название книги, Автор1, кол-во страниц. Сортировка по Коду книги

20.4. Создать отчет на основании запроса.

20.5. Создать основную кнопочную форму.

Вариант 21. База данных «Отдел кадров»

21.1. База Данных (БД) должна содержать следующие поля: Табельный номер, ФИО, должность, разряд, оклад, Номер цеха, Специализация цеха, начальник цеха, Дата рождения, количество несовершеннолетних детей.

21.2. Создать запрос: Табельный номер, ФИО, должность, номер цеха. Сортировка по номеру цеха и затем по табельному номеру.

21.3. Создать отчет на основании запроса.

21.4. Создать основную кнопочную форму.

Вариант 22. База данных «Учет наличия товаров торговой компании»

22.1. База Данных (БД) должна содержать следующие поля: Название склада, Адрес склада, ФИО старшего кладовщика, тел. склада, Товар, единицы измерения товара, кол-во на складе, цена за единицу.

22.2. Создать запрос: Товар, Название склада, единицы измерения, количество. Выводятся данные только для указанного пользователем товара.

22.3. Создать отчет на основании запроса.

22.4. Создать основную кнопочную форму.

Вариант 23. База данных «Расчет зарплаты»

23.1. База Данных (БД) должна содержать следующие поля: Табельный номер, ФИО работника, Отдел, Начальник отдела, внутренний телефон отдела, Должность, разряд, количество отработанных дней, оклад, Сумма на руки.

23.2. Создать запрос: Табельный номер, ФИО работника, Должность. Выводятся все записи для указанного пользователем отдела.

23.3. Создать отчет на основании запроса.

23.4. Создать основную кнопочную форму.

Вариант 24.База данных «Подшивки журналов»

- 24.1.База Данных (БД) должна содержать следующие поля: Номер подшивки, Название статьи, автор, журнал, номер, www-адрес журнала, e-mail журнала, кол-во страниц в статье, тематика статьи.
- 24.2.Создать запрос: Автор, Название статьи, количество страниц в статье. Сортировка по автору. Выводятся все статьи указанной пользователем тематики.
- 24.3.Создать отчет на основании запроса.
- 24.4.Создать основную кнопочную форму.
-
-

Вариант 25.База данных «Банк»

- 25.1.База Данных (БД) должна содержать следующие поля: Номер счета клиента, Наименование клиента, Адрес клиента, ИНН клиента, тел. клиента, сумма на счету, Дата последней операции, Основание операции, Тип платежа.
- 25.2.Создать запрос: Наименование клиента, ИНН Клиента, Сумма на счету. Сортировка по наименованию клиента.
- 25.3.Создать отчет на основании запроса.
- 25.4.Создать основную кнопочную форму.
-
-

Вариант 26.База данных «Учет больничных листов»

- 26.1.База Данных (БД) должна содержать следующие поля: Номер карточки, ФИО больного, Номер участка, ФИО участкового врача, тел. уч. врача, диагноз, кол-во дней, пособие в день, сумма.
- 26.2.Создать запрос: ФИО врача, Номер участка, ФИО больного. Выводятся все записи для указанного пользователем участка.
- 26.3.Создать отчет на основании запроса.
- 26.4.Создать основную кнопочную форму.
-
-

Вариант 27.База данных «Поликлиника»

- 27.1.База Данных (БД) должна содержать следующие поля: Номер участка, ФИО участкового врача, ФИО больного, Адрес больного, Номер страхового полиса, Дата посещения, Диагноз.
- 27.2.Создать запрос: Номер участка, ФИО участкового врача, количество принятых больных за текущий месяц.
- 27.3.Создать отчет на основании запроса.
- 27.4.Создать основную кнопочную форму.
-
-

Вариант 28.База данных «Кинопрокат»

- 28.1.База Данных (БД) должна содержать следующие поля: Название фильма, Длительность, кинотеатр, Адрес Кинотеатра, Тел. автоответчика, число мест, жанр фильма, Цена билета, Рейтинг фильма.
 - 28.2.Создать запрос: Название фильма, Жанр, средняя по всем кинотеатрам цена билета. Сортировка по Названию фильма.
 - 28.3.Создать отчет на основании запроса.
 - 28.4.Создать основную кнопочную форму.
-
-

Вариант 29.База данных «Кафедра»

- 29.1.База Данных (БД) должна содержать следующие поля: ФИО преподавателя, Должность, Ученое звание, Предмет, Количество часов по предмету, количество учебников по предмету, Вид отчетности (зачет/экзамен).
 - 29.2.Создать запрос: ФИО преподавателя, Должность, Количество часов по всем предметам в течении года. Сортировка по ФИО преподавателя.
 - 29.3.Создать отчет на основании запроса.
 - 29.4.Создать основную кнопочную форму.
-
-

Вариант 30.База данных «Компьютерный клуб»

- 30.1.База Данных (БД) должна содержать следующие поля: Номер компьютера, Номер зала, ФИО администратора зала, ФИО игрока, возраст игрока, любимые игры игрока, Дата посещения, время проведенное за компьютером.
 - 30.2.Создать запрос: Номер зала, Номер компьютера, Время работы за текущий месяц за компьютером. Выводятся все записи для указанного пользователем игрока.
 - 30.3.Создать отчет на основании запроса.
 - 30.4.Создать основную кнопочную форму.
-
-

Вариант 31.База данных «Учебные курсы»

- 31.1.База Данных (БД) должна содержать следующие поля: Название учебного курса, ФИО преподавателя, Телефон преподавателя, Адрес преподавателя, Стоимость одного часа работы данного преподавателя, Количество часов на курс, Дата разработки, Область знаний курса, стоимость полного курса.
 - 31.2.Создать запрос: ФИО преподавателя, Название курса, Количество часов. Сортировка по ФИО преподавателя. Выводятся все записи для указанной пользователем области знаний.
 - 31.3.Создать отчет на основании запроса.
 - 31.4.Создать основную кнопочную форму.
-
-

Вариант 32.База данных по учету отгрузки товаров

- 32.1.База Данных (БД) должна содержать следующие поля: Номер накладной, Дата, Имя заказчика, Адрес заказчика, телефон заказчика, Расчетный счет заказчика (20 цифр), Товар, кол-во, цена за ед., единицы измерения, сумма за товар.
 - 32.2.Создать формы для ввода информации в каждую из созданных таблиц.
 - 32.3.Создать запрос: Общие суммы заказов для каждого заказчика
 - 32.4.Создать отчет на основании запроса.
 - 32.5.Создать основную кнопочную форму.
-

Вариант 33.База данных по учету успеваемости студентов факультета.

- 33.1.База Данных (БД) должна содержать следующие поля: ФИО, Группа, Предмет, Преподаватель по предмету, количество часов по предмету, зачет/экзамен, оценка, кол-во пропусков, процент посещения занятий.
- 33.2.Создать формы для ввода информации в каждую из созданных таблиц.
- 33.3.Создать запрос: ФИО студента, Группа, Средний балл.
- 33.4.Создать отчет на основании запроса.
- 33.5.Создать основную кнопочную форму.

Примечание: считаем, что в группе не может быть студентов у которых Фамилия, Имя и Отчество совпадают полностью.

Вариант 34.База данных по учету видеокассет.

- 34.1.База Данных (БД) должна содержать следующие поля: Номер видеокассеты, название фильма, Режиссер, год выхода фильма, Актер 1, год рождения Актера1, основные фильмы актера1, гл.роль в др. фильмах Актера 1, Актер 2, год рождения Актера1, основные фильмы актера 2, гл.роль в др. фильмах Актера 2, жанр фильма, оригинальный язык фильма, тип перевода.
 - 34.2.Создать формы для ввода информации в каждую из созданных таблиц.
 - 34.3.Создать запрос: Номер видеокассеты, название фильма, Актер1, Актер2, жанр. Выводятся только фильмы вышедшие в указанном году. Год вводится пользователем.
 - 34.4.Создать отчет на основании запроса.
 - 34.5.Создать основную кнопочную форму.
-

Вариант 35.База данных «Библиотека»

- 35.1.База Данных (БД) должна содержать следующие поля: Код книги, Название, Автор1, Автор2, Автор3, кол-во стр., Цена, Жанр, Издание, Изд-во, Адрес изд-ва, e-mail изд-ва, www-адрес изд-ва.

- 35.2. Создать формы для ввода информации в каждую из созданных таблиц.
- 35.3. Создать запрос: Название книги, Автор1, Автор2, Автор3, кол-во страниц, жанр. Выводятся все записи, указанного пользователем жанра. Сортировка по авторам.
- 35.4. Создать отчет на основании запроса.
- 35.5. Создать основную кнопочную форму.
-
-

Вариант 36. База данных «Отдел кадров»

- 36.1. База Данных (БД) должна содержать следующие поля: Табельный номер, ФИО, должность, разряд, оклад, Номер цеха, Специализация цеха, начальник цеха, Дата рождения, количество несовершеннолетних детей.
- 36.2. Создать запрос: Табельный номер, ФИО, сумма налоговой льготы. Налоговая льгота рассчитывается по следующему правилу: 400 руб. на работающего + по 600 рублей на каждого несовершеннолетнего ребенка.
- 36.3. Создать отчет на основании запроса.
- 36.4. Создать основную кнопочную форму.
-
-

Вариант 37. База данных «Учет наличия товаров торговой компании»

- 37.1. База Данных (БД) должна содержать следующие поля: Название склада, Адрес склада, ФИО старшего кладовщика, тел. склада, Товар, единицы измерения товара, кол-во на складе, цена за единицу.
- 37.2. Создать запрос: Номер склада, Общая стоимость всех товаров. Сортировка по номеру склада.
- 37.3. Создать отчет на основании запроса.
- 37.4. Создать основную кнопочную форму.
-
-

Вариант 38. База данных «Расчет зарплаты»

- 38.1. База Данных (БД) должна содержать следующие поля: Табельный номер, ФИО работника, Отдел, Начальник отдела, Внутренний телефон отдела, Должность, разряд, Количество отработанных дней, оклад, Сумма на руки.
- 38.2. Создать запрос: Табельный номер, ФИО работника, должность, сумма на руки. Алгоритм расчета заработной платы. Начислено: Оклад + уральские (оклад*15%) + Количество отработанных дней/количество дней в месяце; Удержано: Начислено*13%; На руки: Начислено - Удержано. Количество рабочих дней в месяце вводится пользователем при вызове запроса.
- 38.3. Создать отчет на основании запроса.
- 38.4. Создать основную кнопочную форму.
-
-

Вариант 39.База данных «Подшивки журналов»

- 39.1.База Данных (БД) должна содержать следующие поля: Номер подшивки, Название статьи, автор, журнал, номер, www-адрес журнала, e-mail журнала, кол-во страниц в статье, тематика статьи.
 - 39.2.Создать запрос: Автор, количество статей данного автора во всех подшивках.
 - 39.3.Создать отчет на основании запроса.
 - 39.4.Создать основную кнопочную форму.
-
-

Вариант 40.База данных «Банк»

- 40.1.База Данных (БД) должна содержать следующие поля: Номер счета клиента, Наименование клиента, Адрес клиента, ИНН клиента, тел. клиента, сумма на счету, Дата последней операции, Основание операции, Тип платежа
 - 40.2.Создать запрос: Номер счета клиента, Дата последней операции, тип платежа. Вывести записи за последний месяц.
 - 40.3.Создать отчет на основании запроса.
 - 40.4.Создать основную кнопочную форму.
-
-

Вариант 41.База данных «Учет больничных листов»

- 41.1.База Данных (БД) должна содержать следующие поля: Номер карточки, ФИО больного, Номер участка, ФИО участкового врача, тел. уч. врача, диагноз, кол-во дней, пособие в день, сумма.
 - 41.2.Создать запрос: ФИО больного, диагноз, кол-во дней, сумма. Сумма рассчитывается по алгоритму: количество дней * пособие в день.
 - 41.3.Создать отчет на основании запроса.
 - 41.4.Создать основную кнопочную форму.
-
-

Вариант 42.База данных «Поликлиника»

- 42.1.База Данных (БД) должна содержать следующие поля: Номер участка, ФИО участкового врача, ФИО больного, Адрес больного, Номер страхового полиса, Дата посещения, Диагноз.
 - 42.2.Создать запрос: ФИО больного, ФИО участкового врача, Количество посещений за текущий месяц.
 - 42.3.Создать отчет на основании запроса.
 - 42.4.Создать основную кнопочную форму.
-
-

Вариант 43.База данных «Кинопрокат»

- 43.1.База Данных (БД) должна содержать следующие поля: Название фильма, Длительность, кинотеатр, Адрес Кинотеатра, Тел. автоответчика, число мест, жанр фильма, Цена билета, Рейтинг фильма.
- 43.2.Создать запрос: Название фильма, Кинотеатр, Цена билета, число мест. Выводятся все записи, соответствующие указанному пользователем фильму.
- 43.3.Создать отчет на основании запроса.
- 43.4.Создать основную кнопочную форму.
-
-

Вариант 44.База данных «Кафедра»

- 44.1.База Данных (БД) должна содержать следующие поля: ФИО преподавателя, Должность, Ученое звание, Предмет, Количество часов по предмету, количество учебников по предмету, Вид отчетности (зачет/экзамен).
- 44.2.Создать запрос: ФИО преподавателя, Должность, Вид отчетности. Выводятся все записи, для указанного пользователем предмета.
- 44.3.Создать отчет на основании запроса.
- 44.4.Создать основную кнопочную форму.
-
-

Вариант 45.База данных «Компьютерный клуб»

- 45.1.База Данных (БД) должна содержать следующие поля: Номер компьютера, Номер зала, ФИО администратора зала, ФИО игрока, возраст игрока, любимые игры игрока, Дата посещения, время проведенное за компьютером.
- 45.2.Создать запрос: Название зала, Количество компьютеров, Среднее время работы всех компьютеров за текущую неделю.
- 45.3.Создать отчет на основании запроса.
- 45.4.Создать основную кнопочную форму.
-
-

Вариант 46.База данных «Учебные курсы»

- 46.1.База Данных (БД) должна содержать следующие поля: Название учебного курса, ФИО преподавателя, Телефон преподавателя, Адрес преподавателя, Стоимость одного часа работы данного преподавателя, Количество часов на курс, Дата разработки, Область знаний курса, стоимость полного курса.
- 46.2.Создать запрос: Название учебного курса, Дата разработки, Стоимость полного курса.
- 46.3.Создать отчет на основании запроса.
- 46.4.Создать основную кнопочную форму.
-
-

5 МЕТОДИКА ОЦЕНКИ УРОВНЯ ЗНАНИЙ

5.1 Критерии оценки знаний

Нормы оценки знаний предполагают учет индивидуальных особенностей студентов, дифференцированный подход к обучению, проверке знаний, умений, навыков. В устных и письменных ответах студентов учитывается глубина знаний, их полнота, владение необходимыми умениями в объеме полной программы, осознанность и самостоятельность применения знаний и способов учебной деятельности, логичность изложения материала, умение обобщать, делать выводы в соответствии с заданным вопросом, соблюдение норм литературной речи.

Промежуточный контроль знаний студентов осуществляется при выполнении и защите каждого задания лабораторной работы.

В качестве заключительного контроля знаний студентов в семестре – зачет.

5.2 Условия допуска и сдачи зачета

Студент считается допущенным к зачету по дисциплине в том случае, если выполнены в полном объеме задания лабораторных работ, предусмотренных в семестре, даны положительные ответы на теоретические вопросы соответствующих тем лабораторных занятий, выполнены в полном объеме индивидуальные задания, успешно (с оценкой не ниже «удовлетворительно») пройдено тестирование, целью которого является промежуточная диагностика уровня знаний.

Во время зачета студент выполняет задания, получая за каждое из них оценку. На основании этого и с учетом набранных баллов за устные и письменные ответы в течение семестра, оценок за тестирование устанавливается средний балл, который округляется в ту или иную сторону с учетом количества пропусков занятий по неуважительным причинам, определяя получен зачет по дисциплине или нет.

В устных и письменных ответах студентов на зачете, оцениваются знания и умения по системе зачета. При этом учитывается: глубина знаний, полнота знаний, а также владение необходимыми умениями и навыками в объеме полной программы; осознанность и самостоятельность применения знаний и способов, логичность изложения материала, включая обобщения выводы в соответствии с заданным вопросом, соблюдение норм литературной речи.

Ставится «ЗАЧЕТ» – материал усвоен в полном объеме; изложен логично; основные умения сформированы и устойчивы; выводы и обобщения точны или в усвоении материала незначительные пробелы: изложение недостаточно

систематизировано; отдельные умения недостаточно устойчивы; в выводах и обобщениях допускаются некоторые неточности.

Ставится «НЕЗАЧЕТ» – в усвоении материала имеются пробелы: материал излагается несистематизировано; отдельные умения недостаточно сформированы; выводы и обобщения аргументированы слабо, в них допускаются ошибки; основное содержание материала неусвоено.

6 РЕКОМЕНДУЕМОЕ УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ

6.1 Рекомендуемая литература

6.1.1 Основная

1. Макарова Н.В. Информатика: Практикум по технологии работы на компьютере. / Под ред. Н. В. Макаровой. – М. : Финансы и статистика, 2005. – 256 с.
2. Информатика: Базовый курс: учеб. пособие / Под ред. С.В. Симоновича. – СПб.: Питер, 2004, 2005, 2007. – 640 с.
3. Информатика: Учебник./Под ред. Н.В. Макаровой. – М.: Финансы и статистика, 2005, 2007. – 268 с.
4. Макарова Н.В., Трофимец В.Я. Статистика в Excel: Учеб. пособие. – М.: Финансы и статистика, 2003. – 366 с.
5. Горелова Г.В., Кацко И.А. Теория вероятностей и математическая статистика в примерах и задачах с применением EXCEL: Учеб. пособие. – Ростов н/Д: Феникс, 2002. – 399 с.
6. Лавренов С.М. Excel: Сб. примеров и задач. – М.: Финансы и статистика, 2003. – 336 с.
7. Левит Б.Ю. Диаграммы Excel в экономических моделях. – М. : Финансы и статистика, 2004. – 400 с.
8. Информатика: Учебник. – 3-е перераб. изд./Под ред. Н.В. Макаровой. – М.: Финансы и статистика, 2005. – 768 с.: ил.
9. Информатика: Базовый курс: учеб. пособие/Под ред. С.В. Симоновича. – СПб.: Питер, 2004, 2005, 2007. – 640 с.
10. Информационные технологии управления. Учеб. пособие/Ред. Г.А. Титоренко. – М.: ЮНИТИ-ДАНА, 2003. – 336 с.
11. Саак А.Э., Пахомов Е.В., Тюшняков В.Н. Информационные технологии управления. Учеб. – СПб.: Питер, 2005. – 319 с.
12. Информатика: Введение в информационную безопасность. Учебное пособие. / Под ред. М.А Вуса. – СПб.: Юрид. центр Пресс, 2004. – 204 с.
13. Базы данных. Учебник для вузов/Под ред. А.Д. Хомоненко. – М.: КОРОНА принт, 2002. – 672 с.
14. Агальцов, В.П. Базы данных: Учеб. пособие: Рек. Мин. обр. РФ. – М.: Мир, 2002. – 376 с.

6.1.2 Дополнительная

1. Савицкий Н.И. Технологии организации, хранения и обработки данных: Учебное пособие. – М.: ИНФРА-М, 2001. – 232 с. (Серия «Высшее образование»)
2. Информационные технологии: Учеб. для вузов/Б.Я. Советов, В.В. Цехановский. – 2-е изд., стер. – М.: Высш. шк., 2005. – 263 с.: ил.

3. Касаев Б.С., Каймин В.А., Информатика: практикум на ЭВМ: Учеб. пособие – 2-е изд., перераб. и доп. – М.: ИНФРА-М, 2003. – 272 с. (Серия «Высшее образование»)
4. Меняев М.Ф. Информационные технологии управления: Учебное пособие. В 3 кн.: Книга 1: Информатика. – М.: Омега-Л, 2003. – 464 с.
5. Меняев М.Ф. Информационные технологии управления: Учебное пособие. В 3 кн.: Книга 2: Информационные ресурсы. – М.: Омега-Л, 2003. – 432 с.
6. Харитонов И., Вольман Н. Программирование в Access 2002. – Питер, 2002. – 476 с.
7. Харитонов И.А. Office Access 2003. – СПб.: Питер, 2004. – 464 с.
8. Золотова, С.И. Практикум по Access: подготовительный курс, предваряющий более глубокое изучение технологии баз данных. – М.: Финансы и статистика, 2005. – 144 с.

6.1.3 Методическое обеспечение дисциплины

1. Абакумова И.В., Тибенко Т.А., Сухова Т.Н.. Обработка данных средствами Excel: Учеб. метод. пособие, – Благовещенск: Изд-во Амур. гос. ун-та, 2006. – 68 с.
2. Плутенко А.Д. Реляционная модель данных: Метод. рук. к лабор. работам. – Благовещенск: Изд-во Амур. гос. ун-та, 1998. – 22 с.
3. Лабораторный практикум Microsoft ACCESS: Учеб. пособие. – Саратов: Artech Limited, 2000. – 199 с.

Татьяна Александровна Тибенко
ассистент кафедры КиТО АмГУ

*Прикладная информатика: Учебно-методический комплекс по дисциплине
для специальности 260902 – «Конструирование швейных изделий»*
