

МИНИСТЕРСТВО СЕЛЬСКОГО ХОЗЯЙСТВА РОССИЙСКОЙ
ФЕДЕРАЦИИ



ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«БАШКИРСКИЙ ГОСУДАРСТВЕННЫЙ АГРАРНЫЙ
УНИВЕРСИТЕТ»

Кафедра информатики
и информационных технологий

Б1.О.15 ИНФОРМАЦИОННЫЕ СИСТЕМЫ И ТЕХНОЛОГИИ

Лабораторные работы. Управление базами данных в СУБД Access

МЕТОДИЧЕСКИЕ УКАЗАНИЯ

Направление подготовки

09.03.03 ПРИКЛАДНАЯ ИНФОРМАТИКА

Квалификация (степень) выпускника
бакалавр

Уфа 2021

Рекомендовано к изданию методической комиссией экономического факультета (25 марта 2021 г., протокол № 8)

Составитель: доцент, к.ф.-м.н. Шамсутдинова Т.М.

Рецензент: ст. преподаватель Прокофьева С.В.

Ответственный за выпуск: зав. кафедрой ИИТ, д.т.н.,
Беляева А.С.

ОГЛАВЛЕНИЕ

Лабораторная работа №1

Разработка реляционной базы данных

Лабораторная работа №2

Создание графических форм

Лабораторная работа №3

Разработка форм и запросов с проведением вычислений

Лабораторная работа №4

Разработка интерфейса БД

Лабораторная работа №5

Разработка отчетов и кнопочных форм

Библиографический список

Лабораторная работа №1

Разработка реляционной базы данных

Цель и задачи работы

Изучить методы разработки таблиц и простых форм в СУБД MS Access.

1 Общие сведения

Создание таблиц БД является первым шагом в разработке СУБД. Таблицы предназначены для хранения информации БД. Создание таблицы состоит из двух этапов: задание структуры таблицы; ввод записей в таблицу.

Формы - объекты MS Access, в виде диалоговых окон, которые используются для различных целей:

- удобный ввод данных в таблицу;
- отображение графической информации и любых других объектов OLE;
- выполнение активных действий, при возникновении разных событий (например, проверка введенного значения);
- специальные диалоги для выбора пользователем различных режимов работы;
- запуск других форм, отчетов, макросов, программ и т.д.

С объектом форма в СУБД MS Access. можно работать в 3-х режимах:

- в режиме **конструктора** возможно создание и изменение любых элементов формы;
- для проверки «поведения» элемента необходимо переключиться в режим **формы**;
- режим **таблицы** используется для табличного отображения источника записей формы (элементы формы, такие как: кнопки, заголовок и примечание формы, рамки объектов, в этом режиме на экран не выводятся).

2 Порядок выполнения работы

Создать БД для торгового предприятия. Для этого понадобится создать БД со следующими компонентами:

- Таблицы - для хранения информации о сотрудниках фирмы и товарах, для учета покупок товаров и учета продажи товаров.
- Формы - для заполнения данных в таблицы БД и получения оперативной информации.

2.1 Создание структуры таблиц БД

После запуска MS Access, программа выведет диалоговое окно, показанное на рисунке 2.

- 1) Щелкните на ссылке **Создать файл** (откроется список возможных действий);
- 2) Выберите ссылку **Новая база данных**.
- 3) В диалоговом окне **Сохранение файла** укажите имя нового файла базы данных MS Access (эти файлы имеют расширение - «*.mdb»).
- 4) Появится окно созданной базы. На нем Вы видите вкладки с названиями объектов БД и пока еще пустое окно элементов БД (в зависимости от версии MS Access вкладки могут размещаться сверху или с левой стороны окна). Нажмите кнопку <Создать>.
- 5) В окне выбора способа создания таблицы выберите конструктор.

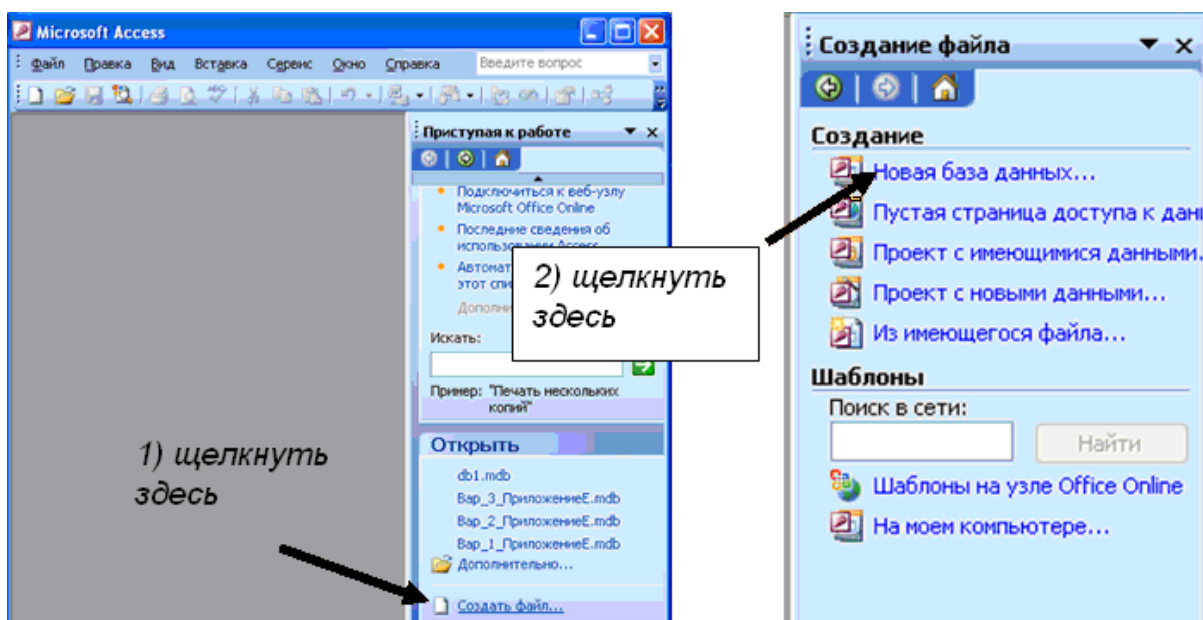


Рисунок 2 Диалоговое окно для создания базы данных

- б) Заполните строки в **конструкторе**, в соответствии с данными в таблице **Товары** (рисунок 3).

| Товары | |
|--------------|---------------------|
| Поле | Тип данных |
| Код товара | «счетчик», ключевое |
| Товар | «текстовый» |
| Цена продажи | «денежный» |
| Кол товара | «числовой» |

Рисунок 3 Названия и свойства полей таблицы Товары

Примечание* - поле Кол товара будет содержать данные о количестве товаров

7) Нажмите кнопку <Сохранить> на панели инструментов и дайте имя таблице. После закрытия конструктора, Вы увидите в окне базы созданную таблицу.

8) С помощью конструктора создайте таблицу **Фирмы**, набор и типы полей которой приведены в таблице на рисунке 4.

| Фирмы | |
|--------------|---------------------|
| Поле | Тип данных |
| Код фирмы | «счетчик», ключевое |
| Название | «текст» |
| Адрес | «текст» |
| Руководитель | «текст» |

Рисунок 4 Набор и типы полей таблицы Фирмы

9) Вновь откройте таблицу **Товары** в режиме конструктора. Для поля [Кол товара] задайте дополнительные свойства: **Условие на значение** установите «>=0», а также **Сообщение об ошибке** введите «Товар закончился на складе!» (рисунок 5). Такие установки не позволят пользователю вводить в поле отрицательные числа, при этом он будет получать заданное сообщение. Закройте таблицу щелчком на соответствующей кнопке строки заголовка.

| Имя поля | Тип данных | Описание |
|--------------|------------|---------------------|
| Код товара | Счетчик | |
| Товар | Текстовый | Наименование товара |
| Цена продажи | Денежный | Моя цена |
| Кол товара | Числовой | Общее количество |

| Свойства поля | |
|-------------------------|----------------------------|
| Общие | Подстановка |
| Размер поля | Длинное целое |
| Формат поля | |
| Число десятичных знаков | Авто |
| Маска ввода | |
| Подпись | |
| Значение по умолчанию | 0 |
| Условие на значение | >=0 |
| Сообщение об ошибке | Товар закончился на складе |
| Обязательное поле | Нет |
| Индексированное поле | Нет |
| Смарт-теги | |

Рисунок 5 Свойства поля Кол товара

10) С помощью конструктора создайте таблицы **Покупки** и **Продажи**, наборы и типы полей которых приведены в соответствующих таблицах на рисунке 6. В свойстве **Значение по умолчанию** полей [Дата покупки] и [Дата продажи] внесите «=Date()». Это встроенная функция VBA, возвращающая текущую дату. Таким образом, в каждую новую запись в указанных полях будет автоматически записываться текущая дата.

| Продажи | | Покупки | |
|--------------|--|--------------|---|
| Поле | Тип данных | Поле | Тип данных |
| Код продажи | «счетчик», ключевое | Код покупки | «счетчик», ключевое |
| Код товара | «числовой», размер: длинное целое | Код фирмы | «числовой», размер: длинное целое |
| Количество | «числовой», размер: целое | Код товара | «числовой», размер: длинное целое |
| Дата продажи | «дата/время» | Цена покупки | «денежный» |
| | | Количество | «числовой», размер: целое |
| | | Дата покупки | «дата/время» |

Рисунок 6 Наборы и типы полей создаваемых таблиц

11) В таблице **Покупки** установите **механизм подстановки** для автоматического заполнения поля [Код товара] из таблицы **Товары** и [Код фирмы] из таблицы **Фирмы**. При этом пользователь должен видеть не цифры кодов, а тексты.

Выполнение.

- Откройте в **конструкторе** таблицу **Покупки**, перейдите на поле [Код товара] и на вкладку **Подстановки** в окне свойств поля (рисунок 7). В поле **Тип элемента управления** выберите **Поле со списком**.

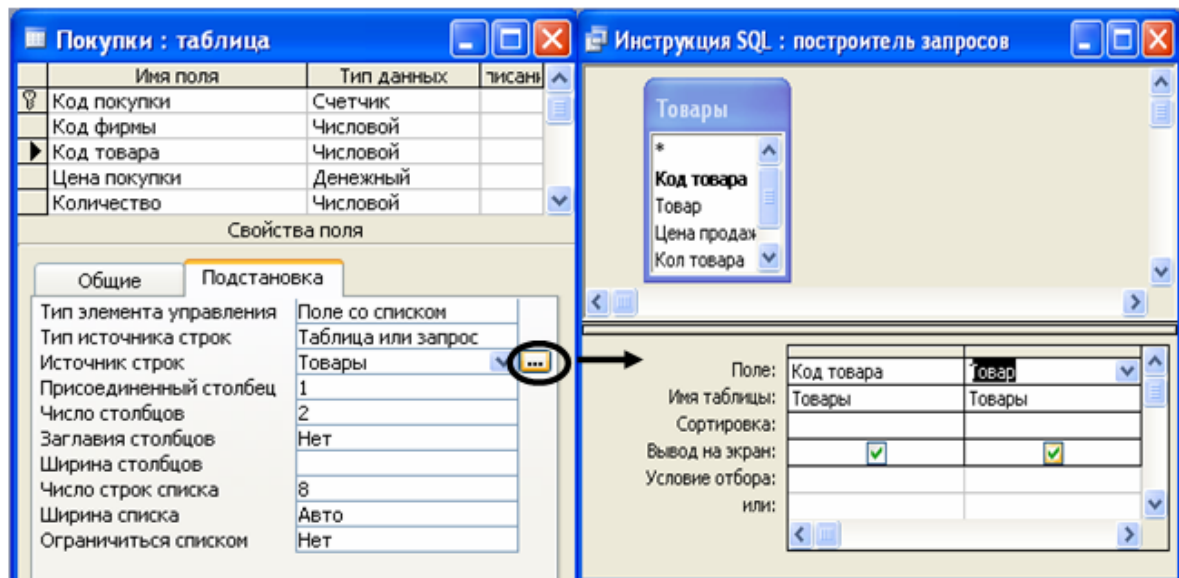


Рисунок 7 Создание инструкции SQL при создании механизма подстановки

- В поле **Источник строк** выберите таблицу **Товары**, затем щелкните на кнопке многоточие и подтвердите открытие окна **Инструкция SQL**. Для того чтобы в поле записывать цифру кода товара, а пользователь производил выбор по названию товара (что, согласитесь, значительно удобней), в запрос включите два столбца - [Код товара] и [Товар] из таблицы **Товары**. Закройте окно запроса и подтвердите сохранение изменений инструкции SQL и обновление свойств.

- Присоединенным (смотри одноименное свойство поля на вкладке подстановка) назначается первый столбец, т.о. в поле будут заноситься данные из поля [Код товара].

- Аналогично установите механизм подстановки для поля [Код фирмы].

Здесь впервые использован интерфейс создания запросов в MS Access. В данном случае создается запрос на выборку:

SELECT Товары.[Код товара], Товары.Товар FROM Товары;

Результатом выполнения такого запроса является объект в виде табличных данных (просмотр результата выполнения запроса возможен в режиме **Режим таблицы**), который в данном примере используется в качестве источника строк для раскрывающегося списка. Из этого списка выбираются данные для подстановки в текущую запись таблицы при ее заполнении.

12) Аналогично п.11 в таблице **Продажи** установите **механизм подстановки** для автоматического заполнения поля [Код товара] из таблицы **Товары**.

2.2 Установка связи между таблицами

13) Откройте **Схему данных** командой **Сервис - Схема данных**. В диалоговом окне перетащите мышью ключевые поля из таблиц **Товары** и

Фирмы на связанные с ними поля в таблицах **Покупки** и **Продажи** как это показано на рисунке 8.

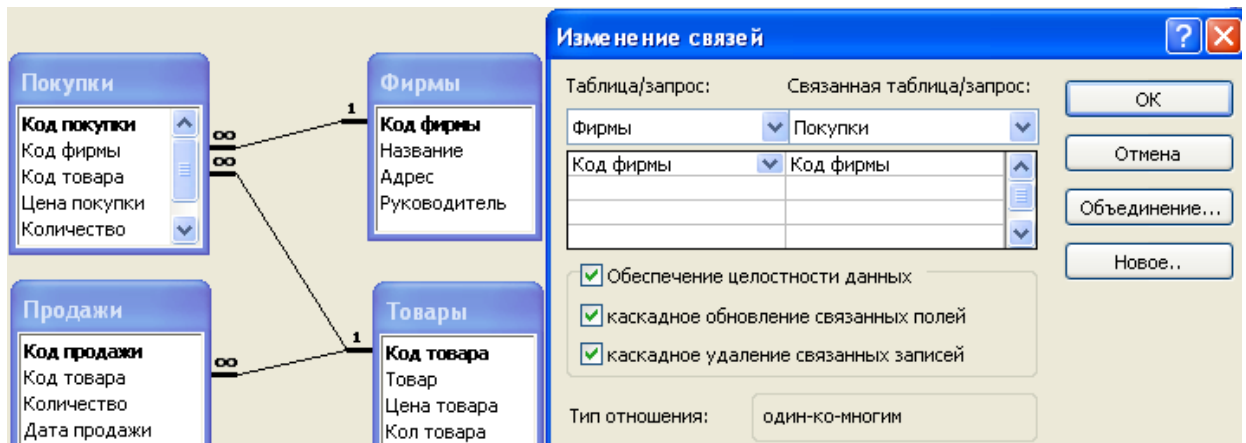


Рисунок 8 Схема данных и окно изменения свойств связи

При каждом соединении двух полей появляется диалоговое окно **Изменение связей**, в котором нужно включить флажки **Обеспечение целостности данных**, **Каскадное обновление связанных полей**, **Каскадное удаление связанных записей**. Эти настройки позволят поддерживать целостность БД для связи «один ко многим». Внимание! Если связь не устанавливается, то Вы ошиблись в типах полей.

2.3 Создание простых форм

14) Создайте простую автоформу в столбец с помощью следующей последовательности действий: выберите пункт меню **Создать**, далее - пункт **Мастер форм**, в качестве источника установите таблицу **Фирмы**. Доступные поля – все поля таблицы;

Внешний вид формы – в один столбец.

15) Заполните через форму **Фирмы** одноименную таблицу пятью записями, как показано на рисунке 9.

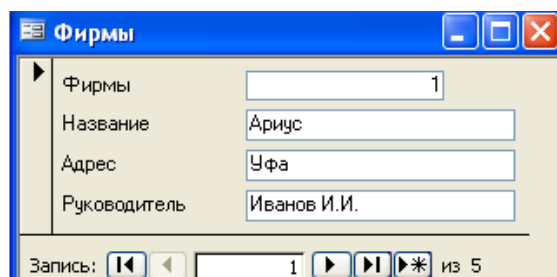



Рисунок 9 Заполнение таблицы данными через автоформу

16) Создайте форму для таблицы **Товары** с помощью мастера. Выполнение.

- В окне БД перейдите на вкладку **Формы**, нажмите кнопку <Создать> и выберите из списка **мастер форм**. Выберите в качестве источника таблицу **Товары**.

- Последовательно с помощью кнопки  перенесите из списка доступных полей поля: [Товар], [Цена продажи], [Кол товара]. Заметьте: (рисунок 10) поле [Код товара] осталось невыбранным. Нажмите <Далее>.

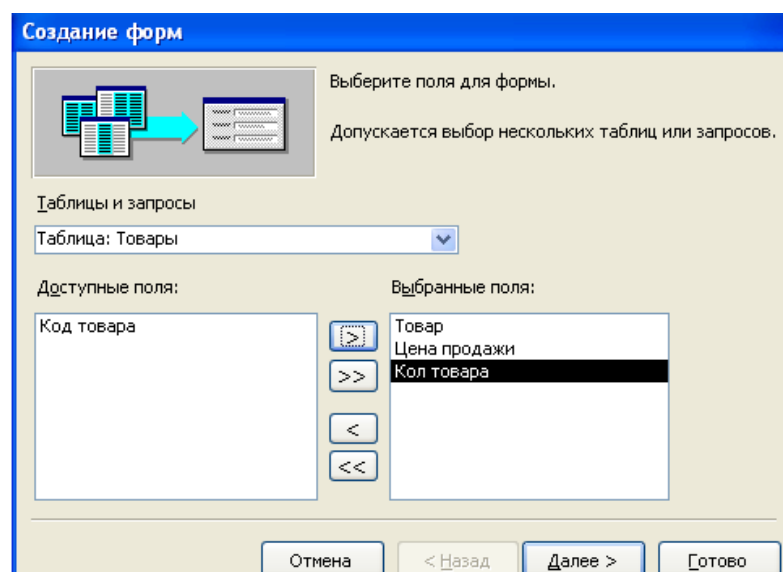


Рисунок 10 Выбор полей формы

- На втором шаге мастера выберите режим формы **Ленточный**.
- На 3-м шаге выберите любой, понравившийся Вам, стиль и нажмите кнопку <Готово>.
- Заполните через форму **Товары** одноименную таблицу пятью записями (рисунок 11). Внимание! Поле [Кол товара] заполнять не нужно.
- Самостоятельно создайте ленточную автоформу для таблицы **Продажи**.
- Создайте с помощью мастера форму в один столбец для таблицы **Покупки**. Внимание. [Код покупки] переносить не нужно. При сохранении формы назовите ее **Закупки**.

| Товар | Цена продажи |
|----------|--------------|
| Бананы | 45,00р. |
| Ананасы | 60,00р. |
| Груши | 50,00р. |
| Яблоки | 25,00р. |
| Виноград | 85,00р. |

Рисунок 11 Заполнение формы Товары

2.4 Работа в конструкторе форм

17) Перейдите в режим **конструктора** формы **Закупки** (выбрать объект **формы**, выделить форму **Закупки**, выбрать пункт меню **конструктор**).

18) После переименования формы (как объекта БД) мы получили несоответствие названия и подписи формы. Для исправления ситуации вызовите окно свойств формы, как это показано на рисунке 12, и на вкладке **Макет** измените свойство **Подпись** на «Закупки».

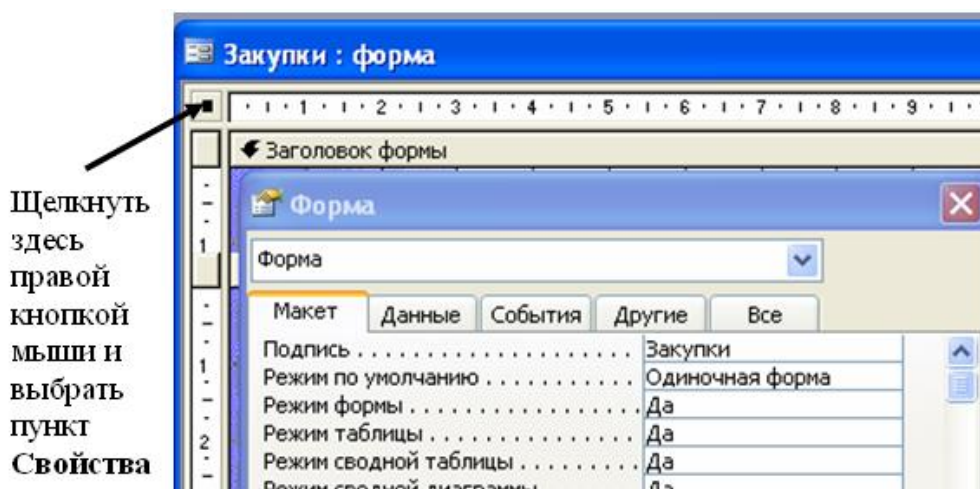


Рисунок 12 Свойства формы в режиме конструктора

19) Откройте форму **Фирмы** в режиме **формы** с помощью кнопки **Режим** в левом углу панели инструментов (рисунок 13).

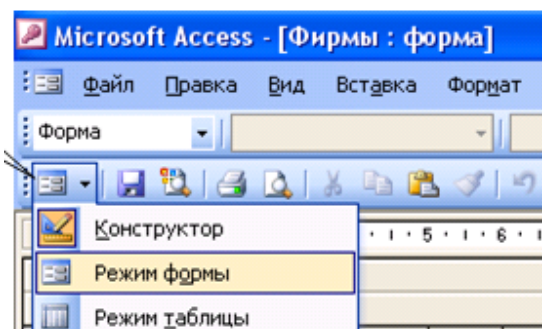


Рисунок 13 Переход между режимами отображения формы

При создании формы **Фирмы** мастер создал в ней поле ввода, присоединенное к полю [Код фирмы] и по умолчанию разрешил к нему доступ (курсор можно поместить в это поле). На самом деле редактировать это поле нельзя т.к. у него тип – Счетчик. Измените, форму **Фирмы**, так чтобы запретить доступ к полю [Код фирмы].

Выполнение.

- Откройте форму **Фирмы** в **конструкторе** и выделите элемент **код фирмы**.

На вкладке **Данные** установите свойства: **Доступ** – **Нет**; **Блокировка** – **Да**, как показано на рисунке 14.

- Перейдите в режим формы и проверьте выполненные изменения (курсор не должен устанавливаться в поле [Код фирмы]).

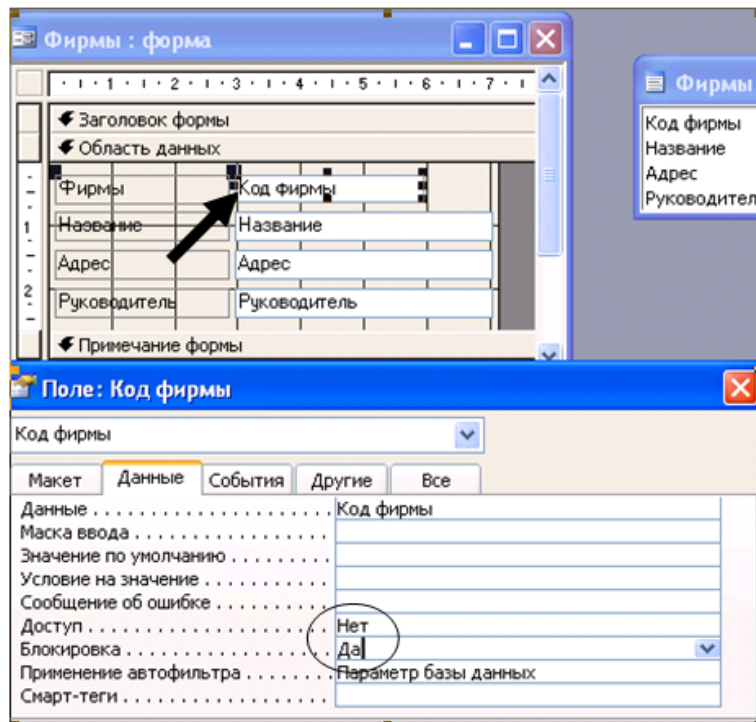


Рисунок 14 Изменение свойств поля Код фирмы

20) Добавьте в форме стандартную кнопку закрытия окна.
Выполнение.

- Переключитесь в режим **конструктор**.
- Убедитесь, что кнопка <Использовать мастера> на **Панели элементов** нажата.

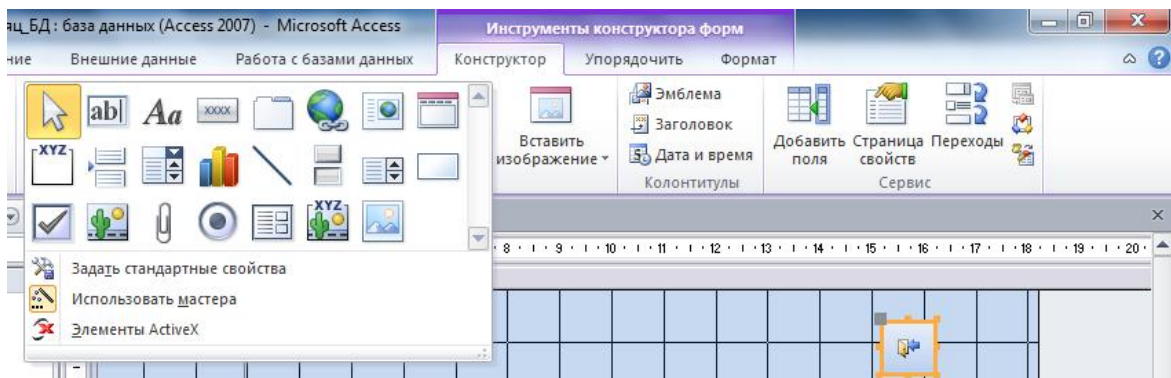


Рисунок 15 Панель добавления элементов формы

- Активизируйте элемент **кнопка** (на той же панели) и укажите место новой кнопки на форме.
- После запуска мастера (рисунок 16) выполните его шаги:
 - 1-й шаг – выбрать категорию: **Работа с формой**; действия: **Закрытие формы**;
 - 2-й шаг - выбрать рисунок для кнопки, или надпись «Стоп»;
 - 3-й шаг - Имя кнопки оставить по умолчанию и нажать <Готово> (рисунок 16).

- В окне свойств кнопки на вкладке **Другие** есть свойство **Всплывающая подсказка**, где Вы можете задать соответствующий текст.

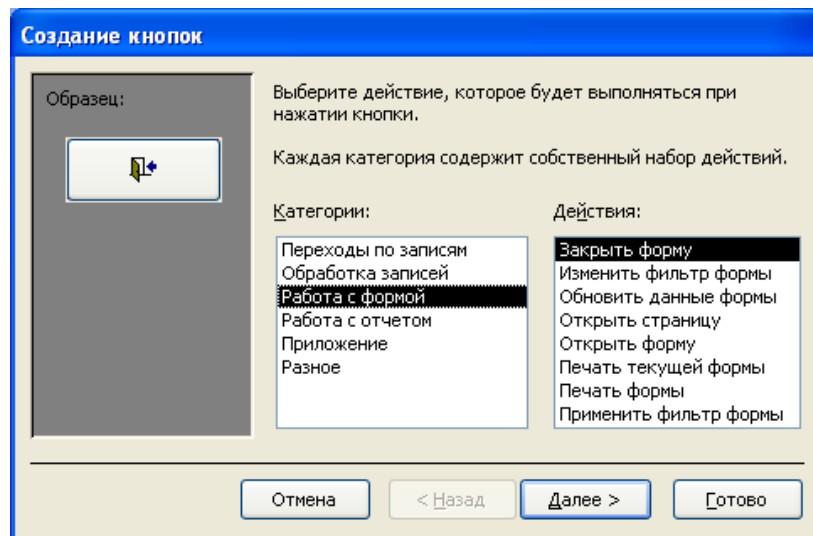


Рисунок 16 Диалог для выбора категории и действия кнопки

- Откройте форму в режиме **формы** и проверьте действие кнопки и подсказки к ней.

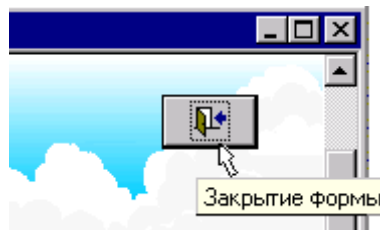


Рисунок 17 Готовый вид установленной кнопки закрытия формы

21) Закройте Access и оформите отчет, который должен содержать: название работы; цель и задачи; краткий ход выполнения работы; ответы на контрольные вопросы.

3 Вопросы для самопроверки знаний

1. Для чего предназначены объекты MS Access таблицы и формы?
2. Какие типы полей существуют в MS Access? Для хранения каких данных они предназначены?
3. Какие свойства полей имеются в MS Access? Что такое формат поля, условие на значение, значение по умолчанию?
4. Для чего предназначен первичный ключ таблицы?
5. Для чего предназначена схема данных?
6. Как установить связь между таблицами и целостность базы данных?
7. Для чего служат формы?
8. Как переключаться между режимами просмотра объектов БД?

9. Как вызвать окно свойств элементов формы?
10. Как на форме Фирмы создать кнопку для открытия формы Товары?

Лабораторная работа №2

«Создание графических форм»

Цель и задачи работы

Изучить принципы разработки сложных форм в СУБД MS Access.

1 Общие сведения

В режиме **конструктора формы** на экране появляется **Панель элементов** (рисунок 1). Первые две кнопки здесь имеют специальное назначение.

Кнопка <Выбор объектов> - по умолчанию всегда активна. В таком состоянии указатель мыши, при щелчке на любом из объектов, производит его выделение. Можно также растянуть рамку вокруг нескольких объектов, и тогда будут выделены все объекты, попавшие в зону рамки.

Кнопка <Использовать мастера> активизирует режим автоматического запуска мастеров по созданию сложных элементов управления.

Используя остальные кнопки на панели можно создавать новые элементы управления формы. Элементы управления формы - это объекты, из которых собственно и состоит вся форма. Каждый раз, при создании на форме нового элемента, ему присваивается уникальное собственное имя.

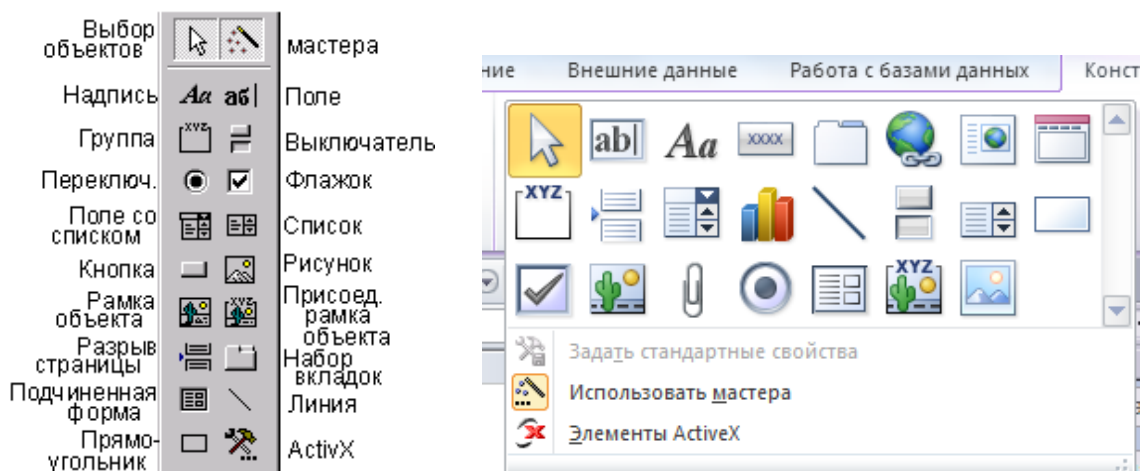


Рисунок 1 Кнопки панели элементов формы
(в разных версиях MS Access)

Все элементы формы (как и сама форма в целом) имеют свойства. Свойства выбранного объекта отображаются (и могут изменяться) в **окне свойств**.

2 Порядок выполнения работы

2.1 Запросы и создание сложных форм

Добавим в разработанные формы возможность просмотра изображений товара.

- 1) Откройте файл БД, созданный ранее.

Проверьте заполнение всех созданных ранее таблиц. Внесите в каждую таблицу порядка пяти записей.

- 2) Добавьте в форму **Товары** возможность показать товар в картинках.

Выполнение:

Подготовьте небольшие по размеру рисунки (в отдельном файле для каждого товара), которые будут использованы в формах для отображения всех товаров.

Например:



Рисунок 2 Образцы картинок для описания товаров

- 3) Откройте таблицу **Товары** в **конструкторе** и добавьте поле [Рисунок] с типом **Поле объекта OLE**. Сохраните изменения и закройте таблицу.

| Имя поля | Тип данных |
|--------------|------------------|
| Код товара | Счетчик |
| Товар | Текстовый |
| Цена продажи | Денежный |
| Кол товара | Числовой |
| Рисунок | Поле объекта OLE |

Рисунок 3 Добавление поля в таблицу Товары

- 4) Откройте в **конструкторе** форму **Товары**. Увеличьте область заголовка формы, для размещения на нем рисунка.

- Переместите надписи полей вниз (чтобы переместить их одновременно, воспользуйтесь для выделения полей клавишей <Shift>).

- Добавьте в область заголовка элемент **Присоединенная рамка объекта** (рис. 4-5).

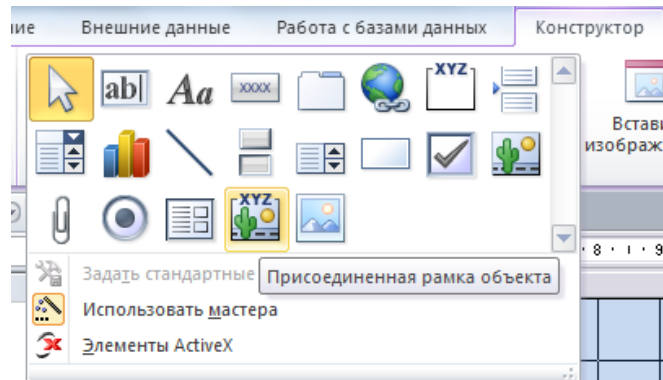


Рисунок 4 Выбор элемента Присоединенная рамка объекта

- Одновременно с рамкой появляется поле с надписью, его необходимо выделить и удалить.

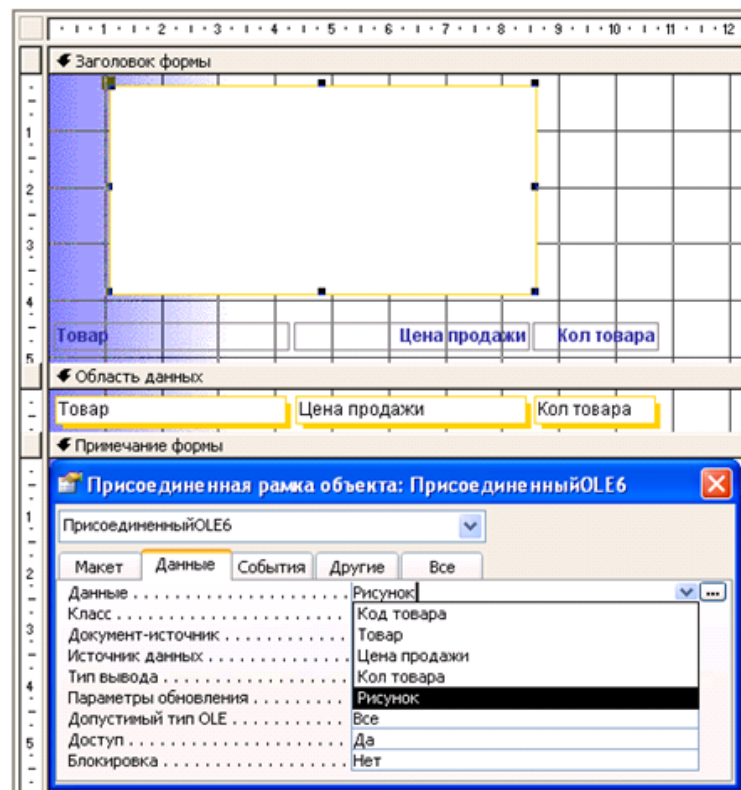


Рисунок 5 Установка свойств рамки

- Откройте свойства элемента и в свойстве **Данные** установите через открывающийся список имя поля [Рисунок].
Внимание! Не надо заполнять это свойство вручную! Его необходимо выбрать из раскрывающего списка полей.
- На вкладке **Макет** установите свойство **Установка размеров - По размеру рамки**.
- Сохраните изменения.

5) Перейдите в режим **формы** для формы **Товары**.

Далее необходимо задать рисунки для товаров, причем каждый рисунок устанавливается отдельно для каждой строки (т.е. каждого товара таблицы).

В контекстном меню на рамке объекта выделите команду **Вставить объект...** Далее выберите Создать новый, тип - Bitmap Image.

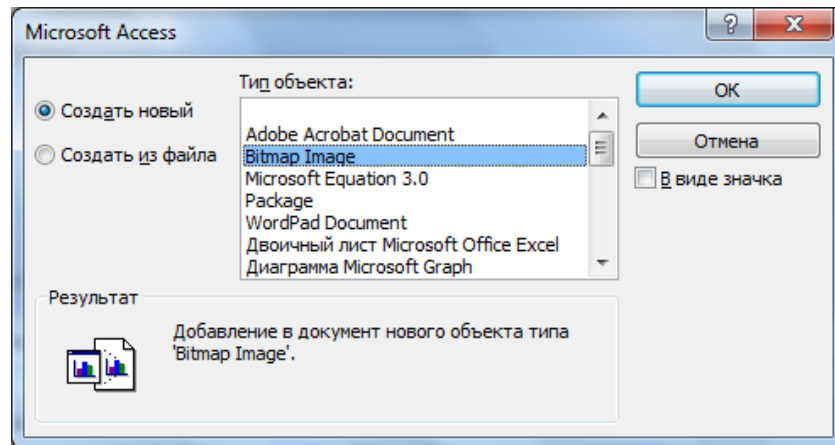


Рисунок 6 Окно выбора параметров типа объекта

В открывшемся при этом редакторе Paint создайте рисунок текущего товара либо вставьте подготовленный ранее рисунок с помощью команды Вставить/ Вставить из... (указать адрес соответствующего файла с изображением).

Нажмите кнопку <Готово> (рисунок 7).




Рисунок 7 Результат присоединения рамки и добавления рисунка

6) В форме **Продажи** в область заголовка выведите картинку для текущего товара.

Картинку товара только показывать (запретить изменение с помощью установки соответствующих параметров свойств **Доступ** и **Блокировка**).

Поле объекта OLE, содержащее рисунок товара, находится в таблице **Товары**. В то же время в таблице **Продажи** имеется поле [Код товара]. Поэтому, для реализации поставленной задачи, нужно создать запрос, который свяжет эти две таблицы, и использовать этот запрос в качестве источника записей для формы **Продажи**.

Выполнение.

- Откройте форму **Продажи** в конструкторе.
- Выведите **окно свойств формы** (используя правую клавишу мыши). На вкладке **Данные** свойство **Источник записей** содержит название таблицы **Продажи**. Его нужно заменить инструкцией SQL, которую можно создать с помощью **построителя запросов**. Для его вызова нажмите кнопку с многоточием, появится окно **конструктора** с одной таблицей **Продажи** внутри.
- Для добавления в запрос новой таблицы используйте кнопку на панели инструментов  (**Отобразить таблицу**). После добавления таблицы **Товары**, закройте окно добавления, чтобы вернуться в окно **конструктора запросов**. Таблицы связаны по полям [Код товара]. Внимание! Если это не так, вернитесь в **схему данных** и установите связь.
- В первое поле запроса через открывающийся список введите «Продажи.*» - это означает **все поля таблицы Продажи**.
- Во второе поле запроса введите поле [Рисунок] таблицы **Товары**.

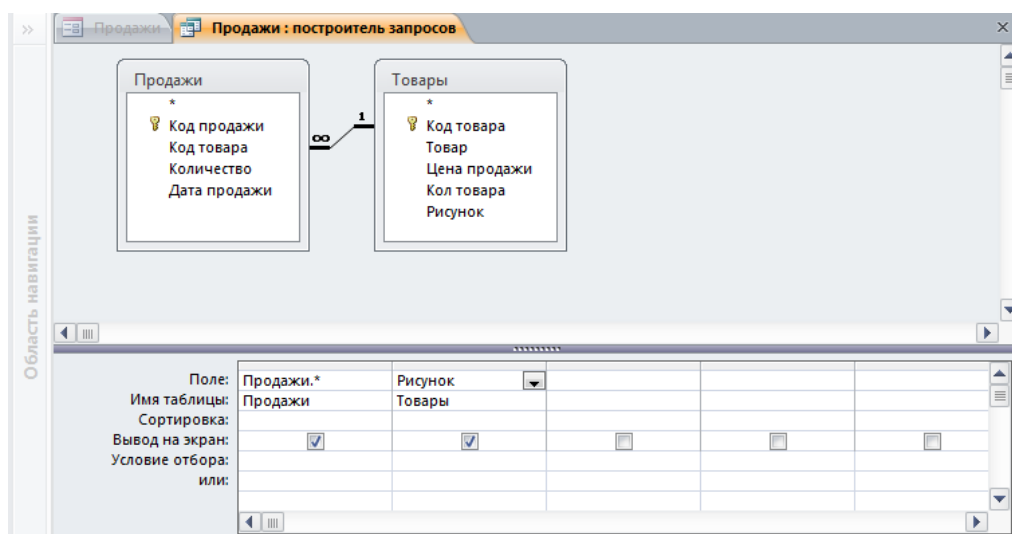


Рисунок 8 Образец выполнения

Примечание: Для проверки правильности переведите конструктор запроса в режим **таблицы**. Поле [Рисунок] (рисунок 9) должно присутствовать.

| | Товар | Товар | Цена продажи | Кол товара | Рисунок |
|-----|-------|---------|--------------|------------|--------------|
| ▶ + | 1 | Яблоки | 45,00р. | 70 | Bitmap Image |
| + | 2 | Ананасы | 60,00р. | 1 | Bitmap Image |

Рисунок 9 Конструктор запросов в режиме таблицы

- Закройте **построитель** и подтвердите изменения. Теперь в свойстве **Источник записей** будет находиться следующий текст SQL-запроса:

```
SELECT Продажи.*, Товары.Рисунок
FROM Продажи
INNER JOIN Товары ON Продажи.[Код товара]=Товары.[Код товара];
```

7) Для формы Продажи добавьте **присоединенную рамку объекта** и присоедините ее к полю [Рисунок]. Делается это точно так же, как Вы уже проделали ранее (**элемент Присоединенная рамка объекта**).

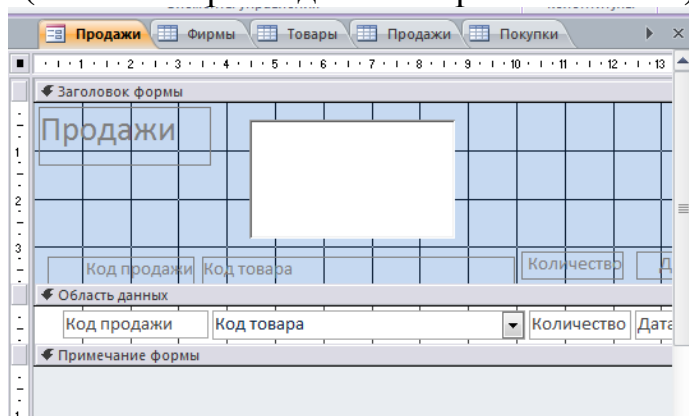
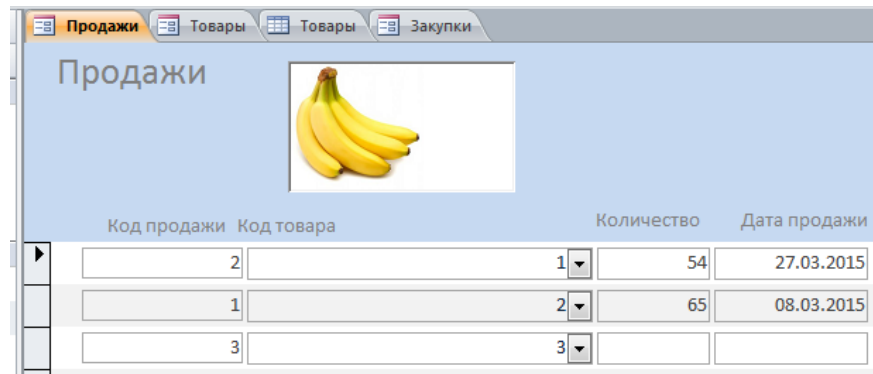


Рисунок 10 Образец выполнения

Примечание: Откройте свойства рамки и в свойстве **Данные** установите через открывающийся список имя поля [Рисунок].

Если в списке свойства **Данные** отсутствует название поля [Рисунок], то Вы ошиблись в создании **Источника записей**.

Проверьте работу формы Продажи – в ней должны выводиться соответствующие товарам картинки.



| Код продажи | Код товара | Количество | Дата продажи |
|-------------|------------|------------|--------------|
| 2 | 1 | 54 | 27.03.2015 |
| 1 | 2 | 65 | 08.03.2015 |
| 3 | 3 | | |

Рисунок 11 Вид формы Продажи

- 8) Оформите отчет, который должен содержать: название работы; цель и задачи; краткий ход выполнения работы; ответы на контрольные вопросы.

3 Вопросы для самопроверки знаний

1. Для чего предназначены формы?
2. Какие типы форм существуют в MS Access?
3. Поясните различия между созданием форм с помощью Мастера и Конструктора.
4. Назовите основные разделы (области) конструктора форм.
5. Какие свойства элементов формы существуют в MS Access?
6. Для чего предназначено свойства **Доступ** и **Блокировка**?
7. Какие существуют способы создания графического объекта, отображаемого в форме?

Лабораторная работа 3

«Разработка форм и запросов с проведением вычислений»

Цель и задачи работы

Изучить принципы разработки сложных запросов и форм в СУБД MS Access.

1 Общие сведения

Запрос - это текстовая строка, содержащая инструкцию SQL. Такую инструкцию можно ввести с клавиатуры (в режиме **Режим SQL**), или создать с помощью конструктора запросов.

Рассмотрим пример создания инструкции SQL на примере выполнения подсчета нового значения общего количества товара (поле [Кол товара] из таблицы **Товары**). Значение общего количества товара можно подсчитать применив формулу:

$O_kol + (n_kol - s_kol)$,

где O_kol - общее количество товара;

n_kol - новое значение в поле [Количество] (это сведения о закупке);

s_kol - старое значение поля [Количество].

Эта формула описывает самый общий случай, т.к. она предусматривает все возможные случаи:

- когда вносятся изменения в существующую запись. В этом случае все операнды формулы нужны и имеют значение;
- когда заносится новая запись, операнд s_kol не будет иметь значения и его просто надо обнулить.
- Для получения значений для этих переменных в MS Access используются свойства элементов управления формы:
- поле [Кол товара] - содержит общее количество (O_kol);
- свойство поля [Количество].OldValue - вернет нам значение из этого поля до внесения в него изменений (s_kol);
- свойство поля [Количество].Value - вернет нам значение из этого поля после внесения в него изменений (n_kol).

Теперь данная формула приобретет вид:

$[Кол\ товара] + ([Количество].Value - [Количество].OldValue)$

Однако арифметические операции нельзя производить со значением Null (т.е. значение не присвоено). Во избежание такой ошибки нужно использовать функцию: Nz(«объект», «значение_если_объект_Null»).

Получаем следующий вид формулы:

$[Кол\ товара] = (Nz([Кол\ товара], 0) + (Nz([Количество].Value, 0) - Nz([Количество].OldValue, 0)))$

Таким образом, значение в поле [Кол товара] таблицы **Товары** будет изменяться после внесения изменений в поле [Количество]. Для этого в MS

Access следует изменить такие свойства элементов управления, как **События**.

2 Порядок выполнения работы

В данной работе необходимо расширить задачи для форм **Закупки** и **Продажи**:

- необходимо получать оперативные сведения о наличии товара на складе (после его дополнительной закупки и (или) продажи);
- из одной и той же фирмы можно закупать различные товары, поэтому одна накладная может быть выписана на несколько покупаемых товаров. Для облегчения работы оператора, вносящего в форму **Закупки** данные о номере накладной и коде фирмы, необходимо организовать автоматическое занесение этих параметров в каждую новую запись о закупках товара;
- необходимо при покупке товара автоматически формировать общую стоимость закупки.

1) Форма **Закупки** построена на основе таблицы **Покупки**. Создайте в этой таблице поле для номера накладной.

Выполнение.

- Откройте таблицу **Покупки** в **конструкторе** и добавьте поле [№ накладной] с типом данных **Текст**. Сохранив изменения в таблице **Покупки**, закройте ее.

2) Создайте запрос, который позволит связать поля двух таблиц.

Выполнение.

- В окне БД перейдите на вкладку **Запросы** и нажмите кнопку <Создать>. В окне выбора способа создания укажите **конструктор**.

- В следующем окне добавьте в запрос таблицы **Покупки** и **Товары**, как показано на рисунке 1.

- Создайте следующие поля запроса:

Покупки.* - это специальный указатель, обозначающий все поля таблицы **Покупки**;

Цена продажи – цена из таблицы **Товары**;

Кол товара - общее количество товара на складе из таблицы **Товары**;

Товар - это поле содержит текст «название товара» из таблицы **Товары**;

Стоимость: [Цена покупки]*[Количество] - это вычисляемое поле, текст которого нужно написать самостоятельно вместо имени поля.

- Сохраните созданный запрос с именем **Закупки**.

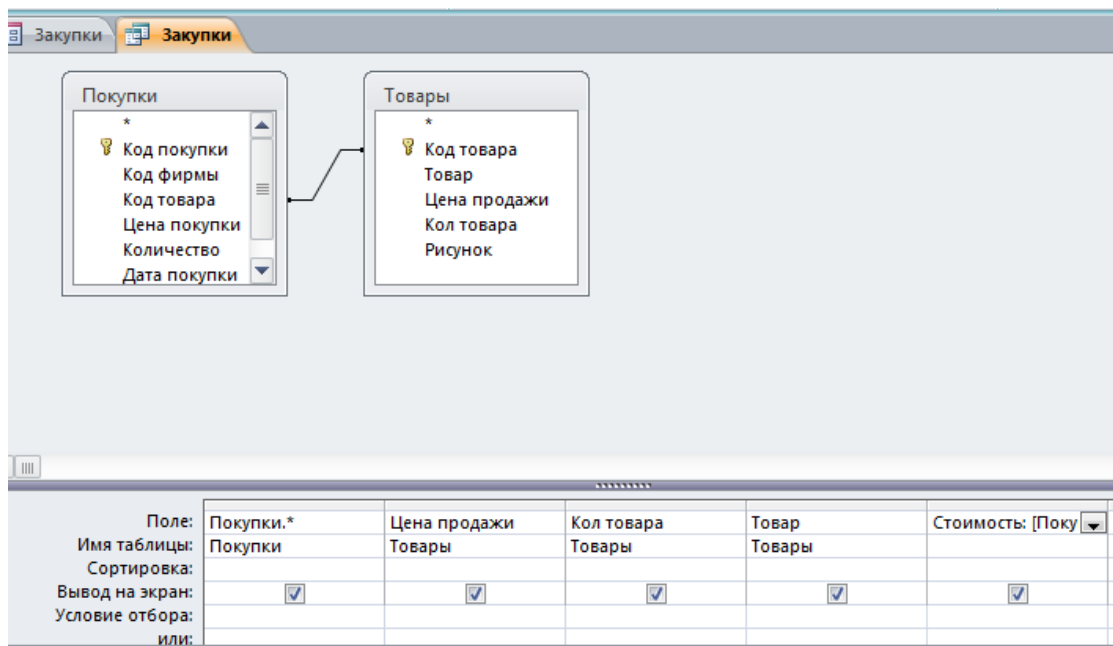


Рисунок 1 Построитель запросов для изменения источника для формы

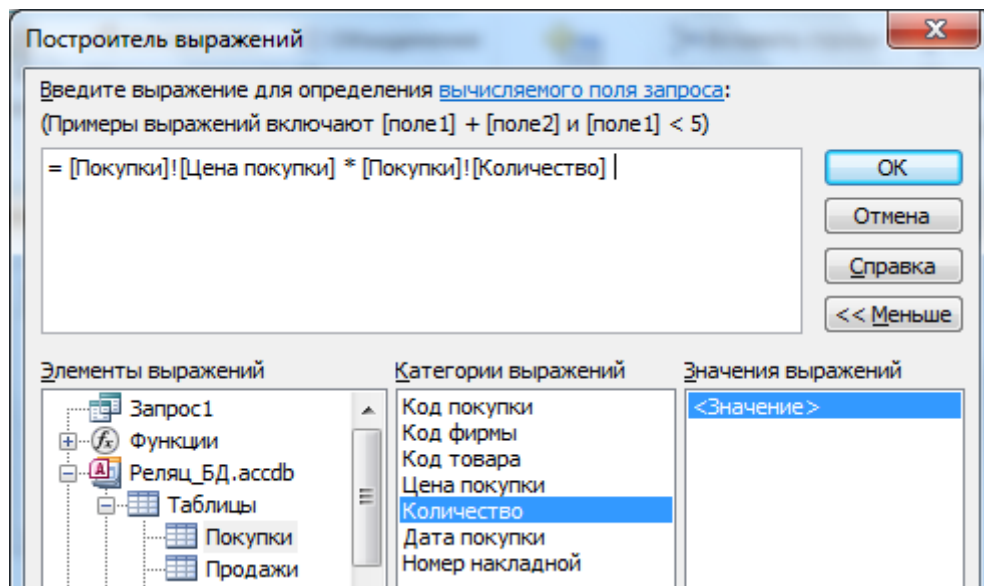


Рисунок 2 Построитель выражений

Проверьте работу (выполнение) построенного запроса.

3) Измените форму **Закупки**, так чтобы при покупке товара автоматически формировалась общая стоимость закупки, как [Количество]*[Цена].

Выполнение:

- Откройте форму **Закупки** в конструкторе и окно **Свойства** формы, щелкнув правой кнопкой по черному квадратику в верхнем углу. На вкладке **Данные** измените свойство **Источник записей** (вместо таблицы **Покупки** поставьте запрос **Закупки**).

- Увеличьте размер области данных и перетащите в него новое поле [Стоимость] из списка доступных полей (рисунок 3). Внимание! Если этого

окна нет на экране, выполнить команду **Конструктор – Добавить поля**. Так как для расчета стоимости была введена формула в источнике записей, то в форме эта величина будет автоматически формироваться при вводе данных в поле [Количество].

- Также добавьте в форму поле **Номер накладной**

Рисунок 3 Область данных формы **Закупки** и список доступных полей после изменения источника записей

4) Выполните такие изменения в форме **Закупки**, чтобы номер накладной и фирма заносились оператором один раз и, затем (во всех последующих записях) устанавливались автоматически.

Выполнение:

- В **Панели элементов** выделите инструмент **поле ab|** и разместите его в области **Примечание формы**. Появится **Свободный элемент** и его подпись (например, подпись - «поле13»). Подпись выделите отдельно и удалите.
- Свободному элементу дайте имя **u_nom** (в свойствах элемента вкладка **Другие**, поле **Имя**).
- В элемент **u_nom** пользователь может записывать набор символов, соответствующих новому номеру накладной. Для того чтобы этот текст автоматически попадал в поле [№ накладной], при создании новой записи, нужно в свойстве поля [№ накладной] **Значение по умолчанию** (вкладка **Данные** или **Все**) ввести вручную или сформировать с помощью построителя (кнопка с многоточием) имя элемента **=u_nom** (рисунок 4).

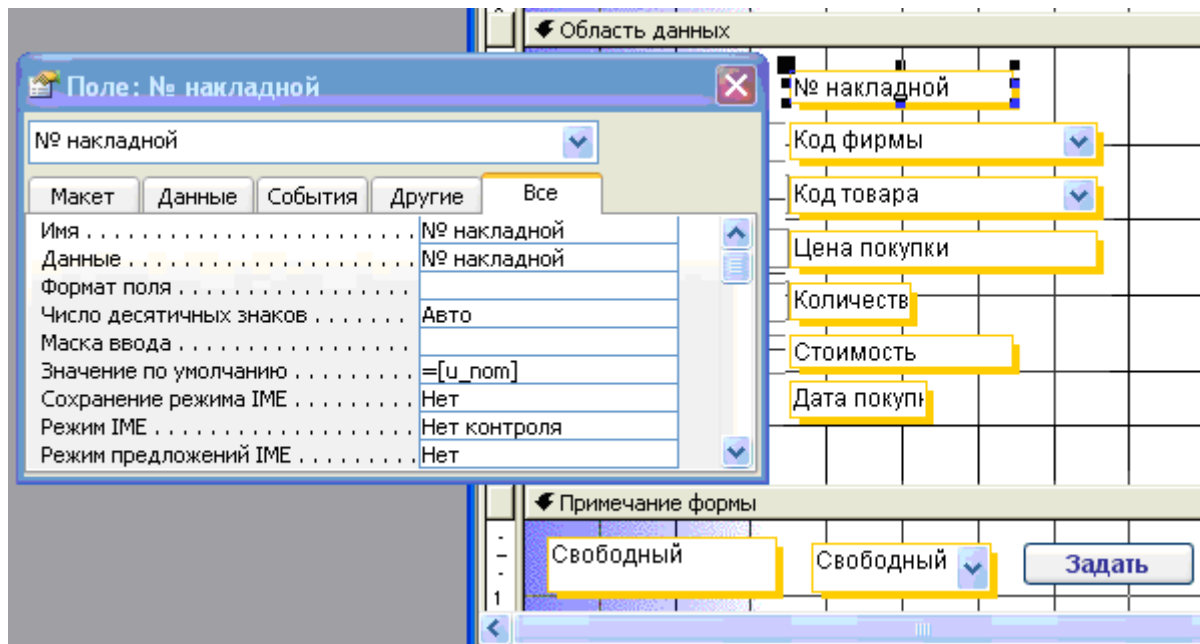


Рисунок 4 Свойства поля Номер накладной

5) Теперь подобным же образом в примечаниях формы создайте свободный элемент **Поле со списком** для автоматического заполнения названия фирмы и задайте ему имя **u_firm**.

Этапы создания поля со списком отображены на рис. 5-7

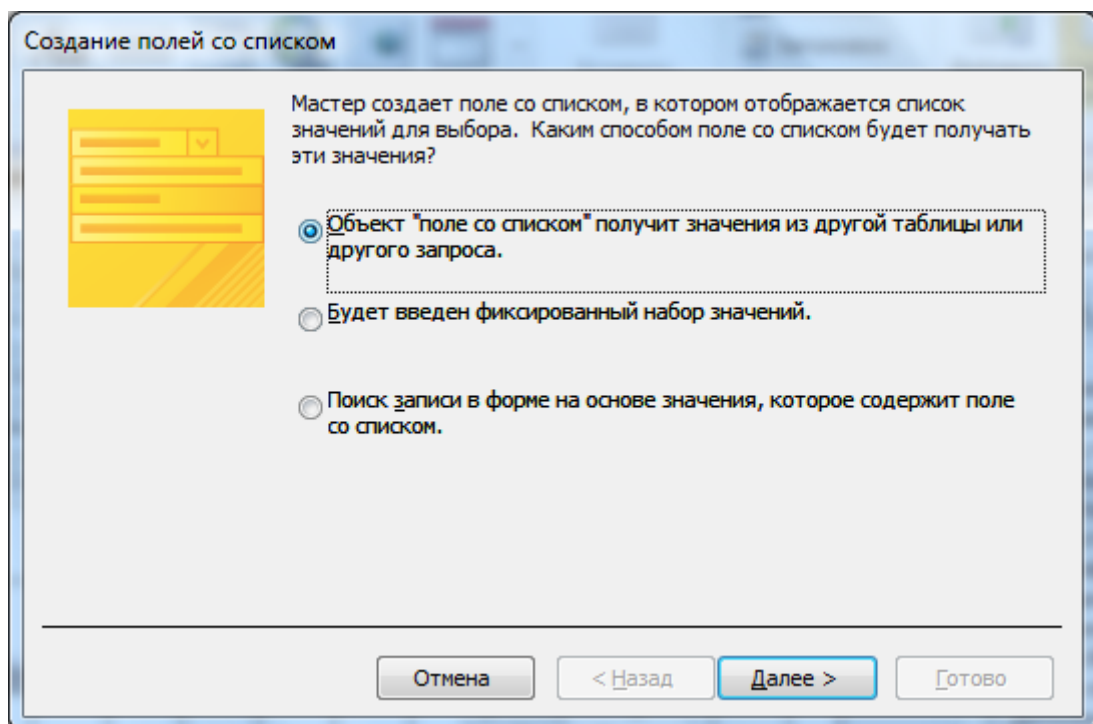


Рисунок 5 Выбор вида источника данных

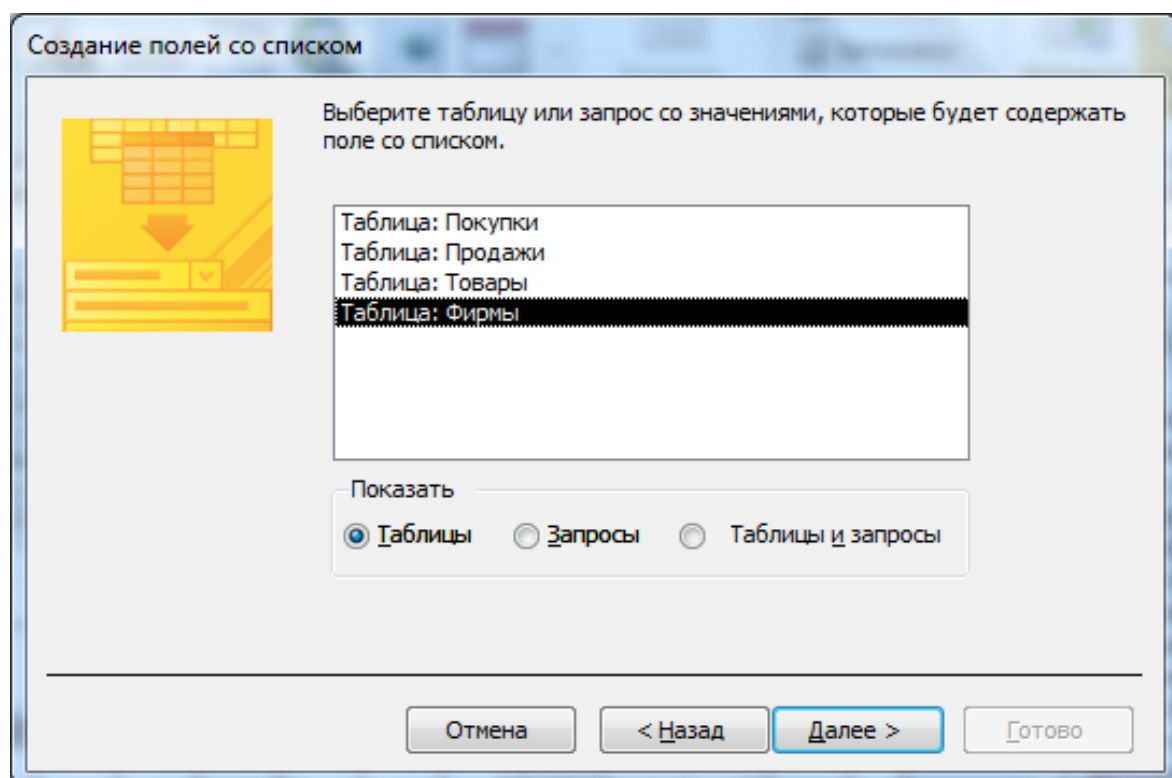


Рисунок 6 Выбор таблиц-источников

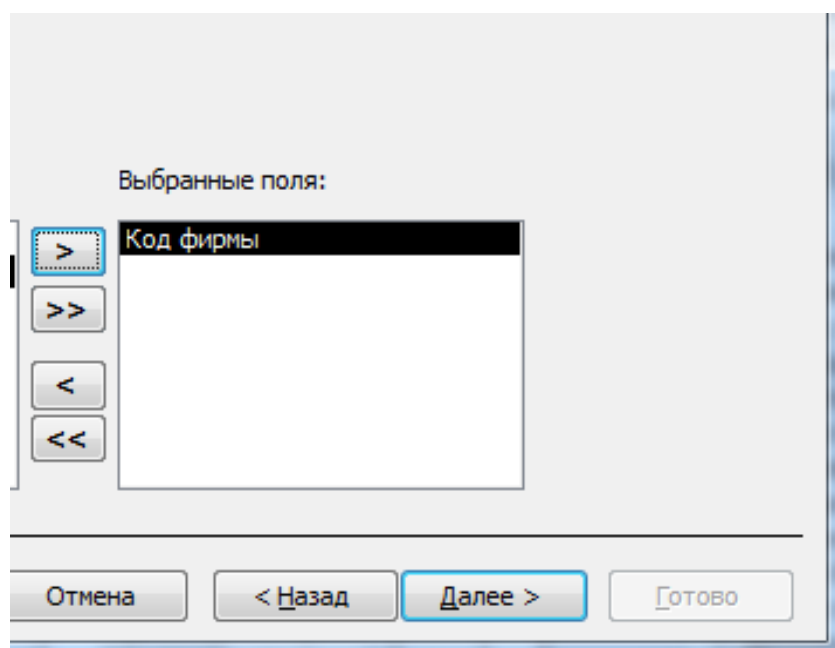


Рисунок 7 Настройка доступных полей

Завершите работу по созданию поля.

Настройте для поля Код фирмы свойство Значение по умолчанию:

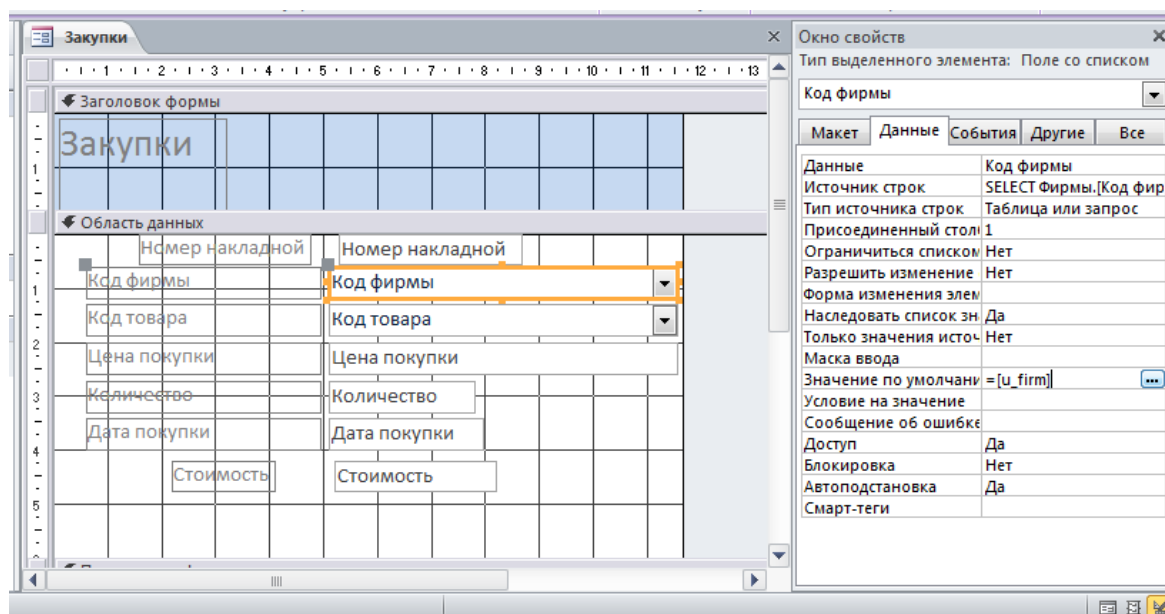


Рисунок 8 Образец настроек параметров элементов формы

б) При открытии формы элемент **u_nom** и **u_firm** не имеют значения, поэтому после того, как оператор запишет в первый элемент номер накладной, а во втором элементе выберет код фирмы, необходимо произвести обновление данных формы.

Выполнение.

- В области **Примечание формы** создайте кнопку (при загрузке мастера создания кнопок отменить его работу кнопкой отмены).

- в окне **свойств кнопки** на вкладке **Все** введите имя и подпись **Задать**.

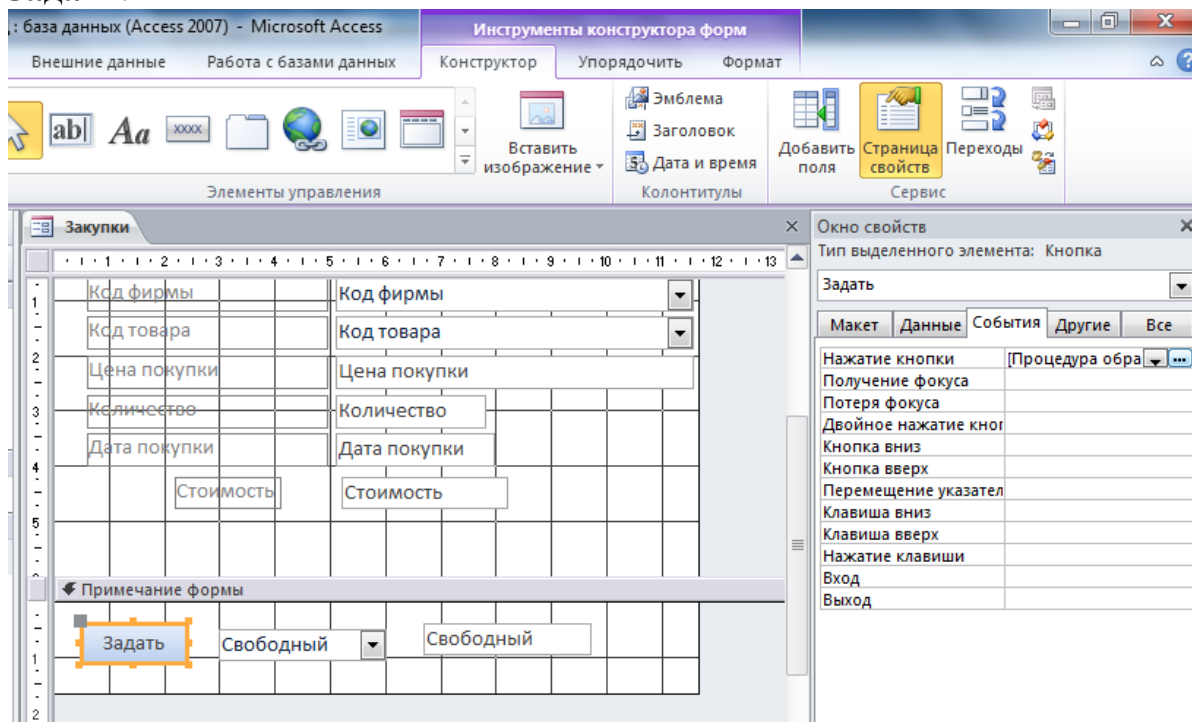


Рисунок 9 Образец выполнения

- В свойстве кнопки на вкладке **События** выберите **Нажатие кнопки**, установите **Процедуру обработки событий**, затем нажмите кнопку вызова **конструктора** для редактирования модуля (в той же строке правая кнопка с изображением многоточия **...**). После открытия окна модуля, в ней находятся две строки для этой процедуры - первая и последняя. Введите текст **Me.Refresh**, как показано на рисунке 10.

```
Private Sub Задать_Click()
Me.Refresh
End Sub
```

Рисунок 10 Процедура обработки события для кнопки Задать


- С помощью элемента Надпись подпишите соответствующие поля ввода данных как **Номер фирмы** и **№ накладной**

The image shows a portion of a form. On the left is a blue button with the text 'Задать'. To its right is a dropdown menu with the label 'Номер фирмы' above it. Further right is a text input field with the label '№ накладной' above it.

Рисунок 11 Возможный вид нижнего фрагмента формы

7) Перейдите в режим просмотра формы и проверьте правильность работы кнопки и созданных полей ввода.

Внимание. Для того, чтобы кнопка работала правильно следует соблюдать следующие правила:

находясь на последней заполненной записи (для быстрого перехода можно нажать кнопку  в нижней части формы), ввести новые значения номера накладной и кода фирмы и нажать кнопку <Задать>.

Затем перейти на новую запись, где уже будут выставлены новые значения. Кнопка <Задать> будет работать для обоих элементов сразу, т.к. она обновляет сразу всю форму. При необходимости, вернитесь в режим конструктора и исправьте ошибки.

8) Добавьте в область заголовка формы **Закупки** поле [Товар], как показано на рисунке 12.

Выполнение. Расширьте область **Заголовков формы** и разместите там поле **Товар**, переместив его из списка доступных полей. Внимание. Если просмотреть свойства поля **Данные**, то там будет указан **Товар**.

- Закройте доступ к вновь созданному элементу (свойство **Блокировка** - **ДА**; свойство **Доступ** - **НЕТ**).

9) Аналогично пункту 11 разместите в области заголовка формы поля [Цена продажи] и [Кол товара] и задайте их свойства (рисунок 12).

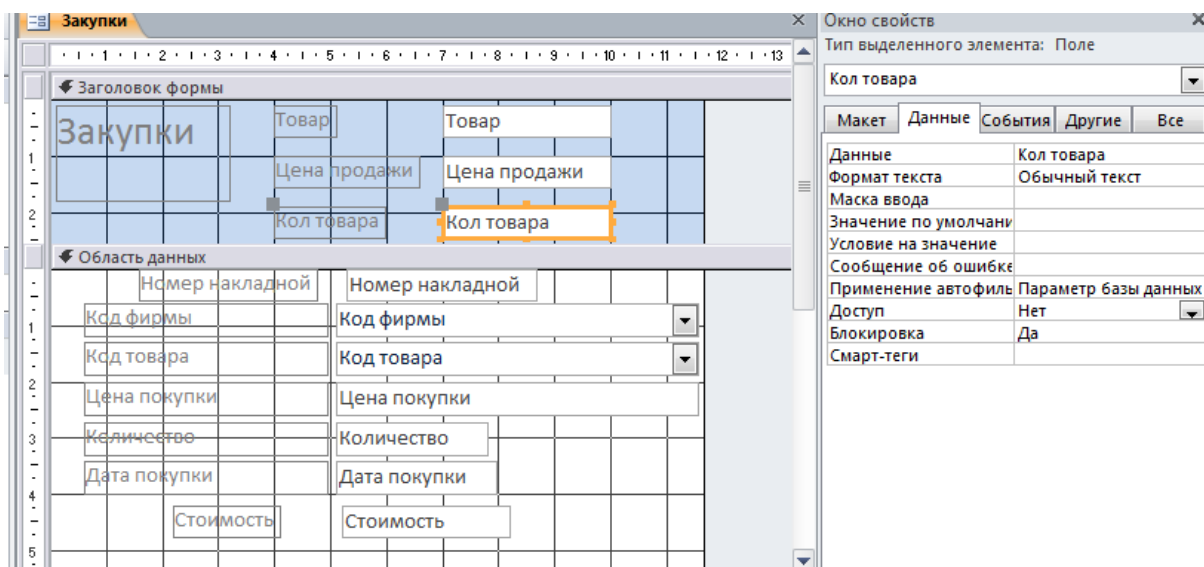


Рисунок 12 Форма закупки

10) Просмотрите выполненные изменения в режиме формы, при необходимости вернитесь в режим **конструктора** и скорректируйте размеры и расположение полей. Внимание! Чтобы передвинуть отдельно друг от друга поле или его подпись, необходимо установить курсор в верхний левый маркер нужного элемента.

11) Выполните изменения в форме такие, что при вводе оператором закупочной цены, производилась бы проверка: закупочная цена не должна превышать цену продажи (последняя установлена в таблице **Товары**). Для этого в режиме **конструктора** у элемента управления формы **цена покупки** свойство **Условие на значение** установите: «<=[цена продажи]» (меньше или равно цене из таблицы **Товары**). В свойство **Сообщение об ошибке** записать текст, например: «При такой закупке мы понесем УБЫТКИ! Установите меньшую закупочную цену».

12) Реализуйте автоматическое изменение общего количества товара (поле [Кол товара] в таблице **Товары**), при занесении новых данных в форму **Закупки**. Если этого не сделать, оператору, после ввода новых данных о произведенных закупках, нужно переключаться в таблицу **Товары** и вручную редактировать это поле. Примечание. Обратите внимание, что эта операция возможна теперь потому, что источником записей формы является запрос, который имеет поле [Кол товара] из таблицы **Товары**.

Выполнение:

- Находясь в конструкторе формы **Закупки** поменяйте местами поля (вместе с их подписями) [Количество] и [Цена покупки].
- В **Окне свойств** поля [Количество] для события **После обновления** установите **Процедура обработки событий**, нажмите кнопку

вызова **конструктора** для редактирования модуля, впишите формулу, как показано на рисунке 13, и затем закройте окно MS Visual Basic.

```
Private Sub количество_AfterUpdate()  
    [Кол товара] = Nz([Кол товара], 0) + (Nz([Количество].Value, 0) -  
    Nz([Количество].OldValue, 0))  
End Sub
```

Рисунок 13 Процедура подсчета количества товара после новой закупки

13) Откройте форму **Закупки** в режиме формы и заполните ее несколькими записями, с учетом того, что один и тот же товар может поставляться разными фирмами. Просмотрите изменения в таблице **Покупки** и **Товары** и проанализируйте их. Обратите внимание на то, что общее количество товара в форме изменяется сразу, а в таблице **Товары** изменяется лишь после перехода на другую запись.

14) Закройте Access и оформите отчет, который должен содержать: название работы; цель и задачи; краткий ход выполнения работы; ответы на контрольные вопросы.

3 Вопросы для самопроверки знаний

- 1) Какие управляющие элементы предлагает панель элементов?
- 2) Для чего предназначена кнопка «Использовать мастера» панели элементов?
- 3) Для чего служат запросы?
- 4) Какова последовательность действий при конструировании запроса?
- 5) Как создать вычисляемое поле в запросе?
- 6) Чем отличаются присоединенное и свободное поле?
- 7) Что означает каждый из элементов процедуры подсчета количества товара после новой закупки? Для чего предназначена эта процедура?

Лабораторная работа №4


Разработка интерфейса БД

Цель и задачи работы


Изучить принципы создания интерфейса для более удобного взаимодействия объектов БД

1 Общие сведения

По умолчанию записи в таблице Access выводятся в последовательности, определенной значениями первичного ключа этой таблицы. Если ключ не определен, то записи выходят в последовательности их ввода. Для изменения последовательности записей в таблице по значению одного поля используется сортировка или фильтр.

Для сортировки записей таблицы по значениям активного поля **по возрастанию** или **по убыванию** используются кнопки  или одноименные команды пункта **Записи**. В таблице одновременно при помощи клавиши Shift можно выбрать несколько соседних столбцов для сортировки, при этом сортировка записей будет начинаться с самого левого столбца из выделенных.

Для выполнения сортировки одновременно по нескольким полям удобнее использовать фильтр, который включается командой **Записи-Фильтр - Расширенный фильтр**. В открывшемся бланке необходимо расположить поля для сортировки в необходимом порядке. Например, чтобы отсортировать таблицу **Студенты** сначала по факультетам, затем студентов каждого факультета – по курсам и в последнюю очередь студентов каждого курса по фамилиям, необходимо в бланке запроса поместить последовательно поля [Факультет], [Курс], [Фамилия]. В строке **Сортировка** бланка запроса выбрать для каждого поля нужный параметр сортировки. После заполнения бланка сортировки необходимо выполнить команду **Записи - Применить фильтр**.

Замечание. Сортировать таблицу по нескольким полям имеет смысл только тогда, когда в поле, по которому сортируются записи в первую очередь, есть повторяющиеся значения. Фильтрация применяется, когда необходимо выбрать в одной таблице данные по простому критерию. Для фильтрации используются кнопки  панели инструментов. Первая кнопка выполняет фильтр по выделенному значению в поле. Например, в поле Товар найдены наименования товаров: Сухари, Сушки, Булочки по выделенной букве «у» с помощью кнопки **Фильтр по выделенному**. Вторая кнопка позволяет изменить текущие настройки фильтра на новые, например, при нажатии на эту кнопку текущий критерий «Like "*у*» можно заменить на Like "*х*", затем нажать на третью кнопку **Применение фильтра**, тогда в таблице останутся товары с наименованиями «Сухари» и «Хлеб». Кнопка **Применение фильтра** при этом превратится в кнопку **Удалить фильтр**.

Все рассмотренные результаты можно также получить, используя команды пункта меню **Записи**.

Таким образом, сортировка и фильтрация данных в СУБД MS Access может выполняться стандартными методами с помощью кнопок панели инструментов, либо командами пункта меню **Записи**. Однако при определенных условиях работы эти элементы управления могут отсутствовать. Наша задача состоит в том, чтобы организовать эти функции в отсутствии меню и панелей инструментов.

2 Порядок выполнения работы

В данной работе необходимо расширить возможности БД по обработке и управлению данными:

- добавить в форму **Закупки** возможность выполнения сортировки данных по определенным полям;
- добавить в форму **Закупки** возможность выполнения фильтрации данных по определенным полям;
- реализовать автоматическое изменение общего количества товара (поле [Кол товара] в таблице **Товары**), при занесении новых данных в форме **Продажи** с помощью создания процедуры аналогичной той, что использовалась для расчета количества товара после его покупки у фирмы.


2.1 Сортировка и фильтрация данных

- 1) Создайте возможность сортировки данных по номеру накладной.
Выполнение.

Откройте созданную ранее форму **Закупки** в режиме конструктор и убедитесь в наличии необходимых полей в области данных.

Рисунок 1 Образец области данных формы Закупки

- Создайте на месте надписи **№ накладной** элемент управления

Выключатель , при этом выполните следующее:

- свойство **Подпись** задайте **№** (на вкладке **Макет**);
- свойство **Имя** задайте **sort_nom** (на вкладке **Другие**);
- свойство **Нажатие кнопки** задайте **Процедура обработки событий** (на вкладке **События**).

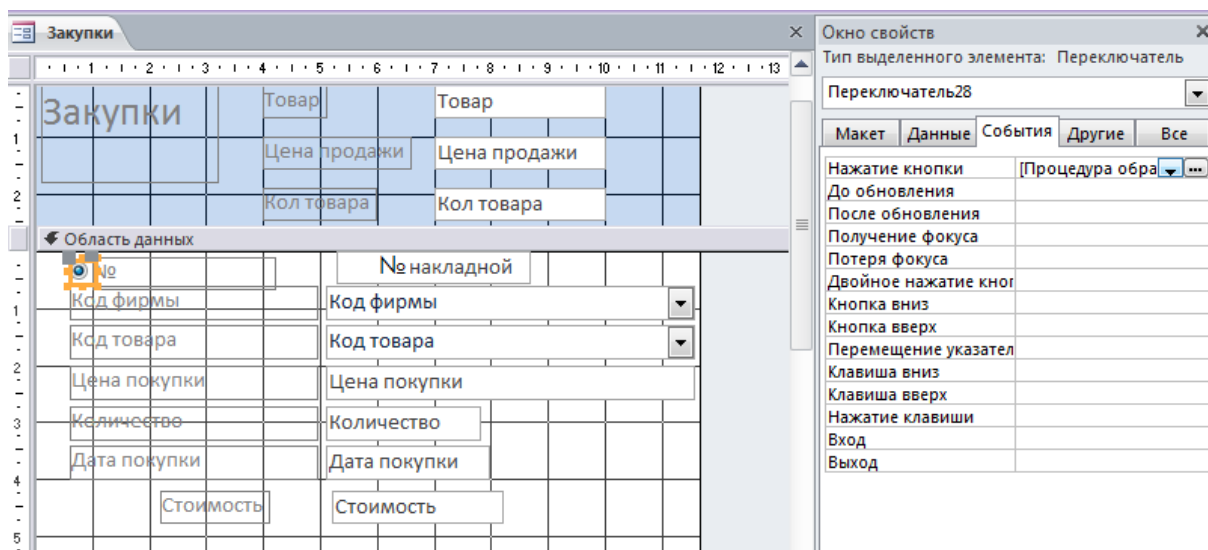


Рисунок 2 Образец создания Выключателя

- На рисунке 3 приведен текст процедуры сортировки по полю **№ накладной** (создается в специальном окне редактора, вызываемом кнопкой с многоточием [...] правее параметра Процедура обработки событий).

```
Private Sub sort_nom_Click()
If Me!sort_nom Then
    Me.OrderBy = "[№ накладной]"
    Me.OrderByOn = True
Else
    Me.OrderByOn = False
End If
End Sub
```

Рисунок 3 Текст процедуры сортировки по полю № накладной

2) Создайте возможность сортировки данных по **коду фирмы** и **дате продажи**.

Выполнение.

На месте других надписей в заголовке формы создайте еще два выключателя, дайте им соответствующие имена (sort_firm и sort_date) и подписи (Фирма и Дата).

Рисунок 4 Образец полученного окна формы

- Создайте как в п.1 процедуры обработки событий для новых кнопок (аналогичные рисунку 3).

Внимание! Дополнительно в процедуры (во все три) надо включить командные строки, которые будут сбрасывать соседние выключатели, для адекватного отображения на экране состояния сортировки, например, процедура на рисунке 5 включает сортировку по полю **Фирма** и при этом сбрасывает действие кнопок сортировки по номеру накладной и дате.

Внимание! Написав одну процедуру, остальные можно создать методом копирования и корректировки некоторых данных (см. ниже образец всех полученных процедур).

```
Private Sub sort_firm_Click()
If Me!sort_firm Then
    Me.OrderBy = "[Код фирмы]"
    Me.OrderByOn = True
Else
    Me.OrderByOn = False
End If
Me!sort_nom = False
Me!sort_date = False
End Sub
```

Рисунок 5 Процедура сортировки по полю Фирма и сброса двух других выключателей

Образец всех полученных процедур:

```
Private Sub sort_nom_Click()

    If Me!sort_nom Then
Me.OrderBy = "[№ накладной]"
        Me.OrderByOn = True
    Else
        Me.OrderByOn = False
    End If
    Me!sort_date = False
    Me!sort_firm = False

End Sub
```

```
Private Sub sort_firm_Click()

    If Me!sort_firm Then
Me.OrderBy = "[Код фирмы]"
        Me.OrderByOn = True
    Else
        Me.OrderByOn = False
    End If
```

```
Me!sort_nom = False
Me!sort_date = False
```

```
End Sub
```

```
Private Sub sort_date_Click()
```

```
    If Me!sort_date Then
        Me.OrderBy = "[Дата покупки]"
        Me.OrderByOn = True
    Else
        Me.OrderByOn = False
    End If
    Me!sort_nom = False
    Me!sort_firm = False
```

```
End Sub
```

Проверьте работу формы и всех ее переключателей.

3) Предусмотрите возможность фильтрации данных по дате и номеру накладной. Фильтрация данных осуществляется в форме с помощью свойства **Filter** объекта **Form**.

Выполнение.

- Создайте в **Примечаниях формы** две управляющие кнопки и два поля (свободных элемента), в которые пользователь сможет заносить образцы для фильтрации данных (рисунок 6):

1- поле с именем «filtr_nom» (вкладка **Другие**).

2- поле с именем: «filtr_date» (вкладка **Другие**).

Кнопка «Установить» с именем: «set_filtр» для применения фильтра.

Кнопка «Снять» с именем: «un_filtр» для сброса фильтра.



Рисунок 6 Возможное расположение полей и кнопок для установки и сброса фильтров

- Создайте процедуру обработки события для кнопки <Установить>, которая устанавливает фильтр в зависимости от внесенных в поля значений (рисунок 7). *Внимание.* Текст процедуры содержит комментарий (начинается с одиночного апострофа), который можно не вводить, т.к. он не влияет на выполнение и нужен только программисту.

```

Private Sub set_filtr_Click()
Dim strF As String 'для формирования строки фильтра
strF = ""
If Not IsNull(Me!filtr_nom) Then
'Если не пустое, то начать формирование строки фильтра
strF = "[№ накладной] = " & Me!filtr_nom & " AND "
'здесь значение поля filtr_nom нужно взять в кавычки
'потому, что оно текстовое. А внутри парных кавычек
'можно использовать одинарные (и наоборот)
End If
' проверка поля по дате
If Not IsNull(Me!filtr_date) Then
strF = strF & BuildCriteria("[дата покупки]", dbDate, Me!filtr_date) & " AND "
"
'для корректного создания условия фильтрации по полю с форматом
' календарной даты используется специальная функция
End If
'Проверим длину строки фильтра
If Len(strF) > 5 Then
'Обрезать концевое " AND "
strF = Left(strF, Len(strF) - 5)
Me.Filter = strF 'в свойство формы Фильтр заносится сформированная
строка
Me.FilterOn = True 'активизация фильтра формы
Me!un_filtr.Enabled = True 'открыть доступ к кнопке "Снять"
Me!un_filtr.SetFocus 'перенести фокус формы на кнопку "Снять"
Me!set_filtr.Enabled = False 'деактивация кнопки "Установить"
Else
MsgBox "Заполните сначала поля!", vbOKOnly, "Установка фильтра"
End If
End Sub

```

Рисунок 7 Текст процедуры установки фильтра

- Создайте процедуру обработки события для кнопки <Снять>, которая сбрасывает предыдущий фильтр и позволяет ввести новые значения для фильтрации (рисунок 8).

```

Private Sub un_filtr_Click()
Me.FilterOn = False 'сброс фильтра формы
Me!set_filtr.Enabled = True 'активизация кнопки "Установить"
Me!set_filtr.SetFocus 'перенос фокуса в форме на кнопку
"Установить"
Me!un_filtr.Enabled = False ' де активизация кнопки "Снять"
End Sub

```

Рисунок 8 Текст процедуры сброса фильтра

Разместите на форме поясняющие надписи (вида **Фильтр по № накладной; По дате**) над соответствующими полями ввода (рисунок 9). Проверьте работу фильтров.

Дата: 28.03.2015

Стоимость: 2 244,00р.

Номер фирмы: [dropdown]

№ накладной: [input]

Задать

Фильтр по № накладной: [input] По дате: [input]

установить Снять

Рисунок 9 Образец полученной формы (фрагмент)

2.2 Создание интерфейса для формы Продажи

4) Измените источник записей формы **Продажи**, так чтобы в него также вошли поля [Товар], [Цена продажи] и [Кол товара] из таблицы **Товары**.

Выполнение:

Для этого выведите на экран окно свойств формы **Продажи** и откорректируйте Источник записей (вкладка **Данные**, далее – **Источник записей**, далее - кнопка вызова конструктора для редактирования в виде многоточия [...]) следующим образом.

Добавьте в открывшийся при этом запрос-источник данных поля [Товар], [Цена продажи] и [Кол товара] (рисунок 10).

| Поле: | Продажи.* | Рисунок | Товар | Цена продажи | Кол товара |
|-----------------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|
| Имя таблицы: | Продажи | Товары | Товары | Товары | Товары |
| Сортировка: | | | | | |
| Вывод на экран: | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| Условие отбора: | | | | | |
| или: | | | | | |

Рисунок 10 Образец преобразованного источника записи

5) Разместите в области заголовка формы **Продажи** поля [Товар], [Цена продажи] и [Кол товара] (команда **Добавить поля** на вкладке **Конструктор**) и запретите к ним доступ (свойства **Блокировка** и **Доступ**).

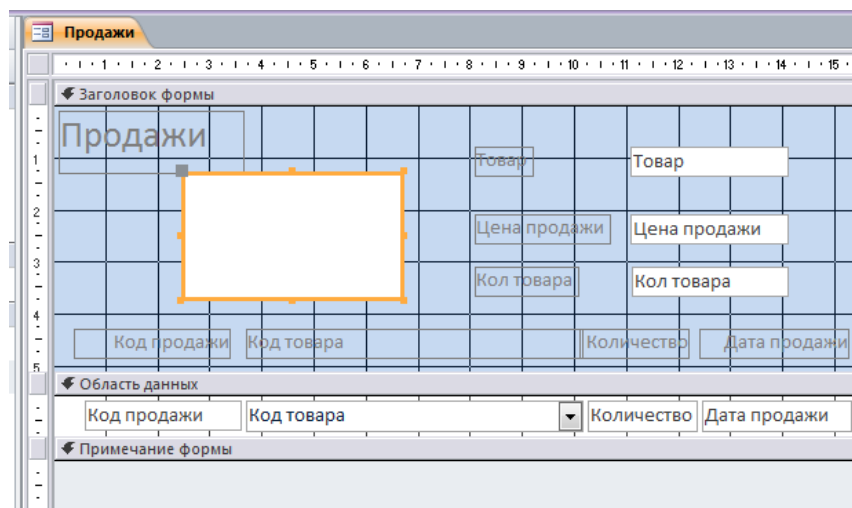


Рисунок 11 Образец окна формы Продажи в режиме конструктора

6) В форме **Продажи** для поля [Дата продажи] задайте в качестве значения по умолчанию текущую дату (функция =Date()).

7) Реализуйте автоматическое изменение общего количества товара при занесении новых данных в форму **Продажа**. Для этого создайте процедуру, аналогичную той, что использовалась для расчета количества товара после его покупки у фирмы. *Внимание.* Процедуру нужно откорректировать, т.к. в данном случае товар продается, следовательно, его количество уменьшается.

Выполнение:

В форме **Продажи** в окне свойств поля [Количество] на вкладке **События** для события **После обновления** установите **Процедура обработки событий**, нажмите кнопку вызова конструктора для редактирования модуля [...], впишите формулу, как показано ниже, и затем закройте окно MS Visual Basic.

```
Private Sub количество_AfterUpdate()
[Кол товара] = Nz([Кол товара], 0) - (Nz([Количество].Value, 0) -
Nz([Количество].OldValue, 0))
End Sub
```

Рисунок 12 Образец программного кода

8) Дополните существующий текст в сообщении об ошибке поля [Кол товара] таблицы **Товары** следующим текстом: «Измените количество продаваемого товара в расходной ведомости!».

9) Перейдите в режим формы **Продажи** и заполните несколько записей о продажах товара. Проанализируйте, как ведет себя форма при внесении данных о количестве товара, превышающем его наличие.

10) Переименуйте форму **Закупки** на «Приходная ведомость», а форму **Продажи** на «Расходная ведомость» как объекты БД (рисунок 13) и измените их подписи (свойства формы, вкладка **Макет, Подпись**).

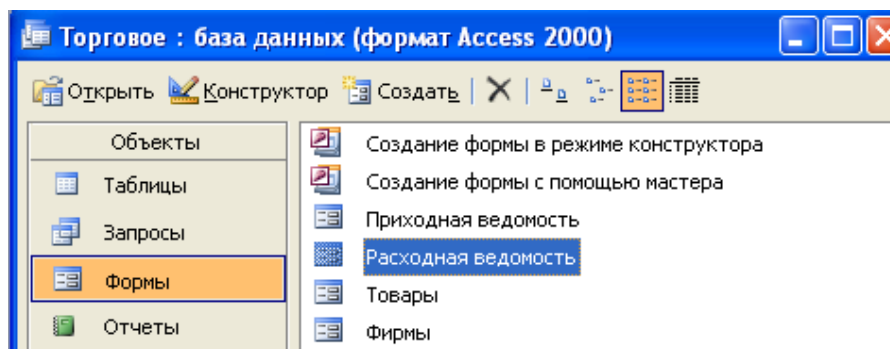


Рисунок 13 Объекты БД после переименования

11) Оформите отчет, который должен содержать: название работы; цель и задачи; краткий ход выполнения работы; ответы на контрольные вопросы.

3 Вопросы для самопроверки знаний

1. Каковы стандартные способы сортировки в БД?
2. Какими способами можно реализовать сортировку данных в форме?
3. Поясните понятие фильтров в БД.
4. Как организована фильтрация данных в форме Продажи после выполнения работы? Что содержит процедура фильтрации поля?
5. Чем отличается формула в процедуре подсчета количества товара после его продажи от процедуры подсчета товара после его покупки?
6. Что содержит сообщение об ошибке поля [Кол товара] и для чего оно предназначено?
7. Какими способами можно разместить новое поле в форме?
8. Как на форме Фирмы создать кнопку для открытия формы Товары?

Лабораторная работа №5

Разработка отчетов и кнопочных форм

Цель и задачи работы

Изучить принципы разработки отчетов и кнопочных форм в СУБД MS Access.

1 Общие сведения

Отчет - это гибкое и эффективное средство для организации данных при выводе на печать. С помощью отчета имеется возможность вывести необходимые сведения в том виде, в каком они требуются.

Главными объектами отчета являются поля отчета и пояснительный текст (рисунок 1). Основные сведения в отчете берутся из базовой таблицы, запроса или инструкции SQL, являющихся источниками данных для отчета. Другие сведения вводятся при разработке отчета и могут являться результатом вычислений над исходными данными. Для создания связи между отчетом и его источником данных применяются элементы управления (так же, как в формах). Такими элементами могут быть поля, надписи, линии или другие графические объекты.

Средства MS Access позволяют создавать отчеты, как в табличном виде, так и в свободной форме. При этом могут быть выполнены самые высокие требования к оформлению документов.

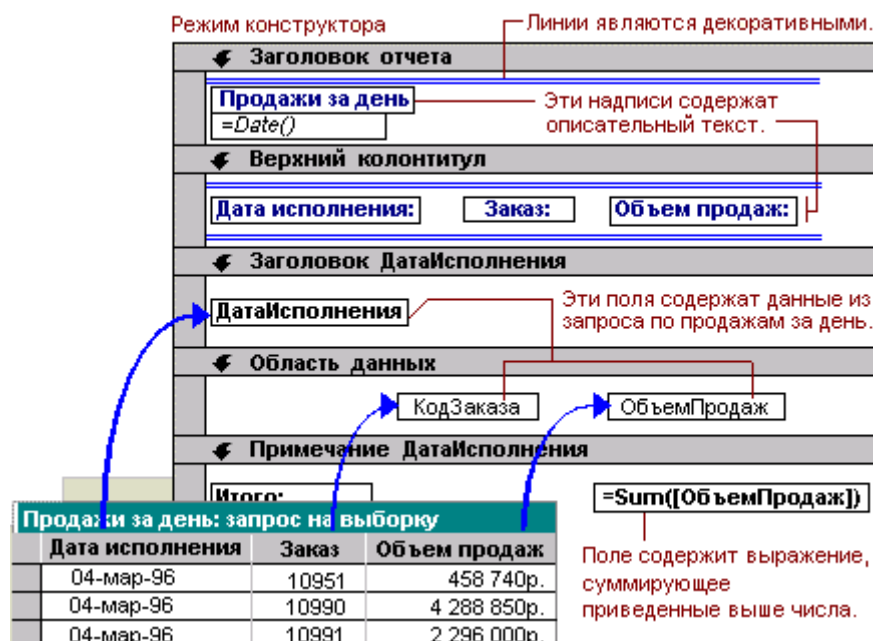


Рисунок 1 Пример структуры отчета и пояснения к его элементам

В MS Access для создания отчетов используются следующие средства:

- **конструктор отчетов** – позволяет самостоятельно разработать отчет с заданными свойствами;
- **мастер отчетов** – позволяет быстро создать отчет на основе выбранных полей при минимуме знаний о механизме его получения;

- **автоотчет: в столбец** и **автоотчет: ленточный** – автоматическое создание отчета с включением всех полей таблицы (запроса);
- **диаграмма** – автоматическое создание отчета, содержащего отображение данных в графическом виде;
- **почтовые наклейки** - автоматическое создание отчета, в виде почтовых наклеек.

Создавая отчет, Вы можете собрать все записи с одинаковым значением в каком-нибудь поле в одну группу и напечатать их как самостоятельный блок отчета. Например, в нашей таблице **Покупки** есть поле [№ накладной]. Группируя записи по этому полю, мы получим отчет, в котором все записи, с одинаковым № накладной, будут собраны в один блок текста отчета. Такой блок называется **группой** и имеет **заголовок**, **область данных** и **примечания** группы. В заголовке, как правило, выводится значение того поля, по которому проведена группировка. В область данных включаются те поля, по которым не проводилась группировка. А в примечания вставляются различного рода итоги: количество записей, сумма по столбцам и т.д.



Рисунок 2 Структура заголовков и примечаний групп в отчетах

С помощью надстройки **Диспетчер кнопочных форм** автоматически создается удобная и функциональная форма, для выполнения основных действий с БД. На кнопочную панель помещаются кнопки, при нажатии которых открываются формы или отчеты (или открываются другие кнопочные формы, с помощью которых открываются другие формы или отчеты), осуществляется выход из MS Access или изменяется сама кнопочная форма. Диспетчер кнопочных форм позволяет создавать кнопочные формы, подобные тем, что создает мастер баз данных.

MS Access имеет возможность управления запуском базы данных. Пользователь может указать, какую форму следует открыть при запуске, разрешается ли изменять панели инструментов, а также являются ли доступными контекстные меню.

2 Порядок выполнения работы

Для улучшения работы с БД необходимо создать отчеты - ведомость закупки, ведомость продажи, сводный отчет об итогах работы за текущий месяц.

2.1 Создание отчета с использованием мастера

1) Создайте отчет с помощью **мастера** для запроса **Приходная ведомость**.

Выполнение.

- Перейдите в окне БД на вкладку **Отчеты** и нажмите кнопку <Мастер отчетов>. В открывшемся окне диалога укажите способ создания отчета и источник – запрос **Закупки**.

- На первом этапе выберите поля, которые должны появиться в отчете. Из левого списка **Доступные поля** в правый список **Выбранные поля** перенесите с помощью кнопки < > следующие поля: **№ накладной**, **Код фирмы**, **Код товара**, **Цена покупки**, **Количество**, **Дата покупки**, **Стоимость**. Выбранные поля должны быть расположены в указанной последовательности.

- На следующем шаге задается **группировка данных**. Укажите, например, группировку по полю **№ накладной**.

- Далее укажите **порядок сортировки** в области данных отчета. MS Access автоматически сортирует данные по группирующим полям, в данном случае по полю **№ накладной**. Если записи в рамках группы должны быть дополнительно отсортированы и по другим полям, то эти поля необходимо указать в окнах уровней сортировки (4 поля) (рисунок 3).

- На этом же шаге, используя кнопку <Итоги> перейдите к определению типов итоговых значений по выбранным полям, которые появятся в отчете в области примечания группы (рисунок 4).

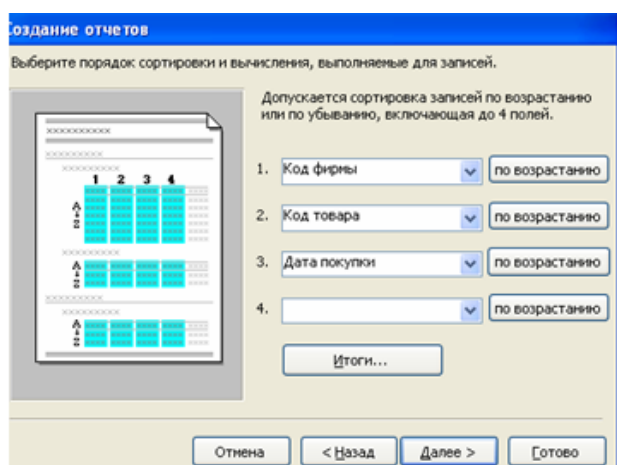


Рисунок 3 Поля для сортировки

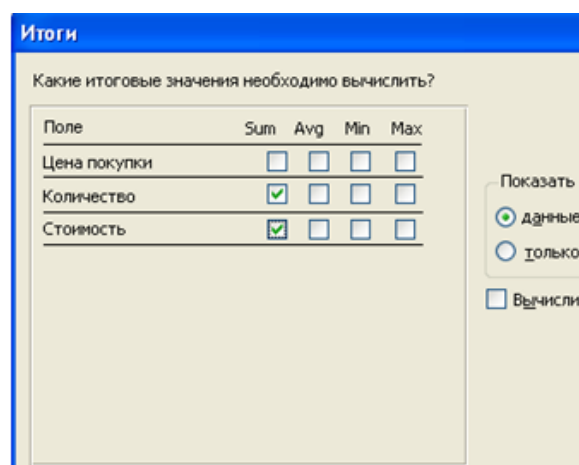


Рисунок 4 Установка типов итогов

- На последних шагах мастера выберите макет Структура 2 и любой из понравившихся стилей отчета. Назовите отчет **Ведомость закупки**.

2) Теперь, перейдите в режим **конструктора отчета** и окончательно «наведите там порядок», изменив шрифты для итоговых значений, выравнивание и расположение заголовков полей и т.д. На рисунке 5 показан пример созданного отчета.

| <i>Ведомость закупки</i> | | | | | |
|---------------------------|-------------------|---------------------|---------------------|-------------------|------------------|
| № накладной <i>a100</i> | | | | | |
| <i>Код фирмы</i> | <i>Код товара</i> | <i>Дата покупки</i> | <i>Цена покупки</i> | <i>Количество</i> | <i>Стоимость</i> |
| 3 | 2 | 09.02.2008 | 59,00р. | 10 | 590,00р. |
| 3 | 3 | 01.03.2008 | 21,00р. | 90 | 1 890,00р. |
| 3 | 4 | 01.03.2008 | 20,00р. | 38 | 760,00р. |
| 3 | 6 | 01.03.2008 | 75,00р. | 20 | 1 500,00р. |
| Суммарные значения | | | | 158 | 4740 |
| № накладной <i>a103</i> | | | | | |
| <i>Код фирмы</i> | <i>Код товара</i> | <i>Дата покупки</i> | <i>Цена покупки</i> | <i>Количество</i> | <i>Стоимость</i> |
| 3 | 2 | 15.02.2008 | 59,00р. | 10 | 590,00р. |
| 3 | 3 | 15.02.2008 | 45,00р. | 20 | 900,00р. |
| 4 | 6 | 12.02.2008 | 30,00р. | 100 | 3 000,00р. |
| Суммарные значения | | | | 130 | 4490 |

Рисунок 5 Возможный вид отчета Ведомость закупки

3) Самостоятельно создайте отчеты:

- ведомость продажи с группировкой по полю [код товара];
- сводный отчет об итогах работы за текущий месяц – с группировкой по нескольким полям.

2.2 Создание кнопочной формы

4) Создайте кнопочную форму для запуска объектов БД (рисунок 6).

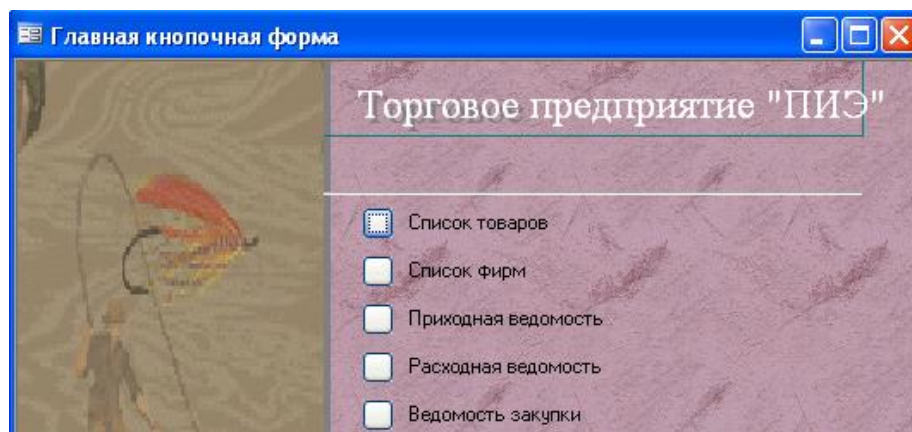


Рисунок 6 Пример созданной кнопочной формы

Выполнение.

- В ранних версиях Access - запустите **Диспетчер кнопочных форм** с помощью команды **Сервис – Служебные программы - Диспетчер кнопочных форм**.

Примечание: Диспетчер кнопочных форм в Access 10 реализуется через панель быстрого доступа.

На панели быстрого доступа должна иметься вкладка **Работа с базами данных**.

Если на этой вкладке нужного раздела нет, то его надо включить:
меню: **ФАЙЛ**

выбрать: **ПАРАМЕТРЫ**

выбрать: **Панель быстрого доступа**.

Затем на вкладке **Настройка панели быстрого доступа** надо выбрать из раскрывающегося списка строку: **Вкладка "Работа с базами данных"** и в списке команд выделить: **Диспетчер кнопочных форм**, далее - кликнуть по кнопке **ДОБАВИТЬ>>**

В левом верхнем углу программного окна, рядом с кнопками **Сохранить** и **Отменить**, появится кнопка **Диспетчер кнопочных форм**.

После запуска диспетчера:

- Нажмите кнопку **<Изменить>** (рисунок 7), при выделенной записи **Главная кнопочная форма** (эта запись создается автоматически).

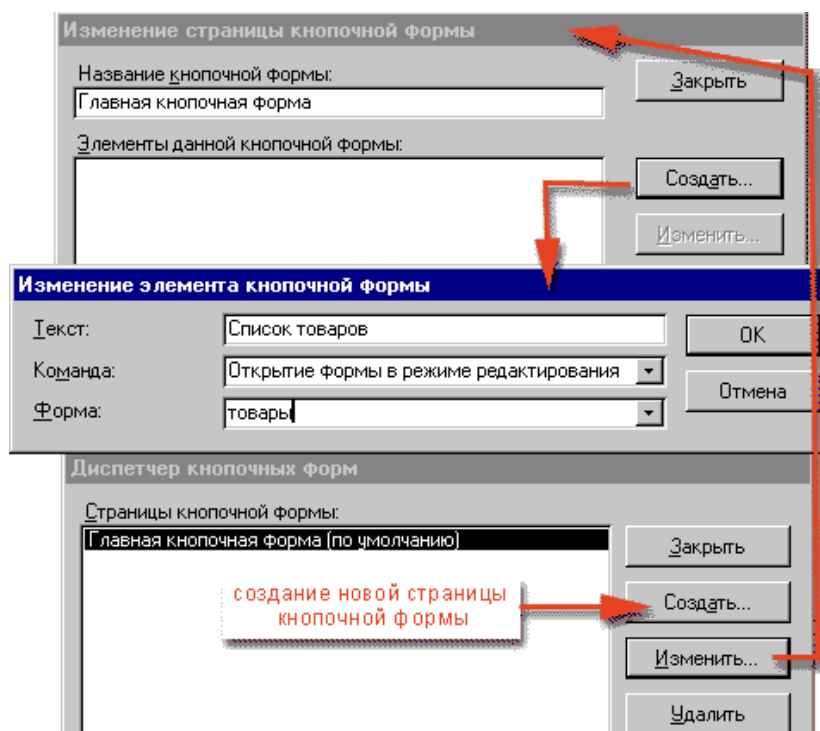


Рисунок 7

- Создайте **элементы** этой формы (с помощью кнопки <Создать>) - кнопки на форме, возле которых выводится текстовая надпись из поля **Текст** и которые выполняют действия, определенные в поле **Команда**.

Для кнопок **Список товаров** и **Список фирм** выберите команду Открытие формы для изменения (т.е. в режиме редактирования), а для кнопок Приходная ведомость и Расходная ведомость – Открыть форму для добавления.

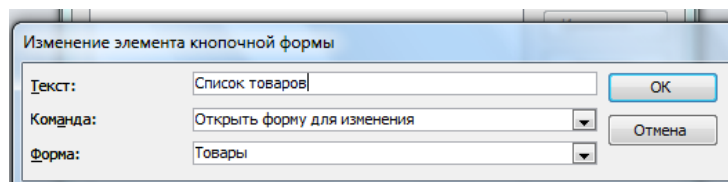


Рисунок 8 Окно создания элемента кнопочной формы

- По окончании заполнения, нажмите кнопку <Заккрыть> и в окне БД на вкладке **Формы** появится новая форма - **Кнопочная форма**. Рисунок и другие оформительские элементы измените через **конструктор форм**.

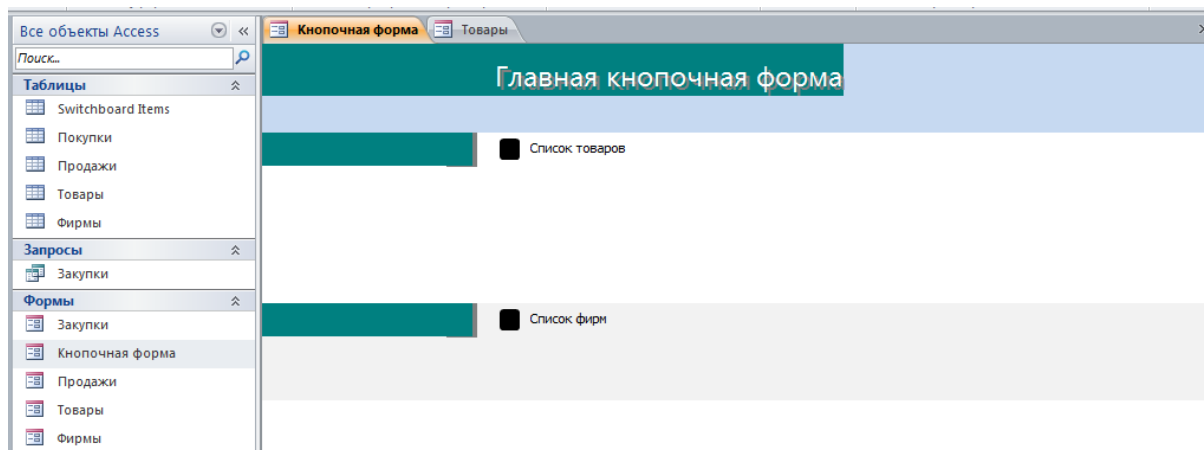


Рисунок 9 Образец кнопочной формы (фрагмент)

2.3 Настройка параметров запуска

5) Настройте свою БД таким образом, чтобы при ее запуске сразу же стартовала, созданная Вами **кнопочная форма**.

Выполнение.

- В ранних версиях Access выберите в меню **Сервис** команду **Параметры запуска** и введите требуемые значения (рисунок 10). Для того чтобы получить справку о конкретном элементе диалогового окна, нажмите кнопку <?> в правом верхнем углу диалогового окна, а затем выберите элемент.

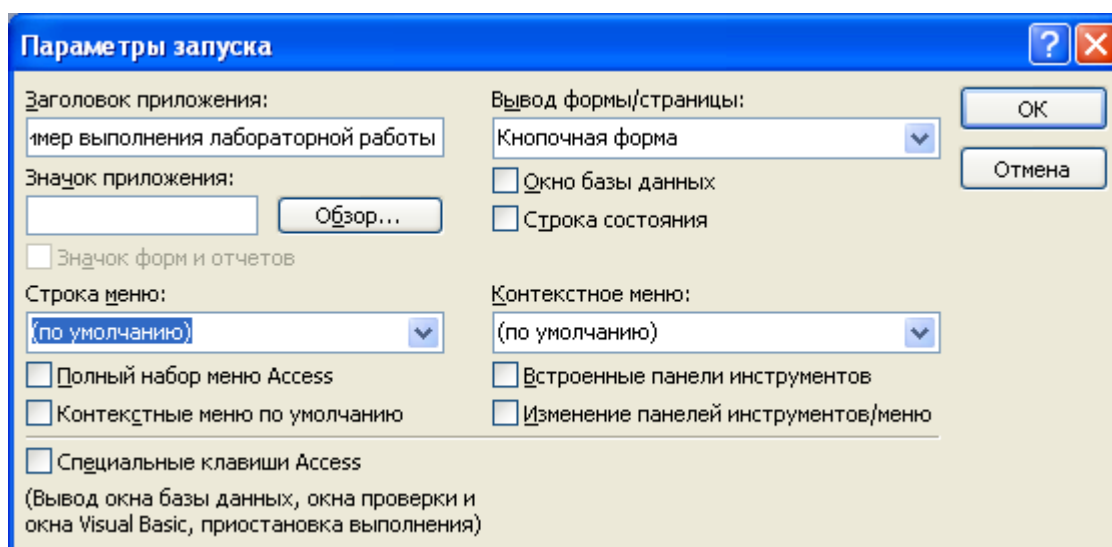


Рисунок 10 Пример настройки параметров запуска базы данных

В версии Access 10 параметры запуска устанавливаются в меню **Файл - Параметры - Параметры для текущей базы данных**.

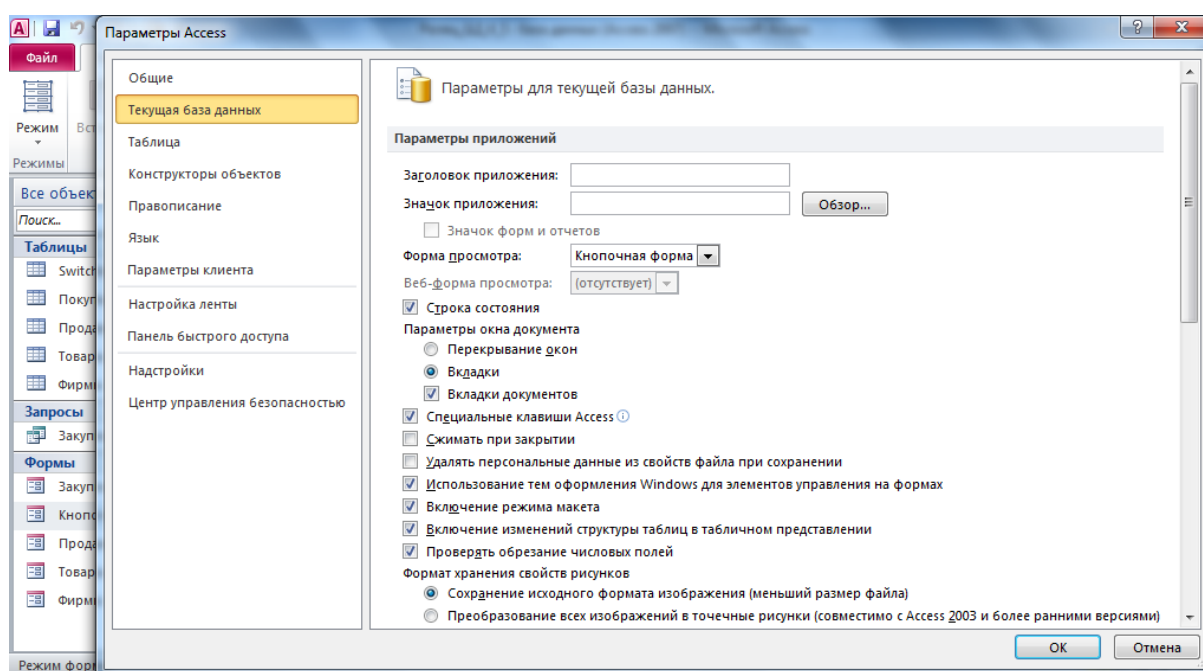


Рисунок 11 Параметры БД

Внимание! Настройки, задаваемые в диалоговом окне **Параметры запуска**, могут быть использованы вместо или в дополнение к макросу AutoExec. Макрос AutoExec (если он существует в данной БД) запускается автоматически при загрузке базы (однако после выполнения настроек сделанных в окне **Параметры запуска**). Для того чтобы обойти настройки параметров запуска, следует открывать базу данных, удерживая нажатой клавишу SHIFT.

6) Закройте Access и оформите отчет, который должен содержать: название работы; цель и задачи; краткий ход выполнения работы; ответы на контрольные вопросы.

3 Вопросы для самопроверки знаний

1. В чем заключается отличие отчетов от форм?
2. Как создать отчет?
3. Поясните процесс группирования данных по трем уровням.
4. Как внедрить рисунок в отчет?
5. Для чего предназначены заголовок и примечание отчета, заголовок и примечание группы?
6. Для кого предназначена кнопочная форма? В чем состоит удобство и безопасность применения кнопочной формы для работы с БД
7. Как внедрить рисунок на кнопочную форму?
8. Какие команды существуют для различных элементов кнопочной формы?
9. Как обойти настройки параметров запуска?

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

- 1) Советов, Б. Я. Базы данных: теория и практика [Текст]: учебник для бакалавров, для студентов вузов, обуч. по направлениям "Информатика и вычислительная техника" и "Информационные системы" / Б. Я. Советов, В. В. Цехановский, В. Д. Чертовский. - 2-е изд. - Москва : Юрайт, 2014. - 463 с.
- 2) Балдин, К. В. Информационные системы в экономике [Текст]: учеб. пособие для студентов вузов, обучающихся по направлению 080100 "Экономика" / К. В. Балдин. - М.: ИНФРА-М, 2013. - 216 с.
- 3) Белов, В. В. Проектирование информационных систем [Текст]: учебник для студентов, обучающихся по направлению "Прикладная информатика" и другим экономическим специальностям / В. В. Белов, В. И. Чистякова ; под ред. В. В. Белова. - 2-е изд., стер. - Москва : Академия, 2015. - 352 с.