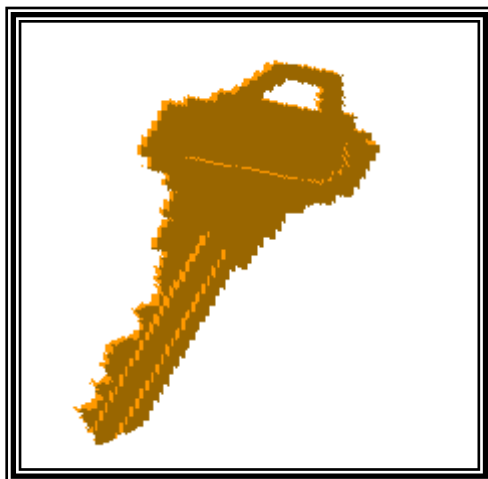


**ДОНЕЦКИЙ НАЦИОНАЛЬНЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

**Кафедра прикладной математики и информатики**

**МЕТОДИЧЕСКИЕ УКАЗАНИЯ И ЗАДАНИЯ  
К ВЫПОЛНЕНИЮ КУРСОВОЙ РАБОТЫ ПО КУРСУ  
« БАЗЫ ДАННЫХ»**

**( направление подготовки 6.050103 "Программная инженерия")**



**Донецк 2011**

**ДОНЕЦКИЙ НАЦИОНАЛЬНЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

**Кафедра прикладной математики и информатики**

**МЕТОДИЧЕСКИЕ УКАЗАНИЯ И ЗАДАНИЯ  
К ВЫПОЛНЕНИЮ КУРСОВОЙ РАБОТЫ ПО КУРСУ**

**« БАЗЫ ДАННЫХ »**

**( направление подготовки 6.050103 ”Программная инженерия”)**

Рассмотрено на заседании кафедры  
Прикладной математики и информатики

Протокол №3 от 08.11.10

Утверждено на заседании

Учебно- издательского совета ДонНТУ

Протокол №1 от 13.01.11

**Донецк 2011**

УДК 681.3.07

Методические указания и задания к выполнению курсовой работы по курсу «Базы данных» (направление подготовки 6.050103 «Программная инженерия»). Сост. Г.Г.Шалдырван, Н.С.Костюкова. Донецк, 2011.- 64 с.

Приведены методические указания по разработке приложений в среде системы управления базами данных Microsoft Access 2003. Излагаются вопросы, связанные с автоматизацией приложения с помощью макросов, разработкой отчетов, кнопочных форм, меню. Рассматриваются требования по содержанию и оформлению пояснительной записки к курсовой работе.

Приведены образцы титульного листа, реферата, содержания пояснительной записки, а также форма задания на курсовую работу.

Составители:

доц. Г.Г.Шалдырван

доц. Н.С.Костюкова

## ВВЕДЕНИЕ

Начиная с 50-х годов прошлого столетия, компьютерная индустрия прошла этапы рождения, юности и зрелости. В настоящее время компьютер является главным инструментом для автоматизации сложных процессов в производстве, бизнесе и управлении.

Компьютерные технологии предполагают широкое использование компьютерных информационных систем, основывающихся на концепции баз данных (БД). Согласно этой концепции основой информационной технологии являются данные, которые организованы в БД с целью адекватного отображения изменяющегося реального мира и удовлетворения информационных потребностей пользователей.

Увеличение объема и структурной сложности хранимых данных, постоянное расширение круга пользователей информационных систем выдвинуло требование создания удобных общесистемных средств интеграции хранимых данных и управления ими. Это привело в конце 60-х годов к появлению первых промышленных систем управления базами данных (СУБД) – специализированных программных средств, предназначенных для организации и ведения БД.

Развитие компьютерной техники в последние годы привело к резкому усложнению и дифференциации СУБД. Для них стало характерным использование идей объектно-ориентированного подхода, управления распределенными базами данных, активного сервера БД, языков программирования четвертого поколения, фрагментации и параллельной обработки запросов, технологии тиражирования данных, многопоточковой архитектуры и других революционных достижений в области обработки данных.

Современные СУБД – это сложные многофункциональные программные системы, функционирующие в открытой распределенной среде. Сегодня они уже доступны в деловой сфере, предоставляя разработчикам мощные средства управления данными и богатый инструментарий для создания прикладных программ и систем.

Microsoft Access является в настоящее время лучшей настольной СУБД, отвечающей современным требованиям к программному обеспечению такого рода: с одной стороны Access предоставляет удобный интерфейс, с другой – мощные средства по представлению и обработке данных с применением самых последних достижений в области информационных технологий.

## 1. ОСНОВНЫЕ ЭТАПЫ РАЗРАБОТКИ ПРИЛОЖЕНИЯ

Для решения задачи обработки данных с использованием реляционной СУБД необходимо выполнить два важнейших шага: спроектировать эффективную структуру таблиц и их взаимосвязей (построить модель данных) и обеспечить удобный пользовательский интерфейс (разработать приложение).

Проектирование базы данных начинается с составления перечня всех основных задач, которые должны решаться в рамках рассматриваемой предметной области. Для каждой задачи следует определить данные, необходимые для ее решения. Данные определяются на основании ознакомления со всеми входными и выходными документами, формами, отчетами, различными справками, циркулирующими в предметной области. После анализа данных нужно выполнить привязку их к объектам предметной области, представить объекты взаимосвязанными нормализованными таблицами.

Проектирование приложения следует начать с объединения задач в тематические группы и представления последовательности их выполнения. Нужно определить общие требования к приложению, продумать систему ввода и контроля данных, сформулировать все необходимые запросы к данным, разработать структуру форм и отчетов. Рекомендуются также создать макет приложения и пользовательского интерфейса.

Чем тщательнее выполнено проектирование, тем легче будет последующая реализация прикладной системы.

По мере разработки разделов приложения их нужно тестировать, а также проверять их работу на реальных данных. При этом может возникнуть необходимость усовершенствования приложения. Выявление необходимых изменений на ранних стадиях разработки приложения позволит существенно сократить время на последующие переделки.

В данных методических указаниях приведены некоторые объекты (отчеты, формы, макросы) приложения, разработанного для учебной базы данных, отображающей деятельность посреднической фирмы, поставляющей вычислительную технику:

**для автоматизации работы посреднической фирмы необходимы данные о фирмах, поставляющих вычислительную технику (название фирмы, город, страна, форма собственности, телефон, адрес), и о представленных ими партиях товаров (наименование товара, фирма-изготовитель, дата поставки партии, цена единицы товара, количество единиц).**

Для описания предметной области сформированы следующие таблицы:

**Поставщики** – сведения о фирмах, поставляющих товары;

**Каталог** – наименование товара и название фирмы, производящей товар;

**Товары** – сведения о поставках товаров;

**Формы собственности, Страны, Города** – таблицы-справочники.

Структура указанных таблиц и связи между ними представлены в следующей схеме данных:



## 2. РАЗРАБОТКА ОТЧЕТОВ

### Общие сведения об отчетах

Отчет — это гибкое и эффективное средство для организации данных при выводе на печать. Отчеты, как и формы, являются средством наглядного представления данных, содержащихся в базе: в них можно сгруппировать записи, вычислить итоговые значения, представить данные в виде диаграмм. Однако, в отличие от форм, основная функция отчетов – подготовка данных в пригодном для печати виде.

Отчеты обладают значительной частью тех же свойств, что и формы. Элементы управления, которые используются в отчетах, также аналогичны тем, которые применяются в формах.

### Создание отчета

Access позволяет создавать отчет самостоятельно или с помощью мастера, который быстро выполняет всю рутинную работу. После вызова мастера выводятся диалоговые окна с приглашением ввести необходимые данные: выбрать базовую таблицу (запрос), поля таблицы, указать вид отчета. Отчет создается на основании полученных ответов. Мастер окажется полезным даже для опытных программистов, так как позволяет быстро разработать макет, служащий основой создаваемого отчета. После этого можно переключиться в режим конструктора и внести изменения в стандартный макет.

Указанным способом можно создавать **диаграммы** (с помощью **Мастера диаграмм**), **почтовые наклейки**, **ленточный автоотчет** (по

аналогии с ленточной формой) или **автоотчет в столбец** (аналог простой формы), а также другие виды отчетов. При желании полученный отчет можно изменить в режиме конструктора.

Существует также возможность сохранения формы в виде отчета. Для этого необходимо в окне базы данных выбрать вкладку **Формы**, щелкнуть правой кнопкой мыши на сохраняемой форме, в появившемся контекстном меню выбрать команду **Сохранить как отчет**. После этого следует ввести имя для отчета в поле **Имя отчета**. Затем следует выбрать вкладку **Отчеты** и дважды щелкнуть мышью на отчете для его предварительного просмотра. Построенный таким образом отчет будет иметь те же элементы управления и свойства, что и форма.

Отчеты, как и формы, можно создавать вручную. При таком способе необходимо выполнить определенную последовательность действий:

1. Создание пустого отчета и выбор источника записей.
2. Сортировка и группировка данных в отчете.
3. Добавление заголовка и примечаний.
4. Отображение данных, текста и итогов в текстовых полях, надписях и других элементах управления.
5. Форматирование, перемещение, изменение размеров и выравнивания надписей, текстовых полей и других элементов управления.
6. Просмотр и распечатка отчета.

Отчеты могут быть открыты в трех режимах: в режиме *конструктора*, в режиме *предварительного просмотра* и в режиме *просмотра образца*. Режим конструктора используют для создания отчета или изменения структуры существующего отчета. Режим предварительного просмотра позволяет просматривать данные в том виде, в котором они будут размещаться на печатных страницах. В режиме просмотра образца выводятся основные элементы макета отчета с образцами данных, демонстрирующих представление данных в отчете.

### **Структура отчета**

Вся информация в отчете разбивается на разделы, каждый из которых имеет специальное назначение: заголовок отчета, верхний колонтитул, область данных, нижний колонтитул, примечание отчета. Например, отчет, содержащий сведения о поставщиках, может выглядеть так:

В режиме конструктора на экране отображается макет каждого раздела отчета в одном экземпляре. Содержимое каждого раздела представлено элементами управления, например, полями, надписями. При печати некоторые разделы могут неоднократно повторяться. Так, верхний и нижний колонтитулы при печати будут присутствовать на каждой странице, а заголовок и примечание отчета печатаются только один раз в начале и в конце отчета соответственно. Элементы управления, расположенные в области данных, составляют содержимое отчета.

Добавление и удаление областей заголовка, примечаний и колонтитулов отчета выполняется командой

**Вид → Заголовок/примечание отчета**  
или

**Вид → Колонтитулы**

Следует отметить, что области заголовка и примечаний, а также верхнего и нижнего колонтитулов всегда добавляются парами. Если не требуется выводить на печать одну из областей такой пары, следует задать для лишней области значение **Нет** для свойства **Вывод на экран** или же удалить все элементы управления из этой области и установить для нее нулевую высоту. Можно также задать значение 0 для свойства **Высота**. При удалении областей заголовка, примечаний или колонтитулов удаляются все элементы управления, содержащиеся в этих областях.

Для отказа от печати верхнего или нижнего колонтитула на одной странице с заголовком или примечаниями отчета необходимо установить указатель на область выделения отчета, открыть окно свойств двойным



нажатием кнопки мыши и выбрать одно из следующих значений свойства отчета **Верхний колонтитул** или **Нижний колонтитул**:

- **все страницы** (колонтитул печатается на всех страницах - значение по умолчанию);
- **без заголовка** (колонтитул не печатается на одной странице с заголовком отчета);
- **без примечания** (колонтитул не печатается на одной странице с примечанием отчета);
- **без заголовка/примечания** (колонтитул не печатается на одной странице ни с заголовком, ни с примечанием отчета).

### **Форматирование отчетов**

**Применение к отчету встроенных форматов:** чтобы использовать встроенные форматы для оформления отчета, необходимо открыть отчет в режиме конструктора. После этого следует выделить форматируемый раздел отчета (или весь отчет), нажать кнопку **Автоформат** на панели инструментов или выполнить команду

#### **Формат → Автоформат**

Затем следует выбрать имя нужного формата из списка. Для указания отдельных атрибутов форматирования (шрифт, цвет или тип границы) используется кнопка **Параметры**.

**Нумерация страниц формы или отчета:** для нумерации страниц отчета следует открыть отчет в режиме конструктора отчета и выполнить команду

#### **Вставка → Номера страниц.**

В открывшемся окне диалога **Номера страниц** необходимо выбрать параметры, определяющие формат, расположение и выравнивание номеров страниц:

- по левому краю (поле номера страницы помещается на границу левого поля страницы);
- по центру (поле номера страницы размещается по центру между левым и правым полями страницы);
- по правому краю (поле номера страницы добавляется на границу правого поля страницы);
- внутри (поле номера страницы помещается на левый и правый края страницы; нечетные номера страниц печатаются слева, а четные - справа);
- снаружи (четные номера страниц печатаются слева, а нечетные справа).

Для печати номера страницы на первой странице следует установить флажок **Отображать номер на первой странице**.

### **Сортировка и группировка записей в отчете**

В отчете допускается сортировка и группировка по 10 и менее полям или выражениям. Для задания порядка сортировки записей необходимо открыть отчет в режиме конструктора и нажать кнопку **Сортировка и группировка** на панели инструментов или выполнить команду

**Вид → Сортировка и группировка**

В открывшемся окне **Сортировка и группировка** в первой строке бланка следует выбрать в ячейке столбца **Поле/выражение** имя поля или ввести выражение, которое будет определять первый (основной) уровень сортировки. Вторая строка бланка определяет второй уровень сортировки и так далее.

При заполнении ячейки столбца **Поле/выражение** в ячейку столбца **Порядок сортировки** автоматически вводится значение **По возрастанию** (при необходимости это значение можно изменить).

**Группировка записей в отчете:** используется то же окно, что и для сортировки. В нем для полей и выражений группировки необходимо задать значения следующих свойств группы:

- **заголовок группы** - добавляет или удаляет заголовок группы, определяемой полем или выражением;
- **примечание группы** - добавляет или удаляет область примечаний группы, определяемой полем или выражением;
- **группировка** - определяет способ группировки записей. Список доступных значений этого свойства зависит от типа данных поля, по которому проводится группировка. При группировке по выражению выводится полный список значений данного свойства;
- **интервал** - определяет любой допустимый интервал значений поля или выражения, по которому проводится группировка;
- **не разрывать** - задает или отменяет обязательную печать группы на одной странице.

После заполнения ячеек в окне сортировки и группировки в макет отчета добавляются соответствующие разделы. Далее нужно в эти разделы переместить элементы управления, отображающие поля группировки.

Для **изменение порядка** сортировки и группировки в отчете необходимо в окне **Сортировка и группировка** выбрать область выделения

строки для поля или выражения, положение которого в бланке требуется изменить, нажать кнопку мыши и, не отпуская ее, переместить строку на новую позицию в бланке.

Если перемещаемые группы имеют заголовки или примечания, то на новую позицию будут перемещены заголовки, области примечаний и все созданные в них элементы управления. Однако при этом может оказаться необходимым уточнение положения элементов управления в областях заголовка и примечаний.

Для **отмены** сортировки или группировки в отчете по конкретному полю или выражению следует выбрать в окне **Сортировка и группировка** область выделения строки для поля или выражения, которое требуется исключить из бланка сортировки и группировки, и нажать клавишу DEL.

### Пример 1.

Создадим отчет с группировкой записей. В качестве источника данных для отчета будем использовать запрос **Поставщики\_товары**, представляющий собой симметричное объединение таблиц **Поставщики** и **Товары**.

Первоначально создадим вручную отчет, который в режиме конструктора имеет следующий вид:

В режиме предварительного просмотра отчет будет выглядеть так:

Поставщик	Город	Товар	Цена единицы	Количество единиц	Дата поставки
SunGroup	Гомель	Flash-карта	1,00	1	06.06.2002
SunGroup	Гомель	Видеокарта	422,00	1	01.01.2001
SunGroup	Гомель	Видеокарта	25,00	1	05.04.2002
SunGroup	Гомель	Flash-карта	1,00	1	06.06.2002
SunGroup	Гомель	Flash-карта	1,00	1	06.06.2002
SunGroup	Гомель	Flash-карта	1,00	1	06.06.2002
SunGroup	Гомель	Flash-карта	1,00	1	06.06.2002
SunGroup	Гомель	Flash-карта	1,00	1	06.06.2002
SunGroup	Гомель	Flash-карта	1,00	1	06.06.2002
SunGroup	Гомель	CD drive	315,00	3	08.12.2002
Альфа	Петербург	Клавиатура	55,00	10	24.12.2002
Альфа	Петербург	Flash-карта	0,00	10	06.02.2002
Альфа	Петербург	Flash-карта	25,50	10	02.07.2001
Альфа	Петербург	Клавиатура	70,00	2	05.01.2002
Альфа	Петербург	Видеокарта	230,00	1	07.01.2001
Альфа	Петербург	Клавиатура	34,00	1	11.04.2002
Альфа	Петербург	Видеокарта	330,00	10	24.12.2002
Альфа	Петербург	Клавиатура	100,00	25	24.12.2002
Артемон	Донецк	Flash-карта	12,00	5	12.09.2002
Артемон	Донецк	Видеокарта	10,00	6	18.05.2002
.	.	.	.	.	.

Как видим, в данном случае имеется большая избыточность: многократно повторяются названия поставщика и города. Для устранения избыточности введем группировку по полю **Поставщик**:

После этого в макете отчета появится дополнительный раздел – **Заголовок группы «Поставщик»**:



Поставщик	Город	Товар	Цена единицы	Количество единиц	Дата поставки
SunGroup	Гомель	Flash-карта	1,00	1	06.06.2002
		Видео карта	422,00	1	01.01.2001
		Видео карта	25,00	1	05.04.2002
		Flash-карта	1,00	1	06.06.2002
		Flash-карта	1,00	1	06.06.2002
		Flash-карта	1,00	1	06.06.2002
		Flash-карта	1,00	1	06.06.2002
		Flash-карта	1,00	1	06.06.2002
		Flash-карта	1,00	1	06.06.2002
		CD drive	315,00	3	08.12.2002
Аль фа	Петербург	Клавиатура	55,00	10	24.12.2002
		Flash-карта	0,00	10	06.02.2002
		Flash-карта	25,50	10	02.07.2001
		Клавиатура	70,00	2	05.01.2002
		Видео карта	230,00	1	07.01.2001
		Клавиатура	34,00	1	11.04.2002
		Видео карта	330,00	10	24.12.2002
		Клавиатура	100,00	25	24.12.2002
Артемон	Донецк	Flash-карта	12,00	5	12.09.2002
		Видео карта	10,00	6	18.05.2002
Интер-Вест	Донецк	CD drive	12,00	1	12.12.2002
		Клавиатура	167,00	5	11.05.2002
		Видео карта	27,00	10	16.12.2002
		Клавиатура	167,00	4	07.05.2002
Омега	Донецк	Flash-карта	14,00	4	29.09.2002
		Клавиатура	34,00	7	12.10.2002
		CD drive	26,00	5	10.10.2002

### Создание вычисляемых элементов управления

При разработке отчетов существует возможность создавать вычисляемые ЭУ. Для этого в свойство **Данные** элемента управления, который будет использоваться как вычисляемый (как правило, это ЭУ **Поле**), следует ввести выражение для вычисления нужного значения. Для облегчения ввода можно воспользоваться **Построителем выражений** или открыть окно **Область ввода** с помощью клавиш SHIFT+F2.

### Замечания:

- 1) В вычисляемом элементе управления перед каждым выражением следует помещать знак равенства (=).
- 2) Если источником данных отчета является запрос, выражение можно поместить в базовый запрос отчета. Это повышает быстродействие отчета и, если предполагается вычислять итоговые значения по группам записей, позволяет использовать имя вычисляемого поля в статистической функции.

Для вычисления в отчете итоговых значений для групп записей или для всех записей **следует добавить вычисляемое поле в одну из следующих областей:**

- 1) для расчета итогового значения по группе записей - в заголовок группы или в примечание группы;
- 2) для расчета итогового значения по всем записям отчета - в заголовок отчета или в примечание отчета.

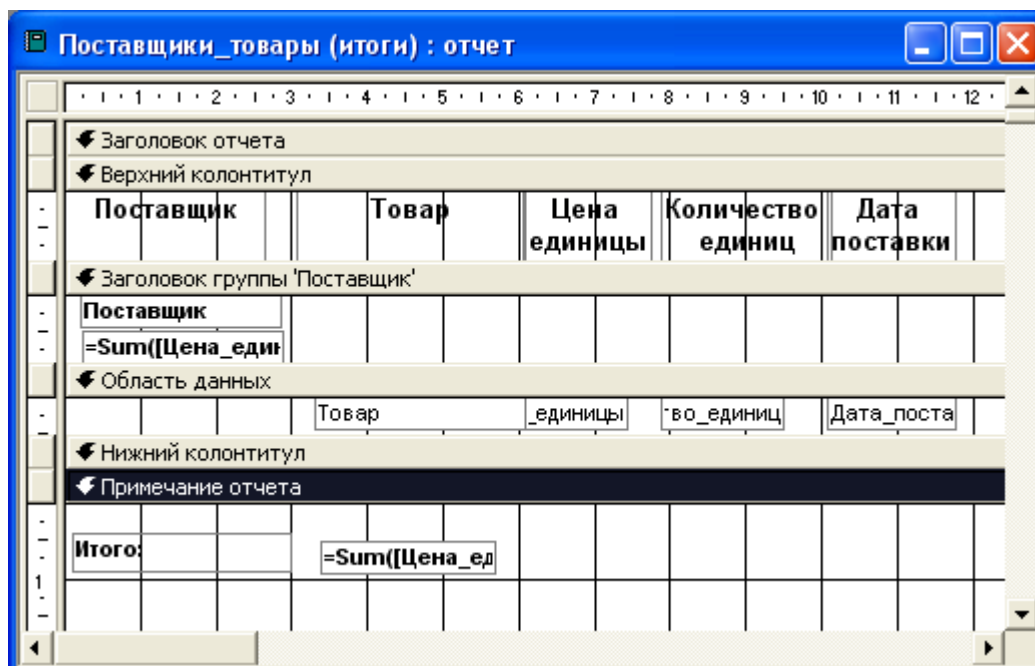
После этого в ячейку свойства **Данные** нужно ввести выражение, содержащее итоговую функцию (например, функцию Sum или Avg).

### **Пример 2.**

Видоизменим отчет, разработанный в предыдущем примере. В заголовок группы «Поставщик» добавим вычисляемое поле **ИтогПоставщика**. В свойстве **Данные** этого поля укажем выражение

**=Sum([Цена\_единицы]\*[Количество\_единиц]),**

а в свойстве **Формат поля** – **Денежный**. Скопируем это поле в примечание отчета, где зададим его имя **ОбщийИтог**:



В режиме предварительного просмотра полученный отчет будет выглядеть следующим образом:

Поставщик	Товар	Цена единицы	Количество единиц	Дата поставки
<b>SunGroup</b> <b>1 399,00 грн.</b>				
	Flash-карта	1,00	1	06.06.2002
	Видеокарта	422,00	1	01.01.2001
	Видеокарта	25,00	1	05.04.2002
	Flash-карта	1,00	1	06.06.2002
	Flash-карта	1,00	1	06.06.2002
	Flash-карта	1,00	1	06.06.2002
	Flash-карта	1,00	1	06.06.2002
	Flash-карта	1,00	1	06.06.2002
	Flash-карта	1,00	1	06.06.2002
	CD drive	315,00	3	08.12.2002
<b>Альфа</b> <b>7 009,00 грн.</b>				
	Клавиатура	55,00	10	24.12.2002
	Flash-карта	0,00	10	06.02.2002
	Flash-карта	25,50	10	02.07.2001
	Клавиатура	70,00	2	05.01.2002
	Видеокарта	230,00	1	07.01.2001
	Клавиатура	34,00	1	11.04.2002
	Видеокарта	330,00	10	24.12.2002
	Клавиатура	100,00	25	24.12.2002
<b>Артемон</b> <b>120,00 грн.</b>				
	Flash-карта	12,00	5	12.09.2002
	Видеокарта	10,00	6	18.05.2002
<b>Интер-Вест</b> <b>1 785,00 грн.</b>				
	CD drive	12,00	1	12.12.2002
	Клавиатура	167,00	5	11.05.2002
	Видеокарта	27,00	10	16.12.2002
	Клавиатура	167,00	4	07.05.2002
<b>Омега</b> <b>424,00 грн.</b>				
	Flash-карта	14,00	4	29.09.2002
	Клавиатура	34,00	7	12.10.2002
	CD drive	26,00	5	10.10.2002
<b>Итого:</b>	<b>10 737,00 грн.</b>			

Для вычисления в отчете **суммы с накоплением**, которая увеличивается для каждой записи, нужно добавить в область данных присоединенное поле или вычисляемое поле. Для расчета суммы с



накоплением по каждой группе записей такое поле добавляется в заголовок группы или в примечание группы. После ввода выражения для вычисления для свойства **Сумма с накоплением** необходимо задать значение, соответствующее типу суммы с накоплением:

а) для группы (значение сбрасывается до 0 в начале каждой группы более высокого уровня);

б) для всего (сумма накапливается до конца отчета).

Для вычисления **относительных значений** в отчете необходимо добавить поля, в которых вычисляются итоговые значения для записей, групп и всего отчета. Затем в подходящем разделе создается поле, в котором будет проводиться расчет относительных значений:

а) чтобы рассчитать долю каждого элемента относительно суммарного значения по группе или относительно общей суммы, поле помещают в область данных;

б) чтобы рассчитать долю суммы по группе относительно общей суммы, поле помещают в заголовок группы или в примечание группы (если отчет включает несколько уровней группировки, то поле следует поместить в заголовок или примечание группы того уровня, для которого требуется вычислить относительное значение).

После этого в ячейку свойства **Данные** следует ввести выражение, в котором значение элемента или промежуточное итоговое значение делится на общее итоговое значение (например, разделить значение элемента управления **ИтогПоставщика** на значение элемента управления **ОбщийИтог**). При необходимости для свойства данного поля **Формат поля** можно задать значение **Процентный**.

#### **Замечание:**

В качестве аргументов итоговой функции, например, функции Sum, нельзя использовать имена вычисляемых элементов управления, созданных в этом же отчете. В этом случае следует повторить выражение в элементе управления (например, =Sum([Количество\_единиц]\*[Цена\_единицы])). Однако, если вычисляемое поле включено в базовый запрос, то имя этого поля можно использовать в качестве аргумента итоговой функции.

Существует также возможность выполнять **нумерацию записей в области данных** отчета. Для этого следует добавить ЭУ Поле в область данных, ввести в ячейку свойства **Данные** выражение =1, в ячейке свойства **Сумма с накоплением** выбрать значение **Для всего**. Во время предварительного просмотра или при выводе отчета на печать номер каждой следующей записи будет увеличиваться на 1 (1, 2, 3...).

#### **Пример 3.**

В отчет, разработанный в примере 2, внесем изменения. В заголовок группы «Поставщики» добавим поле **ДоляПоставщика**, у которого в свойстве **Данные** указано выражение

**=[ИтогПоставщика]/[ОбщийИтог],**

а в свойстве **Формат поля – Процентный**. В область данных отчета добавим вычисляемое поле **Номер**, у которого свойство **Данные** имеет значение **=1**, а свойство **Сумма с накоплением – Для группы**. Получим следующий отчет:

Поставщик	Товар	Цена единицы	Количество единиц	Дата поставки	Стоимость партии
<b>SunGroup</b>	<b>1 399,00 грн.</b>	<b>13,03%</b>			
1	Flash-карта	1,00	1	06.06.2002	1,00 грн.
2	Видеокарта	422,00	1	01.01.2001	422,00 грн.
3	Видеокарта	25,00	1	05.04.2002	25,00 грн.
4	Flash-карта	1,00	1	06.06.2002	1,00 грн.
5	Flash-карта	1,00	1	06.06.2002	1,00 грн.
6	Flash-карта	1,00	1	06.06.2002	1,00 грн.
7	Flash-карта	1,00	1	06.06.2002	1,00 грн.
8	Flash-карта	1,00	1	06.06.2002	1,00 грн.
9	Flash-карта	1,00	1	06.06.2002	1,00 грн.
10	CD drive	315,00	3	08.12.2002	945,00 грн.
<b>Альфа</b>	<b>7 009,00 грн.</b>	<b>65,28%</b>			
1	Клавиатура	55,00	10	24.12.2002	550,00 грн.
2	Flash-карта	0,00	10	06.02.2002	0,00 грн.
3	Flash-карта	25,50	10	02.07.2001	255,00 грн.
4	Клавиатура	70,00	2	05.01.2002	140,00 грн.
5	Видеокарта	230,00	1	07.01.2001	230,00 грн.
6	Клавиатура	34,00	1	11.04.2002	34,00 грн.
7	Видеокарта	330,00	10	24.12.2002	3 300,00 грн.
8	Клавиатура	100,00	25	24.12.2002	2 500,00 грн.
<b>Артемон</b>	<b>120,00 грн.</b>	<b>1,12%</b>			
1	Flash-карта	12,00	5	12.09.2002	60,00 грн.
2	Видеокарта	10,00	6	18.05.2002	60,00 грн.
<b>Интер-Вест</b>	<b>1 785,00 грн.</b>	<b>16,62%</b>			
1	CD drive	12,00	1	12.12.2002	12,00 грн.
2	Клавиатура	167,00	5	11.05.2002	835,00 грн.
3	Видеокарта	27,00	10	16.12.2002	270,00 грн.
4	Клавиатура	167,00	4	07.05.2002	668,00 грн.
<b>Омега</b>	<b>424,00 грн.</b>	<b>3,95%</b>			
1	Flash-карта	14,00	4	29.09.2002	56,00 грн.
2	Клавиатура	34,00	7	12.10.2002	238,00 грн.
3	CD drive	26,00	5	10.10.2002	130,00 грн.
<b>Итого:</b>	<b>10 737,00 грн.</b>				

Для подсчета **количества записей во всем отчете** следует добавить вычисляемое поле в заголовок отчета или в примечание отчета и ввести в ячейку свойства **Данные** вычисляемого поля выражение **=Count(\*)**. Это позволит выполнить подсчет всех записей отчета, включая записи, имеющие пустые значения в некоторых полях.

Для подсчета **количества записей в каждой группе отчета** следует добавить вычисляемое поле в область данных и задать указанные значения для следующих свойств:

Свойство	Значение
Имя	ЧислоЗаписей
Данные	=1
Сумма с накоплением	Для группы
Вывод на экран	Нет

После этого нужно добавить вычисляемое поле в примечание группы и задать в качестве значения свойства **Данные** имя ЭУ из области данных, в котором рассчитывается сумма с накоплением, например, **=[ЧислоЗаписей]**.

### Размещение данных на страницах отчета

Иногда возникает необходимость печатать группу записей на одной странице или в одном столбце. Для этого при указании уровней группировки записей следует задать в бланке свойств группы свойству **Не разрывать** одно из значений:

- **полную группу** (заголовок группы, все записи в области данных и область примечаний группы будут печататься на одной странице или в одном столбце);
- **первую область данных** (заголовок группы будет печататься на странице или в столбце только в том случае, если удастся напечатать на той же странице первую запись из области данных).

Для печати каждой записи, группы или раздела на отдельной странице формы или отчета следует в окне свойств раздела выбрать одно из следующих значений свойства **Конец страницы**:

- **отсутствует**: раздел печатается на текущей странице (значение по умолчанию);
- **до раздела**: раздел печатается с начала новой страницы;
- **после раздела**: следующий раздел печатается с начала новой страницы;
- **до и после раздела**: раздел печатается с начала новой страницы; печать следующего раздела также начнется с новой страницы.

### **Замечания:**

- 1) Свойство **Конец страницы** определено для всех разделов, кроме верхнего и нижнего колонтитулов.
- 2) Для того чтобы печатать каждую запись на отдельной странице, следует задать значение **После раздела** для свойства **Конец страницы** области данных. Для печати каждой группы на отдельной странице необходимо задать значение **До раздела** для свойства **Конец страницы** заголовка группы или значение **После раздела** для свойства **Конец страницы** области примечаний группы.

В отчете с отсортированными или сгруппированными записями можно скрыть повторяющиеся данные. Для этого в окне свойств ЭУ, который может содержать повторяющиеся данные, необходимо задать для свойства **Не выводить повторы** значение **Да**. При этом Access будет печатать встретившееся значение в первой записи, в которой оно появляется, а если значение повторяется в записях на другой странице, то в начале новой страницы.

Существует ряд правил, соблюдение которых позволяет **повысить скорость печати отчетов:**

- 1) не создавайте перекрывающиеся элементы управления;
- 2) не создавайте лишние рисунки и другие графические объекты;
- 3) по возможности, не выполняйте сортировку и группировку по выражениям;
- 4) индексируйте поля, по которым проводится сортировка и группировка;
- 5) создавайте подчиненные отчеты на базе запросов, а не на базе таблиц; включайте в запросы только необходимые поля: лишние поля замедляют выполнение подчиненных отчетов;
- 6) индексируйте в подчиненном отчете все поля, связанные с главным отчетом;
- 7) *индексируйте в подчиненном отчете все поля, используемые в условиях отбора;*
- 8) старайтесь не использовать статистические функции по подмножеству; включите поле в базовый запрос отчета или используйте подчиненный отчет.

### 3. АВТОМАТИЗАЦИЯ ПРИЛОЖЕНИЯ С ПОМОЩЬЮ МАКРОСОВ

Разработанные отдельные объекты Access (таблицы, запросы, формы, отчеты) должны быть связаны в единое приложение. Для связывания объектов можно использовать *макросы*, которые, по сути, представляют собой небольшие программы на языке *макрокоманд* системы Access.

#### 3.1 Синтаксис ссылок

При создании макросов приходится *ссылаться* на формы, элементы управления (ЭУ) форм, а также на их свойства.

При ссылке на форму или отчет нужно сначала указать семейство, в которое входит объект, а затем – имя объекта. Открытые формы входят в семейство **Forms (Формы)**, открытые отчеты – в семейство **Reports (Отчеты)**. Для активной формы (то есть формы, которой передан *фокус*) в ссылках можно использовать простые имена ее ЭУ, например, **Код\_поставщика, Поставщик, Товар,...** Для других открытых форм нужно указывать *полные ссылки*.

*Полная ссылка* на форму состоит из двух частей:

Forms![имяФормы]

( например, Forms![Поставщики] ),

ссылка на ЭУ формы – из трех частей:

Forms![имяФормы]![имяЭлемента]

( например, Forms![Поставщики]![Код\_поставщика] ).

При написании ссылок квадратные скобки обязательны только в том случае, если имя объекта содержит пробелы или специальные символы.

Однако лучше всегда заключать имена в скобки, чтобы ссылки в любом случае оказались синтаксически верными.

Значения свойств формы и ее элементов управления задают обычно в режиме конструктора, используя окно свойств. Если же в процессе выполнения приложения возникает необходимость изменения значений некоторых свойств, то эти изменения можно указать с помощью макросов (такие действия возможны только для форм, у которых свойство **Допустимый режим** имеет значение **Форма**).

В окне свойств имена свойств выводятся с пробелами, но истинное имя свойства пробелов не содержит и должно указываться латинскими

буквами. Например, **Полосы прокрутки** – ScrollBars, **Вывод на экран** – Visible, **Значение** - Value. Имя свойства можно узнать, установив на него курсор в окне свойств и вызвав справку.

Ссылка на свойство формы имеет вид:

Forms![имяФормы].имяСвойства

( например, Forms![Поставщики].ScrollBars),

ссылка на свойство ЭУ формы:

Forms![имяФормы]![имяЭлемента].имяСвойства

( например, Forms![Поставщики]![Код\_поставщика].Visible)

Форма, внедренная в другую форму, содержится в ЭУ **Подчиненная форма** (ПФ). Ссылаться на подчиненную форму можно так же, как и на любой другой ЭУ формы:

Forms![имяФормы]![имяПФ]

( например, Forms![Составная]![Товары] ).

Ссылка на свойство ПФ:

Forms![имяФормы]![имяПФ].имяСвойства

( например, Forms![Составная]![Товары].Visible ).

При ссылке на ЭУ подчиненной формы используется Form — специальное свойство ЭУ **Подчиненная форма**, которое позволяет ссылаться на форму, внедренную в данный элемент:

Forms![имяФормы]![имяПФ].Form![имяЭлементаПФ]

Ссылка на свойство ЭУ подчиненной формы:

Forms![имяФормы]![имяПФ].Form![имяЭлементаПФ].имяСвойства

### 3.2 Примеры использования макросов

#### Пример 1

Рассмотрим форму, в которой представлены сведения о поставщике в следующем виде:

Форма **Сведения о поставщике** - Простая форма, она основана на таблице **Поставщики**, имя которой указано в свойстве **Источник записей**. С этой формой связан макрос **Поставщики**, представляющий собой *группу макросов*. Свойству формы **Открытие** назначен макрос **Поставщики.Спрятать список**, который запускается при открытии формы, но перед отображением ее ЭУ. Свойство **Разрешить добавление** должно иметь значение **Нет**.

Элементы управления формы:

- **Поля** - отображают соответствующие поля таблицы **Поставщики**. Для всех полей, кроме **Код\_поставщика**, назначены следующие значения свойств доступа: **Доступ** - Нет, **Блокировка** - Да, что позволяет только *просматривать* значения полей. По полю **Код\_поставщика** происходит поиск поставщика, название которого выбирается из списка. Поэтому свойства доступа этого поля должны иметь значения: **Доступ** - Да, **Блокировка** - Нет (поиск осуществляется при активизации кнопки **Найти**). В таблице **Поставщики** поля **Телефон** и **Адрес** являются необязательными.
- **Список Выбор** - предназначен для выбора имени поставщика, сведения о котором нужно вывести на экран. При открытии формы список не выводится, а появляется на экране только при щелчке на кнопке **Найти**. Свойству списка **После обновления** назначен макрос **Поставщики.Найти** (событие **После обновления** возникает в случае, когда в списке выбрано новое значение). При выборе из списка имени поставщика списку присваивается значение поля **Код\_поставщика** (свойство списка **Присоединенный столбец** имеет значение 1), которое затем учитывается в макрокоманде **НайтиЗапись**. Для вывода

списка с заголовком свойству **Заглавия столбцов** нужно задать значение **Да**.

- Командные кнопки (КК): **Поставки товаров**, **Новый товар**, **Новый поставщик**, **Все поставщики**, **Удалить** (на кнопке изображен крестик, всплывающая подсказка - **Удалить поставщика**), **Найти** (изображен бинокль, всплывающая подсказка - **Найти поставщика**), **Выход** (заккрыть форму).

КК **Поставки товаров** - свойству **Нажатие кнопки** назначен макрос **Поставщики.Товары**. По этой кнопке открывается форма **Товары, представленные поставщиком**, связанная с формой **Сведения о поставщике** по полю **Код\_поставщика**. Связь указывается в МК **ОткрытьФорму** в аргументе **Условие отбора** в виде следующего равенства:

[Код\_поставщика]=Forms![Сведения о поставщике]! [Код\_поставщика]

В левой части этого равенства приводится имя поля связи активной формы (форма **Товары, представленные поставщиком**), в правой части равенства должна использоваться полная ссылка на поле связи формы **Сведения о поставщике**.

Поля связи можно не выводить в формах, лишь бы они содержались в объектах, на которых основаны формы. Если некоторый поставщик в текущий момент не представил товар, то вместо пустой формы выводится сообщение: *У поставщика нет товаров!*

Сообщение выдается макросом **Поставщики.Просмотр товаров**, назначенным свойству **Открытие** формы **Товары, представленные поставщиком**. Этот макрос выполняется после открытия формы, но перед выводом ее элементов управления.

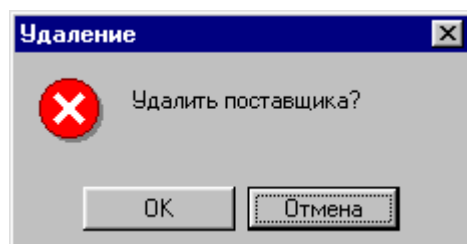
КК **Новый товар** - свойству **Нажатие кнопки** назначен макрос **Поставщики.Новый товар**. По этой кнопке открывается форма **Новый товар**, с помощью которой осуществляется ввод данных о новой поставке товара в таблицу **Товары**.

КК **Новый поставщик** – открывается форма **Новый поставщик**, предназначенная для ввода данных о поставщике товаров. Свойству кнопки **Нажатие кнопки** соответствует макрос **Поставщики.Новый поставщик**.

КК **Все поставщики** - открывается форма **Все поставщики**, в которой выводятся сведения о всех имеющихся поставщиках.

КК **Удалить** - предназначена для удаления выведенной на экран записи о поставщике. Для контроля за действием удаления выводится соответствующее сообщение, требующее подтверждения пользователя на удаление:





В макросе **Поставщики.Удалить**, связанном с этой кнопкой, использована системная функция *MsgBox()*, предназначенная для вывода диалоговых окон сообщений, ожидания ответа пользователя и возврата значения, основанного на выборе пользователя.

*Синтаксис функции:*

`MsgBox(“текст-сообщения”[; тип-сообщения[; “заголовок”]])`

*“текст-сообщения”* - строка, выводимая в качестве сообщения в окне;

*тип-сообщения* - числовое выражение, определяющее кнопки и пиктограммы окна;

*“заголовок”* - строка, выводимая в заголовке окна сообщения.

Если присутствует только первый аргумент функции, то в окне будет только кнопка ОК (без пиктограмм и заголовка). Второй аргумент функции определяется с помощью следующих трех таблиц:

Тип и количество кнопок:

Значение	Отображаемая кнопка
0	ОК
1	ОК, Отмена
2	Прервать, Повторить, Пропустить
3	Да, Нет, Отмена
4	Да, Нет
5	Повторить, Отмена

Тип пиктограммы:

Значение	Отображение
0	Пиктограмма отсутствует
16	Критическая ошибка
32	? Предупреждение

<b>48</b>	! Предупреждение
<b>64</b>	Информация

Кнопка, заданная по умолчанию:

<b>Значение</b>	<b>Отображаемая кнопка</b>
<b>0</b>	Первая
<b>256</b>	Вторая
<b>512</b>	Третья

Для используемой в макросе **Поставщики.Удалить** функции *MsgBox()* второй параметр получается так:  $1+16+256=273$ .

Возвращаемое функцией значение зависит от выбранной кнопки:

<b>Выбранная кнопка</b>	<b>Значение</b>
ОК	<b>1</b>
Отмена	<b>2</b>
Прервать	<b>3</b>
Повторить	<b>4</b>
Пропустить	<b>5</b>
Да	<b>6</b>
Нет	<b>7</b>

КК **Найти** - по нажатию кнопки запускается макрос **Поставщики.Вывести список**, который выводит список **Выбор**, содержащий перечень поставщиков, на экран и передает ему фокус. При выборе из списка элемента (щелчок по некоторому имени) возникает событие **После обновления**, которому назначен макрос **Поставщики.Найти**. С помощью этого макроса в таблице **Поставщики** отыскивается выбранный поставщик. И так как найденная запись становится текущей, то она выводится на экран.

Макрос **Поставщики** имеет вид:

Поставщики : макрос			
Имя макроса	Условие	Макрокоманда	Примечание
Просмотр товаров	IsNull([Код_поставщика])	Сообщение	1
	...	Заккрыть	2
Товары		ОткрытьФорму	3
Новый товар		ОткрытьФорму	4
Все поставщики		ОткрытьФорму	5
Удалить	MsgBox("Удалить поста	ОтменитьСобытие	6
	...	ОстановитьМакрос	7
		КомандыКлавиатуры	8
		ВыполнитьКоманду	9
Спрятать список		ЗадатьЗначение	10
Вывести список		ЗадатьЗначение	11
		КЭлементуУправления	12
Найти		КЭлементуУправления	13
		НайтиЗапись	14
		ЗапускМакроса	15
▶ Новый поставщик		ОткрытьФорму	16

### Макрокоманды макроса:

#### Поставщики.Просмотр товаров

##### 1) МК *Сообщение*

Сообщение: У поставщика нет товаров!

Сигнал: Да

Тип: Информационное

Условие выполнения МК: IsNull([Код\_поставщика])

##### 2) МК *Заккрыть*

Тип объекта: Форма

Имя объекта: Товары, представленные поставщиком

Сохранение: Да

#### Поставщики.Товары

##### 3) МК *ОткрытьФорму*

Имя формы: Товары, представленные поставщиком

Режим: Форма

Условие отбора:

[Код\_поставщика]=Forms![Сведения о поставщике]![Код\_поставщика]

Режим данных: Только чтение

Режим окна: Обычное

#### Поставщики.Новый товар

##### 4) МК *ОткрытьФорму*

Имя формы: Новый товар

Режим данных: Добавление

#### Поставщики.Все поставщики

- 5) МК *ОткрытьФорму*  
Имя формы: Все поставщики  
Режим данных: Только чтение

#### **Поставщики.Удалить**

- 6) МК *ОтменитьСобытие*  
Условие выполнения МК :  
MsgBox(“Удалить поставщика?”;273;”Удаление”)<>1

- 7) МК *ОстановитьМакрос*

- 8) МК *КомандыКлавиатуры*  
Клавиши: Enter  
Пауза: Да

- 9) МК *ВыполнитьКоманду*  
Команда: DeleteRecord

#### **Поставщики.Спрятать список**

- 10) МК *ЗадатьЗначение*  
Элемент: [Выбор].[Visible]  
Выражение:Нет

#### **Поставщики.Вывести список**

- 11) МК *ЗадатьЗначение*  
Элемент: [Выбор].[Visible]  
Выражение:Да

- 12) МК *КЭлементуУправления*  
Имя элемента: Выбор

#### **Поставщики.Найти**

- 13) МК *КЭлементуУправления*  
Имя элемента: Код\_поставщика

- 14) МК *НайтиЗапись*  
Образец поиска: =[Выбор]  
Совпадение: Поля целиком  
Область поиска: Все  
С учетом формата поля: Да  
С учетом регистра: Нет  
Только в текущем поле: Да  
Первое вхождение: Да

15) МК *ЗапускМакроса*

Имя макроса: Поставщики.Спрятать список

**Поставщики.Новый поставщик**

16) МК *ОткрытьФорму*

Имя формы: Новый поставщик

Режим данных: Добавление

---

Форма **Новый товар** используется для ввода новой записи в таблицу **Товары** (свойство формы **Ввод данных** имеет значение **Да**, свойство **Подпись** – значение **Сведения о новой поставке товара**):

(с формой связан макрос **Новый товар**).

В форме содержатся элементы управления типа **Поле**, **Поле со списком**, **Командная кнопка** (КК).

Ключевое поле **Код\_поставки** является **Счетчиком**, поэтому система автоматически заносит его значение. Это поле можно не выводить в форме, так как оно не является для пользователя информационным.

Значение поля **Код\_поставщика** передается из формы **Сведения о поставщике**, поэтому свойство **Значение по умолчанию** этого поля имеет следующее значение:

=[Forms]![Сведения о поставщике]![Код\_поставщика]

Так как пользователь не должен заполнять поля **Код\_поставки** и **Код\_поставщика**, то они выводятся затененными (свойства **Доступ** и **Блокировка** имеют значения **Нет**). Остальные поля формы должны

заполняться, так как соответствующие им поля таблицы **Товары** являются **обязательными**.

КК **Занести** связана с макросом **Новый товар.Занести\_товар**. Если все поля формы **Новый товар** заполнены, то в таблицу **Товар** заносится новая запись. Если же не для всех полей заданы значения, то пользователю выдается сообщение: *Не все поля заданы!* В этом случае новая запись не создается.

КК **Отменить** связана с макросом **Новый товар.Отменить\_товар**. При щелчке по этой кнопке операция ввода новой записи отменяется.

КК, находящаяся справа от поля **Товар** (всплывающая подсказка **Новый вид товара**), может использоваться в том случае, если в раскрывающемся списке товаров не содержится название поступающего товара (то есть в таблице **Каталог** нет соответствующей записи).

При щелчке по кнопке открывается форма **Новый вид товара**, в которой можно указать название нового товара (с формой связан макрос **Новый вид товара**):

The image shows two overlapping forms from a database application. The background form, titled "Сведения о новой поставке товара", has fields for "Код поставки" (with a "(Счетчик)" icon), "Код поставщика" (containing the number 7), "Товар" (a dropdown menu), "Дата поставки", "Цена единицы", and "Количество ед". The foreground form, titled "Новый вид товара", has fields for "Код товара" (containing the number 29), "Товар", and "Производитель". Both forms have "Отменить" and "Занести" buttons at the bottom.

Поля **Товар** и **Производитель** этой формы должны заполняться пользователем. Ключевое поле **Код\_товара** имеет тип **Числовой**, поэтому вычисление его значения можно автоматизировать.

Для этого в свойстве **Значение по умолчанию** нужно указать выражение

$$=DMax("[Код_товара]"; "[Каталог]") + 1$$

Функция **DMax** относится к категории *доменных агрегатных функций* (*функций по подмножеству*), которые используются в тех случаях, когда необходимо вычислить итоговые значения для набора записей закрытых таблиц или запросов.

Синтаксис указанных функций имеет вид:

*ИмяФункции(выражение; домен[; условие])*

*домен* – имя таблицы или запроса, для которых вычисляется показатель;

*выражение* - имя поля или выражение, по значениям которого вычисляется показатель;

*условие* - используется для отбора записей из домена (необязательный параметр).

Каждый параметр должен быть заключен в кавычки.

Наиболее часто используются следующие агрегатные функции:

DCount, DMin, DMax, DAvg, Dsum.

Действуют они таким же образом, что и соответствующие *итоговые функции*

После заполнения полей формы **Новый вид товара** по кнопке **Занести** осуществляется ввод новой записи в таблицу **Каталог** и закрытие формы.

Для того чтобы вновь введенное название товара отобразилось в раскрывающемся списке, в макросе **Новый вид товара** используется МК **Обновление**. Эта МК обновляет данные в объекте БД путем повторного просмотра *источника данных* (например, повторно выполняется запрос, указанный в свойстве **Источник данных** формы или ее ЭУ).

МК имеет один аргумент: **Имя элемента**. Его значением является в рассматриваемом примере имя поля со списком **Код\_товара**. МК **Обновление** обновляет только один объект. Поэтому, если в форме существуют ЭУ, требующие обновления отображаемых данных, то для каждого ЭУ нужно использовать отдельную МК **Обновление**.

**Внимание!** При создании ЭУ **Поле со списком** с помощью мастера система присваивает ему имя типа **ПолеСоСписком1**. Такое системное имя нужно заменить на реальное, например, на **Код\_товара** для поля со списком формы **Новый товар**.

КК, находящаяся справа от поля **Дата\_поставки** (всплывающая подсказка **Использование календаря**), открывает форму **Выбор даты**, если пользователь желает ввести дату не вручную, а с помощью ЭУ **Календарь**:

Сведения о новой поставке товара

Код поставки

(Счетчик)

Код поставщика

7

Товар

Дата поставки

Цена единицы

Количество единиц

Отменить

Занести

Укажите дату поставки:

янв 2010

янв

2010

Пн	Вт	Ср	Чт	Пт	Сб	Вс
28	29	30	31	1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	31
1	2	3	4	5	6	7

Занести

КК Занести (форма **Выбор даты**) связана с макросом **Новый товар.Занести\_дату**.

Макрос **Новый товар** имеет вид:

Новый товар : макрос				
	Имя макроса	Условие	Макрокоманда	Примечание
	Занести_товар	IsNull([Формы])!	Сообщение	1
		...	ОстановитьМакрос	2
			Закрыть	3
	Отменить_товар		ЗадатьЗначение	4
			ЗадатьЗначение	5
			ЗадатьЗначение	6
			ЗадатьЗначение	7
			ВыполнитьКоманду	8
			Закрыть	9
	Новый вид товара		ОткрытьФорму	10
	Вызов календаря		ОткрытьФорму	11
	Занести_дату		ВыделитьОбъект	12
			КЭлементуУправления	13
			ЗадатьЗначение	14
			ВыделитьОбъект	15
			Закрыть	16

**Макрокоманды макроса:**

**Новый товар.Занести\_товар**



1) МК *Сообщение*

Сообщение: Не все поля заданы!

Сигнал: Да

Тип: Предупреждение!

Условие выполнения МК:

IsNull([Forms]![Новый товар]![Дата\_поставки]) OR

IsNull([Forms]![Новый товар]![Цена\_единицы]) OR ...

2) МК *ОстановитьМакрос*

3) МК *Закрыть*

Тип объекта: Форма

Имя объекта:Новый товар

**Новый товар.Отменить\_товар**

4) МК *ЗадатьЗначение*

Элемент: [Код\_товара]

Выражение: DMax("[Код\_товара]";"[Товары]")

5) МК *ЗадатьЗначение*

Элемент: [Дата\_поставки]

Выражение: Date()

6) МК *ЗадатьЗначение*

Элемент: [Цена\_единицы]

Выражение: 1

7) МК *ЗадатьЗначение*

Элемент: [Количество\_единиц]

Выражение: 1

8) МК *ВыполнитьКоманду*

Команда: DeleteRecord

9) МК *Заккрыть*

Тип объекта: Форма

Имя объекта: Новый товар

**Новый товар.Новый вид товара**

10) МК *ОткрытьФорму*

Имя формы: Новый вид товара

Режим: Форма

**Новый товар.Вызов календаря**

11) МК *ОткрытьФорму*

Имя формы: Выбор даты

Режим: Форма

**Новый товар.Занести\_дату**

12) МК *ВыделитьОбъект*

Тип объекта: Форма

Имя объекта: Новый товар

13) МК *КЭлементуУправления*

Имя элемента: [Дата\_поставки]

14) МК *ЗадатьЗначение*

Элемент: [Формы]![Новый товар]![Дата\_поставки]

Выражение: [Формы]![Выбор даты]![Календарь].[Value]

15) МК *ВыделитьОбъект*

Тип объекта: Форма

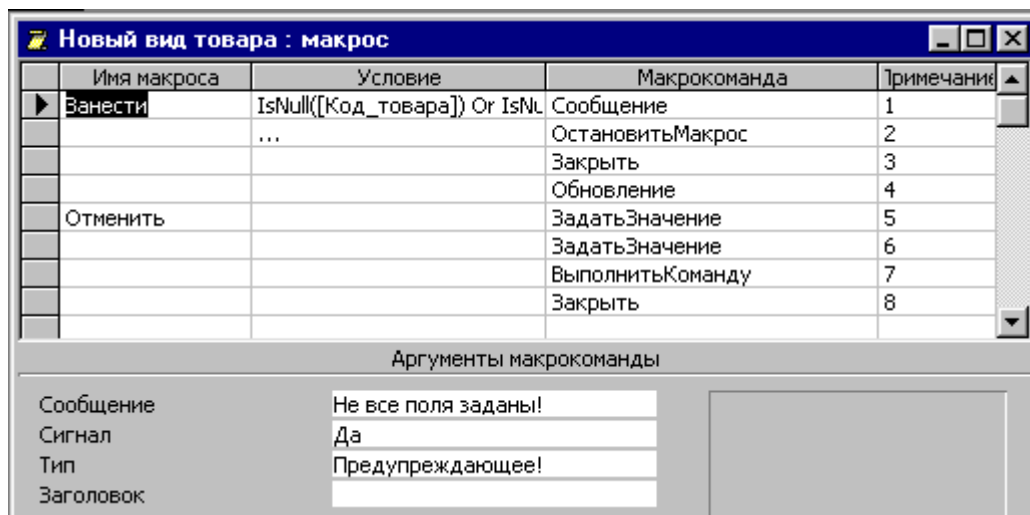
Имя объекта: Выбор даты

### 16) МК *Заккрыть*

Тип объекта: Форма

Имя объекта: Выбор даты

Макрос **Новый вид товара** имеет вид:



### Макрокоманды макроса:

#### Новый вид товара.Занести

##### 1) МК *Сообщение*

Сообщение: Не все поля заданы!

Тип: Предупреждающее!

Условие выполнения МК:

IsNull([Товар]) OR IsNull([Производитель])

##### 2) МК *ОстановитьМакрос*

##### 3) МК *Заккрыть*

Тип объекта: Форма

Имя объекта: Новый вид товара

##### 4) МК *Обновление*

Имя элемента: Код\_товара

#### Новый вид товара.Отменить

##### 5) МК *ЗадатьЗначение*

Элемент: [Товар]

Выражение: 'а'

6) МК *ЗадатьЗначение*

Элемент: [Производитель]

Выражение: 'а'

7) МК *ВыполнитьКоманду*

Команда: DeleteRecord

8) МК *Заккрыть*

Тип объекта: Форма

Имя объекта: Новый вид товара

---

Форма **Новый поставщик** предназначена для ввода новой записи в таблицу **Поставщики** (свойство формы **Ввод данных** имеет значение **Да**):

(с формой связан макрос **Новый поставщик**).

В форме содержатся элементы управления типа **Поле**, **Поле со списком**, **Командная кнопка** (КК).

Ключевое поле **Код\_поставщика** является **Счетчиком**, поэтому система автоматически заносит его значение. Это поле можно не выводить в форме, так как оно не является для пользователя информационным. Остальные поля формы, кроме полей **Телефон** и **Адрес**, должны заполняться, так как соответствующие им поля таблицы **Поставщики** являются **обязательными**.

КК **Занести** связана с макросом **Новый поставщик.Занести\_пост.** Если все обязательные поля формы **Новый поставщик** заполнены, то в

таблицу **Поставщики** заносится новая запись. Если же не для всех полей заданы значения, то пользователю выдается сообщение:

*Не все обязательные поля заданы!*

В этом случае новая запись не создается.

КК **Отменить** связана с макросом **Новый поставщик.Отменить\_пост.** При щелчке по этой кнопке операция ввода новой записи отменяется.

КК, находящаяся справа от поля **Город** (всплывающая подсказка **Новый город**), может использоваться в том случае, если в раскрывающемся списке городов не содержится название нужного города (то есть в таблице **Города** нет соответствующей записи).

При щелчке по кнопке открывается форма **Новый город**, в которой можно указать название нового города (с кнопкой связан макрос **Новый поставщик.Новый город**). Если же нужный город находится в стране, не содержащейся в таблице **Страны**, то можно также ввести и название новой страны:

The image shows three overlapping Windows forms. The background form is titled 'Новый поставщик' and contains fields for 'Код поставщика:' (with a '(Счетчик)' label), 'Поставщик:', 'Город:' (with a dropdown arrow and a small icon button), 'Форма собственности:', 'Телефон:', and 'Адрес:'. A red 'Отменить' button is at the bottom. Overlaid on this is the 'Новый город' form, which has fields for 'Страна:' (with a dropdown arrow and icon button), 'Город:', and 'Код города:'. It also has a red 'Отменить' button. The foreground form is titled 'Новая страна' and contains fields for 'Страна:' and 'Код страны:' (with a '(Счетчик)' label). It has red 'Отменить' and 'Занести' buttons at the bottom.

КК, находящаяся справа от поля **Форма собственности** (всплывающая подсказка **Новая форма собственности**), может использоваться в том случае, если в раскрывающемся списке не содержится название нужной формы собственности.

При щелчке по кнопке открывается форма **Новая собственность**, в которой можно указать новое название (с кнопкой связан макрос **Новый поставщик.Новая форма собственности**):

The image shows two overlapping Windows forms. The background form, titled "Новый поставщик", has a light blue header and a white body. It contains the following fields: "Код поставщика:" with a text box and "(Счетчик)" label; "Поставщик:" with a text box; "Город:" with a dropdown menu and a button with a list icon; "Форма собственности:" with a dropdown menu and a button with a list icon. The foreground form, titled "Новая форма собственности", also has a light blue header and a white body. It contains: "Форма собственности:" with a text box; "Код собственности:" with a text box and "(Счетчик)" label; and two buttons at the bottom: "Отменить" (highlighted in red) and "Занести".

Макрос **Новый поставщик** имеет следующий вид:

Новый поставщик : макрос				
	Имя макроса	Условие	Макрокоманда	Примечание
▶	Новый город		ОткрытьФорму	1
	Новая страна		ОткрытьФорму	2
	Занести_стр	IsNull([Формы])!([Новая с	Сообщение	3
		...	ОстановитьМакрос	4
			Закрыть	5
			Обновление	6
	Занести_гор	IsNull([Формы])!([Новый	Сообщение	7
		...	ОстановитьМакрос	8
			Закрыть	9
			Обновление	10
	Занести_пост	IsNull([Поставщик]) Or	Сообщение	11
		...	ОстановитьМакрос	12
			Закрыть	13
			Обновление	14
			Обновление	15
	Новая собственность		ОткрытьФорму	16
	Занести_собств	IsNull([Форма_собствен	Сообщение	17
		...	ОстановитьМакрос	18
			Закрыть	19
			Обновление	20
	Отменить_пост		ЗадатьЗначение	21
			ЗадатьЗначение	22
			ЗадатьЗначение	23
			ВыполнитьКоманду	24
			Закрыть	25

	Отменить_город		ЗадатьЗначение	26
			ЗадатьЗначение	27
			ВыполнитьКоманду	28
			Закрыть	29
▶	Отменить_страну		ЗадатьЗначение	30
			ВыполнитьКоманду	31
			Закрыть	32
	Отменить_собств		ЗадатьЗначение	33
			ВыполнитьКоманду	34
			Закрыть	35

### Макрокоманды макроса:

#### Новый поставщик.Новый город

##### 1) МК *ОткрытьФорму*

Имя формы: Новый город  
Режим: Форма

#### Новый поставщик.Новая страна

##### 2) МК *ОткрытьФорму*

Имя формы: Новая страна  
Режим: Форма

### **Новый поставщик.Занести\_стр**

#### 3) МК *Сообщение*

Сообщение: Не все поля заданы!

Тип: Предупреждающее!

Условие выполнения МК:

IsNull([Формы]![Новая страна]![Страна])

#### 4) МК *ОстановитьМакрос*

#### 5) МК *Закрыть*

Тип объекта: Форма

Имя объекта: Новая страна

#### 6) МК *Обновление*

Имя элемента: Код\_страны

### **Новый поставщик.Занести\_гор**

#### 7) МК *Сообщение*

Сообщение: Не все поля заданы!

Тип: Предупреждающее!

Условие выполнения МК:

IsNull([Формы]![Новый город]![Код\_страны]) or

IsNull([Формы]![Новый город]![Город])

#### 8) МК *ОстановитьМакрос*

#### 9) МК *Закрыть*

Тип объекта: Форма

Имя объекта: Новый город

#### 10) МК *Обновление*

Имя элемента: Код\_города

### **Новый поставщик.Занести\_пост**

#### 11) МК *Сообщение*

Сообщение: Не все обязательные поля заданы!

Тип: Предупреждающее!

Условие выполнения МК:

IsNull([Поставщик]) or IsNull([Код\_города]) or IsNull([Код\_собственности])

#### 12) МК *ОстановитьМакрос*

#### 13) МК *Закрыть*



Тип объекта: Форма  
Имя объекта: Новый поставщик

14) МК *Обновление*

Имя элемента: Код\_поставщика

15) МК *Обновление*

Имя элемента:

**Новый поставщик.Новая собственность**

16) МК *ОткрытьФорму*

Имя формы: Новая форма собственности

Режим: Форма

**Новый поставщик.Занести\_собств**

17) МК *Сообщение*

Сообщение: Не все поля заданы!

Тип: Предупреждающее!

Условие выполнения МК: IsNull([Форма\_собственности])

18) МК *ОстановитьМакрос*

19) МК *Закрыть*

Тип объекта: Форма

Имя объекта: Новая форма собственности

20) МК *Обновление*

Имя элемента: Код\_собственности

**Новый поставщик.Отменить\_пост**

21) МК *ЗадатьЗначение*

Элемент: [Поставщик]

Выражение: 'a'

22) МК *ЗадатьЗначение*

Элемент: [Код\_города]

Выражение: Dmax("[Код\_города]"; "[Поставщики]")

23) МК *ЗадатьЗначение*

Элемент: [Код\_собственности]

Выражение: Dmin("[Код\_собственности]"; "[Поставщики]")

24) МК *ВыполнитьКоманду*

Команда: УдалитьЗапись

25) МК *Заккрыть*

Тип объекта: Форма

Имя объекта: Новый поставщик

#### **Новый поставщик.Отменить\_город**

26) МК *ЗадатьЗначение*

Элемент: [Город]

Выражение: 'а'

27) МК *ЗадатьЗначение*

Элемент: [Код\_страны]

Выражение: Dmax("[Код\_страны]"; "[Города]")

28) МК *ВыполнитьКоманду*

Команда: УдалитьЗапись

29) МК *Заккрыть*

Тип объекта: Форма

Имя объекта: Новый город

#### **Новый поставщик.Отменить\_страну**

30) МК *ЗадатьЗначение*

Элемент: [Страна]

Выражение: 'а'

31) МК *ВыполнитьКоманду*

Команда: УдалитьЗапись

32) МК *Заккрыть*

Тип объекта: Форма

Имя объекта: Новая страна

#### **Новый поставщик.Отменить\_собств**

33) МК *ЗадатьЗначение*

Элемент: [Форма\_собственности]

Выражение: 'а'

34) МК *ВыполнитьКоманду*

Команда: УдалитьЗапись

35) МК *Заккрыть*

Тип объекта: Форма

Имя объекта: Новая форма собственности

---

## Пример 2

Рассмотрим форму **Сведения о товарах**, в которой для организации меню используется ЭУ **Группа**:

The screenshot shows a form titled "СВЕДЕНИЯ О ТОВАРАХ". Inside the form, there is a menu group with four radio buttons. The first radio button, labeled "Товары за период", is selected. The other three radio buttons are labeled "Товары поставщика", "Поставки товара", and "Поставщик-товары". At the bottom of the form, there are two buttons: "Просмотр" and a button with a document icon and a plus sign.

С формой связан макрос **Сведения о товарах**, представляющий собой *группу макросов*. Свойству формы **Открытие** назначен макрос **Сведения о товарах. Скрыть**, который запускается при открытии формы, но перед отображением ее ЭУ. С помощью этого макроса список поставщиков и список товаров при открытии формы “скрываются” и выводятся на экран только при выборе соответствующих пунктов меню.

Элементы управления формы:

- Группа **Выбор** - имеет четыре переключателя (то есть четыре пункта меню); по умолчанию активным является первый переключатель. Свойству **После обновления** назначен макрос **Сведения о товарах. Вывод списка**, который запускается после того, как были обновлены данные в **Группе** (обновление здесь понимается как выбор нового пункта меню, то есть щелчок по соответствующему переключателю).
- Список **Выбор поставщика** - отображает на экране список поставщиков. При выборе имени некоторого поставщика списку присваивается значение поля **Код\_поставщика** (свойство списка

**Присоединенный столбец** имеет значение **1**), которое затем учитывается в условии отбора записей макрокоманды, открывающей форму **Товары поставщика**. Для вывода списка с заголовком свойству **Заглавия столбцов** нужно задать значение **Да**. Рекомендуется также задать первоначальное значение списка. Для этого свойству **Значение по умолчанию** нужно присвоить выражение

=[Выбор поставщика].[ItemData](1)

Задание значения по умолчанию исключает ситуацию, когда при невыбранном параметре нажимается кнопка **Просмотр**.

- Список **Выбор товара** - отображает на экране перечень возможных товаров. При выборе товара списку присваивается значение поля **Код\_товара**, которое будет учитываться в условии отбора запроса **Партии товара**. Список также нужно выводить с заголовком (смотри предыдущие рекомендации).
- КК **Просмотр** - свойству **Нажатие кнопки** назначен макрос **Сведения о товарах. Просмотр**. По этой кнопке открывается форма, соответствующая выбранному пункту меню (активизированному переключателю группы **Выбор**).

Пункт **Товары за период** ([Выбор]=1): открывается форма **Выбор периода**.

Пункт **Товары поставщика** ([Выбор]=2): значение поля **Код\_поставщика**, соответствующее выбранному из списка имени поставщика, передается свойству **Условие отбора** макрокоманды, открывающей форму **Товары поставщика**.

Для того чтобы не выводилась на экран пустая форма (это возможно в случае, если поставщик в текущий момент не представил товар), свойству **Открытие формы Товары поставщика** назначается макрос **Сведения о товарах. Проверка\_2**.

**СВЕДЕНИЯ О ТОВАРАХ**

☐ Товары за период  
☒ Товары поставщика  
☐ Поставки товара  
☐ Поставщик-товары

Укажите поставщика :

Поставщик	Город
SupGroup	Гомель
Альфа	Петербург
Интер-Вест	Донецк
МКС	Киев

Просмотр

Пункт **Поставки товара** ([Выбор]=3): значение поля **Код\_товара**, соответствующее выбранному наименованию товара, передается в запрос **Партии товара** в качестве значения параметра запроса. В запросе параметр должен принять вид:

Forms![Сведения о товарах]![Выбор товара])

На основе этого запроса создана форма **Партии товара**. Свойству формы **Открытие** назначается макрос **Сведения о товарах. Проверка\_3**.

**СВЕДЕНИЯ О ТОВАРАХ**

☐ Товары за период  
☐ Товары поставщика  
☒ **Поставки товара**  
☐ Поставщик-товары

Выберите товар :

Товар	Производитель
CD drive	Mitsumi
CD drive	Panasonic
Pentium-166 RAM1	Вист
ZIP drive	Paragon
Видеокарта	Matrox
Винчестерские ди	Seagate
Винчестерские ди	Quantum

Пункт **Поставщик-Товары** ([Выбор]=4): открывается составная форма, в которой выводятся сведения о поставщике и представленных им товарах.

**Составная**

**СВЕДЕНИЯ О ПОСТАВЩИКЕ И ЕГО ТОВАРАХ**

Поставщик: 
 Город:

Собственность: 
 Страна:

Телефон: 
 Адрес:

Товар	Производитель	Дата поставки	Цена единицы	Количество единиц
CD drive	Mitsumi	13.12.2000	125	5
Видеокарта	Matrox	08.12.2002	315	3
Материнская плата	Acer	01.01.2000	12	1
Модуль памяти	KingStone	06.06.2002	1	1
Модуль памяти	KingStone	06.06.2002	1	1

Общая сумма:

Запись:      из 12

Запись:      из 4

Макрос **Сведения о товарах** имеет вид:

	Имя макроса	Условие	Макрокоманда	Примечание
▶	Вывод списка		ЗапускМакроса	1
		[Выбор]=2	ЗадатьЗначение	2
		...	ОстановитьМакрос	3
		[Выбор]=3	ЗадатьЗначение	4
	Просмотр	[Выбор]=1	ОткрытьФорму	5
		...	ОстановитьМакрос	6
		[Выбор]=2	ОткрытьФорму	7
		...	ОстановитьМакрос	8
		[Выбор]=3	ОткрытьФорму	9
		...	ОстановитьМакрос	10
		[Выбор]=4	ОткрытьФорму	11
		...	ОстановитьМакрос	12
	Проверка_2	IsNull([Forms]![Тов	Сообщение	13
		...	Закрыть	14
	Проверка_3	IsNull([Forms]![Пар	Сообщение	15
		...	Закрыть	16
	Скрыть		ЗадатьЗначение	17
			ЗадатьЗначение	18
	Выход		Закрыть	19

### Макрокоманды макроса:

#### Сведения о товарах.Вывод списка

##### 1) МК *ЗапускМакроса*

Имя макроса:                      Сведения о товарах.Скрыть

##### 2) МК *ЗадатьЗначение*

Элемент:                            [Выбор поставщика].[Visible]

Выражение:                        Да

##### 3) МК *ОстановитьМакрос*

##### 4) МК *ЗадатьЗначение*

Элемент:                            [Выбор товара].[Visible]

Выражение:                        Да

#### Сведения о товарах.Просмотр

##### 5) МК *ОткрытьФорму*

Имя формы:                        Выбор периода

Режим данных:                    Изменение

##### 6) МК *ОстановитьМакрос*

##### 7) МК *ОткрытьФорму*

Имя формы:                        Товары поставщика

Режим данных:                    Только чтение

Условие выполнения МК:

[Код\_поставщика]=[Forms]![Сведения о товарах]![Выбор поставщика]

8) МК *ОстановитьМакрос*

9) МК *ОткрытьФорму*

Имя формы: Партии товара  
Режим данных: Только чтение

10) МК *ОстановитьМакрос*

11) МК *ОткрытьФорму*

Имя формы: Составная

12) МК *ОстановитьМакрос*

### **Сведения о товарах.Проверка\_2**

13) МК *Сообщение*

Сообщение: У данного поставщиков нет поставок!

Условие выполнения МК:

IsNull([Forms]![Товары поставщика]![Код\_поставщика])

14) МК *Закреть*

Тип объекта: Форма  
Имя объекта: Товары поставщика

### **Сведения о товарах.Проверка\_3**

15) МК *Сообщение*

Сообщение: Данный товар не поступал!

Условие выполнения МК:

IsNull([Forms]![Партии товара]![Код\_товара])

16) МК *Закреть*

Тип объекта: Форма  
Имя объекта: Партии товара

### **Сведения о товарах..Скрыть**

17) МК *ЗадатьЗначение*

Элемент: [Выбор поставщика].[Visible]  
Выражение: Нет

18) МК *ЗадатьЗначение*

Элемент: [Выбор товара].[Visible]  
Выражение: Нет

### **Сведения о товарах.Выход**



19) МК *Заккрыть*

Тип объекта:

Форма

Имя объекта

Сведения о товарах

Форма **Выбор периода** имеет вид:

Укажите диапазон дат:

Декабрь 1999

Вс	Пн	Вт	Ср	Чт	Пт	Сб
28	29	30	1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	31	1
2	3	4	5	6	7	8

Начальная дата

Конечная дата

ОТМЕНА

ОК

(связана с макросом **Выбор периода**).

Форма имеет следующие ЭУ:

- поля **Начальная дата** и **Конечная дата** (свободные ЭУ),
- элемент ActiveX **Календарь**,
- КК с надписями **Начальная дата**, **Конечная дата**, **Отмена**, **ОК**.

На **Календаре** набирается дата (год, месяц, день месяца) и щелчком по КК **Начальная дата** засылается в поле **Начальная дата** (макрос **Выбор периода. начальная дата**).

По кнопке **Конечная дата** выбранное значение даты засылается в поле **Конечная дата** с помощью макроса **Выбор периода. конечная дата**.

С кнопкой **ОК** связан макрос **Выбор периода.ОК**. В этом макросе проверяется правильность задания значений **Начальная дата** и **Конечная дата**, и, в случае ошибки, выдаются соответствующие сообщения. Если даты заданы правильно, то открывается форма **Список товаров за период** (свойству формы **Открытие** назначен макрос **Выбор периода.Нет**, который проверяет, не является ли форма пустой). Эта форма основана на запросе

**Товары за период**, в котором для поля **Дата поставки** указано условие отбора:

Between [Начальная дата] and [Конечная дата]

Это условие нужно заменить на

Between Forms![Выбор периода]![Начальная дата] and  
Forms![Выбор периода]![Конечная дата]

Макрос **Выбор периода:**

Выбор периода : макрос				
	Имя макроса	Условие	Макрокоманда	Примечание
▶	OK	IsNull([Начальная	Сообщение	1
		...	ОстановитьМакрос	2
		[Начальная дата]	Сообщение	3
		...	ОстановитьМакрос	4
			ОткрытьФорму	5
	Нет	IsNull([Forms]![Сп	Сообщение	6
		...	Закреть	7
	начальная дата		ЗадатьЗначение	8
	конечная дата		ЗадатьЗначение	9
	Отмена		Закреть	10

**Макрокоманды макроса:**

**Выбор периода.ОК**

1) МК *Сообщение*

Сообщение: Диапазон не задан!

Условие выполнения МК:

IsNull([Начальная дата]) Or IsNull([Конечная дата])

2) МК *Остановить Макрос*

3) МК *Сообщение*

Сообщение: Неправильно задан диапазон!

Условие выполнения МК:

[Начальная дата]>[Конечная дата]

4) МК *ОстановитьМакрос*

5) МК *ОткрытьФорму*

Имя формы: Список товаров за период

Режим данных: Только чтение

**Выбор периода.Нет**

6) МК *Сообщение*

Сообщение: Нет товаров в заданный период!

Условие выполнения МК:  
IsNull(Forms![Список товаров за период]![Поставщик])

7) МК *Заккрыть*

Тип объекта: Форма  
Имя объекта: Список товаров за период

**Выбор периода.начальная дата**

8) МК *ЗадатьЗначение*

Элемент: [Начальная дата]  
Значение: [Календарь].[Value]

**Выбор периода.конечная дата**

9) МК *ЗадатьЗначение*

Элемент: [Конечная дата]  
Значение: [Календарь].[Value]

**Выбор периода.Отмена**

10) МК *Заккрыть*

Тип объекта: Форма  
Имя объекта: Выбор периода

---

**Пример 3**

Рассмотрим пример организации пользовательского командного меню, состоящего из следующих пунктов:

**Итоги для поставщика** — подсчет итоговых показателей по выбранному поставщику;

**Поставки на сумму** — вывод сведений о товарах, общая стоимость которых превышает заданную сумму;

**Поставки по кварталам** — вывод сведений о количестве поставок товаров по кварталам заданного года;

**Выход** — закрытие формы.

Создадим макрос с именем **меню Сведения о товарах**, содержащий команды *ДобавитьМеню* для каждого пункта. Затем создадим группы макросов для каждого пункта меню. Текст, указанный в столбце **Имя макроса**, будет выводиться в качестве пункта меню, а текст из столбца **Примечание** данной строки будет отображаться в строке состояния при активизации данного пункта меню. Если нужно разделить пункты

выпадающего меню горизонтальной линией, то следует ввести символ «минус» в столбце **Имя макроса** между соответствующими пунктами.

Рассматриваемое в примере пользовательское меню связано с формой **Меню\_фон**. Чтобы меню выводилось на экран при открытии формы, нужно задать свойству **Строка меню** значение **меню Сведения о товарах** – имя макроса, описывающего пункты меню. Данный макрос состоит из следующих команд:

	Макрокоманда	Примечание
	ДобавитьМеню	1
	ДобавитьМеню	2
	ДобавитьМеню	3
▶	ДобавитьМеню	4

### Макрокоманды макроса:

#### 1) МК *ДобавитьМеню*

Название меню: Итоги для поставщика

Имя макроса: меню Итоги для поставщика

#### 2) МК *ДобавитьМеню*

Название меню: Поставки на сумму

Имя макроса: меню Поставки на сумму

#### 3) МК *ДобавитьМеню*

Название меню: Поставки по кварталам

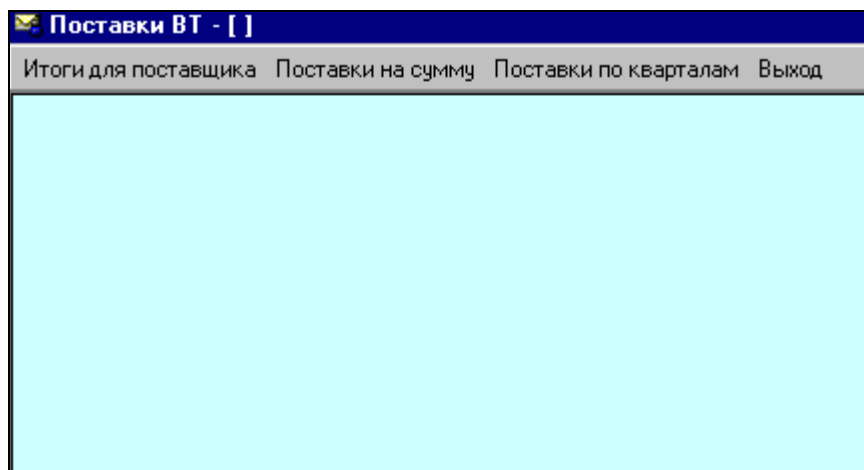
Имя макроса: меню Поставки по кварталам

#### 4) МК *ДобавитьМеню*

Название меню: Выход

Имя макроса: меню Выход

После открытия формы **Меню\_фон** экран будет иметь следующий вид:



При щелчке по пункту меню **Итоги для поставщика** будет выполняться макрос меню **Итоги для поставщика**, открывающий форму, в которой после выбора поставщика выведется количество выполненных поставок и их общая стоимость:

В данной форме свойство **Данные** для поля **Количество** имеет значение

**=DCount**("\*"; "[Товары]"; "[Код\_поставщика]=Forms![меню]![спПоставщик]")

а для поля **Стоимость** –

**=DSum**("[Товары].[Цена\_единицы]\*[Товары].[Количество\_единиц]";  
"[Товары]"; "[Код\_поставщика]=Forms![меню]![спПоставщик]")

#### 4. ЗАДАНИЕ НА КУРСОВУЮ РАБОТУ

Целью курсовой работы является усвоение теоретических основ и формирование практических навыков по созданию реляционной базы данных и разработке удобного приложения.

При создании БД нужно решить такие задачи:

- выделить объекты предметной области;
- спроектировать эффективную структуру таблиц, отображающих эти объекты;
- выполнить нормализацию таблиц (привести таблицы к третьей нормальной форме);
- описать таблицы, указав типы и свойства полей;
- создать связи между таблицами в виде схемы данных.

Разработка приложения начинается с создания необходимых запросов к БД, проектирования форм и отчетов. Далее созданные объекты нужно связать в единый комплекс. Для этого необходимо разработать систему иерархических меню, в которых будут отражены все функции приложения.

**В приложении должны быть реализованы следующие требования:**

- разработать формы для **всех** созданных запросов, включить эти формы в приложение;
- главное меню следует представить в виде кнопочной формы; для меню нижних уровней нужно использовать кнопочные формы, элементы управления **Группа** и макрокоманду **ДобавитьМеню** (обычно каждой форме или отчету ставится в соответствие пункт меню);
- разработать формы, подобные формам **Сведения о поставщике** (пример 1) и **Сведения о товарах** (пример 2);
- командные кнопки должны быть связаны с макросами; для создания кнопок, выполняющих закрытие формы или переход по записям, можно использовать Мастера;
- макросы, связанные с формой и ее элементами управления, представлять в виде *группы макросов*; использовать в макросах условия;
- значения параметров передавать в запросы **только через формы**; для указания значений текстовых параметров использовать элементы управления **Список** и **Поле со списком**; при задании дат пользоваться **Календарем**;
- использовать команду **ОткрытьФорму** с условием отбора записей;
- использовать МК **ЗадатьЗначение**, **КЭлементуУправления**, **Обновление**, **НайтиЗапись**, **ОстановитьМакрос**, **Сообщение**;
- обработать в приложении такие события, как **Получение фокуса**, **Потеря фокуса**, **До обновления**, **После обновления**, **Открытие**, **Закрытие**, **Нажатие кнопки** и другие.

## 5. СОДЕРЖАНИЕ ПОЯСНИТЕЛЬНОЙ ЗАПИСКИ

Пояснительная записка к курсовой работе (отчет) должна содержать следующие части:

- титульный лист,
- реферат,
- содержание,
- текст записки (отчета),
- приложения.

В приложении А данных методических указаний приведен образец *титульного листа* для пояснительной записки, в приложении Б - пример *реферата*.

Реферат предназначен для ознакомления с отчетом. Он должен быть кратким, информативным и содержать сведения, позволяющие принять решение о целесообразности прочтения всего документа. Реферат помещается непосредственно за титульным листом.

В реферате должны быть представлены сведения об объеме отчета, количестве частей, иллюстраций, таблиц, приложений, источников по перечню ссылок. В эти сведения должны быть включены также такие же сведения и по приложениям отчета. Далее приводится краткая информация о работе и перечень ключевых слов.

Информация в реферате излагается в такой последовательности:

- объект исследования или разработки;
- цель работы;
- используемые инструментальные и технические средства;
- результаты и их новизна;
- рекомендации по использованию результатов работы;
- область применения.

Реферат должен разместиться на одной странице формата А4, поэтому его объем не должен превышать 500 слов.

Ключевые слова помещаются после текста реферата (через две пустые строки). Перечень ключевых слов включает от 5 до 15 слов (словосочетаний), печатаются они прописными буквами в именительном падеже через запятые. Перенос ключевых слов не разрешается. Ключевые слова должны характеризовать объект разработки и используемый инструментарий.

Пример оформления «Содержания» пояснительной записки (без указания страниц) приведен в приложении В.

Текст пояснительной записки начинается с «Введения». В этом пункте можно рассказать об автоматизации обработки больших объемов информации с помощью СУБД, изложить кратко историю развития баз данных, рассмотреть состояние этого вопроса в настоящее время.

В разделе «Проблемы автоматизации...» нужно показать актуальность проблемы автоматизации обработки данных для конкретной предметной области, указать недостатки «ручной» обработки данных, рассмотреть возможности прикладной системы при работе с информацией, содержащейся в базе данных.

В разделе «Разработка системы...» необходимо рассказать о назначении системы, указать, какие функции она выполняет, дать определение концептуальной модели данных, кратко охарактеризовать реляционную МД, описать объекты рассматриваемой предметной области, показать представление их таблицами. Нужно привести определение базовых таблиц и таблиц-справочников, описать связи между таблицами, показать схему данных.

При описании запросов, реализованных в системе, нужно привести словесную формулировку запроса, определить, к какому типу запросов он относится, указать QBE – бланк или инструкцию SQL - запроса. Необходимо также показать индекс, используемый для запроса.

Завершается данный раздел описанием структуры приложения. Нужно показать иерархию частей приложения, описать главную кнопочную форму и формы нижних уровней, привести примеры макросов, автоматизирующих работу приложения, описать события, с которыми макросы связаны.

Далее на отдельной странице помещаются «Выводы». В «Выводах» нужно кратко описать результаты выполненной работы, показать области ее применения, дать оценку разработанной системе, указать недостатки и пути их устранения. Требуется также перечислить технические средства и программное обеспечение, необходимое для функционирования системы.

Список источников (книги, журналы), которые используются при разработке системы, оформляется на отдельной странице под заголовком «Перечень ссылок» и располагается за «Выводами». В соответствующих местах текста отчета должны быть ссылки на эти источники. Последовательность описания в «Перечне» должна соответствовать порядку, в котором источники впервые упоминаются в тексте.

Библиографические описания источников в «Перечне» выполняются в соответствии с действующими стандартами по библиотечному и издательскому делу. Например,

1. С. Робинсон. Microsoft Access 2003: учебный курс – СПб: издательство «Питер», 2005. – 512 с.:ил.

Ссылки в тексте отчета на источники нужно указывать порядковыми номерами, выделенными квадратными скобками.

Завершают пояснительную записку «Приложения». В «Приложения» помещаются:

- тексты разработанных SQL - запросов;



- основные формы, формы-результаты, отчеты;
- руководство пользователя;
- задание на курсовую работу (образец оформления задания приведен в приложении Д);

В «Руководстве пользователя» должен быть подробно описан порядок действий пользователя по всем пунктам меню приложения.

## **6. ПРАВИЛА ОФОРМЛЕНИЯ ПОЯСНИТЕЛЬНОЙ ЗАПИСКИ**

Пояснительная записка (отчет) оформляется в соответствии с государственным стандартом Украины «Документация. Отчеты в сфере науки и техники. Структура и правила оформления» ДСТУ 3008-95.

Должны использоваться листы формата А4 (печать на одной стороне листа). Ширина полей страницы: верхнее и нижнее – 20 мм, левое – 30 мм, правое – не менее 10 мм.

Плотность строк на странице не должна превышать 40 (шрифт 14, межстрочный интервал 1,5). Выравнивание текста выполняется по ширине страницы. Не допускается выделение текста с помощью курсива, подчеркивания, полужирного начертания. Номер страницы указывается арабскими цифрами в правом верхнем углу (без точки). Нумерация страниц должна быть сквозной по всему отчету, начиная с титульного листа, но первый номер ставится на «ВВЕДЕНИИ» (обычно, это страница 4).

Текст отчета разбивается на разделы, подразделы, пункты и подпункты. Разделы и подразделы должны иметь заголовки, пункты и подпункты могут их иметь. Каждый раздел должен начинаться с новой страницы (нумерация разделов: 1, 2, 3, ...). Его заголовок пишется прописными (заглавными) буквами. Номер указывается без точки и отделяется от заголовка одним пробелом. Точка в конце заголовка не ставится (это относится к заголовкам всех частей отчета). Текст заголовка размещается по центру страницы.

Подразделы нумеруются в пределах раздела, используется двухуровневая нумерация. Заголовки указываются строчными буквами (первая буква – заглавная) и располагаются с отступом на величину абзаца (1,27 см или, примерно, 5 символов). Величина абзаца должна быть одинаковой по всему тексту.

Заголовки разделов и подразделов отделяются от предыдущего и последующего текста пустыми строками. Если заголовок состоит из двух или более предложений, их разделяют точкой. Переносы слов в заголовке не допускаются. Не допускается также размещение заголовка в нижней части страницы, если после него расположена только одна строка текста.

Для указания пунктов используется трехуровневая нумерация, подпунктов – четырехуровневая. После их заголовков пустая строка не ставится.

Внутри разделов (подразделов, пунктов, подпунктов) могут быть приведены перечисления. Перед перечислением ставится двоеточие. Перед каждой позицией перечисления ставится дефис или строчная буква со скобкой. Перечисления печатаются строчными буквами с абзацного отступа.

Рисунки и таблицы нумеруются в пределах раздела (используется двухуровневая нумерация). Они размещаются по ходу текста. Большой рисунок (таблицу) можно поместить на отдельной (следующей) странице. Подпись указывается под рисунком по центру страницы. Например,

Рисунок 2.1 – Схема данных

Таблица размещается с отступом на величину абзаца. Подпись ставится над таблицей. Например,

Таблица 2.1 Поля таблицы ТОВАРЫ


Подпись таблицы (рисунка) можно не указывать, если она очевидна из текста.

Заголовки граф таблицы печатаются с прописных букв, а подзаголовки – со строчных, если они представляют одно предложение с заголовком. Подзаголовки, имеющие самостоятельное значение, пишутся с прописной буквы. В конце заголовков и подзаголовков точка не ставится.

Материал, не включенный в основную часть отчета, размещается в приложениях (например, рисунки, схемы, таблицы, экранные формы, программы, инструкции, техническое задание). Приложения должны иметь общую с основной частью отчета сквозную нумерацию страниц. Каждое приложение должно иметь номер. Для нумерации используются буквы украинского алфавита: А, Б, В, Д, Е, ... (буквы Г, З исключаются).

При размещении в приложениях рисунков (таблиц) действуют следующие правила.

Если в приложении должен находиться только один рисунок, то вверху страницы по центру располагаются две строки (заголовков), например,

Приложение А  
Схема данных

Под рисунком нужно указать подпись (название). Если название рисунка соответствует названию приложения, то его можно опустить. Например,

**Рисунок А.1**

Если приложение занимает несколько страниц, то заголовок указывается только на первой из них и под каждым рисунком ставится подпись. Например,

**Рисунок Б.3 Результаты поквартального отчета**

В случае, когда рисунок не помещается по ширине листа, его можно развернуть на 90 градусов против часовой стрелки. Заголовок приложения при этом должен размещаться на отдельной странице (по ее центру).

При ссылках на разделы, подразделы, приложения нужно указать их номера. Например,

«... в разделе 2 ...» или «... смотри 2.1 ...»,  
«... на рис. 1.3 ...» или «... на рисунке 1.3 ...»,  
«... в таблице 1.2 ...» или «... см. табл. 3.2 ...»,  
«... в приложении Б ...».

## **ЛИТЕРАТУРА**

1. Дейт К. Введение в системы баз данных. К.; М; СПб.: Издательский дом «Вильямс», 2000.-848 с.: ил.
2. Т.С. Карпова. Базы данных: модели, разработка, реализация – СПб.: Питер, 2001. – 304 с.: ил.
3. С. Робинсон. Microsoft Access 2003: учебный курс – СПб: издательство «Питер», 2005. – 512 с.: ил.
4. Д. Вейскас. Эффективная работа с Microsoft Access 2003 СПб.: Питер, 2004. –850 с.: ил.
5. Керри Н. Праг, Майкл Р. Ирвин. Access 2003. Библия пользователя: Пер. с англ. М.: Издательский дом “Вильямс”, 2005.- 1040 с.: ил
6. Михеева В.Д., Харитоновна И.А. Microsoft Access 2002.-СПб.: БХВ-Петербург, 2002.-1040 с.: ил.

## **Приложение А**

*(Образец титульного листа)*

**ДОНЕЦКИЙ НАЦИОНАЛЬНЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**  
**Кафедра прикладной математики и информатики**

**ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**  
**к курсовой работе по курсу**  
**“ Базы данных ”**

**Тема работы:**  
**«Поставки вычислительной техники»**

Руководители: \_\_\_\_\_ Н.С. Костюкова  
\_\_\_\_\_ А.В. Чернышова  
\_\_\_\_\_ Г.Г. Шалдырван  
(подпись) (дата)

Разработал: \_\_\_\_\_ В.И.Иванов  
ст. гр. ПС-10а (подпись) (дата)

**Донецк - 2012**

**Приложение Б**  
**( Пример реферата )**

## **РЕФЕРАТ**

Пояснительная записка к курсовой работе: 25 стр., 4 прил., 2 библ., 9 ил., 1 табл..

Работа посвящена разработке прикладной системы, предназначенной для автоматизации учета деятельности посреднической фирмы по поставкам вычислительной техники. Целью автоматизации является упрощение и повышение надёжности обработки и хранения информации.

В прикладной системе реализованы функции ввода, корректировки и просмотра данных о фирмах, товарах, поставках товаров. Реализованы запросы, позволяющие получать разнообразную информацию по базе данных. В качестве инструментария использована СУБД Microsoft Access 2003.

ФИРМА, ТОВАР, ПОСТАВКИ ТОВАРОВ, БАЗА ДАННЫХ,  
РЕЛЯЦИОННАЯ МОДЕЛЬ ДАННЫХ, СИСТЕМА УПРАВЛЕНИЯ  
БАЗАМИ ДАННЫХ, СУБД MICROSOFT ACCESS

**Приложение В**  
**( Пример «Содержания» пояснительной записки )**  
**СОДЕРЖАНИЕ**

Введение .....	
1 Проблемы автоматизации посреднической деятельности .....	
2 Разработка системы "Поставки вычислительной техники" .....	
2.1 Назначение и функции системы .....	
2.2 Проектирование концептуальной модели данных .....	
2.3 Формирование запросов .....	
2.3.1 Сведения о поставщиках .....	
2.3.2 Поставка товаров .....	
2.3.3 Итоговые показатели .....	
2.4 Структура системы .....	
Выводы .....	
Перечень ссылок .....	
Приложение А SQL-запросы .....	
Приложение Б Примеры форм .....	
Приложение В Примеры отчетов .....	
Приложение Д Руководство пользователя .....	
Приложение Е Задание на курсовую работу .....	

**Приложение Д**

*( Образец оформления задания на курсовую работу )*

**Донецкий национальный технический университет**

Кафедра **ПМИ**

Дисциплина **Базы данных**

Направление подготовки **Программная инженерия**

Курс **2**      Группа **ПС-10а**      Семестр **4**

**ЗАДАНИЕ**

на курсовую работу студенту

**Иванову Виктору Игоревичу**

1. ТЕМА РАБОТЫ: Разработка информационной системы для автоматизации деятельности посреднической фирмы по поставкам вычислительной техники.

2. . ДАТА ВЫДАЧИ ЗАДАНИЯ: 02.02.2012

3. ЗАДАНИЕ:

3.1. Создать базу данных и разработать приложение для следующей предметной области:

Для автоматизации работы посреднической фирмы, занимающейся продажей вычислительной техники, необходима информация о фирмах, поставляющих вычислительную технику (название фирмы, страна, город, адрес, телефон, форма собственности) и о представленных ими товарах (название товара, фирма- изготовитель, цена за единицу, количество единиц, дата поставки).

3.2. По БД реализовать следующие виды запросов: итоговые, перекрестные, внешние объединения, запрос на запросе, запросы на изменение данных.

3.3. Для запросов разработать формы и связать их с помощью макросов, создав автоматизированное приложение.

#### 4. ТРЕБОВАНИЯ К ХАРАКТЕРИСТИКАМ ИСПОЛЬЗУЕМЫХ ИНСТРУМЕНТАЛЬНЫХ СРЕДСТВ:

База данных должна быть разработана в среде СУБД Microsoft Access 2003.

#### 5. ТРЕБОВАНИЯ К ПАРАМЕТРАМ ИСПОЛЬЗУЕМЫХ ТЕХНИЧЕСКИХ СРЕДСТВ:

Для функционирования системы необходим компьютер IBM PC с процессором Intel Pentium и ОЗУ объемом не менее 512Mb, операционная система Microsoft WindowsXP Professional, пакет Office 2003.

6. СРОК СДАЧИ: 10.05.2012

Студент \_\_\_\_\_ ( В.И.Иванов)

Руководители работы \_\_\_\_\_ (Н.С.Костюкова)

\_\_\_\_\_ (А.В.Чернышова)

\_\_\_\_\_ (Г.Г.Шалдырван)



## СОДЕРЖАНИЕ

Введение . . . . .	3
1. Основные этапы разработки приложения . . . . .	4
2. Разработка отчетов . . . . .	5
3. Автоматизация приложения с помощью макросов . . . . .	20
2.1 Синтаксис ссылок . . . . .	20
2.2 Примеры использования макросов. . . . .	21
4. Задание на курсовую работу . . . . .	53
5. Содержание пояснительной записки . . . . .	54
6. Правила оформления пояснительной записки . . . . .	56
Литература . . . . .	58
Приложение А. Образец титульного листа . . . . .	59
Приложение Б. Пример реферата . . . . .	60
Приложение В. Пример «Содержания» пояснительной записки. . . . .	61
Приложение Д. Образец оформления задания на курсовую работу . . . . .	62

## МЕТОДИЧЕСКИЕ УКАЗАНИЯ И ЗАДАНИЯ к курсовой работе по курсу « БАЗЫ ДАННЫХ»

( направление подготовки 6.050103 ”Программная инженерия”)

Составители:                      Галина Георгиевна Шалдырван  
   Наталья Стефановна Костюкова