

## **Методические рекомендации по выполнению практической части курсовой работы "Создание базы данных «Магазин одежды»"**

Настоящие методические рекомендации предназначены для студентов государственного профессионального образовательного учреждения «Печорский промышленно-экономический техникум» очной формы обучения специальности *09.02.07 Информационные системы и программирование*.

Требуется спроектировать локальную однопользовательскую базу данных "Магазин одежды". Идеей для создания и прототипом базы данных послужил сайт компании Lamoda Group – ведущей онлайн-платформы в сфере моды и lifestyle России и СНГ (<https://www.lamoda.ru/>). Готовая база данных «Магазин одежды» может выступать в свою очередь в роли прообраза базы данных любого другого магазина. Структуру базы можно переделать под базу данных магазина косметики, продуктов, бытовой или компьютерной техники, мебели и др.. Спроектированная база данных является законченным продуктом готовым для эксплуатации, но не готовым проектом. В конце методических рекомендаций приведены возможные пути дальнейшего развития и модернизации базы данных.

Данный проект выполнен в рамках курсовой работы по дисциплине *ОП.08 Основы проектирования баз данных* студентами специальности *09.02.07 Информационные системы и программирование*. В качестве примера рассматриваются скриншоты работы студентки второго курса группы И-203 Вероники М.

**Среда разработки:** система управления базами данных (СУБД) MS Access 2010 (допускается версия MS Access не ниже 2007).

**Цель работы:** создание локальной однопользовательской базы данных "Магазин одежды".

### **Задачи работы:**

- совершенствовать коммуникативно-познавательные умения, направленные на систематизацию и углубление знаний;
- систематизировать, закрепить и расширить теоретические знания о базах данных на основе реляционной модели;
- приобрести практические навыки при решении конкретных задач разработки объектов автоматизации;
- приобрести опыт проектирования реляционных баз данных. Выполнить переход от составления простых баз данных к проектной деятельности;
- разработать объекты проектируемой базы данных: таблицы, запросы, формы, отчеты;
- спроектировать локальную однопользовательскую базу данных "Магазин одежды".

### **Для работы над проектом студент:**

**должен знать:**

- основные объекты баз данных;
- типы связей в реляционных базах данных;
- теорию нормальных форм;
- основы программирования в среде VBA.

*Должен уметь:*

- демонстрировать первичные навыки работы в СУБД MS Access;
- создавать обработчики событий, писать простой код обработчика;
- работать в любом графическом (растровом) редакторе.

***По итогам выполнения проекта студент:***

*должен знать:*

- приведение базы данных к нормальным формам;
- назначение индексов, индексирование таблиц;
- виды представления форм;
- связывание подчиненных форм.

*Должен уметь:*

- создавать таблицы, ключи, индексы, схемы данных;
- создавать справочники;
- создавать формы;
- писать простой кодирование форм.

*Иметь практический опыт:*

- моделирования предметной области проектируемой системы;
- анализа структуры баз данных информационных систем;
- проектирования простых информационных систем.

***Этапы работы:***

1. Создание схемы данных.
2. Создание пользовательских форм.
3. Создание выпадающего списка навигации формы *Основная*.
4. Создание инструмента фильтрации записей.
5. Создание отчетов.
6. Заключение.

***Внимание:*** Некоторые решения при проектировании базы данных не являются оптимальными и выполнены исключительно в учебных целях с целью обеспечения наглядности и максимального соответствия учебному плану.

***Ход работы:***

## 1. Создание схемы данных

Схема данных должна содержать следующие таблицы, описывающие объекты предметной области разрабатываемой базы данных:

Таблица **Категория** содержит атрибуты:

Имя поля	Тип данных	Индекс
КодКатегория	Счетчик	PrimaryKey
Категория	Текстовый	
Примечание	Текстовый	

Таблица **Наименование** содержит атрибуты:

Имя поля	Тип данных	Индекс
КодНаименование	Счетчик	PrimaryKey
КодКатегория	Числовой	ForeignKey
Наименование	Текстовый	
ЦелеваяАудитория	Текстовый	
Примечание	Текстовый	

Таблица **Вид** содержит атрибуты:

Имя поля	Тип данных	Индекс
КодВид	Счетчик	PrimaryKey
КодНаименование	Числовой	ForeignKey
Вид	Текстовый	
Примечание	Текстовый	

Таблица **Изделие** содержит атрибуты:

Имя поля	Тип данных	Индекс
КодИзделие	Счетчик	PrimaryKey
КодВид	Числовой	ForeignKey
Наименование	Текстовый	
Материал	Текстовый	
Цвет	Текстовый	
Размер	Текстовый	
Бренд	Текстовый	
Застежка	Текстовый	
Стиль	Текстовый	
Фото	Поле объекта OLE	
Цена	Денежный	
Скидка	Числовой	
Корзина	Логический	
Примечание	Текстовый	

Таблица **Бренд** содержит атрибуты:

Имя поля	Тип данных	Индекс
Бренд	Текстовый	PrimaryKey

Создадим файл базы данных **Магазин одежды.acscdb**. В созданном файле создадим таблицы проектируемой базы данных.

Для поля **ЦелеваяАудитория** таблицы **Наименование** выполнить следующие настройки:

- Тип элемента управления – *Список*;
- Тип источника строк – *Список значений*;

- Источник строк – *Мужчинам;Женщинам; Подросткам;Детям.*

Для поля *Скидка* таблицы **Изделие** выполнить следующие настройки:

- Тип элемента управления – *Список;*
- Тип источника строк – *Список значений;*
- Источник строк – *5;10;15;20;25;30;35;40;50.*

Таблица **Бренд** является справочником для поля *Бренд* таблицы **Изделие**. Для создания выпадающего списка значений справочника полю *Бренд* таблицы **Изделие** присвоить следующие свойства:

- Тип элемента управления - *Поле со списком;*
- Тип источника строк - *Таблица или запрос;*
- Источник строк - *Бренд.*

Построить схему данных. Создать связи между таблицами (рис.1).

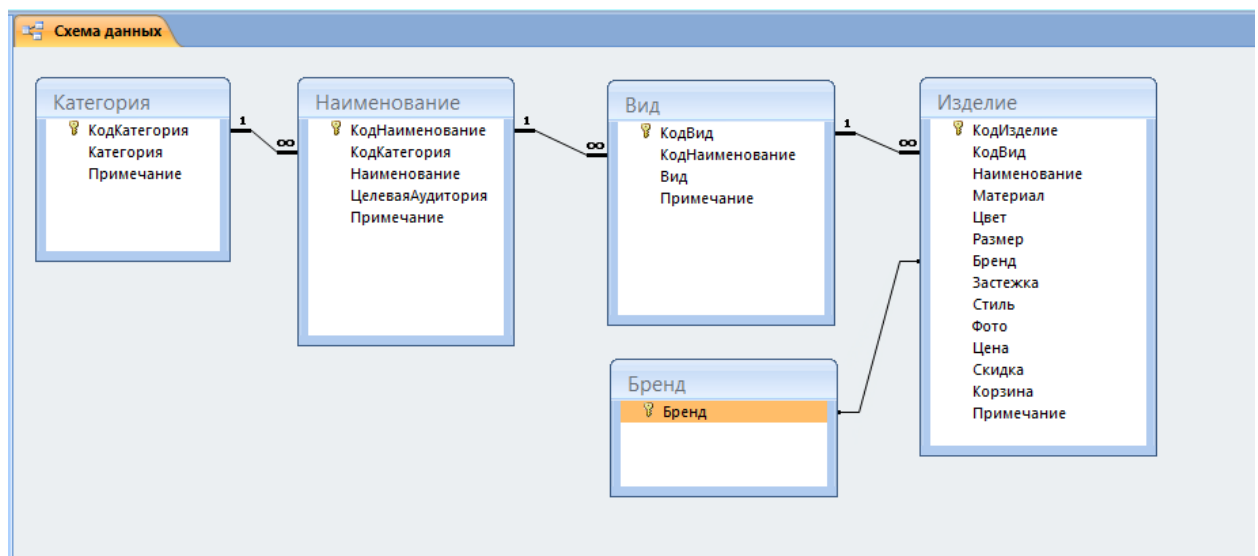


Рисунок 1. Схема данных

Анализ схемы данных подтверждает, база данных нормализована и соответствует третьей нормальной форме (3НФ):

- все таблицы содержат первичный ключ;
- отсутствуют составные первичные ключи (основные таблицы содержат суррогатные первичные ключи с типом данных – счетчик);
- данные в таблицах атомарны;
- в таблицах отсутствуют транзитивные зависимости.

## 2. Создание пользовательских форм

Под пользовательскими формами будем понимать формы нашего проекта:

- **Основная** - стартовая форма, с источником записей – таблица **Категория**. Содержит две подчиненные формы:

- подчиненная форма с источником записей – форма **Наименование подчиненная форма**,
- подчиненная форма с источником записей – форма **Вид подчиненная форма**.
- **Изделие** – форма, хранящая список товаров выбранного вида, источник записей – запрос **ЗапросИзделиеФорма**.
- **Бренд** – форма-справочник, источник записей – таблица **Бренд**.

### 2.1. Создание формы «Вид подчиненная форма»

Для таблицы **Вид** с помощью инструмента *Мастер форм* создать форму **Вид подчиненная форма**. Выбрать все доступные поля (рис.2). Внешний вид формы – *табличный* (рис.3).

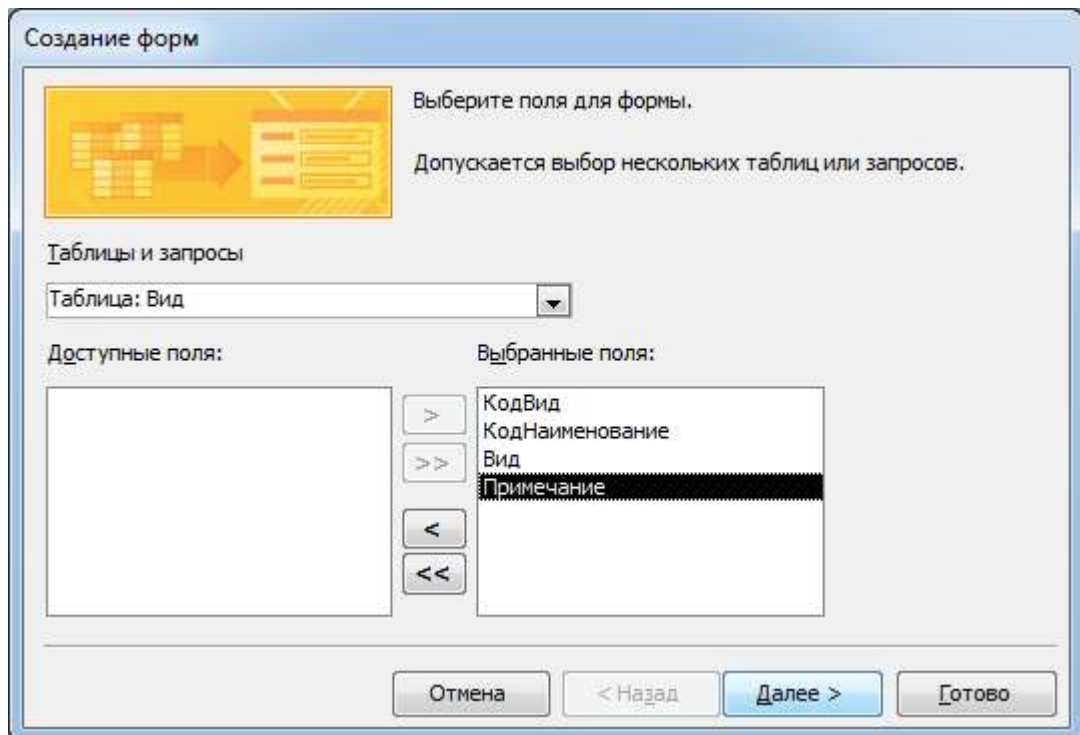


Рисунок 2. Выбор доступных полей для формы

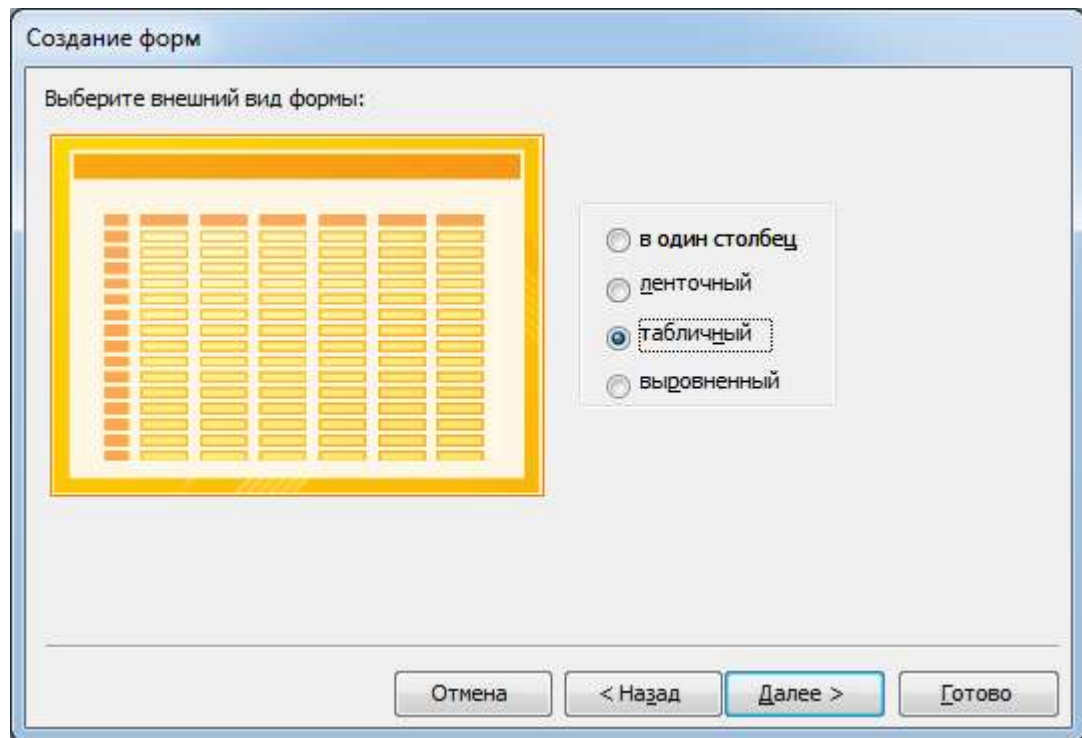


Рисунок 3. Выбор внешнего вида формы

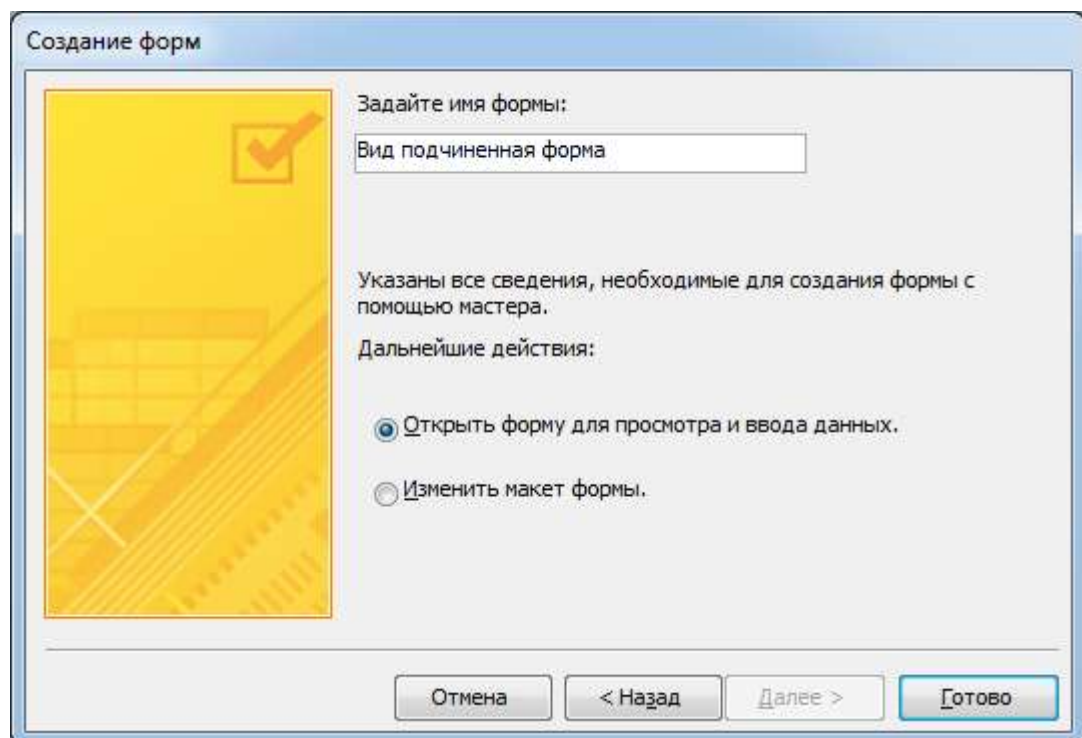


Рисунок 4. Задание имени формы

## 2.2. Создание формы «Основная»

Для работы с проектируемой базой данных создадим пользовательскую форму **Основная**. Для этого выполним следующий алгоритм действий:

- с помощью мастера создадим форму для таблиц **Категория** и **Наименование** с видом представления *Главная-Подчиненная*;
- в созданную форму в качестве второй подчиненной вставим имеющуюся форму **Вид подчиненная форма**, свяжем подчиненные формы;
- отформатируем форму.

### 2.2.1. Создание формы «Категория – Наименование»

Используя инструмент *Мастер форм*, создать форму для таблиц **Категория** и **Наименование**. Вид представления данных: *Подчиненные формы* (рис.7). При создании задать следующие имена: главная форма - *Основная*; подчиненная форма - *Наименование подчиненная форма*.

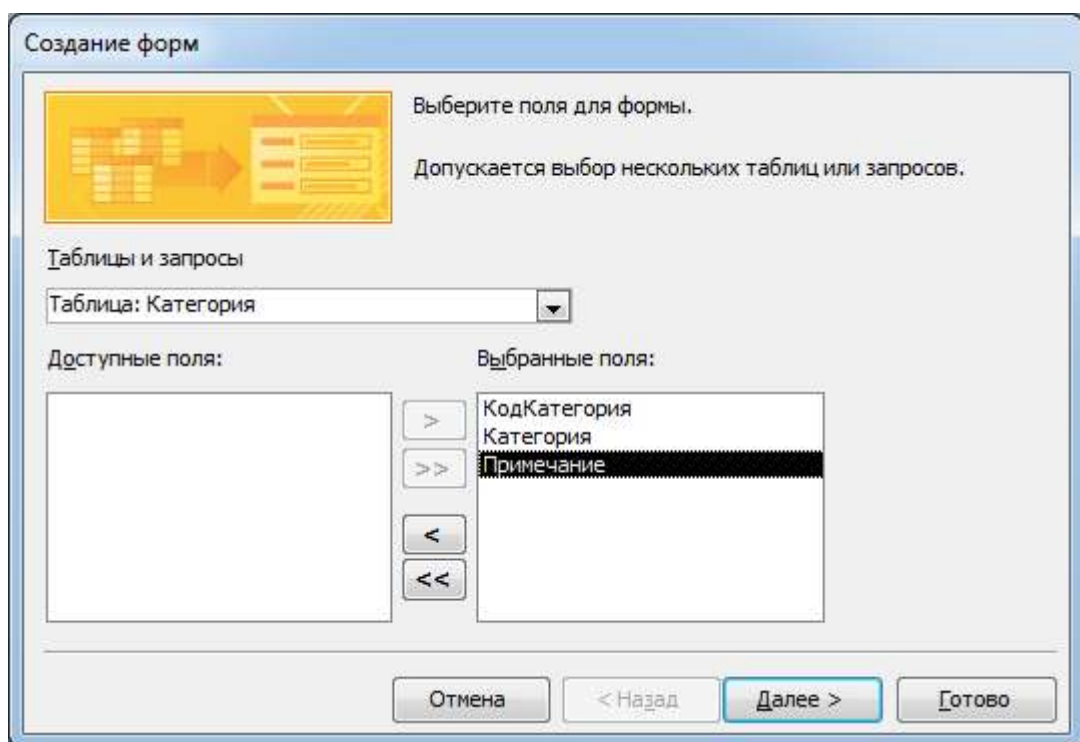


Рисунок 5. Выбор полей таблицы *Категория*

**Создание форм**

Выберите поля для формы.  
Допускается выбор нескольких таблиц или запросов.

Таблицы и запросы  
Таблица: Наименование

Доступные поля:

Выбранные поля:

- Категория.КодКатегория
- Категория
- Категория.Примечание
- КодНаименование
- Наименование.КодКатегория
- Наименование
- ЦелеваяАудитория
- Наименование.Примечание

Отмена < Назад **Далее >** Готово

Рисунок 6. Выбор полей таблицы *Наименование*

**Создание форм**

Выберите вид представления данных:

- Категория  
- Наименование

Категория\_КодКатегория, Категория,  
Категория\_Примечание

КодНаименование,  
Наименование\_КодКатегория,  
Наименование, ЦелеваяАудитория,  
Наименование\_Примечание

☒ Подчиненные формы ☐ Связанные формы

Отмена < Назад **Далее >** Готово

Рисунок 7. Выбор вида представления данных



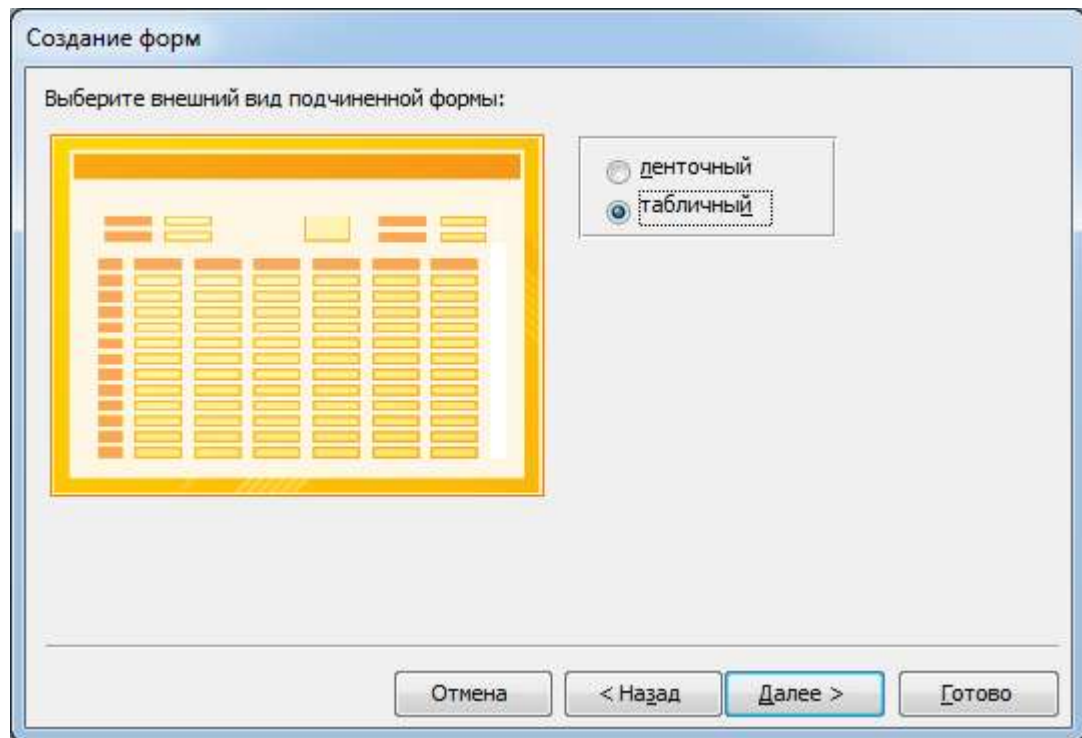


Рисунок 8. Выбор внешнего вида подчиненной формы

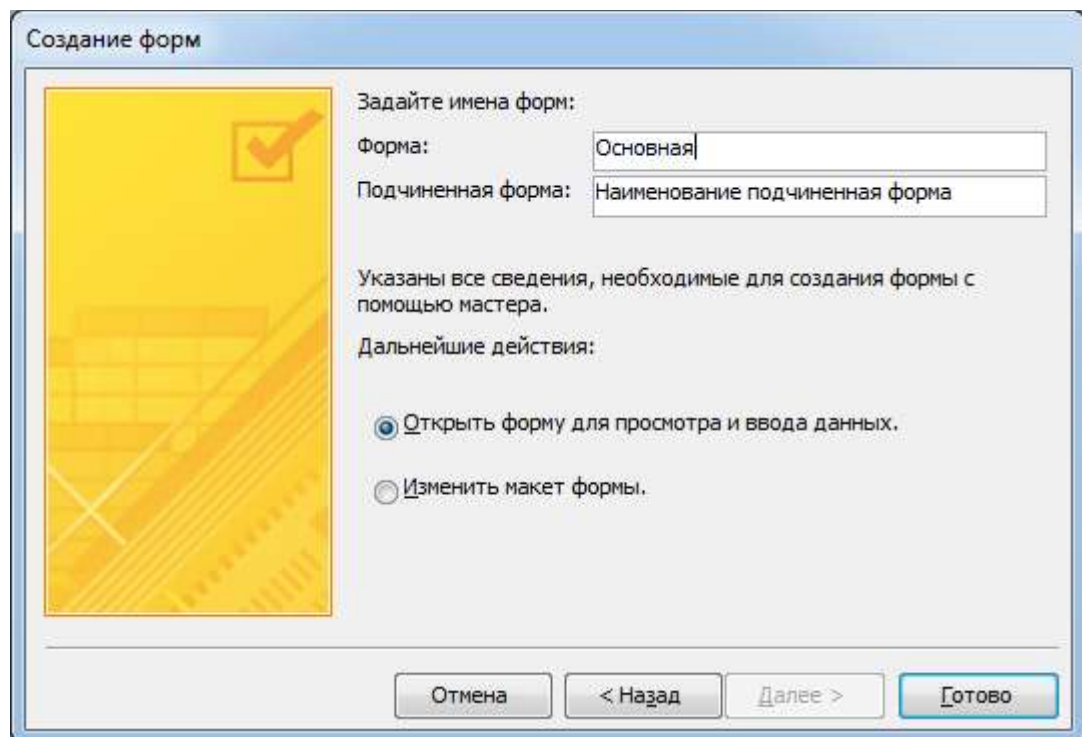


Рисунок 9. Задание имен форм

Рисунок 10. Примерный вид формы *Основная*

### 2.2.2. Встраиваем форму «Вид подчиненная форма» в качестве второй подчиненной в форму «Основная»

Поместить созданную форму *Вид подчиненная форма* в качестве второй подчиненной в форму **Основная**. Для этого в режиме *Конструктор* формы **Основная** вставить объект *Подчиненная форма, отчет*, при этом режим использования мастера должен быть включен (рис 11). Используя мастер подчиненных форм указать наименование формы загружаемой в объект *Подчиненная форма, отчет* (рис. 12).

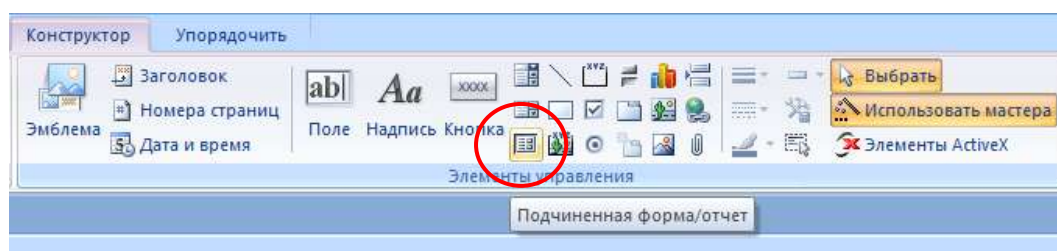


Рисунок 11. Встраивание подчиненной формы

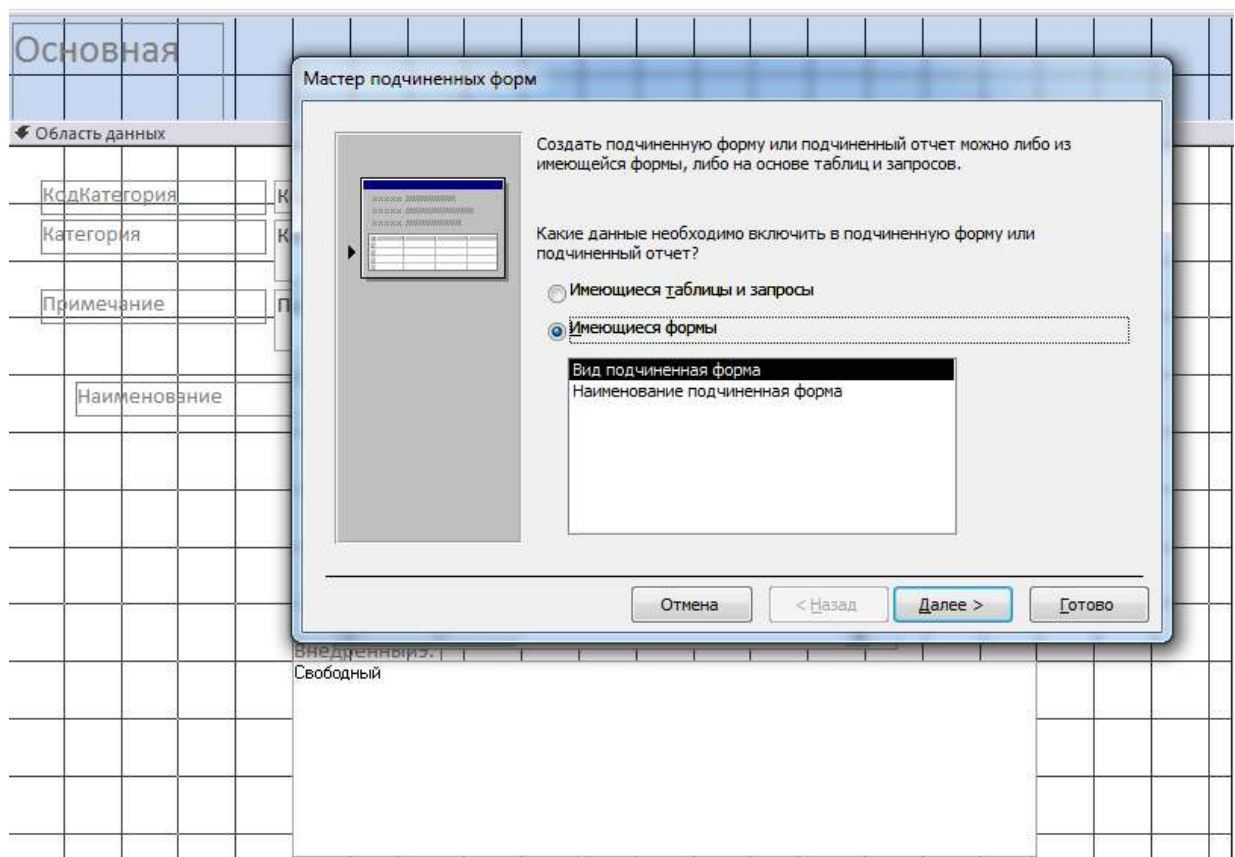


Рисунок 12. Определение встраиваемой подчиненной формы

Таблицы **Категория** и **Вид** не имеют общих полей для связи (рис.1), поэтому выбираем вариант – *Отсутствуют* и нажимаем кнопку **Далее** (рис.13).

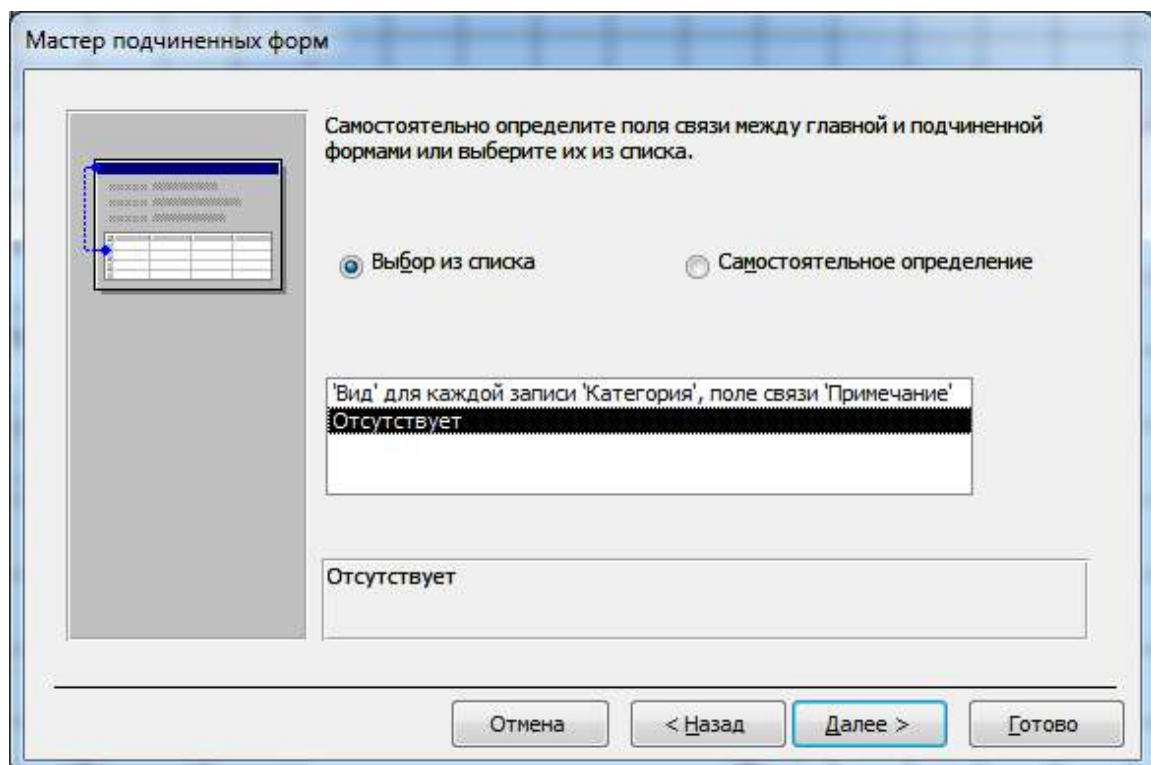


Рисунок 13. Выбор полей связи

Вводим название для подчиненной формы и нажимаем кнопку **Готово** (рис. 14).

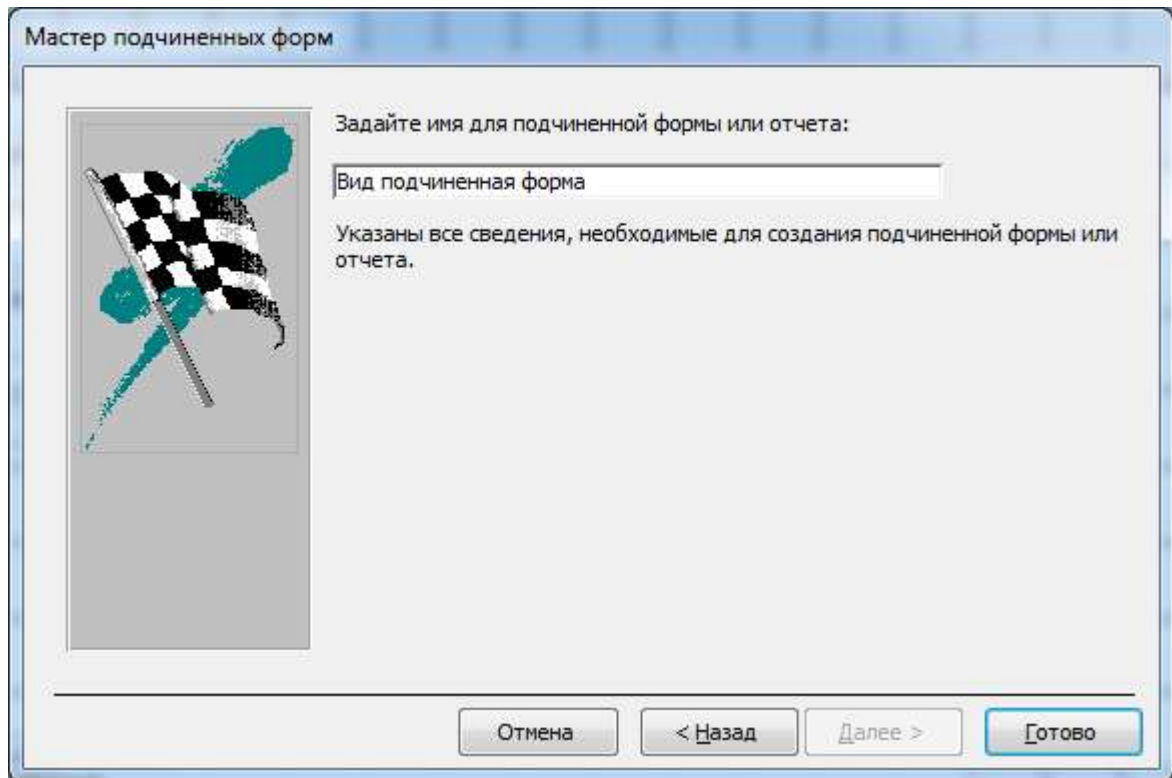


Рисунок 14. Имя подчиненной формы

В режиме *Конструктор* форматируем объекты пользовательского интерфейса созданной формы. Объекту форма устанавливаем следующие свойства:

- Область выделения (вкладка *Макет*) – *Нет*;
- Полосы прокрутки (вкладка *Макет*) – *Только по вертикали*;
- Всплывающее окно (вкладка *Другие*) – *Да*.

Получаем предварительный результат формы *Основная*, приведенный на рисунке 15.

Рисунок 15. Примерный вид формы *Основная* с двумя подчиненными формами

Для обеспечения связи между двумя подчиненными формами *Наименование - Вид*, необходимо создать дополнительное текстовое поле. Таким образом, две подчиненные формы будут связаны между собой через текстовое поле.

**Шаг1.** Связываем подчиненную форму *Наименование подчиненная форма* с текстовым полем *txtLinkField*. В режиме *Конструктор* в любом свободном месте создаем текстовое поле, назовем его *txtLinkField*. В созданном текстовом поле, в свойстве *Данные*, пишем ссылку на связующее поле таблицы *Наименование*: *=[Наименование подчиненная форма]![КодНаименование]* (рис. 16).

**Шаг 2.** Связываем текстовое поле *txtLinkField* с подчиненной формой *Вид подчиненная форма*. Выделим вторую подчиненную форму *Вид подчиненная форма*. В окне свойств подчиненной формы на вкладке *Данные* укажем следующие настройки (рис. 17):

- Основные поля – *txtLinkField*;
- Подчиненные поля – *КодНаименование*.

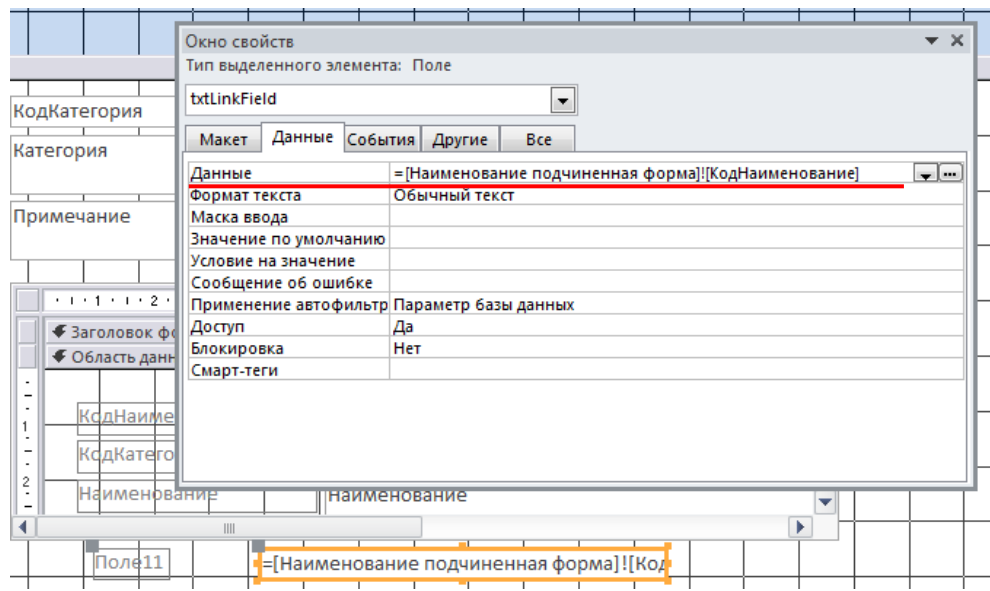


Рисунок 16. Создание текстового поля связи подчиненных форм

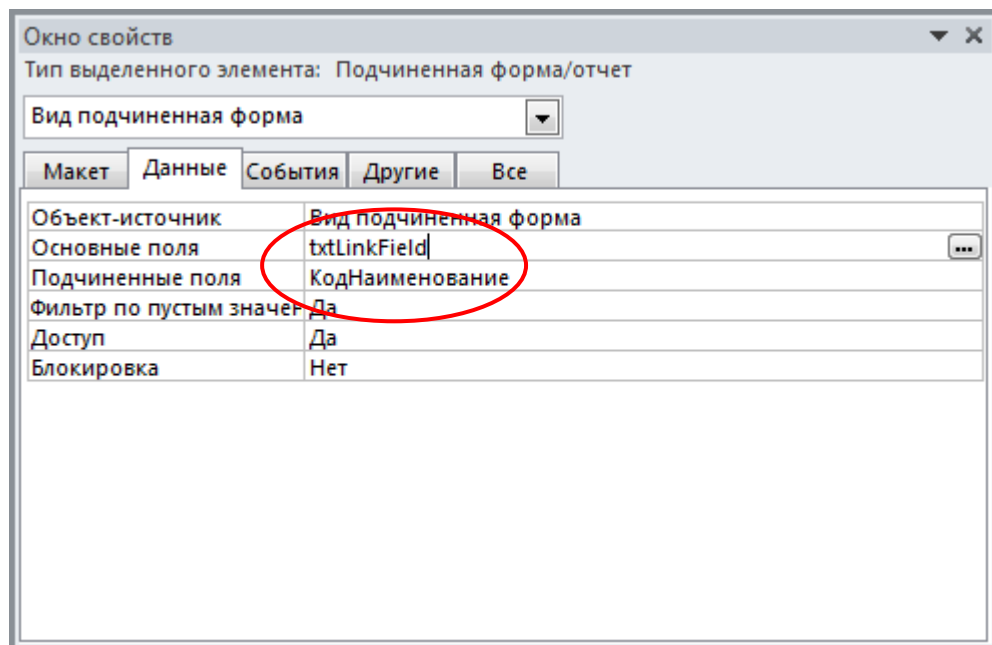


Рисунок 17. Настройки подчиненной формы Вид подчиненная форма

После того, как подчиненные формы связаны можно скрыть текстовое поле связи *txtLinkField*, для этого свойству текстового поля *Вывод на экран* (вкладка *Макет*) установим значение – *Нет*.

Используя данные сайта *Lamoda* и созданную форму **Основная** заполните таблицы базы данных. Количество записей внесенных в таблицы:

- Категория: 2 – 3;
- Наименование: 4 – 6 (на каждую категорию);
- Вид: 6 – 8 (на каждое наименование).

Переходя по записям формы **Основная** проверить ее работоспособность. Смена текущей записи **Категория** приводит к изменениям записей подчиненных форм **Наименование** и **Вид** (рис.18).

The screenshot shows a software interface titled 'Основная'. It contains two main data tables. The first table, 'Наименование', has columns for 'КодНаименов', 'КодКатегори', 'Наименование', and 'ЦелеваяАудит'. The second table, 'Вид', has columns for 'КодВид', 'КодНаимено', 'Вид', and 'Примечани'. Both tables are populated with data related to clothing items.

КодНаименов	КодКатегори	Наименование	ЦелеваяАудит
1	1	Брюки	Мужчинам
2	1	Верхняя одежда	Мужчинам
3	1	Джемперы, свитеры и кардига	Женщинам
4	1	Джинсы	Мужчинам
5	1	Домашняя одежда	Женщинам
6	1	Майки	Женщинам
7	1	Нижнее белье	Женщинам

КодВид	КодНаимено	Вид	Примечани
1	1	Прямые брюки	
2	1	Спортивные брюки	
3	1	Тайтсы	
4	1	Утепленные брюки	
5	1	Горнолыжные брюки	
6	1	Зауженные брюки	

Рисунок 18. Вид формы **Основная** с заполненными таблицами

### 2.3. Создание формы «Изделие»

#### 2.3.1. Создание источника записей для формы «Изделие»

При открытии формы **Изделие**, должны открываться записи текущего **Вида**. Для этого форма **Изделие** в качестве источника записей должна использовать не таблицу **Изделие**, а созданный на основе таблицы запрос.

Запрос создаем в режиме **Конструктор запросов**. При этом удобно, чтобы форма **Основная** была открыта.

Двойной клик по полю таблицы добавит это поле в список выводимых полей. Добавить все поля таблицы (рис.19).



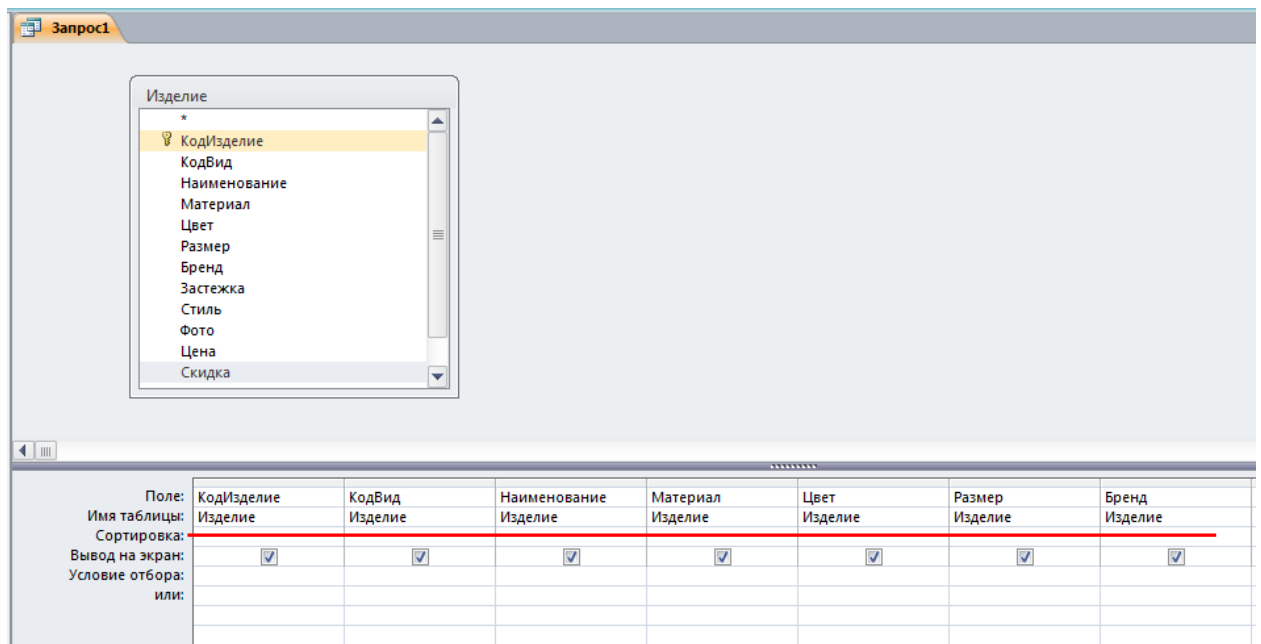


Рисунок 19. Конструктор создания запроса

Установить курсор в строку *Условие отбора* поля *КодИзделие*. С помощью мастера **Построитель** (рис.20) создать ссылку на поле *КодВид* подчиненной формы **Вид подчиненная форма** загруженной формы **Основная** (рис.21-22).

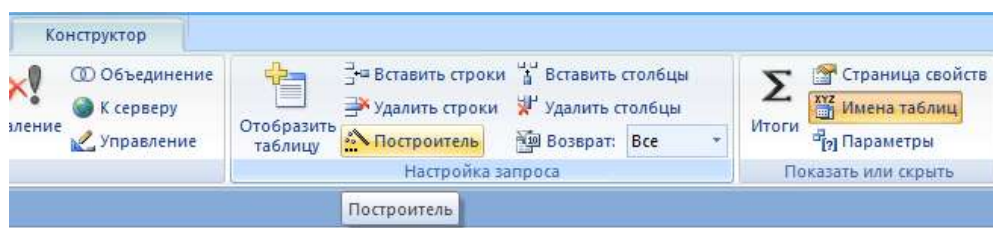
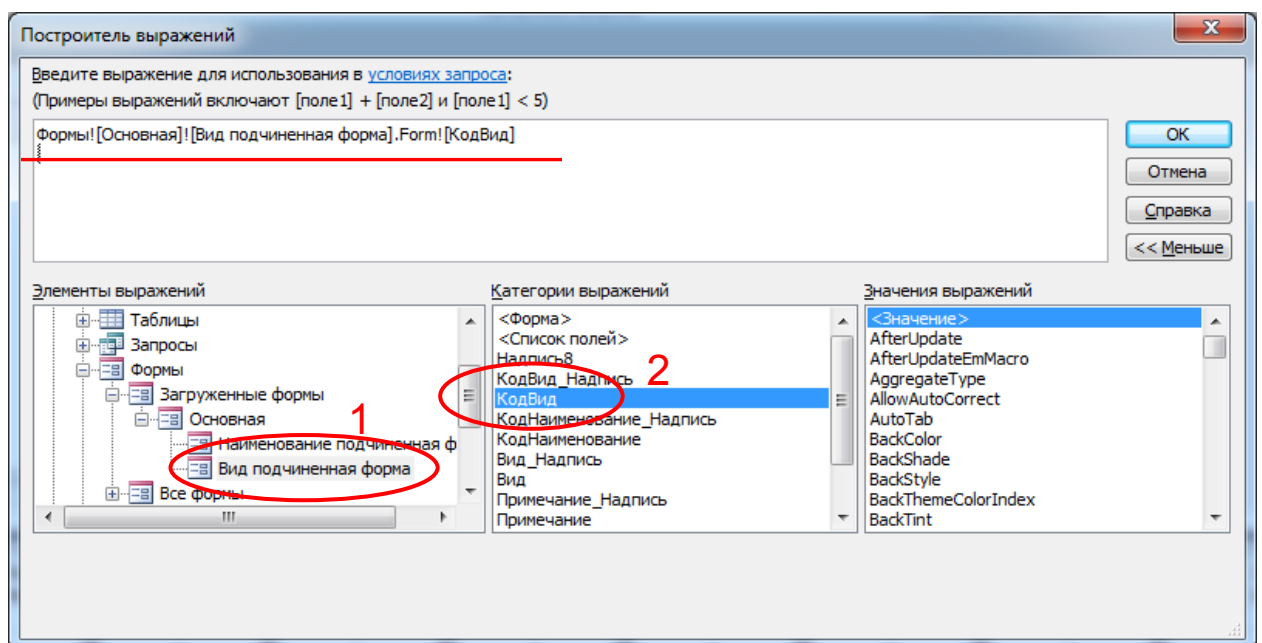


Рисунок 20. Вызов построителя выражений

Рисунок 21. Построение выражения для поля *КодВид*



Поле:	КодИзделие	КодВид	Наименование	Материал
Имя таблицы:	Изделие	Изделие	Изделие	Изделие
Сортировка:				
Вывод на экран:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Условие отбора:	[Формы]![Основная]![Вид подчиненная форма].[Form]![КодВид]			
или:				

Рисунок 22. Запрос для формы *Изделие*

Сохранить запрос с именем **ЗапросИзделиеФорма**.

### 2.3.2. Создание формы «Изделие» на основе запроса

Запустить мастер форм. В выпадающем списке *Таблицы и запросы* в качестве источника записей выбираем запрос **ЗапросИзделиеФорма**. В список *Выбранные поля* переносим все доступные поля запроса (рис. 23). Внешний вид формы - в один столбец (рис. 24). Сохраним форму с именем **Изделие** (рис. 25).

**Создание форм**

Выберите поля для формы.

Допускается выбор нескольких таблиц или запросов.

**Таблицы и запросы**

Запрос: ЗапросИзделиеФорма

**Доступные поля:**

**Выбранные поля:**

- Бренд
- Застежка
- Стиль
- Фото
- Цена
- Скидка
- Корзина
- Примечание

Отмена < Назад **Далее >** Готово

Рисунок 23. Выбор полей формы

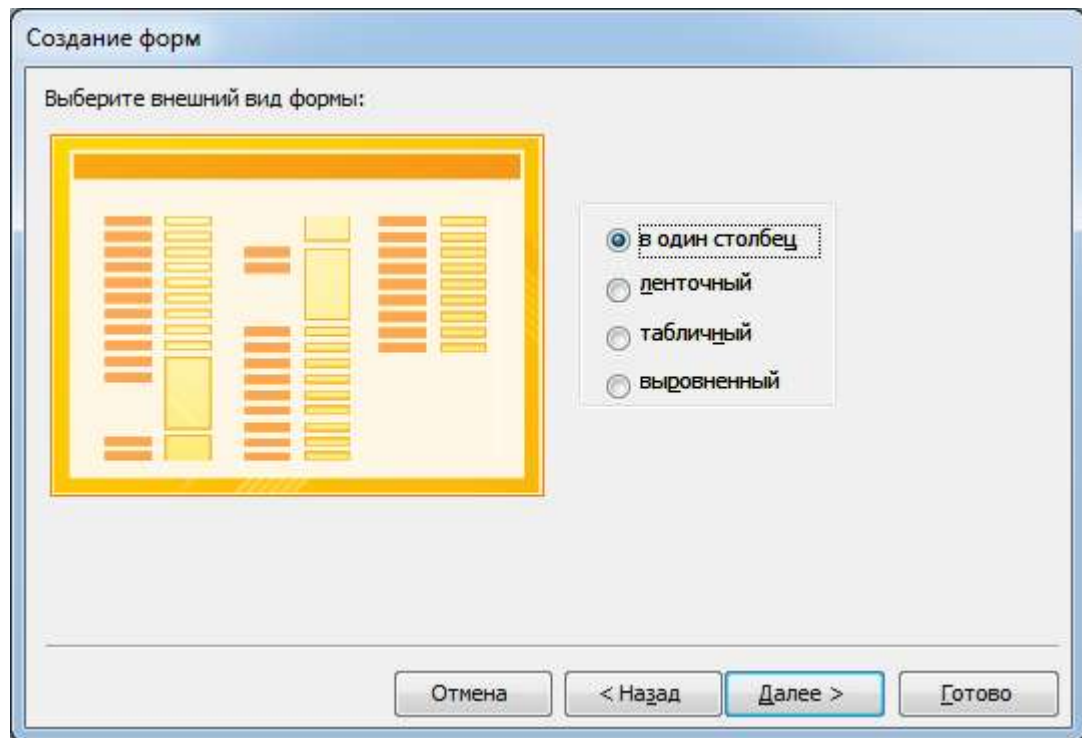


Рисунок 24. Выбор внешнего вида

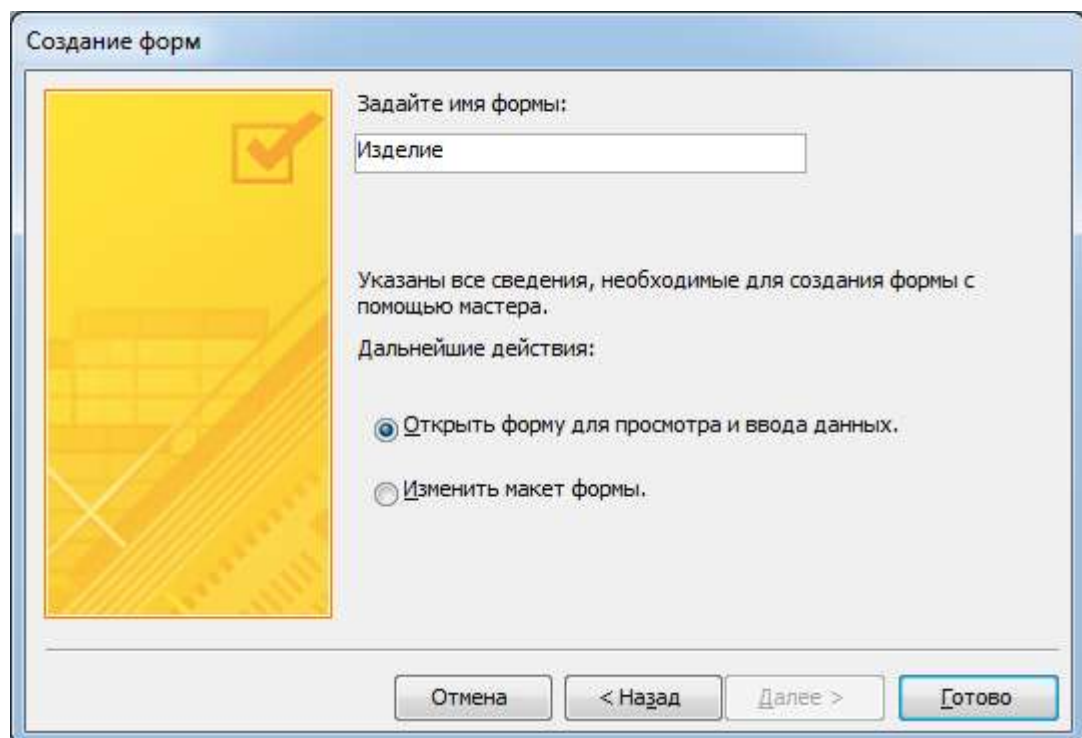


Рисунок 25. Задание имени формы

В режиме *Конструктор* редактируем объекты пользовательского интерфейса формы. Объекту форма устанавливаем следующие свойства:

- Область выделения (вкладка *Макет*) – *Нет*;
- Полосы прокрутки (вкладка *Макет*) – *Отсутствуют*;
- Всплывающее окно (вкладка *Другие*) – *Да*.

Примерный вид пользовательской формы **Изделие** представлен на рисунке 26.

Изделие

КодИзделие 1

КодВид

Наименовани

Материал

Цвет

Размер

Бренд

Застежка

Стиль

Фото

Цена

Скидка

Запись: 1 из 1

Нет фильтра

Поиск

Рисунок 26. Примерный вид формы **Изделие**

На форме **Основная**, используя мастер, создать командную кнопку для открытия формы **Изделие** (рис.27).

Основная

КодКатегория: 1

Категория: Одежда

Примечание:

КодНаименов	КодКатегории	Наименование	ЦелеваяАудит
1	1	Брюки	Мужчинам
2	1	Верхняя одежда	Мужчинам
3	1	Джемперы, свитеры и кардиганы	Женщинам
4	1	Джинсы	Мужчинам
5	1	Домашняя одежда	Женщинам
6	1	Майки	Женщинам
7	1	Нижнее белье	Женщинам

Изделие

КодВид	КодНаименов	Вид	Примечания
1	1	Прямые брюки	
2	1	Спортивные брюки	
3	1	Тайтсы	
4	1	Утепленные брюки	
5	1	Горнолыжные брюки	
6	1	Зауженные брюки	
7	1	Классические брюки	

Рисунок 27. Кнопка открытия формы *Изделие*

### 2.3.3. Создание кода для вставки новых записей

При открытии формы *Изделие* и заполнении текстовых полей формы, текстовое поле *КодВид* остается пустым. В этом случае происходит нарушение *ссылочной целостности* между таблицами *Вид* и *Изделие*. Для предотвращения нарушения целостности необходимо для события *До вставки()* формы написать код, обеспечивающий заполнение поля *КодВид* формы *Изделие*.

Листинг 1. Заполнение поля *КодВид* формы *Изделие*

```
Private Sub Form_BeforeInsert(Cancel As Integer)
Me.КодВид = [Forms]![Основная]![Вид подчиненная форма].[Form]![КодВид]
End Sub
```

### 2.3.4. Создание текстового поля цены изделия с учетом наличия возможной скидки

На этапе проектирования таблицы *Изделие* были созданы атрибуты *Цена* и *Скидка*. Первичный анализ формы *Изделие* показал, пользователю базы не хватает информации о конечной цене за изделие с учетом существующей скидки. Создать атрибут *Цена со скидкой* в таблице не позволяет требование нормализации, поэтому создадим дополнительное текстовое поле для хранения конечной цены изделия. Пересчет значения поля будет выполняться

автоматически. Назовем созданное текстовое поле **txtDiscountPrice** (рис.28). Выполним настройки текстового поля **txtDiscountPrice**:

- Формат поля (вкладка *Макет*) – *Денежный*;
- Данные (вкладка *Данные*) -  $= [Цена] - ([Цена] * [Скидка] / 100)$ ;
- Доступ (вкладка *Данные*) – *Нет*.

Рисунок 28. Создание текстового поля *Цена со скидкой*

### 2.3.5. Создание графического образа поля *скидка*

Поле *Скидка* показывает существующие скидки на изделие. Для лучшего визуального восприятия информации пользователем сделаем графическое дублирование текстового поля *Скидка*. Графический образ поля *Скидка* представляет из себя файл рисунка, изображающий размер предлагаемой скидки, созданный в любом графическом редакторе. Целесообразно все файлы изображений хранить в отдельной директории (рис.29).

Графический образ будет меняться:

- при выборе значения в поле со списком **Скидка**, событие *Нажатие кнопки (Скидка\_Click)*
- при переходе по записям формы **Изделие**, событие *Текущая запись (Form\_Current)*.

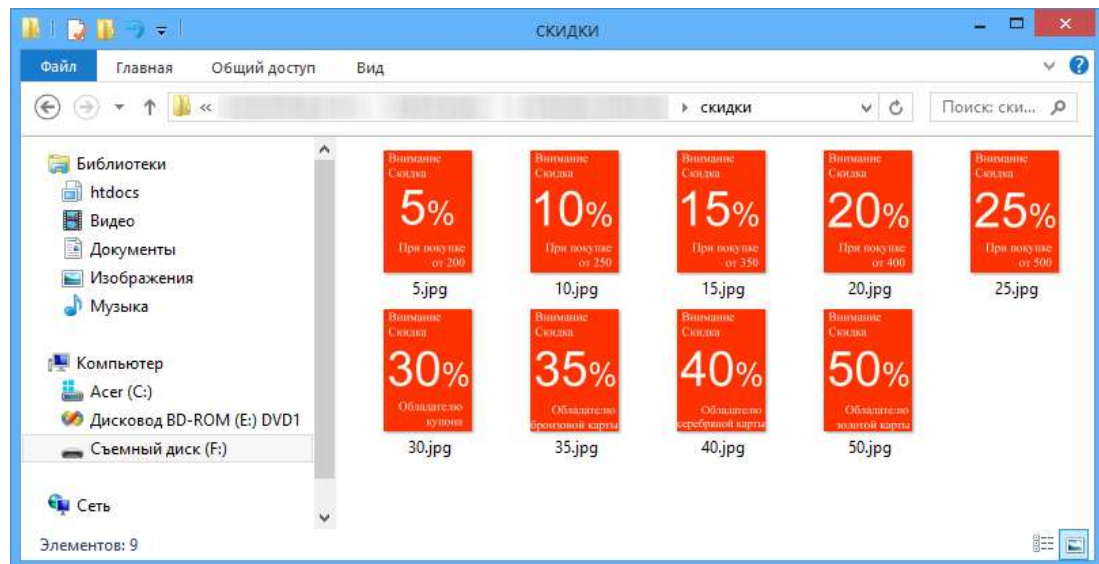
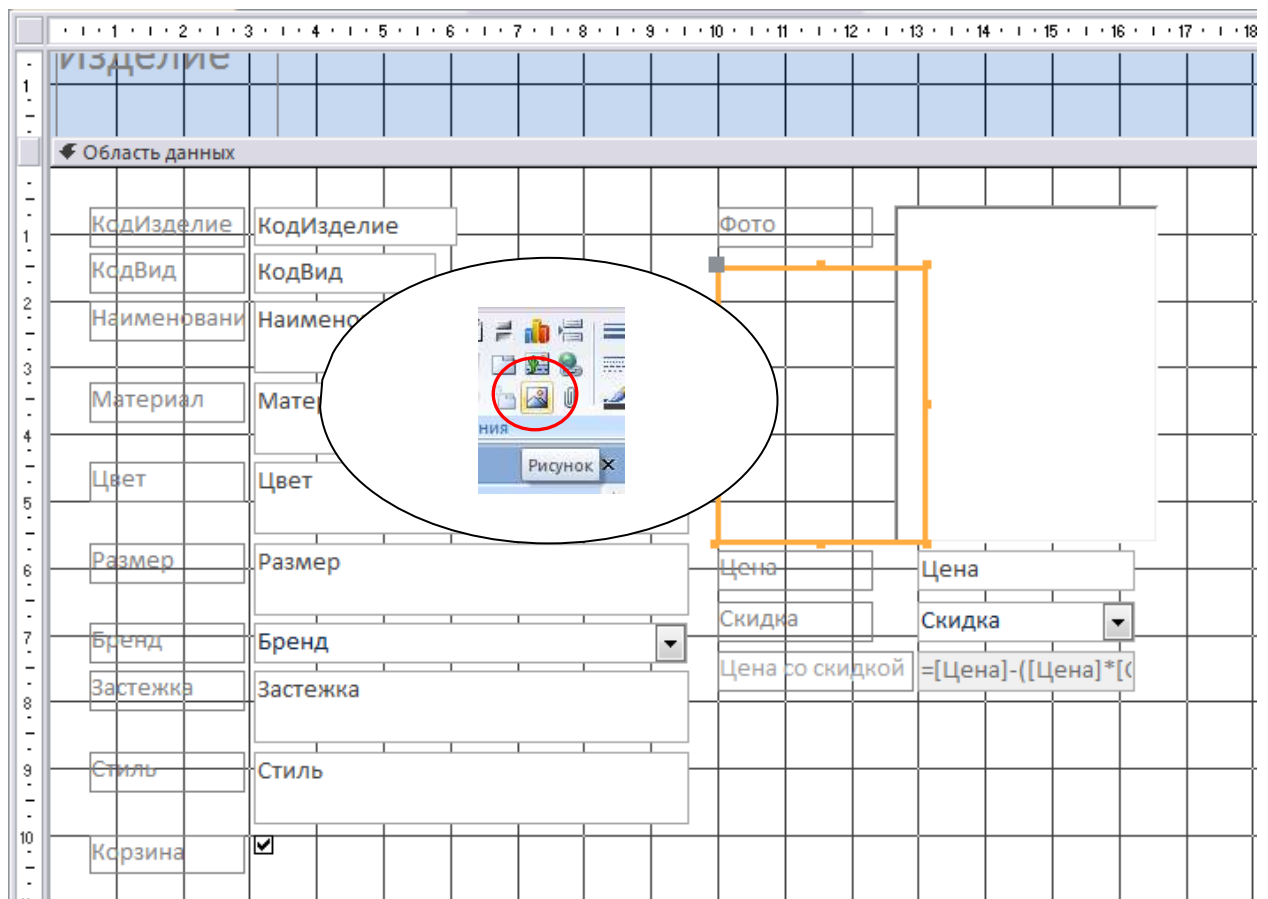


Рисунок 29. Графические изображения предлагаемых скидок

Вставим на форму *Изделие* объект *Рисунок*, назовем объект **imgSale** (рис.30).

Рисунок 30. Создание графического образа поля *Скидка*

Создадим соответствующие обработчики событий, напомним кодировку обработчиков.

Предлагаемый код обработчиков написан с учетом того, что графические файлы изображений хранятся в директории **скидки**.

**Важно.** Если графические файлы изображений хранятся в другой директории, не забудьте внести соответствующие изменения в листинг обработчиков событий.

Листинг 2. Код обработчика события *Текущая запись* формы *Изделия*

```
Private Sub Form_Current()  
On Error Resume Next  
    If (IsNull(Скидка)) Then  
        imgSale.Picture = ""  
    Else  
        imgSale.Picture = CurrentProject.Path & "\скидки\" & Скидка.Value & ".jpg"  
    End If  
End Sub
```

Листинг 3. Код обработчика события *Нажатие кнопки* списка *Скидка*

```
Private Sub Скидка_Click()  
On Error Resume Next  
    If (IsNull(Скидка)) Then  
        imgSale.Picture = ""  
    Else  
        imgSale.Picture = CurrentProject.Path & "\скидки\" & Скидка.Value & ".jpg"  
    End If  
End Sub
```

Проверить функциональную работоспособность поля со списком *Сидка*. При выборе нового значения меняется графический файл скидки на текущее изделие.

#### 2.4. Создание формы «Бренд»

С помощью *Мастера форм* создадим форму **Бренд**, выполняющую функцию справочника брендов. В качестве источника записей выбираем таблицу *Бренд*. Внешний вид формы - в *один столбец* (рис. 31-33).

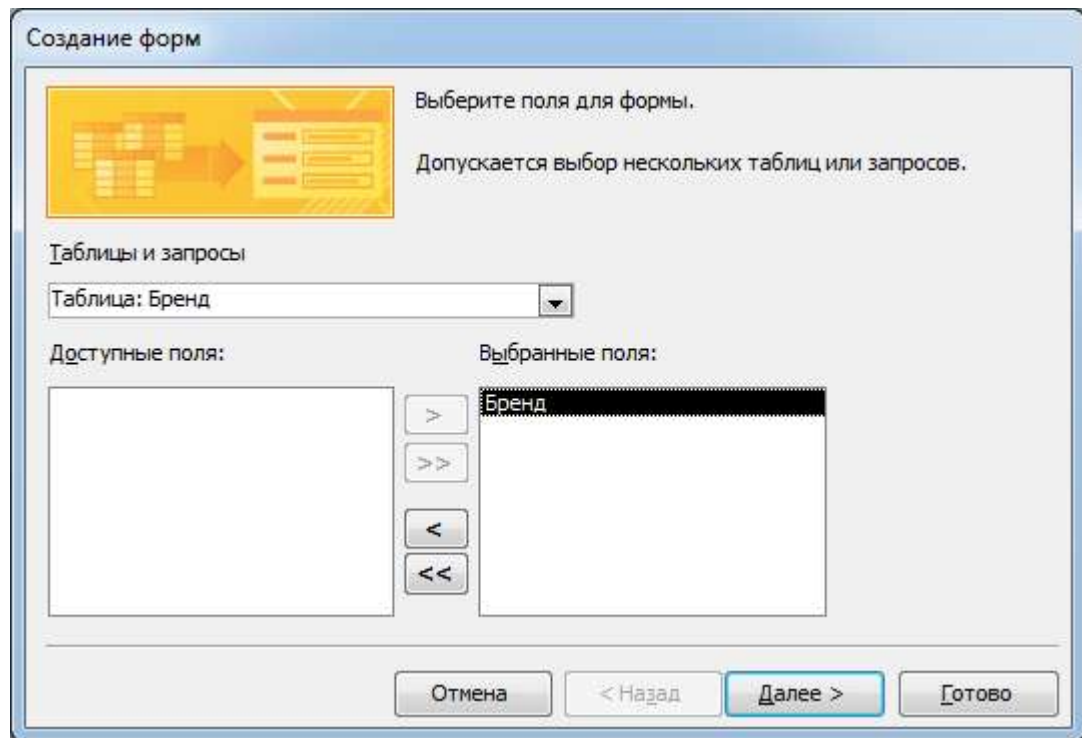


Рисунок 31. Выбор полей для формы

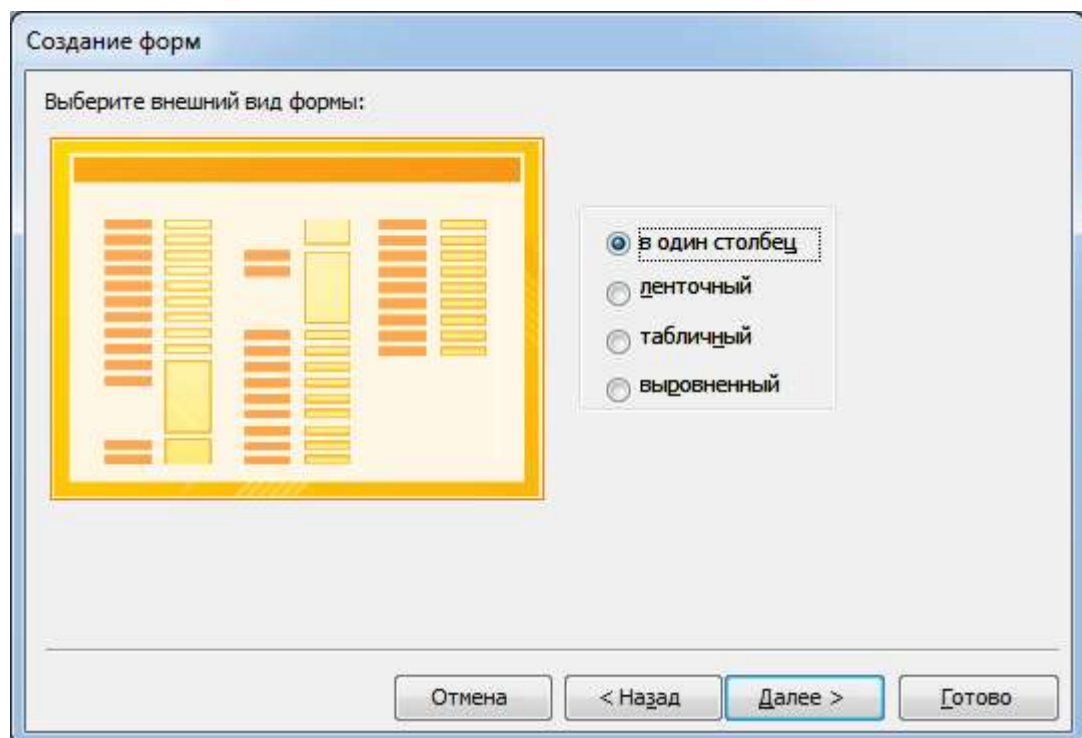


Рисунок 32. Выбор внешнего вида



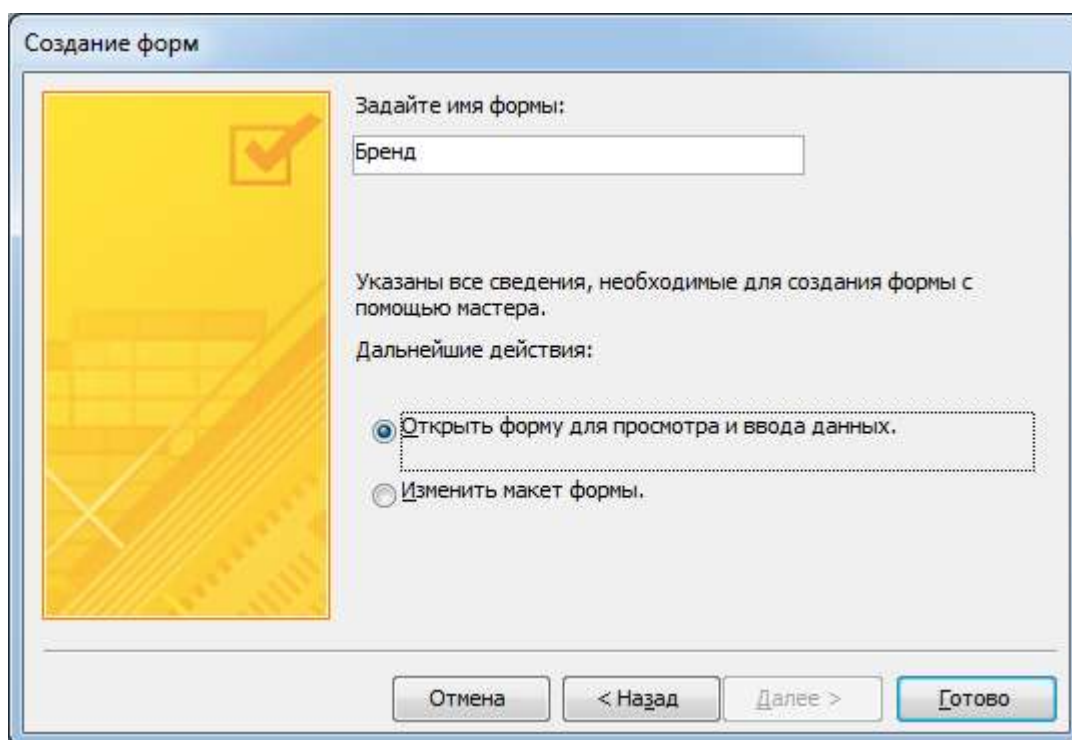


Рисунок 33. Задание имени новой формы

При необходимости отформатировать элементы управления формы *Бренд*.

Переходим в режим *Конструктор* формы *Изделие*. Используя мастер, создать командную кнопку для открытия формы-справочника **Бренд**, командную кнопку закрытия формы - *Закрыть форму*. Редактируем объекты пользовательского интерфейса. Объекту форма устанавливаем следующие свойства:

- Всплывающее окно (вкладка *Другие*) – *Да*.
- Область выделения (вкладка *Макет*) – *Нет*.

Примерный вид формы **Изделие** показан на рисунке 34.

Проверить работоспособность базы данных. Используя данные сайта *Lamoda* и созданные формы заполнить:

- таблицу **Бренд**. Количество записей таблицы: 10 – 15.
- таблицу **Изделие**. Количество записей таблицы: 5 – 10.

**Изделие**

КодИзделие:

КодВид:

Наименован:

Материал:

Цвет:

Размер:

Бренд:

Застежка:

Стиль:

Корзина: ☐

Фото:

**Внимание Скидка 30% Обладателю купона**

Цена:

Скидка:

Цена со скидкой:

Записи: 1 из 4 Нет фильтра Поиск

Рисунок 34. Примерный вид формы *Изделие*

### 3. Создание выпадающего списка навигации формы «Основная»

Создадим выпадающий список навигации, меняющий записи подчиненных форм для выбранной категории. Для этого в режиме *Конструктор*, формы **Основная**, на панели элементов выбираем пиктограмму *Поле со списком* и помещаем объект в верхнем свободном месте формы. Откроется диалог создания списков (рис.35-39).

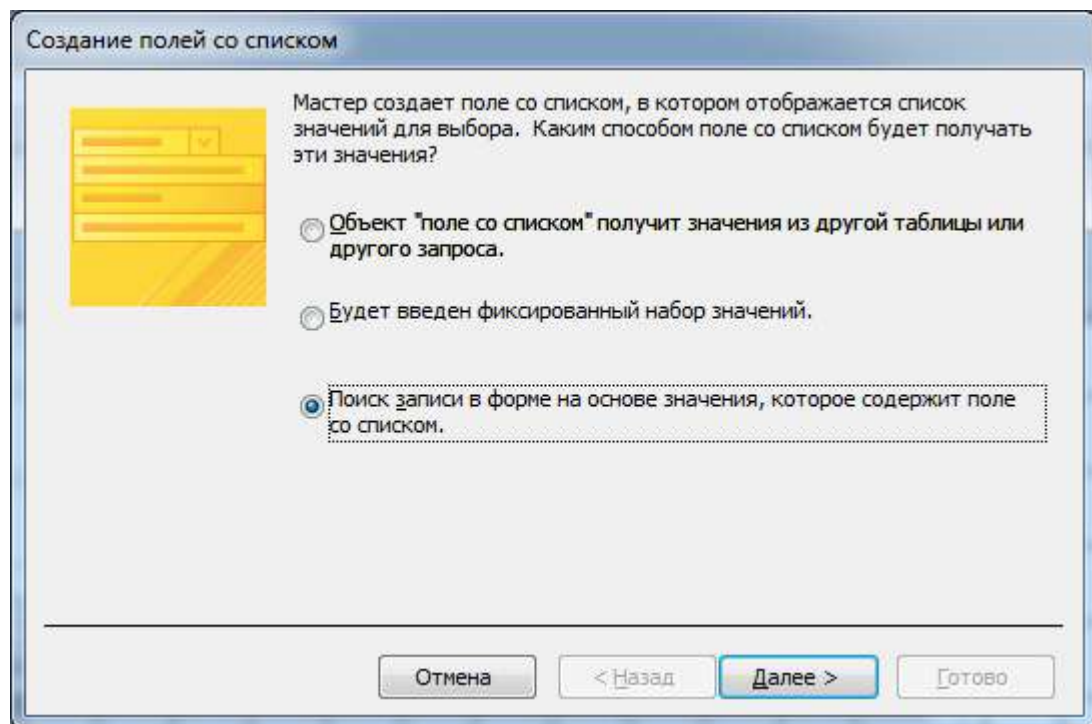


Рисунок 35. Создание выпадающего списка. Шаг 1

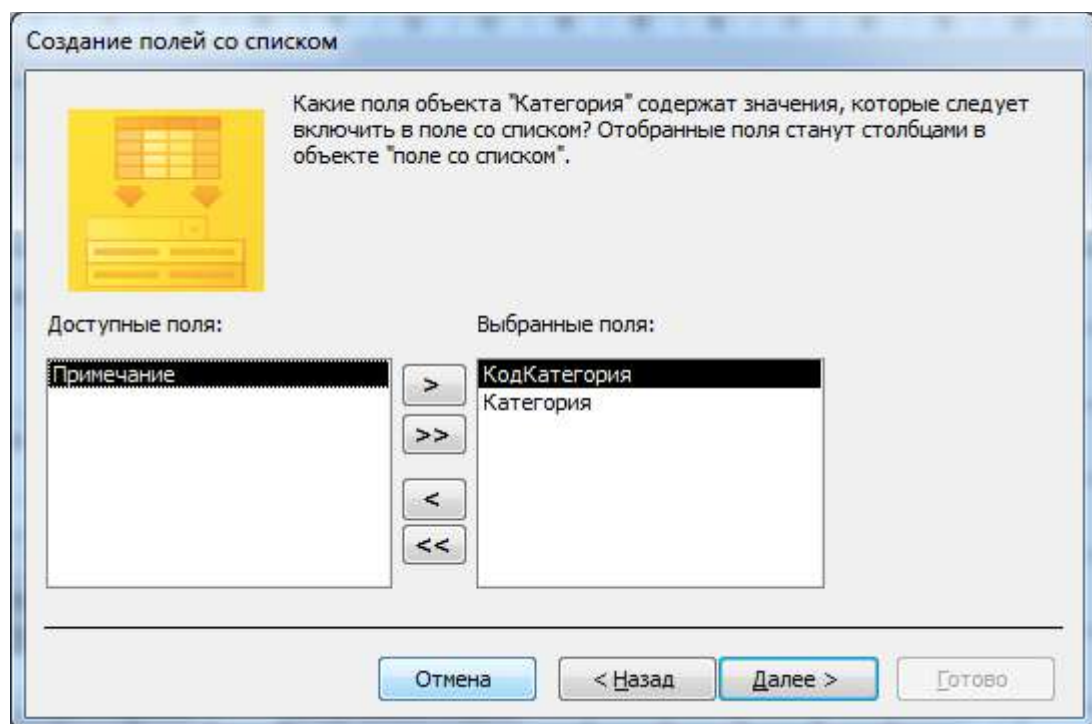


Рисунок 36. Создание выпадающего списка. Шаг 2

Создание полей со списком

Задайте ширину столбцов, которые содержит поле со списком.

Перетащите правую границу заголовка столбца на нужную ширину или дважды щелкните ее для автоматического подбора ширины.

☒ Скрыть ключевой столбец (рекомендуется)

Категория					
Одежда					
Обувь					
Аксессуары					

Отмена < Назад Далее > Готово

Рисунок 37. Создание выпадающего списка. Шаг 3

Создание полей со списком

Задайте подпись, которую содержит поле со списком.

☒ Быстрый переход к категории

Указаны все сведения, необходимые мастеру, чтобы создать поле со списком.

Отмена < Назад Далее > Готово

Рисунок 38. Создание выпадающего списка. Шаг 4

КодНаименов	КодКатегори	Наименование	ЦелеваяАудитс
1	1	Брюки	Мужчинам
2	1	Верхняя одежда	Мужчинам
3	1	Джемперы, свитеры и кардига	Женщинам
4	1	Джинсы	Мужчинам
5	1	Домашняя одежда	Женщинам
6	1	Майки	Женщинам
7	1	Нижнее белье	Женщинам

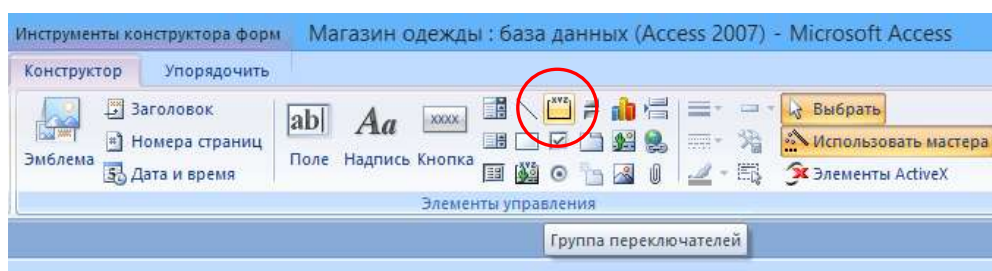
КодВид	КодНаимено	Вид	Примечани
1	1	Прямые брюки	
2	1	Спортивные брюки	
3	1	Тайтсы	

Рисунок 39. Примерный вид формы *Основная* с панелью быстрого перехода

Редактируем элементы интерфейса. Примерный вид формы **Основная** изображен на рисунке 39. Проверить работоспособность поля быстрой навигации формы *Основная*.

#### 4. Создание инструмента фильтрации записей

Реализуем механизм фильтрации записей таблицы *Наименование* по полю *ЦелеваяАудитория* для формы *Основная*. Фильтрацию будем выполнять с помощью добавленного объекта **Группа переключателей** (рис.40-45).

Рисунок 40. Создание объекта *Группа переключателей*

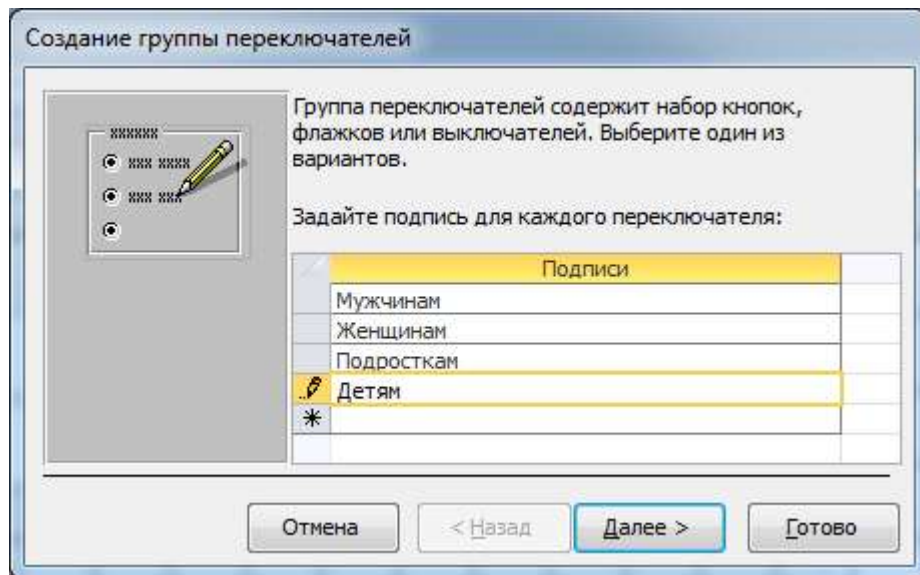


Рисунок 41. Создание подписей к переключателям

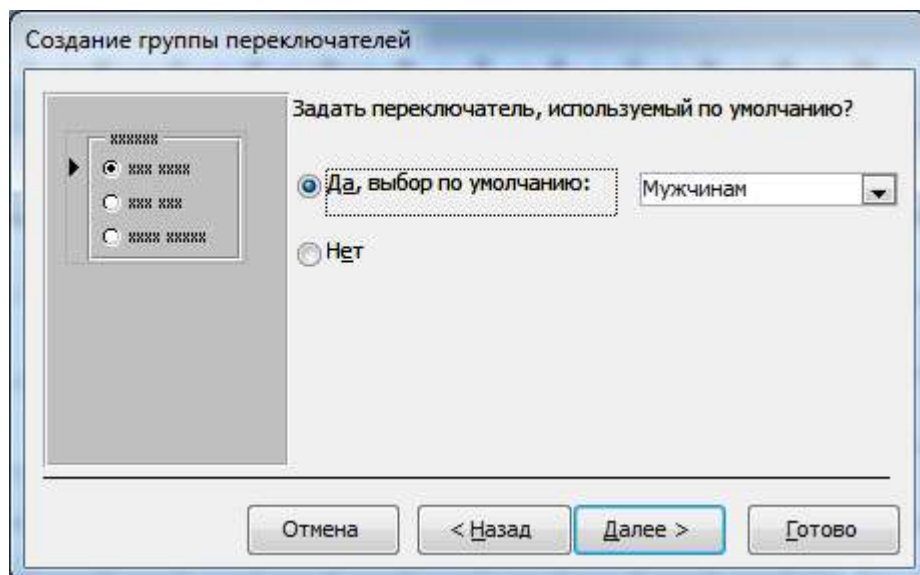


Рисунок 42. Создание переключателя по умолчанию



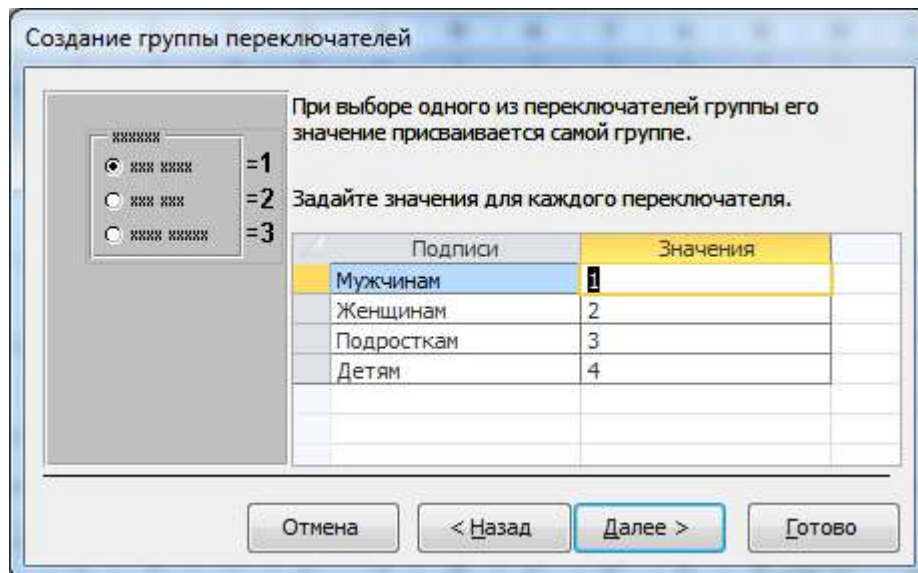
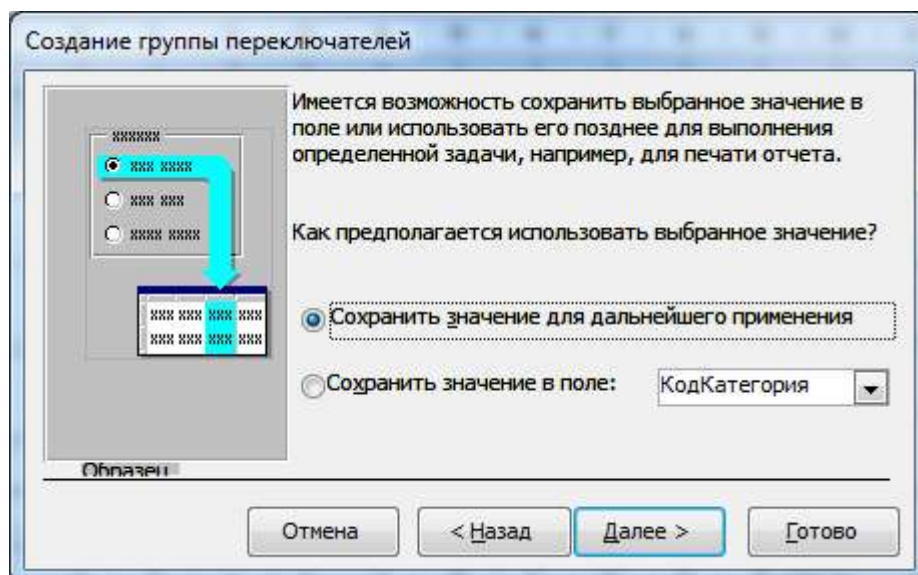


Рисунок 43. Создание значения каждого переключателя группы

Рисунок 44. Нажимаем кнопку *Далее*

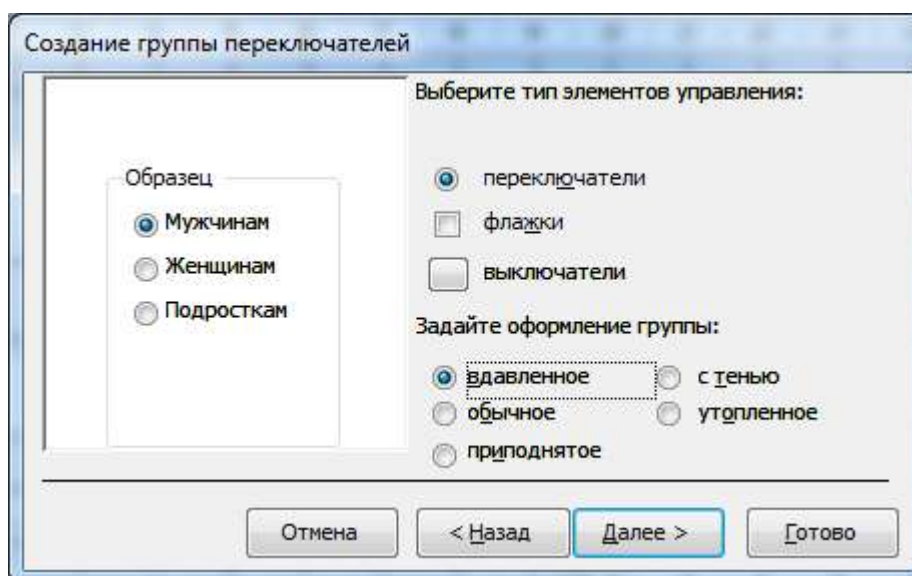


Рисунок 45. Выбор типа элемента управления

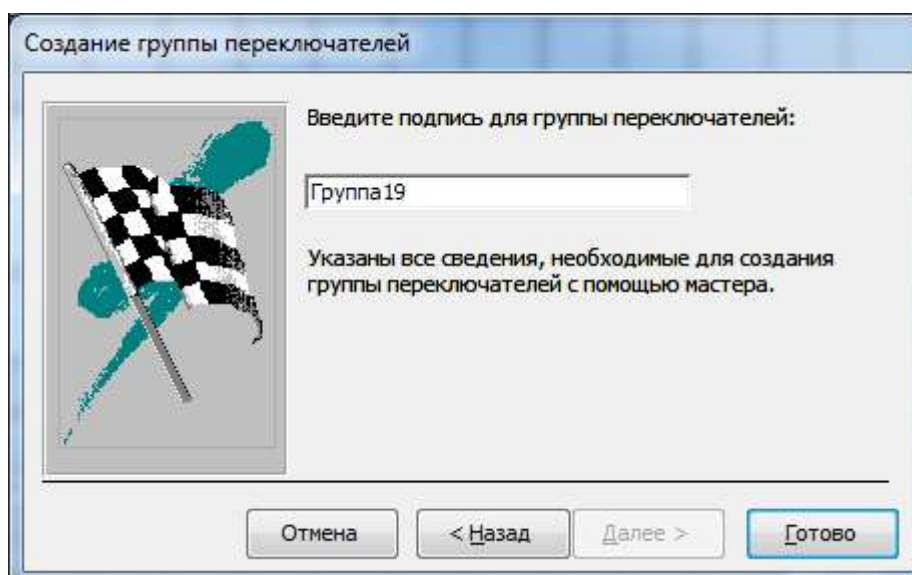
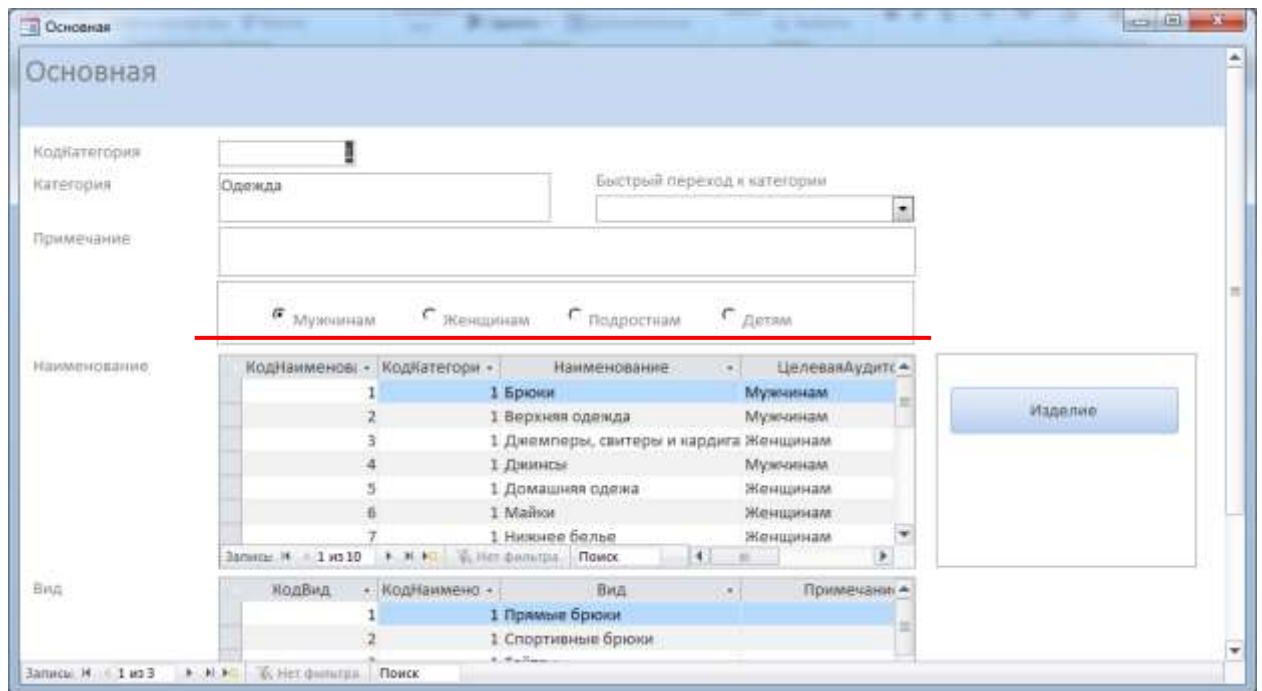


Рисунок 46. Подпись для группы переключателей

После форматирования панели фильтрации получаем конечный вид формы *Основная*.



Рисунок 47. Форма *Основная* с панелью фильтров таблицы *Наименование*

Добавленный объект *Группа переключателей* состоит из четырех переключателей, дадим каждому из них смысловые имена.

- переключатель надписи *Мужчинам* – ***cmbSwitchFilter\_man***.
- переключатель надписи *Женщинам* – ***cmbSwitchFilter\_wom***;
- переключатель надписи *Подросткам* – ***cmbSwitchFilter\_teen***;
- переключатель надписи *Детям* – ***cmbSwitchFilter\_kid***.

Фильтром по умолчанию мы выбрали значение *Мужчинам*. Для выполнения фильтрации создаем обработчик события *Загрузка* формы, пишем код, устанавливающий значение фильтра и включающий фильтрацию.

Листинг 4. Приводим фильтр формы к начальному значению. Включаем фильтр

```
Private Sub Form_Load()
    Me.Наименование_подчиненная_форма.Filter = "ЦелеваяАудитория = 'Мужчинам'"
    Me.Наименование_подчиненная_форма.FilterOn = True
End Sub
```

Клик по радиокнопкам должен приводить к изменению значения фильтра поля *ЦелеваяАудитория*. Для этого используем событие *Получение фокуса* для каждого переключателя панели фильтрации.

Листинг 5. Клик по переключателю *Мужчинам*

```
Private Sub cmbSwitchFilter_man_GotFocus()
    Me.Наименование_подчиненная_форма.Filter = "ЦелеваяАудитория = 'Мужчинам'"
End Sub
```

```
End Sub
```

#### Листинг 6. Клик по переключателю *Женщинам*

```
Private Sub cmbSwitchFilter_wom_GotFocus()  
Ме.Наименование_подчиненная_форма.Form.Filter = "ЦелеваяАудитория =  
'Женщинам' "  
End Sub
```

#### Листинг 7. Клик по переключателю *Подросткам*

```
Private Sub cmbSwitchFilter_teen_GotFocus()  
Ме.Наименование_подчиненная_форма.Form.Filter = "ЦелеваяАудитория =  
'Подросткам' "  
End Sub
```

#### Листинг 8. Клик по переключателю *Детям*

```
Private Sub cmbSwitchFilter_kid_GotFocus()  
Ме.Наименование_подчиненная_форма.Form.Filter = "ЦелеваяАудитория =  
'Детям' "  
End Sub
```

Проверить работоспособность средств фильтрации формы.

## 5. Создание отчетов

### 5.1. Создание отчета «Изделие»

#### 5.1.1. Создание источника записей для отчета «Изделие»

Для создания отчета необходимо в качестве источника записей создать запрос, выводящий информацию о текущей записи формы **Изделие**. Запрос создаем с помощью инструмента *Конструктор запросов* (рис. 48). Запрос должен выводить текущую запись, поэтому для поля *КодИзделия* вызываем *Построитель выражений*. В построителе выражений создаем ссылку на активную запись формы **Изделие** (рис. 49).

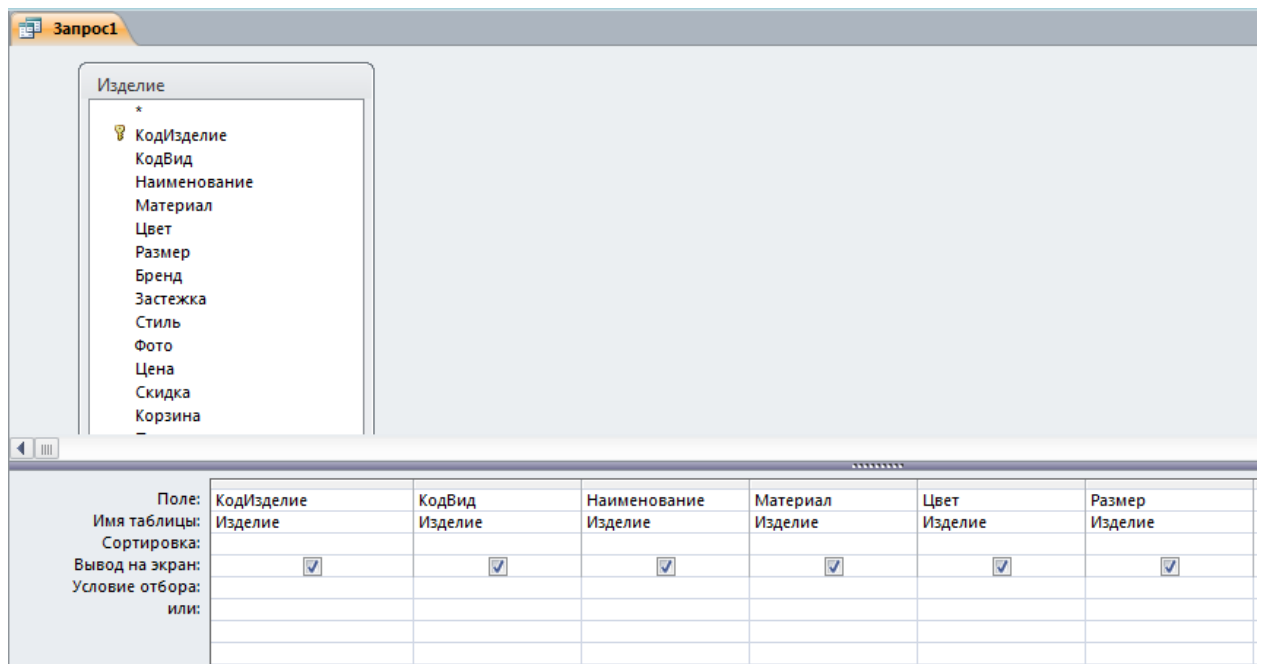
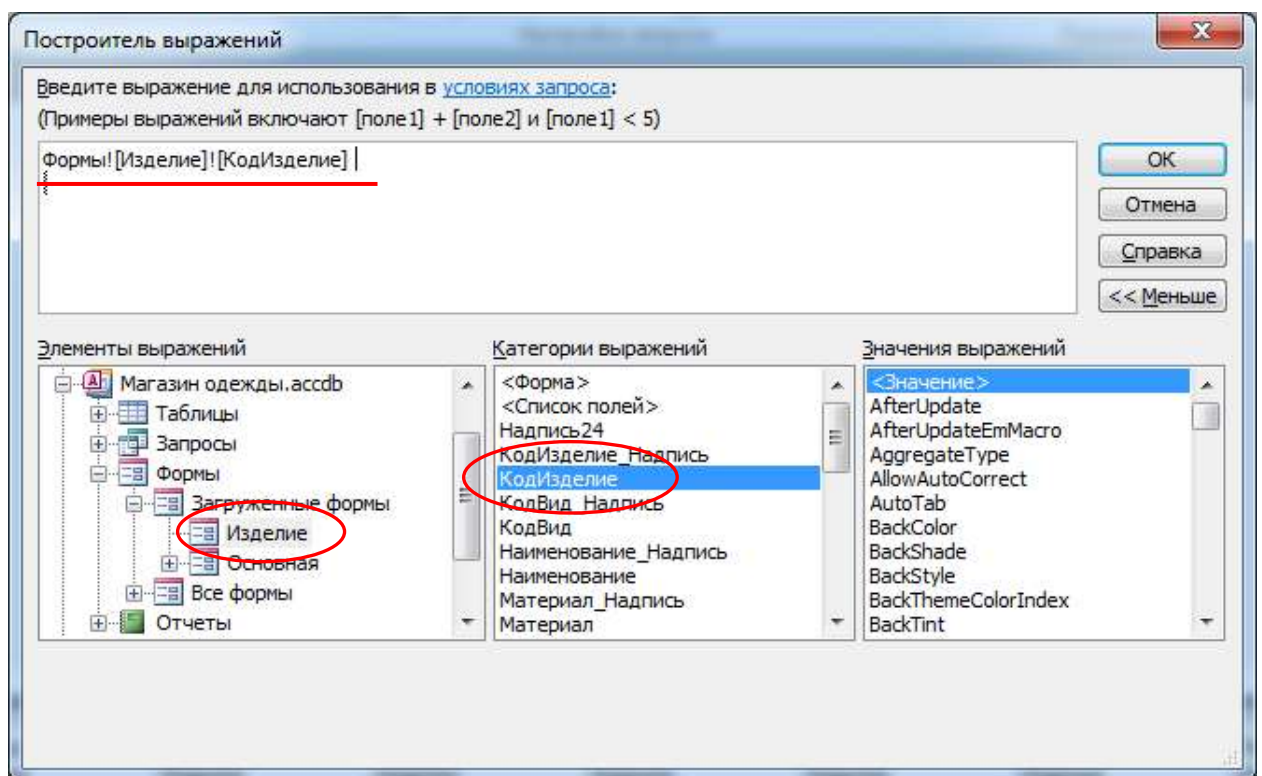


Рисунок 48. Построение запроса для отчета

Рисунок 49. Создание ссылки на активную запись формы *Изделие*

Поле:	Имя таблицы:	Сортировка:	Вывод на экран:	Условие отбора:
КодИзделие	Изделие		<input checked="" type="checkbox"/>	
КодВид	Изделие		<input checked="" type="checkbox"/>	
Наименование	Изделие		<input checked="" type="checkbox"/>	
Материал	Изделие		<input checked="" type="checkbox"/>	
Цвет	Изделие		<input checked="" type="checkbox"/>	
Размер	Изделие		<input checked="" type="checkbox"/>	
Бренд	Изделие		<input checked="" type="checkbox"/>	
Застежка	Изделие		<input checked="" type="checkbox"/>	
Стиль	Изделие		<input checked="" type="checkbox"/>	
Фото	Изделие		<input checked="" type="checkbox"/>	
Цена	Изделие		<input checked="" type="checkbox"/>	
Скидка	Изделие		<input checked="" type="checkbox"/>	
Корзина	Изделие		<input checked="" type="checkbox"/>	
Примечание	Изделие		<input checked="" type="checkbox"/>	

Рисунок 50. Запрос для отчета с условием отбора

Сохранить запрос с именем **ЗапросИзделиеОтчет**.

### 5.1.2. Создание отчета

Используя инструмент *Мастер отчетов* построить отчет **Изделие**. В качестве источника записей выбрать созданный запрос **ЗапросИзделиеОтчет** (рис. 51).

Выберите поля для отчета.

Допускается выбор нескольких таблиц или запросов.

Таблицы и запросы

Запрос: ЗапросИзделиеОтчет

Доступные поля:

Выбранные поля:

- Бренд
- Застежка
- Стиль
- Фото
- Цена
- Скидка
- Корзина
- Примечание

Отмена < Назад Далее > Готово

Рисунок 51. Выбор источника записей

Так как в отчет выводится единственная запись группировка и сортировка записей не требуется.

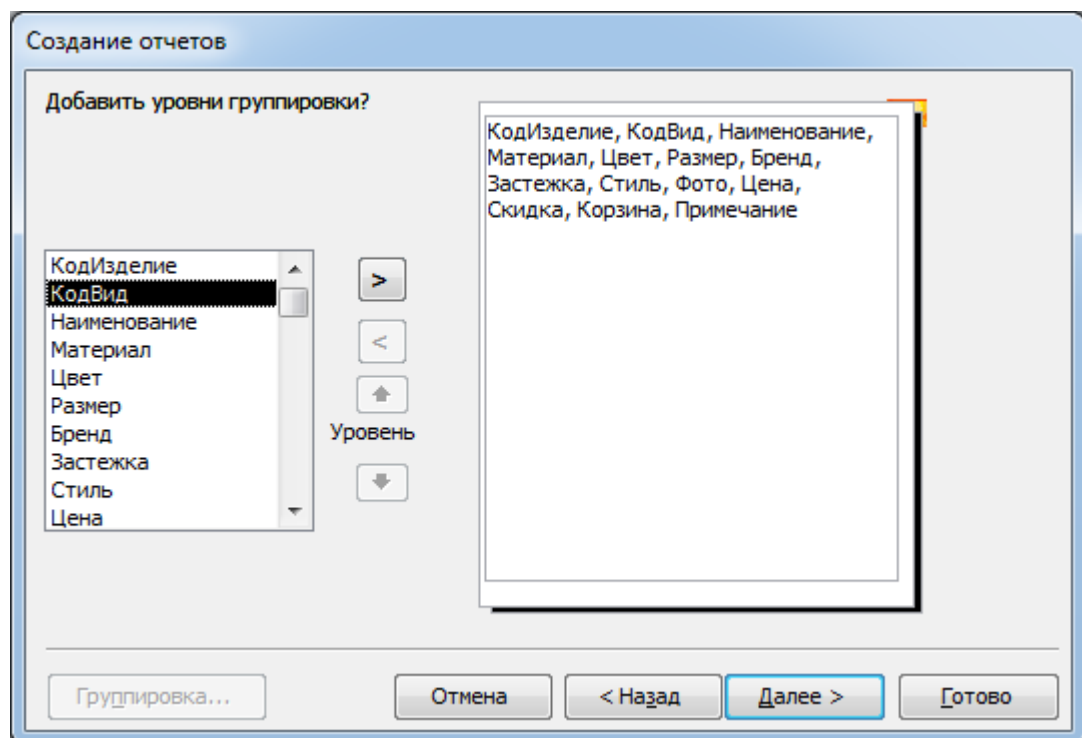


Рисунок 52. Группировку не используем

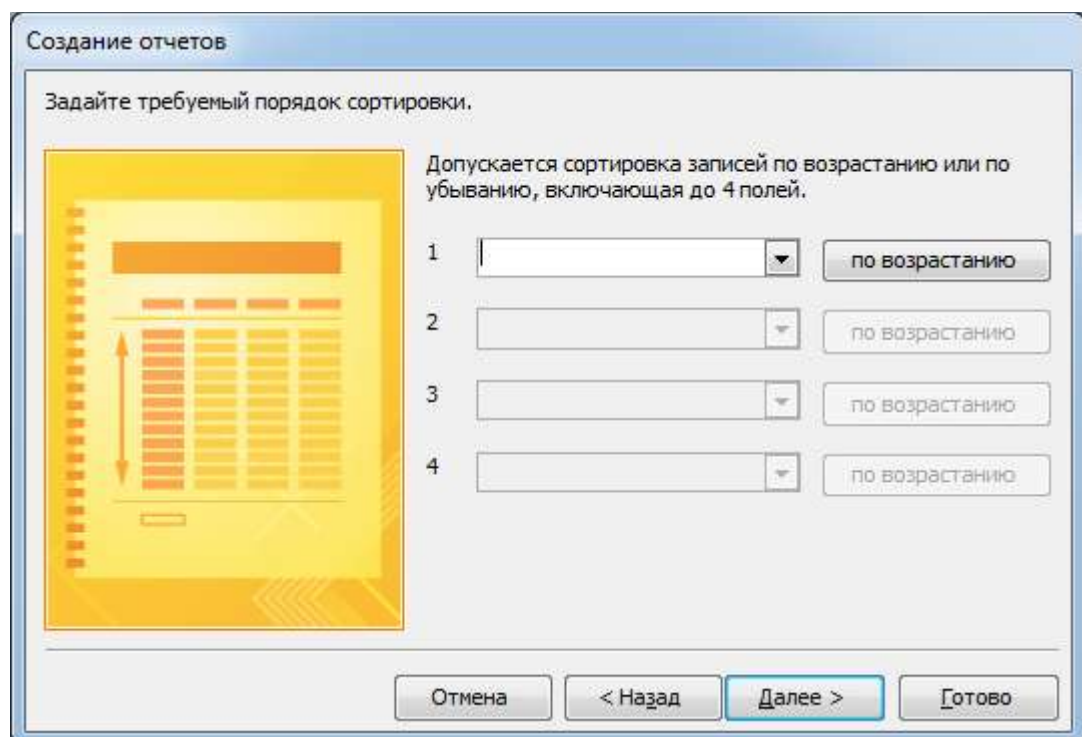


Рисунок 53. Сортировку не используем

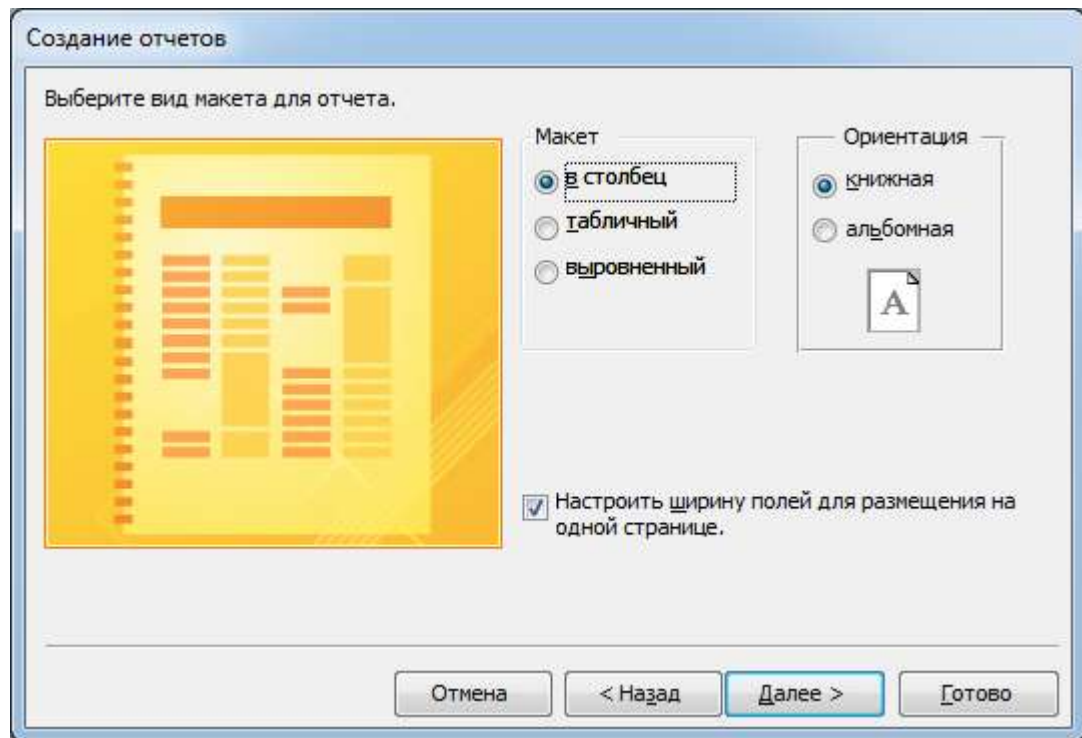


Рисунок 54. Выбор макета

Сохраним отчет с именем **Изделие**.

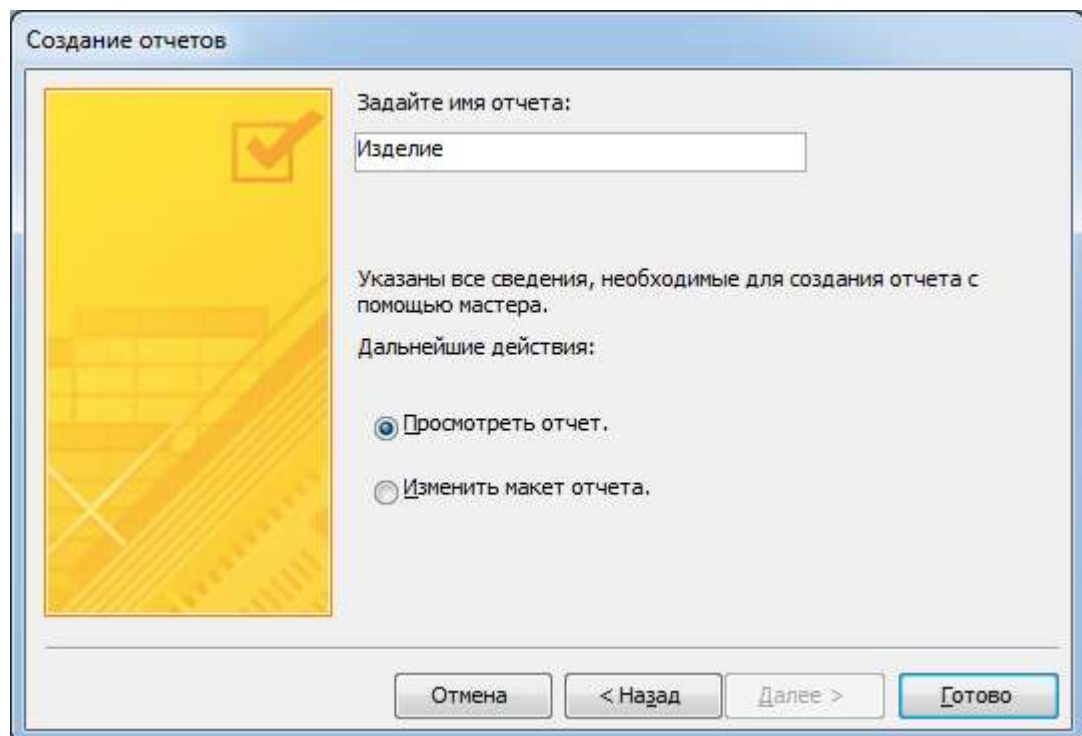


Рисунок 55. Задание имени отчета

Для созданного отчета в режиме *Конструктор* выполнить настройки:

- Всплывающее окно (вкладка *Другие*) присвоим значение – *Да*.
- Полосы прокрутки (вкладка *Макет*) – *Только по вертикали*.

- Для полей *КодИзделие*, *КодВид* свойство *Вывод на экран* (вкладка *Макет*) – *Нет* (можно скрыть эти поля и в форме *Изделие*).

На форме *Изделие* создать управляющую командную кнопку для вывода отчета текущего изделия.

Примерный вид формы *Изделие* представлен на рисунке 56.

The screenshot shows a web application window titled "Изделие". The form contains the following fields:

- КодИзделие:
- КодВид:
- Наименование:
- Материал:
- Цвет:
- Размер:
- Бренд: - Застежка:
- Стиль:
- Корзина: ☐

On the right side of the form:

- Фото:
- Внимание Скидка 30% Обладателю купона
- Цена:
- Скидка: - Цена со скидкой:
- Бренд:
- Открыть отчет:
- Закрыть форму:

At the bottom of the window, there is a status bar with the text: "Запись: 1 из 4", "Нет фильтра", and "Поиск".

Рисунок 56. Примерный вид формы *Изделие*

Примерный вид отчета *Изделие* представлен на рисунке 57.



The screenshot shows a web application window titled "Изделие". The main content area displays product information for "LC Wakiki" trousers. The details include:
 

- Наименование: LC Wakiki
- Материал: Хлопок, Эластан
- Цвет: Хаки
- Размер: 46, 48, 50, 52, 54
- Бренд: LC Wakiki (in a text input field)
- Застежка: Молния
- Стиль: (empty text input field)
- Фото: A photograph of a person wearing the trousers.
- Цена: 1 499,00р.
- Скидка: 30 (in a text input field)
- Корзина: An unchecked checkbox.
- Примечание: (empty text area)

 At the bottom left, the date "13 сентября 2019 г." is displayed. At the bottom right, it says "Стр. 1 из 1".

Рисунок 57. Примерный вид отчета *Изделие*

Проверить правильность вывода информации в отчете. Проверить общую работоспособность созданной базы данных.

## 5.2. Создание отчета «Корзина»

Добавим возможность выбора в корзину и последующей печати отчета понравившихся изделий пользователем. Для реализации такой возможности у нас предусмотрено поле логического типа *Корзина*.

### 5.2.1. Создание запроса на выборку отобранных изделий

Создадим запрос на выборку изделий отмеченных пользователем. Для этого скопируем запрос **ЗапросИзделиеФорма**, и вставим в базу с именем **ЗапросКорзина** (рис.58-59).



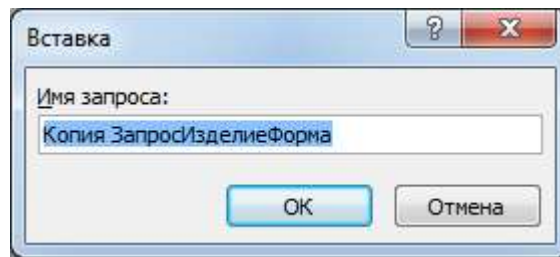
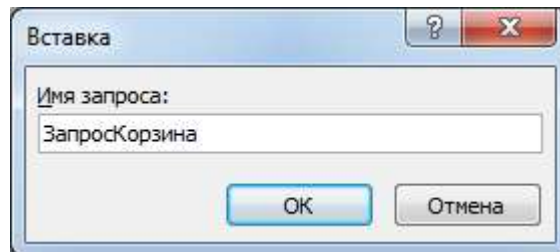
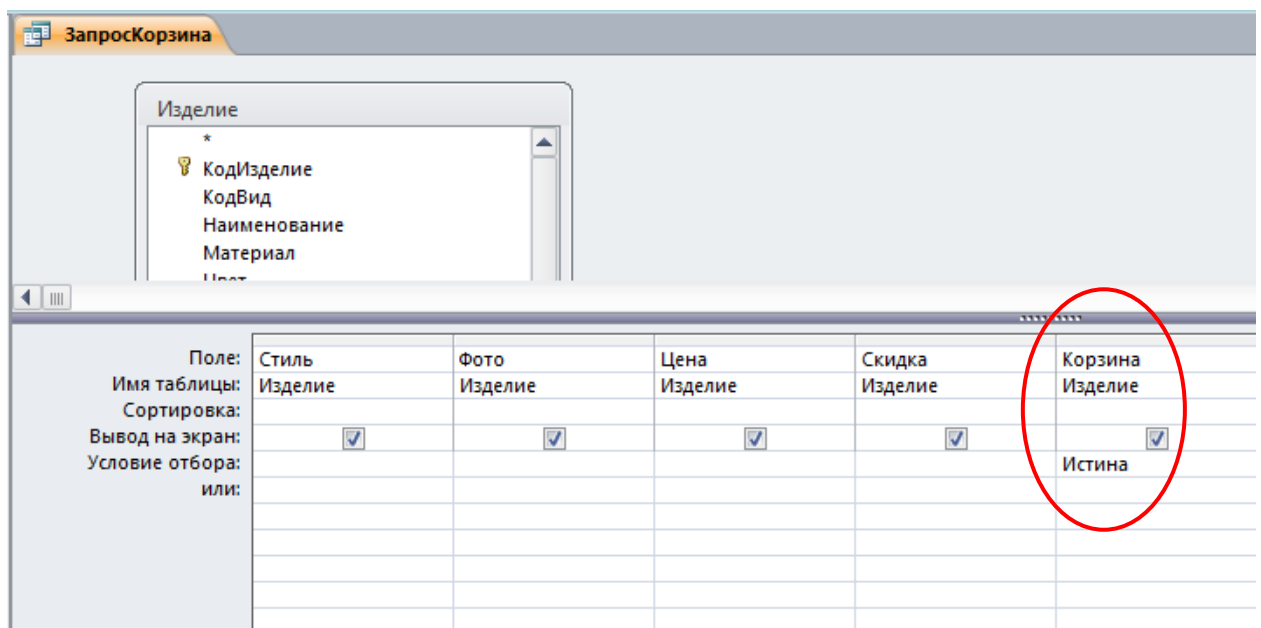
Рисунок 58. Вставка запроса *ЗапросИзделиеФорма*

Рисунок 59. Переименование запроса

Откроем запрос **ЗапросКорзина** в режиме *Конструктор*. Удалим условие отбора поля **КодВид** и создадим условие для поля **Корзина** (рис.60).

Рисунок 60. Создание условия отбора для поля *Корзина*

### 5.2.2. Создание отчета

Создадим отчет, отображающий изделия, выбранные пользователем. Для этого скопируем отчет **Изделие**, и вставим в базу с именем **Корзина** (рис.61-62).

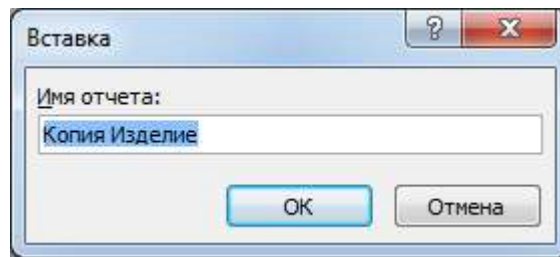
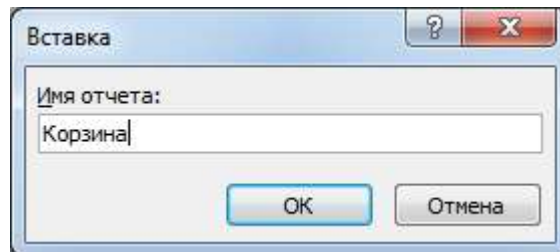
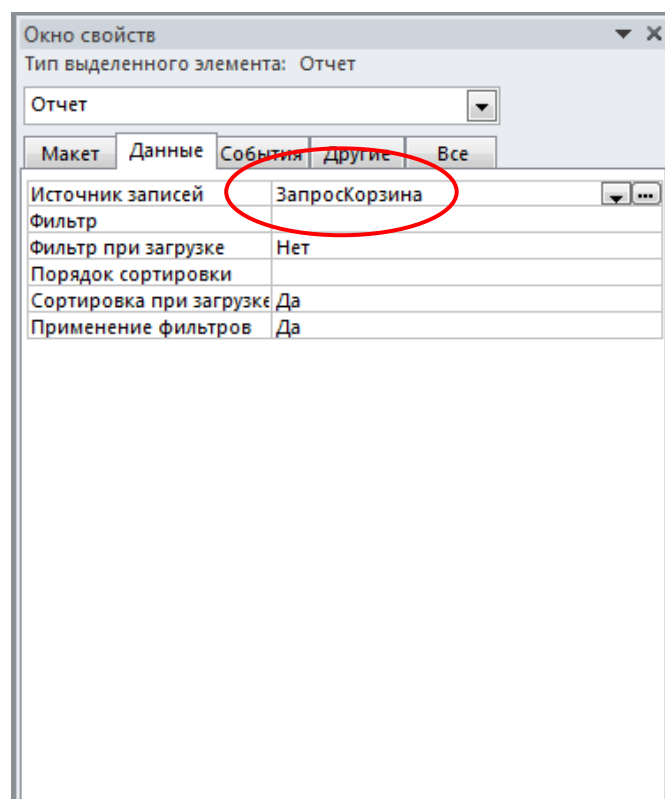
Рисунок 61. Вставка отчета *Изделие*

Рисунок 62. Переименование отчета

Откроем созданный отчет **Корзина** в режиме *Конструктор*. В окне свойств отчета, на вкладке *Данные* поменяем значение свойства *Источник записей* (рис.63):

- старое значение – *ЗапросИзделиеОтчет*;
- новое значение – *ЗапросКорзина*.

Рисунок 63. Указание источника записей отчета *Корзина*

На форме **Основная** с помощью *Мастера* создадим управляющую кнопку открытия отчета **Корзина** (рис.64)

### 5.2.3. Создание запроса обнуляющего выбор пользователя

Для работы с базой данных необходимо предусмотреть возможность обнуления выбора пользователя. Для этого откроем форму **Основная**. Создадим управляющую кнопку **Очистить корзину** с именем **cmdResetSelection**. В обработчик события **Нажатие кнопки** кнопки **Очистить корзину** (cmdResetSelection\_Click) вставим следующий код.

Листинг 9. Код отмены выбранных позиций пользователем

```
Private Sub cmdResetSelection_Click ()
    strSQL = "UPDATE Изделие SET Корзина = False WHERE Корзина = True"
    CurrentDb.Execute strSQL
End Sub
```

Отформатируйте элементы интерфейса, примерный вид формы **Основная** показан на рисунке 64. Протестируйте работоспособность созданной базы данных.

Рисунок 64. Примерный вид формы **Основная**

## 6. Заключение

Результатом выполнения практической части курсовой работы стала разработанная база данных «Магазин одежды», позволяющая автоматизировать работу по вводу, учету, поиску и выбору интересующей информации предметной области.

Спроектированная база данных состоит из четырех основных таблиц: *Категория*, *Наименование*, *Вид*, *Изделие*, хранящих полный набор данных объектов автоматизации и таблицы-справочника: *Бренд*.

В методических рекомендациях по выполнению практической части курсовой работы поставлены и решены следующие задачи:

- систематизированы теоретические знания о базах данных на основе реляционной модели;
- приобретены практические навыки при решении задач разработки объектов автоматизации;
- приобретен опыт проектирования реляционных баз данных;
- разработаны объекты проектируемой базы данных.

Спроектированная база данных является готовым для эксплуатации продуктом, который в дальнейшем может быть развит и модифицирован согласно задач разработчика.

В качестве основных шагов развития проекта можно предложить:

1. разработать графический интерфейс, придать формам максимально привлекательный внешний вид согласно выбранной предметной области;
2. расширить список атрибутов таблицы **Изделие**. Создать возможность классифицировать изделия по атрибутам *Бренды*, *Аксессуары*, *Premium* др. (<https://www.lamoda.ru/>);
3. создать многопользовательский вариант разработанной базы. Создать таблицу **Пользователь**, написать кодировку систем регистрации, авторизации, аутентификации пользователей;
4. создать вариант сетевой базы данных. Исходя из технических возможностей и уровня подготовки студентов база данных может быть реализована как в версии *файл-сервер*, так и в *клиент-серверном* исполнении.