Лабораторный практикум по курсу  
«Информационные системы и технологии»  
Семестр 2

**Лабораторные работы**

**«Работа с базами данных в MS Access 2007»**

**Подготовила А.А. Коробецкая**

Лабораторный практикум содержит 5 частей, охватывающих базовые аспекты создания и обработки баз данных MS Access 2007.

К сдаче предоставляется база данных (файл mdb) и отчет о проделанной работе (файл docx). Результат лабораторного практикума служит основой для курсового проекта.

Содержание

[Задание на лабораторный практикум 2](#_Toc382432115)

[Л.р. 1. Создание и заполенение базы данных 3](#_Toc382432116)

[Л.р. 2. Сортировка, фильтрация и поиск в таблице 9](#_Toc382432117)

[Л.р. 3. Создание запросов 11](#_Toc382432118)

[Л.р. 4. Создание форм 23](#_Toc382432119)

[Л.р. 5. Создание отчетов 31](#_Toc382432120)

[Варианты заданий 42](#_Toc382432121)

# Задание на лабораторный практикум

Вариант индивидуального задания выдается преподавателем.

1. Разработать структуру базы данных (схема БД и описание таблиц).
2. Заполнить базу данных.
3. Осуществить сортировку, фильтрацию и поиск по таблицам.
4. Выполнить в базе данных запросы на выборку:

* запрос на выборку полей одной таблицы;
* запрос на выборку полей двух таблиц;
* запрос на выборку с различными критериями отбора;
* запрос с вычисляемыми полями;
* запрос с итогами;
* запрос с параметром;
* перекрестный запрос.

1. Создать в базе данных формы:

* автоформы в один столбец, табличную и ленточную;
* составные формы подчиненную и связанную.

1. Создать в базе данных отчеты:

* автоотчет в один столбец;
* отчет по нескольким полям одной таблицы;
* отчет с группировкой и итогами.

Результаты отражаются в отчете по лабораторной работе. Отчет создается по разработанному в первом семестре шаблону. Отчет должен содержать:

1. Титульный лист.
2. Задание (общее и индивидуальное).
3. Результат выполнения каждого из пунктов 1-6:
   * Схема данных
   * Описание полей таблиц
   * Заполненные таблицы
   * Скриншоты результатов сортировки, фильтрации и поиска с указанием их параметров
   * Запросы и результаты их вызова
   * Скриншоты форм (в режиме просмотра)
   * Отчеты из базы данных

Как добавить в отчет те или иные результаты, указано в методичке.

## Пример индивидуального задания

Создать базу данных «Продажи товаров». В базе данных хранятся сведения о товарах (артикул, наименование, цена, количество, единицы измерения), о магазинах (наименование, город, адрес, телефон, фио директора), о продажах (№ чека, дата, товар, кол-во магазин), о наличии товаров в магазинах (товар, остаток на складе). Один товар может присутствовать в нескольких магазинах, в одном магазине продается много товаров. В одном чеке может быть много товаров, каждый товар может продаваться по разным чекам.

# Создание и заполенение базы данных

1. Изучите описание БД в своем индивидуальном задании. Попробуйте изобразить **примерную схему данных** (на бумаге). При необходимости уточните задание у преподавателя. Предложите варианты справочных таблиц.

На схеме необходимо указать таблицы и связи между ними, **кратность** связей. Если на схеме есть связи m-n (многие-ко-многим), их необходимо преобразовать к 1-n (один-ко-многим), введя вспомогательные таблицы.

Окончательный вариант схемы для примера:

**Товар**

**Магазин**

**Город**

**Продажа**

**Наличие**

1

n

n

1

1

1

1

n

n

n

В отчет схему вставлять не нужно.

1. Создайте в MS Access **новую базу данных** (пустую). Сохраните ее на локальный диск в формате mdb.

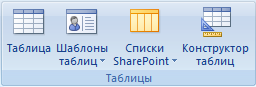
MS Access является локальной СУБД. Вся база данных хранится в одном файле (формат mdb).

Не рекомендуется работать с БД, расположенной на съемном носителе или в сети. Это замедляет работу и увеличивает риск повреждения БД.

1. **Создание таблиц** осуществляется через «Создание», группа кнопок «Таблица». С таблицами можно работать в двух основных режимах:

«*Таблица*» – ввод и редактирование данных.

«*Конструктор*» – формирование структуры таблице, определение полей, их типов и свойств.



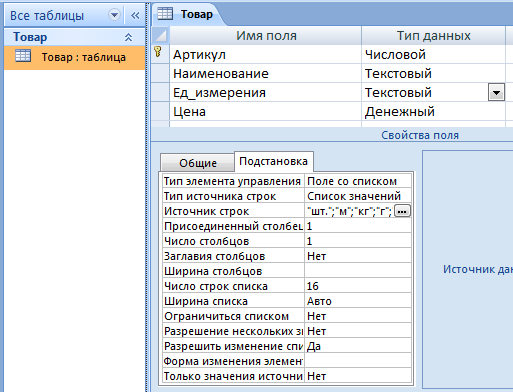
По умолчанию база данных содержит одну таблицу «Таблица1». Переименуйте ее в соответствии со своим вариантом.

Введите названия полей, укажите их типы. Подберите подходящий размер поля.

1. Названия таблицы и полей всегда пишутся в ед. числе.
2. В названиях допускаются буквы (русские не всегда), цифры, знаки - и \_.
3. Если у нескольких таблиц есть поля с одинаковыми названиями, их желательно называть по-разному, например, *Код\_магазина*, *Код\_города*.

Укажите ключевое поле или поля. При необходимости добавьте поле «Код» в качестве ключа.

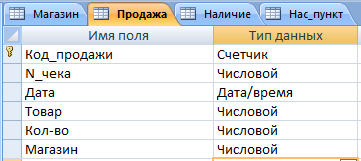
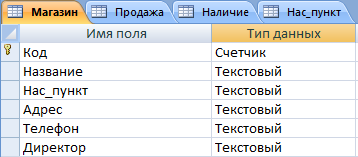
Для ключей-кодов часто используется тип «Счетчик». Это целочисленное поле, в которое при добавлении новой записи автоматически вводится следующее по порядку число. Таким образом, значения счетчика никогда не повторяются. Редактировать их вручную нельзя. Никакого смысла счетчик обычно не несет, и служит только для связывания таблиц, чтобы не использовать в качестве ключа поля с длинными значениями.

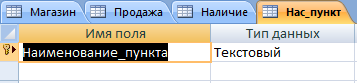
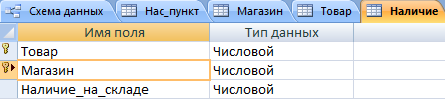


Аналогично создайте остальные таблицы.

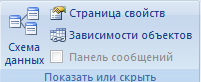
В рассматриваемом примере можно добавить ключи «Код» типа «Счетчик» в таблицы «Продажа» и «Наличие». Ни одно из полей этих таблиц не является уникальным. Использование нескольких полей одновременно возможно, но сильно загружает базу данных.

С другой стороны, ключи этих таблиц не участвуют в связях, поэтому создание ключа из нескольких полей не является ошибкой.





1. **Схема данных** создается через вкладку «Работа с базами данных» – «Показать или скрыть» – «Схема данных». Сами таблицы должны быть закрыты.



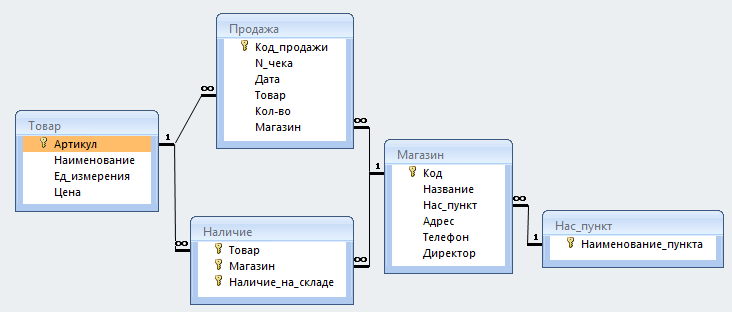
Добавьте на схему все таблицы БД. Разместите их на поле и добавьте связи.

Связь добавляется перетаскиванием поля одной таблицы к полю другой. Кратность связи задается автоматически. Типы данных связываемых полей должны совпадать (счетчик можно связать с длинным целым).

В свойствах связи установите флажок «Каскадное обновление» и снимите флажок «Каскадное удаление». Каскадное удаление может привести к серьезному повреждению базы при неосторожном удалении связанных записей.

Добавьте в отчет по первой работе скриншот схемы данных. Он должен быть хорошо читаемым.

Схема данных для примера.



1. **Настройка таблиц**.

Там, где это возможно, задайте в свойствах полей:

* уникальность значений (могут ли значения поля повторяться в разных записях);
* значение по умолчанию;
* шаблон ввода, например для даты \_\_.\_\_.20\_\_;
* условия на значения, например, количество >=0;
* выпадающий список для выбора одного из постоянных значений или подстановки значения из справочника;
* другие свойства.

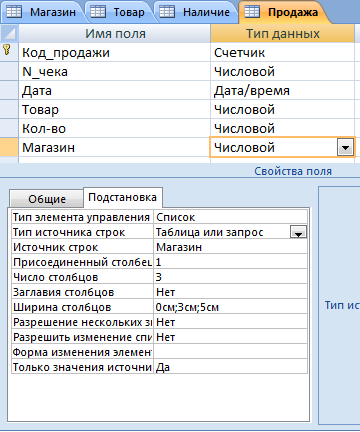
Для полей, которые участвуют в связях лучше использовать подстановку значений из списка. Это значительно упрощает ввод данных и позволяет избежать ошибок.

Список может содержать значения, введенные вручную, либо из другой таблицы или запроса.

Например, в таблицу «Наличие» код магазина и артикул товара желательно брать из соответствующих таблиц.

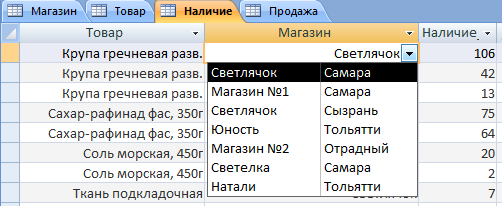
Удобнее всего то, что в списке может отображаться не только ничего не значащий столбец «Код», но и другие – «Наименование» и т.п.

Пример настройки подстановки значений в поле «Магазин» таблицы «Наличие»:

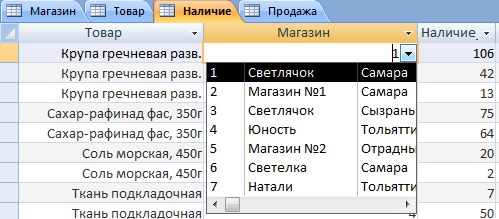


Обратите внимание на ширину столбцов: первый столбец (собственно, «Код\_магазина») имеет ширину 0. Он вообще не будет отображаться в списке, и даже в самом поле. Хотя реально в БД будет храниться именно код.

Результат в режиме таблицы:

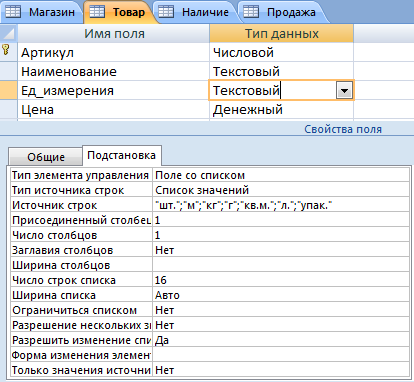


Если установить ширину первого столбца равной 1см, то таблица будет выглядеть так:



Можно построить и сложные запросы для выбора данных. Например, в таблице «Продажа» указывать не все товары, а только имеющиеся в наличии в данном магазине. Построение запросов рассматривается в л.р.3.

Пример подстановки из списка, введенного вручную, для поля «Ед\_измерения» таблицы «Товар»:

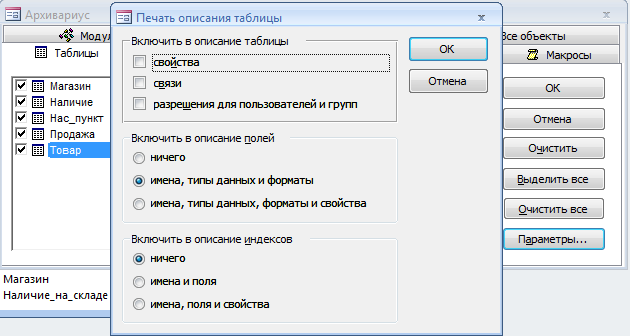


Здесь выбран тип элемента управления «Поле со списком», а не «Список», а свойство «Только значения источника» выставлено в «Нет». Это значит, что можно использовать и единицы измерения, не указанные в списке.

1. **Добавление описания таблиц в отчет.** Выполняется полуавтоматически, через «Работа с базой данных» – «Архивариус».

Установите «Параметры», как показано на скриншоте. Отметьте флажками все таблицы.

Откроется окно с перечнем таблиц и их полей. Выполните экспорт в MS Word (формат rtf). Удалите лишние строки, разрывы, настройте оформление шрифта и абзаца, и добавьте результат в отчет. Выделите **жирным ключевые поля**.



Результат для примера:

**Таблица: Магазин**

*Имя Тип Размер*

**Код Длинное целое 4**

Название Текстовый 20

Нас\_пункт Текстовый 15

Адрес Текстовый 255

Телефон Текстовый 15

Директор Текстовый 255

**Таблица: Наличие**

*Имя Тип Размер*

**Товар Длинное целое 4**

**Магазин Длинное целое 4**

**Наличие\_на\_складе Длинное целое 4**

**Таблица: Нас\_пункт**

Имя Тип Размер

**Наименование\_пункта Текстовый 255**

**Таблица: Продажа**

*Имя Тип Размер*

**Код\_продажи Длинное целое 4**

N\_чека Длинное целое 4

Дата Дата/время 8

Товар Длинное целое 4

Кол-во Длинное целое 4

Магазин Длинное целое 4

**Таблица: Товар**

*Имя Тип Размер*

**Артикул Длинное целое 4**

Наименование Текстовый 255

Ед\_измерения Текстовый 8

Цена Денежный 8

1. **Заполнение таблиц**.

Убедитесь, что схема данных, таблицы и их поля сформированы верно. Только после этого приступайте к заполнению таблиц.

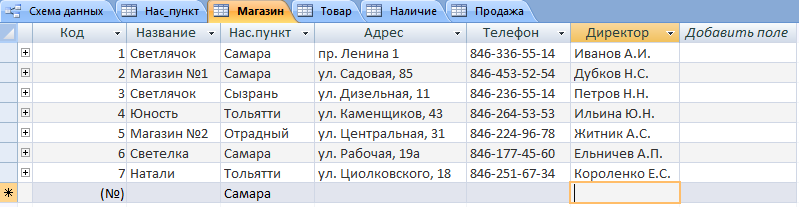
Последовательность заполнения: сначала заполняются таблицы на стороне связи 1, а потом – на стороне связи «много».

Необходимо заполнить:

* в таблицах на стороне связи «много» – не менее 15 записей,
* во таблицах на стороне связи «один» – не менее 7,
* в справочниках по необходимости.

При заполнении добавьте повторяющиеся значения там, где это возможно.

Пример заполненной таблицы:



1. Экспорт заполненных таблиц в отчет выполняется через «Внешние данные» – «Экспорт» – «Экспорт в файл rtf».

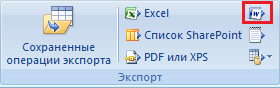


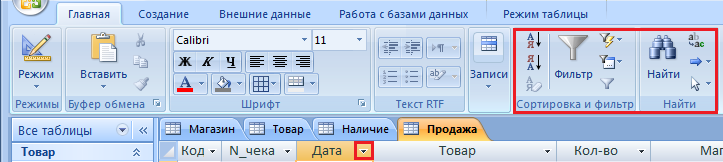
Таблица должна быть выделена или открыта.

# Сортировка, фильтрация и поиск в таблице

Реальные БД содержат десятки, сотни тысяч и даже миллионы записей. Просмотреть все эти записи вручную, чтобы найти нужную информацию, для человека почти невозможно.

Поэтому основными и одними из наиболее важных операций являются сортировка, фильтрация и поиск нужной записи. Они практически аналогичны сортировке и фильтрации в Excel.

Сортировка и фильтрация выполняется либо через вкладку «Главная», либо по клику непосредственно по наименованию поля в режиме «Таблица». Поиск также находится на вкладке «Главная», либо вызывается комбинацией клавиш Ctrl+F.

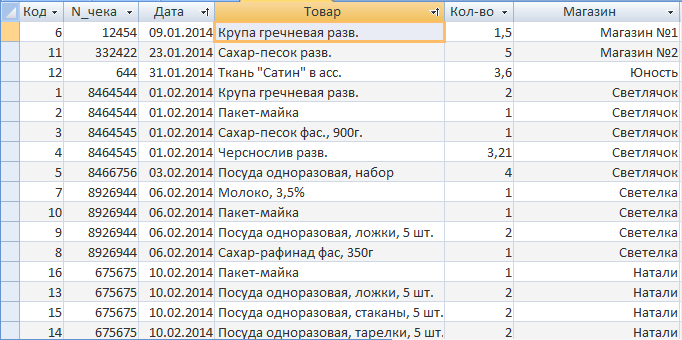


1. Откройте одну из основных таблиц для сортировки.

В нашем примере выполним сортировку продаж сначала по дате (по возрастанию), затем по наименованию товара (по алфавиту).

Многоуровневая сортировка в Access отсутствует (точнее, она выполняется через запросы). Для сортировки по нескольким столбцам, она выполняется для каждого столбца в обратном порядке.

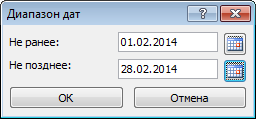
Т.е. в нашем примере необходимо сначала отсортировать продажи по наименованию товара, а потом по дате продажи. У отсортированных столбцов в заголовке появляется значок стрелки, показывающий направление сортировки:



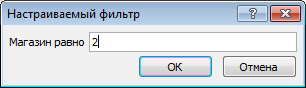
В отчете необходимо указать по каким полям выполнялась сортировка (не менее 2-х) и в каком направлении; вставить скриншот результата.

1. Выберите и откройте таблицу для фильтрации.

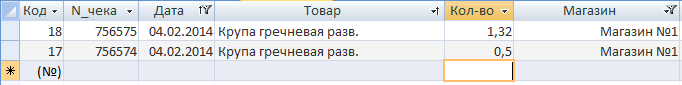
В нашем примере найдем все продажи в магазине №1 в феврале. Фильтр даты в столбце «Дата» можно указать диапазон дат с 01.02.14 по 31.02.14 или выбрать «В текущем месяце» (методичка составлялась в феврале 2014г.).



Фильтр поля «Магазин»:



Указывать надо фильтр для реально содержащихся в таблице данных, т.е. для кода магазина.



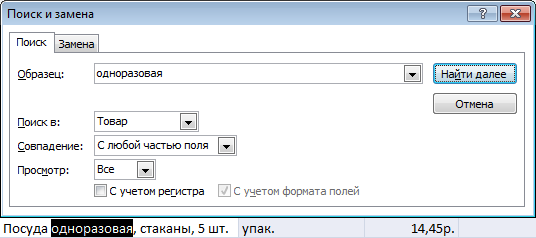
В отчет необходимо вставить скриншоты условий фильтрации и полученной таблицы. Фильтр желательно выбрать таким образом, чтобы в результате получилось несколько строк.

При закрытии таблицы фильтрация сбрасывается, а сортировка сохраняется.

1. Отдельную запись в таблице помогает найти «Поиск». Во многом его функции пересекаются с фильтрацией. Основные отличия:
   * переход к найденной записи (можно сразу редактировать);
   * возможность поиска не в одном поле, а во всей таблице.

Также возможна автоматическая замена значений.

Пример: найти в таблице «Товар» записи, содержащие слово «одноразовая».



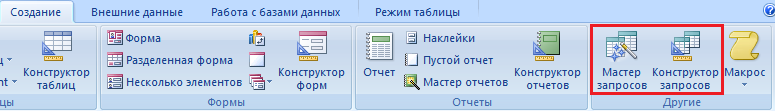
В отчет вставить скриншот окна поиска и найденной записи.

# Создание запросов

Запросы являются одним из основных инструментов работы с БД. Они позволяют выбирать из таблиц нужные данные, объединять таблицы по связанным полям, редактировать содержимое таблиц, выполнять расчеты и многое другое.

Фактически, любое действие с БД является по сути запросом. Просто в Access многие запросы автоматизированы, и пользователь их не видит. Однако в прошлых лабораторных работах, когда вы просматривали таблицы, вводили в них данные, выполняли сортировку и фильтрацию, Access каждый раз формировал нужные запросы к БД.

Теперь составим запросы самостоятельно. Создание запроса осуществляется через вкладку «Создание» и возможно двумя способами: через «Мастер запросов» или «Конструктор». Мастер позволяет создавать несколько типовых запросов. Затем их всегда можно отредактировать в Конструкторе.



Существует большое число различных видов запросов. В данном практикуме мы выполним следующие:

* запрос на выборку полей одной таблицы;
* запрос на выборку полей двух таблиц;
* запрос на выборку с различными критериями отбора;
* запрос с вычисляемыми полями;
* запрос с итогами;
* запросы с параметром;
* перекрестный запрос.

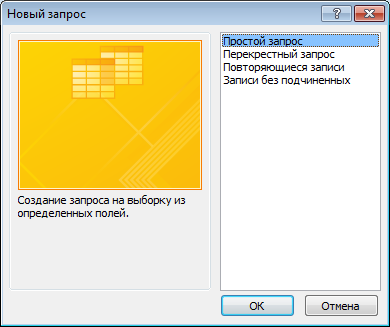
Замечание: вам необходимо создать только один запрос каждого типа.

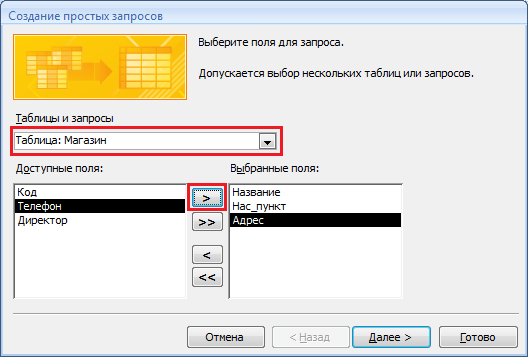
1. *Запрос на выборку полей из одной таблицы* – самый простой вид запросов. Он позволяет отобразить только те столбцы таблицы, которые нам нужны. Чаще всего они используются для выборок из больших таблиц, которые содержат 10 столбцов и более.

В нашем примере выполним выборку из таблицы «Магазин». Отобразим столбцы «Название», «Нас.пункт» и «Адрес».

Данный запрос можно выполнить как с помощью мастера, так и с помощью Конструктора.

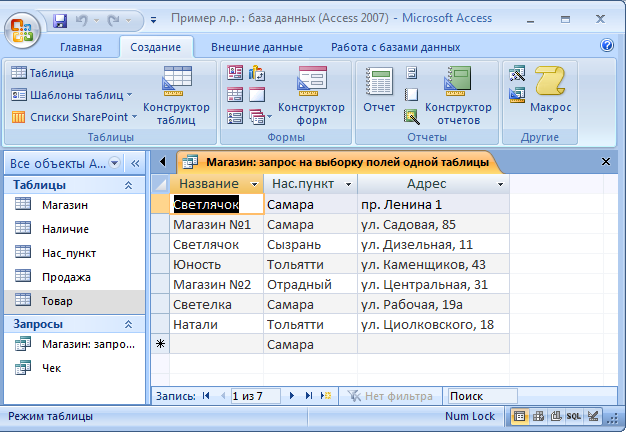
Мастер запросов:



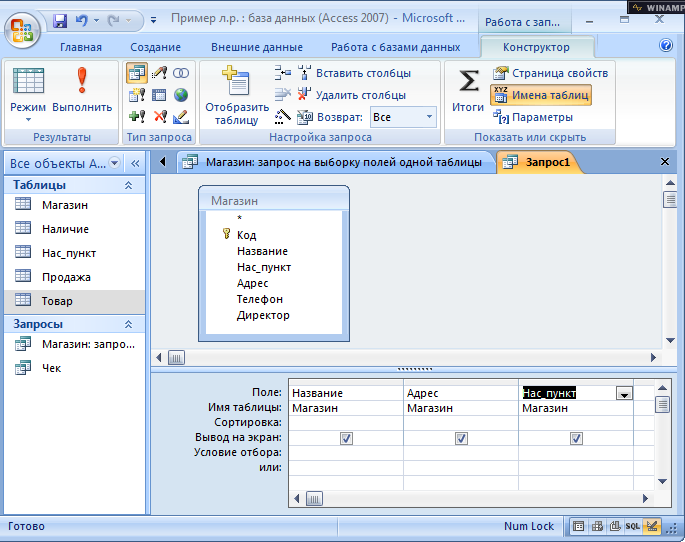


В последнем окне задайте имя запроса и нажмите «Готово». Название запроса должно отражать сущность содержащихся в нем данных: «Адреса магазинов».

Результат:

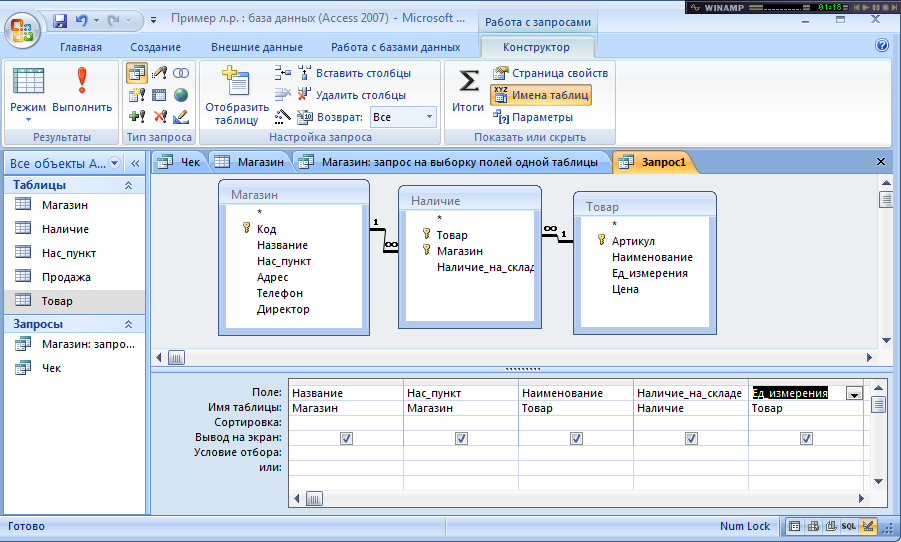


Этот же запрос можно создать или изменить с помощью Конструктора. Запустите конструктор и добавьте таблицу, из которой будете делать запрос. Нужные столбцы отбираются двойным кликом или перетаскиванием.

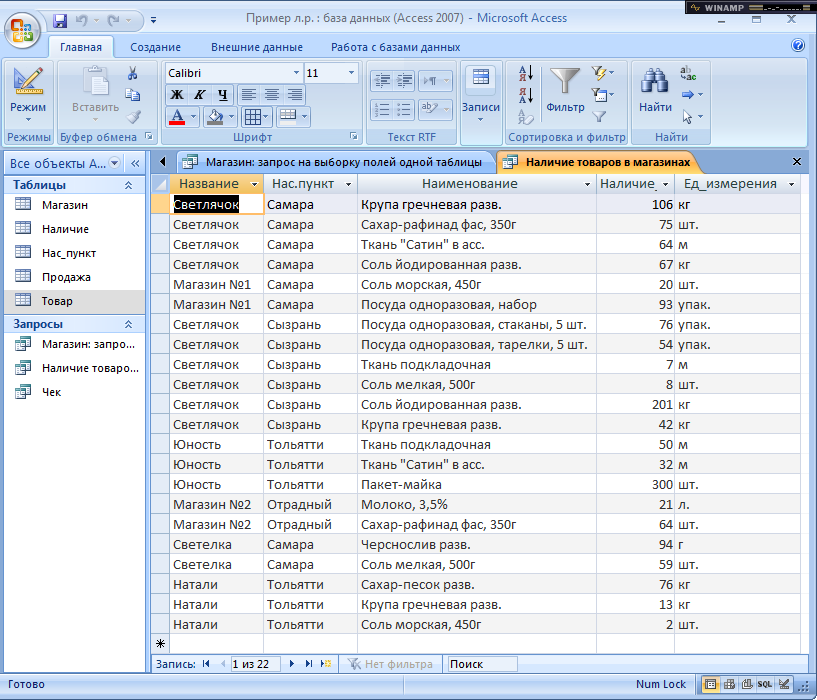


1. *Запрос на выборку полей двух таблиц* выполняется полностью аналогично, за исключением того, что в запрос добавляются поля из двух или более связанных таблиц.

В нашем примере создадим запрос «Наличие товаров в магазинах», отобразим в запросе название магазина и населенный пункт, наименование и ед. измерения товара, количество на складе:



В результате получим:



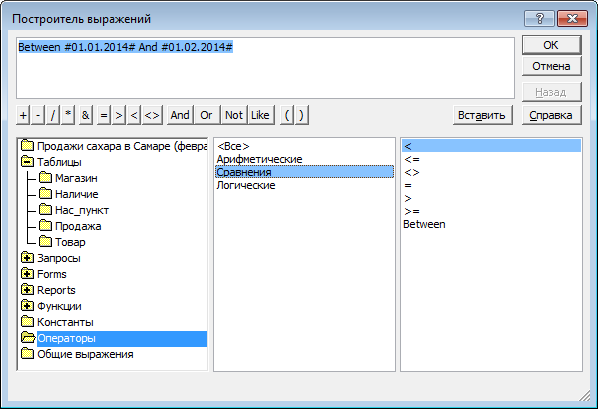
1. *Запрос на выборку с различными критериями отбора* напоминает фильтрацию данных: нам нужно отобразить не все записи, а только удовлетворяющие какому-то критерию.

Выберем магазины в Самаре, в которых в феврале продавался сахар (любой). Также отсортируем запрос по магазинам и количеству проданного сахара. Для этого запроса нам потребуются таблицы «Магазин», «Продажа», «Товар».

Требование: запрос должен состоять минимум из двух условий и возвращать не менее 2 записей! Сортировка обязательно минимум по 1 полю.

Критерии отбора указываются в Конструкторе в строке «Условие отбора». Например, для поля «Нас\_пункт» напишем "Самара" (в кавычках, это текстовая константа).

Для построения сложных условий удобно пользоваться мастером **** во всплывающем меню. В нем можно найти все доступные в запросах данные (из таблиц, других запросов, форм), функции и операторы:



Диапазон дат или значений можно задать с помощью оператора Between. Даты нужно указывать между символами ##.

Between #01.02.2014# And #28.02.2014#

А, например, даты за последние 30 дней можно получить с помощью функции Date() – текущая дата. Напомним, что дата представляется в памяти как число ней, прошедших с 01.01.1990 (=0).

>=Date()-30

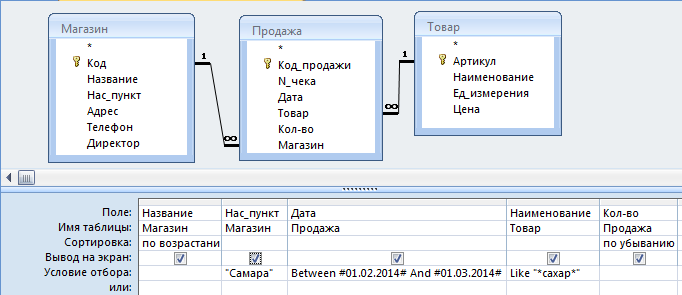
Для выбора текстовых значений используется оператор Like. В образце отбора можно использовать маски: \* обозначает любую последовательность символов, ? один любой символ. Регистр символов не учитывается.

Like "\*сахар\*"

Если бы мы написали в условии отбора просто "Сахар", то ничего бы не нашлось, т.к. у нас нет товара в точности с таким именем.

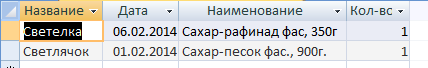
Примеры других запросов вы можете найти в справке  по Access (F1) в разделе «Примеры условий запроса».

Сортировка столбцов указывается в соответствующей строке. Порядок сортировки определяется порядком столбцов в запросе.



Но нужно ли нам отображать город? Мы и так знаем, что в запросе отображаются только магазины из Самары. Поэтому столбец «Нас\_пункт» можно скрыть, сняв флажок «Вывод на экран».

В нашей маленькой БД нашлось только 2 таких записи:



1. *Запрос с вычисляемыми полями*. Вычисления – одна из наиболее распространенных функций запросов. Благодаря им, нам не нужно хранить в базе, например, сумму продажи, мы всегда можем вычислить ее через количество и цену.

Построим запрос «Стоимость продаж» из таблиц «Продажа» и «Товар». Отобразим дату продажи, номер чека, название товара, количество, единицы измерения, цену и сумму.

Первые столбцы добавляются в запрос как обычно.

Формула для вычисления суммы (цена\*кол-во) записывается прямо в строке «Поле». Основные правила:

* сначала указывается название будущего столбца, ставится двоеточие, потом пишется формула:

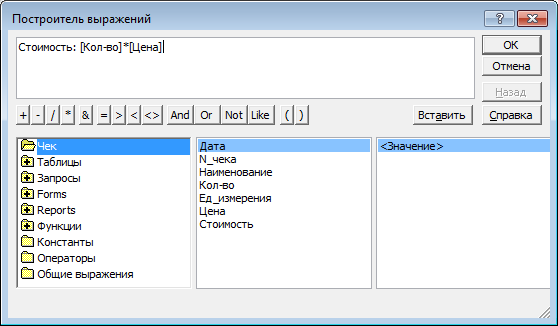
Название: формула

* названия полей таблиц, с которыми выполняются действия, и самих таблиц, пишутся в [], между таблицей и полем ставится восклицательный знак. Название таблицы можно не писать, поле с таким именем встречается только в одной таблице

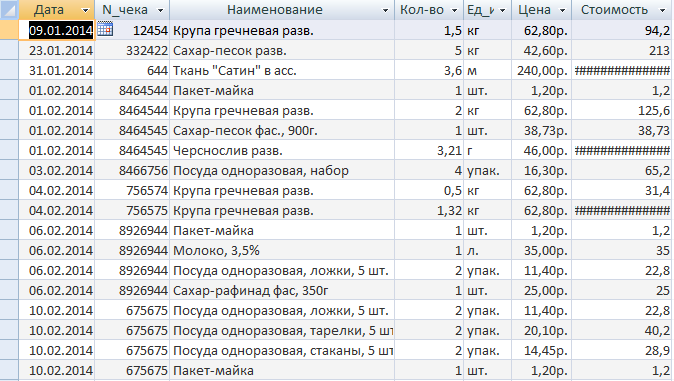
[Таблица]![Поле]

[Поле]

Как и для условий отбора, можно воспользоваться мастером «Построить...».

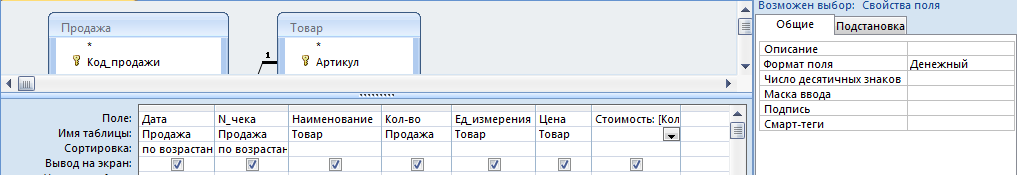


В итоге получим:

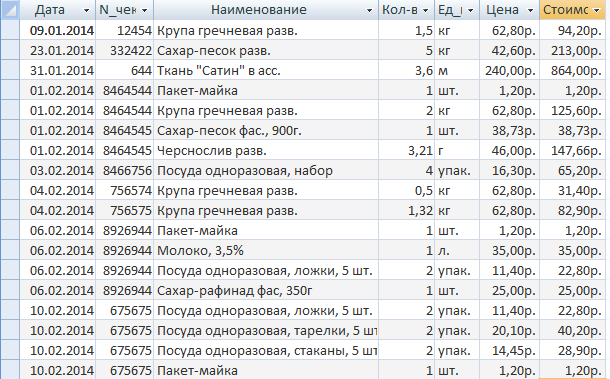


Но в поле сумма должны отображаться денежные единицы! Решетки ##### означают, что число слишком длинное и не умещается в столбец.

Чтобы это исправить, вернемся в Конструктор. Правый клик по полю «Стоимость», пункт меню «Свойства». Справа откроется окно свойств. Зададим формат поля – денежный. Кроме того, не помешает отсортировать запрос по дате продажи и номеру чека.



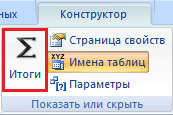
Теперь наш запрос немного похож на настоящие чеки, идущие без перерыва на чековой ленте:



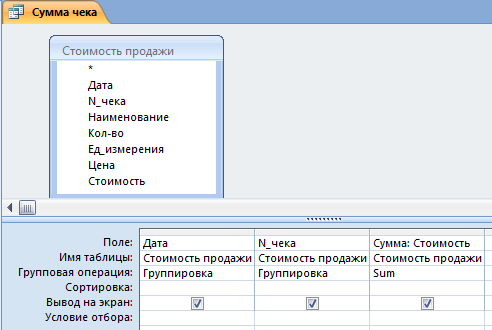
1. *Запрос с итогами (запрос с группировкой)*. В предыдущем примере явно не хватает общей суммы по чеку. К сожалению, вывести ее в том же запросе не получится, нужно создать отдельный. Потом эти запросы можно будет вывести в общем отчете или на одной форме.

Добавим запрос «Сумма по чеку». Чтобы не повторять предыдущие действия, его мы будем строить не на основе таблиц, а на основе прошлого запроса «Стоимость продажи».

Добавим в новый запрос поля «Дата», «№ чека» и «Стоимость». Далее необходимо включить подведение итогов в запросе.



В Конструкторе появится строчка «Группировка». Группировка объединяет совпадающие значения в одну запись. Нам нужно сгруппировать чеки по номеру и дате, а стоимость – просуммировать (**Sum**). Будет лучше переименовать поле «Стоимость» в поле «Сумма\_по\_чеку», это делается также как и для формул – через двоеточие. Сумме чека также нужно присвоить тип «Денежная».



Доступны и другие действия группировки:

**Avg** – среднее значение;

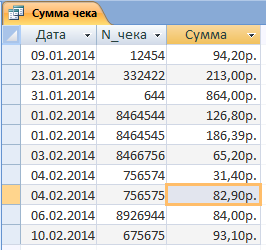
**Min, Max** – минимальное и максимальное значения;

**Count** – подсчет количества сгруппированных записей;

**StDev, Var** – стандартное отклонение и дисперсия;

**First, Last** – первое и последнее по порядку значение.

Результат выполнения запроса:

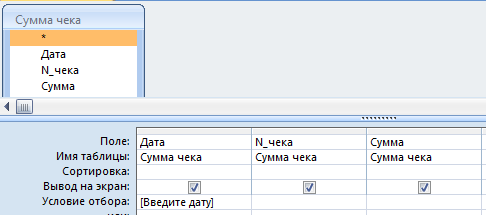


1. *Запрос с параметром*. Не во всех случаях мы заранее знаем параметр отбора строк в запросе. Например, мы хотим выводить чеки за разные даты. Конечно, можно каждый раз открывать Конструктор и исправлять запрос. Но обычные пользователи БД этого не умеют. Они должны просто ввести дату и получить чеки за это число. А составить запрос – задача администратора БД или программиста.

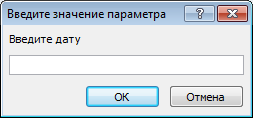
Параметр запроса – это как раз то условие, которое мы будем запрашивать у пользователя.

Параметры указываются в условии запроса в квадратных скобках [] (не путать с именами полей в формулах!). В скобках следует написать пояснение для пользователя – что ему нужно ввести.

[Введите дату]



В результате при выполнении запроса выведется окно:



Введем 01.02.14 и получим:



Параметр может быть частью условия отбора или формулы. Например, найдем все чеки на сумму более или равную указанной пользователем:

>=[Введите минимальную сумму чека:]

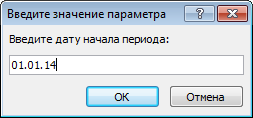
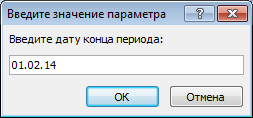
Отбор через оператор Like может выглядеть так:

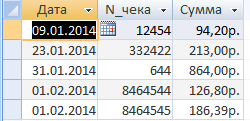
Like "\*"+[Введите подстроку для поиска товара:]+"\*"

Отбор по диапазону дат:

Between [Введите дату начала периода:] And [Введите дату конца периода:]

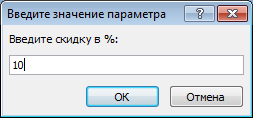
Здесь будет выдано два запроса подряд.

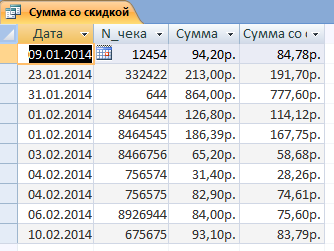
 



Пример использования параметра в формуле - вычисление суммы с произвольной скидкой:

Сумма со скидкой: [Сумма]\*(100-[Введите скидку в %:])/100



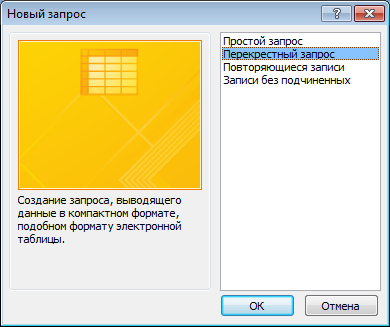


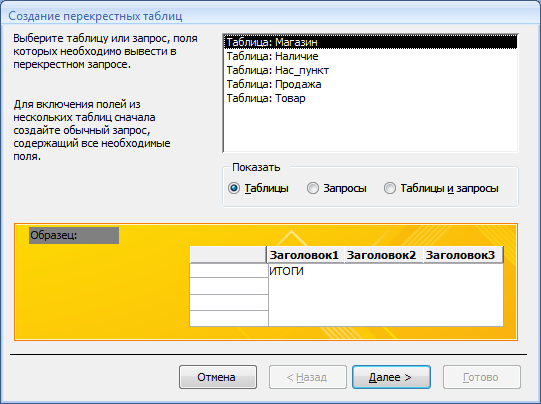
Замечание: вы можете создать запрос с параметром как в условии отбора, так и в формуле.

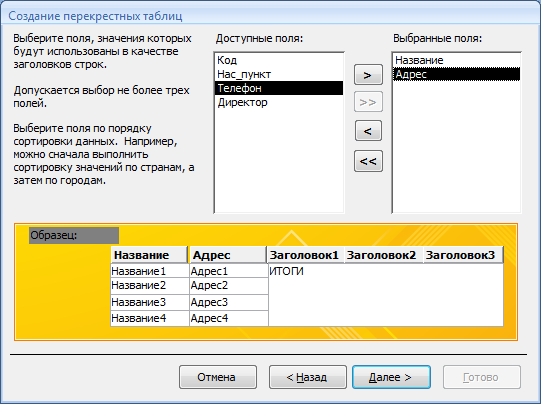
1. *Перекрестный запрос* подобен сводной таблице в Excel. Он позволяет превратить значения из записей в заголовки столбцов.

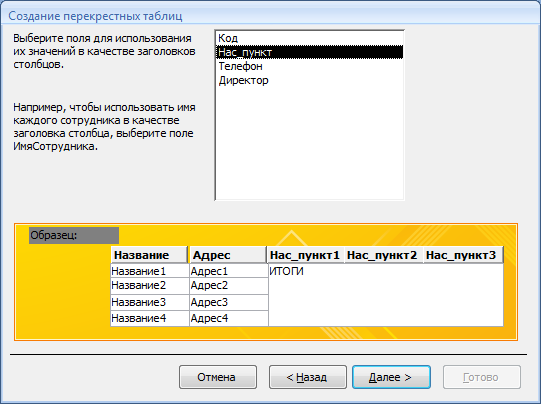
Простые перекрестные запросы – из одной таблицы или запроса – создаются через Мастер.

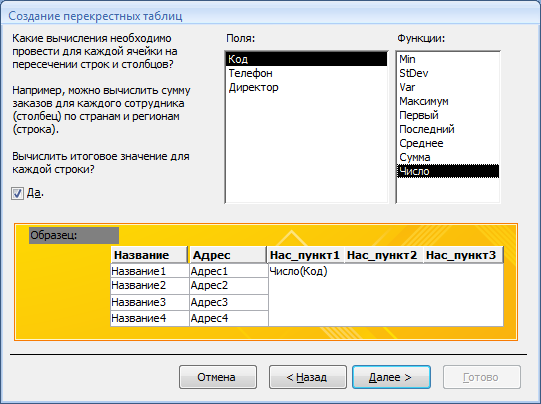
Например, покажем наличие магазинов по адресам в различных населенных пунктах:



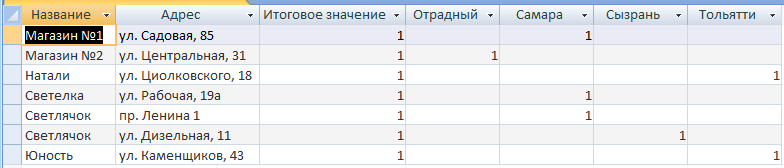








Результат:

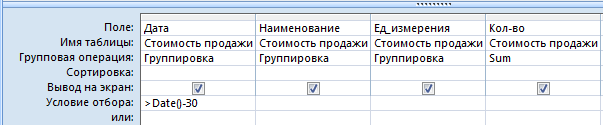


Если бы у нас было несколько магазинов по одному адресу, то в соответствующих ячейках отобразились бы 2, 3 и т.д. Кроме того, Access добавил столбец «Итоговое значение» с указанием общего числа таких магазинов.

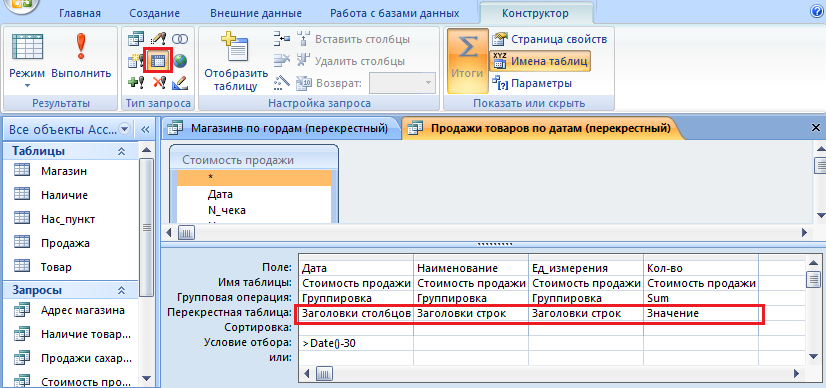
Более сложные запросы (из нескольких таблиц, с вычислениями, условиями отбора) создаются в конструкторе.

Создадим перекрестный запрос, в котором покажем продажи товаров за указанный диапазон дат.

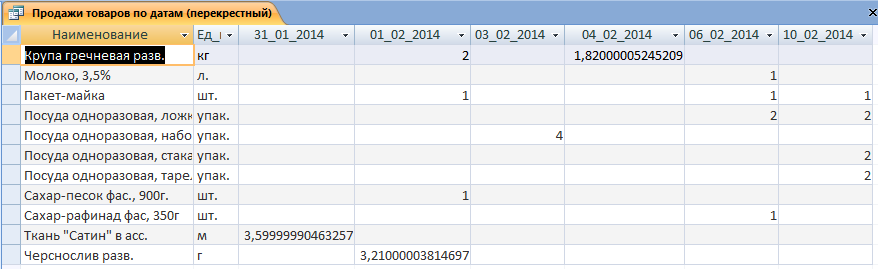
Сначала создадим обычный запрос с группировкой:



Теперь изменим его тип на «перекрестный». Вместо строки «Вывод на экран» появится «Перекрестная таблица». Там нужно указать, какие поля станут заголовками строк, какие – заголовками столбцов, какие – значениями:



Результат:



К сожалению, в перекрестных запросах имеются серьезные проблемы с округлением. Формат поля также не влияет на вывод значений.

В отчет необходимо вставить представление каждого запроса в Конструкторе и результат его выполнения (скриншотом или через экспорт, как для таблиц). Снабдите каждый запрос кратким пояснением – какие именно данные выбраны и из каких таблиц. Если условия запроса или формулы не умещаются на скриншоте, выпишите их отдельно.

# Создание форм

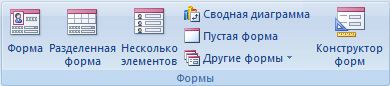
Формы необходимы для более удобного редактирования таблиц и просмотра запросов. Соответственно, различают формы, ориентированные на ввод и редактирование данных, и формы, ориентированные на просмотр.

В данном практикуме мы создадим только формы для просмотра:

* автоформы в один столбец, табличную и ленточную;
* составные формы (подчиненную и связанную).

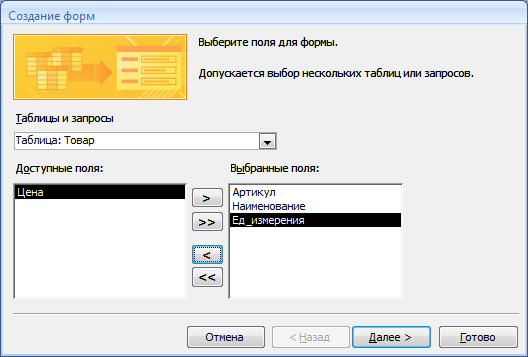
Формы можно создавать на основе таблиц или запросов. Автоформы в один столбец, табличная и ленточная создаются на основе одной таблицы, а составные формы - на основе нескольких связанных таблиц.

Как и запросы, формы можно создавать через Мастер (автоформы) или Конструктор. Некоторые формы вынесены непосредственно на Ленту.

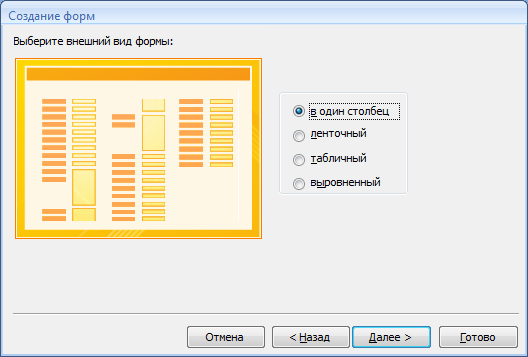


1. Добавим автоформу в один столбец. Это самый простой вид форм, на которой все поля располагаются одно под другим. Отобразим на ней содержимое таблицы «Товар», за исключением цены.

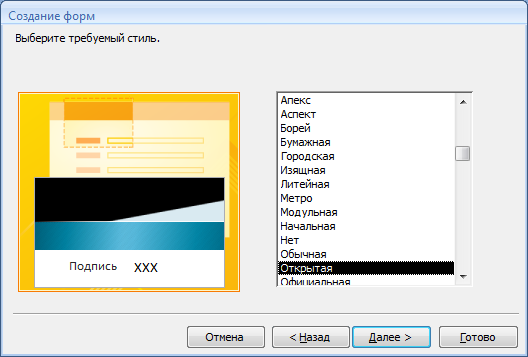
В раскрывающемся списке «Другие формы» выберите «Мастер форм». Выберите нужную таблицу или однотабличный запрос. С помощью кнопок со стрелками переместите выбранные поля в правый столбец.



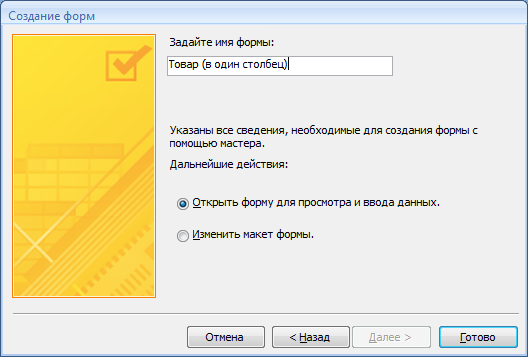
Вид формы - «в один столбец».



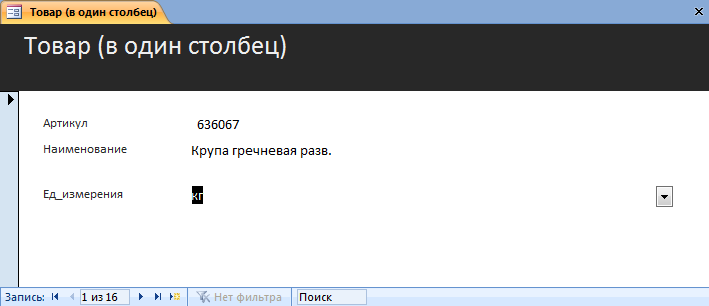
Выберите любой стиль, который Вам нравится:



Задайте имя формы:

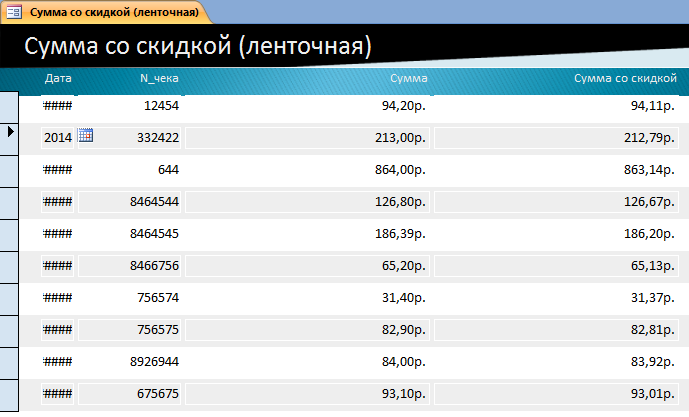


Результат:



На такой форме отображается одна запись «за раз». Переход между записями осуществляется с помощью кнопок внизу. Если форма создана на основе таблицы, ее можно использовать как для просмотра, так и для ввода и редактирования данных.

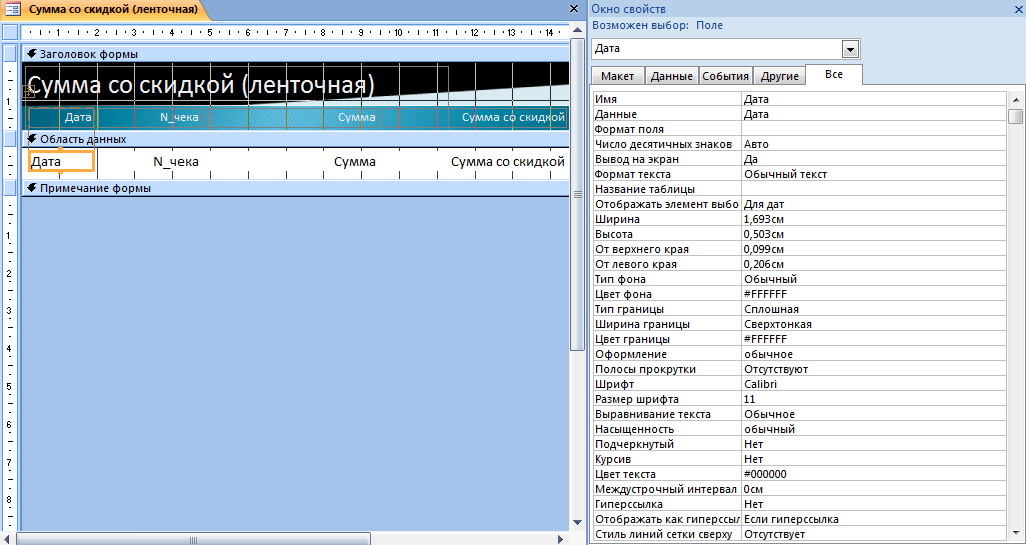
1. Ленточная автоформа создается полностью аналогично. Отличается лишь выбор вида формы на втором шаге и результат. На ленточной форме все записи отображаются вместе, как в таблице:



Данная форма создана на основе запроса «Сумма со скидкой».

Решетки в столбце «Дата» отображаются из-за того, что автоматической ширины столбца недостаточно для отображения даты. В то же время столбцы «Сумма» и «Сумма со скидкой» слишком широкие.

Исправим это в Кострукторе:

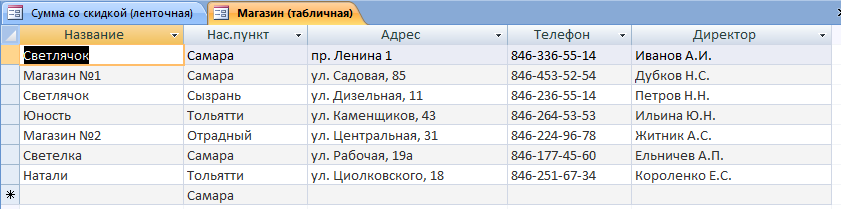


Здесь можно изменить размер и положение любых полей, настроить их свойства (справа).

*Замечание*: используйте Конструктор по мере необходимости, если автоматически созданная форма неудовлетворительна.



1. Табличная автоформа создается также и выглядит полностью как обычная таблица или запрос:



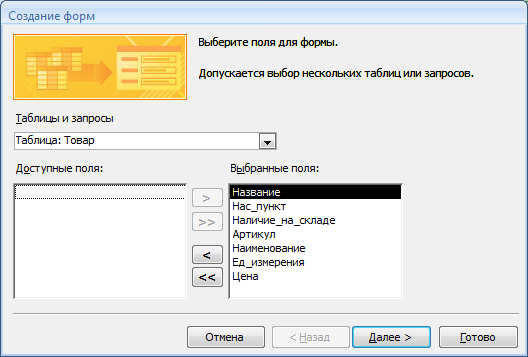
Такие формы используются редко, обычно, как вспомогательные или часть составных форм.

В Конструкторе табличная форма отображается как форма в один столбец.

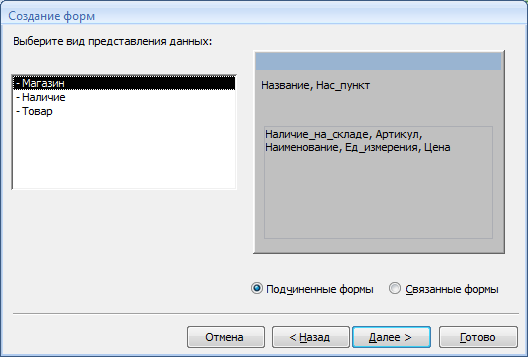
1. Подчиненные формы отображают одновременно несколько связанных таблиц. Их также можно создать с помощью Мастера.

В примере покажем на одной форме товары, имеющиеся в каждом магазине.

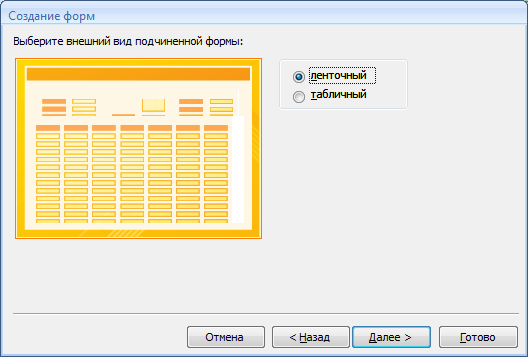
На первом шаге необходимо выбрать поля из нескольких связанных таблиц:



На втором шаге выбираем подчиненную форму и вид ее представления (какая таблица будет основной, какая – подчиненной). Основная форма отображается по одной записи, и одновременно можно увидеть все связанные с данной записи в подчиненной форме.

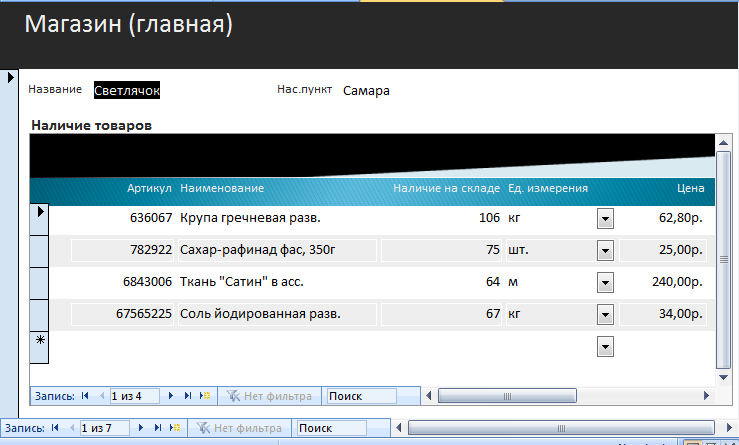


Внешний вид подчиненной формы можно выбрать любой:



Дальнейшие шаги пояснений не требуют.

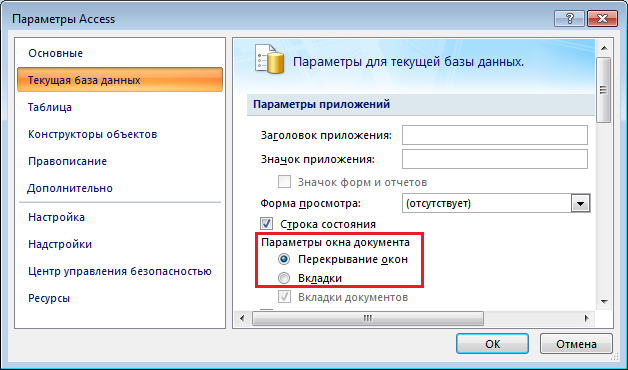
В результате будет создано две формы. Но подчиненная отображается как еще одно поле на главной форме. Размещение элементов на форме потребовалось отредактировать в Конструкторе.



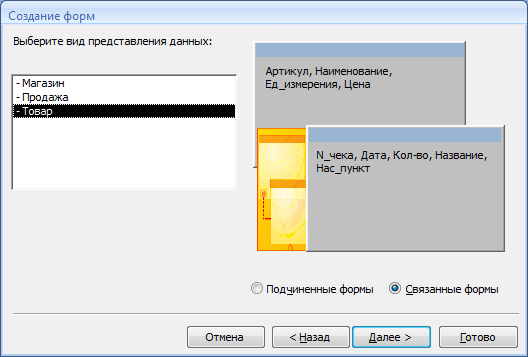
*Замечание*: для свободного перемещения элементов на форме может потребоваться отключить кнопку «Табличный» или «В столбик» на вкладке «Упорядочить».

1. Связанные формы также создаются из нескольких таблиц или подчиненных запросов. При этом на главной форме размещается кнопка, нажатие на которую вызывает подчиненную форму.

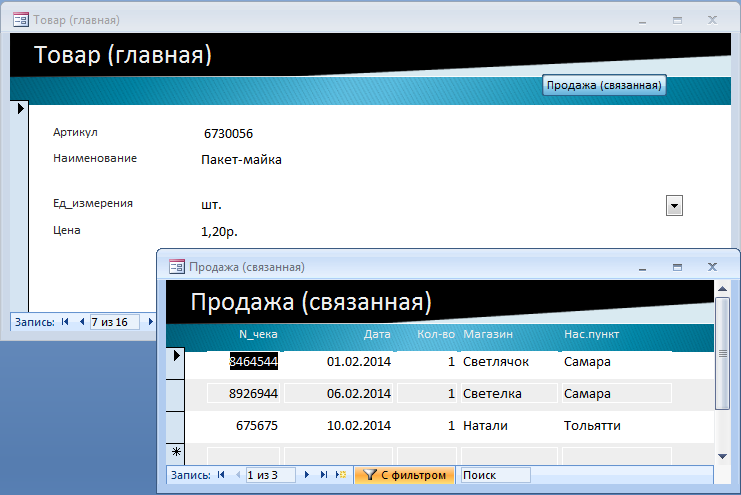
Связанные формы удобнее использовать, изменив настройки Access по умолчанию: формы лучше отображать не на вкладках, а как обычно. Чтобы настройки вступили в силу, нужно перезапустить БД.



Создаются связанные формы аналогично подчиненным, только на втором шаге нужно указать тип формы – связанная.



Связанные формы в примере созданы на основе таблиц «Продажа», «Товар», «Магазин». Вид форм отредактирован в Конструкторе.



Таком образом, мы можем увидеть, где и когда был продан выбранный товар. При перелистывании записей на главной форме, содержимое подчиненной или связанной формы сразу меняется.

В отчет по работе необходимо вставить скриншоты готовых форм. Связанные формы можно отдельными скриншотами.

# Создание отчетов

Отчеты необходимы для вывода сведений из БД на печать. Обычно отчеты содержат итоговые сведения из запросов.

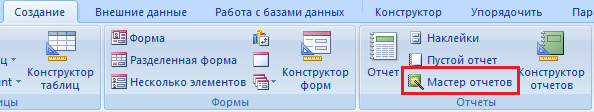
Отчеты, как и формы, и запросы, можно создавать с помощью Мастера отчетов (автоотчеты) или через Конструктор. Любые отчеты можно отредактировать в Конструкторе.

Создадим следующие типы отчетов:

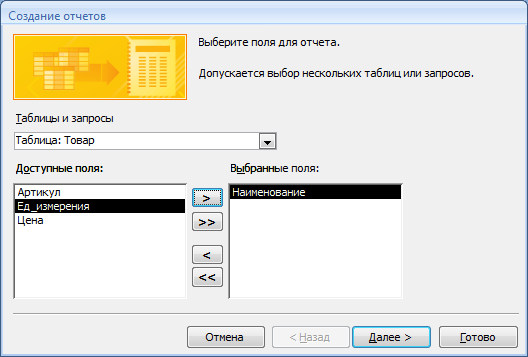
* автоотчет в один столбец;
* отчет по нескольким полям одной таблицы;
* отчет с группировкой и итогами.

1. *Автоотчет в один столбец* создается с помощью Мастера и позволяет вывести список всех значений какого-либо поля. Например, отобразим список всех указанных в БД товаров (наименований) в алфавитном порядке.

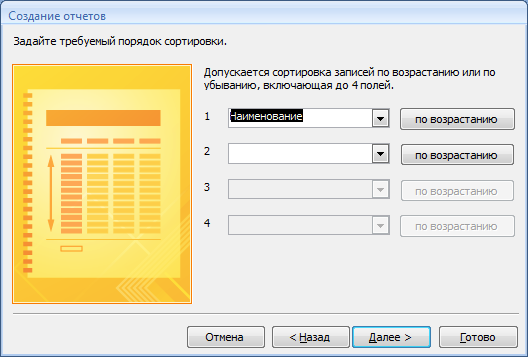
Создадим отчет с помощью Мастера.



В первую очередь, необходимо выбрать поля из таблиц и/или запросов, которые будут отображаться в БД.



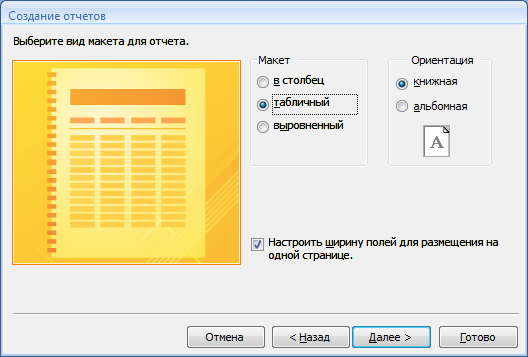
Затем указываем порядок и направление сортировки (если требуется).



Макет отчета можно выбрать любой:

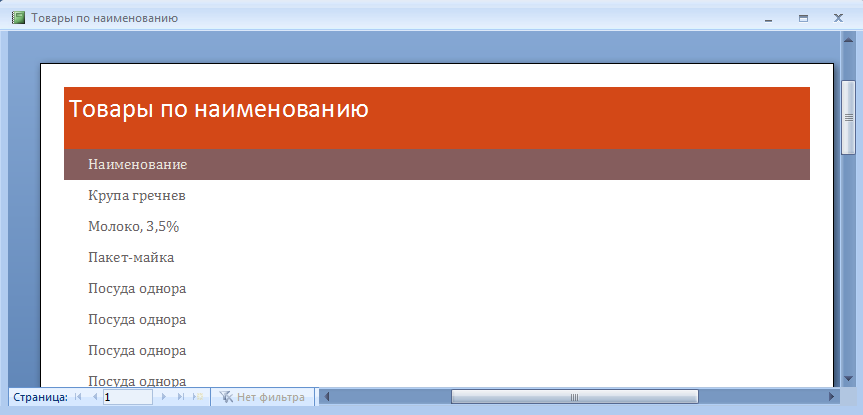
* *табличный* подходит для большинства отчетов;
* *в столбец* выводит данные наподобие формы, подходит для отображения полей с большим содержимым (длинный текст, рисунки);
* *выровненный* стоит использовать в отчетах с большим количеством столбцов, где они имеют разный размер или просто не умещаются в одну строчку таблицы.

Желательно использовать флажок «Настроить ширину полей для размещения на одной странице», иначе часть столбцов может «уехать» на другие страницы.



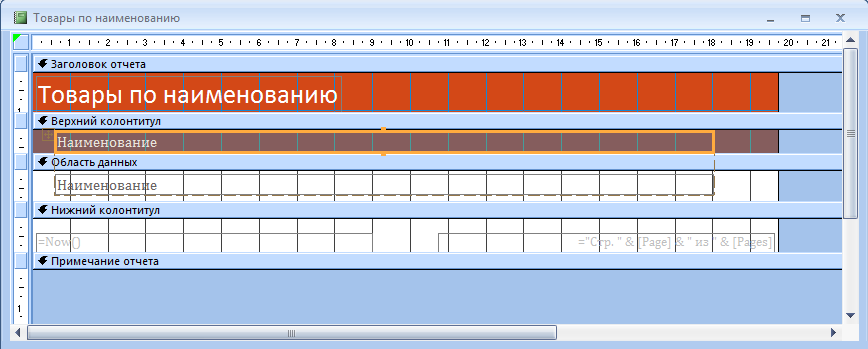
Далее выбирается стиль оформления (в примере выбран стиль «Справедливость») и задается имя отчета (Товары по наименованию). Имя лучше изменить – таблицы БД называются в единственном числе, а отчеты не обязательно.

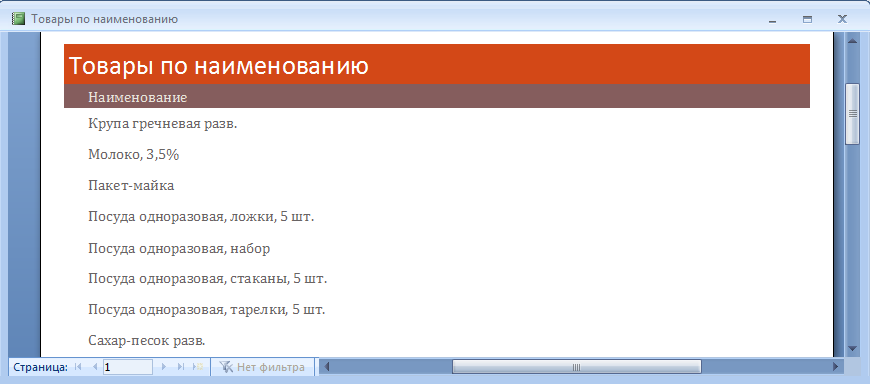
Результат:



Ширина столбца, очевидно, недостаточна. Исправим структуру отчета через Конструктор (кнопка «Закрыть окно предварительного просмотра»).

Поскольку столбец всего один, его можно растянуть на всю ширину страницы.





Полученный отчет можно распечатать из Access или экспортировать в Word или другую программу.

Для добавления в отчет по лабораторной работе, выполните экспорт в Word. В конечном документе исправим шрифт (заголовок напечатан белым цветом), уберем колонтитулы. Чтобы отделить отчеты друг от друга добавьте вокруг них границы (как для таблиц: выделить отчет, кнопка  на вкладке «Главная»).

Результат:

Товары по наименованию

**Наименование**

Крупа гречневая разв.

Молоко, 3,5%

Пакет-майка

Посуда одноразовая, ложки, 5 шт.

Посуда одноразовая, набор

Посуда одноразовая, стаканы, 5 шт.

Посуда одноразовая, тарелки, 5 шт.

Сахар-песок разв.

Сахар-песок фас., 900г.

Сахар-рафинад фас, 350г

Соль йодированная разв.

Соль мелкая, 500г

Соль морская, 450г

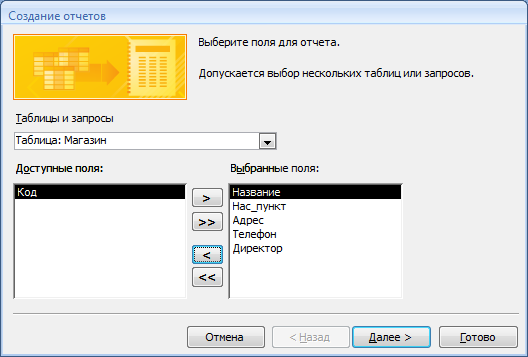
Ткань "Сатин" в асс.

Ткань подкладочная

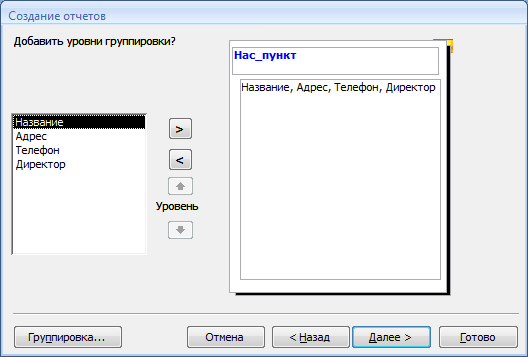
Черснослив разв.

1. В отчете из нескольких полей одной таблицы покажем сведения о магазинах.

В целом, процесс создания отчета аналогичен.



Однако на втором шаге предлагается выбрать группировку записей. Название группы будет выводиться в виде подзаголовка.



В данном случае достаточно логично объединить магазины по населенному пункту, в котором они находятся. В других вариантах можно обойтись без группировки.

Сортировку выполним по адресу.

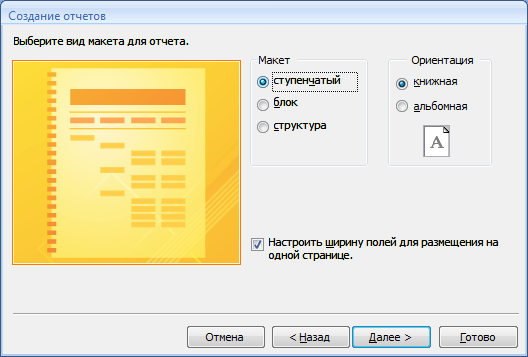
При наличии группировки типы макетов будут другими:

* ступенчатый;
* блок;
* структура.

«Ступенчатый» и «блок» похожи, они представляют собой таблицу, в которой не будут повторяться одинаковые значения записей.

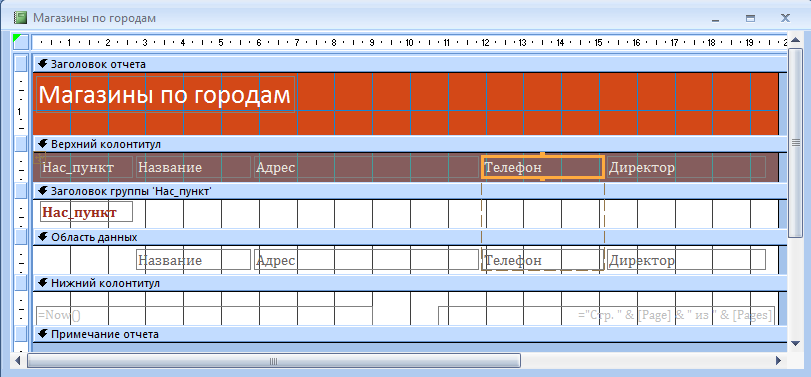
В «структуре» сгруппированные поля отображаются как подзаголовки.

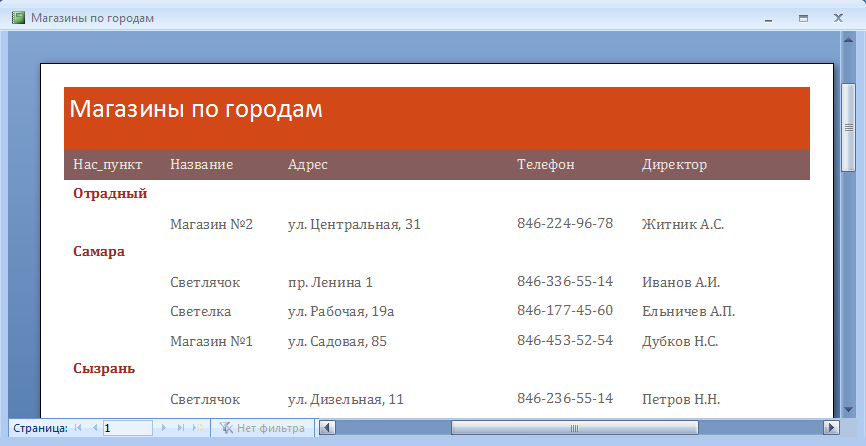
В примере выберем ступенчатый макет.



Стиль – «Справедливость», название отчета «Магазины по городам».

Ширину столбцов также потребовалось отредактировать в Конструкторе, а также поменять порядок столбцов: хоть мы и сортируем магазины по адресу, логично первым столбцом показать название.





Результат экспорта в Word:

Магазины по городам

**Нас\_пункт Название Адрес Телефон Директор**

**Отрадный**

Магазин №2 ул. Центральная, 31 846-224-96-78 Житник А.С.

**Самара**

Светлячок пр. Ленина 1 846-336-55-14 Иванов А.И.

Светелка ул. Рабочая, 19а 846-177-45-60 Ельничев А.П.

Магазин №1 ул. Садовая, 85 846-453-52-54 Дубков Н.С.

**Сызрань**

Светлячок ул. Дизельная, 11 846-236-55-14 Петров Н.Н.

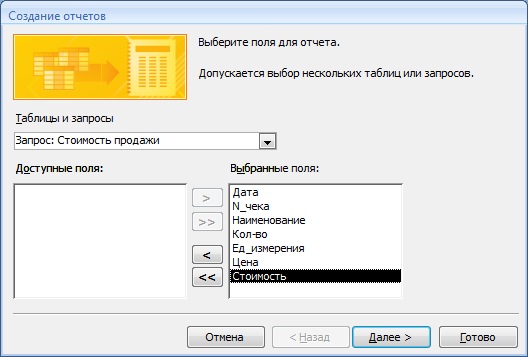
**Тольятти**

Юность ул. Каменщиков, 43 846-264-53-53 Ильина Ю.Н.

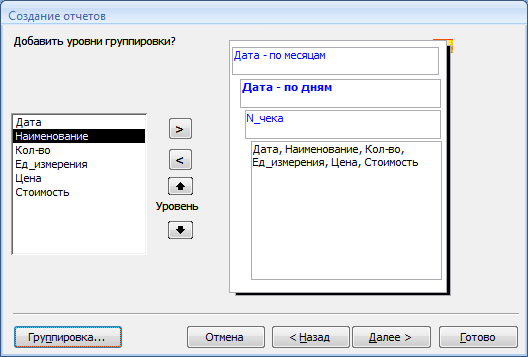
Натали ул. Циолковского, 18 846-251-67-34 Короленко Е.С.

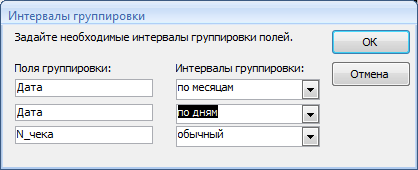
1. Отчет с группировкой и подведением итогов можно сделать на основе нескольких таблиц или запроса. В пример воспользуемся ранее созданным запросом «Стоимость продаж».

В отчет добавим все столбцы запроса.



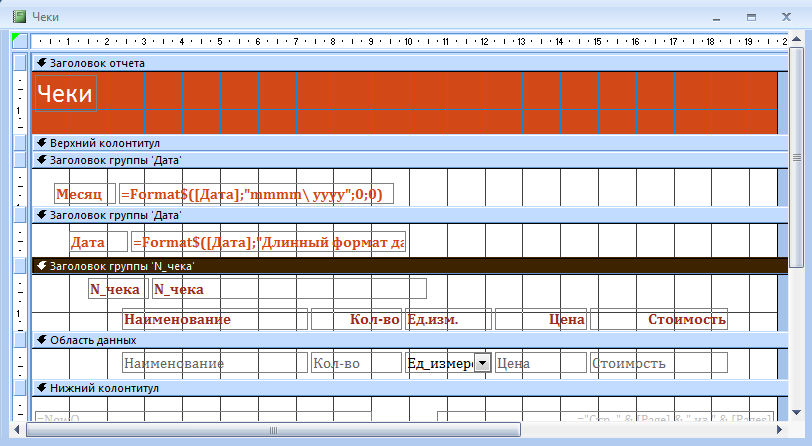
Группировку желательно делать многоуровневой. Для дат может потребоваться дополнительная настройка через кнопку «Груупирвока...». В данном случае по датам сгруппируем дважды: по месяцам и по дням.



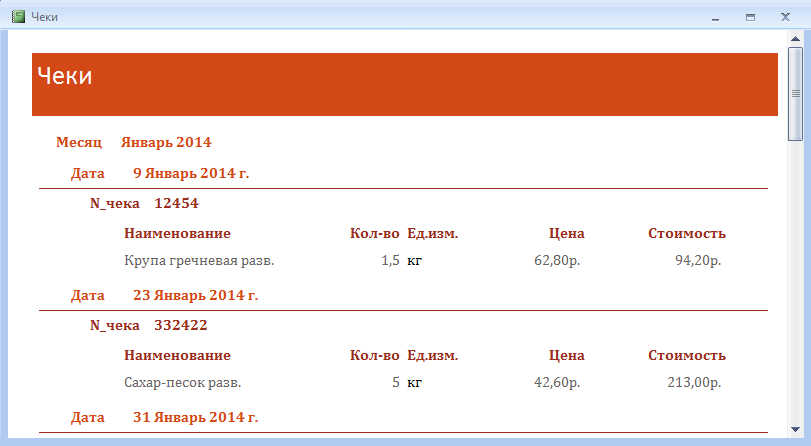


Сортировку в данном случае выполнять не будем. Макет выберем «Структура», стиль также «Справедливость». Название отчета – «Чеки».

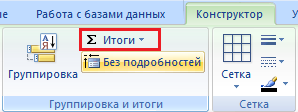
Отображение отчета также потребовалось исправить через Конструктор: убрать лишнее поле «Дата», настроить ширину и расположение столбцов, горизонтальные разделительные линии и т.п.



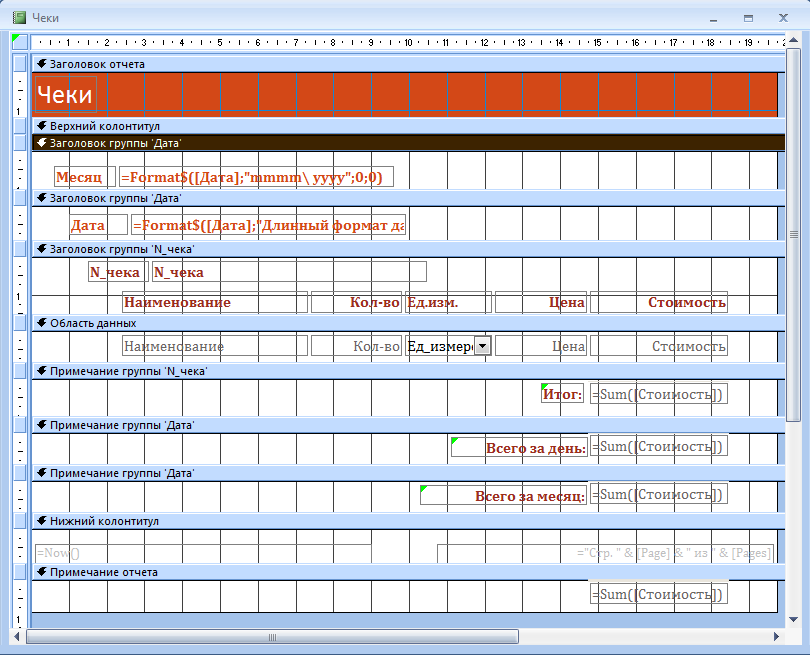
В результате отчет выглядит так:



Теперь необходимо добавить итоги. Это можно выполнить через Конструктор. Итоги будет подводить по сумме чека. Сначала необходимо выделить столбец «Стоимость», затем нажать на кнопку «Итоги».



В результате в макет отчета были добавлены разделы «Примечание группы...», а также «Примечание отчета». Вручную добавим текстовые подписи к каждому итогу: сумма по чеку, за день, за месяц.



Результат имеет вид:



Чеки

**Месяц** **Январь 2014**

**Дата** **9 Январь 2014 г.**

**N\_чека** **12454**

**Наименование** **Кол-во** **Ед.изм.** **Цена** **Стоимость**

Крупа гречневая разв. 1,5 кг 62,80р. 94,20р.

**Итог:** 94,20р.

**Всего за день:** 94,20р.

**Дата** **23 Январь 2014 г.**

**N\_чека** **332422**

**Наименование** **Кол-во** **Ед.изм.** **Цена** **Стоимость**

Сахар-песок разв. 5 кг 42,60р. 213,00р.

**Итог:** 213,00р.

**Всего за день:** 213,00р.

**Дата** **31 Январь 2014 г.**

**N\_чека** **644**

**Наименование** **Кол-во** **Ед.изм.** **Цена** **Стоимость**

Ткань "Сатин" в асс. 3,6 м 240,00р. 864,00р.

**Итог:** 864,00р.

**Всего за день:** 864,00р.

**Всего за месяц:** 1 171,20р.

**Месяц** **Февраль 2014**

**Дата** **1 Февраль 2014 г.**

**N\_чека** **8464544**

**Наименование** **Кол-во** **Ед.изм.** **Цена** **Стоимость**

Пакет-майка 1 шт. 1,20р. 1,20р.

Крупа гречневая разв. 2 кг 62,80р. 125,60р.

**Итог:** 126,80р.

**N\_чека** **8464545**

**Наименование** **Кол-во** **Ед.изм.** **Цена** **Стоимость**

Сахар-песок фас., 900г. 1 шт. 38,73р. 38,73р.

Черснослив разв. 3,21 г 46,00р. 147,66р.

**Итог:** 186,39р.

**Всего за день:** 313,19р.

**Дата** **3 Февраль 2014 г.**

**N\_чека** **8466756**

**Наименование** **Кол-во** **Ед.изм.** **Цена** **Стоимость**

Посуда одноразовая, набор 4 упак. 16,30р. 65,20р.

**Итог:** 65,20р.

**Всего за день:** 65,20р.

**Дата** **4 Февраль 2014 г.**

**N\_чека** **756574**

**Наименование** **Кол-во** **Ед.изм.** **Цена** **Стоимость**

Крупа гречневая разв. 0,5 кг 62,80р. 31,40р.

**Итог:** 31,40р.

**N\_чека** **756575**

**Наименование** **Кол-во** **Ед.изм.** **Цена** **Стоимость**

Крупа гречневая разв. 1,32 кг 62,80р. 82,90р.

**Итог:** 82,90р.

**Всего за день:** 114,30р.

**Дата** **6 Февраль 2014 г.**

**N\_чека** **8926944**

**Наименование** **Кол-во** **Ед.изм.** **Цена** **Стоимость**

Пакет-майка 1 шт. 1,20р. 1,20р.

Молоко, 3,5% 1 л. 35,00р. 35,00р.

Посуда одноразовая, 2 упак. 11,40р. 22,80р.

Сахар-рафинад фас, 350г 1 шт. 25,00р. 25,00р.

**Итог:** 84,00р.

**Всего за день:** 84,00р.

**Дата** **10 Февраль 2014 г.**

**N\_чека** **675675**

**Наименование** **Кол-во** **Ед.изм.** **Цена** **Стоимость**

Посуда одноразовая, 2 упак. 11,40р. 22,80р.

Посуда одноразовая, 2 упак. 20,10р. 40,20р.

Посуда одноразовая, 2 упак. 14,45р. 28,90р.

Пакет-майка 1 шт. 1,20р. 1,20р.

**Итог:** 93,10р.

**Всего за день:** 93,10р.

**Всего за месяц:** 669,79р.

1 840,99р.

В отчет по лабораторной работе необходимо добавить экспортированные отчеты из Access. Скриншоты необязательны.

Допускается отчеты выносить в отдельный раздел с более узкими полями, а также уменьшать межабзацные отступы для экономии места.

# Варианты заданий

**1**. Создать базу данных «Сведения о жителях города». В базе данных хранятся сведения о жителях города Самары (номер паспорта, фио, дата рождения, телефон, пол, категория (дошкольник, школьник, студент, служащий, пенсионер)), о месте их работы (код организации, название организации, телефон, должность), о жилье (почтовый индекс, название района, название улицы, номер дома, номер квартиры). Один житель может иметь в собственности несколько домов и работать в нескольких организациях. У одного вида жилья может быть несколько владельцев.

**2.** Создать базу данных «Адресная книга». В базе данных хранятся сведения о себе (номер паспорта, фио, адрес, телефон), о родственниках (фио, адрес, телефон, степень родства, дата рождения), о друзьях и знакомых (фио, адрес, телефон, место работы, должность). Все сведения ориентированы на одного человека, поэтому друзей и родственников может быть много, но только для одного.

**3.** Создать базу данных «Аптека». В базе данных хранятся сведения о лекарстве (название, категория, производитель, поставщик, форма выпуска, вид упаковки, количество в упаковке, дозировка, дата поступления, срок годности), об аптеке (номер аптеки, адрес, телефон, компания, время работы), о наличии лекарств (количество, цена). Одно лекарство может продаваться в разных аптеках по разным ценам.

**4.** Создать базу данных «Студенческая библиотека». В базе данных хранятся сведения о книгах (инвентарный номер, фио автора, название, место издания), о читателях (номер читательского билета, фио, номер группы, телефон, адрес), о разделах библиотеки (название раздела (учебный или научный абонемент, номер комнаты). В каждом разделе библиотеки имеется много книг, одна и та же книга хранится в одном разделе. Книга может быть выдана только одному читателю, один читатель может получить несколько книг.

**5.** Создать базу данных «Биржа». В базе данных хранятся сведения о акциях предприятий (наименование предприятия, адрес предприятия, цене акции, количестве акций к продаже, величине контрольного пакета), о покупателях (фио директора или частного лица, адрес, телефон, почтовый ящик), о сделке (купленное количество акций, дата сделки). Один и тот же покупатель может участвовать во многих сделках, акции одного предприятия могут быть куплены несколькими покупателями.

**6.** Создать базу данных «Больница». В базе данных хранятся сведения о больных (номер паспорта, фио, адрес, дата рождения, номер полиса, дата поступления, дата выписки, диагноз), о палатах (номер палаты, отделение, фио лечащего врача), о назначениях (лекарство, дозировка, количество, периодичности, вид процедуры). Врач одному пациенту может сделать несколько назначений, в одной палате может быть несколько больных, один врач может обслуживать несколько палат.

**7.** Создать базу данных «Бытовое обслуживание населения». В базе данных хранятся сведения о потребителе услуг (фио клиента, адрес, телефон), о перечне услуг (номер квитанции, вид услуги, описание услуги, единица измерения, цена за единицу, дата заказа, время заказа, скидка, дата выполнения, время выполнения), о поставщике услуг (фио поставщика услуг, квалификация). Один поставщик услуг может выполнять несколько видов услуг, один вид услуг может выполняться несколькими поставщиками. Один потребитель услуг может сделать заявки на несколько услуг.

**8.** Создать базу данных «Дума». В базе данных хранятся сведения о депутатах (фио, номер паспорта, дата рождения, округ, должность оклад, участие в заседаниях), о фракциях (название, лидер, телефон, адрес), о партии (название, лидер, телефон, адрес). Каждый депутат входит в одну из партий. В одну партию могут входить несколько депутатов. В одну фракцию может входить несколько партий, каждая партия входит в одну фракцию.

**9.** Создать базу данных «Перемещения кадров предприятия». В базе данных хранятся сведения о работниках предприятия (номер паспорта, фио, дата рождения, адрес, домашний телефон), об отделах предприятия (название, фио начальника отдела, телефон, местоположение на предприятии), о назначениях и перемещениях (дата, должность, основание, номер приказа). Один работник может трудиться в текущий момент только в одном отделе предприятия, в одном отделе работает много сотрудников, один сотрудник может несколько раз переходить из отдела в отдел с повышением или понижением должности.

**10.** Создать базу данных «Повышение квалификации сотрудников». В базе данных хранятся сведения о сотрудниках предприятия (номер паспорта, фио, дата рождения, адрес, домашний телефон, рабочий телефон, должность), о повышении квалификации сотрудника (номер свидетельства о пройденном повышении квалификации, продолжительность в днях, место, специальность, год), об аттестации (номер свидетельства аттестации, год переаттестации, специальность, результат). Один сотрудник может неоднократно за время работы на предприятии проходить переаттестацию и повышать квалификацию.

**11.** Создать базу данных «Преподаватели кафедры». В базе данных хранятся сведения о преподавателях (номер паспорта, фио, должность, ученое звание, ученая степень), о курсах (название курса, вид занятий (лекции, практика, лабораторные, курсовой проект), номер семестра, вид отчетности), о группах (номер группы, специальность, количество студентов). Преподаватель может читать несколько курсов с разными видами занятий. Один курс с определенным видом занятий в определенном семестре читается одним преподавателем. В одной группе изучается несколько курсов, один курс может читаться в нескольких группах.

**12.** Создать базу данных «Труды кафедры». В базе данных хранятся сведения о преподавателях (номер паспорта, фио, ученое звание, ученая степень), о трудах (название, место издания, тираж, объем в печатных листах, год издания, цена). Один труд может быть подготовлен несколькими авторами, при этом задается процент участия в изданном труде каждого соавтора. Один преподаватель может подготовить несколько трудов.

**13.** Создать базу данных «Квартплата». В базе данных хранятся сведения о квартиросъемщике (номер паспорта, фио, адрес, площадь, количество проживающих, наличие льгот), о потреблении (номер квитанции, год, месяц, вид платежа (газ, электроэнергия, водоснабжение, отопление, горячая вода, канализация), дата оплаты, размер оплаты), о тарифах на одного человека (год, месяц, газоснабжение, электроснабжение, водоснабжение, отопление, горячая вода, канализация). Квартиросъемщик оплачивает квартплату ежемесячно, тарифы могут меняться многократно.

**14.** Создать базу данных «Конференция». В базе данных хранятся сведения об участниках конференции (номер паспорта, фио, дата рождения, организация, адрес, телефон, ученое звание, ученая степень), о комитетах (название комитета, фио руководителя, описание помещения, телефон), об оплате (номер квитанции, дата оплаты, сумма, способ оплаты). Участник конференции может входить только в один комитет, в одном комитете присутствует несколько участников. Участник может оплачивать частями несколько раз. В одной квитанции об оплате может быть оплачено участие в конференции нескольких участников.

**15.** Создать базу данных «Продажа недвижимости». В базе данных хранятся сведения о квартирах (район, улица, номер дома, номер квартиры, тип дома, этаж, общая площадь, жилая площадь, коэффициент комфортности), о покупателях (номер паспорта, фио, телефон, место работы, должность), о продавцах (номер паспорта, фио, телефон), о сделках (номер договора, дата, стоимость продажи). Один покупатель может совершить покупку нескольких квартир, один продавец может продавать жилье неоднократно.

**16.** Создать базу данных «Лекарства». В базе данных хранятся сведения о лекарстве (название, категория, форма выпуска, упаковка, количество в упаковке, дозировка, срок годности, цена), о производителях (название фирмы, адрес, телефон, фио представителя), о поставщиках (название фирмы поставщика, адрес, телефон, фио поставщика). Один производитель выпускает много лекарственных препаратов. Одно лекарство может быть выпущено только одним производителем, иначе изменится его название. Поставщик может поставить несколько лекарственных препаратов. Одно лекарство может поставляться несколькими поставщиками, при этом факт поставки уточняется количеством и датой поступления.

**17.** Создать базу данных «Нагрузка кафедры». В базе данных хранятся сведения о преподавателе (фио, должность, плановая нагрузка в часах), о нагрузке преподавателя (предмет, количество студентов, вид отчетности (зачет, экзамен), объем лекционных, лабораторных, практических, курсовых, дипломных часов), о нагрузке кафедры (факультет, предмет, количество студентов, объем лекционных, лабораторных, практических, курсовых, дипломных часов). На кафедре работает несколько преподавателей, один преподаватель ведет несколько предметов, один предмет может читаться несколькими преподавателями с учетом их должности.

**18.** Создать базу данных «Начисление зарплаты». В базе данных хранятся сведения о работниках (фио, номер паспорта, разряд, должность, надбавка, льгота на налоги), о штатном расписании (должность, количество должностей, оклад), о налогах (вид налога, величина налога), о зарплате (дата, сумма). Каждый работник получает зарплату с учетом оклада по штатному расписанию, надбавок, налогов.

**19.** Создать базу данных «Поликлиника». В базе данных хранятся сведения о врачах (фио, дата рождения, специальность, стаж по специальности), о днях приема врачей (день приема, время начала, время окончания приёма), о посетителях (фио, номер полиса, адрес, пол, дата рождения), о посещениях пациентов врачей (дата, время, диагноз). Один врач может принять нескольких пациентов, один пациент посетить несколько врачей.

**20.** Создать базу данных «Путевки». В базе данных хранится информация о путевках (код путевки, место отдыха, начало, окончание, количество, стоимость, льготы), о заявлениях (фио, дата подачи, дата рассмотрения, место работы, должность, стаж, результат рассмотрения ), о получателе (фио, код путевки, количество). Получатель может взять несколько путевок, на одну путевку может быть несколько заявлений. В заявлении указывается только одна путевка.

**21.** Создать базу данных «Расписание». В базе данных хранятся сведения о преподавателях (фио, должность), о парах (номер пары, начало1, окончание1, начало2, окончание2), о расписании (неделя (первая или вторая), день недели, номер пары, группа, вид занятий, аудитория), о предметах (название предмета). В расписании учитывается предмет и преподаватель, его ведущий.

**22.** Создать базу данных «Рецепты приготовления блюд». В базе данных хранятся сведения о рецептах (название рецепта, описание, тип пищи, время приготовления, количество порций, калорийность), о компонентах рецепта (название компонента, количество, цена, единица измерения, калорийность). Один компонент может входить в несколько рецептов, в одном рецепте может быть несколько компонент.

**23.** Создать базу данных «Назначение стипендии». В базе данных хранятся сведения о студентах (номер зачетки, фио, дата рождения, номер группы), о сессии (год, семестр, количество зачетов, количество экзаменов, дата завершения сессии), о результатах сессии (вид отчетности (экзамен1, экзамен2, зачет1, зачет2 и т.д.), название дисциплины, результат сдачи (оценка или признак сдачи зачета, дата сдачи экзамена или зачета), о стипендии (номер позиции в стипендиальном приказе, величина стипендии). Один студент сдает несколько сессий, одну сессию сдает несколько студентов, стипендия назначается каждому студенту, сдавшему сессию в срок.

**24.** Создать базу данных «Сведения о сессии». В базе данных хранятся сведения о студентах (номер зачетки, фио, группа), о преподавателях (фио, кафедра, должность), о предметах, сдаваемых в данную сессию (название предмета, вид отчетности (экзамен, зачет, зачет с оценкой). Студенты сдают в сессию несколько предметов, один предмет сдает несколько студентов, преподаватель может принимать в сессию несколько предметов, один предмет в данную сессию принимает один преподаватель.

**25.** Создать базу данных «Склад». В базе данных хранятся сведения о поставщиках товаров на склад (номер поставщика, банковские реквизиты, адрес, телефон), о получателях товаров на склад (номер получателя, банковские реквизиты, адрес, телефон), о приходе на склад товара (номер накладной прихода, наименование товара, описание, ед. измерения, количество, цена, дата поступления), о расходе со склада (номер накладной расхода, наименование товара, о количестве, дате расхода). Поставщик может поставлять несколько видов товаров, товар может получать несколько получателей.

**26.** Создать базу данных «Спорт». В базе данных хранятся сведения о сотрудниках спортивного комплекса (фио, адрес, телефон, дата рождения, должность, стаж работы, образование, оклад), о клиентах (фио, адрес, дата рождения, пол, место работы, количество занятий, телефон), о занятиях (вид занятий, дата начала занятий, количество занятий, цена), о спорткомплексе (название адрес, телефон, фио директора). Один клиент может заниматься по разным видам спорта, причем занятия одного вида могут проводиться различными преподавателями.

**27.** Создать базу данных «Телефонные переговоры». В базе данных хранятся сведения об абонентах (номер телефона абонента, фио, адрес, льготы), о переговорах (код города, дата, время, количество минут, дата оплаты), о кодах города и тарифах (код города, город, тариф за одну минуту), о тарифных коэффициентах (начало периода, конец периода, коэффициент). Один абонент может звонить неоднократно. Стоимость переговоров учитывает город и период времени звонка, так как стоимость звонка зависит от времени суток звонка.

**28.** Создать базу данных «Товары». В базе данных хранятся сведения о товарах (код товара, наименование, цена, количество, скидка, дата поступления), о магазинах (наименование, адрес, телефон, фио директора), о фирмах, привозящих товар (наименование, адрес, телефон, фио директора). Фирма может поставлять несколько товаров в различные магазины.

**29.** Создать базу данных «Фототека». В базе данных хранятся сведения о пленках (код пленки, цена, чувствительность, тип пленки (цветная, негативная), количество кадров, дата начала съемки, дата проявления, место проявления, место хранения), о кадрах (дата съёмки, место съемки, тема, участники), о фотографиях (размер, бумага, количество, фио изготовителя, цена, место нахождения). В фототеке ведется учет изготовленных фотографий с учетом пленки и кадра. Из одной пленки может быть напечатано несколько кадров и для каждого кадра несколько фотографий.

**30.** Создать базу данных «Футбольный турнир». В базе данных хранятся сведения о команде(название, общество, фио главного тренера, фио директора), об игроках (фио, номер, амплуа, возраст), о матчах (номер матча, название команды1, название команды2, дата матча, место, время, стадион, результат, количество зрителей, цена билета), о составе на игру (начало игры, окончание игры). В данной игре могут играть разные игроки, то есть состав игроков команды в данную игру может меняться. Одна команда может участвовать в нескольких матчах, один игрок может входить в одну команде, в одной команде несколько игроков.

**31.** Создать базу данных «Личное имущество». В базе данных хранятся сведения об имуществе(название, описание, стоимость, дата приобретения, место нахождения, принадлежность), о ремонте (вид ремонта, описание, стоимость, организация, фио мастера, дата ремонта), о категориях имущества(номер, название, описание). Категория имущества включает несколько конкретных наименований имущества, имущество может неоднократно подвергаться ремонту.