



Кафедра цифровых технологий  
и прикладной информатики

Б1.О.18 ИНФОРМАЦИОННЫЕ СИСТЕМЫ И ТЕХНОЛОГИИ

**Лабораторные работы. Проектирование логической модели базы  
данных**

МЕТОДИЧЕСКИЕ УКАЗАНИЯ

Направление подготовки  
09.03.03 ПРИКЛАДНАЯ ИНФОРМАТИКА

Квалификация (степень) выпускника  
бакалавр

Уфа 2023

Рекомендовано к изданию методической комиссией экономического  
факультета (протокол № 7 от 23.03.2023 г.)

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

Рецензент: ст. преп. Иванова Г.Р.

Ответственный за выпуск: зав. кафедрой ЦТиПИ  
доцент А.С.Беляева

г. Уфа, БГАУ, Кафедра цифровых технологий и прикладной информатики

## **Оглавление**

Лабораторная работа №1

Проектирование логической модели предметной области в  
ERWIN

Лабораторная работа №2

Создание логической модели базы данных в MS Visio

Библиографический список

Приложение А

Приложение Б

Приложение В

Приложение Г

## Лабораторная работа №1

### Проектирование логической модели предметной области в ERWIN

**Цель работы:** Изучить приемы проектирования и нормализации БД

**Требования к организации рабочего места:** Лабораторная работа должна проводиться в компьютерном классе с установленным приложением ERWIN.

## 1 Основные элементы информационной модели логического уровня

### 1.1 Сущности и атрибуты

**Сущность** – это множество **реальных** или **абстрактных объектов** (людей, предметов, документов и т.п.), **обладающих общими атрибутами или характеристиками**. Любой объект системы может быть представлен только одной сущностью, которая должна быть уникально идентифицирована. *Именованная сущность* осуществляется с помощью *существительного в единственном числе*. При этом имя сущности должно отражать **тип** или **класс** объекта, а не его **конкретный экземпляр** (например, **Студент**, а не **Петров**) (рис. 1).

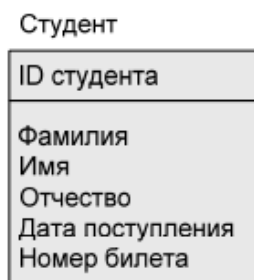


Рисунок 1 - Графическое представление сущности «Студент»

Любая сущность характеризуется набором атрибутов (свойств).

**Атрибут сущности** – характеристика сущности, то есть свойство реального объекта. Например, на рис. 1 атрибутами сущности «Студент» являются «ID студента», «Фамилия», «Имя», «Отчество», «Дата поступления» и «Номер билета».

В свою очередь, атрибуты сущности делятся на 2 вида: собственные и наследуемые. Собственные атрибуты являются уникальными в рамках модели. Наследуемые атрибуты передаются от сущности-родителя при установке связи с другими сущностями.

**Первичный ключ (Primary Key, PK).** Каждая сущность должна обладать атрибутом или комбинацией атрибутов, чьи значения однозначно определяют каждый экземпляр сущности. Эти атрибуты образуют первичный ключ сущности.

**Внешний ключ (Foreign Key, FK).** Если между двумя сущностями имеется специфическое отношение связи или категоризации, то атрибуты, входящие в первичный ключ родительской или общей сущности, наследуются

в качестве атрибутов сущностью-потомком или категориальной сущностью соответственно. Эти атрибуты и называются внешними ключами.

### *1.2 Отношения в IDEF1X-модели*

При построении информационной модели различают следующие типы отношений между сущностями: идентифицирующее, не идентифицирующее, не специфическое (многие-ко-многим) и отношения категоризации.

**Идентифицирующей связью** называется связь, которая добавляет признаки идентичности в дочернюю сущность путем миграции ключей родительской сущности в область ключевых атрибутов дочерней и, таким образом, делая дочернюю сущность зависимой от родительской в смысле своей идентичности.

Можно задать также и такую связь, которая не ставит дочернюю сущность в зависимость от родительской. Этот тип связи называется **неидентифицирующей** связью. Например, в ERwin такая связь обозначается пунктирной линией с жирной точкой на конце, соответствующем дочерней связи. При неидентифицирующей связи атрибуты первичного ключа родительской сущности мигрируют в область данных (неключевая область), которая расположена под чертой в дочерней сущности. Если атрибуты, которые мигрировали в неключевую область дочерней сущности, не нужны в этой сущности, то связь называется необязательной неидентифицирующей связью, что подразумевает, что мигрировавшие атрибуты не нужны дочерней сущности для ее идентификации и что она может существовать и без этих атрибутов. В ERwin необязательная неидентифицирующая связь обозначается пунктирной линией с жирной точкой на одном конце (дочернем) и ромбиком на другом (родительском).

Связь является **идентифицирующей** тогда и только тогда, когда первичный ключ дочерней сущности содержит внешний ключ, идущий от родительской сущности. Если этого нет - связь будет неидентифицирующей.

Например, есть две сущности - ВОПРОС и ОТВЕТ. Связь ВОПРОС-ОТВЕТ является идентифицирующей, поскольку сущность ОТВЕТ не может быть однозначно определена, если не задана сущность ВОПРОС (просто ответов без вопроса нет). Или сущности ДОМ и КВАРТИРА. Сущность КВАРТИРА не может быть однозначно определена, если не задана сущность ДОМ.

**Мощность отношения** служит для обозначения отношения числа экземпляров родительской сущности к числу экземпляров дочерней.

### *1.3 Нормализация данных*

**Нормализация** – это процесс проверки и реорганизации сущностей и атрибутов с целью удовлетворения требований к реляционной модели данных. Процесс нормализации сводится к последовательному приведению структур данных к нормальным формам – формализованным требованиям к организации данных.

## 2 Знакомство с ERwin

### 2.1 Общие сведения о ERwin

ERwin - CASE-средство проектирования баз данных от фирмы Computer Associates. ERwin сочетает графический интерфейс Windows, инструменты для построения ER-диаграмм, редакторы для создания логического и физического описания модели данных и прозрачную поддержку ведущих реляционных СУБД. Для удобства изложения материала здесь и далее использована оригинальная терминология, принятая в ERwin.

ERwin не привязан к технологии какой-либо конкретной фирмы, поставляющей СУБД или средства разработки. Он поддерживает различные серверы баз данных и настольные СУБД, а также может обращаться к базе данных через интерфейс ODBC. Так, в текущей версии ERwin встроена поддержка 23 СУБД, среди которых: Oracle; Microsoft SQL Server и т.п. Заметим лишь, что речь идет только о реляционных СУБД.

Процесс моделирования в ERwin базируется на методологии проектирования реляционных баз данных IDEF1X.

### 2.2 Отображение модели данных в ERwin. Физическая и логическая модель данных

ERwin имеет два уровня представления модели - логический и физический.

**Логический уровень** - это абстрактный взгляд на данные, на нем данные представляются так, как выглядят в реальном мире, и могут называться так, как они называются в реальном мире, например «Постоянный клиент», «Отдел» или «Фамилия сотрудника». Объекты модели, представляемые на логическом уровне, называются сущностями и атрибутами. Логическая модель данных может быть построена на основе другой логической модели, например на основе модели процессов (см. BPwin).

Логическая модель данных является универсальной и никак не связана с конкретной реализацией СУБД.

**Физическая модель** данных, напротив, зависит от конкретной СУБД, фактически являясь отображением системного каталога. В физической модели содержится информация о всех объектах БД. Поскольку стандартов на объекты БД не существует (например, нет стандарта на типы данных), физическая модель зависит от конкретной реализации СУБД. Следовательно, одной и той же логической модели могут соответствовать несколько разных физических моделей.

Разделение модели данных на логические и физические позволяет решить несколько важных задач:

- документирование модели;
- масштабирование.

ERwin позволяет решить задачу по переносу структуры данных с одного сервера на другой.

### 2.3 Основы работы с ERwin

Запуск ERwin - Кнопка Start /ERwin.

Появляется диалоговое окно Computer Associates ERwin. В этом окне выберите опцию Create a new model (Создание новой модели) и щелкните по кнопке OK (Создать). Откроется диалоговое окно Create Model – Select Template (Создание модели – Выбор шаблона).

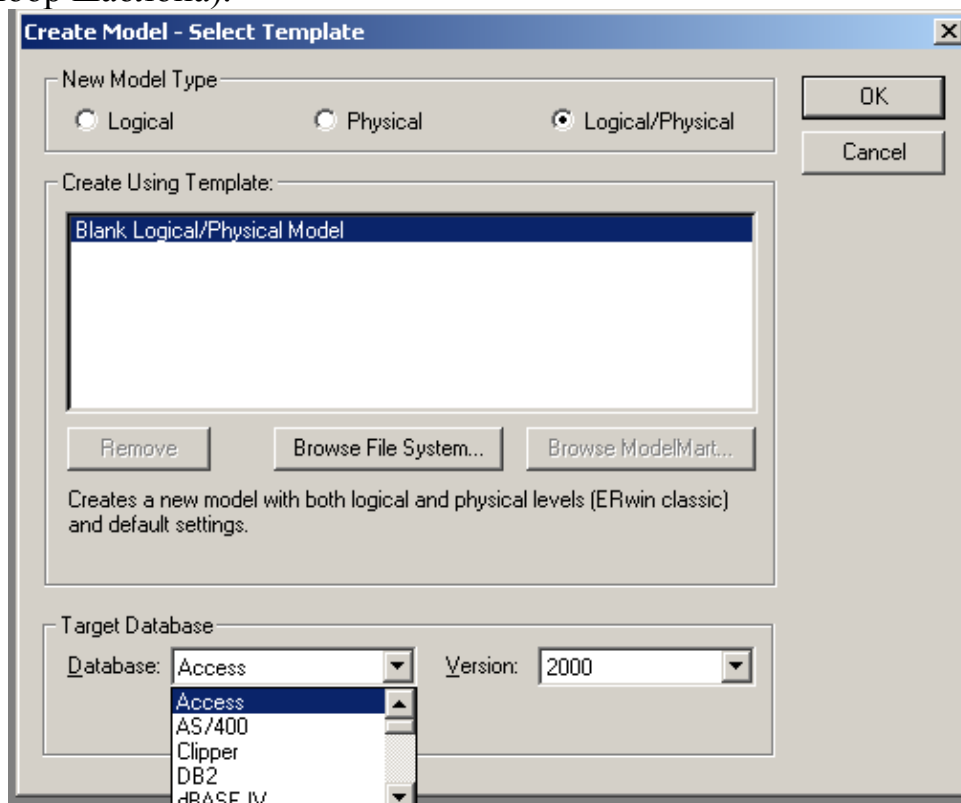


Рисунок 2 - Выбор шаблона модели

Во фрейме New Model Type установите опцию Logical/Physical, а во фрейме Target Database (Целевой сервер базы данных) установите в раскрывающемся окне списка систему управления базой данных (СУБД), например, Access, и ее версию.

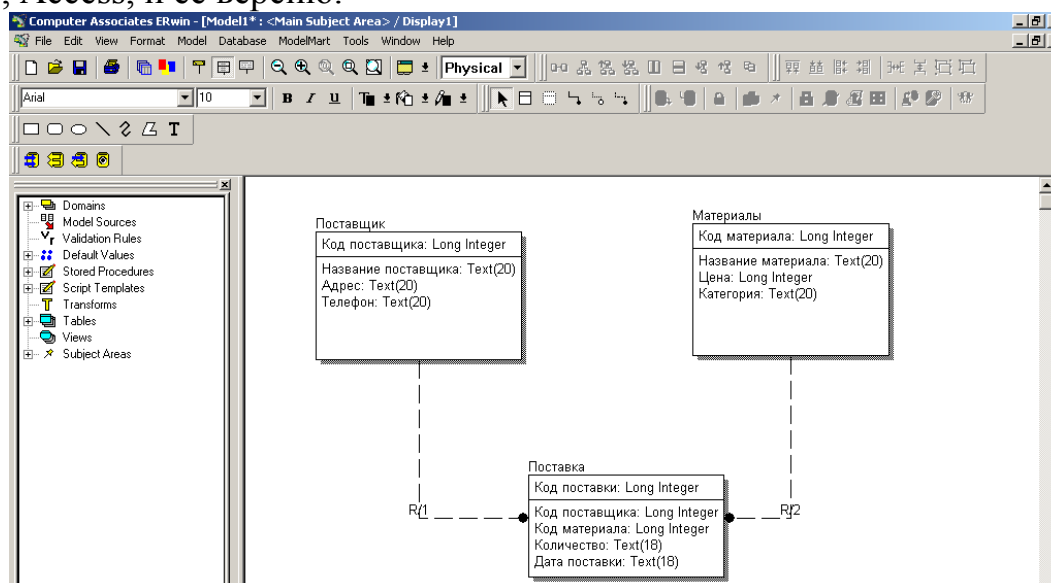


Рисунок 3 - Рабочее пространство ERwin

Таблица 1.1 - Стандартная панель инструментов (Standart)

Кнопки	Назначение кнопок
	Создание, открытие, сохранение и печать модели
	Вызов диалога Report Browser для генерации отчета
	Изменение уровня просмотра модели: уровень сущностей, уровень атрибутов и уровень определений
	Изменение масштаба просмотра модели
	Создание и переключение между подмножествами модели – Subject Area
Logical	Отображение логической модели данных
Physical	Отображение физической модели данных

Таблица 1.2 - Инструментарий (палитра инструментов - Toolbox)





Кнопки	Назначение кнопок
	Кнопка указателя (режим мыши) – в этом режиме можно устанавливать фокус на каком-либо объекте модели.
	Кнопка внесения сущности – для внесения сущности нужно щелкнуть левой кнопкой мыши по кнопке внесения сущностей и по свободному пространству на модели. Повторный щелчок приведет к внесению в модель новой сущности. Для редактирования сущностей или других объектов модели необходимо перейти в режим указателя
	Кнопка категории.. Для установления категориальной связи нужно щелкнуть левой кнопкой мыши по кнопке категории, затем один раз щелкнуть по сущности - родовому предку, затем - по сущности - потомку
	Идентифицирующая связь между сущностями
	Связь между сущностями типа многие-ко-многим
	Неидентифицирующая связь между сущностями

Таблица 1.3 - Панель инструментов шрифт и цвет (Font and Color)

Кнопки	Назначение кнопок
	Выбор наименования шрифта
	Выбор размера шрифта
	Выбор стиля шрифта
	Выбор цвета символов
	Выбор цвета заливки
	Выбор цвета линий



Таблица 1.4 - Панель инструментов базы данных (Database)

Кнопки	Назначение кнопок
	Генерация схемы БД на основе логической модели данных (прямая задача)
	Реконструкция логической модели данных на основе схемы БД (обратная задача)
	Анализ адекватности результатов генерации схемы БД на основе логической модели данных
	Выбор целевого сервера СУБД.

**Пример:**

Рассмотрим задачу разработки проекта БД о поставке строительных материалов для строительной фирмы.

В среде редактора ERwin необходимо создать логическую и физическую модель системы.

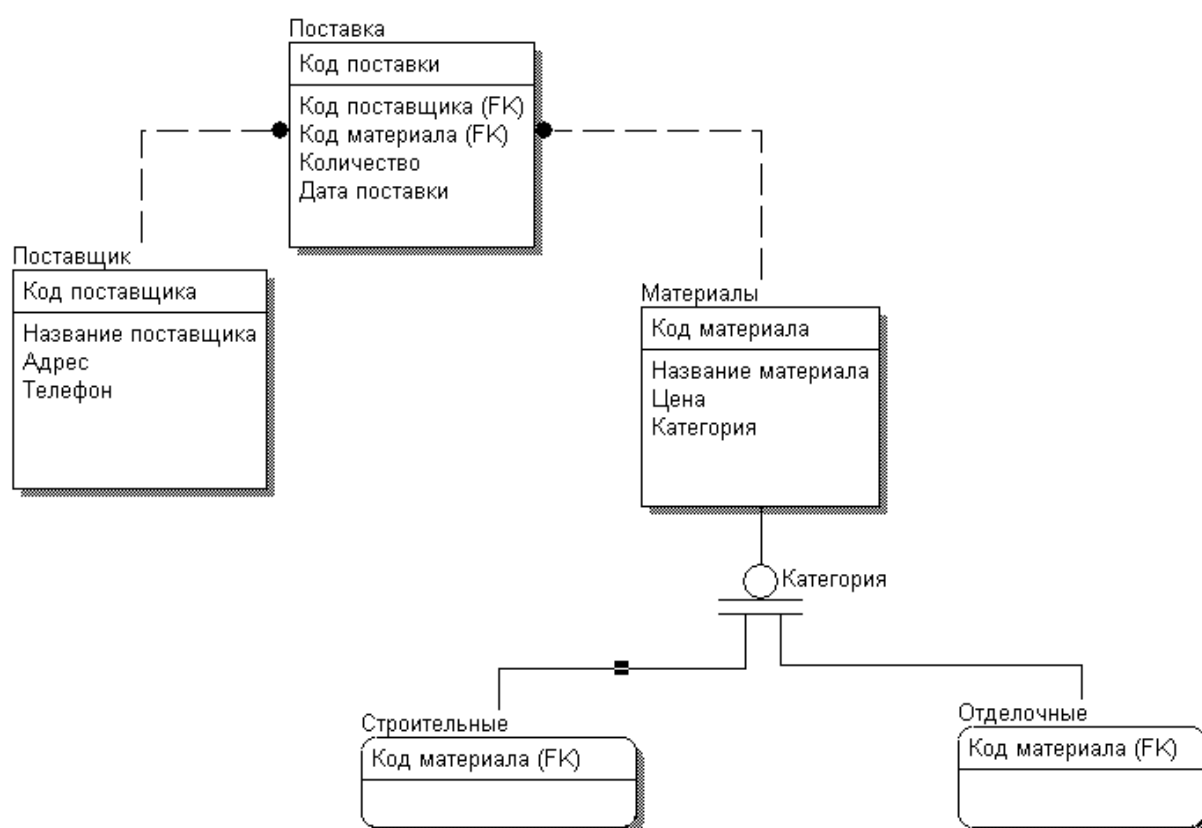


Рисунок 4 - Пример модели предметной области

**Этапы работы:**

1. Откроем редактор ERwin.

2. Создадим новую модель. Укажем при этом, что потребуется и **логический** и **физический** уровни представления модели. Выберем сервер СУБД – MS Access.

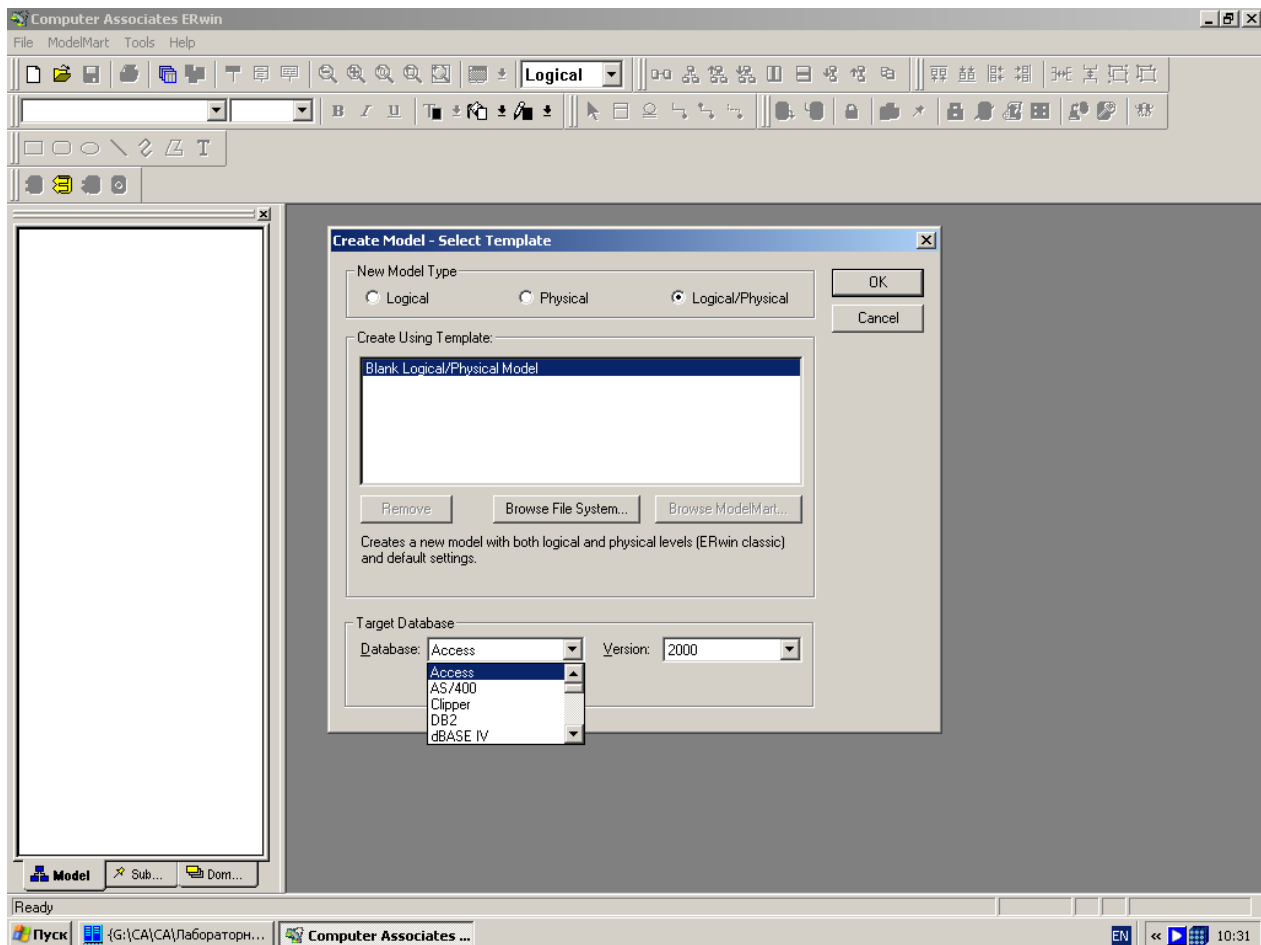





Рисунок 5 – Выбор сервера СУБД

3. Переключимся в режим работы с логической моделью. При помощи инструментов ERwin создадим Модель Сущность-Связь.

Режим просмотра – уровень сущностей: .

Создание объектов: инструменты , .

Для внесения сущности в модель необходимо (убедившись предварительно, что вы находитесь на уровне логической модели - переключателем между логической и физической моделью служит раскрывающийся список в правой части панели инструментов) "кликнуть" по кнопке сущности на панели инструментов (ERWin Toolbox) , затем "кликнуть" по тому месту на диаграмме, где необходимо расположить новую сущность. Щелкнув правой кнопкой мыши по сущности и выбрав из всплывающего меню пункт Entity Editor, можно вызвать диалог Entity Editor, в котором определяются имя, описание и комментарии сущности

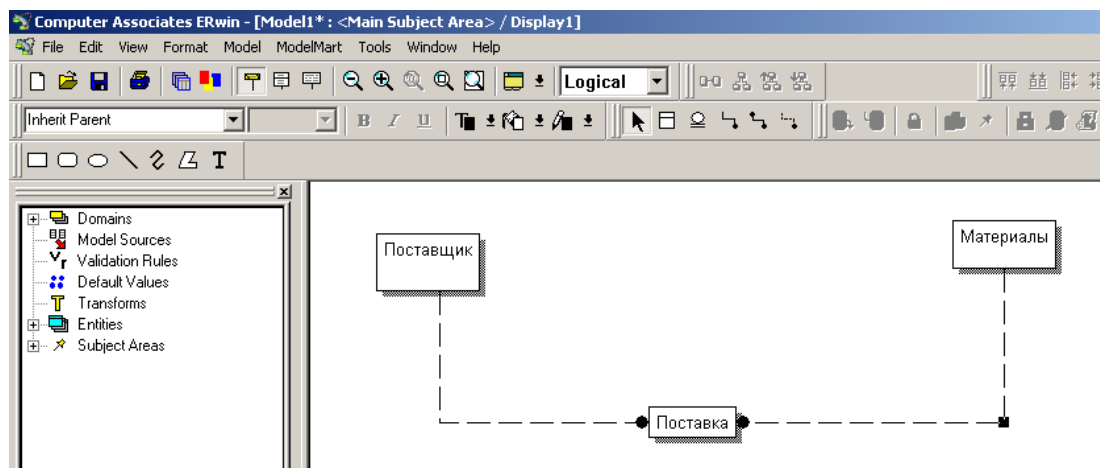



Рисунок 6 – Создание связи

4. Переключимся в режим модели данных, основанной на ключах . Дополним сущности ключевыми атрибутами Код Поставщика, Код материала, Код поставки.

**Первичный ключ (primary key)** - это атрибут или группа атрибутов, однозначно идентифицирующая экземпляр сущности. Атрибуты первичного ключа на диаграмме не требуют специального обозначения - это те атрибуты, которые находятся в списке атрибутов выше горизонтальной линии. При внесении нового атрибута в диалоге Attribute Editor для того, чтобы сделать его атрибутом первичного ключа, нужно включить флажок Primary Key в нижней части закладки General. На диаграмме неключевой атрибут можно внести в состав первичного ключа, воспользовавшись режимом переноса атрибутов (кнопка в палитре инструментов).


**Внешние ключи (Foreign Key)** создаются автоматически, когда связь соединяет сущности: связь образует ссылку на атрибуты первичного ключа в дочерней сущности и эти атрибуты образуют внешний ключ в дочерней сущности (миграция ключа). Атрибуты внешнего ключа обозначаются символом (FK) после своего имени.

*Выполнение:*

Для описания атрибутов следует, "кликнув" правой кнопкой по сущности, выбрать в появившемся меню пункт Attribute Editor. Появляется диалог Attribute Editor.

Если щелкнуть по кнопке New, то в появившемся диалоге New Attribute можно указать имя атрибута, имя соответствующей ему в физической модели колонки и домен. Домен атрибута будет использоваться при определении типа колонки на уровне физической модели.

Для атрибутов первичного ключа в закладке General диалога Attribute Editor необходимо сделать пометку в окне выбора Primary Key.

5. Переключимся в режим полной атрибутивной модели . Опишем сущности набором их атрибутов.

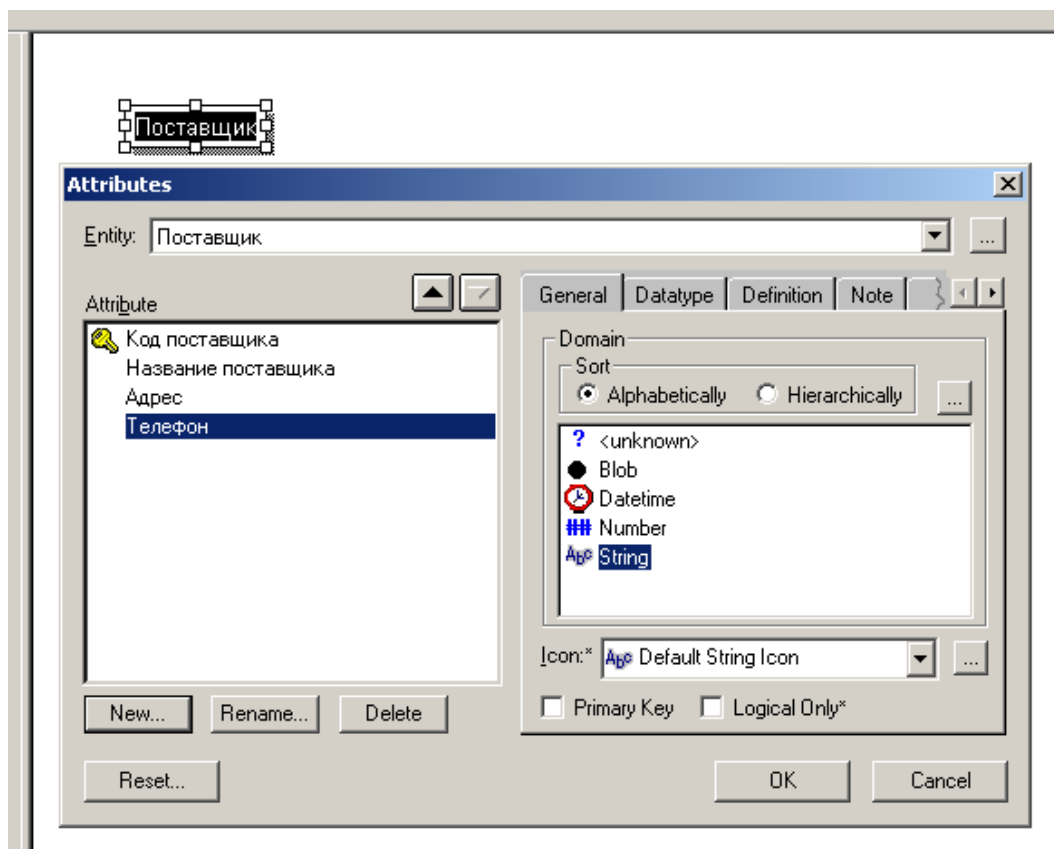


Рисунок 7 – Создание атрибутов

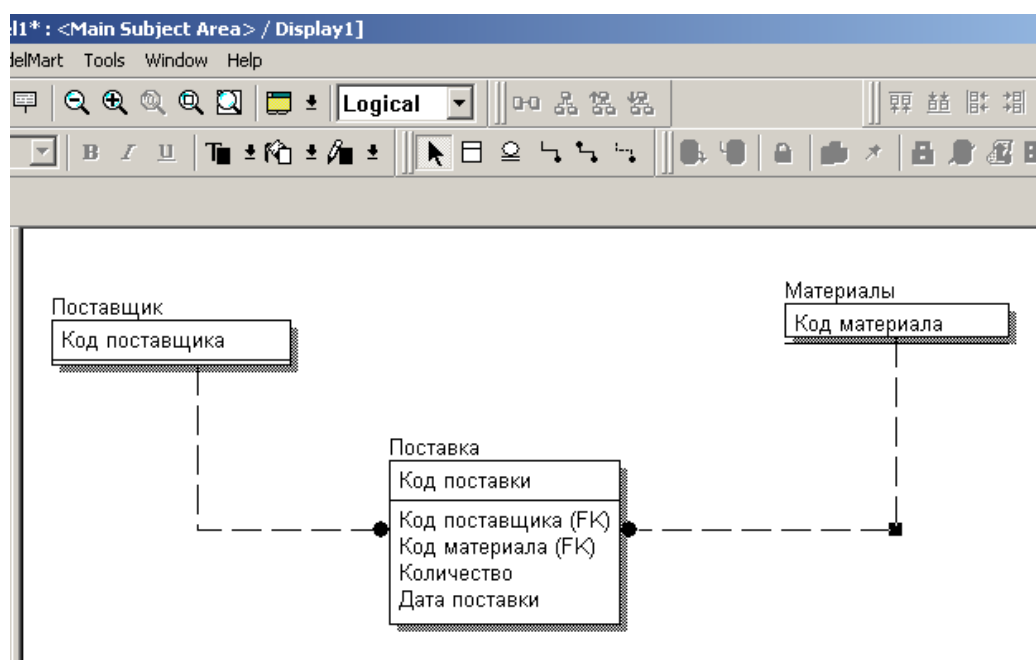


Рисунок 8 – Связанные таблицы

Обратите внимание на то, что при описании сущности Поставка атрибуты внешнего ключа Код\_Поставщика и Код\_Материала не описываются. Они мигрируют в описание сущности Поставка при установлении связей между сущностями.

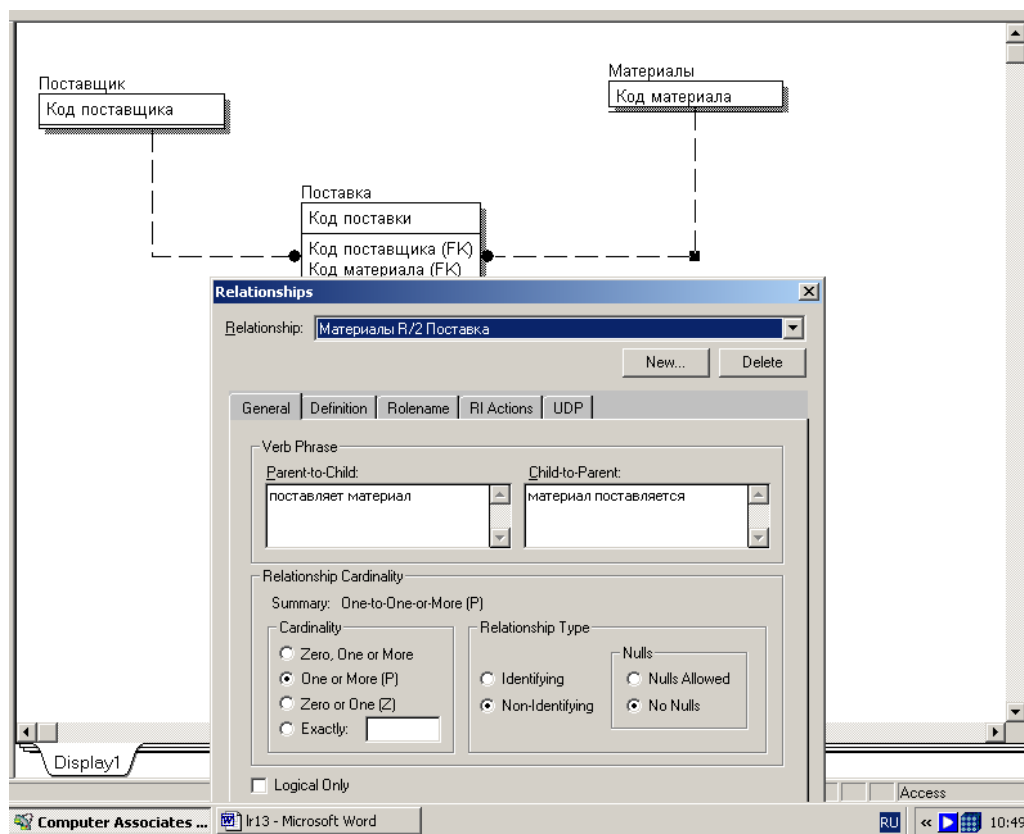


Рисунок 9 – Настройка параметров

Будьте внимательны при установке связей между сущностями. Правильно опишите тип связей. Так как мигрирующие атрибуты Код\_Поставщика и Код\_Материала не являются частью первичного ключа сущности поставка, то тип связи устанавливается «неидентифицирующая». При этом нулевые значения для мигрирующих атрибутов не разрешаются.

6. Переключитесь в режим работы с физической моделью. Теперь мы можем увидеть модель данных на языке описания СУБД, которую мы определили в момент создания модели. В нашем случае MS Access.

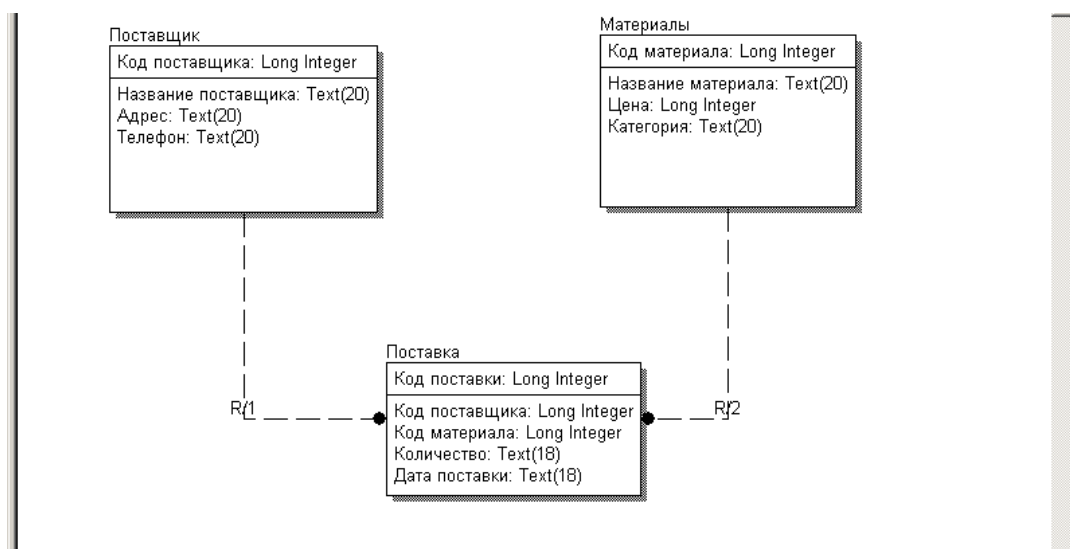



Рисунок 10 – Просмотр результатов работы

## 7. Создадим категориальную связь.

Для создания категориальной связи следует:

- установить курсор на кнопке  в палитре инструментов и нажать левую кнопку мыши;
- щелкнуть сначала по родовому предку, а затем по потомку;
- для установления второй связи в иерархии категории следует сначала щелкнуть по символу категории, затем по второму потомку.

В качестве категоризатора определим атрибут Категория. Так как у нас имеются только 2 категории материалов: строительные и отделочные, то категоризатор будет полный:

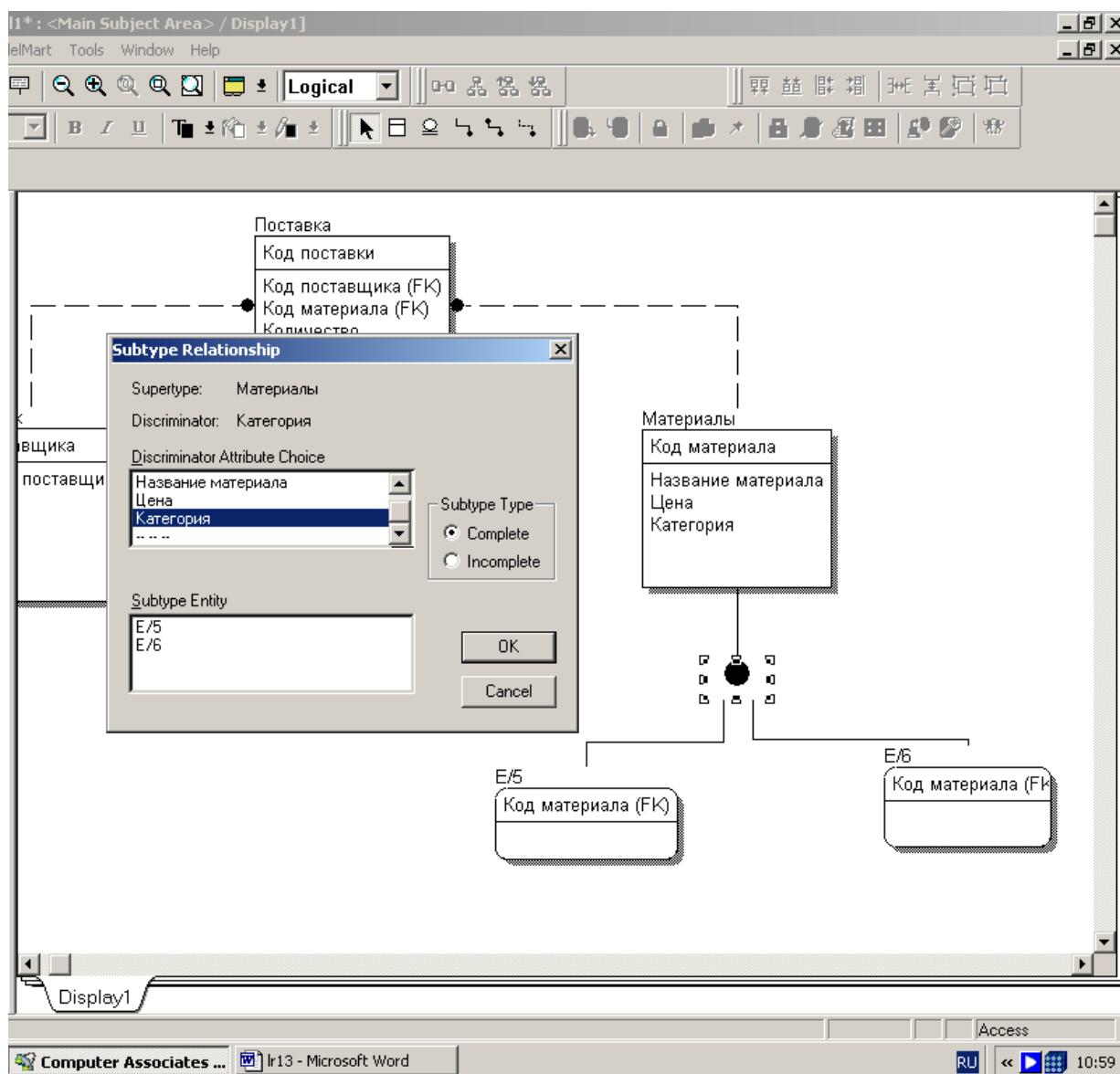


Рисунок 11 – Создание категориальной связи

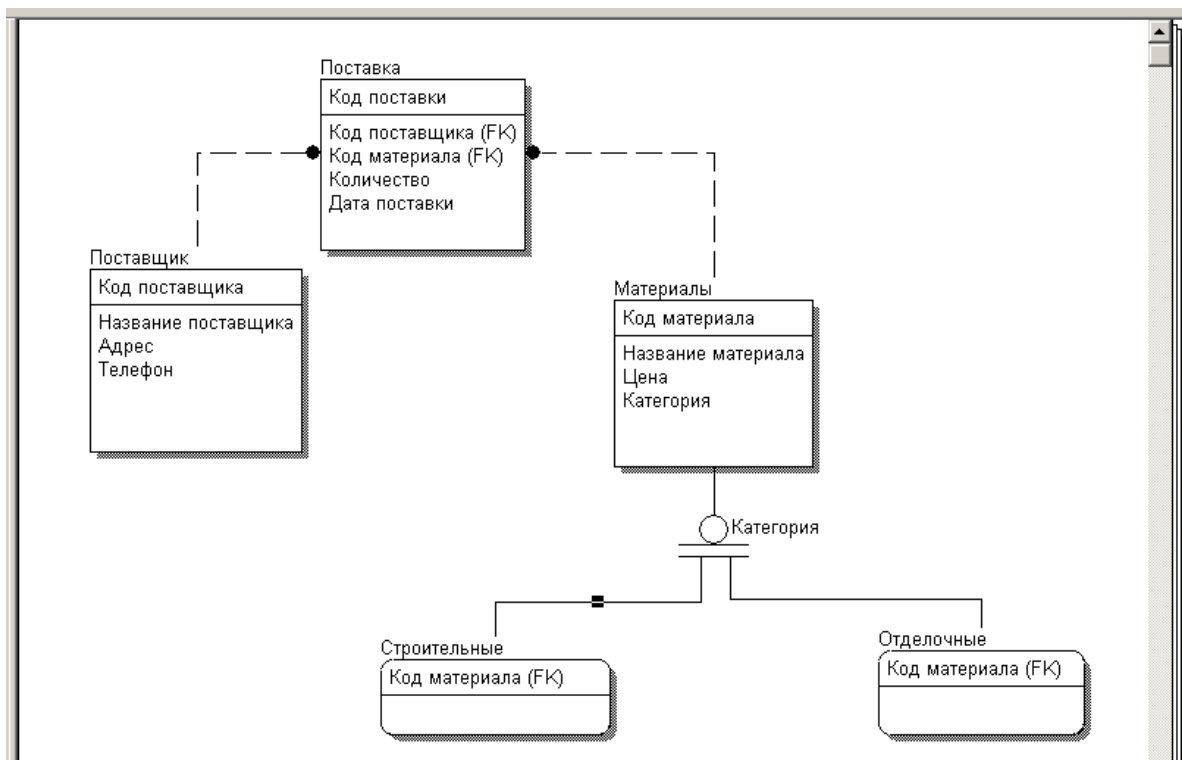


Рисунок 12 - Результат выполнения задания

### Содержание работы:

2.1 Реализуйте пример логической модели (рис. 12) из текста данной лабораторной работы.

2.2 Выполните задание для самостоятельного выполнения по образцу из приложения А.

2.3 Выполните задание для самостоятельного выполнения по индивидуальному варианту из приложения Б.

2.4 Оформите отчет по данной лабораторной работе.

Оформите отчет по данной лабораторной работе.

### Контрольные вопросы

1. Для чего предназначен пакет ERwin?
2. Что такое логическая модель данных?
3. Дайте понятие нормализации данных.
4. Дайте понятие идентифицирующих и неидентифицирующих связей.
5. Каково назначение кнопок стандартной панели инструментов ERwin?

## **Лабораторная работа №2**

### **Создание логической модели базы данных в MS Visio**

**Цель работы:** Изучить приемы проектирования логической модели БД

**Требования к организации рабочего места:** Лабораторная работа должна проводиться в компьютерном классе с установленным приложением MS Visio.

### **3 Понятие и элементы информационной модели логического уровня**

Методология IDEF1X – один из подходов к семантическому моделированию данных, основанный на концепции Сущность - Отношение (Entity – Relationship). Концептуально IDEF1X-модель можно рассматривать как проект логической схемы базы данных для проектируемой системы. Рассмотрим ключевые понятия методологии IDEF1X.

#### **Сущности**

«Сущность» представляет множество реальных или абстрактных предметов (людей, объектов, мест, событий, состояний, идей, пар предметов и т.д.), обладающих общими атрибутами или характеристиками.

Отдельный элемент этого множества называется «экземпляром сущности».

Каждая сущность может обладать любым количеством отношений с другими сущностями. Сущность является «независимой», если каждый экземпляр сущности может быть однозначно идентифицирован без определения его отношений с другими сущностями.

Сущность называется «зависимой», если однозначная идентификация экземпляра сущности зависит от его отношения к другой сущности.

Сущность обладает одним или несколькими атрибутами, которые либо принадлежат сущности, либо наследуются через отношение.

Сущность обладает одним или несколькими атрибутами, которые однозначно идентифицируют каждый образец сущности.

Каждая сущность может обладать любым количеством отношений с другими сущностями модели.

Если внешний ключ целиком используется в качестве первичного ключа сущности или его части, то сущность является зависимой от идентификатора. И наоборот, если используется только часть внешнего ключа или вообще не используются внешние ключи, то сущность является независимой от идентификатора.

#### **Отношения**

Отношение связи («отношение родитель-потомок») – это связь между сущностями, при которой каждый экземпляр одной сущности (родительской) ассоциирован с произвольным (в том числе нулевым) количеством экземпляров



другой сущности (сущностью-потомком), а каждый экземпляр сущности-потомка ассоциирован в точности с одним экземпляром сущности-родителя.

Если экземпляр сущности-потомка однозначно определяется своей связью с сущностью-родителем, то отношение называется «идентифицирующим отношением». В противном случае отношение называется «неидентифицирующим».

Отношению дается имя, выражаемое грамматическим оборотом глагола. Имя отношения всегда формируется с точки зрения родителя, так что может быть образовано предложение, если соединить имя сущности-родителя, имя отношения, выражение мощности и имя сущности-потомка. Отношение дополнительно определяется с помощью указания мощности: какое количество экземпляров сущности-потомка может существовать для сущности-родителя.

### **Атрибуты**

Сущность обладает одним или несколькими атрибутами, которые являются либо собственными для сущности, либо наследуются через отношение. Атрибуты однозначно идентифицируют каждый экземпляр сущности. Каждый атрибут идентифицируется уникальным именем.

Атрибуты изображаются в виде списка их имен внутри блока ассоциированной сущности, причем каждый атрибут занимает отдельную строку. Определяющие первичный ключ атрибуты размещаются наверху списка и отделяются от других атрибутов горизонтальной чертой.

### **Правила атрибутов**

Каждый атрибут должен иметь уникальное имя, одному и тому же имени должно соответствовать одно и то же значение. Одно и то же значение не может соответствовать различным именам.

Сущность может обладать любым количеством атрибутов. Каждый атрибут принадлежит в точности одной сущности.

Сущность может обладать любым количеством наследуемых атрибутов, но наследуемый атрибут должен быть частью первичного ключа соответствующей сущности-родителя или общей сущности.

Для каждого экземпляра сущности должно существовать значение каждого его атрибута (правило необращения в нуль).

Ни один из экземпляров сущности не может обладать более чем одним значением для связанного с ней атрибута (правило неповторения).

## **2 Создание логической модели данных в программе Microsoft Visio**

Microsoft Visio - векторный графический редактор, редактор диаграмм и блок-схем для Windows.

Выпускается в нескольких редакциях: Standard, Professional и Pro for Office 365.

Для создания логической модели модели данных в программе Microsoft Visio выберите категорию шаблонов «**Программы и базы данных**» в окне выбора шаблонов.

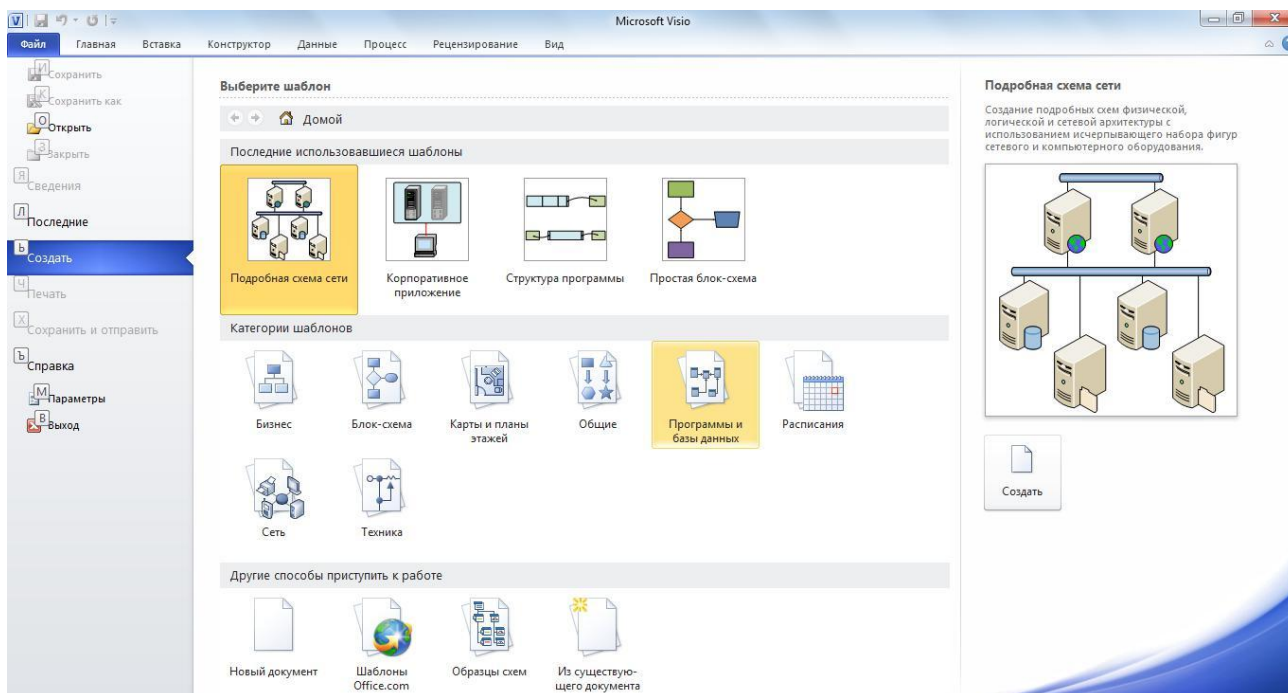


Рисунок 1 – Вид программы

В выбранной категории выберите шаблон «Схема модели базы данных» (либо в др. версиях - шаблон «Нотация базы данных Crow's Foot» ).

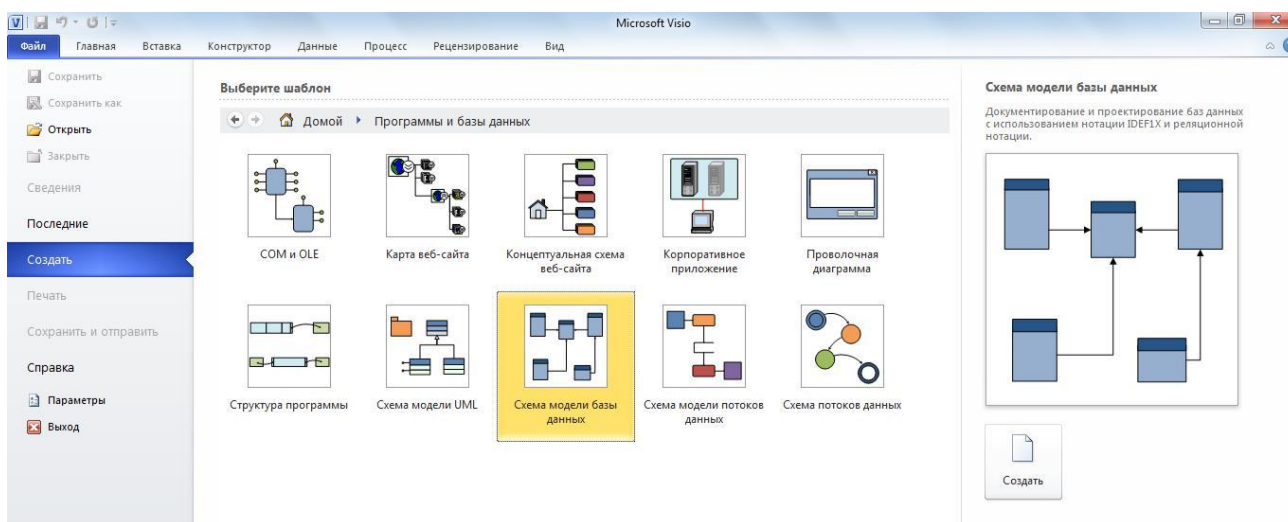


Рисунок 2 – Выбор шаблона

Далее создайте сущности логической модели базы данных, задайте им уникальное имя, определите атрибуты, задав им имя и тип, определите первичные и внешние ключи (рис. 3).

После размещения в модели сущности можно задать в ней первичный и внешний ключ с помощью контекстного меню строки «имя атрибута», для настроек типов соединительных линий следует использовать команду «Формат фигуры» из контекстного меню линий связи.

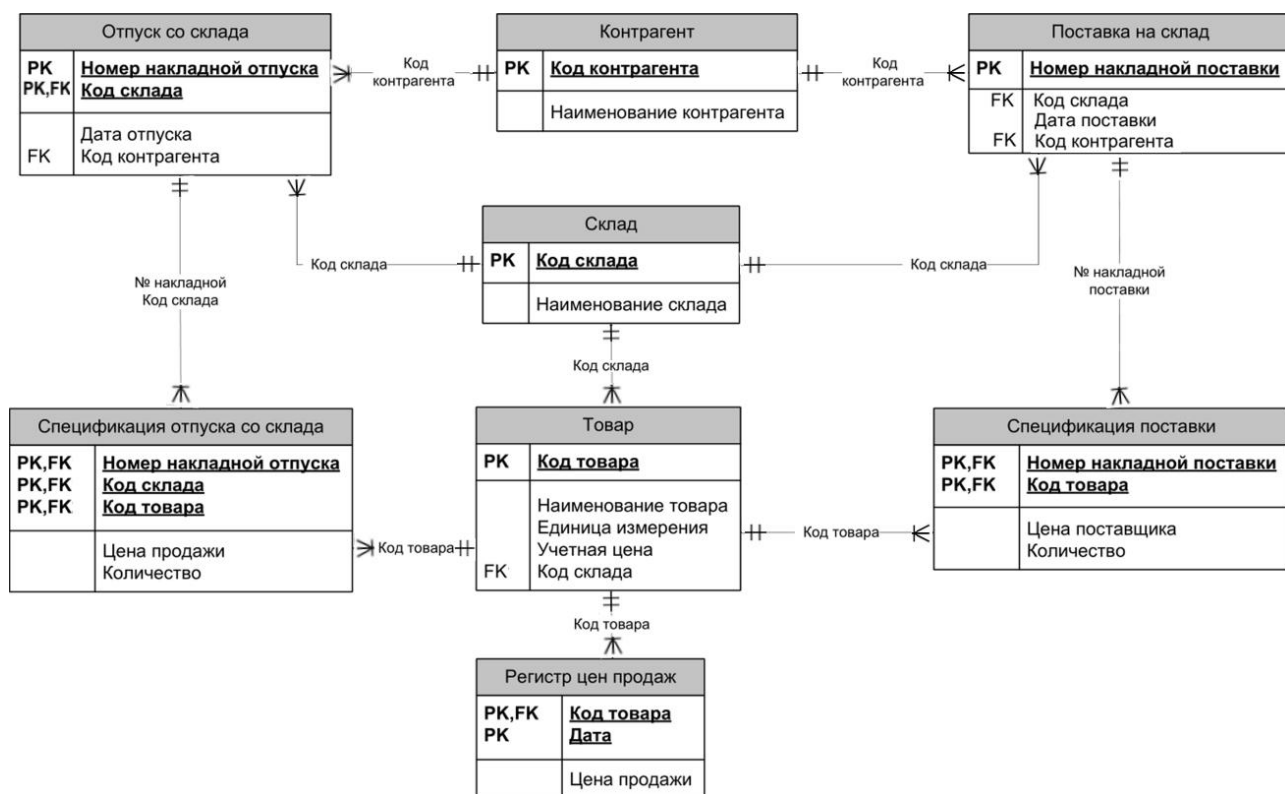


Рисунок 3 – Пример логической модели складского учета в нотации Crow's Foot

## 2 Содержание работы

3.1 Реализуйте пример логической модели складского учета (рис. 3) из текста данной лабораторной работы.

2.2 Выполните задание для самостоятельного выполнения по образцу из приложения В.

2.3 Выполните задание для самостоятельного выполнения по индивидуальному варианту из приложения Г.

2.4 Оформите отчет по данной лабораторной работе.

### Контрольные вопросы

1. Для чего предназначен пакет Visio? Каковы его основные возможности?
2. Что такое ER-модель данных?
3. Дайте понятие сущности и ее атрибутов.
4. Поясните на примерах различия между идентифицирующими и неидентифицирующими связями.

### **Библиографический список**

- 1) Советов, Б. Я. Базы данных: теория и практика [Текст]: учебник для бакалавров, для студентов вузов, обуч. по направлениям "Информатика и вычислительная техника" и "Информационные системы" / Б. Я. Советов, В. В. Цехановский, В. Д. Чертовский. - 2-е изд. - Москва: Юрайт, 2014. - 463 с.
- 2) Карпова, И. П. Базы данных [Текст]: курс лекций и материалы для практических занятий: учебное пособие для студентов техн. фак-тов, изучающих автоматизированные информационные системы и системы управления базами данных / И. П. Карпова. - Санкт-Петербург [и др.]: Питер, 2014. - 240 с.
- 3) Кузин, А. В. Базы данных [Текст]: учеб. пособие для студ. вузов, обуч. по напр. 654600 "Информатика и вычислительная техника" : допущено УМО по образованию / А. В. Кузин, С. В. Левонисова. - 3-е изд., стер. - М.: Академия, 2008. - 315 с.
- 4) Балдин, К. В. Информационные системы в экономике [Текст]: учебник для студ. вузов, обуч. по спец. 351400 "Прикладная информатика (по областям)" и другим междисциплинарным спец. : рек. УМО по образованию / К. В. Балдин, В. Б. Уткин. - 6-е изд. - М.: Дашков и К, 2010. - 392 с.
- 5) Гвоздева, В. А. Информатика, автоматизированные информационные технологии и системы [Текст]: учебник для студентов технических специальностей / В. А. Гвоздева. - М.: ФОРУМ : ИНФРА-М, 2011. - 541 с.

## Приложение А

### Образцы заданий для самостоятельного выполнения

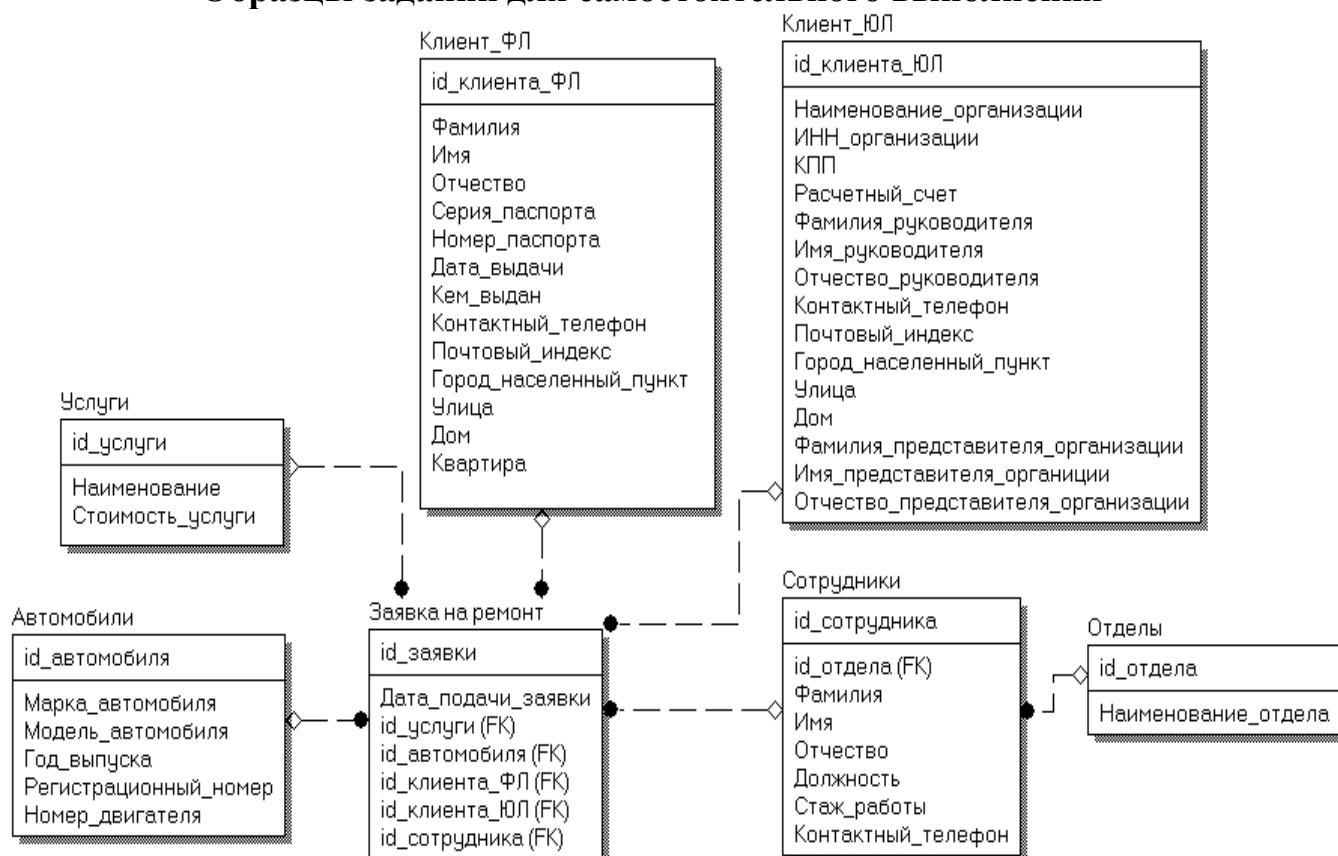


Рисунок А1 – Логическая модель базы данных «Учет заявок на ремонт транспорта от физических и юридических лиц (ФЛ и ЮЛ)»

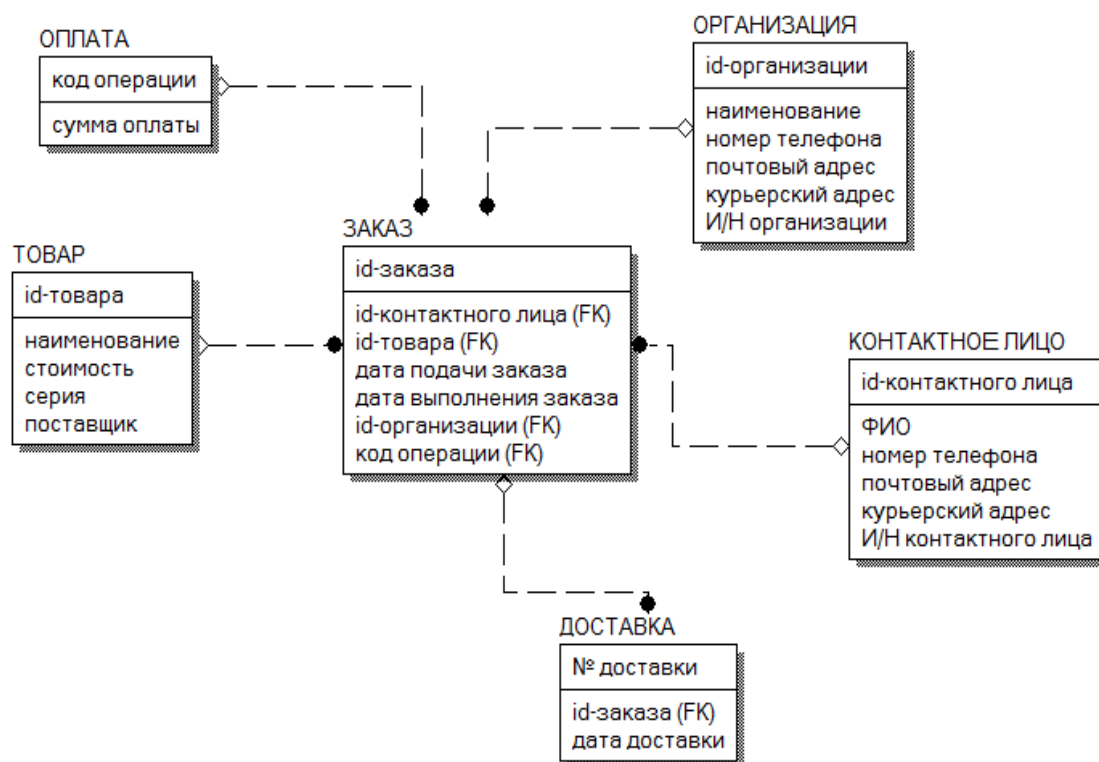


Рисунок А2 – Логическая модель базы данных «Заказ товаров»

## **Приложение Б**

### **Варианты индивидуальных заданий**

#### **1. Библиотечный каталог**

Основные таблицы:

- авторы (фамилия, имя, отчество);
- книги (наименование, жанр, издательство, количество страниц и др.);
- жанры книг;
- издательства;

Правила:

- у книги может быть несколько авторов.

#### **2. Сессия**

Основные таблицы:

- студенты (фамилия, имя, отчество и др.);
- предметы;
- преподаватели;
- экзаменационные ведомости;

Правила:

- каждый студент может иметь оценки по нескольким предметам, каждый предмет сдают несколько студентов.

#### **3. Аптека**

Основные таблицы:

- лекарственный препарат (наименование, цена, производитель и др.);
- заводы-производители лекарств;
- поступление (номер партии, товар, количество);
- продажа;

Правила:

- продажа может быть из нескольких поступлений.

#### **4. Администрирование пользователей**

Основные таблицы:

- пользователи (фамилия, имя, отчество, логин, пароль);
- группа пользователей (наименование и др.);
- программы (наименование программы, фирма-разработчик и др.);
- фирмы-разработчики программ;

Правила:

- каждый пользователь может входить в несколько групп пользователей;
- каждая группа пользователей имеет права на некоторые программы.

## Приложение В

### Образец задания для самостоятельного выполнения

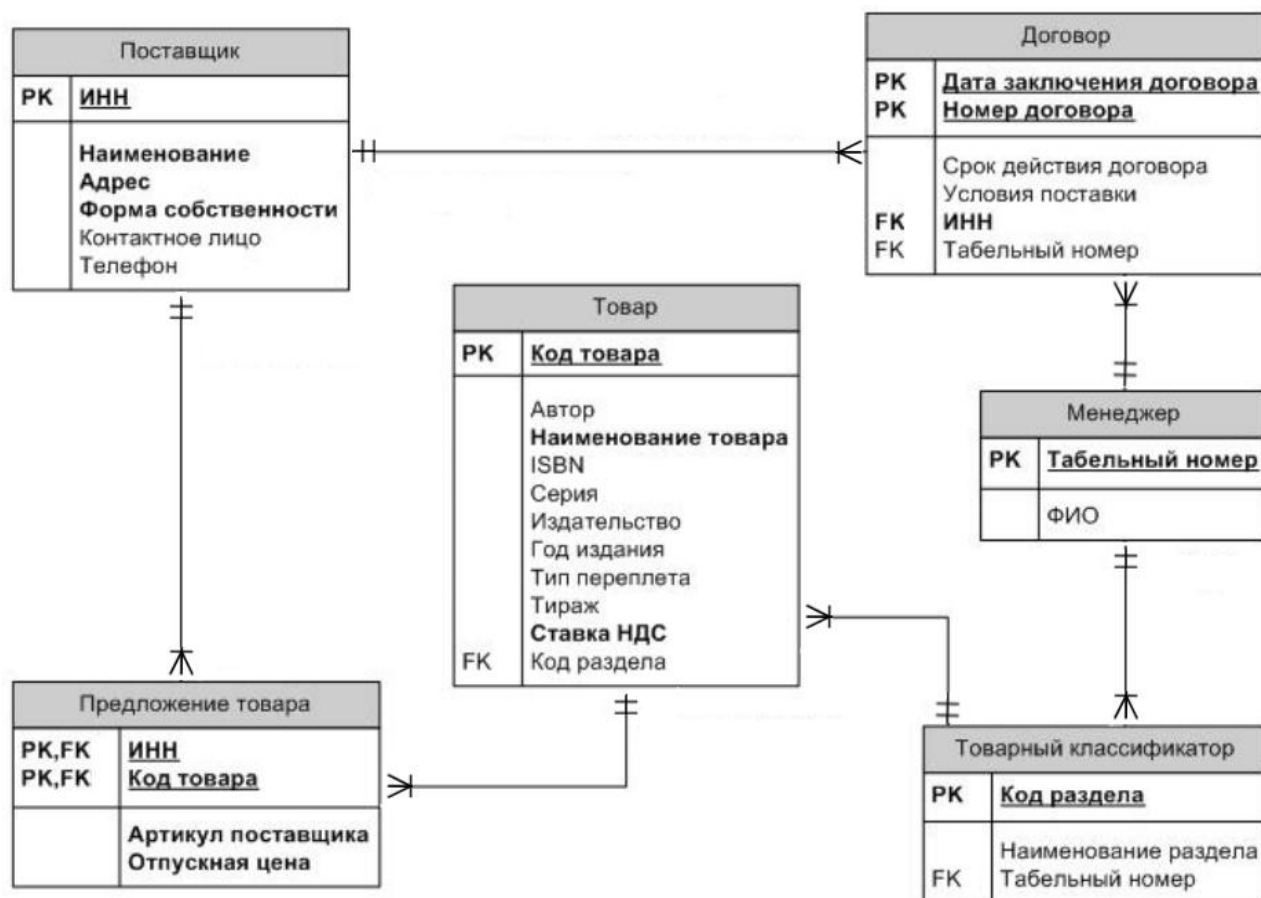


Рисунок В1 – Логическая модель базы данных по продаже книг в нотации Crow's Foot

## **Приложение Г**

### **Варианты заданий для самостоятельного выполнения**

- 1) Предметная область – Отдел реализации путевок туристической фирмы. Решаемые задачи: Учет сотрудников фирмы, учет клиентской базы, учет видов туров, учет договоров о реализации путевок.
- 2) Предметная область – Отдел кадров малого предприятия. Решаемые задачи: Учет анкетных данных о сотрудниках, учет подразделений (отделов), учет штатных должностных единиц предприятия, учет сведений о выполнении сотрудниками должностных обязанностей.
- 3) Предметная область – Отдел абонентского обслуживания библиотеки. Решаемые задачи: Учет сведений об абонентах библиотеки, учет книг в книжном каталоге, учет информации о выдаче книг абонентам, учет сотрудников библиотеки.
- 4) Предметная область – Отдел сбыта предприятия легкой промышленности. Решаемые задачи: Учет производимой продукции, учет рынка сбыта (магазинов), учет договоров поставок продукции, учет сотрудников отдела сбыта.
- 5) Предметная область – Аптечный склад фармацевтической фирмы. Решаемые задачи: Учет хранимой продукции, учет поставщиков лекарственных препаратов, учет закупок лекарственных препаратов, учет сотрудников склада.
- 6) Предметная область – Частное автотранспортное предприятие. Решаемые задачи: Учет автотранспортного парка, учет сведений о водительском составе, учет сведений о перевозках, учет ГСМ.
- 7) Предметная область – Отдел договоров с населением страховой фирмы. Решаемые задачи: Учет клиентской базы, учет форм и видов страхования, учет заключенных договоров страхования, учет сотрудников отдела.
- 8) Предметная область – Отдел закупок предприятия общественного питания. Решаемые задачи: Учет продуктов, учет поставщиков продуктов, учет закупок продуктов, учет сотрудников отдела.
- 9) Предметная область – Книжный магазин. Решаемые задачи: каталог продаваемых книг, учет поставщиков книг, учет поставок книг поставщиками, учет сотрудников магазина.
- 10) Предметная область – Отдел по работе с населением предприятия коммунального обслуживания. Решаемые задачи: Учет закрепленного за предприятием населения, учет предоставляемых населению услуг, учет договоров на предоставление услуг, учет сотрудников отдела.
- 11) Предметная область – Отдел договоров компании сотовой связи. Решаемые задачи: Учет клиентской базы, учет тарифных планов, учет договоров об обслуживании, учет сотрудников отдела.
- 12) Предметная область – Отдел реализации товаров предприятия мелкооптовой торговли. Решаемые задачи: Учет рынка сбыта, учет товаров, учет договоров о реализации товаров, учет сотрудников отдела.