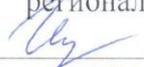


Документ подписан простой электронной подписью  
Информация о владельце:  
ФИО: Косогорова Людмила Алексеевна  
Должность: Ректор  
Дата подписания: 01.12.2022 11:05:24  
Уникальный программный ключ:  
4a47ce4135cc0671229e80c031ce72a914b0b6b4



**ЧАСТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ  
ВЫСШЕГО ОБРАЗОВАНИЯ  
«ИНСТИТУТ УПРАВЛЕНИЯ, БИЗНЕСА И ТЕХНОЛОГИЙ»**

УТВЕРЖДАЮ:  
Проректор по учебной работе и  
региональному развитию  
 Шульман М.Г.  
« 13 » февраля 2019 г.

**МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ К СТРУКТУРЕ, СОДЕРЖАНИЮ И  
ОФОРМЛЕНИЮ КУРСОВЫХ РАБОТ  
ПО КУРСУ «Базы данных»**

для студентов, обучающихся по направлению

09.03.03 Прикладная информатика

(профиль «Прикладная информатика в экономике»)

Уровень бакалавриата

# 1. ОСНОВЫ ОРГАНИЗАЦИИ ВЫПОЛНЕНИЯ КУРСОВОЙ РАБОТЫ

## 1.1. Общие положения

Цель курсовой работы – применение на практике знаний, полученных в процессе изучения курса «Базы данных», и получение практических навыков создания автоматизированных информационных систем (АИС), основанных на базах данных.

Предлагаемые методические указания к курсовым работам предназначены для студентов специальности Прикладная информатика (в экономике), по курсу «Базы данных». Методы проектирования базы данных является основными методами построения эффективных структур данных, а умелое владение этим инструментом организации данных - целью подготовки специалистов.

Главной задачей курсовых работ является овладение основами технологии разработки баз данных различного типа и их практического использования. Курсовая работа помогает студентам овладеть практическими навыками разработки баз данных при решении задач в различных предметных областях.

Курсовая работа ориентирована на развитие у студентов навыков и умений разрабатывать модели данных процессов и систем, и охватывает основные этапы разработки баз данных для различных предметных областей. При этом приобретаются навыки по проведению анализа предметной области и формулированию информационных требований; концептуальному (инфологическое) проектированию базы данных; логическому и физическому проектированию базы данных.

В связи с этим все курсовая работа позволяет поэтапно освоить процессы формирования базы данных, методологию ее проектирования, а также методы построения элементов системы управления данными. В качестве объектов исследования предлагаются предприятия, службы и технические устройства, которые используют в своей работе большой объем данных.

Курсовая работа направлена на исследование предметной области и проектирование базы данных в соответствии с индивидуальным заданием. Здесь отрабатывается технология построения базы данных, осуществляется поиск эффективных структур данных. Реализация спроектированной базы данных предполагается с использованием популярного инструмента построения баз данных (MS Access, SQL-сервер, Oracle и т.п.). Эти работы требуют от студентов творческого использования приобретенных знаний и умений для успешного применения методологии проектирования баз данных в конкретной предметной области с использованием современных инструментальных средств.

## 1.2. Задание на курсовую работу

Задание содержит название конкретной предметной области, для которой необходимо:

- сформулировать цель проектирования базы данных;
- описать возможного пользователя базы данных;
- определить круг запросов и задач, которые предполагается решать с использованием созданной базы данных;
- построить концептуальную модель;
- сформулировать требования к базе данных;
- построить реляционную модель и выполнить её нормализацию;
- осуществить выбор СУБД и технических средств;
- создать спроектированную базу данных в среде выбранной СУБД;

- рассчитать объём базы данных и/или привести формулы для расчёта объёма;
- разработать приложение для реализации запросов и решения задач;
- оценить базу данных с точки зрения возможностей её дальнейшего развития.

### 1.3. Этапы выполнения курсовой работы

В ходе выполнения курсовой работы рекомендуется придерживаться календарного плана, приведённого в таблице.

Содержание этапа	Продолжительность этапа
1. Выбор темы и утверждение технического задания	1 неделя
2. Разработка концептуальной модели.	3 недели
3. Разработка реляционной модели и выбор средств реализации.	3 недели
4. Создание базы данных и разработка приложения.	5 недель
5. Оформление пояснительной записки.	3 недели
6. Сдача курсовой работы на проверку.	1 неделя
7. Защита курсовой работы.	1 неделя

## 2. ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

### 2.1. Общие положения разработки БД

Проектирование базы данных (БД) является одной из наиболее сложных и ответственных задач, связанных с созданием АИС.

Проектирование базы данных – это процесс, который подразумевает использование определённой технологии. Никто не сомневается в том, что в случае нарушения технологии изготовления печатной платы, например, эта плата либо вообще не будет работать, либо не будет соответствовать заявленным характеристикам. Но почему-то считается, что соблюдать технологию проектирования БД (и вообще программного обеспечения) совершенно необязательно. И начинают работу по реализации реляционной БД с создания таблиц. Получившаяся в ходе такого "проектирования" база данных будет ненадёжной, неэффективной и сложной в сопровождении. (Исключением могут быть случаи простых предметных областей, которые можно отразить в базе данных, состоящей из 3-4 таблиц). Поэтому при создании базы данных необходимо придерживаться определённой технологии проектирования БД.

Опишем вкратце процесс проектирования реляционной базы данных.

База данных – это, фактически, модель предметной области (ПрО). Значит, для создания БД надо сначала проанализировать ПрО и создать её модель (это называется **инфологическим проектированием**).

Основой для *анализа предметной области* служат документы, которые отражают ПрО, и информация, которую можно получить от специалистов этой предметной области в процессе общения с ними.

Для анализа берутся те документы, которые имеют отношение к решаемой задаче. Изучение документов позволяет выявить объекты (сущности ПрО) и атрибуты сущностей – данные, которые должны храниться в БД.

Из общения со специалистами необходимо извлечь сведения об особенностях ПрО, которые позволяют установить ограничения целостности, зависимости и связи между объектами (субъектами) предметной области. Также специалисты обладают знаниями о

том, каковы алгоритмы обработки данных и какие задачи ставятся перед информационной системой.

Модель ПрО может быть описана любым удобным для разработчика способом (словесное описание, набор формул, диаграмма потоков данных и т.п.). Но, если при проектировании баз данных используется метод сущность–связь, то схема ПрО выполняется в виде ER–диаграммы (entity-relation diagram, диаграмма «сущность–связь»).

После создания модели ПрО определяются **требования к операционной обстановке**: какое аппаратное и программное обеспечение необходимо для реализации БД и АИС в целом. Основные технические параметры (объём оперативной и дисковой памяти, наличие сетевой платы и др.) определяются исходя из планируемого объёма БД, режима работы (локальный или удалённый доступ) и требований к эффективности работы системы (например, ко времени реакции на запрос пользователя или к общей производительности БД). В зависимости от планируемой нагрузки (интенсивности запросов) и требований к надёжности выбирается операционная система. Затем осуществляется **выбор СУБД**, под управлением которой будет работать создаваемая база данных.

На следующем этапе – этапе **логического проектирования** – ER-диаграмма формальным способом преобразуется в схему реляционной базы данных (РБД). На основании схемы РБД и описания сущностей ПрО составляются отношения (таблицы) базы данных. Потом выполняется нормализация отношений. Это необходимо сделать для того, чтобы исключить нарушения логической целостности данных и повысить таким образом надёжность и достоверность данных. В отдельных случаях после нормализации может выполняться денормализация, но причина для этого может быть только одна: повышение эффективности выполнения критических запросов.

В результате всех этих операций создаётся концептуальная схема БД – основной документ для базы данных.

Далее, на этапе **физического проектирования** полученные отношения описываются на языке DDL (Data definition language) – языке определения данных, который поддерживается выбранной СУБД. Также необходимо определить способы хранения данных (кластеризация, хеширование) и способы доступа к данным (индексирование) и создать соответствующие индексы и кластеры (если нужно). Если пользователей АИС можно разделить на группы по характеру решаемых задач, то для каждой группы создаётся свой набор прав доступа к объектам БД.

## 2.2. Последовательность проектирования базы данных

Процесс проектирования включает в себя следующие шаги:

1. Определение задач, стоящих перед базой данных.
2. Сбор и анализ документов, относящихся к исследуемой предметной области.
3. Описание особенностей ПрО, которые позволяют установить зависимости и связи между объектами (субъектами) предметной области.
4. Создание модели предметной области.
5. Определение групп пользователей и перечня задач, стоящих перед каждой группой.
6. Выбор аппаратной и программной платформы для реализации БД.
7. Выбор СУБД (системы управления базой данных).
8. Создание логической схемы БД.
9. Создание схем отношений, определение типов данных атрибутов и ограничений целостности.
10. Нормализация отношений (до третьей или четвёртой нормальной формы).
11. Определение прав доступа пользователей к объектам БД.
12. Написание текста создания основных объектов базы данных на языке SQL в синтаксисе выбранной СУБД (пользователи, таблицы и др.).

13. Написание текста создания вспомогательных объектов базы данных (представления, индексы, триггеры, роли и т.д.).

Эти шаги можно объединить с 5 этапов:

1. Инфологическое проектирование (1-5).
2. Определение требований к операционной обстановке, в которой будет функционировать информационная система (6).
3. Выбор системы управления базой данных (СУБД) и других инструментальных программных средств (7).
4. Логическое проектирование БД (8-11).
5. Физическое проектирование БД (12-13).

На сегодняшний день не существует формальных способов моделирования реальности, но инфологический подход закладывает основы методологии проектирования базы данных как модели предметной области.

### 2.3. Инфологическое проектирование

Основными задачами этапа инфологического проектирования являются определение предметной области системы и формирование взгляда на неё с позиций сообщества будущих пользователей БД, т.е. информационно-логической модели ПрО.

Инфологическая модель ПрО представляет собой описание структуры и динамики ПрО, характера информационных потребностей пользователей в терминах, понятных пользователю и не зависящих от реализации БД. Это описание выражается в терминах не отдельных объектов ПрО и связей между ними, а их типов, связанных с ними ограничений целостности и тех процессов, которые приводят к переходу ПрО из одного состояния в другое.

Основными подходами к созданию инфологической модели предметной области являются:

1. Функциональный подход к проектированию БД ("от задач").
2. Предметный подход к проектированию БД ("от предметной области").
3. Метод "сущность-связь" (entity–relation, ER–method).

Мы будем использовать метод "сущность–связь" как наиболее распространённый. Приведём основные термины, которыми мы будем пользоваться:

**Сущность** – это объект, о котором в системе будут накапливаться данные. Для сущности указывается название и тип (сильная или слабая). Сильные сущности существуют сами по себе, а существование слабых сущностей зависит от существования сильных.

**Атрибут** – свойство сущности. Различают:

- 1) *Идентифицирующие и описательные атрибуты*. Идентифицирующие позволяют отличить один экземпляр сущности от другого. Описательные атрибуты включают в себе интересующие нас свойства сущности.
- 2) *Составные и простые атрибуты*. Простой атрибут имеет неделимое значение. Составной атрибут является комбинацией нескольких элементов, возможно, принадлежащих разным типам данных (ФИО, адрес и др.).
- 3) *Однозначные и многозначные атрибуты* (могут иметь соответственно одно или много значений для каждого экземпляра сущности). Например, дата рождения – это однозначный атрибут, а номер телефона – многозначный.
- 4) *Основные и производные атрибуты*. Значение основного атрибута не зависит от других атрибутов; значение производного атрибута вычисляется на основе значений других атрибутов. Например, возраст вычисляется на основе даты рождения и текущей даты.
- 5) *Обязательные и необязательные* (первые должны быть указаны при размещении данных в БД, вторые могут не указываться).

Для каждого атрибута необходимо определить название, указать тип данных и описать ограничения целостности – множество значений, которые может принимать данный атрибут.

**Связь** – это осмысленная ассоциация между сущностями. Для связи указывается название, тип (факультативная или обязательная), степень (1:1, 1:n или m:n) и кардинальность (унарная, бинарная, тернарная или n-арная).

На рис. 1 приведены обозначения, которые мы будем использовать в ER-диаграммах.

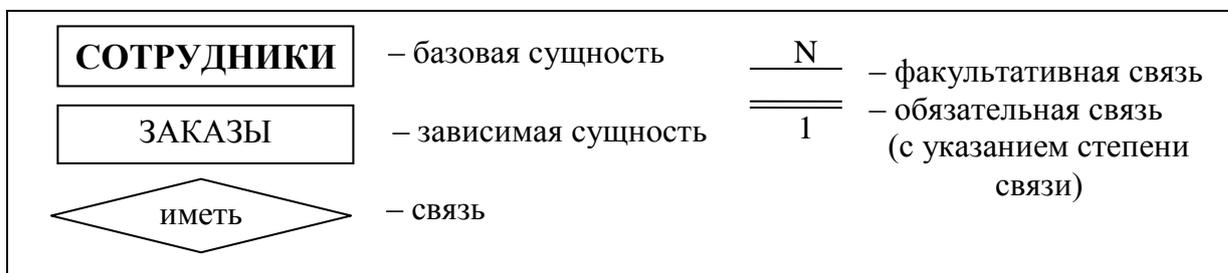


Рис.1. Обозначения, используемые в ER-диаграммах

## 2.4. Определение требований к операционной обстановке

На этом этапе производится оценка требований к вычислительным ресурсам, необходимым для функционирования системы, определение типа и конфигурации конкретной ЭВМ, выбор типа и версии операционной системы. Объем вычислительных ресурсов зависит от предполагаемого объема проектируемой базы данных и от интенсивности их использования. Если БД будет работать в многопользовательском режиме, то требуется подключение её к сети и наличие соответствующей многозадачной операционной системы.

## 2.5 Выбор СУБД и других программных средств

Выбор СУБД осуществляется на основании таких критериев, как тип модели данных и её адекватность потребностям рассматриваемой ПрО; характеристики производительности; набор функциональных возможностей; удобство и надежность СУБД в эксплуатации; стоимость СУБД и дополнительного программного обеспечения.

## 2.6 Логическое проектирование реляционной БД

На этапе логического проектирования разрабатывается логическая (концептуальная) структура БД. Для реляционной модели существуют формальные правила, которые позволяют преобразовать инфологическую модель ПрО в виде ER-диаграммы в логическую схему базы данных. Кроме получения схемы БД в целом на этом этапе выполняют создание схем отношений и их нормализацию. Этот этап более подробно рассмотрен в п.2 "Пример проектирования реляционной базы данных".

## 2.7. Физическое проектирование БД

Этап физического проектирования заключается в определении схемы хранения, т.е. физической структуры БД. Схема хранения зависит от той физической структуры, которую поддерживает выбранная СУБД. Физическая структура БД, с одной стороны, должна адекватно отражать логическую структуру БД, а с другой стороны, должна обеспечивать эффективное размещение данных и быстрый доступ к ним. Результаты этого этапа документируются в форме схемы хранения на языке определения данных (DDL, Data

Definition Language) выбранной СУБД. Принятые на этом этапе решения оказывают огромное влияние на производительность системы.

Одной из важнейших составляющих проекта базы данных является разработка средств защиты БД. Защита данных имеет два аспекта: защита от сбоев и защита от несанкционированного доступа. Для защиты от сбоев на этапе физического проектирования разрабатывается стратегия резервного копирования. Для защиты от несанкционированного доступа каждому пользователю доступ к данным предоставляется только в соответствии с его правами доступа, набор которых также является составной частью проекта БД.

## 2.8. Особенности проектирования реляционной базы данных

Проектирование реляционной базы данных проходит в том же порядке, что и проектирование БД других моделей данных, но имеет свои особенности.

Проектирование схемы БД должно решать задачи минимизации дублирования данных и упрощения процедур их обработки и обновления. При неправильно спроектированной схеме БД могут возникнуть аномалии модификации данных. Они обусловлены отсутствием средств явного представления типов множественных связей между объектами ПрО и неразвитостью средств описания ограничений целостности на уровне модели данных.

Для решения подобных проблем проводится **нормализация отношений**.

Механизм нормализации реляционных отношений разработал Э.Ф. Кодд (E.F. Codd). Этот механизм позволяет по формальным признакам любое отношение преобразовать к третьей нормальной форме.

Нормализация схемы отношения выполняется путём декомпозиции схемы. **Декомпозицией** схемы отношения R называется замена её совокупностью схем отношений  $A_i$  таких, что

$$R = \bigcup_i A_i,$$

и не требуется, чтобы отношения  $A_i$  были непересекающимися.

Первая нормальная форма относится к понятию простого и сложного (составного или многозначного) атрибута (см. п.1.2.1).

### Первая нормальная форма (1НФ).

Отношение приведено к 1НФ, если все его атрибуты простые.

Для того чтобы привести к 1НФ отношение, содержащее сложные атрибуты, нужно:

- разбить составные атрибуты на простые,
- построить декартово произведение всех многозначных атрибутов с кортежами, к которым они относятся.

Для идентификации кортежа в этом случае понадобится составной ключ, включающий первичный ключ исходного отношения и все многозначные атрибуты.

Вторая нормальная форма основана на понятии *функциональной зависимости*. Пусть X и Y – атрибуты некоторого отношения. Если в любой момент времени каждому значению X соответствует единственное значение Y, то говорят, что Y функционально зависит от X ( $X \rightarrow Y$ ). Атрибут X в функциональной зависимости  $X \rightarrow Y$  называется *детерминантом* отношения.

В нормализованном отношении все неключевые атрибуты функционально зависят от ключа отношения. Неключевой атрибут функционально полно зависит от составного ключа, если он функционально зависит от ключа, но не находится в функциональной зависимости ни от какой части составного ключа.

### Вторая нормальная форма (2НФ).

Отношение находится во 2НФ, если оно приведено к 1НФ и каждый неключевой атрибут функционально полно зависит от составного первичного ключа.

(Таким образом, если отношение в 1НФ имеет простой первичный ключ, оно сразу находится во второй нормальной форме).

Для того чтобы привести отношение ко 2НФ, нужно:

- построить его проекцию, исключив атрибуты, которые не находятся в функционально полной зависимости от составного первичного ключа;
- построить дополнительно одну или несколько проекций на часть составного ключа и атрибуты, функционально зависящие от этой части ключа.

Третья нормальная форма основана на понятии *транзитивной зависимости*. Пусть  $X$ ,  $Y$ ,  $Z$  – атрибуты некоторого отношения. При этом  $X \rightarrow Y$  и  $Y \rightarrow Z$ , но обратное соответствие отсутствует, т.е.  $Z$  не зависит от  $Y$  или  $Y$  не зависит от  $X$ . Тогда говорят, что  $Z$  транзитивно зависит от  $X$  ( $X \twoheadrightarrow Z$ ).

#### **Третья нормальная форма (3НФ).**

Отношение находится в 3НФ, если оно находится во 2НФ и каждый неключевой атрибут нетранзитивно зависит от первичного ключа.

Для того чтобы привести отношение к 3НФ, нужно:

- построить проекцию, исключив транзитивно зависящие от ключа атрибуты;
- построить дополнительно одну или несколько проекций на детерминанты исходного отношения и атрибуты, функционально зависящие от них.

Исключение составляют случаи, когда для транзитивной зависимости  $X \twoheadrightarrow Z$  ( $X \rightarrow Y$  и  $Y \rightarrow Z$ ) либо  $Z$  зависит от  $Y$ , либо  $Y$  зависит от  $X$ , т.е. между атрибутами  $X$  и  $Y$ , например, существует связь 1:1. В такой ситуации декомпозиция отношения не производится.

Четвертая нормальная форма основана на понятии *многозначной зависимости*. Многозначная зависимость существует, если заданным значениям атрибута  $X$  соответствует множество, состоящее из нуля (или более) значений атрибута  $Y$  ( $X \twoheadrightarrow Y$ ).

Различают тривиальные и нетривиальные многозначные зависимости. *Тривиальной* называется такая многозначная зависимость  $X \twoheadrightarrow Y$ , для которой  $Y \subset X$  или  $X \cup Y = R$ , где  $R$  – рассматриваемое отношение. Тривиальная многозначная зависимость не нарушает 4НФ. Если хотя бы одно из двух этих условий не выполняется, то такая зависимость называется *нетривиальной*.

#### **Четвертая нормальная форма (4НФ).**

Отношение находится в 4НФ, если оно находится в 3НФ и в нём отсутствуют нетривиальные многозначные зависимости.

Для того чтобы привести отношение к 4НФ, нужно построить две или более проекции исходного отношения, каждая из которых содержит ключ и одну из многозначных зависимостей.

### 3. ТРЕБОВАНИЯ ОФОРМЛЕНИЮ КУРСОВОЙ РАБОТЫ

#### 3.1 Требования к оформлению листов текстовой части.

Текстовая часть курсовой работы выполняется на листах формата А4 (210 x 297 мм) без рамки, соблюдением следующих размеров полей:

- левое – 30 мм,
- правое – 15 мм,
- верхнее – 20 мм,
- нижнее – 20 мм.

Страницы текста подлежат обязательной нумерации, которая проводится арабскими цифрами с соблюдением сквозной нумерации по всему тексту. Номер страницы проставляют в центре нижней части листа без точки в конце. Первой страницей считается титульный лист, но номер страницы на нем не проставляется.

Текстовую часть можно выполнить одним из следующих способов:

- с применением печатающих и графических устройств вывода ЭВМ согласно ГОСТ 2.004;
- машинописным – через полтора-два интервала (ГОСТ 13.1.002). Шрифт пишущей машинки должен быть четким, высотой не менее 2,5 мм, лента только черного цвета (полужирная);
- рукописным – чертежным шрифтом по ГОСТ 2.304 с высотой букв не менее 2,5 мм, а цифр – 5 мм. Цифры и буквы выполняются тушью или пастой (чернилами) черного цвета.

При выполнении текстовой части работы на компьютере текст должен быть оформлен в текстовом редакторе *Word for Windows*.

Тип шрифта: *Times New Roman Cyr*. Шрифт основного текста: обычный, размер 14 пт. Шрифт заголовков разделов: полужирный, прописной, размер 14 пт. Шрифт заголовков подразделов: полужирный, строчной, размер 14 пт.

Межсимвольный интервал: обычный. Межстрочный интервал: одинарный.

Опечатки, описки и графические неточности, обнаруженные в процессе выполнения, допускается исправлять подчисткой или закрасиванием белой краской и нанесением в том же месте исправленного текста машинописным способом или черными чернилами. Помарки и следы не полностью удаленного прежнего текста не допускаются. Возможно наклеивание рисунков и фотографий.

#### 3.2 Требования к структуре текста.

Курсовая работа должна быть выполнена с соблюдением требованиями ЕСКД<sup>1</sup>. Текст основной части разделяют на разделы, подразделы, пункты (ГОСТ 2.105-95, ГОСТ 7.32-2001).

*Разделы* должны иметь порядковые номера в пределах всего текста, обозначенные арабскими цифрами без точки. Каждый раздел пояснительной записки рекомендуется начинать с нового листа. Наименование разделов записываются в виде заголовков (симметрично тексту) с прописной буквы шрифта *Times New Roman*, размер 14 пт.

Подразделы должны иметь нумерацию в пределах каждого раздела, номера подразделов состоят из номера раздела и подраздела, разделенной точкой. В конце номера подраздела точка не ставится. Наименование подразделов записываются в виде заголовков (с абзаца) строчными буквами (кроме первой прописной), шрифт *Times New Roman*, размер 14 пт.

Подраздел допускается разбивать на пункты, нумерация которых выполняется аналогично.

*Пример:* 1.2.3 - обозначает раздел 1, подраздел 2, пункт 3

Внутри пунктов или подпунктов могут быть приведены перечисления. Перед каждой позицией перечисления следует ставить дефис или, при необходимости ссылки в тексте на

---

<sup>1</sup> ЕСКД – Единая Система Конструкторской Документации.

одно из перечислений, строчную букву, после которой ставится скобка. Для дальнейшей детализации перечислений необходимо использовать арабские цифры, после которых ставится скобка, а запись производится с абзацного отступа.

*Пример:*

- а) \_\_\_\_\_
- б) \_\_\_\_\_
- 1) \_\_\_\_\_
- 2) \_\_\_\_\_
- в) \_\_\_\_\_

«Введение» и «Заключение» не нумеруются.

Наименования разделов и подразделов должны быть краткими. Наименование разделов и подразделов записывают с абзацного отступа с первой прописной буквы без точки в конце, не подчеркивая. Переносы слов в заголовках не допускаются.

Расстояние между заголовками и текстом должно быть равно 3 интервалам. Расстояние между заголовками разделов и подраздела – 2 интервала. Расстояние между последней строкой текста и последующим заголовком подраздела – 3 интервала. Каждый раздел рекомендуется начинать с нового листа.

### **3.3 Требования к изложению текста.**

Изложение содержания пояснительной записки должно быть кратким и четким. В тексте должны применяться научно-технические термины, обозначения и определения, установленные соответствующими стандартами или общепринятые в научно-технической литературе.

Условные буквенные обозначения величин, а также условные графические обозначения должны соответствовать требованиям государственных стандартов (это относится и к единицам измерения).

Условные буквенные обозначения должны быть тождественными во всех разделах записки. Если в пояснительной записке принята особая система сокращения слов или наименований, то в ней должен быть приведен перечень принятых сокращений, который помещают перед «содержанием».

В тексте, за исключением формул, таблиц и рисунков, не допускается:

- применять математический знак минус (-) перед отрицательными значениям величин (следует писать слово «минус»);
- применять знак «□» для обозначения диаметра (следует писать слово «диаметр»). При указании размера или предельных отклонений диаметра на чертежах, помещенных в тексте документа, перед размерным числом следует писать знак «□»;
- применять без числовых значений математические знаки, например:
- (больше), < (меньше), = (равно), > (больше или равно), ≤ (меньше или равно),
- ≠ (не равно), а также № (номер), % (процент);
- применять индексы стандартов, технических условий без регистрационного номера.

Правила печатания знаков. Знаки препинания (точка, запятая, двоеточие, точка с запятой, многоточие, восклицательный и вопросительный знаки) от предшествующих слов пробелом не отделяют, а от последующих отделяют одним пробелом.

Дефис от предшествующих и последующих элементов не отделяют.

Тире от предшествующих и последующих элементов отделяют обязательно.

Кавычки и скобки не отбивают от заключенных в них элементов. Знаки препинания от кавычек и скобок не отбивают.

Знак № применяют только с относящимися к нему числами, между ними ставят пробел.

Знаки сноски (звездочки или цифры) в основном тексте печатают без пробела, а от текста сноски отделяют одним ударом (напр.: слово<sup>1</sup>, <sup>1</sup> Слово).

Знаки процента и промилле от чисел отбивают.

Знаки углового градуса, минуты, секунды, терции от предыдущих чисел не отделяют, а от последующих отделяют пробелом (напр.: 5° 17'').

Знак градуса температуры отделяется от числа, если за ним следует сокращенное обозначение шкалы (напр., 15 °С, но 15° Цельсия).

Числа и даты. Многочисленные числа пишут арабскими цифрами и разбивают на классы (напр.: 13 692).

Не разбивают четырехзначные числа и числа, обозначающие номера.

Числа должны быть отбиты от относящихся к ним наименований (напр.: 25 м). Числа с буквами в обозначениях не разбиваются (напр.: в пункте 2б). Числа и буквы, разделенные точкой, не имеют отбивки (напр.: 2.13.6).

Основные математические знаки перед числами в значении положительной или отрицательной величины, степени увеличения от чисел не отделяют (напр.: -15, ×20).

Для обозначения диапазона значений употребляют один из способов: многоточие, тире, знак ÷, либо предлоги от ... до ... . По всему тексту следует придерживаться принципа единообразия.

Сложные существительные и прилагательные с числами в их составе рекомендуется писать в буквенно-цифровой форме (напр.: 150-летний, 30-градусный, 25-процентный).

Стандартной формой написания дат является следующая: 20.03.93 г. Возможны и другие как цифровые, так и словесно-цифровые формы: 20.03.1993 г., 22 марта 1993 г., 1 сент. 1999 г.

Все виды некалендарных лет (бюджетный, отчетный, учебный), т.е. начинающихся в одном году, а заканчивающихся в другом, пишут через косую черту: В 2011/12 учебном году. Отчетный 2011/2012 год.

Сокращения. Используемые сокращения должны соответствовать правилам грамматики, а также требованиям государственных стандартов.

Однотипные слова и словосочетания везде должны либо сокращаться, либо нет (напр.: в 1919 году и XX веке или в 1919 г. и XX в.; и другие, то есть или и др., т.е.).

Существует ряд общепринятых графических сокращений:

Сокращения, употребляемые самостоятельно: и др., и пр., и т.д., и т.п.

Употребляемые только при именах и фамилиях: г-н, т., им., акад., д-р., доц., канд. физ.-мат. наук, ген., чл.-кор. Напр.: доц. Иванов И.И.

Слова, сокращаемые только при географических названиях: г., с., пос., обл., ул., просп. Например: в с. Н. Павловка.

Употребляемые при ссылках, в сочетании с цифрами или буквами: гл.5, п.10, подп.2а, разд.А, с.54 – 598, рис.8.1, т.2, табл.10 – 12, ч.1.

Употребляемые только при цифрах: в., вв., г., гг., до н.э., г.н.э., тыс., млн., млрд., экз., к., р. Например: 20 млн. руб., 5 р. 20 к.

Используемые в тексте сокращения поясняют в скобках после первого употребления сокращаемого понятия. Напр.:... заканчивается этапом составления технического задания (ТЗ).

В пояснительной записке следует применять стандартизованные единицы физических величин, их наименования и обозначения в соответствии с ГОСТ 8.417 или ГОСТ 8.430. В качестве обозначений предусмотрены буквенные обозначения и специальные знаки, напр.: 20.5 кг, 438 Дж/(кг/К), 36 °С. При написании сложных единиц комбинировать буквенные обозначения и наименования не допускается. Наряду с единицами СИ, при необходимости, в скобках указывают единицы ранее применявшихся систем, разрешенных к применению.

### 3.4 Требования к оформлению формул.

Формулы должны быть оформлены в редакторе формул *Equation Editor* и вставлены в документ как объект.

Размеры шрифта для формул:

- обычный – 14 пт;
- крупный индекс – 10 пт;

- мелкий индекс – 8 пт;
- крупный символ – 20 пт;
- мелкий символ – 14 пт.

Значения указанных символов и числовых коэффициентов, входящих в формулу, должны быть приведены непосредственно под формулой, причем каждый символ и его размерность пишутся с новой строки и в той последовательности, в которой они приведены в формуле. Первая строка расшифровки должна начинаться со слова «где» без двоеточия после него.

*Пример:*

Урожай соломы при 19% влажности определяется по формуле:

$$Y = \frac{X(100 - B)}{81}, \quad (3.1)$$

где  $X$  – урожай соломы в поле, ц/га;

$B$  – фактическая влажность соломы, %.

Все формулы нумеруются арабскими цифрами, номер ставят с правой стороны листа на уровне формулы в круглых скобках. Номер формулы состоит из 2-х частей, разделенный точкой, например (3.1), первая часть выделена под номер раздела, вторая часть – номер формулы. Допускается нумерация формул в пределах пояснительной записки.

При переносе формулы номер ставят напротив последней строки в край текста. Если формула помещена в рамку, номер помещают вне рамки против основной строки формулы.

Группа формул, объединенных фигурной скобкой, имеет один номер, помещаемый точно против острия скобки.

При ссылке на формулу в тексте ее номер ставят в круглых скобках. *Например:* Из формулы (3.1) следует...

В конце формулы и в тексте перед ней знаки препинания ставят в соответствии с правилами пунктуации. Формулы, следующие одна за другой, отделяют запятой или точкой с запятой, которые ставят за формулами до их номера. Переносы формул со строки на строку осуществляются в первую очередь на знаках отношения (=; ≠; ≥, ≤ и т.п.), во вторую – на знаках сложения и вычитания, в третью – на знаке умножения в виде косога креста. Знак следует повторить в начале второй строки. Все расчеты представляются в системе СИ.

### 3.5 Требования к оформлению иллюстраций.

Иллюстрации, сопровождающие пояснительную записку, могут быть выполнены в виде диаграмм, номограмм, графиков, чертежей, карт, фотоснимков и др. Указанный материал выполняется на формате А4, т.е. размеры иллюстраций не должны превышать формата страницы с учетом полей. Если ширина рисунка больше 8 см, то его располагают симметрично посередине. Если его ширина менее 8 см, то рисунок, как правило, располагают с краю, в обрамлении текста.

Допускается размещение нескольких иллюстраций на одном листе. Иллюстрации могут быть расположены по тексту пояснительной записки, а также даны в приложении.

Сложные иллюстрации могут выполняться на листах формата А3 и больше со сгибом для размещения в пояснительной записке.

Все иллюстрации нумеруются в пределах текста арабскими буквами (если их более одной). Нумерация рисунков может быть как сквозной, например, *Рисунок 1*, так и индексационной (по главам пояснительной записки, например, *Рисунок 3.1*).

Иллюстрации могут иметь, при необходимости, наименование и экспликацию (поясняющий текст или данные). Наименование помещают под иллюстрацией, а экспликацию под наименованием. В тексте, где идет речь о теме, связанной с иллюстрацией, помещают ссылку либо в виде заключенного в круглые скобки выражения (*рисунок 4*) либо в виде оборота типа «...как это видно на рисунке 4».

При оформлении графиков оси (абсцисс и ординат) вычерчиваются сплошными линиями. На концах координатных осей стрелок не ставят (рисунок 4). Числовые значения

масштаба шкал осей координат пишут за пределами графика (левее оси ординат и ниже оси абсцисс). По осям координат должны быть указаны условные обозначения и размерности отложенных величин в принятых сокращениях. На графике следует писать только принятые в тексте условные буквенные обозначения. Надписи, относящиеся к кривым и точкам, оставляют только в тех случаях, когда их немного, и они являются краткими. Многословные надписи заменяют цифрами, а расшифровку приводят в подрисуночной подписи.

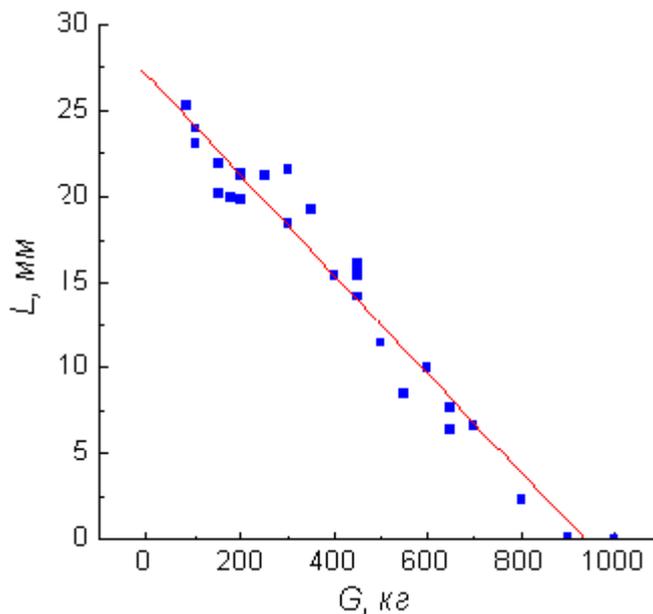


Рисунок 2 - Зависимость веса груза от линейных размеров

Схемы выполняют без соблюдения масштаба и пространственного расположения.

Иллюстрации должны быть вставлены в текст одним из следующих способов:

- либо командами ВСТАВКА-РИСУНОК (используемые для вставки рисунков из коллекции, из других программ и файлов, со сканера, созданные кнопками на панели рисования, автофигуры, объекты Word Art, а так же диаграммы). При этом все иллюстрации, вставляемые как рисунок, должны быть преобразованы в формат графических файлов, поддерживаемых Word;
- либо командами ВСТАВКА-ОБЪЕКТ. При этом необходимо, чтобы объект, в котором создана вставляемая иллюстрация, поддерживался редактором Word стандартной конфигурации.

### 3.6 Требования к оформлению таблицы.

Цифровой материал принято помещать в таблицы. Таблицы помещают непосредственно после абзацев, содержащих ссылку на них, а если места недостаточно, то в начале следующей страницы.

Все таблицы должны быть пронумерованы. Все таблицы нумеруются в пределах раздела арабскими цифрами. Номер таблицы состоит из номера раздела и порядкового номера, разделенного точкой. Название следует помещать над таблицей без значка № перед цифрой и точки после нее. Например, Таблица 1.1 — Себестоимость 1 т портландцемента М-500. Допускается сквозная нумерация в пределах пояснительной записки. Таблицы снабжают тематическими заголовками, которые располагаются посередине страницы и пишут прописным шрифтом без точки на конце. Заголовок и слова таблица начинают писать с прописной буквы. Высота таблицы с записями в одну строку должна быть не более 8 мм. Если в таблице встречается повторяющийся текст, то при первом же повторении допускается писать слово «то же», а далее кавычками ( ” ). Ставить кавычки вместо повторяющихся цифр, марок, знаков, символов не допускается. Если цифровые или текстовые данные не приводятся в какой-либо строке таблицы, то на ней

ставят прочерк (-). Цифры в графах таблиц располагают так, чтобы они следовали одни под другими.

Таблица 1 — Себестоимость 1т продукции в ОАО «Стройцемент»

Вид продукции	Себестоимость 1 т продукции, руб.			2008 г. в % к	
	2006 г.	2007 г.	2008 г.	2006 г.	2007 г.
Щебень	643.5	1093.8	1245.3	193.5	113.9
Портландцемент, М-500	5201.5	6224.1	8192.8	157.5	131.6

При переносе таблицы на другой лист заголовок помещают над первой частью, над последующими пишут надписи «*продолжение таблицы 1.1*». Единственная таблица не нумеруется. Сноски к таблице печатают непосредственно под ней.

### 3.7 Оформление списка использованных источников.

Сведения о книгах (монографии, учебники, справочники и т.п.) должны включать: фамилию и инициалы автора (авторов), название книги, город, издательство, год издания, количество страниц.

При наличии трех и более авторов допускается указывать фамилию и инициалы только первого из них и слова «и др.». Наименование места издания необходимо приводить полностью в именительном падеже, допускается сокращение названия только двух городов – Москва (М) и Санкт-Петербург (СПб).

Сведения о статье из периодического издания должны включать: фамилию и инициалы автора, заглавие статьи, наименование издания (журнала), наименование серии, год выпуска, том, номер издания (журнала), страницы, на которых помещена статья.

Сведения об отчете по НИР должны включать: заглавие отчета (после заглавия в скобках приводят слово «отчет»), его шифр, инвентарный номер, наименование организации, выпустившей отчет, фамилию и инициалы руководителя НИР, город и год выпуска, количество страниц отчета.

Сведения о стандарте должны включать: обозначение и наименование стандарта.

*Примеры:*

#### ***Книги одного, двух, трех авторов***

1. Кондраков Н.П. Иванова М. А.. Бухгалтерский управленческий учет: Учебное пособие– М.: ИНФРА-М.- 2011. – 368 с.
2. Головизнова А.Т., Архипова О.И. Бухгалтерский управленческий учет: Учеб. Пособие.-М.: ТК Велби, Изд-во Проспект, 2004.-184с.
3. Иванов В.Б. Автоматизированное управление запасами предприятия / Куликов Г.Г., Речкалов Я.А. — Уфимск. гос. Авиац. Техн. ун-т. -Уфа, 2002.-104с.
4. Хоршреп И.Т., Фостер Дж. Бухгалтерский учет: управленческий аспект:Пер.с англ./Под ред. Я.В.Соколова. — М.: Финансы и статистика, 2000
5. Flanaut, J. Les elements des terres rares / J. Flanaut. — Paris: Masson, 1969. — 165 р.

#### ***Книги четырех и более авторов, а также сборники статей***

6. Бойко Е.А. Бухгалтерский управленческий учет / Е.А. Бойко и др. - Под ред. Проф., д.э.н. А.Н. Кизлова, проф., д.э.н. И.Н. Богатой. - Ростов н/Д: «Феникс», 2005. - 380с.
7. Финансы предприятий: Учебник для вузов / Н.В. Колчина, Г.Б. Поляк, Л.П. Павлова и др., Под ред. проф.Н.В. Колчиной. - 2-е изд.перераб.и доп.-М.:ЮНИТИ-ДИНА, 2001. - 447с.
8. Бухгалтерский учет. Учебник для вузов / БезрукихП.С., Ивашкевич В.Б., Кондраков Н.П.и др.под ред. П.С. Безруких 3-е изд., перераб.и доп.-М. : Бухгалтерский учет, 1999-624 с.
9. Experiments in materials science/ E.C. Subbarac, D. Chakravorty, M.F. Merriam, V. Raghavan. — New York a.c: Mc Graw-Hill, 1972. — 274 p.

### **Статьи из журналов и газет**

10. Волков Н.Г. Учет производственных запасов // Бухгалтерский учет, № 4, 2001, с.5-17.
11. Голубева М., Лялин Р. Метод ЛИФО в бухгалтерском учете отменен. Разбираемся с последствиями // Главбух, 2008 г. №2, с.15-17.
12. Давыдова О.В. Составляющие стоимости приобретенных материалов // Автономные организации: бухгалтерский учет и налогообложение, 2008г.-№10, с.25-27.
13. Из истории хлебопечения // Хлебопродукты, 2002г. №2, с.44-45
14. Кабанов А. Как управлять запасами // Финансовый директор. 2004.-№2-с.31
15. Лумпов Н.А. Исследование методов оценки стоимости запасов / Н.А. Лумпов // Финансовый менеджмент. - 2004.-№3-с.19-41
16. Mukai, K. Determination of phosphorus in hypereutectic aluminium-silicon alloys/ К. Mukai // Talanta. — 1972.-Уо1. 19, № 4 — P. 489-495.

### **Статья из продолжающегося издания**

17. Живописцев, В.П. Комплексный анализ в сельском хозяйстве / В.П. Живописцев, Л.П. Пятосин // Учен. зап. / Перм. ун-т. — 1970. — № 207. — С. 184-191.

### **Статьи из неперидических сборников**

18. Любомилова, Г.В. Анализ оборотных средств / Г.В. Любомилова, А.Д. Миллер // Новые метод, исслед. по экономическому анализу. — М., 1970. — С. 90-93.
19. Маркович, Дж. Анализ циклов в экономике / Дж. Маркович, А. Кертеc // Проблемы мировой экономики: Докл. Межд. конф., Гетеборг, Швеция, 27 авг. — 1 сент. 1966. — М., 1971. — С. 223-231.

### **Диссертация**

20. Матчинов, В.А. Бухгалтерский учет и анализ эффективности использования государственной помощи в сельскохозяйственных организациях: Дис.экон. наук: 08.00.12 / В.А. Матчинов. — Москва, 2009. — 138 с.

### **Автореферат диссертации**

21. Матчинов, В.А. Бухгалтерский учет и анализ эффективности использования государственной помощи в сельскохозяйственных организациях: Автореф. дис.канд. экон. наук: 08.00.12 / В.А. Матчинов. — Москва, 2009. — 20 с.

### **Депонированные научные работы**

22. Крылов, А.В. Потери минеральных удобрений при перевозке/ А.В. Крылов, В.В. Бабкин; Редкол. «Журн. прикладной химии». — Л., 1982. — 11 с. — Деп. в ВИНТИ 24.03.82; № 1286-82.
23. Кузнецов, Ю.С. Теплотехнические свойства элементов в холодильных конструкциях / Ю. С. Кузнецов; Моск. хим.-технол. ин-т. — М., 1982. — 10 с. — Деп. в ВИНТИ 27.05.82; № 2641.

### **Патентные документы**

24. А.с. 1007970 СССР, МКИ4 В 03 С 7/12, А 22 С 17/04. Устройство для разделения многокомпонентного сырья / Б.С. Бабакин, Э.И. Каухчешвили, А.И. Ангелов (СССР). — № 3599260/28-13; Заявлено 2.06.85; Опубл. 30.10.85, Бюл. № 28. — 2 с.
25. Пат. 4194039 США, МКИ3 В 32 В 7/2, В 32 В 27/08. Multi-layer poivolefin shrink film / W.B. Muelier; W.R. Grace & Co. — № 896963; Заявлено 17.04.78; Опубл. 18.03.80. — 3 с.
26. Заявка 54-161681 Япония, МКИ2 В 29 D 23/18. Способ изготовления гибких трубок / Йосиаки Инаба; К.К. Тое Касэй. — № 53-69874; Заявлено 12.06.78; Опубл.21.12.79. — 4 с.

### **Стандарт**

27. ГОСТ 10749.1-80. Спирт этиловый технический. Методы анализа. — Взамен ГОСТ 10749-72; Введ. 01.01.82 до 01.01.87. — М.: Изд-во стандартов, 1981. — 4 с.

28. Отчет о НИР. Проведение испытания свойств полимерных покрытий тепличных сооружений — 12-ВЗ: Отчет о НИР (промежуточ.) / Всесоюз. заоч. ин-т пищ. пром-сти (ВЗИПП); Руководитель В. М. Шавра. — ОЦО 102ТЗ; Кг ГР 80057138; Инв.№Б119699.-М., 1981. — 90 с.

#### *Электронные ресурсы*

29. Н.И. Кубракова, О.М. Васильева; под ред. Н.И. Размариловой. — Электрон. текстовые дан. (1 файл). — Томск, 2004. — Режим доступа: <http://www.lib.tru.ru/fullex/m/2004/m26.pdf>, свободный. — Загл. с экрана.

30. Российская государственная библиотека [Электронный ресурс]/Центр информ. Технологий РГБ; ред. Власенко Т.В.; Wed-мастер Козлова Н.В. — Электрон. Дан. — М.: Рос.гос. б.ка, 1977 — Режим доступа: <http://www.rsb.ru>, свободный. — Загл. с экрана.

#### *Реферат*

31. [Реферат]// Химия: РЖ. — 1981. — № 1, вып. 19С — С. 38 (1 С138). Реф. Ст.: Richardson, S. M. Simulation of injection moulding / S.M. Richardson, H.J. Pearson, J.R.A. Pearson// Plast and Rubber: Process. — 1980. — Vol. 5, № 2. — P. 55 — 60.

#### *Рецензия*

Гаврилов А.В. Как звучит?/Андрей Гаврилов—Кн.обозрение. — 2002. 11 марта (№10-11). — С. 2 — Рец. на кн.: Музыкальный запас. 70-е: Проблемы, портреты, случаи/Т. Чередниченко. — М.: Новое лит. Обозрение, 2002 — 592 с.

## **4. ПРИМЕР ПРОЕКТИРОВАНИЯ РЕЛЯЦИОННОЙ БАЗЫ ДАННЫХ**

В качестве примера возьмем базу данных проектной организации. Основной вид деятельности такой организации — выполнение проектов по договорам с заказчиками.

### **4.1. Анализ предметной области**

#### **4.1.1.Общее описание предметной области**

База данных создаётся для информационного обслуживания руководства организации, руководителей проектов и участников проектов. БД должна содержать данные об отделах организации, сотрудниках и проектах.

В соответствии с предметной областью система строится с учётом следующих особенностей:

- Каждый сотрудник работает в определённом отделе, в каждом отделе могут работать несколько сотрудников.
- Каждый проект относится к определённому отделу, каждый отдел может отвечать за выполнение нескольких проектов.
- Каждый сотрудник может принимать участие в выполнении нескольких проектов, над каждым проектом может трудиться несколько сотрудников.
- Для каждого проекта назначается руководитель из числа сотрудников того отдела, к которому относится проект.
- Каждый проект должен быть выполнен в заданные сроки, каждый проект может состоять из нескольких этапов. Если проект состоит из одного этапа, то сроки его выполнения должны совпадать со сроками выполнения проекта в целом.
- Оклад сотрудника зависит от занимаемой должности, за участие в проектах сотрудник получает дополнительное вознаграждение.
- Виды участия сотрудников в проектах: руководитель, консультант, исполнитель.
- Каждый отдел занимает одно или несколько помещений (комнат), в каждом помещении может быть один или несколько стационарных телефонов.

**Примечание.** Описания особенностей Про должно быть достаточно для того, чтобы создать ER–диаграмму.

#### 4.1.2. Анализ информационных задач и круга пользователей системы

Определим группы пользователей, их основные задачи и запросы к БД:

1. Руководители организации:
  - заключение новых договоров;
  - назначение руководителей проектов;
  - получение списка всех участников проектов;
  - изменение должностных окладов и штатного расписания;
  - получение полной информации о проектах;
  - внесение изменений в данные о проектах;
  - архивирование данных по завершённым проектам.

**Примечание.** Архивирование данных в этом пособии подробно не рассматривается. Это сделано для того, чтобы не перегружать схему БД.

2. Руководитель проекта:
  - назначение участников проекта;
  - получение списка сотрудников, работающих над конкретным проектом;
  - получение полной информации о проекте, руководителем которого он является;
  - получение сведений о сотрудниках, которые могут стать участниками проекта;
  - определение размера дополнительного вознаграждения сотрудников по конкретному проекту;
  - внесение изменений в данные об этапах проекта.
3. Сотрудники отдела кадров:
  - приём/увольнение сотрудников;
  - внесение изменений в данные о сотрудниках.
4. Бухгалтеры:
  - получение ведомости на выплату зарплаты.
5. Сотрудники – участники проектов:
  - просмотр данных о других участниках проекта;
  - просмотр данных о сроках сдачи проекта и форме отчётности.

### 4.2. Разработка БД

#### 4.2.1. Построение концептуальной модели

Для создания ER-модели необходимо выделить сущности предметной области:

- 1) **Отделы.** Атрибуты: название, аббревиатура, комнаты, телефоны.
- 2) **Сотрудники.** Атрибуты: ФИО, паспортные данные, дата рождения, пол, ИНН (индивидуальный номер налогоплательщика), номер пенсионного страхового свидетельства, адреса, телефоны (рабочий, домашний, мобильный), данные об образовании (вид образования (высшее, средне-специальное и т.д.), специальность, номер диплома, дата окончания учебного заведения), должность, оклад, логин (имя пользователя).

**Примечания:** 1. Логин потребуется нам для назначения дифференцированных прав доступа.

2. В нашем задании не предусмотрена полная информационная поддержка сотрудников отдела кадров, поэтому мы не будем отражать в БД такие сведения как дату поступления сотрудника на работу, его переводы с одной должности на другую, уходы в отпуска и т.п.

- 3) **Проекты.** Атрибуты: номер договора; полное название проекта; сокращённое название проекта; дата подписания договора; заказчик; контактные данные

заказчика; дата начала проекта; дата завершения проекта; сумма по проекту; дата реальной сдачи проекта; сумма, полученная по проекту на текущую дату.

- 4) **Этапы проекта.** Атрибуты: номер по порядку, название, дата начала этапа, дата завершения этапа, форма отчетности, сумма по этапу, дата реальной сдачи этапа; сумма, полученная по этапу на текущую дату.

Исходя из выявленных сущностей, построим ER–диаграмму (рис. 3). Напомним, что пометки у линий означают степень связи: 1:1, 1:N и N:M.

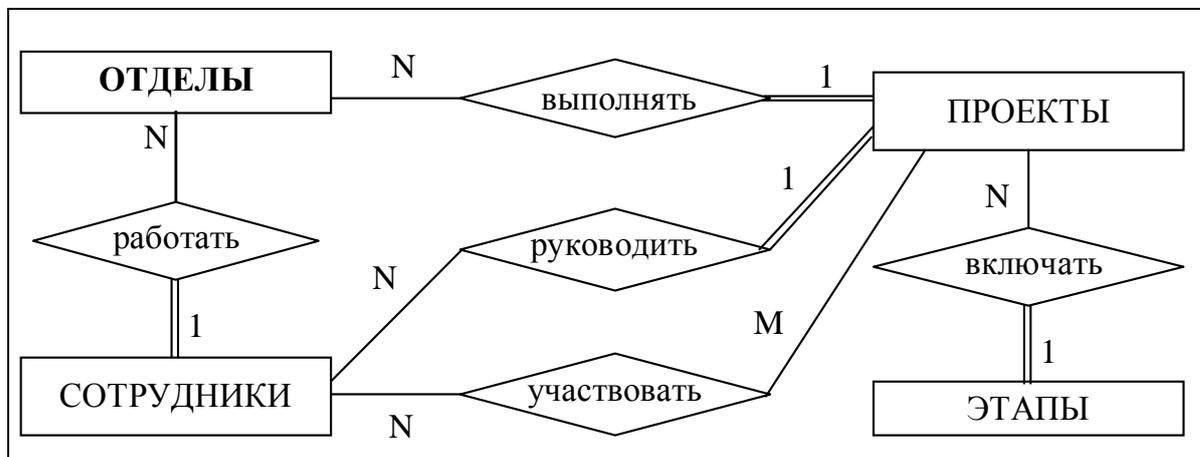


Рис. 3. ER–диаграмма ПрО «Проектная организация»

#### 4.2.2. Определение требований к операционной обстановке

Для выполнения этого этапа необходимо знать (хотя бы ориентировочно) объём работы организации (т.е. количество проектов и сотрудников), а также иметь представление о характере и интенсивности запросов.

Объём внешней памяти, необходимый для функционирования системы, складывается из двух составляющих: память, занимаемая модулями СУБД (ядро, утилиты, вспомогательные программы), и память, отводимая под данные ( $M_d$ ). Для реальных баз данных обычно наиболее существенным является  $M_d$ .

На основе результатов анализа ПрО можно приблизительно оценить объём памяти, требуемой для хранения данных. Примем ориентировочно, что:

- одновременно осуществляется около десяти проектов, работа над проектом продолжается в среднем год (по 1К на каждый проект);
- каждый проект состоит в среднем из четырёх этапов (по 0,5К на этап);
- в компании работают 100 сотрудников (по 0,5К на каждого сотрудника);
- в выполнении каждого проекта в среднем участвуют 10 сотрудников (по 0,2К);
- устаревшие данные переводятся в архив (накапливаются в архиве БД).

Тогда объём памяти для хранения данных за первый год примерно составит:

$$M_d = 2(10 \cdot 1 + 10 \cdot 4 \cdot 0,5 + 100 \cdot 0,5 + (10 \cdot 10 \cdot 0,2)) = 200 \text{ К,}$$

Коэффициент 2 необходим для того, чтобы учесть необходимость выделения памяти под дополнительные структуры (например, индексы). Объём памяти будет увеличиваться ежегодно на столько же при сохранении объёма работы.

Требуемый объём оперативной памяти определяется на основании анализа интенсивности запросов и объёма результирующих данных. Для нашей БД требуемый объём памяти мал, поэтому никаких специальных требований к объёму внешней и оперативной памяти компьютера не предъявляется.

### 4.2.3. Выбор СУБД и других программных средств

Анализ информационных задач показывает, что для реализации требуемых функций подходят почти все СУБД для ПЭВМ (MS Access, Firebird, MySQL и др.). Все они поддерживают реляционную модель данных и предоставляют разнообразные возможности для работы с данными.

Объём внешней и оперативной памяти, требующийся для функционирования СУБД, обычно указывается в сопроводительной документации.

Для того чтобы в учебном примере не привязываться к конкретной СУБД, выполним описание логической схемы БД на SQL-92.

**Примечание.** При выполнении курсового проекта необходимо обосновать выбор конкретной СУБД для реализации проекта и реализовать базу данных под управлением выбранной СУБД.

### 4.2.4. Логическое проектирование реляционной БД

#### Определение связей логической модели БД

База данных создаётся на основании схемы базы данных. Для преобразования ER-диаграммы в схему БД приведём уточнённую ER-диаграмму, содержащую атрибуты сущностей (рис. 4).

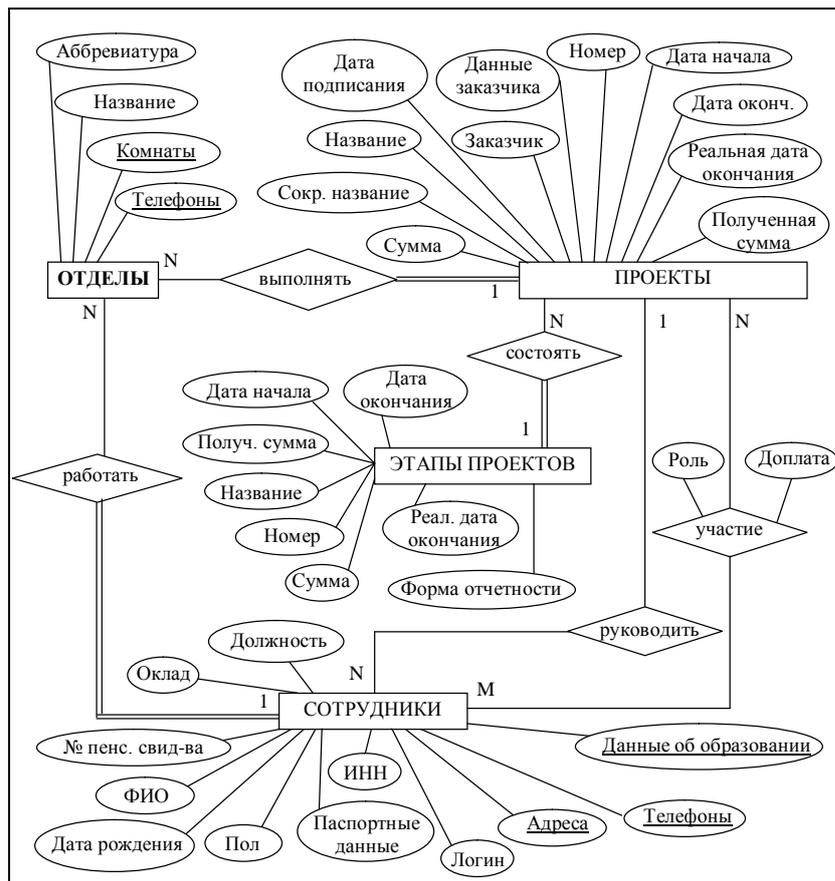


Рис. 4. Уточнённая ER-диаграмма проектной организации

**Примечание.** Многочисленные атрибуты на рисунке выделены подчеркиванием.

Преобразование ER-диаграммы в схему БД выполняется путем сопоставления каждой сущности и каждой связи, имеющей атрибуты, отношения (таблицы) БД. Связь типа 1:n (один-ко-многим) между отношениями реализуется через внешний ключ. Ключ вводится для того отношения, к которому осуществляется множественная связь. Внешнему

ключу должен соответствовать первичный или уникальный ключ основного (родительского) отношения.

Связь *участвовать* между *ПРОЕКТАМИ* и *СОТРУДНИКАМИ* принадлежит к типу n:m (многие-ко-многим). Этот тип связи реализуется через вспомогательное отношение *Участие*, которое содержит комбинации первичных ключей соответствующих исходных отношений.

Для схемы БД будем использовать обозначения, представленные на рис. 5.

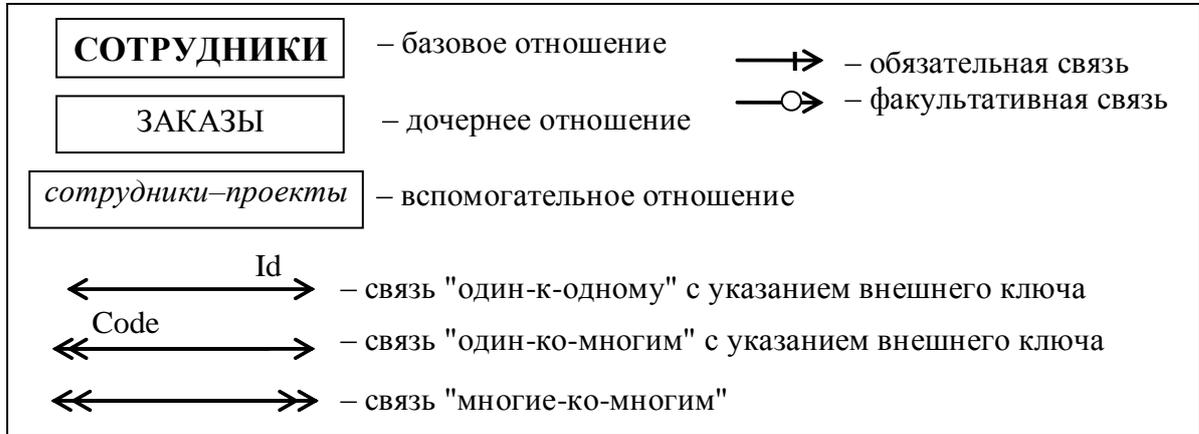


Рис. 5. Обозначения, используемые на схеме базы данных

Полученная схема реляционной базы данных (РБД) приведена на рис. 6.

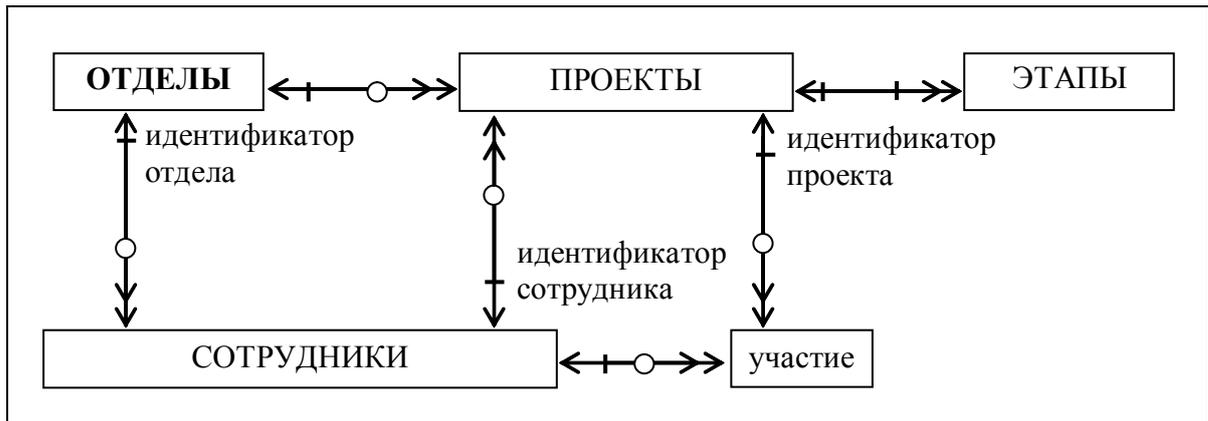


Рис. 6. Схема РБД, полученная из ER-диаграммы проектной организации

Бинарная связь между отношениями не может быть обязательной для обоих отношений. Такой тип связи означает, что, например, прежде чем добавить новый проект в отношение *ПРОЕКТЫ*, нужно добавить новую строку в отношение *ЭТАПЫ*, и наоборот. Поэтому для такой связи необходимо снять с одной стороны условие обязательности. Так как все эти связи будут реализованы с помощью внешнего ключа, снимем условие обязательности связей для отношений, содержащих первичные ключи.

Схема на рис. 5 содержит три цикла: "сотрудники-проекты-участие-сотрудники", "отделы-сотрудники-проекты-отделы" и "отделы-сотрудники-участие-проекты-отделы". Цикл допустим только в том случае, если связи, входящие в него, независимы друг от друга. Например, для нашей ПрО справедливо такое правило: сотрудник любого отдела может быть участником (исполнителем или консультантом) проекта любого отдела. Эти связи независимы, поэтому цикл "отделы-сотрудники-участие-проекты-отделы" не будет приводить к нарушению логической целостности данных.

С другой стороны, только сотрудник отдела, отвечающего за выполнение проекта, может быть руководителем проекта. Но система не помешает нам назначить руководителем

проекта сотрудника любого отдела. При добавлении проекта с внешним ключом Руководитель система проверит только, что такой человек есть в таблице СОТРУДНИКИ. А значение внешних ключей Отдел в таблицах СОТРУДНИКИ и ПРОЕКТЫ сравнивать не будет.

Таким образом, остальные циклы могут приводить к возможности нарушения логической целостности данных. Существует несколько подходов для разрешения ситуаций, в которых связи, входящие в цикл, зависят друг от друга.

Рассмотрим эту ситуацию в общем случае. Сначала слегка упростим схему: реализуем связь "руководить" через таблицу УЧАСТИЕ – это позволит не отвлекаться на малозначительные детали.

Будем считать, что в выполнении проекта могут участвовать только сотрудники, работающие в том же отделе, к которому относится проект (рис. 6,а). При циклической схеме СУБД не сможет гарантировать логическую целостность данных без использования дополнительных средств.

Один из способов разрешения таких ситуаций – разорвать цикл, исключив одну из связей (рис. 6,б) или введя промежуточное отношение (рис. 6,в). В нашем случае можно было бы разорвать связь "сотрудники–проекты", если бы каждый сотрудник участвовал во всех проектах своего отдела. Промежуточное отношение можно было бы использовать, если бы существовала общая связь между сущностями, входящими в цикл. Например, если бы каждый сотрудник заключал договор с отделом на выполнение работ в рамках проекта, то отношение ДОГОВОРЫ отражало бы связь между отделом, сотрудником и проектом.

Другой способ разрешения цикла заключается в том, что в промежуточное отношение СОТРУДНИКИ – ПРОЕКТЫ, которое реализует связь многие-ко-многим, добавляются (мигрируют) внешние ключи Код отдела (D\_id) из отношений СОТРУДНИКИ и ПРОЕКТЫ (рис. 7,г). Эти ключи проверяются на равенство друг другу с помощью соответствующего ограничения целостности (check). Использование этого способа возможно в том случае, когда соответствующие связи (отдел–проект и отдел–сотрудник) имеют тип один-ко-многим и являются обязательными.

В тех ситуациях, когда все эти способы непригодны, логическая целостность контролируется программно или вручную. Если принято решение переложить обязанности по контролю за логической целостностью данных на пользователя, то эти обязанности должны быть отражены в документации (в руководстве пользователя).

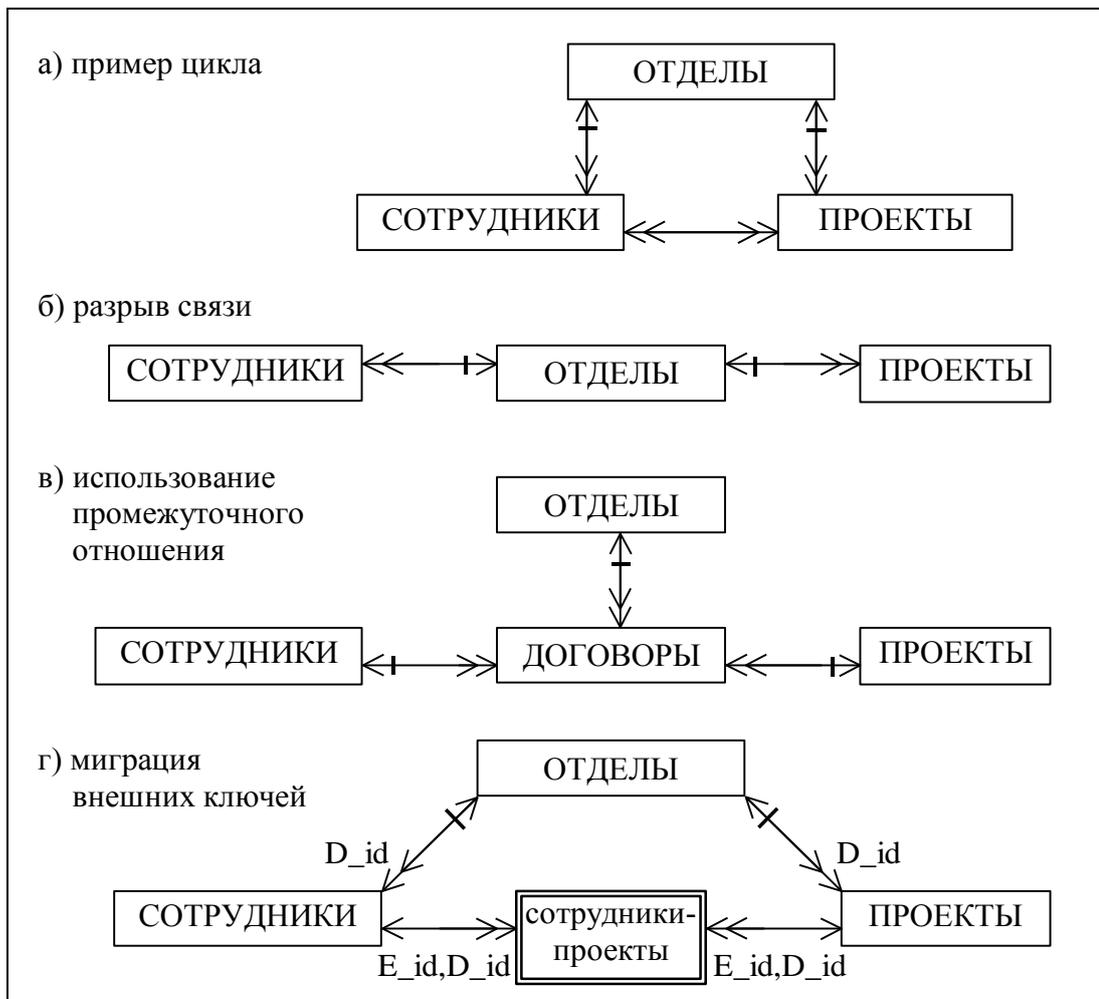


Рис.7. Некоторые способы разрешения циклов в схеме базы данных

Примем для нашей ПрО, что руководитель проекта может одновременно выполнять и другие обязанности в этом проекте, чтобы цикл "сотрудники–проекты–участие–сотрудники" не приводил к возможности нарушения логической целостности данных. Зато цикл "отделы–сотрудники (руководители)–проекты–отделы" включает зависимые связи: руководитель проекта назначается из того отдела, который отвечает за выполнение проекта в целом. Здесь можно было бы применить разрыв связи "отделы–проекты" и определять, к какому отделу относится проект через руководителя (по отделу руководителя проекта). Но такой подход в данном случае имеет существенный недостаток. Заменив руководителя проекта сотрудником другого отдела, можно одновременно изменить отдел, отвечающий за выполнение проекта, т.е. объединить в одно действие два независимых изменения, а это недопустимо.

Исходя из вышесказанного мы не будем разрывать связь, а примем решение реализовать эту проверку программно. Приложение должно будет при назначении руководителя проекта выдавать список сотрудников того отдела, который отвечает за выполнение данного проекта. Руководителя можно будет выбрать только из этого списка, а не вводить вручную.

#### *Добавление атрибутов в логическую модель БД*

Каждое реляционное отношение соответствует одной сущности (объекту ПрО) и в него вносятся все атрибуты этой сущности. Для каждого отношения определяются первичный ключ и внешние ключи (в соответствии со схемой БД). В том случае, если базовое отношение не имеет потенциальных ключей, вводится *суррогатный первичный ключ*, который не несёт смысловой нагрузки и служит только для идентификации записей.

Отношения приведены в табл. 2-6. Для каждого отношения указаны атрибуты с их внутренним названием, типом и длиной. Типы данных обозначаются так: N – числовой, С – символьный тип фиксированной длины, V – символьный тип переменной длины, D – дата (этот тип имеет стандартную длину, зависящую от СУБД, поэтому она не указывается).

Потенциальными ключами отношения ОТДЕЛЫ являются атрибуты Аббревиатура и Название отдела. Первый занимает меньше места, поэтому мы выбираем его в качестве первичного ключа.

Таблица 2. Схема отношения ОТДЕЛЫ (Departs)

Содержание поля	Имя поля	Тип, длина	Примечания
Аббревиатура отдела	D_ID	C(10)	<b>первичный ключ</b>
Название отдела	D_NAME	V(100)	обязательное поле
Комнаты	D_ROOMS	V(20)	обязательное многозначное поле
Телефоны	D_PHONE	V(40)	обязательное многозначное поле

Потенциальными ключами отношения СОТРУДНИКИ являются поля Паспортные данные, ИНН и Номер страхового пенсионного свидетельства. Все они занимают достаточно много места, а паспортные данные кроме того могут меняться. Введём суррогатный первичный ключ Номер сотрудника.

Таблица 3. Схема отношения СОТРУДНИКИ (Employees)

Содержание поля	Имя поля	Тип, длина	Примечания
Номер	E_ID	N(4)	<b>суррогатный первичный ключ</b>
Фамилия, имя, отчество	E_NAME	V(50)	обязательное поле
Дата рождения	E_BORN	D	обязательное поле
Пол	E_SEX	C(1)	обязательное поле, 'м' или 'ж'
Паспортные данные	E_PASP	V(50)	обязательное поле
ИНН	E_INN	C(12)	обязательное уникальное поле
Номер пенсионного страхового свидетельства	E_PENS	C(14)	обязательное уникальное поле
Отдел	E_DEPART	C(10)	внешний ключ (к Departs)
Должность	E_POST	V(30)	обязательное поле
Оклад	E_SAL	N(8,2)	обязательное поле, > 4500 руб.
Данные об образовании	E_EDU	V(200)	обязательное многозначное поле
Адреса	E_ADDR	V(100)	многозначное поле
Телефоны	E_PHONE	V(30)	многозначное поле
Логин	E_LOGIN	V(30)	

**Примечание.** Суррогатный первичный ключ также может вводиться в тех случаях, когда потенциальный ключ имеет большой размер (например, длинная символьная строка) или является составным (не менее трёх атрибутов).

В отношении ПРОЕКТЫ три потенциальных ключа: Номер проекта, Название проекта и Сокращённое название. Меньше места занимает первый из них, но он малоинформативен. Зато сокращённое название, используемое в качестве внешнего ключа в других таблицах, позволит специалисту идентифицировать проект без необходимости соединения с отношением ПРОЕКТЫ.

Таблица 4. Схема отношения ПРОЕКТЫ (Projects)

Содержание поля	Имя поля	Тип, длина	Примечания
Номер проекта	P_ID	N(6)	обязательное уникальное поле

Название проекта	P_TITLE	V(100)	обязательное поле
Сокращённое название	P_ABBR	C(10)	<b>первичный ключ</b>
Отдел	P_DEPART	C(10)	внешний ключ (к Departs)
Заказчик	P_COMPANY	V(40)	обязательное поле
Данные заказчика	P_LINKS	V(200)	обязательное поле
Руководитель	P_CHIEF	N(4)	внешний ключ (к Employees)
Дата начала проекта	P_BEGIN	D	обязательное поле
Дата окончания проекта	P_END	D	обязательное поле, больше даты начала проекта
Реальная дата окончания	P_FINISH	D	
Стоимость проекта	P_COST	N(10)	обязательное поле
Полученная сумма	P_SUM	N(10)	обязательное поле, значение по умолчанию – 0

Потенциальным ключом отношения ЭТАПЫ является комбинация внешнего ключа и номера этапа, а потенциальным ключом вспомогательного отношения УЧАСТИЕ является комбинация первых трёх полей этого отношения. Можно вообще не вводить первичный ключ для данных отношений, т.к. на них никто не ссылается. Но уникальность этих комбинации является в данном случае ограничением целостности данных, поэтому мы возьмём эти комбинации в качестве первичных ключей соответствующих отношений.

Таблица 5. Схема отношения ЭТАПЫ ПРОЕКТА (Stages)

Содержание поля	Имя поля	Тип, длина	Примечания	
Проект	S_PRO	C(10)	внешний ключ (к Projects)	<b>составной первичный ключ</b>
Номер этапа	S_NUM	N(2)		
Название этапа	S_TITLE	V(200)	обязательное поле	
Дата начала этапа	S_BEGIN	D	обязательное поле	
Дата окончания этапа	S_END	D	обязательное поле, > даты начала	
Реальная дата окончания	S_FINISH	D	больше даты начала этапа	
Стоимость этапа	S_COST	N(10)	обязательное поле	
Полученная сумма по этапу	S_SUM	N(10)	обязательное поле, значение по умолчанию – 0	
Форма отчётности	S_FORM	V(100)	обязательное поле	

Таблица 6. Схема отношения УЧАСТИЕ (Job)

Содержание поля	Имя поля	Тип, длина	Примечания *
Проект	J_PRO	C(10)	внешний ключ (к Projects)
Сотрудник	J_EMP	N(4)	внешний ключ (к Employees)
Роль	J_ROLE	V(20)	обязательное поле
Доплата	J_BONUS	N(2)	

\* – в отношении УЧАСТИЕ первичный ключ состоит из первых 3-х полей этого отношения.

#### *Нормализация полученных отношений (до 4НФ)*

Механизм нормализации подразумевает определённую последовательность преобразования отношений к третьей нормальной форме. Мы не будем чётко придерживаться этой последовательности, т.к. она избыточна, и многозначные атрибуты сразу вынесем в отдельные отношения на первом же этапе.

**1НФ.** Для приведения таблиц к 1НФ требуется составить прямоугольные таблицы (одно значение атрибута – одна ячейка таблицы) и разбить сложные атрибуты на простые.

**Примечание.** В реальных БД сложные атрибуты разбиваются на простые, если:

- а) этого требует внешнее представление данных;
- б) в запросах поиск может осуществляться по отдельной части атрибута.

Разделим атрибут *Фамилия, имя, отчество* на два атрибута *Фамилия* и *Имя, отчество*, *Паспортные данные* на *Номер паспорта* (уникальный), *Дата выдачи* и *Кем выдан*, а *Данные об образовании* – на *Вид образования*, *Специальность*, *Номер диплома* и *Год окончания учебного заведения*.

Многозначные атрибуты *Комнаты* и *Телефоны* из отношения *ОТДЕЛЫ* вынесем в отдельное отношение *КОМНАТЫ*, а домашние и мобильные телефоны и адреса сотрудников – в отношение *АДРЕСА-ТЕЛЕФОНЫ*. Так как в комнате может не быть телефона, первичный ключ отношения *КОМНАТЫ* не определен (ПК не может содержать null-значения), но на этих атрибутах можно определить составной уникальный ключ. В отношении *АДРЕСА-ТЕЛЕФОНЫ* также нет потенциальных ключей: оставим это отношение без первичного ключа, т.к. на это отношение никто не ссылается. Данные об образовании сотрудников также вынесем в отдельное отношение.

Что касается рабочих телефонов сотрудников, то один из этих номеров – основной – определяется рабочим местом сотрудника (рассматриваются только стационарные телефоны). Будем хранить этот номер в атрибуте *Рабочий телефон*. Наличие других номеров зависит от того, есть ли в том же помещении (комнате) другие сотрудники, имеющие стационарные телефоны. Добавим в отношение *СОТРУДНИКИ* атрибут *Номер комнаты*, чтобы дополнительные номера телефонов сотрудника можно было вычислить из других кортежей с таким же номером комнаты.

Связь между отношениями *СОТРУДНИКИ* и *КОМНАТЫ* реализуем через составной внешний ключ (*Номер комнаты, Рабочий телефон*).

Мы также удалим вычисляемый атрибут *Полученная сумма* из отношения *ПРОЕКТЫ*, т.к. он является суммой значений аналогичного атрибута из отношения *ЭТАПЫ ПРОЕКТОВ*. Но атрибут *Стоимость проекта* оставим, т.к. она фигурирует в документации по проекту. А для обеспечения логической целостности данных предусмотрим в приложении проверку того, что сумма по всем этапам совпадает со стоимостью проекта.

**2НФ.** В нашем случае составные первичные ключи имеют отношения *ЭТАПЫ ПРОЕКТА* и *УЧАСТИЕ*. Неключевые атрибуты этих отношений функционально полно зависят от составных первичных ключей.

**3НФ.** В отношении *ПРОЕКТЫ* атрибут *Данные заказчика* зависит от атрибута *Заказчик*, а не от первичного ключа, поэтому его следует вынести в отдельное отношение *ЗАКАЗЧИКИ*. Но при этом первичным ключом нового отношения станет атрибут *Заказчик*, т.е. длинная символьная строка. Целесообразнее перенести в новое отношение атрибуты *Заказчик* и *Данные заказчика* и ввести для него суррогатный ПК. Так как с каждым заказчиком может быть связано несколько проектов, связь между отношениями *ПРОЕКТЫ* и *ЗАКАЗЧИКИ* будет 1:n и суррогатный ПК станет внешним ключом для отношения *ПРОЕКТЫ*.

В отношении *СОТРУДНИКИ* атрибут *Оклад* зависит от атрибута *Должность*. Поступим с этой транзитивной зависимостью так же, как в предыдущем случае: создадим отношение *ДОЛЖНОСТИ*, перенесём в него атрибуты *Должность* и *Оклад*, а первичным ключом сделаем название должности.

В отношениях *СОТРУДНИКИ* и *ОБРАЗОВАНИЕ* атрибуты (*Дата выдачи* и *Кем выдан*) и (*Номер диплома* и *Год окончания учебного заведения*) зависят не от первичного ключа, а от атрибутов соответственно *Номер паспорта* и *Специальность*. Но если мы выделим

их в отдельное отношение, то получим связи типа 1:1. Следовательно, здесь декомпозиция нецелесообразна.

**4НФ.** Отношение АДРЕСА-ТЕЛЕФОНЫ нарушают 4НФ, т.к. не всякий телефон привязан к конкретному адресу (т.е. мы имеем две многозначных зависимости в одном отношении). Но выделять Телефоны в отдельное отношение не стоит, т.к. эти сведения носят справочный характер и не требуется их автоматическая обработка.

#### 4.2.5. Разработка физической модели БД

Отношения, полученные после нормализации, приведены в табл. 7-16.

Таблица 7. Схема отношения ОТДЕЛЫ (Departs)

Содержание поля	Имя поля	Тип, длина	Примечания
Аббревиатура отдела	D_ID	V(12)	<b>первичный ключ</b>
Название отдела	D_NAME	V(100)	обязательное поле

Таблица 8. Схема отношения КОМНАТЫ (Rooms)

Содержание поля	Имя поля	Тип, длина	Примечания
Отдел	R_DEPART	V(12)	внешний ключ (к Departs)
Номер комнаты	R_ROOM	N(4)	<b>составной уникальный ключ</b>
Телефон	R_PHONE	V(20)	

Таблица 9. Схема отношения ДОЛЖНОСТИ (Posts)

Содержание поля	Имя поля	Тип, длина	Примечания
Название должности	P_POST	V(30)	<b>первичный ключ</b>
Оклад	P_SAL	N(8,2)	обязательное поле, > 4500 руб.

Таблица 10. Схема отношения СОТРУДНИКИ (Employees)

Содержание поля	Имя поля	Тип, длина	Примечания
Идентификатор сотрудника	E_ID	N(4)	<b>суррогатный первичный ключ</b>
Фамилия	E_FNAME	V(25)	обязательное поле
Имя, отчество	E_LNAME	V(30)	обязательное поле
Дата рождения	E_BORN	D	обязательное поле
Пол	E_SEX	C(1)	обязательное поле
Серия и номер паспорта	E_PASP	C(10)	обязательное уникальное поле
Когда выдан паспорт	E_DATE	D	обязательное поле
Кем выдан паспорт	E_GIVEN	V(50)	обязательное поле
ИНН	E_INN	C(12)	обязательное уникальное поле
Номер пенсионного страхового свидетельства	E_PENS	C(14)	обязательное уникальное поле
Отдел	E_DEPART	V(12)	внешний ключ (к Departs)
Должность	E_POST	V(30)	внешний ключ (к Posts)
Номер комнаты	E_ROOM	N(4)	составной внешний ключ (к Rooms)
Рабочий телефон	E_PHONE	V(20)	
Логин	E_LOGIN	V(30)	

Таблица 11. Схема отношения ОБРАЗОВАНИЕ (Edu)

<i>Содержание поля</i>	<i>Имя поля</i>	<i>Тип, длина</i>	<i>Примечания</i>
Идентификатор сотрудника	U_ID	N(4)	внешний ключ (к Employees)
Вид образования	U_TYPE	V(20)	обязательное поле
Специальность	U_SPEC	V(40)	
Номер диплома	U_DIPLOM	V(15)	
Год окончания учебного заведения	U_YEAR	N(4)	обязательное поле

Таблица 12. Схема отношения АДРЕСА-ТЕЛЕФОНЫ (AdrTel)

<i>Содержание поля</i>	<i>Имя поля</i>	<i>Тип, длина</i>	<i>Примечания</i>
Идентификатор сотрудника	A_ID	N(4)	внешний ключ (к Employees)
Адрес	A_ADDR	V(50)	
Телефон	A_PHONE	V(30)	

Таблицы ОБРАЗОВАНИЕ и АДРЕСА-ТЕЛЕФОНЫ не имеют потенциальных ключей, но мы не будем вводить суррогатные первичные ключи, т.к. на эти таблицы никто не ссылается.

Таблица 13. Схема отношения ЗАКАЗЧИКИ (Clients)

<i>Содержание поля</i>	<i>Имя поля</i>	<i>Тип, длина</i>	<i>Примечания</i>
Номер заказчика	C_ID	N(4)	<b>суррогатный первичный ключ</b>
Заказчик	C_COMPANY	V(40)	обязательное поле
Адрес заказчика	C_ADR	V(50)	обязательное поле
Контактное лицо	C_PERSON	V(50)	обязательное поле
Телефон	C_PHONE	V(30)	

Таблица 14. Схема отношения ПРОЕКТЫ (Projects)

<i>Содержание поля</i>	<i>Имя поля</i>	<i>Тип, длина</i>	<i>Примечания</i>
Номер проекта	P_ID	N(6)	обязательное уникальное поле
Название проекта	P_TITLE	V(100)	обязательное поле
Сокращённое название	P_ABBR	C(10)	<b>первичный ключ</b>
Отдел	P_DEPART	V(12)	внешний ключ (к Departs)
Заказчик	P_COMPANY	N(4)	внешний ключ (к Clients)
Руководитель	P_CHIEF	N(4)	внешний ключ (к Employees)
Дата начала проекта	P_BEGIN	D	обязательное поле
Дата окончания проекта	P_END	D	обязательное поле, больше даты начала проекта
Реальная дата окончания	P_FINISH	D	больше даты начала проекта
Стоимость проекта	P_COST	N(10)	обязательное поле, > 0

Таблица 15. Схема отношения ЭТАПЫ ПРОЕКТА (Stages)

<i>Содержание поля</i>	<i>Имя поля</i>	<i>Тип, длина</i>	<i>Примечания</i>	
Проект	S_PRO	C(10)	внешний ключ (к Projects)	<b>составной первичный ключ</b>
Номер этапа	S_NUM	N(2)		
Название этапа	S_TITLE	V(200)	обязательное поле	
Дата начала этапа	S_BEGIN	D	обязательное поле	
Дата окончания этапа	S_END	D	обязательное поле, больше даты начала этапа	

Реальная дата окончания	S_FINISH	D	больше даты начала этапа
Стоимость этапа	S_COST	N(10)	обязательное поле
Полученная сумма по этапу	S_SUM	N(10)	обязательное поле, значение по умолчанию – 0
Форма отчётности	S_FORM	V(100)	обязательное поле

Таблица 16. Схема отношения УЧАСТИЕ (Job)

Содержание поля	Имя поля	Тип, длина	Примечания	
Проект	J_PRO	C(10)	внешний ключ (к Projects)	<b>составной ПК</b>
Сотрудник	J_EMP	N(4)	внешний ключ (к Employees)	
Роль	J_ROLE	V(20)	обязательное поле	
Доплата	J_BONUS	N(2)		

Схема базы данных после нормализации приведена на рис. 8.

*Определение дополнительных ограничений целостности*

Перечислим ограничения целостности, которые не указаны в табл. 7–16.

1. Атрибут Вид образования может принимать одно из следующих значений: 'начальное', 'среднее', 'средне-специальное', 'высшее'.
2. Атрибут Роль может принимать одно из двух значений: 'исполнитель' или 'консультант'.
3. В поле Доплата хранится величина доплаты сотруднику за участие в проекте (в процентах к его окладу). Значение поля больше либо равно 0.
4. Нумерация в поле Номер этапа начинается с 1 и является непрерывной для каждого проекта.

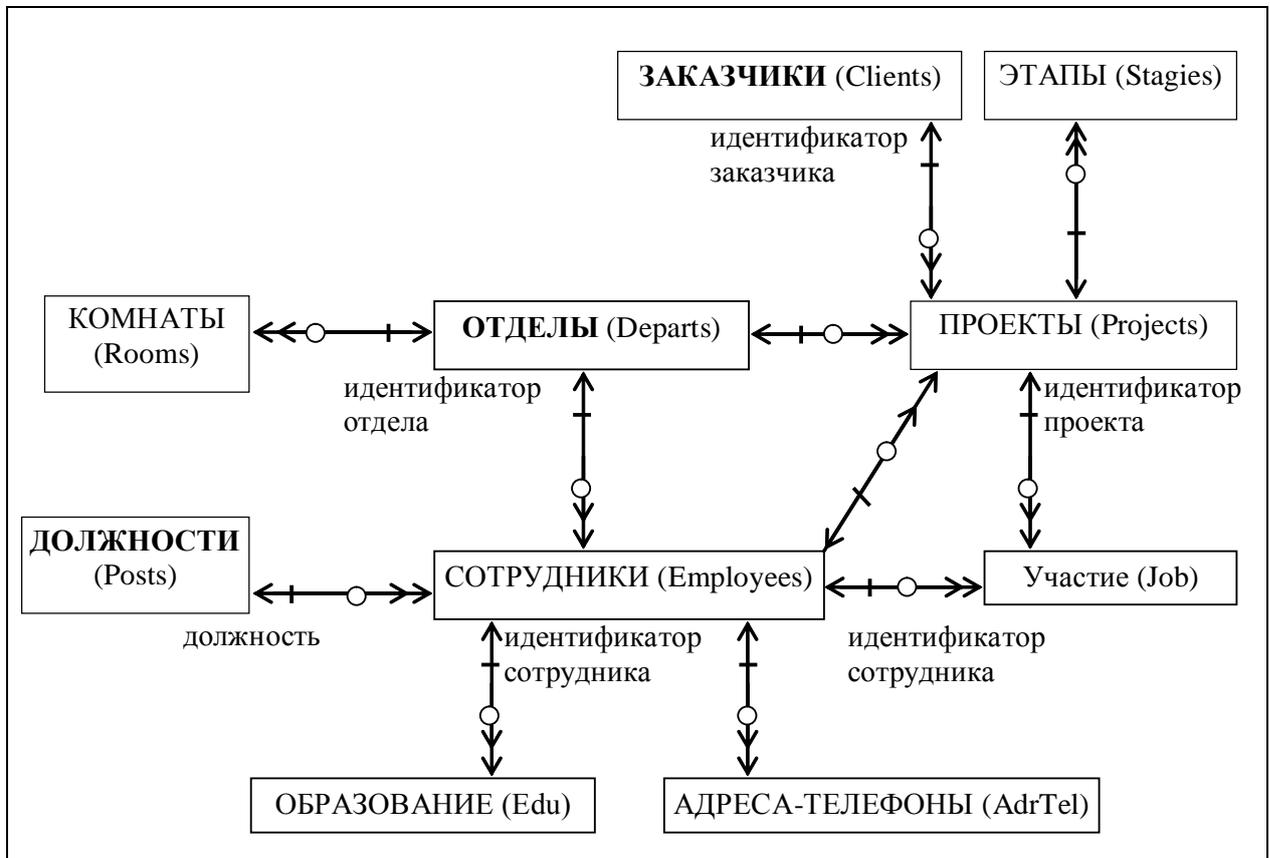


Рис. 8. Окончательная схема БД проектной организации

5. Дата начала первого этапа проекта должна соответствовать началу проекта в целом, дата завершения последнего этапа должна соответствовать завершению проекта в целом. Этапы не должны пересекаться по времени и между ними не должно быть разрывов.
  6. Стоимость проекта должна быть равна сумме стоимостей всех этапов этого проекта.
- Ограничения 4-6 нельзя реализовать в схеме отношения. В реальных БД подобные ограничения целостности реализуются вручную или программно (через внешнее приложение или специальную процедуру контроля данных – триггер).

ТЕМЫ КУРСОВЫХ РАБОТ

**Задание №1:** БД Банка

Таблицы:

1) Сотрудники (Код сотрудника, ФИО, Возраст, Пол, Адрес, Телефон, Паспортные данные, Код должности)[10 записей].

2) Должности (Код должности, Наименование должности, Оклад, Обязанности, Требования)[5 записей].

3) Вклады (Код вклада, Наименование вклада, Минимальный срок вклада, Минимальная сумма вклада, Код валюты, Процентная ставка, Дополнительные условия)[5 записей].

4) Валюта (Код валюты, Наименование, Обменный курс)[3 записи].

5) Вкладчики (ФИО вкладчика, Адрес, Телефон, Паспортные данные, Дата вклада, Дата возврата, Код вклада, Сумма вклада, Сумма возврата, Отметка о возврате вклада, Код сотрудника)[10 записей].

Запросы:

1) Отдел кадров (Связывает таблицы "Сотрудники" и "Должности" по полю "Код должности").

2) Вклады (Связывает таблицы "Вклады" и "Валюта" по полю "Код валюты").

3) Вкладчики (Связывает таблицы "Вкладчики", "Вклады" и "Сотрудники" по полям "Код вклада" и "Код сотрудника").

**Задание №2:** БД Больницы.

Таблицы:

1) Сотрудники (Код сотрудника, ФИО, Возраст, Пол, Адрес, Телефон, Паспортные данные, Код должности)[10 записей].

2) Должности (Код должности, Наименование должности, Оклад, Обязанности, Требования)[5 записей].

3) Лекарства (Код лекарства, Наименование, Показания, Противопоказания, Упаковка, Стоимость)[5 записей].

4) Болезни (Код болезни, Наименование, Симптомы, Продолжительность, Последствия, Код лекарства 1, Код лекарства 2, Код лекарства 3)[10 записей].

5) Пациенты (ФИО пациента, Возраст, Пол, Адрес, Телефон, Дата обращения, Код болезни, Код сотрудника, Результат лечения)[10 записей].

Запросы:

1) Отдел кадров (Связывает таблицы "Сотрудники" и "Должности" по полю "Код должности").

2) Болезни (Связывает таблицы "Болезни" и "Лекарства" по полю "Код лекарства", "Код лекарства 1", "Код лекарства 2" и "Код лекарства 3").

3) Пациенты (Связывает таблицы "Пациенты", "Болезни" и "Сотрудники" по полям "Код болезни" и "Код сотрудника").

**Задание №3:** БД Гостиницы.

Таблицы:

1) Сотрудники (Код сотрудника, ФИО, Возраст, Пол, Адрес, Телефон, Паспортные данные, Код должности)[10 записей].

2) Должности (Код должности, Наименование должности, Оклад, Обязанности, Требования)[5 записей].

3) Номера (Код номера, Наименование, Вместимость, Описание, Стоимость, Код сотрудника)[5 записей].

4) Услуги (Код услуги, Наименование, Описание, Стоимость)[5 записей].

5) Клиенты (ФИО, Паспортные данные, Дата заселения, Дата выезда, Код номера, Код услуги 1, Код услуги 2, Код услуги 3, Стоимость, Код сотрудника)

Запросы:

1) Отдел кадров (Связывает таблицы "Сотрудники" и "Должности" по полю "Код должности").

2) Номера (Связывает таблицы "Сотрудники" и "Номера" по полю "Код сотрудника").

3) Клиенты (Связывает таблицы "Клиенты", "Номера", "Услуги" и "Сотрудники" по полям "Код номера", "Код услуги", "Код услуги 1", "Код услуги 2", "Код услуги 3" и "Код сотрудника").

#### **Задание №4: БД МВД.**

Таблицы:

1) Сотрудники (Код сотрудника, ФИО, Возраст, Пол, Адрес, Телефон, Паспортные данные, Код должности, Код звания)[10 записей].

2) Должности (Код должности, Наименование должности, Оклад, Обязанности, Требования)[5 записей].

3) Звания (Код звания, Наименование, Надбавка, Обязанности, Требования)[5 записей].

4) Виды преступлений (Код вида преступления, Наименование, Статья, Наказание, Срок)[5 записей].

5) Преступники (Номер дела, ФИО, Дата рождения, Пол, Адрес, Код вида преступления, Код пострадавшего, Состояние, Код сотрудника)[10 записей].

6) Пострадавшие (Код пострадавшего, ФИО, Дата рождения, Пол, Адрес)[5 записей].

Запросы:

1) Отдел кадров (Связывает таблицы "Сотрудники", "Должности" и "Звания" по полям "Код должности" и "Код звания").

2) Преступники (Связывает таблицы "Преступники", "Виды преступлений", "Пострадавшие" и "Сотрудники" по полям "Код вида преступления", "Код пострадавшего" и "Код сотрудника").

#### **Задание №5: БД Аэропорта.**

Таблицы:

1) Сотрудники (Код сотрудника, ФИО, Возраст, Пол, Адрес, Телефон, Паспортные данные, Код должности)[10 записей].

2) Должности (Код должности, Наименование должности, Оклад, Обязанности, Требования)[5 записей].

3) Самолёты (Код самолёта, Марка, Вместимость, Грузоподъемность, Код типа, Технические характеристики, Дата выпуска, Налётано часов, Дата последнего ремонта, Код сотрудника)[5 записей].

4) Типы самолётов (Код типа, Наименование, Назначение, Ограничения).

5) Экипажи (Код экипажа, Налётано часов, Код сотрудника 1, Код сотрудника 2, Код сотрудника 3)[5 записей].

6) Рейсы (Код рейса, Дата, Время, Откуда, Куда, Код экипажа, Код самолёта, Время полёта)[5 записей].

7) Билеты (ФИО пассажира, Паспортные данные, Место, Код рейса, Цена)

Запросы:

1) Отдел кадров (Связывает таблицы "Сотрудники" и "Должности" по полю "Код должности").

2) Самолёты (Связывает таблицы “Самолёты”, “Типы самолётов” и “Сотрудники” по полям “Код типа” и “Код сотрудника”)

3) Экипажи (Связывает таблицы “Экипажи” и “Сотрудники” по полям “Код сотрудника” “Код сотрудника 1”, “Код сотрудника 2” и “Код сотрудника 3”)

4) Рейсы (Связывает таблицы “Рейсы”, “Самолёты” и “Экипажи” по полям “Код экипажа” и “Код самолёта”)

5) Билеты (Связывает таблицы “Билеты” и “Рейсы” по полю “Код рейса”)

**Задание №6:** БД Видео проката.

Таблицы:

1) Сотрудники (Код сотрудника, ФИО, Возраст, Пол, Адрес, Телефон, Паспортные данные, Код должности)[10 записей].

2) Должности (Код должности, Наименование должности, Оклад, Обязанности, Требования)[5 записей].

3) Жанры (Код жанра, Наименование жанра, Описание)[5 записей].

4) Кассеты (Код кассеты, Наименование фильма, Год создание, Производитель, Страна, Главный актёр, Дата записи, Код жанра, Цена)[10 записей].

5) Клиенты (ФИО, Адрес, Телефон, Паспортные данные, Дата взятия, Дата возврата, Отметка об оплате, Отметка о возврате, Код кассеты 1, Код кассеты

2, Код кассеты 3, Код сотрудника)[10 записей].

Запросы:

1) Отдел кадров (Связывает таблицы "Сотрудники" и "Должности" по полю "Код должности").

2) Кассеты (Связывает таблицы "Кассеты" и "Жанры" по полю "Код жанра").

3) Кассеты на руках (Связывает таблицы “Клиенты”, “Кассеты” и “Сотрудники” по полям “Код кассеты”, “Код кассеты 1”, “Код кассеты 2”, “Код кассеты 3” и “Код сотрудника”).

**Задание №7:** БД Библиотеки.

Таблицы:

1) Сотрудники (Код сотрудника, ФИО, Возраст, Пол, Адрес, Телефон, Паспортные данные, Код должности)[10 записей].

2) Должности (Код должности, Наименование должности, Оклад, Обязанности, Требования)[5 записей].

3) Издательства (Код издательства, Наименование, Город, Адрес)[5 записей].

4) Жанры (Код жанра, Наименование, Описание) [5 записей].

5) Книги (Код книги, Наименование, Автор, Код издательства, Год издания, Код жанра) [10 записей].

6) Читатели (Код читателя, ФИО, Дата рождения, Пол, Адрес, Телефон, Паспортные данные) [10 записей].

7) Выданные книги (Код книги, Код читателя, Дата выдачи, Дата возврата, Отметка о возврате, Код сотрудника) [10 записей].

Запросы:

1) Отдел кадров (Связывает таблицы "Сотрудники" и "Должности" по полю "Код должности").

2) Каталог (Связывает таблицы “Книги”, “Издательства” и “Жанры” по полям “Код издательства” и “Код жанра”).

3) Книги на руках (Связывает таблицы “Выданные книги”, “Книги”, “Читатели” и “Сотрудники” по полям “Код книги”, “Код читателя” и “Код сотрудника”)

**Задание №8:** БД Радиостанции.

Таблицы:

- 1) Сотрудники (Код сотрудника, ФИО, Возраст, Пол, Адрес, Телефон, Паспортные данные, Код должности)[10 записей].
- 2) Должности (Код должности, Наименование должности, Оклад, Обязанности, Требования)[5 записей].
- 3) Исполнители (Код исполнителя, Наименование, Описание)[5 записей].
- 4) Жанры (Код жанра, Наименование, Описание)[5 записей].
- 5) Записи (Код записи, Наименование, Код исполнителя, Альбом, Год, Код жанра, Дата записи, Длительность, Рейтинг)[10 записей].
- 6) График работы (Дата, Код сотрудника, Время 1, Код записи 1, Время 2, Код записи 2, Время 3, Код записи 3)[10 записей].

Запросы:

- 1) Отдел кадров (Связывает таблицы "Сотрудники" и "Должности" по полю "Код должности").
- 2) Музыкальный архив (Связывает таблицы "Записи", "Исполнители" и "Жанры" по полям "Код исполнителя" и "Код жанра").
- 3) Сетка вещания (Связывает таблицы "График работы", "Сотрудники" и "Записи" по полям "Код сотрудника", "Код записи", "Код записи 1", "Код записи 2" и "Код записи 3").

**Задание №9:** БД Таксопарка.

Таблицы:

- 1) Сотрудники (Код сотрудника, ФИО, Возраст, Пол, Адрес, Телефон, Паспортные данные, Код должности)[10 записей].
- 2) Должности (Код должности, Наименование должности, Оклад, Обязанности, Требования)[5 записей].
- 3) Марки (Код марки, Наименование, Технические характеристики, Стоимость, Специфика)[5 записей].
- 4) Тарифы (Код тарифа, Наименование, Описание, Стоимость)[5 записей].
- 5) Дополнительные услуги (Код услуги, Наименование, Описание услуги, Стоимость)[5 записей].
- 6) Автомобили (Код автомобиля, Код марки, Регистрационный номер, Номер кузова, Номер двигателя, Год выпуска, Пробег, Код сотрудника-шофёра, Дата последнего ТО, Код сотрудника-механика, Специальные отметки)[10 записей].
- 7) Вызовы (Дата, Время, Телефон, Откуда, Куда, Код тарифа, Код услуги, Код автомобиля, Код сотрудника-оператора)[10 записей].

Запросы:

- 1) Отдел кадров (Связывает таблицы "Сотрудники" и "Должности" по полю "Код должности").
- 2) Автопарк (Связывает таблицы "Автомобили", "Марки" и "Сотрудники" по полю "Код марки" и "Код сотрудника").
- 3) Список вызовов (Связывает таблицы "Вызовы", "Тарифы", "Услуги", "Автомобили" и "Сотрудники" по полю "Код тарифа", "Код услуги", "Код автомобиля" и "Код сотрудника-диспетчера").

**Задание №10:** БД Туристического агентства.

Таблицы:

- 1) Сотрудники (Код сотрудника, ФИО, Возраст, Пол, Адрес, Телефон, Паспортные данные, Код должности)[10 записей].
- 2) Должности (Код должности, Наименование должности, Оклад, Обязанности, Требования)[5 записей].

- 3) Виды отдыха (Код вида, Наименование, Описание, Ограничения)[5 записей].
- 4) Отели (Код отеля, Наименование, Страна, Город, Адрес, Телефон, Количество звёзд, Контактное лицо)[10 записей].
- 5) Дополнительные услуги (Код услуги, Наименование, Описание, Цена) [5 записей].
- 6) Клиенты (Код клиента, ФИО, Дата рождения, Пол, Адрес, Телефон, Паспортные данные)[5 записей].
- 7) Путёвки (Дата начала, Дата окончания, Продолжительность, Код отеля, Код вида, Код услуги 1, Код услуги 2, Код услуги 3, Код клиента, Код сотрудника, Отметка о бронировании, Отметка об оплате)[10 записей].

Запросы:

- 1) Отдел кадров (Связывает таблицы "Сотрудники" и "Должности" по полю "Код должности").
- 2) Список путёвок (Связывает таблицы "Путёвки", "Отели", "Виды отдыха", "Дополнительные услуги", "Клиенты" и "Сотрудники" по полям "Код отеля", "Код вида", "Код услуги", "Код услуги 1", "Код услуги 2", "Код услуги 3", "Код клиента" и "Код сотрудника").

**Задание №11:** БД Страховой компании.

Таблицы:

- 1) Сотрудники (Код сотрудника, ФИО, Возраст, Пол, Адрес, Телефон, Паспортные данные, Код должности)[10 записей].
- 2) Должности (Код должности, Наименование должности, Оклад, Обязанности, Требования)[5 записей].
- 3) Риски (Код риска, Наименование, Описание, Средняя вероятность)[5 записей].
- 4) Виды полисов (Код вида полиса, Наименование, Описание, Условия, Код риска 1, Код риска 2, Код риска 3)[5 записей].
- 5) Группы клиентов (Код группы, Наименование, Описание)[5 записей].
- 6) Клиенты (Код клиента, ФИО, Дата рождения, Пол, Адрес, Телефон, Паспортные данные, Код группы)[10 записей].
- 7) Полисы (Номер полиса, Дата начала, Дата окончания, Стоимость, Сумма выплаты, Код вида полиса, Отметка о выплате, Отметка об окончании, Код клиента, Код сотрудника)[10 записей].

Запросы:

- 1) Отдел кадров (Связывает таблицы "Сотрудники" и "Должности" по полю "Код должности").
- 2) Риски полисов (Связывает таблицы "Виды полисов" и "Риски" по полям "Код риска", "Код риска 1", "Код риска 2", "Код риска 3").
- 3) Список клиентов (Связывает таблицы "Клиенты" и "Группы клиентов" по полю "Код группы").
- 4) Список полисов (Связывает таблицы "Полисы", "Виды полисов", "Клиенты" и "Сотрудники" по полям "Код вида полиса", "Код клиента" и "Код сотрудника").

**Задание №12:** БД Сервис центра.

Таблицы:

- 1) Сотрудники (Код сотрудника, ФИО, Возраст, Пол, Адрес, Телефон, Паспортные данные, Код должности)[10 записей].
- 2) Должности (Код должности, Наименование должности, Оклад, Обязанности, Требования)[5 записей].
- 3) Запчасти (Код запчасти, Наименование, Функции, Цена)[5 записей].

4) Ремонтируемые модели (Код модели, Наименование, Тип, Производитель, Технические характеристики, Особенности)[5 записей].

5) Виды неисправностей (Код вида, Код модели, Описание, Симптомы, Методы ремонта, Код запчасти 1, Код запчасти 2, Код запчасти 3, Цена работы)[5 записей].

6) Обслуживаемые магазины (Код магазина, Наименование, Адрес, Телефон)[5 записей].

7) Заказы (Дата заказа, Дата возврата, ФИО заказчика, Серийный номер, Код вида неисправности, Код магазина, Отметка о гарантии, Срок гарантии ремонта, Цена, Код сотрудника)[10 записей].

Запросы:

1) Отдел кадров (Связывает таблицы "Сотрудники" и "Должности" по полю "Код должности").

2) Список неисправностей (Связывает таблицы "Виды неисправностей", "Ремонтируемые модели" и "Запчасти" по полям "Код модели", "Код запчасти", "Код запчасти 1", "Код запчасти 2", "Код запчасти 3").

3) Список заказов (Связывает таблицы "Заказы", "Виды неисправностей", "Обслуживаемые магазины" и "Сотрудники" по полям "Код вида неисправности", "Код магазина" и "Код сотрудника").

**Задание №13:** БД Школы.

Таблицы:

1) Сотрудники (Код сотрудника, ФИО, Возраст, Пол, Адрес, Телефон, Паспортные данные, Код должности)[10 записей].

2) Должности (Код должности, Наименование должности, Оклад, Обязанности, Требования)[5 записей].

3) Ученики (ФИО, Дата рождения, Пол, Адрес, ФИО отца, ФИО матери, Код класса, Дополнительная информация) [10 записей].

4) Классы (Код класса, Код сотрудника-классного руководителя, Код вида, Количество учеников, Буква, Год обучения, Год создания)[5 записей].

5) Виды классов (Код вида, Наименование, Описание)[5 записей].

6) Предметы (Код предмета, Наименование, Описание, Код сотрудника-учителя)[10 записей].

7) Расписание (Дата, День недели, Код класса, Код предмета, Время начала, Время окончания)[10 записей].

Запросы:

1) Отдел кадров (Связывает таблицы "Сотрудники" и "Должности" по полю "Код должности").

2) Список учеников (Связывает таблицы "Ученики" и "Классы" по полю "Код класса").

3) Список классов (Связывает таблицы "Классы", "Виды классов" и "Сотрудники" по полям "Код вида" и "Код сотрудника").

4) Список предметов (Связывает таблицы "Предметы" и "Сотрудники" по полю "Код сотрудника").

5) Расписание занятий (Связывает таблицы "Расписание", "Классы" и "Предметы" по полям "Код класса" и "Код предмета").

**Задание №14:** БД Транспортной компании.

Таблицы:

1) Сотрудники (Код сотрудника, ФИО, Возраст, Пол, Адрес, Телефон, Паспортные данные, Код должности)[10 записей].

- 2) Должности (Код должности, Наименование должности, Оклад, Обязанности, Требования)[5 записей].
- 3) Виды автомобилей (Код вида автомобиля, Наименование, Описание)[5 записей].
- 4) Марки автомобилей (Код марки, Наименование, Технические характеристики, Описание) [5 записей].
- 5) Виды грузов (Код вида груза, Наименование, Код вида автомобиля для транспортировки, Описание)[5 записей].
- 6) Грузы (Код груза, Наименование, Код вида груза, Срок годности, Особенности)[5 записей].
- 7) Автомобили (Код автомобиля, Код марки, Код вида автомобиля, Регистрационный номер, Номер кузова, номер двигателя, Год выпуска, Код сотрудника-водителя, Дата последнего ТО, Код сотрудника-механика)[5 записей].
- 8) Рейсы (Код автомобиля, Заказчик, Откуда, Куда, Дата отправления, Дата прибытия, Код груза, Цена, Отметка об оплате, Отметка о возвращении, Код сотрудника)[10 записей].

Запросы:

- 1) Отдел кадров (Связывает таблицы "Сотрудники" и "Должности" по полю "Код должности").
- 2) Транспортировка (Связывает таблицы "Виды грузов" и "Виды автомобилей" по полю "Код вида автомобиля").
- 3) Перевозимые грузы (Связывает таблицы "Грузы" и "Виды грузов" по полю "Код вида груза").
- 4) Автопарк (Связывает таблицы "Автомобили", "Марки автомобилей", "Виды автомобилей" и "Сотрудники" по полям "Код марки", "Код вида автомобиля" и "Код сотрудника").
- 5) Заказы (Связывает таблицы "Рейсы", "Автомобили", "Грузы" и "Сотрудники" по полям "Код автомобиля", "Код груза" и "Код сотрудника").

**Задание №15:** БД Проката автомобилей.

Таблицы:

- 1) Сотрудники (Код сотрудника, ФИО, Возраст, Пол, Адрес, Телефон, Паспортные данные, Код должности)[10 записей].
- 2) Должности (Код должности, Наименование должности, Оклад, Обязанности, Требования)[5 записей].
- 3) Марки автомобилей (Код марки, Наименование, Технические характеристики, Описание) [5 записей].
- 4) Дополнительные услуги (Код услуги, Наименование, Описание, Цена)[5 записей].
- 5) Автомобили (Код автомобиля, Код марки, Регистрационный номер, Номер кузова, Номер двигателя, Год выпуска, Пробег, Цена автомобиля, Цена дня проката, Дата последнего ТО, Код сотрудника-механика, Специальные отметки, Отметка о возврате)[10 записей].
- 6) Клиенты (Код клиента, ФИО, Пол, Дата рождения, Адрес, Телефон, Паспортные данные) [5 записей].
- 7) Прокат (Дата выдачи, Срок проката, Дата возврата, Код автомобиля, Код клиента, Код услуги 1, Код услуги 2, Код услуги 3, Цена проката, Отметка об оплате, Код сотрудника)[10 записей].

Запросы:

- 1) Отдел кадров (Связывает таблицы "Сотрудники" и "Должности" по полю "Код должности").

2) Автопарк (Связывает таблицы "Автомобили", "Марки автомобилей" и "Сотрудники" по полям "Код марки" и "Код сотрудника").

3) Автомобили в прокате (Связывает таблицы "Прокат", "Автомобили", "Клиенты", "Дополнительные услуги" и "Сотрудники" по полям "Код автомобиля", "Код клиента", "Код услуги", "Код услуги 1", "Код услуги 2", "Код услуги 3" и "Код сотрудника")