**Правительство Российской Федерации**

**Федеральное государственное образовательное учреждение**

**Высшего образования**

**«Национальный исследовательский университет**

**«Высшая школа экономики»**

Московский институт электроники и математики

Департамент компьютерной инженерии

**Домашняя работа по курсу**

Базы данных

На тему: «Проектирование реляционных баз данных отдела лабораторной диагностики»

Выполнил:

Никитин О.А. – БИВ151

Арцыбашева В. А – БИВ155

Проверил:

Карпова И.П.

«\_\_\_» 2017 г.

Оглавление

[1. Инфологическое проектирование 3](#_Toc495659827)

[1.1. Анализ предметной области 3](#_Toc495659828)

[1.2. Анализ информационных задач и круга пользователей системы 4](#_Toc495659829)

# 1. Инфологическое проектирование

## 1.1. Анализ предметной области

База данных создаётся для информационного обслуживания отдела лабораторной диагностики. БД должна содержать данные об исследованиях, сотрудниках и пациентах.

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

* Каждый клиент может сдать несколько анализов, каждому образцу анализа соответствует один клиент.
* Каждый образец материала берется в определенном филиале, в каждом филиале может собираться несколько образцов материала.
* Каждый образец берется определенным сотрудником, каждый сотрудник может собрать несколько образцов.
* Каждый сотрудник может либо только собирать образцы материала, либо только проводить тесты, либо и то, и другое, либо не делать ничего.
* Каждый образец может тестироваться несколько раз, результат теста записывается. Каждому результату теста соответствует ровно один образец.
* Каждый тест проводится определенным сотрудником, каждый сотрудник может проводить несколько тестов.
* Каждый тест проводится в определенном филиале, в каждом филиале может проводиться несколько тестов.
* В каждом филиале могут либо только собираться образцы материала, либо только проводиться тесты, либо и то, и другое, либо не делаться ничего.
* Собираться и анализироваться образцы материала могут собираться в одном и том же либо в разных филиалах.
* У каждого теста имеется его унифицированный тип, к каждому типу теста может принадлежать несколько конкретных тестов анализа образца.
* У каждого унифицированного типа теста имеется список необходимых для его проведения материалов, каждый материал может быть необходим нескольким типам тестов.
* В каждом филиале могут храниться несколько видов материалов (или оборудования), один и тот же вид материал может храниться в нескольких различных филиалах.

Для создания ER-модели необходимо выделить сущности предметной области:

1. **Клиенты**. Атрибуты: ФИО, номер страхового свидетельства, адрес регистрации, дата рождения, пол, телефон, email.
2. **Образцы**. Атрибуты: тип образца, место сбора (например, кровь из пальца или кровь из вены), способ сбора, дата сбора.
3. **Сотрудники**. Атрибуты: ФИО, паспортные данные, дата рождения, пол, ИНН (индивидуальный номер налогоплательщика), номер пенсионного страхового свидетельства, название департамента, должность, оклад, рабочая электронная почта, рабочий номер телефона.
4. **Филиал** (отдельное здание или комплекс зданий). Атрибуты: название, адрес (регион, город, улица, номер дома, почтовый индекс).
5. **Эксперимент (тест)**. Атрибуты: дата проведения, результат.
6. **Унифицированный тип теста**. Атрибуты: название, стоимость.
7. **Сопутствующие материалы**. Атрибуты: название.

Также были выделены атрибуты у некоторых связей:

1. Связь **Требуются материалы** (между сущностями **Тип теста** и **Материалы**). Атрибуты: количество и единицы измерения.
2. Связь **Хранятся** (между сущностями **Филиал** и **Материалы**). Атрибуты: количество и единицы измерения.

Исходя из выявленных сущностей, построим ER–диаграмму (Рисунок 1).

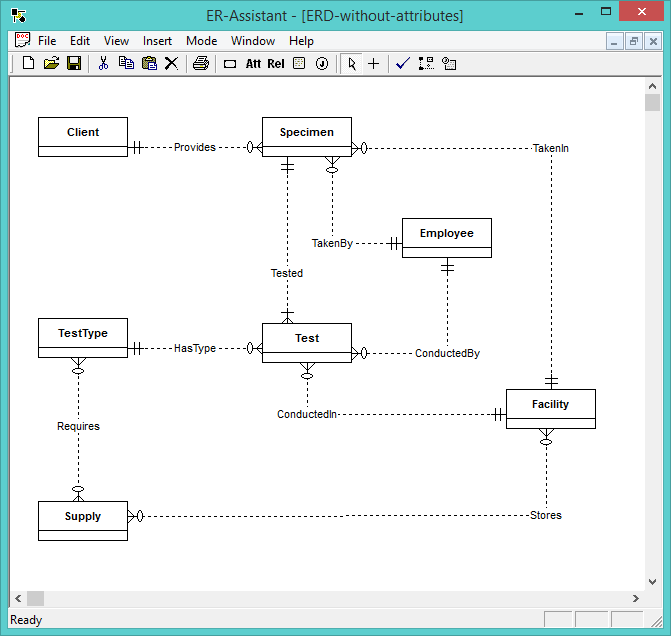


Рисунок 1. ER-диаграмма ПрО

## 1.2. Анализ информационных задач и круга пользователей системы

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

1. Руководители отдела лабораторной диагностики:

* получение списка сотрудников различных департаментов;
* изменение должностных окладов;
* получение полной информации об экспериментах.

1. Руководитель департамента:

* назначение сотрудников на проведения исследований (тестирований);
* получение списка сотрудников, работающих над конкретным типом тестов в данном департаменте;
* получение сведений о сотрудниках, которые собирают анализы.

1. Руководители по развитию и продвижению:

* планирование новых филиалов (например, в других городах);
* найм и увольнение сотрудников;
* планирование закупок нового оборудования.

1. Бухгалтеры:

* получение ведомости на выплату зарплаты.

1. Сотрудники, собирающие анализы:

* просмотр данных о клиентах;
* ввод данных о полученных образцах;

1. Сотрудники – врачи, проводящие исследования:

* просмотр данных о клиентах;
* просмотр данных об образцах;
* просмотр данных о типах исследований;
* добавление результатов тестирования;
* назначение дополнительных тестирований.

1. Сотрудники, отвечающие за снабжение:

* просмотр данных о типах исследований;
* изменение данных о сопутствующих материалах.

1. Клиенты:

* получение выписки о результатах тестирования.

# 2.1. Логическое проектирование реляционной БД

### 2.1.1. Преобразование ER–диаграммы в схему базы данных

База данных создаётся на основании схемы базы данных. 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 дочернего отношения. Внешнему ключу должен соответствовать первичный или уникальный ключ основного (родительского) отношения.

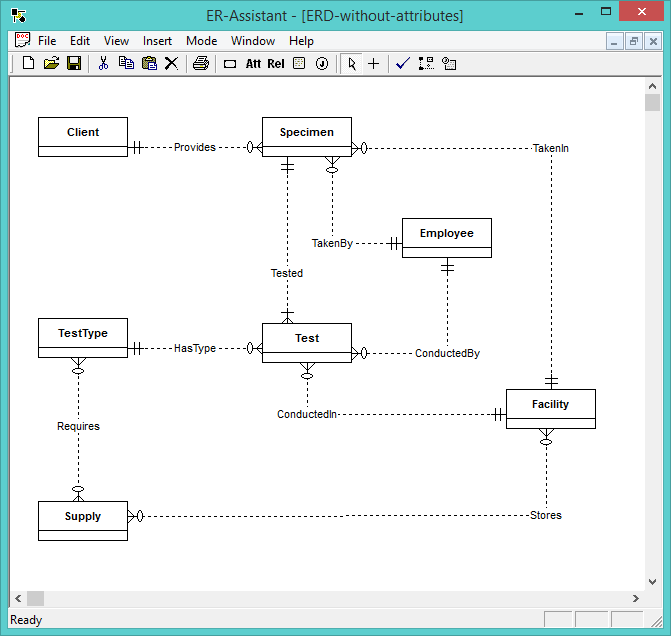


Рис. . ER-диаграмма ПрО

Связь **Требуются материалы** (между сущностями **Тип теста** и **Материалы**) принадлежит к типу n:m (Рис. 2). Этот тип связи реализуется через вспомогательное отношение **Требование материала (Requirement)**, которое содержит комбинации первичных ключей соответствующих исходных отношений и атрибуты связи.

Для схемы БД будем использовать нотацию *вороньей лапки* (также известной как Crow’s foot notation). Подробнее о ней можно прочитать в оригинальном источнике (Everest, 1976).

Полученная схема реляционной базы данных (РБД) приведена на Рис. 3.

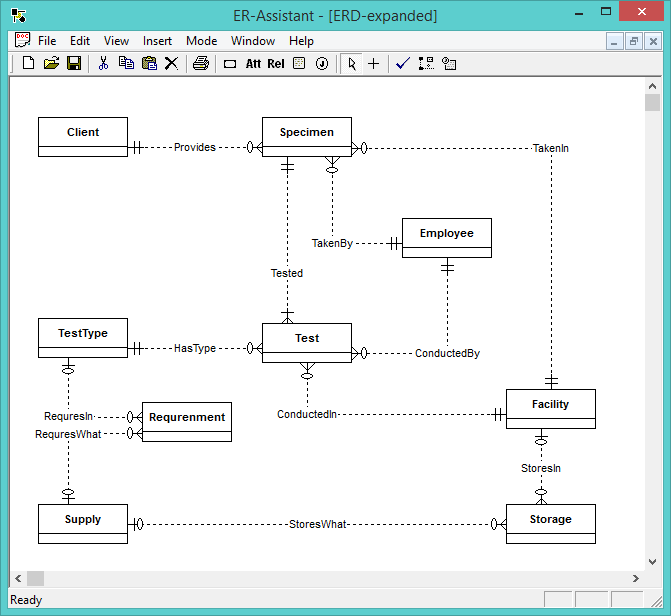


Рис. . Схема РБД, полученная из ER–диаграммы проектной организации

### 2.1.2. Составление реляционных отношений

Отношения приведены в табл. 1-5. Для каждого отношения указаны атрибуты с их внутренним названием, типом и длиной. Типы данных обозначаются так: N – числовой, C – символьный тип фиксированной длины, V – символьный тип переменной длины, D – дата (этот тип имеет стандартную длину, зависящую от СУБД, поэтому она не указывается).

Сущность ***Клиенты*** не имеет простого потенциального первичного ключа, поэтому введем суррогатный первичный ключ ***Номер клиента***.

Таблица 1. Схема отношения **Клиенты** (Clien)

|  |  |  |  |
| --- | --- | --- | --- |
| *Содержание поля* | *Имя поля* | *Тип, длина* | *Примечания* |
| Номер | ClientNo | N(8) | **суррогатный первичный ключ** |
| ФИО | ClientName | V(150) | обязательное поле |
| Страховое свидетельство | ClientInsurance | V(15) | обязательное поле |
| Адрес регистрации | ClientAddress | V(100) | необязательное многозначное поле |
| Дата рождения | ClientDOB | D | обязательное поле |
| Пол | ClientSex | C(1) | обязательное поле, ‘M’ или ‘F’ |
| Телефон | ClientPhone | V(50) | необязательное многозначное поле |
| E-mail | ClientEmail | V(100) | необязательное многозначное поле |

Сущность ***Образцы*** не имеет простого потенциального первичного ключа, поэтому введем суррогатный первичный ключ ***Номер образца***.

Таблица 2. Схема отношения ***Образцы*** (Specimen)

|  |  |  |  |
| --- | --- | --- | --- |
| *Содержание поля* | *Имя поля* | *Тип, длина* | *Примечания* |
| Номер | SpecNo | N(8) | **суррогатный первичный ключ** |
| Зона сбора | SpecArea | V(20) | необязательное поле |
| Способ сбора | SpecCollMethod | V(30) | обязательное поле |
| Дата | SpecDate | D | обязательное поле |
| Тип образца | SpecType | V(20) | обязательное поле |

В отношении ***Сотрудники*** три потенциальных ключа: *Номер проекта*, *Название проекта* и *Сокращённое названиие*. 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 3. 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 | С(10) | **первичный ключ** |
| 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 4. Схема отношения *ЭТАПЫ ПРОЕКТА* (Stages)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *Содержание поля* | *Имя поля* | *Тип, длина* | *Примечания* | |
| Проект | S\_PRO | C(10) | внешний ключ (к Projects) | **составной первичный ключ** |
| 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 5. Схема отношения *УЧАСТИЕ* (Job)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *Содержание поля* | *Имя поля* | *Тип, длина* | *Примечания \** | |
| Проект | J\_PRO | C(10) | внешний ключ (к Projects) | **составной первичный ключ** |
| Сотрудник | J\_EMP | N(4) | внешний ключ (к Employees) |
| Роль | J\_ROLE | V(20) | 'консультант' или 'исполнитель' |
| Доплата | J\_BONUS | N(2) |  | |

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

### 2.1.3. 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 (мы учитываем 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 вычислимый 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 необходимо предусмотреть в приложении проверку того, что сумма стоимостей по всем этапам совпадает с общей 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.

Отношения, полученные после нормализации, приведены в табл. 6-15.

Таблица 6. Схема отношения *ОТДЕЛЫ* (Departs)

|  |  |  |  |
| --- | --- | --- | --- |
| *Содержание поля* | *Имя поля* | *Тип, длина* | *Примечания* |
| Аббревиатура отдела | D\_ID | V(12) | **первичный ключ** |
| Название отдела | D\_NAME | V(100) | обязательное поле |

Таблица 7. 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) | **составной уникальный ключ** |
| Телефон | R\_PHONE | V(20) |

Таблица 8. Схема отношения *ДОЛЖНОСТИ* (Posts)

|  |  |  |  |
| --- | --- | --- | --- |
| *Содержание поля* | *Имя поля* | *Тип, длина* | *Примечания* |
| Название должности | P\_POST | V(30) | **первичный ключ** |
| Оклад | P\_SAL | N(8,2) | обязательное поле, > 4500 руб. |

Таблица 9. Схема отношения *СОТРУДНИКИ* (Employees)

|  |  |  |  |
| --- | --- | --- | --- |
| *Содержание поля* | *Имя поля* | *Тип, длина* | *Примечания* |
| Идентификатор сотрудника | E\_ID | N(4) | **суррогатный первичный ключ** |
| Фамилия | E\_FNAME | V(25) | обязательное поле |
| Имя, отчество | E\_LNAME | V(30) | обязательное поле |
| Дата рождения | E\_BORN | D | обязательное поле |
| Пол | E\_GENDER | 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 10. Схема отношения *ОБРАЗОВАНИЕ* (Edu)

|  |  |  |  |
| --- | --- | --- | --- |
| *Содержание поля* | *Имя поля* | *Тип, длина* | *Примечания* |
| 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 11. Схема отношения *АДРЕСА-ТЕЛЕФОНЫ* (AdrTel)

|  |  |  |  |
| --- | --- | --- | --- |
| *Содержание поля* | *Имя поля* | *Тип, длина* | *Примечания* |
| 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 12. Схема отношения *ЗАКАЗЧИКИ* (Clients)

|  |  |  |  |
| --- | --- | --- | --- |
| *Содержание поля* | *Имя поля* | *Тип, длина* | *Примечания* |
| Номер заказчика | C\_ID | N(4) | **суррогатный первичный ключ** |
| 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 13. 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 | С(10) | **первичный ключ** |
| 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 14. Схема отношения *ЭТАПЫ ПРОЕКТА* (Stages)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *Содержание поля* | *Имя поля* | *Тип, длина* | *Примечания* | |
| Проект | S\_PRO | C(10) | внешний ключ (к Projects) | **составной первичный ключ** |
| 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 15. Схема отношения *УЧАСТИЕ* (Job)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *Содержание поля* | *Имя поля* | *Тип, длина* | *Примечания* | |
| Проект | J\_PRO | C(10) | внешний ключ (к Projects) | **состав-ной ПК** |
| 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. 7.



Рис. 7. Окончательная схема БД проектной организации

### 2.1.4. Определение дополнительных ограничений целостности

Перечислим ограничения целостности, которые не указаны в табл. 6–15.

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.
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).

**Примечание.** Вопросы архивирования данных в этом пособии подробно не рассматриваются. Но следует отметить, что обычно архив является частью БД и представляет собой набор отдельных таблиц, которые не связаны с оперативной частью БД внешними ключами. Структура архивных таблиц либо соответствует структуре тех оперативных таблиц, данные которых подлежат архивированию, либо представляет собой денормализованную таблицу, соответствующую декартову произведению оперативных таблиц. Данные в архивные таблицы переносятся специальной программой (или набором запросов) автоматически или по команде пользователя. По истечении периода хранения данные могут удаляться из архива.

### 2.1.5. Описание групп пользователей и прав доступа

Опишем для каждой группы пользователей права доступа к каждой таблице. Права доступа должны быть распределены так, чтобы для каждого объекта БД был хотя бы один пользователь, который имеет право добавлять и удалять данные из объекта. Права приведены в табл. 16. Используются следующие сокращения:

s – чтение данных (select);

i – добавление данных (insert);

u – модификация данных (update);

d – удаление данных(delete).

Таблица 16. Права доступа к таблицам для групп пользователей

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Таблицы | Группы пользователей (роли) | | | | |
| Руководители организации | Сотрудники отд. кадров | Руководители проектов | Бухгалтеры | Участники проектов |
| Отделы | S | SIUD | S | S |  |
| Комнаты | S | SUID | S | S | S |
| Должности | SIUD |  |  | S |  |
| Сотрудники | S | SUID | S | S |  |
| Адреса-телефоны | S | SUID | S | S |  |
| Образование | S | SUID | S | S |  |
| Заказчики | SIUD |  | S |  |  |
| Проекты | SIUD |  | S |  |  |
| Этапы проектов | SIUD |  | SUI |  |  |
| Участие | S |  | S | S |  |

Права на изменение данных в таблице *УЧАСТИЕ* будут назначены через представление, т.к. изменять данные этой таблицы может только руководитель проекта. Описание представлений приведено в п.2.5.2. "Создание представлений (готовых запросов)". Права назначает администратор БД (или администратор безопасности, если система сложная и администраторов несколько).