

Министерство образования и науки Российской Федерации
НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

В.М. СТАСЬШИН

ПРОЕКТИРОВАНИЕ
ИНФОРМАЦИОННЫХ СИСТЕМ
И БАЗ ДАННЫХ

Утверждено Редакционно-издательским советом университета
в качестве учебного пособия

НОВОСИБИРСК
2012

УДК 004.65(075.8)
С 779

Рецензенты: *Персова М.Г.*, д-р техн. наук, профессор
Кобылянский В.Г., канд. техн. наук, доцент

Работа подготовлена на кафедре программных систем и баз данных
для студентов IV курса ФПМИ дневного отделения
(направления 01040062, 01050062)

Стасьшин В.М.

С 779 Проектирование информационных систем и баз данных :
учеб. пособие / В.М. Стасьшин. – Новосибирск: Изд-во НГТУ,
2012. – 100 с.

ISBN 978-5-7782-2121-5

Рассмотрены вопросы проектирования реляционных баз данных, входящие в программу лекций курсов «Базы данных и экспертные системы», «Базы данных и СУБД», читаемых студентам IV курса факультета прикладной математики и информатики Новосибирского государственного технического университета. В учебном пособии представлены современные методологии проектирования информационных систем и основные этапы проектирования баз данных, инфологическое и логическое проектирование, разработка триггеров и хранимых процедур. Главы иллюстрированы примерами для более успешного усвоения материала.

Пособие может быть полезно специалистам, связанным с информационными технологиями, и тем, кто самостоятельно осваивает вопросы проектирования и разработки приложений над базами данных.

УДК 004.65(075.8)

ISBN 978-5-7782-2121-5

© Стасьшин В.М., 2012
© Новосибирский государственный
технический университет, 2012

ОГЛАВЛЕНИЕ

Введение	5
1. Проектирование информационных систем	7
1.1. Методология функционального моделирования работ (SADT)	7
1.2. Методология диаграмм потоков данных (DFD)	11
1.3. Методология объектного проектирования UML	14
1.3.1. Диаграммы прецедентов	14
1.3.2. Диаграммы деятельности	16
2. Проектирование баз данных	20
2.1. Этапы проектирования базы данных	20
2.2. Инфологическое проектирование	20
2.3. Логическое проектирование реляционных баз данных	27
2.3.1. Полная функциональная зависимость и вторая нормальная форма	29
2.3.2. Транзитивные зависимости и третья нормальная форма ...	32
2.3.3. Нормальная форма Бойса–Кодла	33
2.3.4. Многозначная зависимость и четвертая нормальная форма	35
2.3.5. Зависимость проекции/соединения и пятая нормальная форма	38
3. Ограничения целостности	39
3.1. Декларативная поддержка ограничений целостности	41
3.2. Процедурная поддержка ограничений целостности	42
4. Задания для выполнения лабораторных, курсовых и исследователь- ских работ по теме «Проектирование информационных систем и баз данных»	53
4.1. Информационная система вуза	55
4.2. Информационная система торговой компании	57
4.3. Информационная система медицинских организаций города	59

4.4. Информационная система автопредприятия	61
4.5. Информационная система проектной организации	62
4.6. Информационная система авиастроительного предприятия	64
4.7. Информационная система военного округа	66
4.8. Информационная система строительной организации	67
4.9. Информационная система библиотечного фонда города.....	69
4.10. Информационная система спортивных организаций города	71
4.11. Информационная система автомобилестроительного предприятия	72
4.12. Информационная система гостиничного комплекса	74
4.13. Информационная система магазина автозапчастей.....	76
4.14. Информационная система представительства туристической фирмы в зарубежной стране.	78
4.15. Информационная система аптеки	80
4.16. Информационная система библиотеки вуза.....	82
4.17. Информационная система туристического клуба.....	84
4.18. Информационная система городской телефонной сети.....	86
4.19. Информационная система театра	88
4.20. Информационная система зоопарка.....	90
4.21. Информационная система ГИБДД.....	92
4.22. Информационная система фотоцентра	93
4.23. Информационная система железнодорожной пассажирской станции	95
4.24. Информационная система городской филармонии	97
Библиографический список	99

ВВЕДЕНИЕ

В учебном пособии рассматриваются вопросы проектирования информационных систем и баз данных. Это очень важная тема, без основательного знакомства с которой невозможно стать не только квалифицированным программистом и разработчиком приложений над базами данных, но и разработчиком любых информационных систем.

Уровень развития информационных технологий заставляет задуматься большинство средних и крупных организаций о создании открытых и распределенных информационных систем баз данных на основе многопользовательских профессиональных СУБД. Информационные системы больших организаций содержат десятки баз данных, нередко распределенных между несколькими взаимосвязанными узлами вычислительной сети различных подразделений. Тенденции развития информационных технологий приводят к постоянному возрастанию сложности информационных систем, создаваемых в различных областях экономики.

Современные крупные информационные системы обладают, как правило, следующими особенностями:

- сложностью описания (т. е. взаимосвязями между большим количеством функций, процессов и элементов данных, требующей тщательного моделирования и анализа);
- наличием совокупности тесно взаимодействующих компонентов (подсистем), имеющих свои локальные задачи и использующих нерегламентированные запросы к данным большого объема;
- отсутствием прямых аналогов (т. е. невозможностью использования типовых проектных решений и прикладных систем);
- функционированием в неоднородной среде на нескольких аппаратных платформах;

- разобщенностью групп разработчиков и их разнородностью по уровню квалификации и по традиции использования инструментальных средств.

Для успешного создания базы данных объект проектирования должен быть прежде всего адекватно описан, должны быть построены полные и непротиворечивые функциональные и информационные модели. Опыт проектирования информационных систем показывает, что это логически сложная, затратная по времени работа, требующая высокой квалификации разработчиков.

При проектировании информационной системы необходимо проанализировать ее цель, выявить требования отдельных пользователей и построить серию адекватных моделей ее описания. Проектирование начинается с изучения процессов, потоков данных, построения диаграмм в соответствии с используемой методологией.

Проектирование базы данных, лежащей в основе большинства информационных систем, состоит в выявлении сущностей и связей между ними и построении ER-модели. Основная цель проектирования – это сокращение избыточности хранимых данных, что ведет не только к экономии объема используемой памяти, но и к уменьшению затрат на многократные операции обновления избыточных копий и к устранению возможных противоречий из-за хранения в разных местах сведений об одном и том же объекте.

В первых трех частях учебного пособия рассмотрены различные аспекты проектирования информационных систем и баз данных, даны рекомендации по разработке триггеров и хранимых процедур (являющихся современными средствами задания ограничений целостности). Приведенные в четвертой части пособия задания рекомендуются для выполнения курсовых, лабораторных и исследовательских работ по дисциплинам «Базы данных» и «Распределенные базы данных».

1. ПРОЕКТИРОВАНИЕ ИНФОРМАЦИОННЫХ СИСТЕМ

На начальных этапах создания информационной системы необходимо понять, как работает организация, которую собираются автоматизировать. Никто в компании не знает, как она работает в тех подробностях, которые необходимы для создания информационной системы. Руководитель хорошо знает работу в целом, но не в состоянии вникнуть в детали работы каждого сотрудника. Сотрудник хорошо знает, что происходит на его рабочем месте, но плохо понимает, как работают коллеги. Поэтому для описания работы предприятия (или его части) следует построить модель (лучше – серию моделей), которая была бы адекватна предметной области и содержала сведения обо всех участниках бизнес-процессов [3;, 5–7].

Наиболее популярными методологиями проектирования являются:

- методология функционального моделирования работ SADT (**Structured Analysis and Design Technique**);
- методология диаграмм потоков данных DFD (**Data Flow Diagrams**);
- методология объектного проектирования UML (**UniFied Modeling Language**) с целой серией моделей.

1.1. Методология функционального моделирования работ (SADT)

В методологии функционального моделирования работ Structured Analysis and Design Technagie (SADT) система представляется как совокупность взаимодействующих *работ (или функций)*. Связи между

работами определяют технологический процесс, или структуру взаимосвязи внутри организации. Содержательно модель SADT – серия диаграмм, разбивающих сложный объект на составные части.

Работа (activity) – поименованный процесс, функция или задача (изображается в виде прямоугольника), которые происходят в течение определенного времени и имеют распознаваемый результат. Каждая из работ на диаграммах следующего уровня может быть декомпозирована на несколько блоков, соединенных интерфейсными дугами. Эти блоки называют *подфункциями (подмодулями)* исходной функции. Каждый из подмодулей может быть декомпозирован аналогичным образом (рис. 1.1).

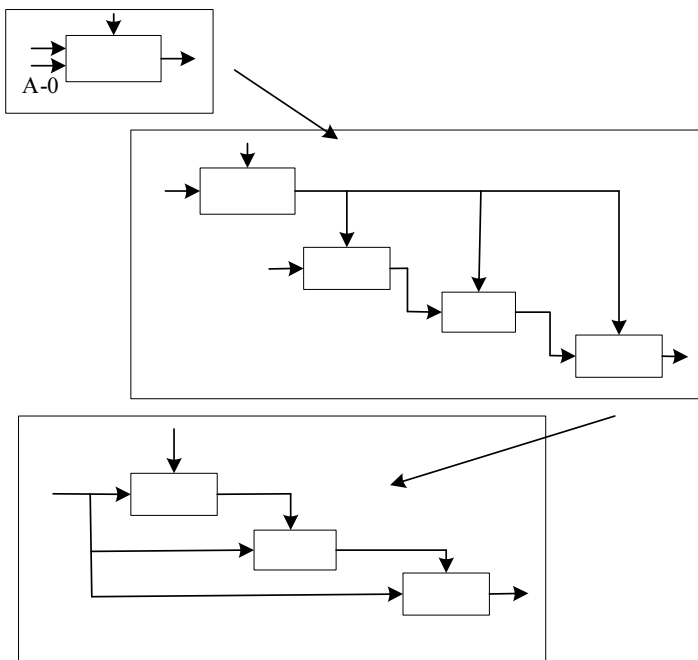


Рис. 1.1. Декомпозиция модели SADT

Взаимодействие работ с внешним миром и между собой описывается в виде стрелок, имеющих следующий смысл.

Вход (Input) – материал или информация, которые используются работой для получения результата (стрелка, входящая в левую грань).

Управление (Control) – правила, стратегии, стандарты, которыми руководствуется работа (стрелка, входящая в верхнюю грань). В отличие от входной информации управление не подлежит изменению.

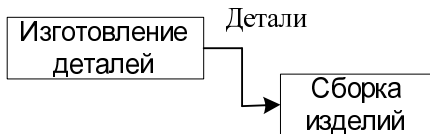
Выход (Output) – материал или информация, которые производятся работой (стрелка, исходящая из правой грани). Каждая работа должна иметь хотя бы одну стрелку выхода, поскольку работа без результата не имеет смысла и не должна моделироваться.

Механизм (Mechanism) – ресурсы, которые выполняют работу (персонал, станки, устройства – стрелка, входящая в нижнюю грань).

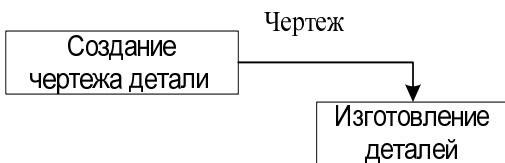
Вызов (Call) – стрелка, указывающая на другую модель работы (стрелка, исходящая из нижней грани).

В SADT-модели различают пять типов связей работ.

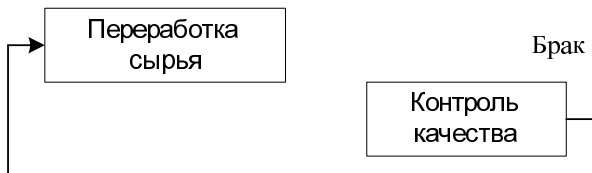
Связь по входу (input-output) возникает, когда выход вышестоящей работы направляется на вход следующей работы.



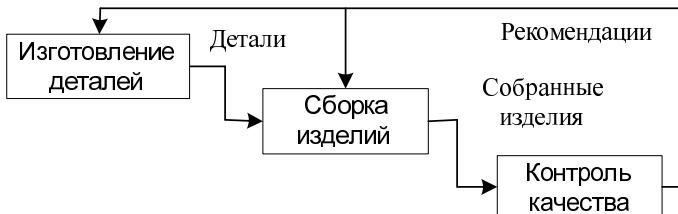
Связь по управлению (output-control) определяет, что выход вышестоящей работы направляется на управление следующей работы. Связь показывает доминирование вышестоящей работы.



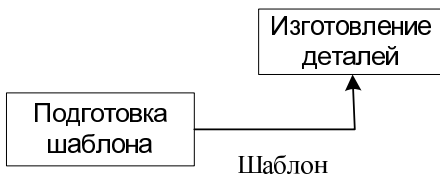
Обратная связь по входу (output-input feedback) возникает, когда выход нижестоящей работы направляется на вход вышестоящей. Используется для описания циклов.



Обратная связь по управлению (output-control feedback) определяет, что выход нижестоящей работы направляется на управление вышестоящей. Показатель эффективности бизнес-процесса.



Связь выход-механизм (output-mechanism) означает, что одна работа подготавливает ресурсы для проведения другой работы.



Из перечисленных блоков, как из отдельных кирпичиков, строится SADT-диаграмма (рис. 1.2).

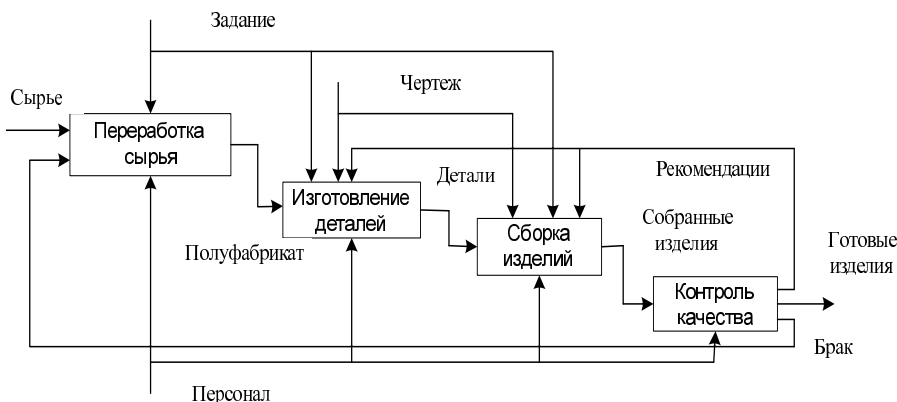


Рис. 1.2. SADT-диаграмма технологического процесса

1.2. Методология диаграмм потоков данных (DFD)

Диаграммы потоков данных (Data Flow Diagrams – DFD) используются для описания движения документов и обработки информации как дополнение к модели SADT. Стрелки в DFD-модели показывают, как объекты и данные движутся от одной работы к другой.

DFD содержит:

- *процессы*, которые преобразуют данные;
- *потоки данных*, которые данные переносят;
- *активные объекты*, которые производят и потребляют данные;
- *хранилища данных*, которые пассивно хранят данные;
- *потоки управления*, которые показывают пути вычисления значений.

Процесс преобразует значения данных. Процессы самого нижнего уровня – это элементарные функции (например, вычисление комиссионного сбора за выполнение проводки с помощью банковской карточки). Весь граф потока данных представляет собой процесс более высокого уровня. На DFD-модели процесс изображается в виде эллипса, внутри которого помещается имя процесса (рис. 1.3).

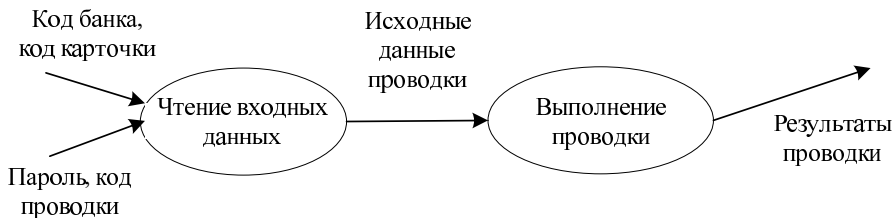


Рис. 1.3. Процессы в диаграмме потока данных

Поток данных соединяет выход объекта (или процесса) со входом другого объекта (или процесса) и представляет промежуточные данные вычислений. Поток данных изображается в виде стрелки между производителем и потребителем данных, помеченной именами соответствующих данных. Дуги могут разветвляться или сливаться, что означает соответственно разделение потока данных на части либо слияние объектов.

Активным объектом называется такой объект, который обеспечивает движение данных, поставляя или потребляя их. Активные объекты обычно бывают присоединены к входам и выходам DFD (рис. 1.4).

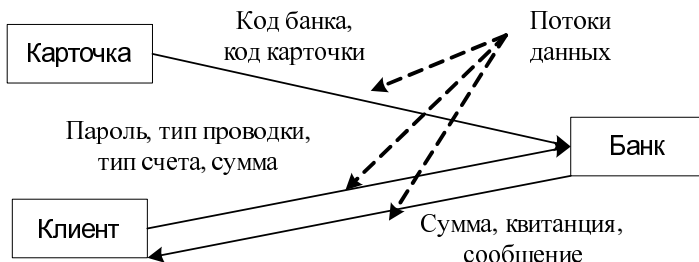


Рис. 1.4. Активные объекты и потоки в диаграмме потока данных

Хранилище данных – это пассивный объект в котором данные сохраняются для последующего доступа (рис. 1.5).

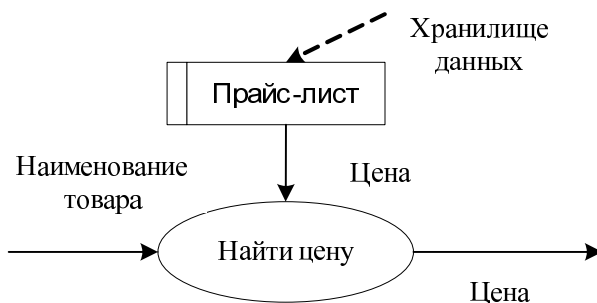


Рис. 1.5. Активные объекты и потоки в диаграммах потоках данных

Поток управления служит для определения условий выполнения соответствующего процесса. Потоки управления изображаются пунктирными стрелками и показывают, в каком порядке вычисляются значения (рис. 1.6).

Для сложных информационных систем строится иерархия контекстных диаграмм. При этом контекстная диаграмма верхнего уровня содержит не один главный процесс, а набор подсистем, соединенных потоками данных. Контекстные диаграммы следующего уровня детализируют контекст и структуру подсистем.

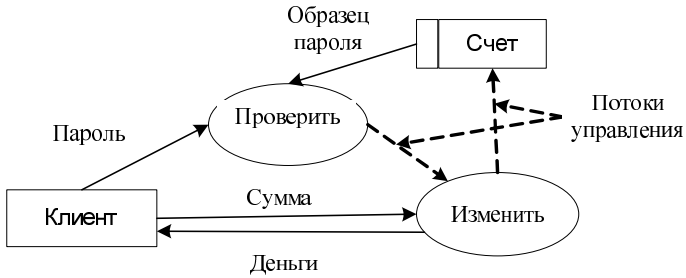


Рис. 1.6. Активные объекты и потоки в диаграммах потоков данных

На рис. 1.7 приведена диаграмма потоков данных верхнего уровня с ее последующим уточнением.

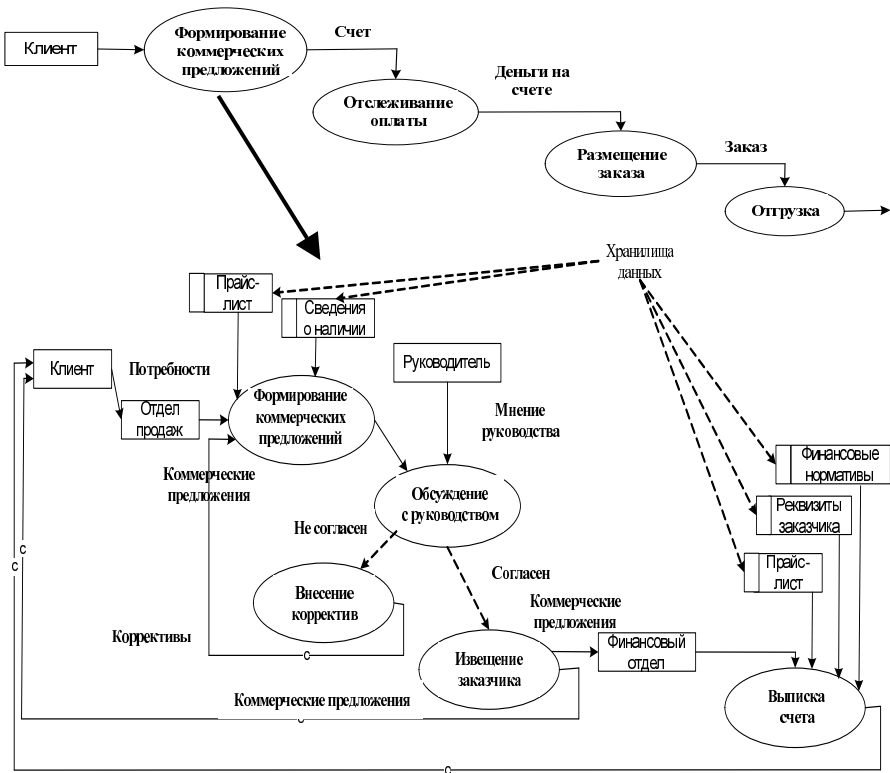


Рис. 1.7. Декомпозиция диаграммы потоков данных

1.3. Методология объектного проектирования UML

В языке моделирования UML (Unified Modeling Language) определено восемь видов диаграмм:

- **диаграмма прецедентов (Use case diagram)** – диаграмма поведения, на которой показаны множество прецедентов и субъектов, а также отношения между ними;
- **диаграмма деятельности (Activity diagram)** – диаграмма поведения, на которой показан автомат и подчеркнуты переходы потока управления от одной деятельности к другой;
- **диаграмма классов (Class diagram)** – структурная диаграмма, на которой показаны множество классов, интерфейсов, коопераций и отношения между ними;
- **диаграмма состояний (Statechart diagram)** – диаграмма поведения, на которой показан автомат и подчеркнуто поведение объектов с точки зрения порядка получения событий;
- **диаграмма последовательностей (Sequence diagram)** – диаграмма поведения, на которой показано взаимодействие и подчеркнута временная последовательность событий;
- **диаграмма кооперации (Collaboration diagram)** – диаграмма поведения, на которой показано взаимодействие и подчеркнута структурная организация объектов, посылающих и принимающих сообщения;
- **диаграмма компонентов (Component diagram)** – диаграмма поведения, на которой показан автомат и подчеркнуто поведение объектов с точки зрения порядка получения событий;
- **диаграмма развертывания (Deployment diagram)** – структурная диаграмма, на которой показаны узлы и отношения между ними.

В последующих разделах рассматриваются первые два вида диаграмм (как наиболее часто используемых).

1.3.1. Диаграммы прецедентов

Диаграмма прецедентов (Use case diagram) – исходная концептуальная модель системы в процессе ее проектирования и разработки. Разработка диаграммы прецедентов преследует цели:

- формулирования общих требований к функциональному поведению проектируемой системы;
- разработки исходной концептуальной модели системы для ее последующей детализации в форме логических и физических моделей;

- подготовка исходной документации для взаимодействия разработчиков системы с ее заказчиками и пользователями.

Диаграммы прецедентов применяются для моделирования вида системы с точки зрения внешнего наблюдателя. На диаграмме прецедентов графически показана совокупность *прецедентов* и *субъектов*, а также отношения между ними.

Субъект (actor) – любая сущность (человек, техническое устройство, программа и т. д.), взаимодействующая с системой извне, или множество логически связанных ролей, исполняемых при взаимодействии с прецедентами. Графически субъект на диаграммах обозначается фигурой человека, под которой записывается конкретное имя субъекта

Прецеденты (use case) – это описание последовательностей действий (включая их варианты), которые выполняются системой для того, чтобы субъект получил результат, имеющий для него определенное значение. Графически прецедент на диаграммах обозначается эллипсом, внутри которого содержится краткое название прецедента или имя в форме глагола с пояснительными словами.

Между субъектами и прецедентами – основными компонентами диаграммы прецедентов могут существовать различные отношения, которые описывают взаимодействие экземпляров одних субъектов и прецедентов с экземплярами других субъектов и прецедентов. В языке UML имеется несколько стандартных видов отношений между субъектами и прецедентами;

- **отношение ассоциации (association)**, обозначаемое сплошной линией, определяет наличие канала связи между экземплярами субъекта и прецедента (или между экземплярами двух субъектов);

- **отношение расширения (extend)**, обозначаемое пунктирной линией, направленной от того прецедента, который является расширением для прецедента исходного. Помечается ключевым словом *extend* (*расширять*), определяет взаимосвязь экземпляров отдельного прецедента с более общим прецедентом, свойства которого определяются на основе способа совместного объединения данных экземпляров;

- **отношение включения (include)**, обозначаемое пунктирной линией со стрелкой, направленной от базового прецедента к включаемому, и помечаемое ключевым словом *include* (*включать*). Указывает, что некоторое заданное поведение для одного прецедента включает в качестве составного компонента поведение другого прецедента;

- **отношение обобщения (generalization)**, обозначаемое сплошной линией с незакрашенной стрелкой которая указывает на родительский прецедент, сообщает что некоторый прецедент А может быть обобщен до прецедента В.

Ниже на рис. 1.8 приведен пример диаграммы прецедентов.

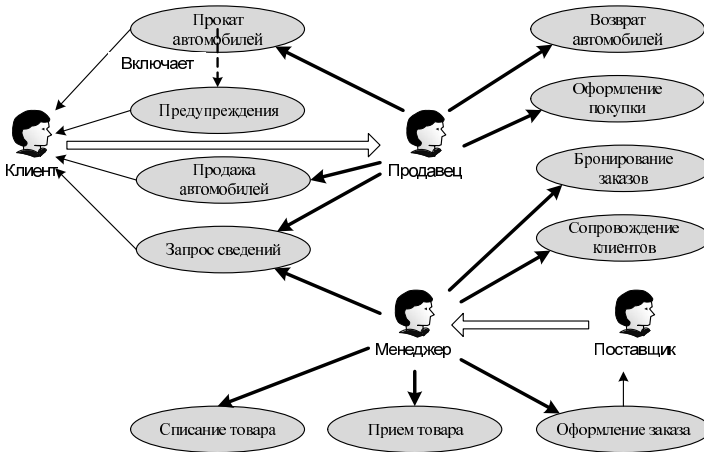


Рис. 1.8. Диаграмма прецедентов

Каждый прецедент на диаграмме прецедентов должен сопровождаться собственной спецификацией, содержащей следующие разделы:

- краткое описание прецедента;
- описание участвующих субъектов;
- предусловия, необходимые для инициирования прецедента;
- описание потока событий;
- постусловия, по достижении которых прецедент завершается.

1.3.2. Диаграммы деятельности

Диаграмма деятельности (Activity diagram) – это по существу блок-схема, которая показывает, как поток управления переходит от одной деятельности к другой. От традиционной блок-схемы диаграмма деятельности отличается более высоким уровнем абстракции и возможностью демонстрации управления параллельными потоками (наряду с последовательным управлением).

Разработка диаграммы деятельности необходима:

- для детализации особенностей алгоритмической и логической реализации выполняемых системой операций и прецедентов;

- выделения последовательных и параллельных потоков управления;
- подготовки детальной документации для взаимодействия разработчиков системы с ее заказчиками и проектировщиками.

Диаграмма деятельности представляется в форме графа деятельности, вершинами которого являются состояния действия (или состояния деятельности), а дугами – переходы от одного состояния действия/деятельности к другому. Саму диаграмму деятельности принято располагать таким образом, чтобы действия следовали сверху вниз. В этом случае начальное состояние будет изображаться в верхней части диаграммы, а конечное – в ее нижней части. Ниже представлены основные элементы диаграммы деятельности.

Состояние деятельности (Activity, Process) – это продолжающийся во времени неатомарный шаг вычислений в автомате. Состояния деятельности могут быть подвергнуты дальнейшей декомпозиции, вследствие чего выполняемую деятельность можно представить с помощью других диаграмм деятельности.

Состояние действия (Action state) – это вычисление атомарного действия, как правило – вызов операции. Состояния действия атомарны и не могут быть подвергнуты декомпозиции.

Состояния деятельности и состояния действия имеют одинаковое стандартное графическое обозначение – прямоугольник с закругленными краями. Внутри такого символа записывают произвольное выражение (action-expression), которое должно быть уникальным в пределах одной диаграммы деятельности.

Начальное и конечное состояния на диаграммах деятельности изображаются в виде закрашенного круга и закрашенного круга внутри окружности соответственно (рис. 1.9).

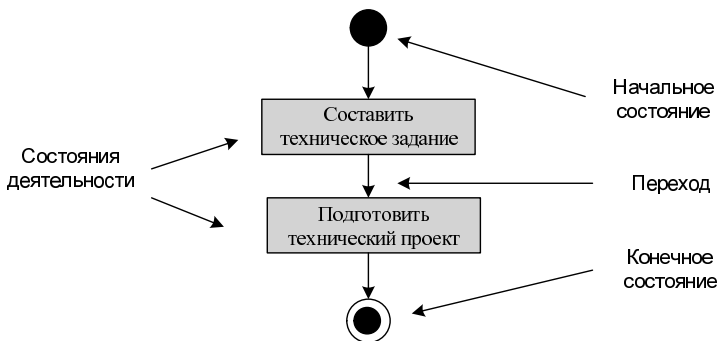


Рис. 1.9. Элементы диаграммы деятельности

Переход (Transitions) – отношение между двумя состояниями, показывающее, что объект, находящийся в первом состоянии, должен выполнить некоторые действия и перейти во второе состояние. На языке UML переход представляется простой линией со стрелкой.

Ветвление описывает различные пути выполнения в зависимости от значения некоторого булевского выражения. Графически точка ветвления представляется ромбом.

Разделение (Concurrent fork) и **слияние (Concurrent join)** служат для организации параллельных потоков, которые часто необходимы для моделирования бизнес-процессов. Графически на языке UML для обозначения разделения и слияния используются обозначения, приведенные на рис. 1.10.



Рис. 1.10. Разделение и слияние в диаграммах деятельности

Дорожка (Swimlane) служит для описания связанных работ, каждая из которых выполняется параллельно (рис. 1.11).

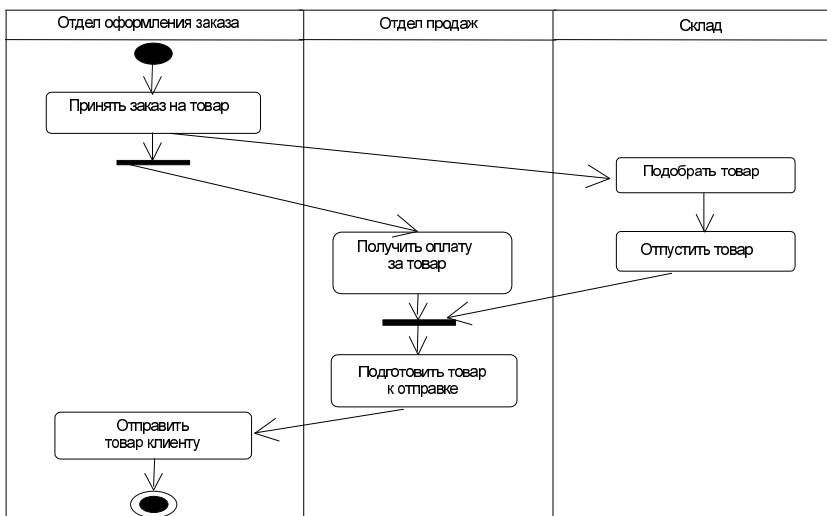


Рис. 1.11. Дорожки в диаграммах деятельности

На рис. 1.12 приведены диаграммы деятельности.

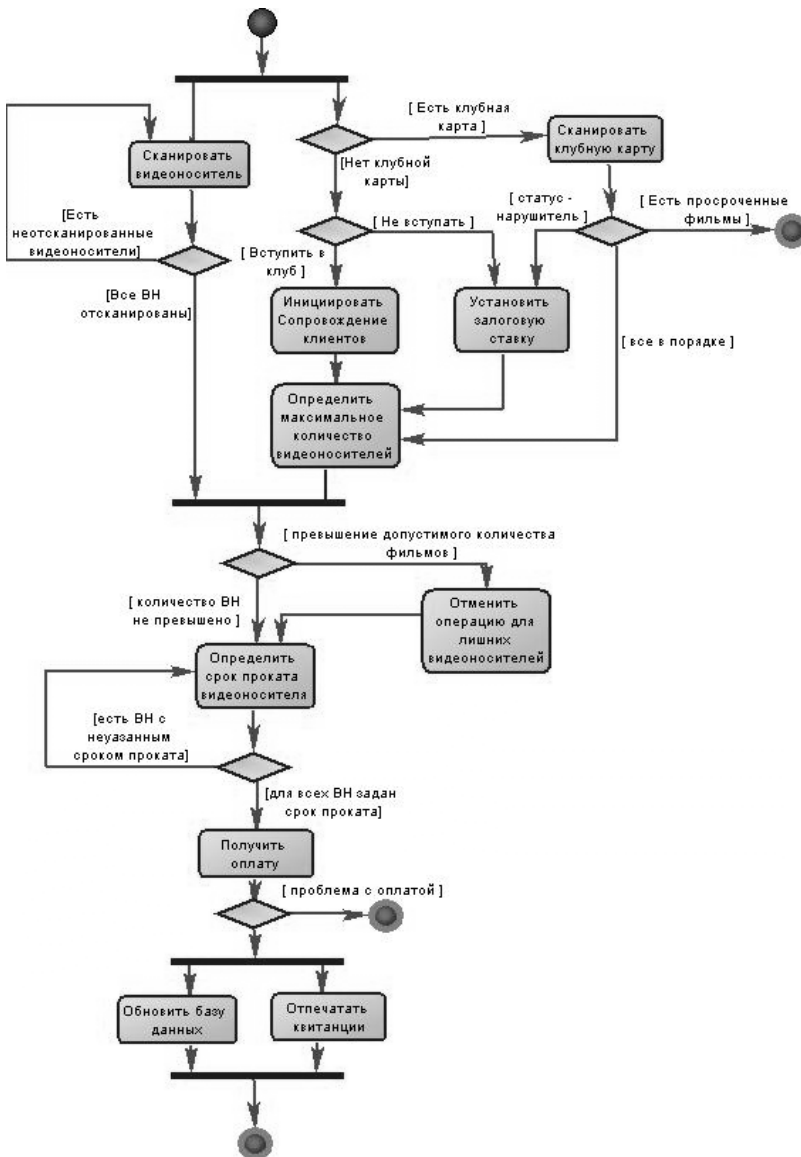


Рис. 1.12. Диаграммы деятельности магазина видеопроката

2. ПРОЕКТИРОВАНИЕ БАЗ ДАННЫХ

2.1. Этапы проектирования базы данных

При проектировании базы данных решаются три основные проблемы.

1. Как адекватно отразить предметную область и информационные потребности пользователей в концептуальной модели? Эту проблему называют *проблемой инфологического проектирования баз данных*. Цель инфологического этапа проектирования состоит в получении семантических (смысловых) моделей, отражающих информационное содержание проблемы.

2. Каким образом представить объекты предметной области в абстрактных объектах моделей данных так, чтобы это отображение не противоречило семантике предметной области и было максимально удобным? Эта проблема известна как *проблема логического проектирования баз данных*. Цель логического этапа проектирования – организация данных, выделенных на предыдущем этапе, в форму, принятую в выбранной СУБД.

3. Как обеспечить эффективность выполнения запросов к базе данных? Каким образом для конкретной СУБД расположить данные во внешней памяти? Создание каких дополнительных структур (например, индексов) необходимо потребовать и т. д.? Эту проблему называют *проблемой физического проектирования баз данных*. Цель физического этапа – выбор рациональной структуры хранения данных и методов доступа к ним.

2.2. Инфологическое проектирование

Одна из наиболее популярных семантических моделей данных на этапе инфологического проектирования – ER-модель (entity-relationship model).

Основные понятия ER-модели – *сущность, связь и атрибут* [1, 2, 4, 5, 7, 8, 11].

Сущность – это реальный или представляемый объект предметной области, информация о котором должна сохраняться и быть доступна. Различают тип сущности и экземпляр сущности. *Тип сущности* относится к набору однородных предметов, событий, личностей, высту-

пающих как единое целое. *Экземпляр сущности* относится к конкретной вещи в наборе. В диаграммах ER-модели сущность представляется в виде прямоугольника, содержащего ее имя.

Для распознавания конкретных экземпляров сущности используются атрибуты, выполняющие роль идентификаторов.

Атрибут – поименованная характеристика сущности, определяющая его свойства и принимающая значения из некоторого множества. Каждому атрибуту дается имя, уникальное в пределах сущности (<**Имя сущности**>.<**Имя атрибута**>.). Например:

Человек.Возраст

Книга.Объем

Диапазон допустимых значений, которые может принимать атрибут, называется *доменом*.

Замечание. Абсолютное различие между типами сущностей и атрибутами отсутствует. Атрибут является таковым только в связи с типом сущности. В другом контексте атрибут может выступать как самостоятельная сущность. Например, для автомобильного завода цвет – это атрибут продукта производства, для лакокрасочной фабрики цвет – тип сущности.

Множество из одного или нескольких атрибутов, значения которых однозначно определяют каждый экземпляр сущности, называется *идентификатором (ключом)*. Каждая сущность должна иметь хотя бы один идентификатор. Атрибуты могут классифицироваться по принадлежности к одному из трех различных типов: *описательные, указывающие, вспомогательные*.

Описательные атрибуты представляют факты, внутренне присущие каждому экземпляру сущности:

Счет.Сальдо

Источник электроснабжения.Полярность

Кошка.Вес

Если значение описательного атрибута меняется, то это говорит о том, что некоторый аспект экземпляра сущности изменился, но сам экземпляр остался прежним.

Указывающие атрибуты используются для присвоения имени или обозначения экземплярам сущности:

Счет.Номер

Груз.Номер накладной

Город.Название

Изменение значения указывающего атрибута говорит о том, что данному экземпляру дается новое имя. Указывающие атрибуты часто используются как идентификатор или как часть идентификатора.

Вспомогательные атрибуты используются для связи экземпляра одной сущности с экземпляром другого:

Счет.ID клиента

Кошка.Имя хозяина

Магнит.Источник электроснабжения

Если значение вспомогательного атрибута меняется, это означает, что теперь другие экземпляры связаны между собой.

Среди бинарных связей можно выделить три фундаментальные безусловные связи, требующие участия каждого экземпляра сущности.

Связь **один-к-одному (1:1)** существует, когда один экземпляр одной сущности связан с единственным экземпляром другой сущности.

Связь **один-ко-многим (1:M)** возникает, когда один экземпляр одной сущности связан с одним или более экземпляром другой сущности и каждый экземпляр второй сущности связан только с одним экземпляром первой сущности.

Связь **многие-ко-многим (M:N)** существует, когда один экземпляр одной сущности связан с одним или более экземпляром другой сущности и каждый экземпляр второй сущности связан с одним или более экземпляром первой сущности.

Примеры связей приведены на рис 2.1.

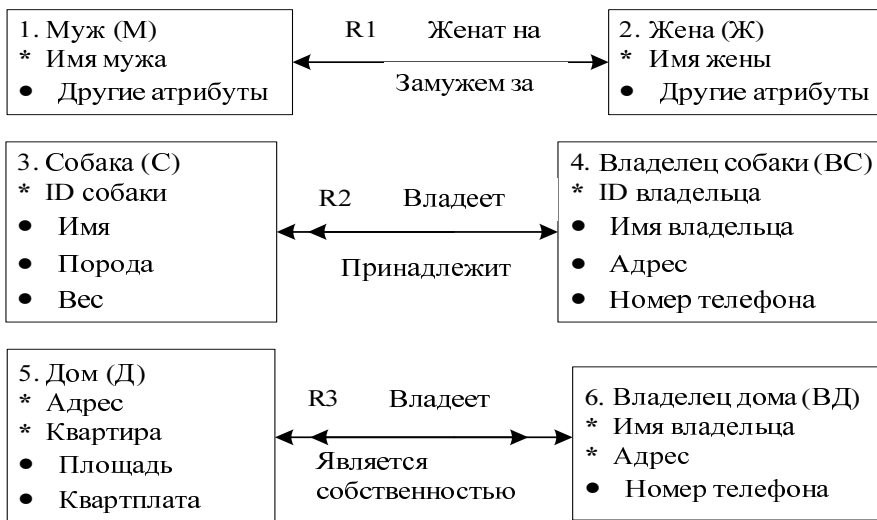


Рис. 2.1. Примеры безусловных связей

В условных связях в отличие от безусловных могут существовать экземпляры сущности, которые не принимают участия в связи. Если связь условная с обеих сторон, она называется *биусловной*. Например, между отношениями **Руководитель – Сотрудник** связь условная, поскольку некоторые сотрудники могут не иметь подчиненных.

Все связи требуют описания. Описание должно включать:

- идентификатор связи;
- формулировку имен связи с точки зрения каждой участвующей сущности;
- вид связи (множественность и условность);
- формулировку того, как связь была формализована.

Цель формулировки связи состоит в том, чтобы позволить установить связь экземпляра одной сущности с экземпляром другой. Это выполняется размещением вспомогательных атрибутов в соответствующих сущностях на модели. Когда это выполнено, говорят, что *связь формализована*.

Для формализации связи *один к одному* вспомогательные атрибуты могут быть добавлены к любой сущности (но не к обеим). Для формализации связи *один ко многим* вспомогательные атрибуты должны быть добавлены к сущности на стороне *многого*, поскольку размещение такого вспомогательного атрибута на стороне *один* будет некорректным. Для формализации связи *многие ко многим* создают отдельную ассоциативную сущность, которая содержит ссылки на идентификаторы каждого из участвующих экземпляров (рис. 2.2).

Различают четыре основных класса сущностей: стержневые, ассоциативные, характеристические и обозначающие.

Стержневая сущность – это независимая сущность. В рассмотренных выше примерах стержневые сущности – это «муж» (но не «жена»), «владелец собаки», «дом», «владелец дома».

Ассоциативная сущность – это сущность, формализующая связь вида *многие ко многим* (... *ко многим* и т. д.) между двумя или более сущностями или связь вида *один к одному* между экземплярами сущностей.

Ассоциативные сущности полноправны: они могут участвовать в других ассоциациях (как стержневые сущности); могут иметь не только набор ключевых атрибутов, необходимых для указания связей, но и любое число других атрибутов, характеризующих связь. В рассмотренных выше примерах ассоциативной является сущность «владение».

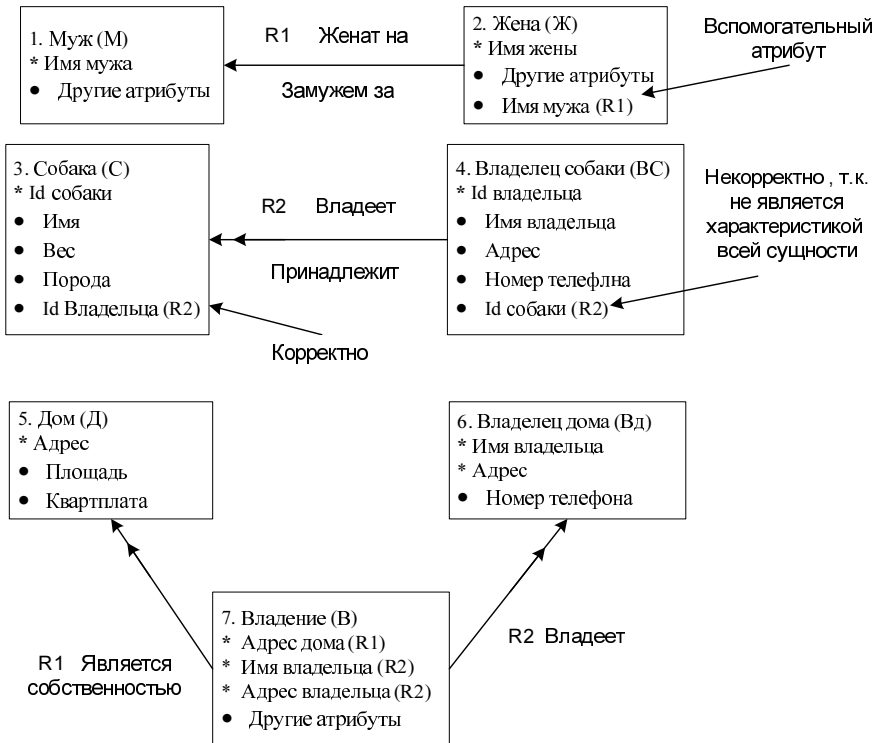


Рис. 2.2. Формализация связей

Характеристическая сущность – это сущность, формализующая связь вида *многие к одной* или *одна к одной* и служащая для уточнения некоторой стержневой сущности.

Существование характеристической сущности полностью зависит от характеризуемой сущности: при удалении экземпляра характеризуемой сущности удаляется экземпляр сущности-характеристики. Характеристическая сущность «жена» характеризует стержневую сущность «муж» (характеризуемая сущность).

Обозначающая сущность – это сущность, формализующая связь вида *многие к одной* или *одна к одной* между двумя сущностями и отличающаяся от характеристической сущности тем, что не зависит от обозначаемой сущности. В рассмотренных выше примерах обозначающая сущность – это сущность «собака», связанная с обозначаемой

сущностью «владелец собаки», имеющая в отличие от характеристики независимое существование (если владелец лишается собаки, конкретный экземпляр собаки продолжает существовать).

К числу более сложных элементов ER-модели относятся:

- подтипы и супертипы сущностей;
- взаимоисключающие связи.

Во многих задачах существуют отчетливые специализированные сущности, имеющие общие атрибуты. Сущность может быть расщеплена на два или более взаимно исключающих *подтипа*, каждый из которых включает общие атрибуты и/или связи. Эти общие атрибуты и/или связи явно определяются один раз на более высоком уровне. В подтипах могут определяться собственные атрибуты и/или связи. Сущность, на основе которой определяются подтипы, называется *супертипом*.

Супертип ЧЕЛОВЕК на рис. 2.3 имеет две группы подтипов, выделенных по различным признакам: по профессиональному признаку – группы специальностей и по половому признаку – МУЖЧИНА и ЖЕНЩИНА. Все эти подтипы наследуют атрибуты и связи супертипа ЧЕЛОВЕК и могут содержать свои собственные атрибуты и связи. Выделенные подтипы в дальнейшем могут содержать свои дополнительные подтипы.



Рис. 2.3. Подтипы и супертипы

Пример диаграммы из двух сущностей со взаимно исключающими связями показан на рис. 2.4. Автомобиль может находиться в рабочем состоянии, и тогда у него имеется только один водитель, или же автомобиль может находиться на ремонте на одном из нескольких возмож-

ных авторемонтных мастерских (каждое предприятие может производить ремонт нескольких автомобилей).

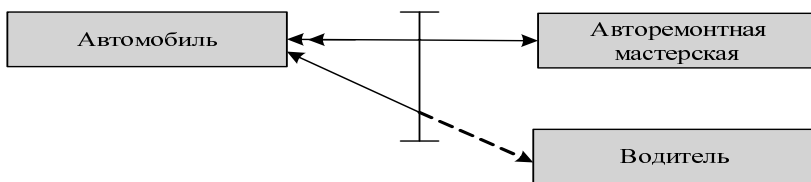


Рис. 2.4. Взаимоисключающие связи

Ниже на рис. 2.5 приведен пример ER-диаграммы, описывающей сущности и связи различных типов на этапе инфологического проектирования для некоторой предметной области.

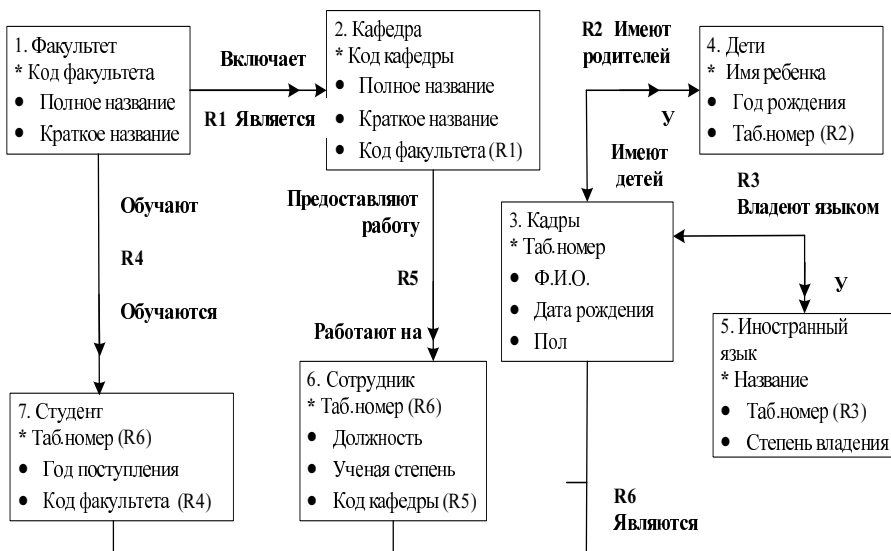


Рис. 2.5. Пример ER-диаграммы

В заключение отметим, что для одной и той же предметной области могут быть построены совершенно разные диаграммы инфологического проектирования, что обусловлено различными бизнес-процессами этой предметной области.

2.3. Логическое проектирование реляционных баз данных

Задача логического проектирования реляционной базы данных состоит в обоснованном принятии решений о том,

- из каких отношений должна состоять база данных и
- какие атрибуты должны быть у этих отношений.

В основе процесса логического проектирования лежит метод нормализации, состоящий в декомпозиции отношения, находящегося в предыдущей нормальной форме, на два или более отношений, которые удовлетворяют требованиям следующей нормальной формы [2, 4–8].

Процесс проектирования реляционной базы на основе метода нормализации преследует две основные цели:

- избежать избыточности хранения данных;
- устранить аномалии обновления отношений.

В теории реляционных баз данных обычно выделяется следующая последовательность нормальных форм:

- первая нормальная форма (1НФ);
- вторая нормальная форма (2НФ);
- третья нормальная форма (3НФ);
- третья усиленная нормальная форма, или нормальная форма Бойса–Кодда (НФБК);
- четвертая нормальная форма (4НФ);
- пятая нормальная форма, или нормальная форма проекции-соединения (5НФ, или PJ/NF);
- устранение аномалий обновления отношений.

Каждая следующая нормальная форма является более ограниченной, но более удобной для практической работы. При переходе к следующей нормальной форме положительные свойства предыдущих нормальных форм сохраняются.

Первая нормальная форма (1НФ) – это отношение, удовлетворяющее фундаментальным свойствам реляционной модели. Отношение в 1НФ обладает следующими свойствами:

- в отношении нет одинаковых кортежей;
- кортежи не упорядочены;

- атрибуты не упорядочены;
- все значения атрибутов атомарны.

Нормализация отношений вплоть до нормальной формы Бойса–Кодда основывается на понятии функциональной зависимости и теореме Хеза, которая гарантирует, что декомпозиция отношений, имеющих функциональные зависимости, будет происходить без потерь информации.

Определение 1. Неключевым атрибутом называется любой атрибут отношения, не входящий в состав ключа (в частности первичного).

Определение 2. В отношении $R\{X, Y, \dots\}$ атрибут Y функционально зависит от атрибута X (функциональная зависимость, functional dependency – FD) в том и только в том случае, если каждому значению X соответствует в точности одно значение Y ($R.X \rightarrow R.Y$).

При этом говорят, что *атрибут X является детерминантом для атрибута Y .*

Определение 3. Функциональная зависимость $R.X \rightarrow R.Y$ называется *полной*, если атрибут Y зависит от атрибута X , но не зависит от любого точного подмножества X .

Считаются правильными такие декомпозиции отношения, при которых возможно собрать исходное отношение из декомпозированных отношений без потери информации. Такие декомпозиции называются *декомпозициями без потерь*.

Определение 4. Под декомпозицией без потерь понимается такой способ декомпозиции отношения, при котором исходное отношение полностью и без избыточности восстанавливается путем естественного соединения полученных отношений. Условие, при котором декомпозиция является декомпозицией без потерь, определяется теоремой Хеза.

Теорема Хеза. Пусть задано отношение $R\{A, B, C\}$ (атрибуты A, B и C в общем случае являются составными атрибутами) и выполняется функциональная зависимость $A \rightarrow B$. Тогда справедливо

$$R = (R \text{ PROJECT } \{A, B\}) \text{ NATURAL JOIN } (R \text{ PROJECT } \{A, C\}).$$

Для отношения, приведенного на рис. 2.6, первая декомпозиция удовлетворяет теореме Хеза и является декомпозицией без потерь,

Исходное отношение

Id_номер	Имя_студента	Стипендия	Номер_КП	Рук_КП
1845	Антонов	1400	2	Борисов
2384	Петров	1400	3	Абрамов

Первая декомпозиция исходного отношения

Id_номер	Имя_студента	Стипендия
1845	Антонов	1400
2384	Петров	1400

Id_номер	Номер_КП	Рук_КП
1845	2	Борисов
2384	3	Абрамов

Вторая декомпозиция исходного отношения

Id_номер	Имя_студента	Стипендия
1845	Антонов	1400
2384	Петров	1400

Стипендия	Номер_КП	Рук_КП
1400	2	Борисов
1400	3	Абрамов

Рис. 2.6. Декомпозиция без потерь

вторая декомпозиция таковой не является, поскольку не удовлетворяет теореме Хеза и не позволяет путем естественного соединения без потерь восстановить исходное отношение.

2.3.1. Полные функциональные зависимости и вторая нормальная форма

Рассмотрим отношение:

СТУДЕНТЫ {Id_номер, Тип_стипендии, Стипендия, Номер_проекта, Номер_задания}.

Тело записанного отношения приведено на рис. 2.7.

Отношение СТУДЕНТЫ

Id_номер	Тип_стипендии	Стипендия	Номер_проекта	Номер_задания
1845	1	1400	1	1
2384	1	1400	1	2
3322	2	1800	1	3
6754	3	2500	1	4
1845	1	1400	2	3
2384	1	1400	2	2
3322	2	1800	2	1

Рис. 2.7. Исходное отношение СТУДЕНТЫ

Будем полагать, что выполняются следующие бизнес-правила.

1. Величина стипендии зависит от ее типа (обычная, повышенная, стипендия мэрии, президентская стипендия и пр.).

2. Каждый студент может участвовать в нескольких исследовательских проектах, но в каждом проекте он выполняет только одно задание.

Исходя из записанных условий единственно возможным ключом отношения является составной атрибут $\{Id_номер, Номер_проекта\}$. Однако для ряда атрибутов (*Тип_стипендии*, *Стипендия*) детерминантом является не возможный ключ отношения $\{Id_номер, Номер_проекта\}$, а часть ключа *Id_номер*. Это приводит к так называемым аномалиям обновления. Под аномалиями обновления понимаются трудности, с которыми приходится сталкиваться при выполнении операций добавления кортежей в отношение (INSERT), удаления кортежей (DELETE) и модификации кортежей (UPDATE):

– **добавление кортежей.** Невозможно занести в отношение СТУДЕНТЫ студента, который в данное время еще не участвует ни в одном проекте (атрибут *Номер_проекта* является частью первичного ключа и не может содержать неопределенных значений);

– **удаление кортежей.** Невозможно сохранить в отношении СТУДЕНТЫ данные о студенте, завершившем участие в своем последнем проекте потому, что значение атрибута *Номер_проекта* становится неопределенным. Однако вероятны ситуации, когда между выполнением проектов у студента могут возникать перерывы;

– **модификация кортежей.** Чтобы изменить стипендию студента, необходимо модифицировать все кортежи с соответствующим значением атрибута *Id_номер*;

– **избыточность данных.** В отношении прослеживается избыточность хранения значений атрибутов *Тип_стипендии* и *Стипендия* в каждом кортеже, описывающем задание студента в некотором проекте.

Для преодоления этих трудностей можно произвести декомпозицию отношения **СТУДЕНТЫ** на два отношения:

СТУДЕНТЫ_СТИПЕНДИЯ {*Id_номер*, *Тип_стипендии*, *Стипендия*} и

СТУДЕНТЫ_ПРОЕКТЫ {*Id_номер*, *Номер_проекта*, *Номер_задания*}.

Поскольку выполняются все условия теоремы Хеза, эта декомпозиция является декомпозицией без потерь (рис. 2.8).

Отношение СТУДЕНТЫ_СТИПЕНДИЯ		
<i>Id_номер</i>	<i>Тип_стипендии</i>	<i>Стипендия</i>
1845	1	1400
2384	1	1400
3322	2	1800
6754	3	2500

Отношение СТУДЕНТЫ_ПРОЕКТЫ		
<i>Id_номер</i>	<i>Номер_проекта</i>	<i>Номер_задания</i>
1845	1	1
2384	1	2
3322	1	3
6754	1	4
1845	2	3
2384	2	2
3322	2	1

Рис. 2.8. Декомпозиция отношения **СТУДЕНТЫ**

С преобразованными отношениями легко выполнить операции обновления (добавление, удаление, модификацию), кроме того, для каждого студента атрибуты *Тип_стипендии* и *Стипендия* хранятся в одном экземпляре.

Определение 5. Неключевой атрибут функционально полно зависит от составного ключа, если он функционально зависит от ключа, но не находится в функциональной зависимости ни от какой части ключа.

Определение 6. Отношение находится в 2НФ, если оно находится в 1НФ и каждый неключевой атрибут функционально полно зависит от первичного ключа. Для приведения отношения в 2НФ необходимо:

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

Любое отношение, находящееся в 1 НФ, но не находящаяся в 2НФ, может быть приведено к набору отношений, находящихся в 2НФ. В результате декомпозиции получается набор проекций исходного отношения, естественное соединение которых восстанавливает исходное отношение.

2.3.2. Транзитивные зависимости и третья нормальная форма

В отличие от отношения **СТУДЕНТЫ_ПРОЕКТЫ** отношению **СТУДЕНТЫ_СТИПЕНДИЯ** присущи аномалии обновления, которые вызваны наличием транзитивной зависимости $Id_номер \rightarrow Стипендия$ ($Id_номер \rightarrow Тип_стипендии$ и $Тип_стипендии \rightarrow Стипендия$). Эти аномалии связаны с избыточностью хранения значения атрибута *Стипендия* в каждом кортеже, характеризующем студентов с одним и тем же атрибутом *Тип_стипендии*. Аномалии обновления проявляются в следующем.

– **Добавление кортежей.** Невозможно сохранить данные о соотношении типа стипендии и ее размера, пока не появится студент с данным типом стипендии (первичный ключ не может содержать неопределенные значения).

– **Удаление кортежей.** При удалении последнего студента с данным типом стипендии теряется информация о размере стипендии данного типа.

– **Модификация кортежей.** При изменении размера стипендии, соответствующей некоторому типу, придется вынужденно изменить значение атрибута *Стипендия* в кортежах всех студентов, которым выплачивается стипендия данного типа.

– **Избыточность данных.** Информация о размере стипендии для всех студентов, получающих стипендию одного типа, дублируется.

Преодолеть эти трудности можно декомпозицией отношения **СТУДЕНТЫ_СТИПЕНДИЯ** на два отношения:

СТУДЕНТЫ{*Id_номер*, *Тип_стипендии*} и
СТИПЕНДИЯ {*Тип_стипендии*, *Стипендия*}.

По теореме Хеза, эта декомпозиция является декомпозицией без потерь (рис. 2.9).

Отношение СТУДЕНТЫ		Отношение СТИПЕНДИЯ	
Id_номер	Тип_стипендии	Тип_стипендии	Стипендия
1845	1	1	1400
2384	1	2	1800
3322	2	3	2500
6754	3		

Рис. 2.9. Декомпозиция отношения СТУДЕНТЫ_ СТИПЕНДИЯ

Легко заметить, что выполненная декомпозиция избавляет от перечисленных выше трудностей при выполнении операций обновления.

Определение 7. Отношение находится в 3НФ только в том случае, если оно находится в 2НФ и каждый неключевой атрибут нетранзитивно зависит от первичного ключа.

Любое отношение, находящееся в 2НФ, но не находящееся в 3НФ, может быть приведено к набору отношений, находящихся в 3НФ, естественное соединение которых воспроизводит исходное отношение.

2.3.3. Нормальная форма Бойса–Кодда

В случае, когда у отношения имеется несколько возможных ключей и при этом некоторые из этих возможных ключей «перекрываются» (т. е. содержат общие атрибуты), могут иметь место аномалии обновления.

Рассмотрим отношение со схемой

СТУДЕНТЫ{Id_номер, Фамилия, Номер_проекта, Номер_задания} и телом, приведенным на рис. 2.10.

Отношение СТУДЕНТЫ			
Id_номер	Фамилия	Номер_проекта	Номер_задания
1845	Антонов	1	1
2384	Петров	3	2
1845	Антонов	2	3
2384	Петров	1	1

Рис. 2.10. Исходное отношение СТУДЕНТЫ

Предполагается выполнение следующих бизнес-правил.

1. Студенты идентифицируются как по номерам студенческих билетов, так и по фамилиям (существуют функциональные зависимости $Id_номер \rightarrow Фамилия$ и $Фамилия \rightarrow Id_номер$).

2. Каждый студент может участвовать в нескольких исследовательских проектах и в каждом проекте он выполняет только одно задание.

Исходя из перечисленных правил, возможные первичные ключи – $\{Id_номер, Номер_проекта\}$ и $\{Фамилия, Номер_проекта\}$.

Хотя в отношении все неключевые атрибуты функционально полно зависят от первичных ключей и транзитивные зависимости отсутствуют, отношению свойственны аномалии обновления, связанные с тем, что в случае изменения фамилии студента требуется обновить атрибут *Фамилия* во всех кортежах отношения, соответствующих данному студенту.

Определение 8. Отношение находится в нормальной форме Бойса–Кодда (НФБК) в том и только в том случае, когда детерминанты всех ее функциональных зависимостей являются потенциальными ключами.

В исходном отношении имеются функциональные зависимости $Id_номер \rightarrow Фамилия$ и $Фамилия \rightarrow Id_номер$, для которых ключ данного отношения не является детерминантом. Отношение **СТУДЕНТЫ** может быть приведено к 3CNF путем декомпозиции на два отношения:

СТУДЕНТЫ1 $\{Id_номер, Фамилия\}$ и

СТУДЕНТЫ_ПРОЕКТЫ $\{Id_номер, Номер_проекта, Номер_задания\}$

Результат декомпозиции представлен на рис. 2.11.

Отношение СТУДЕНТЫ1		Отношение СТУДЕНТЫ_ПРОЕКТЫ		
Id_номер	Фамилия	Id_номер	Номер_проекта	Номер_задания
1845	Антонов	1845	1	1
2384	Петров	2384	3	2
		1845	2	3
		2384	1	1

Рис. 2.11. Декомпозиция отношения **СТУДЕНТЫ**

В полученных отношениях других функциональных зависимостей, кроме зависимости от ключа, нет, поэтому данная декомпозиция устраняет трудности, связанные с обновлением отношения **СТУДЕНТЫ**.

2.3.4. Многозначная зависимость и четвертая нормальная форма

Рассмотрим еще одну возможную интерпретацию отношения **СТУДЕНТЫ**. Предположим, что каждый студент может участвовать в нескольких исследовательских проектах одновременно, но в каждом проекте должен выполнять одни и те же задания (рис. 2.12).

Отношение **СТУДЕНТЫ**

Id_номер	Номер_проекта	Номер_задания
1845	1	1
1845	1	2
1845	2	1
1845	2	2
...
2384	1	1
2384	1	3

Рис. 2.12. Отношение **СТУДЕНТЫ**

При данных условиях единственно возможный ключ – заголовок отношения $\{Id_номер, Номер_проекта, Номер_задания\}$. Никаких функциональных зависимостей, кроме тривиальных, нет. Отношение находится в НФБК. Однако в отношении присутствуют следующие аномалии обновления.

- **Добавление кортежа.** Если уже участвующий в проектах студент присоединяется к новому проекту, то к телу отношения **СТУДЕНТЫ** необходимо добавить столько кортежей, сколько заданий выполняет этот студент.
- **Удаление кортежей.** Если студент прекращает участие в проектах, то невозможно сохранить данные о заданиях, которые он может выполнять.
- **Модификация кортежей.** При изменении одного из заданий студента необходимо изменить значение атрибута *Номер_задания* в стольких кортежах, в скольких проектах участвует студент.

Перечисленные трудности решает декомпозиция исходного отношения на два отношения (рис. 2.13):

СТУДЕНТЫ_ПРОЕКТЫ {*Id_номер*, *Номер_проекта*} и
СТУДЕНТЫ_ЗАДАНИЯ {*Id_номер*, *Номер_задания*}.

Id_номер	Номер_проекта
1845	1
1845	2
...	...
2384	1

Id_номер	Номер_задания
1845	1
1845	2
...	...
2384	1
2384	3

Рис. 2.13. Декомпозиция отношения **СТУДЕНТЫ**

Легко проверить, что данная декомпозиция является декомпозицией без потерь и что она решает перечисленные выше проблемы с обновлением отношения **СТУДЕНТЫ**.

- **Добавление кортежа.** Если уже участвующий в проектах студент присоединяется к новому проекту, то к телу значения отношения **СТУДЕНТЫ_ПРОЕКТЫ** требуется добавить еще один кортеж, соответствующий новому проекту.

- **Удаление кортежей.** Если студент прекращает участие в проектах, то данные о заданиях, которые он может выполнять, остаются в отношении **СТУДЕНТЫ_ЗАДАНИЯ**.

- **Модификация кортежей.** При изменении одного из заданий студента необходимо изменить значение атрибута *Номер_задания* в одном кортеже отношения **СТУДЕНТЫ_ЗАДАНИЯ**.

Поскольку в исходном отношении **СТУДЕНТЫ** отсутствуют нетривиальные функциональные зависимости, принципы нормализации на основе FD здесь неприменимы. Для данного варианта отношения **СТУДЕНТЫ** имеет место зависимость, впервые выявленная Р. Фейджином и названная им *многозначной (multi-valued dependency – MVD)*, поскольку в ней одному значению детерминанта соответствует множество значений зависимого атрибута. В отношении **СТУДЕНТЫ** выполняются две такие многозначные зависимости: $Id_номер \twoheadrightarrow Номер_проекта$ и $Id_номер \twoheadrightarrow Номер_задания$.

Первая MVD означает, что каждому значению атрибута *Id_номер* соответствует определяемое только этим значением множество значений атрибута *Номер_проекта*. Вторая MVD означает, что каждому значению атрибута *Id_номер* соответствует определяемое только этим значением множество значений атрибута *Номер_задания*.

Определение 9. В отношении R с атрибутами A, B, C (в общем случае, составными) имеется многозначная зависимость B от A (обозначается $A \twoheadrightarrow B$) в том и только в том случае, если множество значений атрибута B, соответствующее паре значений атрибутов A и C, зависит от значения A и не зависит от значения C.

Многозначные зависимости обладают свойством *двойственности* (лемма Фейджина): в отношении R {A, B, C} выполняется многозначная зависимость $A \twoheadrightarrow B$ в том и только в том случае, когда выполняется многозначная зависимость $A \twoheadrightarrow C$. Свойство двойственности обозначается как $A \twoheadrightarrow B \mid C$.

Полученные путем декомпозиции отношения **СТУДЕНТЫ_ПРОЕКТЫ** и **СТУДЕНТЫ_ЗАДАНИЯ** содержат единственную многозначную зависимость. Правомочность этой декомпозиции доказывается приведенной ниже теоремой Фейджина, которая является обобщением теоремы Хеза.

Теорема Фейджина. Пусть имеется отношение R с атрибутами A, B, C (в общем случае – составными). Отношение R декомпозируется без потерь на проекции {A, B} и {A, C} тогда и только тогда, когда для него выполняется MVD $A \twoheadrightarrow B \mid C$ (когда есть две многозначные зависимости).

Теорема Фейджина служит основой для декомпозиции отношений, удаляющих аномальные многозначные зависимости, с приведением отношений в четвертую нормальную форму.

Определение 10. Отношение R находится в четвертой нормальной форме (4НФ) в том и только в том случае, если оно находится в 3НФ (или НФБК) и при наличии единственной многозначной зависимости $A \twoheadrightarrow B$, все остальные атрибуты отношения R функционально зависят от детерминанта этой многозначной зависимости.

Отношение **СТУДЕНТЫ** не находится в 4НФ, поскольку содержит две MVD. Отношения **СТУДЕНТЫ_ПРОЕКТЫ** и **СТУДЕНТЫ_ЗАДАНИЯ** находятся в 4НФ, поскольку содержат только одну многозначную зависимость, и других функциональных зависимостей в них нет.

2.3.5. Зависимость проекции/соединения и пятая нормальная форма

Однако бывают (хотя и нечасто) случаи, когда декомпозиция без потерь на две проекции невозможна, но можно произвести декомпозицию без потерь на большее число проекций.

Определение 11. В отношении R с произвольными атрибутами A, B, Z (в общем случае – составными) удовлетворяется *зависимость проекции/соединения* (*project-join dependency – PJD*) * (A, B, \dots, Z) тогда и только тогда, когда отношение R можно получить путем естественного соединения проекций этого отношения на атрибуты A, B, \dots, Z .

Определение 12. Отношение находится в пятой нормальной форме, или нормальной форме проекции-соединения (*project-join normal form – 5НФ*, или *PJ/NF*) в том и только в том случае, если в каждой нетривиальной PJD все проекции содержат возможный ключ.

В лекции по курсу есть пример приведения отношения к 5НФ через декомпозицию его на три отношения. Обнаружение всех зависимостей соединения – нетривиальная задача, для решения которой нет общих методов.

Выбор степени нормализации отношений зависит от существующих бизнес-правил предметной области и характера запросов, с которыми чаще всего обращаются к базе данных.

Как правило, третьей нормальной формы достаточно в подавляющем большинстве практических случаев. При этом можно дать следующие рекомендации.

Критерий	Отношения слабо нормализованы (1НФ, 2НФ)	Отношения сильно нормализованы (3НФ)
Адекватность базы данных предметной области	ХУЖЕ (–)	ЛУЧШЕ (+)
Легкость разработки и сопровождения базы данных	СЛОЖНЕЕ (–)	ЛЕГЧЕ (+)
Скорость выполнения вставки, обновления, удаления	МЕДЛЕННЕЕ (–)	БЫСТРЕЕ (+)
Скорость выполнения выборки данных	БЫСТРЕЕ (+)	МЕДЛЕННЕЕ (–)

Как видно из таблицы, сильно нормализованные отношения оказываются лучше спроектированы. Они больше соответствуют предметной области, легче в разработке, для них быстрее выполняются операции модификации базы данных. Все это достигается ценой некоторого замедления выполнения операций выборки данных.

У слабо нормализованных отношений единственное преимущество: запросы к базе данных выполняются быстрее. Это связано с тем, что в таких отношениях уже произведено их соединение и на это не тратится время при выборке данных.

3. ОГРАНИЧЕНИЯ ЦЕЛОСТНОСТИ

Ограничение целостности представляет собой некоторое утверждение, которое может быть истинным или ложным в зависимости от состояния базы данных.

Примеры ограничений:

- 1) возраст сотрудника не может быть меньше 18 и больше 65 лет;
- 2) каждый студент имеет уникальный идентификационный номер;
- 3) студент обязан числиться только в одной группе;
- 4) зарплата сотрудника одновременно не может быть увеличена более чем на 100 единиц;
- 5) сумма накладной для всех входящих в нее товаров должна равняться сумме произведений цен товаров на количество товаров.

Некоторые из ограничений целостности являются ограничениями реляционной модели данных. Например, ограничение 2 представляет ограничение, реализующее целостность сущности, ограничение 3 представляет ограничение, реализующее ссылочную целостность. Остальные ограничения (1, 4, 5) достаточно произвольны.

База данных находится в *согласованном (целостном) состоянии*, если выполнены все определенные для нее ограничения целостности [9, 10].

По способам реализации различают:

- декларативную поддержку ограничений целостности, выполняемую средствами языка определения данных (SQL);

- процедурную поддержку ограничений целостности, выполняемую посредством триггеров и хранимых процедур.

В информационных системах, работающих с базами данных, можно выделить три уровня:

- выполнения функций сохранения и выборки данных;
- выполнения правил обработки данных;
- управления интерфейсом пользователя.

Первый уровень – это набор таблиц, а также декларативные и триггерные ограничения целостности, которые в совокупности обеспечивают хранение и непротиворечивость данных.

Третий уровень – это специально разработанное приложение, обеспечивающее корректный и комфортный для пользователя диалог с информационной системой. Такое приложение, как правило, не имеет прямого доступа к базе данных, а обращается ко второму уровню.

Второй уровень – промежуточный – представляет собой набор процедур и функций, которые инкапсулируют элементарные операции с базой данных (SQL-запросы) и обеспечивают соблюдение бизнес-правил.

Чтобы оптимально спроектировать первый и второй уровни, необходимо научиться различать ограничения целостности и бизнес-правила.

К ограничениям целостности относятся наиболее жесткие и критичные ограничения, которые не могут быть нарушены никогда. Например, билет нельзя сдать раньше, чем он был куплен; один самолет не может выполнять два рейса одновременно. Такие ограничения при создании базы данных реализуются декларативными средствами и триггерами.

Если все же существуют обстоятельства (пусть редкие, даже исключительные), при которых допустимо нарушение ограничения, то такое ограничение следует отнести к бизнес-правилам. Например, при исключительных обстоятельствах на работу может быть принято больше сотрудников, чем указано в штатном расписании, билет может быть сдан после того как самолет улетел.

Триггеры для таких ограничений писать не рекомендуется. Бизнес-правила должны быть реализованы через набор процедур и функций для работы с базой данных. При этом обязательно должна быть предусмотрена возможность для обработки нестандартных ситуаций.

3.1. Декларативная поддержка ограничений целостности

Ниже приведены примеры задания ограничений целостности средствами SQL.

1. Задание неопределенных значений столбца в качестве значений по умолчанию и указание недопустимости неопределенных значений столбца.

```
Create table tabnum (
...
name char(20) default null,
...;
Create table tabnum (
...
parol char(8) not null,
...;
```

2. Контроль значений столбца (фраза *check* при описании атрибутов столбца в операторах *Create/Modify table*).

Перечисление значений

```
Create table taboper (
...
type char(1) not null check (type in ('u', 'к',
'a', 'o')),
uchastok smallint default 3 not null check
(uchastok in (1, 3, 4)),
...;
```

Задание диапазона

```
Create table tabzid (
...
ves_s float default 0 not null check (ves_s be-
tween 0 and 2000),
...;
```

Задание условий

```
Create table taboper (
...
shifr_oper smallint check (shifr_oper > 0),
date_sozd date default TODAY not null check
(date_sozd >»01.01.97»),
kol_smallint not null check ((kol >=0 and
kol<=2000)
or (kol >=5000 and kol <= 7000)),
...;
```

3. Задание первичного ключа.

```
Create table taboper (
...
primary key (shifr_oper, zakaz),
...);
```

4. Задание внешнего ключа (*foreign key* и/или фраза *references* при описании атрибутов столбца в операторах *Create/Modify table*).

```
Create table taboper (
```

```
...
```

```
zakaz char(4) not null references tabtechproc,
```

```
techuslovie char(17) references tabtechusl,
```

```
...
```

```
foreign key (shifr_pred_oper, zakaz) references taboper,
```

```
foreign key (n_partii, n_izd) references tabbrak (n_par, n_izd)
```

```
..);
```

5. Каскадное удаление

```
Create table Catalog ( . . . .
```

```
.....
```

```
foreign key (stock_num) references Stock (stock_num) on delete cascade
```

При удалении строк из *Stock* удаляются связанные с ней строки из *Catalog*.

3.2. Процедурная поддержка ограничений целостности

Ниже приведены основные требования к разрабатываемым триггерам.

Правило 1. Триггеры должны реализовать только те ограничения целостности, которые не могут быть заданы декларативными средствами, т. е. с помощью ограничений *NOT NULL*, *UNIQUE*, *PRIMARY KEY*, *DEFAULT*, *CHECK* и *FOREIGN KEY REFERENCES*.

Предположим, что в базе данных библиотеки есть таблица «Выдача литературы»

```
EATE TABLE delivery1
```

```
(id integer PRIMARY KEY,
```

```
-- код выдачи
```

```
id_exemplar integer REFERENCES exemplar(id),
```

```
-- код экземпляра книги
```

```
id_reader integer REFERENCES reader(id),
```

```
-- код читателя
```

```
id_librarian integer REFERENCES librarian(id),
```

```
-- код библиотекаря
```

```
delivery_date date NOT NULL
```

```
-- дата выдачи
```

```
check (delivery_date <= now()),
```

```
return_date date check (return_date <= now()));
```

```
-- дата возврата
```

Неправильное решение

Триггер, проверяющий следующее ограничение: дата возврата книги не может предшествовать дате выдачи.

```
CREATE FUNCTION tf_delivery_bef_ins_upd () RETURNS SETOF trigger
LANGUAGE plpgsql
AS $$
BEGIN
```

```
    if NEW.return_date < NEW.delivery_date --
    then raise exception 'Дата возврата предшествует дате выдачи';
    return NULL;
```

```
    end if;
```

```
    return NEW;
```

```
END;
```

```
$$;
```

```
CREATE TRIGGER trg_delivery_bef_ins_upd
BEFORE INSERT OR UPDATE ON delivery
FOR EACH ROW
EXECUTE PROCEDURE tf_delivery_bef_ins_upd();
```

Написание такого триггера нецелесообразно, поскольку данное ограничение можно создать более простыми декларативными средствами:

```
ALTER TABLE ONLY delivery
```

```
ADD CONSTRAINT ddd CHECK ((delivery_date <= return_date));
```

Правильное решение

Триггер, проверяющий ограничение: экземпляр книги не может одновременно находиться более чем у одного читателя. В терминах базы данных это ограничение можно записать следующим образом: среди строк таблицы **delivery1** с одним и тем же значением **id_exemplar**:

- может быть максимум одна строка с пустой датой возврата (**return_date is null**);

- не может быть строк с пересечением периодов с **delivery_date** по **return_date**.

```
CREATE FUNCTION tf_delivery_bef_ins_upd () RETURNS SETOF trigger
LANGUAGE plpgsql
AS $$
```

```
BEGIN
```

```
if (exists (select id
```

```

from delivery t
where t.id<>new.id
    and
    t.id_exemplar = new.id_exemplar
    and
    t.delivery_date < case when new.return_date is null
                        then now()
                        else new.return_date
                    end
    and
    case when t.return_date is null
        then now()
        else t.return_date
    end >new.delivery_date
)
)
THEN
    raise exception 'Экземпляр одновременно у двух читателей';
    return NULL;
end if;
return NEW;
END;
CREATE TRIGGER trg_delivery_bef_ins_upd
    BEFORE INSERT OR UPDATE ON delivery
    FOR EACH ROW
    EXECUTE PROCEDURE tf_delivery_bef_ins_upd();

```

Правило 2. Триггеры должны быть написаны для всех операций и всех таблиц, которые затрагивает ограничение.

Ограничение целостности, которое реализует триггер, должно быть четко сформулировано как в терминах бизнес-области, так и в терминах базы данных. При формулировании ограничения в терминах базы данных нужно описать допустимое состояние данных в таблицах вне зависимости от производимых над ними действий, а затем выявить все операции с базой данных, которые могут привести к нарушению ограничения. Триггеры должны быть определены для каждой из этих операций, причем недопустима ситуация, когда ограничение касается двух таблиц, но при этом для одной триггер написан, а для другой – нет.

Предположим, что в базе данных гостиничного комплекса имеются таблицы:

```
CREATE TABLE hotel -- Гостиницы
  (id_hotel integer PRIMARY KEY, -- код гостиницы
  name_hotel character(20) NOT NULL, -- название гостиницы
  max_floor integer NOT NULL check( max_ -- количество этажей
floor >0), .....);
CREATE TABLE room -- Номера
  (id_room integer PRIMARY KEY -- код номера
  id_hotel integer REFERENCES Hotel (id_hotel), -- код гостиницы
  number_room character(20) NOT NULL -- номер комнаты
  max_guest integer NOT NULL check (max_ -- количество мест
guest>0),
  floor integer NOT NULL check(floor >0), -- этаж
  .....);
```

Неправильное решение

Проверяются следующие ограничения.

1. При добавлении нового номера: его этаж должен быть не больше количества этажей в гостинице.

2. При вставке новой записи: этаж номера не больше этажности гостиницы.

```
CREATE FUNCTION tf_room_bef_ins() RETURNS SETOF trigger
  LANGUAGE plpgsql
  AS $$
BEGIN
  if (new.floor > (select max_floor from hotel t
                    where t.id_hotel=new.id_hotel)
  )
  THEN
    raise exception 'У номера слишком высокий этаж!';
    return NULL;
  end if;
  return NEW;
END;
$$;
CREATE TRIGGER trg_room_bef_ins
BEFORE INSERT ON room
FOR EACH ROW
EXECUTE PROCEDURE tf_room_bef_ins()
```

Правильное решение

Ограничение в терминах бизнес-области имеет вид: ни один гостиничный номер не может располагаться на этаже, превышающем этажность гостиницы.

Это ограничение для терминов базы данных трансформируется в следующее: значение поля **floor_room** в строке таблицы **room** не должно быть больше, чем значение поля **max_floor** в строке таблицы **hotel** со значением поля **Id_hotel = room.Id_hotel**.

Нарушение ограничения возможно при следующих операциях: вставка и модификация строк таблицы **room**, модификация строк таблицы **hotel**. Для решения задачи требуются два триггера:

- перед вставкой или обновлением строки в таблице **room**;
- перед обновлением строки в таблице **hotel**.

```
CREATE FUNCTION tf_room_bef_ins() RETURNS SETOF trigger  
LANGUAGE plpgsql
```

```
AS $$
```

```
DECLARE
```

```
fl integer;
```

```
BEGIN
```

```
select max_floor into fl  
from hotel
```

```
where hotel.id_hotel=new.id_hotel;
```

```
if (new.floor > fl)
```

```
THEN
```

```
raise exception 'У номера слишком высокий этаж. Максимум –  
%', fl;
```

```
return NULL;
```

```
end if;
```

```
return NEW;
```

```
END;
```

```
$$;
```

```
CREATE TRIGGER trg_room_bef_ins
```

```
BEFORE INSERT OR UPDATE ON room
```

```
FOR EACH ROW
```

```
EXECUTE PROCEDURE tf_room_bef_ins());
```

```
CREATE FUNCTION tf_hotel_bef_upd() RETURNS SETOF trigger
```

```
LANGUAGE plpgsql
```

```
AS $$
```

```
BEGIN
```

```

if (exists (select id_room
            from room t
            where t.floor>new.max_floor)
)
THEN
    raise exception 'Существует номер с этажом > %',new.max_floor ;
    return NULL;
end if;
return NEW;
END;
$$;
CREATE TRIGGER trg_hotel_bef_upd
BEFORE UPDATE ON hotel
FOR EACH ROW
EXECUTE PROCEDURE tf_hotel_bef_upd());

```

Правило 3. При нарушении ограничения триггер должен формировать достаточно информативное сообщение об ошибке.

Предположим, что в базе данных аэропорта имеется таблица «Фактические полеты».

```

CREATE TABLE realrun
(id integer PRIMARY KEY,
 id_flight integer REFERENCES flight (id),
 id_plane integer REFERENCES plane (id),
 id_team integer REFERENCES team (id),
 begin_time timestamp with time zone NOT
NULL check (begin_time<= current_timestamp),
 end_time timestamp with time zone ,
... ..
check ( begin_time < end_time ));

```

Необходимо разработать триггер, который проверяет ограничение: самолет не может одновременно выполнять более чем один полет.

В терминах базы данных это ограничение имеет вид: среди строк таблицы **realrun** с одним и тем же значением **id_plane**

- может быть максимум одна строка с пустым временем приземления (**end_time is null**);

- не может быть строк с пересечением периодов с **begin_time** по **end_time**.

Неправильное решение

```
CREATE FUNCTION tf_realrun_bef_ins_upd ()RETURNS SETOF trigger
LANGUAGE plpgsql
```

```
AS $$
```

```
DECLARE
```

```
new_et timestamp;
```

```
BEGIN
```

```
new_et:= case when new.end_time is null
            then current_timestamp + interval'1 hour'
            else new.end_time
        end;
```

```
if (exists (select id
            from realrun t
            where t.id<>new.id
            and
            t.id_plane = new.id_plane
            and
            t.begin_time < new_et
            and
            case when t.end_time is null
            then current_timestamp
            else t.end_time
            end >new.begin_time
        )
) OR
```

```
(exists (select id
            from realrun t
            where t.id<>new.id
            and
            t.id_team = new.id_team
            and
            t.begin_time new_et
            and
            case when t.end_time is null
            then current_timestamp
            else t.end_time
            end >new.begin_time
```



```

    )
)
THEN
    raise exception 'ERROR: incorrect data';
    return NULL;
end if;
return NEW;

```

Правильное решение

```

CREATE FUNCTION tf_realrun_bef_ins_upd () RETURNS SETOF trigger
LANGUAGE plpgsql
AS $$
DECLARE
new_et timestamp;
BEGIN
new_et:= case when new.end_time is null
    then current_timestamp + interval'1 hour'
    else new.end_time
end;
if (exists (select id
    from realrun t
    where t.id<>new.id
        and
        t.id_plane = new.id_plane
        and
        t.begin_time < new_et
        and
        case when t.end_time is null
            then current_timestamp + interval'1 hour'
            else t.end_time
        end >new.begin_time
    )
)
THEN
    raise exception 'Самолет не может выполнять более одного полета
одновременно';
if (exists (select id
    from realrun t
    where t.id<>new.id
        and

```

```

        t.id_team = new.id_team
        and
        t.begin_time < new_et
    and
    case when t.end_time is null
        then current_timestamp + interval'1 hour'
        else t.end_time
    end > new.begin_time
)
)
THEN
    raise exception 'Экипаж не может выполнять более одного полета
одновременно';
    return NULL;
end if;
return NEW;
END;
$$

```

Правило 4. Триггеры не должны компенсировать недостатки проектирования базы данных.

Большое количество триггеров для поддержания согласованности данных свидетельствует о недостаточно хорошо спроектированной и/или ненормализованной структуре базы данных.

Например, на это указывают внешние ключи с опцией **on delete cascade** и одновременно использование запрещающего триггера.

Правило 5. Триггеры не должны быть бессмысленными.

Пример бессмысленного триггера из студенческого проекта.

Предполагается, что в базе данных имеются следующие таблицы:

```

CREATE TABLE tu
    ( id integer PRIMARY KEY,
      name_tu character(20) NOT
NULL, .....);
CREATE TABLE soldier
    ( id integer PRIMARY KEY,
      fio character(50) NOT NULL,
      db date NOT NULL, .....);
-- Военная часть
-- код военной части
-- название военной части
-- Военнослужащий
-- код военнослужащего
-- Ф.И.О.
-- дата рождения

```

```

CREATE TABLE post -- должность
  ( id integer PRIMARY KEY, -- код должности
    name_post character(20) NOT -- название должности
  NULL, -- дата, с которой должность
    cansel_date date, -- неактуальна
    .....);
CREATE TABLE job -- назначение на должность
  ( id integer PRIMARY KEY, -- код записи
    id_soldier integer NOT NULL REFERENCE -- код военнослужащего
  soldier (id),
    id_mu integer NOT NULL REFERENCE -- код военной части
  mu (id),
    id_post integer NOT NULL REFERENCE -- код должности
  post (id),
    date_post date NOT NULL, -- дата назначения
    date_cansel date .....); -- дата перевода/увольнения

```

Описание триггера: триггер при удалении записи из таблицы «Должность» ставит в столбец **cansel_date** текущую дату, тем самым помечая запись как неиспользуемую.

```

CREATE FUNCTION tf_post_aft_del() RETURNS SETOF trigger
  LANGUAGE plpgsql
  AS $$
BEGIN
  OLD.cansel_date:=current_date;
  insert into post values (OLD.id, OLD.name_post, OLD.cansel_date);
  return OLD;
END;
$$;

```

```

CREATE TRIGGER trg_post_aft_del
AFTER DELETE ON post
FOR EACH ROW
EXECUTE PROCEDURE tf_post_aft_del();

```

Рассмотрим, как будет работать этот триггер.

Возможны два варианта развития событий.

1. Из таблицы **post** удаляется запись, на которую есть хотя бы одна ссылка в таблице **job**. В этом случае сработает ограничение внешнего ключа, так как оно проверяется перед удалением записи. Ограничение не позволит удалить запись, а поскольку запись не будет удалена,

триггер не будет выполнен. Этот вариант наиболее вероятен, так как предполагается, что на основную массу строк таблицы **post** в таблице **job** будут ссылки.

2. Если из таблицы **post** удаляется запись, на которую нет ни одной ссылки в таблице **job**, триггер восстановит только что удаленную запись и поставит в столбец **cansel_date** текущую дату. Получается, что должность, на которую до настоящего момента никто ни разу не был назначен и которая с текущего момента не актуальна (фактически совершенно ненужная должность), однажды вставленная в таблицу, не может быть из нее удалена.

В итоге, это не просто бесполезный, а в какой-то мере даже вредный триггер.

В такой ситуации нужен не триггер, а процедура удаления должности, которая будет проверять, можно ли ее удалить. Если можно – будет удалять, если нельзя (когда есть ссылки) – будет пометать запись как более не используемую.

```
CREATE FUNCTION delete_post1(inid_post integer) RETURNS void
LANGUAGE plpgsql
AS $$
BEGIN
delete from post where id=inid_post;
exception
when integrity_constraint_violation then
update post set cansel_date=current_date
where id=inid_post and cansel_date is null;
when others then
raise exception 'error NUM:%, DETAILS:%', SQLSTATE, SQLERRM;
END;
$$;
```

Кроме того, рекомендуется для операций удаления запрашивать дополнительное подтверждение, чтобы минимизировать возможность ошибочного удаления данных.

Есть два основных требования к разрабатываемым процедурам и функциям.

Правило 1. Процедуры и функции должны инкапсулировать элементарные операции с данными (SQL-запросы) и обрабатывать данные в соответствии с бизнес-правилами.

Например, для информационной системы гостиничного комплекса могут быть написаны функции:

Неверно	Верно
функция получения значения отдельного атрибута номера (количества мест, уровня комфортности) по коду номера (первичному ключу)	функция, возвращающая параметризованный курсор-список свободных номеров вместе с их атрибутами
Процедура вставки записи в таблицу постояльцы	процедура поселения постояльца в гостиницу, которая выполняет: <ul style="list-style-type: none"> ▪ вставку или обновление записи таблицы постояльцы, ▪ вставку или обновление записи таблицы поселение-бронь, ▪ вставку записи в таблицу оплата

Правило 2. Процедуры и функции должны корректно обрабатывать ошибочные ситуации и формировать информативные сообщения об ошибках.

4. ЗАДАНИЯ ДЛЯ ВЫПОЛНЕНИЯ ЛАБОРАТОРНЫХ, КУРСОВЫХ И ИССЛЕДОВАТЕЛЬСКИХ РАБОТ ПО ТЕМЕ «ПРОЕКТИРОВАНИЕ ИНФОРМАЦИОННЫХ СИСТЕМ И БАЗ ДАННЫХ»

Для выбранного варианта задания необходимо выполнить следующий комплекс работ по проектированию информационной системы.

1. Проанализировать описание предметной области варианта задания, при необходимости самостоятельно уточнить и дополнить ее.
2. Проанализировать предметную область, в том числе:
 - определить состав подразделений (подсистем) информационной системы;
 - определить перечень функций и задач системы в целом и каждого подразделения (подсистемы) в отдельности;
 - подробно описать работу каждого подразделения (подсистемы), их взаимоотношения; описать отдельные сценарии работ;
 - построить диаграммы работ, диаграммы потоков данных и диаграммы на языке UML для всей информационной системы в целом и для входящих в нее подразделений (подсистем).

3. Выполнить инфологическое проектирование модели базы данных, в том числе:

- определить необходимый набор сущностей, отражающих предметную область и информационные потребности пользователей, необходимый набор атрибутов каждой сущности, идентифицирующие атрибуты;
- классифицировать сущности (стержневые, характеристические, ассоциативные, обозначающие);
- классифицировать атрибуты каждой сущности;
- выделить сущности вида подтип/супертип (где это необходимо);
- проанализировать связи между сущностями, удалить избыточные;
- определить множественность и условность связей;
- классифицировать связи по типу сущности (1:1, 1:M, M:N);
- формализовать связи;
- построить ER-диаграмму модели базы данных;
- описать модель базы данных на языке инфологического проектирования.

4. Выполнить логическое проектирование модели базы данных, в том числе:

- описать состав отношений базы данных и набора атрибутов каждого отношения;
- выделить первичные и внешние ключи отношений;
- нормализовать полученные отношения с приведением модели базы данных к третьей нормальной форме;
- определить необходимые декларативные ограничения целостности исходя из специфики предметной области;
- представить связи между внешними и первичными ключами в виде вертикальной диаграммы.

5. Записать выражения на языке SQL, выполняющие выборку данных из созданной базы данных для указанных в варианте задания запросов.

6. Разработать набор хранимых процедур и триггеров для обеспечения ограничений целостности и выполнения законченных фрагментов действий.

7. Разработать приложение с использованием одной из изученных технологий работы с базами данных.

4.1. Информационная система вуза

Студенты, организованные в группы, учатся на одном из факультетов, возглавляемом деканатом, в функции которого входит контроль за учебным процессом. В учебном процессе участвуют преподаватели кафедр, административно относящиеся к одному из факультетов. Преподаватели делятся на следующие категории: ассистенты, преподаватели, старшие преподаватели, доценты, профессора. Ассистенты и преподаватели могут обучаться в аспирантуре, старшие преподаватели и доценты могут возглавлять научные темы, профессора – научные направления. Преподаватели любой из категории могли защитить кандидатскую, а доценты и профессора – и докторскую диссертацию, при этом преподаватели могут занимать должности доцента и профессора, только если у них есть соответствующие ученые звания.

Учебный процесс регламентируется учебным планом, который определяет, какие учебные дисциплины на каких курсах и в каких семестрах читаются для студентов каждого года набора, с указанием количества часов на каждый вид занятий (лекция, семинар, лабораторная работа, консультация, курсовая работа, ИР и т. д.) и формы контроля (зачет, экзамен). Перед началом семестра деканаты раздают на кафедры учебные поручения, в которых указывается, какие кафедры (не обязательно относящиеся к данному факультету) должны вести в очередном семестре какие дисциплины и для каких групп. Руководствуясь поручениями, на кафедрах распределяют нагрузки, при этом по одной дисциплине в одной группе разные виды занятий может вести как один, так и несколько преподавателей. При этом учитывается их категория. Например, ассистент не может читать лекции, а профессор никогда не будет проводить лабораторные работы. Преподаватель может вести занятия по одной или нескольким дисциплинам для студентов как своего, так и чужого факультетов. Сведения о проведенных экзаменах и зачетах собираются деканатом.

В конце обучения студент выполняет дипломную работу, руководит которой преподаватель с кафедры, относящейся к тому же факультету, где обучается студент. При этом преподаватель может руководить несколькими студентами.

Виды запросов в информационной системе

1. Получить перечень и общее число студентов указанных групп либо указанного курса (курсов) факультета полностью, по полу, году рождения, возрасту, наличию детей, по признаку получения и размеру стипендии.

2. Получить список и общее число преподавателей указанных кафедр либо указанного факультета полностью либо указанных категорий (ассистенты, доценты, профессора и т. д.) по полу, году рождения, возрасту, наличию и количеству детей, размеру заработной платы, являющихся аспирантами, защитивших кандидатские, докторские диссертации в указанный период.

3. Получить перечень и общее число тем кандидатских и докторских диссертаций, защищавшихся сотрудниками указанной кафедры либо указанного факультета.

4. Получить перечень кафедр, проводящих занятия в указанной группе либо на указанном курсе указанного факультета в указанном семестре, либо за указанный период.

5. Получить список и общее число преподавателей, проводивших (проводящих) занятия по указанной дисциплине в указанной группе либо на указанном курсе указанного факультета.

6. Получить перечень и общее число преподавателей, проводивших (проводящих) лекционные, семинарские и другие виды занятий в указанной группе либо на указанном курсе указанного факультета в указанном семестре, либо за указанный период.

7. Получить список и общее число студентов указанных групп, сдавших зачет либо экзамен по указанной дисциплине с указанной оценкой.

8. Получить список и общее число студентов указанных групп или указанного курса указанного факультета, сдавших указанную сессию на отлично, без троек, без двоек.

9. Получить перечень преподавателей принимающих (принимавших) экзамены в указанных группах, по указанным дисциплинам, в указанном семестре.

10. Получить список студентов указанных групп, которым заданный преподаватель поставил некоторую оценку за экзамен по определенной дисциплине, в указанных семестрах, за некоторый период.

11. Получить список студентов и тем дипломных работ, выполняемых ими на указанной кафедре либо у указанного преподавателя.

12. Получить список руководителей дипломных работ с указанной кафедры либо факультета полностью и отдельно по некоторым категориям преподавателей.

13. Получить нагрузку преподавателей (название дисциплины, количество часов), ее объем по отдельным видам занятий и общую нагрузку в указанном семестре для конкретного преподавателя либо для преподавателей указанной кафедры.

4.2. Информационная система торговой компании

Торговая компания работает в торговых точках разных типов: супермаркеты, магазины, киоски и т. д., в штате которых трудятся продавцы. Торговая компания может иметь в собственности супермаркет и/или магазин, а может арендовать в них отдельные помещения (секции). Как супермаркеты, так и магазины могут иметь несколько залов (секций), в которых работает определенное число продавцов разных торговых компаний. Супермаркеты, магазины, киоски могут иметь такие характеристики, как размер торговой точки, платежи за аренду, коммунальные услуги, количество прилавков и т. д. Кроме того, в супермаркетах и магазинах учет проданных товаров ведется персонализированно с фиксацией имен и характеристик покупателя, чего в киосках сделать невозможно.

Заказы поставщику составляются на основе заявок, поступающих из торговых точек. На основе заявок менеджеры торговой компании выбирают поставщика, формируют заказы, в которых перечисляются наименования товаров и их количество (оно может отличаться от запроса из торговой точки). Если указанный товар ранее не поставлялся, то его наименование пополняет справочник номенклатуры товаров. Рынок поставщиков постоянно изучается, поэтому могут появляться новые поставщики и исчезать старые. При этом одни и те же товары торговая компания может получать от разных поставщиков и, естественно, по различным ценам.

Поступившие товары распределяются по торговым точкам и в любой момент можно получить такое распределение.

Продавцы торговых точек продают товары, учитывая все совершенные продажи, фиксируя номенклатуру и количество проданного товара, а продавцы супермаркетов и магазинов дополнительно фиксируют имена и характеристики покупателей, что позволяет вести учет покупателей и сделанных ими покупок. Торговые точки вправе менять цены на товары в зависимости от спроса и предложения, а также по согласованию передавать товары в другую торговую точку.

Виды запросов в информационной системе

1. Получить перечень и общее число поставщиков указанного вида товара либо некоторый товар в объеме не менее заданного за весь период сотрудничества либо за указанный период.

2. Получить перечень и общее число покупателей указанного вида товара за некоторый период либо сделавших покупку товара в объеме не менее заданного.

3. Получить номенклатуру и объем товаров в указанной торговой точке.

4. Получить сведения об объеме и ценах на указанный товар по всем торговым точкам, по торговым точкам заданного типа, по конкретной торговой точке.

5. Получить данные о выработке на одного продавца за указанный период по всем торговым точкам, по торговым точкам заданного типа.

6. Получить данные о выработке отдельно взятого продавца определенной торговой точки за указанный период.

7. Получить данные об объеме продаж указанного товара за некоторый период по всем торговым точкам, по торговым точкам заданного типа, по конкретной торговой точке.

8. Получить данные о заработной плате продавцов по всем торговым точкам, по торговым точкам заданного типа, по конкретной торговой точке.

9. Получить сведения о поставках определенного товара указанным поставщиком за все время поставок либо за некоторый период.

10. Получить данные об арендованных помещениях, о количестве и перечне сотрудников, работающих в помещениях указанного типа, в конкретном помещении, по всем арендованным помещениям.

11. Получить данные о рентабельности торговой точки: соотношение объема продаж к накладным расходам (суммарная заработная плата продавцов + платежи за аренду, коммунальные услуги) за указанный период.

12. Получить сведения о поставках товаров по указанному номеру заказа.

13. Получить сведения о покупателях указанного товара за обозначенный (или за весь) период по всем торговым точкам, по торговым точкам указанного типа, по данной торговой точке.

14. Получить сведения о наиболее активных покупателях по всем торговым точкам, по торговым точкам указанного типа, по данной торговой точке.

15. Получить данные о товарообороте торговой точки либо всех торговых точках определенной группы за указанный период.

4.3. Информационная система медицинских организаций города

Каждая больница города состоит из одного или нескольких корпусов, в каждом из которых размещается одно или несколько отделений, специализирующихся на лечении определенной группы болезней; каждое отделение имеет некоторое количество палат на определенное число коек. Поликлиники могут быть административно прикрепленными к больницам, а могут быть и не прикрепленными. В больницах и поликлиниках работают врачебный (хирурги, терапевты, невропатологи, окулисты, стоматологи, рентгенологи, гинекологи и пр.) и обслуживающий персонал (медсестры, санитары, уборщицы и пр.). Каждая категория врачей обладает уникальными характеристиками и по-разному участвует в связях: хирурги, стоматологи и гинекологи могут проводить операции (их параметры: число проведенных операций, число операций с летальным исходом); рентгенологам и стоматологам прибавляют коэффицент к зарплате за вредные условия труда; у рентгенологов и невропатологов более длинный отпуск. Врачи любого профиля могут иметь степень кандидата или доктора медицинских наук. Степень доктора медицинских наук дает право на присвоение звания профессора, а степень кандидата медицинских наук – на присвоение звания доцента. Разрешено совмещение, так что каждый врач может работать либо в больнице, либо в поликлинике, либо и в одной больнице и в одной поликлинике. Врачи со званием доцента или профессора могут консультировать в нескольких больницах или поликлиниках.

Лаборатории, выполняющие медицинские анализы, могут обслуживать различные больницы и поликлиники при наличии договора на обслуживание с соответствующим лечебным заведением. При этом каждая лаборатория имеет один или несколько профилей: биохимический, физиологический, химический.

Пациенты амбулаторно лечатся в одной из поликлиник и по направлению из нее могут стационарно лечиться в больнице, к которой относится поликлиника. Если специализация больницы, к которой приписана поликлиника, не позволяет провести требуемое лечение, то можно лечиться в любой другой. Больницы и поликлиники ведут персонализированный учет пациентов, полную историю их болезней, все назначения, операции и т. д. В больнице у пациента может быть только один лечащий врач, в поликлинике – несколько.

Виды запросов в информационной системе

1. Получить перечень и общее число врачей указанного профиля для конкретного медицинского учреждения (больницы либо поликлиники) либо всех медицинских учреждений города.

2. Получить перечень и общее число обслуживающего персонала указанной специальности для конкретного медицинского учреждения либо всех медицинских учреждений города.

3. Получить перечень и общее число врачей указанного профиля, сделавших число операций не менее заданного для конкретного медицинского учреждения, либо всех медицинских учреждений города.

4. Получить перечень и общее число врачей указанного профиля, стаж работы которых не менее заданного для конкретного медицинского учреждения либо всех медицинских учреждений города.

5. Получить перечень и общее число врачей указанного профиля со степенью кандидата или доктора медицинских наук, со званием доцента или профессора для конкретного медицинского учреждения либо больницы, либо поликлиники, либо всех медицинских учреждений города.

6. Получить перечень пациентов указанной больницы, отделения либо конкретной палаты выбранного отделения с указанием даты поступления, состояния, температуры, лечащего врача.

7. Получить перечень пациентов, прошедших стационарное лечение в указанной больнице либо у конкретного врача за некоторый промежуток времени.

8. Получить перечень пациентов, наблюдающихся у врача указанного профиля в конкретной поликлинике.

9. Получить общее число палат, коек указанной больницы в общем и по каждому отделению, а также число свободных коек по каждому отделению и число полностью свободных палат.

10. Получить общее число кабинетов указанной поликлиники, число посещений каждого кабинета за определенный период.

11. Получить данные о выработке (среднее число принятых пациентов в день) за указанный период для конкретного врача либо всех врачей поликлиники, либо для всех врачей названного профиля.

12. Получить данные о загрузке (число пациентов, у которых врач в настоящее время является лечащим) для указанного врача либо всех врачей больницы, либо для всех врачей названного профиля.

13. Получить перечень пациентов, перенесших операции в указанной больнице, либо поликлинике, либо у конкретного врача за некоторый промежуток времени.

14. Получить данные о выработке лаборатории (среднее число проведенных обследований в день) за указанный период для данного медицинского учреждения либо всех медицинских учреждений города.

4.4. Информационная система автопредприятия

Автопредприятие занимается организацией пассажирских и грузовых перевозок внутри города. В ведении предприятия находится автотранспорт различного назначения: автобусы, такси, маршрутные такси и прочий легковой транспорт; грузовой транспорт; транспорт вспомогательного характера, представленный различными марками. Каждая перечисленная категория транспорта обладает определенными особенностями: грузовой транспорт – грузоподъемностью, пассажирский – вместимостью и т. д. Со временем транспорт стареет и списывается (возможно, продается), однако предприятие пополняется и новыми машинами.

За каждым автомобилем может быть закреплен только один водитель. Обслуживающий персонал (техники, сварщики, слесари, сборщики и др.) занимается техническим обслуживанием автомобильной техники, при этом различные вышеперечисленные категории также могут иметь уникальные для данной категории атрибуты. Обслуживающий персонал и водители объединяются в бригады, которыми руководят бригадиры, бригадирами руководят мастера, мастерами – начальники участков и цехов. В ведении предприятия находятся объекты гаражного хозяйства (цеха, гаражи, боксы и пр.), где содержится и ремонтируется автомобильная техника.

Пассажирский автотранспорт (автобусы, маршрутные такси) перевозит пассажиров по определенным маршрутам, за каждым из которых закреплены отдельные единицы автотранспорта. Ведется учет числа перевозимых пассажиров, на основании которого транспорт перераспределяется с одного маршрута на другой. Учитываются также пробег, число ремонтов и затраты на ремонт по всему автотранспорту, объем грузоперевозок для грузового транспорта, интенсивность использования транспорта вспомогательного назначения. Учитывается интенсивность работы бригад по ремонту (число ремонтов, объем выполненных работ), число замененных и отремонтированных узлов и агрегатов

(двигателей, коробок передач, мостов, шасси и т. д.) по каждой автомашине и суммарно по участку, цеху, предприятию.

Виды запросов в информационной системе

1. Получить данные об автопарке предприятия.
2. Получить перечень и общее число водителей по предприятию, по указанной автомашине.
3. Получить распределение водителей по автомобилям.
4. Получить данные о распределении пассажирского автотранспорта по маршрутам.
5. Получить сведения о пробеге автотранспорта определенной категории или конкретной автомашины за указанный день, месяц, год.
6. Получить данные о числе ремонтов и их стоимости для автотранспорта определенной категории, отдельной марки автотранспорта или указанной автомашины за указанный период.
7. Получить данные о подчиненности персонала: рабочие – бригаиры – мастера – начальники участков и цехов.
8. Получить сведения о наличии гаражного хозяйства в целом и по каждой категории транспорта.
9. Получить сведения о грузоперевозках, выполненных указанной автомашиной за обозначенный период.
10. Получить данные о числе использованных для ремонта указанных узлов и агрегатов для транспорта определенной категории, отдельной марки автотранспорта или конкретной автомашины за указанный период.
11. Получить сведения о купленной и списанной автотехнике за указанный период.
12. Получить состав подчиненных указанного бригадира, мастера и т. д.
13. Получить данные о работах, выполненных указанным специалистом (сварщиком, слесарем и т. д.) за обозначенный период в целом и по конкретной автомашине.

4.5. Информационная система проектной организации

В проектной организации работают следующие категории сотрудников: конструкторы, инженеры, техники, лаборанты, прочий обслуживающий персонал. Каждая из категорий может иметь свойственные только ей атрибуты. Например, у конструкторов есть определенное число автор-

ских свидетельств, у техников – оборудование, которое они могут обслуживать, кроме того, инженер или конструктор могут руководить договором или проектом и т. д. Сотрудники разделены на отделы, руководимые начальником так, что каждый сотрудник числится только в одном отделе.

В рамках заключаемых договоров с заказчиками выполняются различные проекты, причем по одному договору может выполняться более одного проекта или один проект может выполняться для нескольких договоров. Суммарная стоимость договора определяется стоимостью всех выполненных для него проектных работ. Каждый договор и проект имеют руководителя и группу сотрудников, причем это могут быть сотрудники не только одного отдела. Проекты выполняются с использованием различного оборудования, часть которого приписана конкретным отделам, а часть – коллективная собственность проектной организации, при этом в процессе работы оборудование может передаваться из отдела в отдел. Для выполнения проекта оборудование предоставляется группе, работающей над проектом, но только оно не используется в другом проекте.

Для выполнения ряда проектов подрядная организация может привлекать субподрядные организации, передавая им объемы работ.

Ведется учет кадров, учет выполнения договоров и проектов, стоимостный учет всех выполненных работ.

Виды запросов в информационной системе

1. Получить данные о составе указанного отдела или всей организации полностью, по указанной категории сотрудников, по возрастному составу.

2. Получить перечень руководителей отделов.

3. Получить перечень договоров или проектов, выполняемых в данный момент или в период указанного интервала времени.

4. Получить информацию о том, какие проекты выполняются (выполнялись) в рамках указанного договора и какие договора поддерживаются указанными проектами.

5. Получить данные о стоимости выполненных в течение указанного периода времени договоров (проектов).

6. Получить данные о распределении оборудования на данный момент или на некоторую указанную дату.

7. Получить сведения об использовании оборудования в указанных проектах (договорах).

8. Получить сведения об участии указанного сотрудника или категории сотрудников в проектах (договорах) за определенный период времени.

9. Получить перечень и стоимость работ, выполненных субподрядными организациями.

10. Получить данные о численности и составе сотрудников в целом и по отдельным категориям, участвующих в указанном проекте.

11. Получить данные об эффективности использования оборудования (объемы проектных работ, выполненных с использованием того или иного оборудования).

12. Получить сведения об эффективности договоров (стоимость договоров, соотнесенная с затраченным временем, или стоимость с учетом привлеченных людских ресурсов).

13. Получить данные о численности и составе сотрудников в целом и по отдельным категориям, участвующих в проектах за указанный период времени.

14. Получить сведения об эффективности проектов (стоимость договоров, соотнесенная с затраченным временем, или стоимость с учетом привлеченных людских ресурсов).

4.6. Информационная система авиастроительного предприятия

Структурно предприятие разбито на цеха, которые подразделяются на участки. Предприятие выпускает самолеты (гражданские, транспортные, военные), планеры, вертолеты, дельтопланы, ракеты (артиллерийские, авиационные, военно-морские), прочие изделия. Каждая категория изделий обладает специфическими атрибутами. Например, для самолетов это число двигателей, для ракеты – мощность заряда и т. д. Может собираться несколько видов каждой категории изделий. Каждой категории инженерно-технического персонала (инженеры, технологи, техники) и рабочих (сборщики, токари, слесари, сварщики и пр.) также свойственны характерные только для этой группы атрибуты. Рабочие объединяются в бригады, которыми руководят бригадиры. Бригадиры выбираются из числа рабочих, мастера, начальники участков и цехов назначаются из числа инженерно-технического персонала.

Каждое изделие собирается в своем цехе (в цехе может собираться несколько видов изделий), в процессе изготовления перемещаясь с одного участка на другой. Все работы по сборке конкретного изделия на определенном участке выполняет одна бригада рабочих, при этом на

участке может работать несколько бригад. Возглавляет работу на участке начальник участка, в подчинении которого находится несколько мастеров. Различные изделия могут проходить одни и те же циклы работ на одних и тех же участках цеха.

Собранное изделие проходит серию испытаний в испытательных лабораториях (полигонах). Испытательные лаборатории могут обслуживать несколько цехов, в свою очередь цеха пользуются, возможно, несколькими испытательными лабораториями. Испытания проводятся испытателями на оборудовании испытательной лаборатории, при этом при испытании конкретного изделия в лаборатории могут быть задействованы различные виды оборудования.

Ведется учет движения кадров и учет выпускаемой продукции.

Виды запросов в информационной системе

1. Получить перечень видов изделий отдельной категории и в целом, собираемых указанным цехом, всем предприятием.
2. Получить число и перечень изделий отдельной категории и в целом, собранных указанным цехом, участком, всем предприятием за определенный отрезок времени.
3. Получить данные о кадровом составе цеха, всего предприятия и по указанным категориям инженерно-технического персонала и рабочих.
4. Получить число и перечень участков указанного цеха, всего предприятия и их начальников.
5. Получить перечень работ, которые проходит указанное изделие.
6. Получить состав бригад указанного участка, цеха.
7. Получить список мастеров указанного участка, цеха.
8. Получить перечень изделий отдельной категории и в целом, собираемых в настоящий момент указанным участком, цехом, предприятием.
9. Получить состав бригад, участвующих в сборке указанного изделия.
10. Получить перечень испытательных лабораторий, участвующих в испытаниях определенного изделия.
11. Получить перечень изделий отдельной категории и в целом, проходивших испытание в указанной лаборатории за определенный период.
12. Получить список испытателей, участвующих в испытаниях указанного изделия, изделий отдельной категории и в целом в некоторой лаборатории за определенный период.

13. Получить состав оборудования, использовавшегося при испытании указанного изделия, изделий отдельной категории и в целом в некоторой лаборатории за определенный период.

14. Получить число и перечень изделий отдельной категории и в целом, собираемых указанным цехом, участком, предприятием в целом в настоящее время.

4.7. Информационная система военного округа

Военные части округа расквартированы по различным местам дислокации, причем в одном месте могут располагаться несколько частей. Каждая воинская часть состоит из рот, роты – из взводов, взводы – из отделений. Воинские части объединяются в дивизии, корпуса или бригады, а те – в армии. Военный округ представлен офицерским составом (генералы, полковники, подполковники, майоры, капитаны, лейтенанты) и рядовым и сержантским составом (старшины, сержанты, прапорщики, ефрейторы, рядовые). Каждая из перечисленных категорий военнослужащих может иметь уникальные характеристики для генералов: дата окончания академии, дата присвоения генеральского звания и т. д. Каждое из подразделений имеет командира, причем военнослужащие офицерского состава могут командовать любым из вышеперечисленных подразделений, а военнослужащие рядового и сержантского состава – только взводом и отделением. У всех военнослужащих есть одна или несколько воинских специальностей.

Каждой воинской части придана боевая и транспортная техника (БМП, тягачи, автотранспорт и пр.) и вооружение (карабины, автоматическое оружие, артиллерия, ракетное вооружение и т. д.). Каждая из перечисленных категорий боевой техники и вооружения также имеет специфические атрибуты; в каждой категории может быть несколько видов техники и вооружения. Инфраструктура военной части представлена сооружениями (сооружение № 1, сооружение № 2 и т. д.), некоторые из них предназначены для дислокации подразделений.

Виды запросов в информационной системе

1. Получить перечень всех частей военного округа, указанной армии, дивизии, корпуса и их командиров.

2. Получить данные по офицерскому составу в целом и по офицерскому составу указанного звания всех частей военного округа, отдельной армии, дивизии, корпуса, военной части.

3. Получить данные по рядовому и сержантскому составу в целом и с учетом указанного звания для всех частей военного округа, отдельной армии, дивизии, корпуса, военной части.

4. Получить цепочку подчиненности для указанного военнослужащего.

5. Получить перечень мест дислокации всех частей военного округа, отдельной армии, дивизии, корпуса, военной части.

6. Получить данные о наличии боевой техники в целом и с учетом указанной категории или вида во всех частях военного округа, в отдельной армии, дивизии, корпусе, военной части.

7. Получить перечень сооружений указанной военной части, перечень сооружений, где дислоцировано более одного подразделения, где не дислоцировано ни одного подразделения.

8. Получить перечень военных частей, в которых число единиц указанного вида боевой техники больше пяти (нет указанной боевой техники).

9. Получить данные о наличии вооружения в целом и с учетом указанной категории или вида во всех частях военного округа, в отдельной армии, дивизии, корпусе, военной части.

10. Получить перечень военных специальностей, по которым в округе, в отдельной армии, дивизии, корпусе, военной части более пяти специалистов (нет специалистов).

11. Получить перечень военнослужащих указанной специальности в округе, в отдельной армии, дивизии, корпусе, военной части, в указанном подразделении некоторой военной части.

12. Получить перечень военных частей, в которых число единиц указанного вида вооружения больше десяти (нет указанного вооружения).

13. Получить данные об армии, дивизии, корпусе, в которые входит больше всего (меньше всего) военных частей.

4.8. Информационная система строительной организации

Строительная организация занимается строительством различного рода объектов: жилых домов, больниц, школ, мостов, дорог и т. д. по договорам с заказчиками (городская администрация, ведомства, частные фирмы и т. д.). Каждая из перечисленных категорий объектов имеет характеристики, свойственные только этой или нескольким категориям: например, к характеристикам жилых домов относятся этажность, тип

строительного материала, число квартир; для мостов – это тип пролетного строения, ширина, количество полос для движения.

Структурно строительная организация состоит из строительных управлений, каждое строительное управление ведет работы на одном или нескольких участках, возглавляемых начальниками участков, которым подчиняется группа прорабов, мастеров и техников. Каждая категория инженерно-технического персонала (инженеры, технологи, техники) и рабочих (каменщики, бетонщики, отделочники, сварщики, электрики, шоферы, слесари и пр.) обладает уникальными атрибутами. Рабочие объединяются в бригады, которыми руководят бригадиры; бригадиры выбираются из числа рабочих; мастера, прорабы, начальники участков и управлений назначаются из числа инженерно-технического персонала.

На каждом участке возводится один или несколько объектов, на каждом объекте работает одна или несколько бригад. Закончив работу, бригада переходит к другому объекту на этом или другом участке. Строительному управлению придается строительная техника (подъемные краны, экскаваторы, бульдозеры и т. д.), которая распределяется по объектам.

Технология строительства того или иного объекта предполагает выполнение определенного набора видов работ, необходимых для сооружения данного типа объекта. Например, для жилого дома – возведение фундамента, кирпичные работы, прокладка водоснабжения и т. д. Каждый вид работ на объекте выполняется одной бригадой. Для организации работ на объекте составляются графики работ, указывающие, в каком порядке и в какие сроки выполняются те или иные работы, а также смета, определяющая, какие строительные материалы и в каких количествах необходимы для сооружения объекта. По результатам выполнения работ составляется отчет с указанием сроков выполнения работ и фактических расходов материалов.

Виды запросов в информационной системе

1. Получить перечень строительных управлений и/или участков и их руководителей.
2. Получить список специалистов инженерно-технического состава обозначенного участка или строительного управления с указанием их должностей.
3. Получить перечень объектов, возводимых указанным строительным управлением и/или участком, и графики их возведения.

4. Получить состав бригад, работавших (работающих) на строительстве указанного объекта.
5. Получить перечень строительной техники, приданной указанному строительному управлению.
6. Получить перечень строительной техники, выделенной на указанный объект либо работавшей там в течение указанного периода времени.
7. Получить график и смету на строительство указанного объекта.
8. Получить отчет о сооружении указанного объекта.
9. Получить перечень объектов, возводимых в некотором строительном управлении или в целом по организации, на которых в обозначенный период времени выполнялся указанный вид строительных работ.
10. Получить перечень видов строительных работ, по которым имело место превышение сроков выполнения на указанном участке, в строительном управлении или в целом по организации.
11. Получить перечень строительных материалов, по которым имело место превышение по смете на указанном участке, строительном управлении или в целом по организации.
12. Получить перечень видов строительных работ, выполненных указанной бригадой в течение обозначенного периода времени с указанием объектов, где эти работы выполнялись.
13. Получить перечень бригад, выполнивших указанный вид строительных работ в течение обозначенного периода времени с указанием объектов, где эти работы выполнялись.

4.9. Информационная система библиотечного фонда города

Библиотечный фонд города составляют библиотеки, расположенные на его территории. В каждой библиотеке есть абонементы и читальные залы. Библиотеками пользуются различные категории читателей: студенты, научные работники, преподаватели, школьники, рабочие, пенсионеры и другие жители города. Каждая категория читателей может обладать непересекающимися характеристиками-атрибутами: для студентов – название учебного заведения, факультет, курс, номер группы, для научного работника – название организации, научная тема и т. д.

Каждый читатель, зарегистрированный в одной из библиотек, получает доступ ко всему библиотечному фонду города.

Библиотечный фонд размещен в залах-хранилищах различных библиотек на определенных местах хранения (номер зала, стеллажа, полки) и идентифицируется номенклатурными номерами. При этом существуют различные правила относительно тех или иных изданий: какие-то можно читать только в читальных залах; для тех, которые выдаются на абонемент, может быть установлен различный срок возврата и т. д. Библиотечный фонд может пополняться новыми книгами и списывать старые.

Произведения авторов, составляющие библиотечный фонд, также можно разделить на различные категории, характеризующиеся собственным набором атрибутов: учебники, повести, романы, статьи, стихи, диссертации, рефераты, тезисы докладов и т. д.

Сотрудники библиотеки, работающие в залах различных библиотек, ведут учет читателей, а также учет размещения и выдачи литературы

Виды запросов в информационной системе

1. Получить список читателей с заданными характеристиками: студентов указанного учебного заведения, факультета, научных работников по определенной тематике и т. д.

2. Получить перечень читателей, у которых находится указанное произведение.

3. Получить список читателей, у которых находится указанное издание (книга, журнал и т. д).

4. Получить перечень читателей, которые в течение указанного промежутка времени получали издание с некоторым произведением, и название этого издания.

5. Получить список изданий, которые в течение некоторого времени получал указанный читатель из фонда библиотеки, где он зарегистрирован.

6. Получить перечень изданий, которыми в течение некоторого времени пользовался указанный читатель из фонда библиотеки, где он не зарегистрирован.

7. Получить список литературы, которая в настоящий момент выдана с определенной полки некоторой библиотеки.

8. Получить список читателей, которые в течение обозначенного периода обслуживались указанным библиотекарем.

9. Получить данные о выработке библиотекарей (число обслуженных читателей в указанный период времени).
10. Получить список читателей с истекшей датой возврата выданных изданий.
11. Получить перечень указанных изданий, которые поступили (были списаны) в течение некоторого периода.
12. Получить список библиотекарей, работающих в указанном читальном зале некоторой библиотеки.
13. Получить список читателей, не посещавших библиотеку в течение указанного времени.
14. Получить список номенклатурных номеров и названий из библиотечного фонда, в которых содержится указанное произведение.
15. Выдать список номенклатурных номеров и названий из библиотечного фонда, в которых содержатся произведения указанного автора.
16. Получить список самых популярных произведений.

4.10. Информационная система спортивных организаций города

Спортивная инфраструктура города представлена спортивными сооружениями различного типа: спортивные залы, манежи, стадионы, корты и т. д. Каждая из категорий спортивных сооружений обладает уникальными атрибутами, стадион характеризуется вместимостью, корт – типом покрытия и т. д.

Спортсмены под руководством тренеров занимаются различными видами спорта, при этом один и тот же спортсмен может заниматься несколькими видами спорта и по одному виду спорта может тренироваться сразу у нескольких тренеров. Все спортсмены объединяются (по виду спорта) в спортивные клубы, при этом каждый из них может выступать только за один клуб.

Организаторы соревнований проводят состязания по отдельным видам спорта на спортивных сооружениях города. По результатам участия спортсменов награждают.

Виды запросов в информационной системе

1. Получить перечень спортивных сооружений указанного типа в целом или удовлетворяющих заданным характеристикам (например, стадионы, вмещающие не менее указанного числа зрителей).

2. Получить список всех спортсменов, занимающихся указанным видом спорта либо спортсменов не ниже определенного разряда.
3. Получить список всех спортсменов, тренирующихся у определенного тренера либо спортсменов не ниже определенного разряда.
4. Получить список спортсменов, занимающихся более чем одним видом спорта (с указанием этих видов спорта).
5. Получить список тренеров указанного спортсмена.
6. Получить перечень всех соревнований, проведенных в течение заданного периода времени либо соревнований проведенных указанным организатором.
7. Получить список призеров указанного соревнования.
8. Получить перечень всех соревнований, проведенных в указанном спортивном сооружении либо по определенному виду спорта.
9. Получить перечень спортивных клубов и число спортсменов этих клубов, участвовавших в спортивных соревнованиях в течение заданного интервала времени.
10. Получить список тренеров по определенному виду спорта.
11. Получить список спортсменов, не участвовавших ни в каких соревнованиях в течение определенного периода времени.
12. Получить список организаторов соревнований и число проведенных ими соревнований в течение определенного периода времени.
13. Получить перечень спортивных сооружений и даты проведения на них соревнований в течение определенного периода времени.

4.11. Информационная система автомобилестроительного предприятия

Структурно предприятие состоит из цехов, которые в свою очередь подразделяются на участки. Выпускаемые изделия предприятия: грузовые, легковые автомобили, автобусы, сельскохозяйственные и дорожно-строительные машины, мотоциклы и прочие изделия. Каждая категория изделий обладает уникальными атрибутами. Например, для автобусов это вместимость, для сельскохозяйственных и дорожно-строительных машин – производительность и т. д. По каждой категории изделий может собираться несколько видов изделий. Каждая категория инженерно-технического персонала (инженеры, технологи, техники) и рабочих (сборщики, токари, слесари, сварщики и пр.) также обладает уникальными атрибутами. Рабочие объединяются в бригады,

которыми руководят бригадиры. Бригадиры выбираются из числа рабочих; мастера, начальники участков и цехов назначаются из числа инженерно-технического персонала.

Каждое изделие собирается в своем цехе (в цехе может собираться несколько видов изделий), в процессе изготовления перемещаясь с одного участка на другой. Все работы по сборке конкретного изделия на определенном участке выполняет одна бригада рабочих, при этом на участке может работать несколько бригад. Возглавляет работу на участке начальник участка, в подчинении которого находится несколько мастеров. Различные изделия могут проходить одни и те же циклы работ на одних и тех же участках цеха.

Собранное изделие проходит серию испытаний в испытательных лабораториях. Испытательные лаборатории могут обслуживать несколько цехов, в свою очередь цеха могут пользоваться несколькими лабораториями. Испытания проводятся специалистами на оборудовании испытательной лаборатории, при этом при испытании конкретного изделия в лаборатории могут быть задействованы различные виды оборудования.

Ведется учет движения кадров и учет выпускаемой продукции.

Виды запросов в информационной системе

1. Получить перечень видов изделий отдельной категории и в целом, собираемых указанным цехом, всем предприятием.
2. Получить число и перечень изделий отдельной категории и в целом, собранных указанным цехом, участком, всем предприятием за определенный отрезок времени.
3. Получить данные о кадровом составе цеха, всего предприятия и по указанным категориям инженерно-технического персонала и рабочих.
4. Получить число и перечень участков указанного цеха, всего предприятия и их начальников.
5. Получить перечень работ, которые проходит указанное изделие.
6. Получить состав бригад указанного участка, цеха.
7. Получить перечень мастеров указанного участка, цеха.
8. Получить перечень изделий отдельной категории и в целом, собираемых в настоящий момент указанным участком, цехом, предприятием.
9. Получить состав бригад, участвующих в сборке указанного изделия.

10. Получить перечень испытательных лабораторий, участвующих в испытаниях определенного изделия.

11. Получить перечень изделий отдельной категории и в целом, проходивших испытание в указанной лаборатории за определенный период.

12. Получить перечень испытателей, участвующих в испытаниях указанного изделия, изделий отдельной категории и в целом в указанной лаборатории за определенный период.

13. Получить состав оборудования, использовавшегося при испытании указанного изделия, изделий отдельной категории и в целом в указанной лаборатории за определенный период.

14. Получить число и перечень изделий отдельной категории и в целом, собираемых указанным цехом, участком, предприятием в настоящее время.

4.12. Информационная система гостиничного комплекса

Гостиничный комплекс состоит из нескольких зданий-гостиниц (корпусов). Каждый корпус имеет ряд характеристик, таких, как класс отеля (от двух до пяти звезд), количество этажей в здании, общее количество комнат, комнат на этаже, вместимость номеров (одно-, двух-, трехместные и т. д.), наличие служб быта: ежедневная уборка номера, прачечная, химчистка, питание (рестораны, бары) и развлечения (бассейн, сауна, бильярд и пр.). От типа корпуса и вместимости номера зависит его стоимость. Химчистка, стирка, дополнительное питание, все развлечения предоставляются за отдельную плату.

С крупными организациями (туристические фирмы, организации, занимающиеся проведением международных симпозиумов, конгрессов, семинаров, карнавалов и т. д.) заключаются договоры, позволяющие этим организациям бронировать номера с большими скидками на определенное время вперед не для одного человека, а для группы людей. Каждая из перечисленных групп организаций обладает уникальными характеристиками. Желательно группы людей от одной организации не расселять по разным этажам. В брони указывается класс отеля, этаж, количество комнат и общее количество людей. Броня может быть отменена за неделю до заселения. Спрос на частичную услугу растет, в результате чего заключаются договора с новыми фирмами.

Гостиница исследует мнения жильцов о ценах и сервисе. Жалобы фиксируются и также исследуются. Изучается статистика популярности номеров. Ведется учет долгов постояльца гостинице за все дополнительные услуги.

Новые жильцы пополняют перечень клиентов гостиницы. Ведется учет свободных номеров, дополнительных затрат постояльцев гостиницы и учет расходов и доходов гостиничного комплекса.

Виды запросов в информационной системе

1. Получить перечень и общее число фирм, забронировавших места в объеме, не менее указанного, за весь период сотрудничества либо за некоторый период.

2. Получить перечень и общее число постояльцев, заселявшихся в номера с указанными характеристиками за некоторый период.

3. Получить количество свободных номеров на данный момент.

4. Получить сведения о количестве свободных номеров с указанными характеристиками.

5. Получить сведения о конкретном свободном номере: в течение какого времени он будет пустовать и о его характеристиках.

6. Получить список занятых сейчас номеров, которые освобождаются к указанному сроку.

7. Получить данные об объеме бронирования номеров данной фирмой за указанный период и каким номерам отдавалось предпочтение.

8. Получить список недовольных клиентов и их жалоб.

9. Получить данные о рентабельности номеров с определенными характеристиками (соотношение объема продаж номеров к накладным расходам за указанный период).

10. Получить сведения о постояльце из заданного номера: виды дополнительных услуг, которыми он пользовался, его счет за них и поступавшие от клиента жалобы.

11. Получить сведения о фирмах, с которыми заключены договора о бронировании на указанный период.

12. Получить сведения о наиболее часто посещающих гостиницу постояльцах по всем корпусам гостиниц, по определенному зданию.

13. Получить сведения о новых клиентах за указанный период.

14. Получить сведения о конкретном человеке: сколько раз он посещал гостиницу, в каких номерах и в какой период останавливался, какие счета оплачивал.

15. Получить сведения о том, кем был занят конкретный номер в определенный период.

16. Получить процентное отношение всех номеров к номерам, бронируемым партнерами.

4.13. Информационная система магазина автозапчастей

Магазин розничной торговли заказывает запчасти в различных странах. Ведется статистика продаж, отражающая спрос на те или иные детали, и соответственно потребность магазина в них (сколько единиц, на какую сумму, какого товара продано за последнее время); на ее основе составляются заказы на требуемые товары. Выбирают поставщика на каждый конкретный заказ менеджеры магазина. В заказах перечисляются наименование товара и его количество. Если указанное наименование товара ранее не поставлялось, оно пополняет справочник номенклатуры товаров.

Поставщики бывают различных категорий: фирмы, непосредственно производящие детали, дилеры, небольшие производства, мелкие поставщики и магазины. В результате поставщики различных категорий имеют различающийся набор атрибутов. Фирмы и дилеры – это самые надежные партнеры, они могут предложить полный пакет документов, скидки, а главное – гарантию, чего не может сделать небольшое производство или мелкий магазин. У фирм и дилеров закупается большой объем продукции. Небольшое производство – это низкие цены, но никакой гарантии качества. В мелких лавках можно выгодно купить небольшое количество простых деталей, причем каждую из них можно проверить на наличие брака. Фирмы и дилеры поставляют детали на основе договоров, чего не делается для небольшого производства и мелкого магазина. В ходе маркетинговых работ изучается рынок поставщиков, в результате чего могут появляться новые поставщики и исчезать старые.

Когда ожидаются новые поставки, магазин собирает заявки от покупателей на свои товары. Груз приходит, производится его таможенное оформление, оплата пошлин, после чего он доставляется на склад в магазин. В первую очередь удовлетворяются заявки покупателей, а оставшийся товар продается в розницу.

В любой момент можно получить любую информацию о деталях, находящихся на складе, либо о поставляемых деталях. Детали хранятся на складе в определенных ячейках. Все ячейки пронумерованы. Касса принимает деньги от покупателей за товар, а также возвращает деньги за брак. Брак, если это возможно, возвращается поставщику, который заменяет бракованную деталь. Информация о браке (поставщик, фирма-производитель, деталь) фиксируется.

Виды запросов в информационной системе

1. Получить перечень и общее число поставщиков определенной категории, продающих указанный вид товара, либо продавших указанный товар в объеме не менее заданного за определенный период.

2. Получить сведения о конкретном виде деталей: какими поставщиками продается, их расценки, время поставки.

3. Получить перечень и общее число покупателей, купивших указанный вид товара за некоторый период, либо сделавших покупку товара в объеме не менее указанного.

4. Получить перечень, объем и номер ячейки для всех деталей, хранящихся на складе.

5. Вывести в порядке возрастания десять самых продаваемых деталей и десять поставщиков с наиболее низкими ценами на указанные детали.

6. Получить среднее число продаж за месяц по любому виду деталей.

7. Получить долю товара конкретного поставщика в процентах, деньгах от всего оборота магазина; прибыль магазина за указанный период.

8. Получить перечень и общее количество непроданного (залежалого) товара на складе за определенный период и его объем от общего товара в процентах.

9. Получить перечень и общее количество бракованного товара, пришедшего за определенный период и список поставщиков, продавших товар.

10. Получить перечень, общее количество и стоимость товара, реализованного за конкретный день.

11. Получить кассовый отчет за определенный период.

12. Получить скорость оборота денежных средств, вложенных в товар (как быстро товар продается).

13. Подсчитать количество пустых ячеек на складе и сколько он сможет вместить товара.

14. Получить перечень и общее количество заявок от покупателей на ожидаемый товар, подсчитать, на какую сумму даны заявки.

4.14. Информационная система представительства туристической фирмы в зарубежной стране

Туристическая фирма в России формирует группу туристов и данные на каждого туриста (Ф.И.О., паспортные данные, пол, возраст, дети, гостиница для проживания) отправляют в представительство. Представительство на основе этих данных заполняет на каждого пакет документов для получения визы, в отделе эмиграции получает визы, готовит списки расселения по разным гостиницам и бронирует номера в этих гостиницах.

Представительство занимается приемом туристов в аэропорту, решает проблемы, связанные с визами и таможней, расселяет группу по гостиницам. Представительство предлагает расписание экскурсий и производит запись на определенные экскурсии. Составляется список тех, кто и на какие экскурсии едет, и затем передается в агентство организации экскурсий.

В обязанности представительства входит также следующее.

- Хранение и отправка груза туристов. На складе на каждого туриста заводится весовая ведомость, проводятся маркировка, взвешивание, упаковка груза. В весовой ведомости указываются количество мест, вес, стоимость упаковки, страховки и итоговая сумма.

- Предоставление полного финансового отчета в головную фирму. Все статьи расхода и дохода – гостиница, перевозки, экскурсии, непредвиденные расходы, расчеты в аэропорту (загрузка самолета, разгрузка, взлет–посадка, диспетчерские услуги, хранение груза) переносятся в финансовый отчет.

Туристическая группа делится на туристов, которые едут отдохнуть (они интересуются экскурсиями и не интересуются складом), на туристов, которые едут за грузом (они интересуются складом и не интересуются экскурсиями) и их детей. Дети не могут получить визу, сами переселиться и ходить без сопровождения родителей. Каждая категория туристов имеет специфические характеристики.

Виды запросов в информационной системе

1. Сформировать список туристов для таможни в целом и по указанной категории.
2. Сформировать списки на расселение по указанным гостиницам в целом и указанной категории.
3. Получить количество туристов, побывавших в стране за указанный период в целом и по определенной категории.
4. Получить сведения о конкретном туристе: сколько раз был в стране, даты прилета–отлета, в каких гостиницах останавливался, какие экскурсии и в каких агентствах заказывал, какой груз сдавал.
5. Получить список гостиниц, в которых туристическая фирма расселяет туристов, с указанием количества занимаемых номеров и проживавших в них человек за определенный период.
6. Получить общее количество туристов, заказавших экскурсии за определенный период.
7. Выбрать самые популярные экскурсии и самые качественные экскурсионные агентства.
8. Получить данные об указанном рейсе самолета на определенную дату: количество мест багажа, вес груза
9. Получить статистику о грузообороте склада: количество мест и вес груза, сданного за определенный период, количество самолетов, вывозивших этот груз, сколько из них грузовых, а сколько грузопассажирских.
10. Получить полный финансовый отчет по указанной группе в целом и для определенной категории туристов.
11. Получить данные о расходах и доходах за определенный период: обслуживание самолета, гостиница, экскурсии, визы, расходы представительства и т. п.
12. Получить статистику по видам отправляемого груза и удельную долю каждого вида в общем грузопотоке.
13. Вычислить рентабельность представительства (соотношение доходов и расходов).
14. Определить процентное отношение отдыхающих туристов к туристам шоп-туров в целом и за указанный период (например, в зависимости от времени года).

4.15. Информационная система аптеки

Аптека продает готовые лекарства (таблетки, мази, настойки и пр.) и лекарства изготавливаемые по рецепту (микстуры, мази, растворы, настойки, порошки).

Различие в типах лекарств отражается в различном наборе их атрибутов. Микстуры и порошки изготавливаются только для внутреннего применения; растворы – для наружного, внутреннего применения и для смешивания с другими лекарствами; мази – только для наружного применения. Лекарства различаются также по способу и по времени приготовления. Порошки и мази изготавливаются смешиванием различных компонентов. При создании растворов и микстур ингредиенты не только смешивают, но и фильтруют, что увеличивает время изготовления лекарства.

В аптеке существует справочник технологий приготовления различных лекарств. В нем указываются: идентификационный номер технологии, название лекарства и сам способ приготовления. На складе на все медикаменты устанавливается критическая норма. Когда какого-либо вещества на складе становится меньше критической нормы, то составляются заявки на эти вещества, и их в срочном порядке привозят с оптовых складов медикаментов.

Чтобы аптека изготовила лекарство, больной должен принести рецепт от лечащего врача. В рецепте должны быть указаны Ф.И.О. пациента, его возраст и диагноз, а также количество лекарства и способ применения. Рецепт должен быть заверен подписью и печатью врача.

Больной отдает рецепт фармацевту, тот принимает заказ и смотрит, есть ли компоненты заказываемого лекарства. Если не все компоненты имеются в наличии, то аптека их дозаказывает и записывает Ф.И.О., телефон и адрес необслуженного покупателя, чтобы сообщить ему, когда доставят нужные компоненты. Данные этого покупателя попадают в справочник заказов, т. е. тех заказов, которые находятся в процессе приготовления, с пометкой, что пока есть не все компоненты. Если в аптеке есть все компоненты, то они резервируются для лекарства. Покупатель выплачивает цену лекарства, ему возвращается рецепт с пометкой о времени изготовления. В назначенное время больной приходит и по тому же рецепту получает готовое лекарство. Такой больной пополняет список отданных заказов.

Ведется статистика по объемам используемых медикаментов. Через определенный промежуток времени производится инвентаризация склада. Это делается для того, чтобы определить, есть ли лекарства с критической нормой, или истек срок хранения, или обнаружился недостаток товара.

Виды запросов в информационной системе

1. Получить сведения о покупателях, которые не пришли забрать свой заказ в назначенное им время, и общее их число.

2. Получить перечень и общее число покупателей, которые ждут прибытия на склад нужных им медикаментов в целом и по указанной категории медикаментов.

3. Получить перечень десяти наиболее часто используемых медикаментов в целом и указанной категории медикаментов.

4. Получить объем использования указанных веществ за определенный период.

5. Получить перечень и общее число покупателей, заказывавших определенное лекарство или определенные типы лекарств за данный период.

6. Получить перечень и типы лекарств, достигших критической нормы или полностью закончившихся.

7. Получить перечень всех лекарств с минимальным запасом на складе и по указанной категории медикаментов.

8. Получить полный перечень и общее число заказов находящихся в производстве.

9. Получить полный перечень и общее число препаратов требующихся для заказов, находящихся в производстве.

10. Получить все технологии приготовления лекарств указанных типов, конкретных лекарств, а также лекарств, находящихся в справочнике заказов в производстве.

11. Получить сведения о ценах на указанное лекарство в готовом виде, об объеме и ценах на все компоненты, требующиеся для этого лекарства.

12. Получить сведения о наиболее часто делающих заказы клиентах на лекарства определенного типа, на конкретные лекарства.

13. Получить сведения о конкретном лекарстве (его тип, способ приготовления, названия всех компонентов, цена, количество на складе).

4.16. Информационная система библиотеки вуза

Библиотека включает в себя абонементы, читальные залы и справочную систему каталогов и картотек.

Читателями библиотеки вуза могут быть: студенты всех форм обучения, профессорско-преподавательский состав университета, аспиранты, ассистенты и другие сотрудники подразделений вуза, слушатели подготовительного отделения, факультета повышения квалификации (ФПК), стажеры, абитуриенты. Различные категории читателей обладают уникальными характеристиками: для студентов – это название факультета, номер группы, для преподавателя – название кафедры, разовые читатели – могут пользоваться только читальными залами.

Читатели библиотеки могут получать книги и другие издания во всех пунктах выдачи библиотеки (абонементных и читальных залах), а также получать необходимые издания по межбиблиотечному абонементу, предварительно сделав заказ. Читатели, приходящие на пункт выдачи, обязаны иметь при себе читательский билет с отметками о записи и перерегистрации текущего года на данном пункте выдачи. При выбытии из вуза (отчисление, окончание обучения, увольнение) читатели обязаны вернуть числящиеся за ними издания и сдать читательские билеты.

За нарушение правил пользования библиотекой читатели лишаются доступа к библиотеке на установленные администрацией сроки (от 1 до 6 месяцев). В случае утери или порчи книг читатель обязан заменить их такими же или другими изданиями, признанными библиотекой равноценными, или же возместить их десятикратную стоимость. В случае невозвращения в библиотеку книг в установленный срок читатель обязан заплатить штраф.

Срок пользования литературой для различных категорий читателей и количество выдаваемых изданий на каждом абонементе определяются администрацией, исходя из вида литературы и категории читателя. Число книг, выдаваемых в читальных залах, не ограничивается.

При поступлении новых изданий в библиотеку они должны быть внесены в картотеку с указанием их количества для каждого абонемента и читального зала. Данные о выдаче книг (сроки, штрафы и т. п.) собираются и обрабатываются администрацией.

Виды запросов в информационной системе

1. Получить перечень и общее число читателей для данного читального зала или абонемента по всей библиотеке, по признаку принадлежности к кафедре, факультету, курсу, группе.

2. Получить список и общее число всех читателей-должников, задолжников со сроком более 10 дней на данном абоненте, по всей библиотеке, по признаку принадлежности к кафедре, факультету, курсу, группе, по категориям читателей.

3. Получить перечень двадцати наиболее часто заказываемых книг в данном читальном зале для данного факультета, для всего вуза.

4. Получить перечень и общее число книг, поступивших и утерянных за последний год, для данного читального зала, абонемента или по всей библиотеке, по указанному автору, году выпуска, году поступления в библиотеку.

5. Определить пункт выдачи, на которой самое большое (маленькое) число читателей, читателей-должников, самая большая сумма задолженности.

6. Получить перечень и общее число книг, заказанных на межбиблиотечном абонемента за последний месяц, семестр, год.

7. Получить количество экземпляров книги для данного читального зала или абонента, во всей библиотеке, всех изданий.

8. Получить перечень и общее число читателей, лишенных права пользования библиотекой сроком более двух месяцев, во всей библиотеке, по признаку принадлежности к кафедре, факультету, курсу, группе, по категориям читателей.

9. Получить перечень и общее число новых читателей, выбывших читателей для данного читального зала или абонента за последний месяц, семестр, год, во всей библиотеке, по признаку принадлежности к кафедре, факультету, курсу, группе, по категориям читателей.

10. Получить перечень и общее число книг, заказанных данным читателем за последний месяц, семестр, год, список книг у него на руках.

11. Определить, есть ли данная книга в наличии на абонементах и в каком количестве.

12. Получить перечень читателей, у которых на руках некоторая книга и читателя, который раньше всех ее должен сдать.

13. Выдать полную информацию о читателе по его фамилии: группу, курс, факультет или кафедру, нарушения правил библиотеки, их количество, штрафы, утерянные книги и т. п.

4.17. Информационная система туристического клуба

Туристы, приходящие в туристический клуб, могут не только ходить в плановые походы, но и заниматься в различных секциях в течение всего года. Для этого они записываются в группы, относящиеся к определенным секциям.

Туристов можно условно разделить на любителей, спортсменов и тренеров. Каждая из перечисленных категорий может иметь свой набор характеристик-атрибутов. Секции клуба возглавляют руководители, в функции которых входит контроль за работой секции. В работе секции участвуют тренеры, административно относящиеся к одной из секций. Руководитель секции назначает каждой группе тренера. Тренер может тренировать несколько групп, причем не обязательно принадлежащих его секции. Спортсмены и тренеры могут участвовать в различных соревнованиях.

Каждый год составляется расписание работы секций. В нем указывается, какие будут проводиться тренировки и в каких секциях; их количество, место, время и т. д. В соответствии с этим руководители секций распределяют нагрузки для тренеров (с учетом их специальности). Сведения о проведенных тренировках и их посещаемости собираются руководителями.

В течение года клуб организует различные походы. Каждый поход имеет свой маршрут, на который отводится определенное количество дней. По маршруту и количеству дней определяется категория сложности данного похода. Поход возглавляет инструктор, которым может быть какой-либо тренер или спортсмен. Он набирает группу количеством от 5 до 15 человек исходя из типа похода (пеший, конный, водный, горный) и физических данных (туристов, не умеющих плавать, никогда не возьмут на сплав, а в пеший поход небольшой категории сложности могут взять любого туриста). Инструктор может водить в походы данной категории сложности, только если он сам его ранее уже прошел.

Походы могут быть плановыми и неплановыми. Для каждого планового похода существует точный план, в котором указываются маршрут, расписание привалов и стоянок на каждый день. Во время планового похода ведется дневник. Неплановые походы имеют только маршрут и полное время его прохождения. Неплановый поход может быть переведен в категорию плановых. Каждому туристу присваивается категория максимально сложного из пройденных им плановых походов.

Виды запросов в информационной системе

1. Получить список и общее число туристов, занимающихся в клубе, в указанной секции, группе, по половому признаку, году рождения, возрасту.

2. Получить список и общее число тренеров указанной секции, по всем секциям, по половому признаку, по возрасту, по размеру заработной платы, по специализации.

3. Получить перечень и общее число соревнований, в которых участвовали спортсмены из указанной секции, по всем секциям.

4. Получить список тренеров, проводивших тренировки в указанной группе, за указанный период времени.

5. Получить список и общее число туристов из некоторой секции, группы, которые ходили в заданное количество походов, ходили в указанный поход, ходили в поход в обозначенное время, ходили по определенному маршруту, были в некоторой точке, имеют соответствующую категорию.

6. Получить перечень руководителей секций полностью, по размеру заработной платы, по году рождения, возрасту, году поступления на работу.

7. Получить нагрузку тренеров (вид занятий, количество часов), ее объем по определенным видам занятий и общую нагрузку за указанный период времени для данного тренера или указанной секции.

8. Получить перечень и общее число маршрутов, по которым ходили туристы из указанной секции, в обозначенный период времени, по которым водил свои группы данный инструктор, по которым прошло указанное количество групп.

9. Получить перечень и общее число маршрутов, которые проходят через некоторую точку, имеют длину больше указанной, могут удовлетворять заданной категории сложности.

10. Получить перечень и общее число туристов из указанной секции, группы, которые могут ходить в указанные типы походов.

11. Получить перечень и общее число инструкторов, инструкторов-спортсменов, инструкторов-тренеров, которые имеют определенную категорию, которые ходили в указанное количество походов, ходили в определенный поход, ходили по некоторому маршруту, были в указанной точке.

12. Получить список туристов из указанной секции, группы, которые ходили в походы со своим тренером в качестве инструктора.

13. Получить список туристов из некоторой секции, группы, которые ходили по всем маршрутам, по указанным маршрутам.

4.18. Информационная система городской телефонной сети

ГТС представляет собой разветвленную сеть локальных АТС. АТС подразделяются на городские, ведомственные и учрежденческие и, возможно, обладают характерным только для этой группы набором атрибутов. У каждой АТС есть свои абоненты. У абонента может стоять телефон одного из трех типов: основной, параллельный или спаренный. За каждым абонентом (у него есть Ф.И.О., пол, возраст и т. д.) закреплен свой номер телефона, причем у нескольких абонентов может быть один и тот же номер (при параллельном или спаренном телефоне). Каждому номеру телефона соответствует адрес (индекс, район, улица, дом, квартира), причем параллельные или спаренные телефоны обязательно должны находиться в одном доме.

Все телефоны городской АТС имеют выход на межгород, но для конкретного абонента он может быть либо открыт, либо закрыт по какой-либо причине (отключен по желанию абонента, за неуплату и т. п.). Ведомственные и учрежденческие АТС имеют свою внутреннюю замкнутую сеть телефонов. Сведения о междугородных переговорах собираются и анализируются на ГТС.

Абоненты обязаны платить абонентскую плату. Плата должна вноситься каждый месяц до 20-го числа. При неуплате после письменного уведомления в течение двух суток телефон абоненту отключается. При задолженности за междугородные разговоры и неоплате после письменного уведомления отключается только возможность выхода на межгород. Включение того и (или) другого производится при оплате стоимости включения, абонентской платы и пени.

Абонентов любой АТС можно подразделить на простых и льготных. К категории льготников относятся пенсионеры, инвалиды и т. д. Льготники платят только 50 % абонентской платы. В соответствии со всем этим (тип телефона, наличие льготы, выход на межгород) рассчитывается размер абонентской платы.

На установку телефона существуют очереди: льготная и обычная. При подходе очередности рассматривается техническая возможность установки (наличие кабеля и свободного канала, наличие свободных телефонных номеров).

В городе также существуют общественные телефоны и таксофоны, расположенные по определенным адресам.

Виды запросов в информационной системе

1. Получить перечень и общее число абонентов указанной АТС полностью, только льготников, по возрастному признаку, по группе фамилий.
2. Получить перечень и общее число свободных телефонных номеров на указанной АТС, по всей ГТС, по признаку возможности установки телефона в данном районе.
3. Получить перечень и общее число должников на указанной АТС, по всей ГТС, по данному району, абонентов, которые имеют задолженность уже больше недели (месяца), по признаку задолженности за межгород и (или) по абонентской плате, по размеру долга.
4. Определить АТС (любого или конкретного типа), на которой самое большое (маленькое) число должников, самая большая сумма задолженности.
5. Получить перечень и общее число общественных телефонов и таксофонов во всем городе, принадлежащих указанной АТС, по признаку нахождения в данном районе.
6. Найти процентное соотношение обычных и льготных абонентов на указанной АТС, по всей ГТС, по данному району, по типам АТС.
7. Получить перечень и общее число абонентов указанной АТС, по всей ГТС, по данному району, по типам АТС, имеющих параллельные телефоны, только льготников, имеющих параллельные телефоны.
8. Определить, есть ли по данному адресу телефон, общее количество телефонов и (или) количество телефонов с выходом на межгород, с открытым выходом на межгород в данном доме, на конкретной улице.
9. Определить город, с которым происходит наибольшее количество междугородных переговоров.
10. Получить полную информацию об абонентах с заданным телефонным номером.
11. Получить перечень спаренных телефонов, для которых есть техническая возможность заменить их на обычные (выделить дополнительный номер).
12. Получить перечень и общее число внутренних на определенной ведомственной или учрежденческой АТС, с которых за некоторый период времени было произведено менее определенного числа внешних звонков.

13. Получить перечень и общее число должников на указанной АТС, по всей ГТС, по данному району, которым следует послать письменное уведомление, отключить телефон и (или) выход на межгород.

4.19. Информационная система театра

Работников театра можно подразделить на актеров, музыкантов, постановщиков и служащих. Каждая из перечисленных категорий имеет уникальные атрибуты-характеристики и может подразделяться (например, постановщики) на более мелкие категории. Театр возглавляет директор, в функции которого входят контроль за постановками спектаклей, утверждение репертуара, принятие на работу новых служащих, приглашение актеров и постановщиков. Актеры, музыканты и постановщики, работающие в театре, могут уезжать на гастроли. Актеры театра могут иметь звания заслуженных и народных артистов, могут быть лауреатами конкурсов. Также актерами театра могут быть и студенты театральных училищ. Каждый актер имеет свои вокальные и внешние данные (пол, возраст, голос, рост и т. п.), которые могут подходить для каких-то ролей, а для каких-то – нет (не всегда женщина может сыграть мужчину и наоборот).

Для постановки любого спектакля необходимо подобрать актеров на роли и дублеров на каждую главную роль. Естественно, что один и тот же актер не может играть более одной роли в спектакле, но может играть несколько ролей в различных спектаклях. У спектакля также имеется режисер-постановщик, художник-постановщик, дирижер-постановщик, автор. Спектакли можно подразделить по жанрам: музыкальная комедия, трагедия, оперетта и пр. С другой стороны, спектакли можно подразделить по возрасту: детские, молодежные и пр. В репертуаре театра указывается, какие спектакли, в какие дни и в какое время будут проходить, а также даты премьер. В кассах театра можно заранее приобрести билеты или абонемент на любые спектакли. Абонемент обычно включает в себя билеты на спектакли либо конкретного автора, либо конкретного жанра. Цена билетов зависит от места и спектакля. На премьеры билеты дороже. Администрация театра фиксирует количество проданных билетов на каждый спектакль.

Виды запросов в информационной системе

1. Получить список и общее число всех работников театра, актеров, музыкантов, по стажу работы в театре, по половому признаку,

году рождения, возрасту, признаку наличия и количества детей, размеру заработной платы.

2. Получить перечень и общее число спектаклей, указанных в репертуаре на данный сезон, уже сыгранных спектаклей, спектаклей указанного жанра, когда-либо сыгранных в этом театре, за указанный период.

3. Получить перечень и общее число всех поставленных спектаклей, спектаклей указанного жанра, когда-либо поставленных в этом театре, поставленных за указанный период.

4. Получить список авторов поставленных спектаклей, авторов, живших в указанном веке, авторов указанной страны, авторов спектаклей указанного жанра когда-либо поставленных в этом театре, поставленных за указанный период времени.

5. Получить перечень спектаклей указанного жанра, некоторого автора, авторов обозначенной страны, спектаклей, написанных в определенном веке, впервые поставленных на сцене указанного театра в обозначенный период времени.

6. Получить список актеров, подходящих по своим данным на указанную роль.

7. Получить общее число и список актеров театра, имеющих звания, получивших их за некоторый период, на указанных конкурсах, по половому признаку, по возрасту.

8. Получить список актеров и постановщиков, приехавших когда-либо на гастроли в театр за указанный период, перечень уезжавших на гастроли в определенное время с данным спектаклем.

9. Получить список для указанного спектакля: актеров, их дублеров, имена режиссера-постановщика, художника-постановщика, дирижера-постановщика, авторов, дату премьеры.

10. Получить перечень и общее число ролей, сыгранных указанным актером всего, за некоторый период времени, в спектаклях определенного жанра, в спектаклях указанного режиссера-постановщика, в детских спектаклях.

11. Получить сведения о числе проданных билетов на все спектакли, на конкретный спектакль, на премьеры, за указанный период, в том числе проданных предварительно.

12. Получить общую сумму вырученных денег за указанный спектакль, за некоторый период времени.

13. Получить перечень и общее число свободных мест на все спектакли, на конкретный спектакль, на премьеры.

4.20. Информационная система зоопарка

Служащих зоопарка можно подразделить на несколько категорий: ветеринары, уборщики, дрессировщики, строители-ремонтники, работники администрации. Каждая из перечисленных категорий работников имеет уникальные атрибуты-характеристики, определяемые профессиональной направленностью. За каждым животным ухаживает определенный круг служащих, причем только ветеринарам, уборщикам и дрессировщикам разрешен доступ в клетки к животным.

В зоопарке обитают животные различных климатических зон, поэтому часть животных на зиму необходимо переводить в отапливаемые помещения. Животных можно подразделить на хищников и травоядных. При расселении животных по клеткам необходимо учитывать не только потребности данного вида, но и их совместимость с животными в соседних клетках (нельзя рядом селить, например, волков и их добычу – различных копытных).

Для кормления животных необходимы различные типы кормов: растительный, живой, мясо и различные комбикорма. Растительный корм – это фрукты и овощи, зерно и сено. Живой корм – мыши, птицы, корм для рыб. Для каждого вида животных рассчитывается свой рацион, который в свою очередь варьируется в зависимости от возраста, физического состояния животного и сезона. Таким образом, у каждого животного в зоопарке имеется меню на каждый день, в котором указываются количество и время кормления, количество и вид пищи (обезьянам необходимы фрукты и овощи, мелким хищникам – хорькам, ласкам, совам, некоторым кошачьим, змеям надо давать мышшей). У зоопарка имеются поставщики кормов для животных. Каждый поставщик специализируется на каких-то конкретных видах кормов. Часть кормов зоопарк может производить сам: запастись сеном, разводить мышшей и т. д.

Ветеринары должны проводить медосмотры, следить за весом, ростом, развитием животного, ставить своевременно прививки и заносить все эти данные в карточку, которая заводится на каждую особь при ее появлении в зоопарке. Больным животным назначается лечение и при необходимости их можно изолировать в стационаре.

При определенных условиях (наличие пары особей, подходящих по возрасту, физическому состоянию) можно ожидать появления потомства. Потомство от данной пары животных при достижении ими положенного возраста можно либо оставить в зоопарке, создав для

них подходящие условия содержания, либо обменяться с другими зоопарками или просто раздать в другие зоопарки – по решению администрации.

Виды запросов в информационной системе

1. Получить список и общее число служащих зоопарка, либо служащих данной категории полностью, по продолжительности работы в зоопарке, по половому признаку, возрасту, размеру заработной платы.

2. Получить перечень и общее число служащих зоопарка, ответственных за указанный вид животных либо за конкретную особь за все время пребывания животного в зоопарке, за указанный период времени.

3. Получить список и общее число служащих зоопарка, имеющих доступ к указанному виду животных либо к конкретной особи.

4. Получить перечень и общее число всех животных в зоопарке либо животных указанного вида, живших в указанной клетке все время пребывания в зоопарке, по половому признаку, возрасту, весу, росту.

5. Получить перечень и общее число нуждающихся в теплом помещении на зиму, полностью животных только указанного вида или указанного возраста.

6. Получить перечень и общее число животных, которым поставлена указанная прививка, либо переболевших некоторой болезнью, по длительности пребывания в зоопарке, половому признаку, возрасту, признаку наличия и количеству потомства.

7. Получить перечень всех животных, совместимых с указанным видом, либо только тех животных, которых необходимо переселить или тех, которые нуждаются в теплом помещении.

8. Получить перечень и общее число поставщиков кормов полностью, либо поставляющих только определенный корм, поставлявших в указанный период, по количеству поставляемого корма, цене, датам поставок.

9. Получить перечень и объем кормов, производимых зоопарком полностью, либо только тех кормов, в поставках которых зоопарк не нуждается (обеспечивает себя сам).

10. Получить перечень и общее число животных полностью, либо указанного вида, которым необходим определенный тип кормов, в указанном сезоне, возрасте или круглый год.

11. Получить полную информацию (рост, вес, прививки, болезни, дата поступления в зоопарк или дата рождения, возраст, количество

потомства) о всех животных или о животных только данного вида, о конкретном животном, об особи, живущей в указанной клетке.

12. Получить перечень животных, от которых можно ожидать потомства в перспективе, в указанный период.

13. Получить перечень и общее число зоопарков, с которыми был произведен обмен животными в целом или животными только указанного вида.

4.21. Информационная система ГИБДД

ГИБДД занимается выделением и учетом номерных знаков на автотранспорт. К автотранспортным средствам относятся легковые, грузовые автомобили, прицепы, полуприцепы, мотоциклы, тракторы, автобусы, микроавтобусы. На разные виды транспорта выдаются разные виды номеров и в базу данных заносятся разные характеристики. Номера могут выделяться как частным владельцам, так и организациям. В справочнике номеров, выданных частным владельцам, фиксируются государственный номер, Ф.И.О. владельца, его адрес, марка автомобиля, дата выпуска, объем двигателя, номера двигателя, шасси и кузова, цвет и т. п. В справочнике номеров, выданных организации, дополнительно фиксируются название организации, район, адрес, руководитель. Существует справочник свободных номеров (серия, диапазон номеров). ГИБДД периодически проводит технический осмотр (ТО) машин. Для прохождения ТО необходима квитанция об оплате налогов, сумма оплаты зависит от объема двигателя. Периодичность прохождения зависит от года выпуска и вида транспортного средства. Технические характеристики, проверяемые на ТО, и допуски также зависят от вида транспортного средства.

ГИБДД занимается учетом и анализом ДТП. При регистрации ДТП фиксируются дата, тип происшествия (наезд на пешехода, наезд на ограждение либо столб, лобовое столкновение, наезд на впереди стоящий транспорт, боковое столкновение на перекрестке и т. п.), место происшествия, марки пострадавших автомобилей, государственный номер, тип машины (легковая, грузовая, специальная), краткое содержание, число пострадавших, сумма ущерба, причина, дорожные условия и т. п. Анализ накопленной по ДТП статистике поможет правильно расставить запрещающие и предупреждающие знаки на улицах города, а также спланировать местонахождение постов патрульных.

Угон либо исчезновение виновника ДТП с места происшествия требует оперативного вмешательства всех постов ГИБДД и патрульных машин. Для информирования о разыскиваемой машине ее данные (включая номера двигателя и кузова) извлекаются из базы зарегистрированных номеров и передаются по радиии всем постам. Ведение статистики угонов, ее анализ и опубликование результатов в СМИ помогут снизить количество угонов, а хозяевам машин – принять необходимые меры (самые угоняемые марки, самый популярный способ вскрытия, самые надежные сигнализации и т. п.).

Виды запросов в информационной системе

1. Получить перечень и общее число организаций, которым выделены номера либо с указанной серией, либо за указанный период.
2. Получить сведения о владельце автотранспортного средства по государственному номеру автомашины.
3. Получить все данные на автомобиль по государственному номеру: номера двигателя, кузова и шасси, участие в ДТП, наличие ТО.
4. Получить перечень и общее число владельцев машин, не прошедших вовремя ТО.
5. Получить статистику по любому типу ДТП за указанный период.
6. Получить результаты анализа ДТП: самые опасные места в городе, самая частая причина ДТП.
7. Получить данные о количестве ДТП, совершаемых водителями в нетрезвом виде и долю таких происшествий в общем количестве ДТП.
8. Получить список машин, отданных в розыск, будь то скрывшиеся с места ДТП или угнанные.
9. Получить данные об эффективности розыскной работы: количество найденных машин в процентном отношении.
10. Получить перечень и общее число угонов за указанный период.
11. Получить статистику по угонам: самые угоняемые марки машин, самые надежные сигнализации и т. п.

4.22. Информационная система фотоцентра

Фотоцентр имеет главный офис и сеть филиалов и киосков приема заказов, расположенных по определенным адресам. Филиалы и киоски различаются количеством рабочих мест. В киосках осуществляется только прием заказов, поэтому каждый киоск прикреплен к опре-

деленному филиалу, в котором эти заказы выполняются. В филиалах имеется необходимое оборудование для проявки пленок и печати фотографий. Филиалы и киоски принимают заказы на проявку пленок, печать фотографий и проявку и печать вместе. В заказе на печать указываются количество фотографий с каждого кадра, общее количество фотографий, формат, тип бумаги и срочность выполнения заказа. При заказе большого количества фотографий предоставляются скидки. Срочные заказы принимаются только в филиалах и имеют цену в два раза больше, чем обычный заказ. При приобретении дисконтной карты клиент получает значительные скидки на печать фотографий. Пленка, приобретенная в том же филиале, куда она принесена на проявку, проявляется бесплатно.

Клиентов можно разделить на профессионалов и любителей. Профессионалам, приносящим заказы в один и тот же филиал, могут быть предложены персональные скидки. Фотомагазины и киоски предлагают к продаже различные фототовары: фотопленки, фотоаппараты, альбомы и другие фотопринадлежности. Фотомагазины также предлагают дополнительные виды услуг: фотографию на документы, реставрацию фотографий, прокат фотоаппаратов, художественное фото, услуги профессионального фотографа.

Сведения о выполненных заказах и продаже различных фототоваров собираются и обрабатываются, и на основе этой информации делается общий заказ на поставку расходных материалов (фотобумага, фотопленка, химические реактивы), фототоваров и оборудования. Полученные товары и материалы распределяются в соответствии с запросами по киоскам и магазинам. У фотоцентра может быть несколько поставщиков, которые специализируются на различных поставках, либо на поставках фототоваров различных фирм.

Виды запросов в информационной системе

1. Получить перечень и общее число пунктов приема заказов на фотоработы по филиалам, по киоскам приема заказов, в целом по фотоцентру.

2. Получить перечень и общее число заказов на фотоработы по филиалам, киоскам приема заказов, в целом по фотоцентру, поступивших в течение некоторого периода времени.

3. Получить перечень и общее число заказов (отдельно простых и срочных) на отдельные виды фоторабот по указанному филиалу, киоску приема заказов, поступивших в течение некоторого периода времени.

4. Получить сумму выручки с заказов (отдельно простых и срочных) на отдельные виды фоторабот по указанному филиалу, киоску приема заказов, поступивших в течение некоторого периода времени.

5. Получить количество отпечатанных фотографий в рамках простых и срочных заказов по указанному филиалу, киоску приема заказов, фотоцентру в целом за некоторый период времени.

6. Получить количество проявленных фотоплёнок в рамках простых и срочных заказов по указанному филиалу, киоску приема заказов, фотоцентру в целом за некоторый период времени.

7. Получить перечень поставщиков в целом по фотоцентру, поставщиков отдельных видов фототоваров, сделавших поставки в некоторый период, поставки определенного объема.

8. Получить список клиентов в целом по фотоцентру, клиентов указанного филиала, имеющих скидки, сделавших заказы определенного объема.

9. Получить сумму выручки от реализации фототоваров в целом по фотоцентру, по указанному филиалу, проданных в течение некоторого периода времени.

10. Получить перечень фототоваров и фирм, их производящих, которые пользуются наибольшим спросом в целом по фотоцентру, в указанном филиале.

11. Получить перечень реализованных фототоваров и объемы их реализации в целом по фотоцентру, по указанному филиалу, проданных в течение некоторого периода времени.

12. Получить перечень рабочих мест фотоцентра в целом и указанного профиля.

4.23. Информационная система железнодорожной пассажирской станции

Работников железнодорожной станции можно подразделить на водителей подвижного состава, диспетчеров, ремонтников подвижного состава, ремонтников путей, кассиров, работников службы подготовки составов, справочной службы и других, которые административно относятся каждый к своему отделу. Каждая из перечисленных категорий работников имеет уникальные атрибуты-характеристики, определяемые профессиональной направленностью. В отделах существует разбиение работников на бригады. Отделы возглавляют начальники,

которые представляют собой администрацию железнодорожной станции. В функции администрации входит планирование маршрутов, составление расписаний, формирование кадрового состава железнодорожной станции. За каждым локомотивом закрепляется локомотивная бригада. За несколькими локомотивами закрепляется бригада техников-ремонтников, выполняющая рейсовый и плановый техосмотр (по определенному графику), ремонт, техническое обслуживание. Водители локомотивов обязаны проходить каждый год медосмотр, не прошедших медосмотр необходимо перевести на другую работу. Локомотив должны своевременно осматривать техники-ремонтники и при необходимости ремонтировать. Подготовка к рейсу включает в себя техническую часть (рейсовый техосмотр, мелкий ремонт) и обслуживающую часть (уборка вагонов, запас продуктов питания и т. п.).

В расписании указываются тип поезда (скорый, пассажирский и пр.), номер поезда, дни и время отправления и прибытия, маршрут (начальный и конечный пункты назначения, основные узловые станции), стоимость билета. Билеты на поезд можно приобрести заранее или забронировать в железнодорожных кассах. До отправления поезда (если есть необходимость) билет можно вернуть. Отправление поездов может быть задержано из-за опозданий других поездов, погодных условий, технических неполадок.

Железнодорожные маршруты можно разделить на следующие категории: внутренние, международные, туристические, специальные. Пассажиры могут сдавать свои вещи в багажное отделение.

Виды запросов в информационной системе

1. Получить перечень и общее число всех работников железнодорожной станции, начальников отделов, работников указанного отдела, по стажу работы на станции, половому признаку, возрасту, по наличию и количеству детей, размеру заработной платы.

2. Получить перечень и общее число работников в бригаде, по всем отделам, в указанном отделе, обслуживающих некоторый локомотив, по возрасту, суммарной (средней) зарплате в бригаде.

3. Получить перечень и общее число водителей локомотивов, прошедших либо не прошедших медосмотр в указанный год, по половому признаку, возрасту, размеру заработной платы.

4. Получить перечень и общее число локомотивов, приписанных к железнодорожной станции, находящихся на ней в указанное время, по времени прибытия на станции, по количеству совершенных маршрутов.

5. Получить перечень и общее число локомотивов, прошедших плановый техосмотр за определенный период времени, отправленных в ремонт в обозначенное время, отремонтированных указанное число раз, по количеству совершенных рейсов до ремонта, по возрасту локомотива.

6. Получить перечень и общее число поездов на указанном маршруте, по длительности маршрута, по цене билета и по всем этим критериям сразу.

7. Получить перечень и общее число отмененных рейсов полностью, в указанном направлении, по указанному маршруту.

8. Получить перечень и общее число задержанных рейсов полностью, по указанной причине, по указанному маршруту, и количество сданных билетов за время задержки.

9. Получить перечень и среднее количество проданных билетов за указанный интервал времени на определенные маршруты, по длительности маршрута, по цене билета.

10. Получить перечень и общее число маршрутов указанной категории, следующих в определенном направлении.

11. Получить перечень и общее число пассажиров на указанном рейсе, уехавших в указанный день, уехавших за границу в указанный день, по признаку сдачи вещей в багажное отделение, по половому признаку, по возрасту.

12. Получить перечень и общее число невыкупленных билетов на указанном рейсе, день, некоторый маршрут.

13. Получить общее число сданных билетов на указанный рейс, день, маршрут.

4.24. Информационная система городской филармонии

Инфраструктура городской филармонии представлена культурными сооружениями различного типа: театры, концертные площадки, эстрады, дворцы культуры и т. д. Каждая из категорий культурных сооружений обладает атрибутами, специфичными только для нее: театр характеризуется вместимостью, кинотеатр – размером экрана.

Артисты под руководством импресарио выступают в различных жанрах, при этом один и тот же артист может выступать в нескольких жанрах и может работать с несколькими импресарио.

Организаторы концертных мероприятий проводят выступления, концерты, конкурсы в культурных сооружениях города, организуя участие в нем артистов. По результатам участия артистов в конкурсах производится награждение.

Виды запросов в информационной системе

1. Получить перечень культурных сооружений указанного типа в целом или удовлетворяющих заданным характеристикам (например, залы, вмещающие не менее указанного числа зрителей).
2. Получить список артистов, выступающих в некотором жанре.
3. Получить список артистов, работающих с некоторыми импресарио.
4. Получить список артистов, выступающих более чем в одном жанре с их указанием.
5. Получить список импресарио указанного артиста.
6. Получить перечень концертных мероприятий, проведенных в течение заданного периода времени в целом либо указанным организатором.
7. Получить список призеров указанного конкурса.
8. Получить перечень концертных мероприятий, проведенных в указанном культурном сооружении.
9. Получить список импресарио определенного жанра.
10. Получить список артистов, не участвовавших ни в каких конкурсах в течение определенного периода времени.
11. Получить список организаторов культурных мероприятий и число проведенных ими концертов в течение определенного периода времени.
12. Получить перечень культурных сооружений, а также даты проведения в них культурных мероприятий в течение определенного периода времени.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Хансен Г. Базы данных: разработка и управление / Г. Хансен, Дж. Хансен. – М.: Бином, 1999. – 699 с.
2. Дейт К. Дж. Введение в системы баз данных / К. Дейт. – 6 изд. – М. и др.: Вильямс, 2000. – 848 с.
3. Карпова Т.С. Базы данных: модели, разработка, реализация : учебник / Т. Карпова. – СПб. и др.: Питер, 2001. – 303 с.
4. Коннолли Т. Базы данных : проектирование, реализация и сопровождение. Теория и практика: учеб. пособие / Т. Коннолли, К. Бегг, А. Страчан. – 2 изд., испр. и доп. / Т. Коннолли, К. Бегг. – М. [и др.]: Вильямс, 2000. – 1120 с.
5. Малыхина М.П. Базы данных: основы, проектирование, использование: учеб. пособие / М.П. Малыхина. – 2-е изд. – СПб.: БХВ-Петербург, 2006. – 517 с.
6. Марков А.С. Базы данных. Введение в теорию и методологию: учебник / А.С. Марков, К.Ю. Лисовский. – М.: Финансы и статистика, 2006. – 510 с.
7. Советов Б.Я. Базы данных. Теория и практика: учебник / Б.Я. Советов, В.В. Цехановский, В.Д. Чертовской. – М.: Высшая школа (ВШ), 2007. – 462 с.
8. Кириллов В.В. Введение в реляционные базы данных / В. Кириллов, Г. Громов. – СПб.: БХВ-Петербург, 2009. – 454 с.
9. Грофф Дж. Р. SQL : полное руководство / Дж. Р. Грофф, Пол Н. Вайнберг. – Киев: ВНУ: Ирина, 2001. – 816 с.
10. Грабер М. Введение в SQL / М. Грабер. – М.: ЛОРИ, 1996. – 375 с.
11. Вербовецкий А.А. Основы проектирования баз данных / А.А. Вербовецкий. – М.: Радио и связь, 2000. – 85 с.

Стасьшин Владимир Михайлович

**ПРОЕКТИРОВАНИЕ ИНФОРМАЦИОННЫХ СИСТЕМ
И БАЗ ДАННЫХ**

Учебное пособие

Редактор *А.Н. Куткин*
Выпускающий редактор *И.П. Брованова*
Корректор *И.Е. Семенова*
Дизайн обложки *А.В. Ладыжская*
Компьютерная верстка *В.Н. Зенина*

Подписано в печать 20.12.2012. Формат 60×84 1/16. Бумага офсетная. Тираж 100 экз.
Уч.-изд. л. 5,81. Печ. л. 6,25. Изд. № 189. Заказ № 72. Цена договорная

Отпечатано в типографии
Новосибирского государственного технического университета
630092, г. Новосибирск, пр. К. Маркса, 20