

ФЕДЕРАЛЬНОЕ АГЕНТСТВО ПО ОБРАЗОВАНИЮ
Государственное образовательное учреждение
высшего профессионального образования
«ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ ПЕДАГОГИЧЕСКИЙ УНИВЕРСИТЕТ»

Кафедра информатики и методики преподавания математики

Учебно-методический комплект дисциплины «Информационные системы»

МЕТОДИЧЕСКИЕ УКАЗАНИЯ
К ЛАБОРАТОРНЫМ РАБОТАМ

Специальность 030100.00 Информатика с дополнительной специальностью

Воронеж
2005

ЛАБОРАТОРНАЯ РАБОТА № 1

Тема: **Реляционные операторы: реляционная алгебра**

Продолжительность 7 часов

1. Целью лабораторных работ является знакомство с математической основой языков манипулирования данными в реляционных информационных системах, а именно – с реляционной алгеброй.

2. Вопросы, подлежащие исследованию, связаны с изучением базовых теоретико-множественных операций реляционной алгебры и специальных реляционных операций.

3. Теоретические и справочно-информационные материалы.

Могут использоваться из Internet по ссылкам:

http://impm.vspu.ac.ru/~chul/Inf_Syst/index.htm

http://www.vspu.ac.ru/~chul/Inf_Syst/index.htm

3.1. Введение

Операторы – третья и последняя часть реляционной модели.

Для получения информации из отношений необходим язык манипулирования данными (ЯМД), способный выполнять соответствующие операции над отношениями. Наиболее важной частью ЯМД является его раздел для формулировки запросов. Поскольку запросы в общем случае представляют собой произвольные функции над отношениями, необходимо решить вопрос о требуемой выразительности языка запросов. Для исследования этого вопроса были разработаны три типа теоретических языков: реляционная алгебра, реляционное исчисление с переменными-кортежами и реляционное исчисление с переменными-доменами.

Языки запросов первого типа – алгебраические языки – основаны на классической теории множеств (с некоторыми уточнениями) и позволяют выражать запросы средствами специализированных операторов, применяемых к отношениям. Неформально реляционную алгебру можно описать как (высокоуровневый) процедурный язык, с помощью которого можно

сообщить СУБД, *как* построить новое отношение из одного или нескольких существующих в базе данных отношений.

Языки запросов второго и третьего типов – языки исчисления – основаны на классическом логическом аппарате исчисления предикатов первого порядка и позволяют выражать свойства желаемого результата, фактически не указывая, как его получить. Поэтому реляционное исчисление, с неформальной точки зрения, представляет собой непроцедурный язык, который можно использовать для определения того, *каким* будет некоторое отношение, созданное на основе одного или нескольких других отношений базы данных.

Оба механизма (алгебра и исчисление) обладают важным свойством: они **замкнуты** относительно понятия отношения. Это означает, что выражения реляционной алгебры и формулы реляционного исчисления определяются над отношениями реляционных БД и результатом вычисления также являются отношения. В результате любое выражение или формула могут пониматься как отношения. Языки этих трех типов были предложены Е.Ф. Коддом и служат теоретической основой для описания действий, выполняемых над отношениями. Определенные в них элементарные операции реализуются в любом реальном языке запросов независимо от его внешнего оформления.

Реальные языки (SQL, QBE и др.) обеспечивают не только функции соответствующего теоретического языка или их комбинации, но и реализуют некоторые дополнительные операции - арифметические операции, команды присваивания и печати и т.п. Конкретный язык манипулирования реляционными БД называется **реляционно полным**, если любой запрос, выраженный с помощью одного выражения реляционной алгебры или одной формулы реляционного исчисления, может быть выражен с помощью одного оператора языка. При этом для любого допустимого выражения реляционной алгебры можно построить эквивалентную формулу реляционного исчисления, и наоборот, то есть эти механизма *эквивалентны*.

3.2. Обзор реляционной алгебры

В основе реляционной алгебры лежит идея о том, что так как отношение – это множество кортежей, то и средства манипулирования отношениями должны быть такими же, как традиционные теорети-

ко-множественные операции, дополненные специфическими для баз данных операциями.

Расширенный начальный вариант алгебры, опеределенный Коддом, состоит из восьми алгебраических операций, которые делятся на два класса – теоретико-множественные операции и специальные реляционные операции.

В состав теоретико-множественных операций входят:

- объединение отношений;
- пересечение отношений;
- взятие разности отношений;
- декартово произведение отношений.

Специальные реляционные операции включают:

- проекцию отношения;
- соединение отношений;
- деление отношений;
- выборка или ограничение отношения.

Приведем общую интерпретацию этих реляционных операций:

- **Выборка:** возвращает отношение, содержащее кортежи из заданного отношения, которые удовлетворяют указанным условиям;
- **Проекция:** возвращает отношение, содержащее все кортежи заданного отношения, которые остались в этом отношении после исключения из него некоторых атрибутов;
- **Произведение:** возвращает отношение, содержащее все возможные кортежи, которые являются сочетанием двух кортежей, принадлежащих соответственно двум заданным отношениям;
- **Объединение:** возвращает отношение, содержащее все кортежи, которые принадлежат либо одному, либо обоим из заданных отношений;
- **Пересечение:** возвращает отношение, содержащее кортежи, которые принадлежат одновременно двум заданным отношениям;

- **Разность:** возвращает отношение, содержащее все кортежи, которые принадлежат первому из двух заданных отношений и не принадлежат второму;
- **Соединение:** возвращает отношение, кортежи которого – это сочетания двух кортежей (принадлежащих соответственно двум определенным отношениям), имеющих общее значение для одного или нескольких общих атрибутов этих двух отношений (и такие общие значения в результирующем кортеже появляются только один раз, а не дважды).
- **Деление:** для двух отношений, бинарного и унарного, возвращает отношение, содержащее все значения одного атрибута бинарного отношения, которые соответствуют (в другом атрибуте) всем значениям в унарном отношении.

Кроме того, в состав алгебры включается операция присваивания, позволяющая сохранить в базе данных результаты вычисления алгебраических выражений, и операция переименования атрибутов, дающая возможность корректно сформировать заголовок (схему) результирующего отношения.

Поскольку результатом любой реляционной операции (кроме операции присваивания) является некоторое отношение, можно образовывать реляционные выражения, в которых вместо отношения-операнда некоторой реляционной операции находится **вложенное** реляционное выражение.

Для алгебры, у которой операции замкнуты относительно понятия отношения, каждая операция должна производить отношение с двумя составляющими: телом и заголовком. Только в этом случае будет действительно возможно строить вложенные выражения. Операции объединения, пересечения, взятия разности (вычитания) требуют от отношений-операндов *совместимых по типу*. Два отношения называются **совместимыми по типу**, если каждое из них имеет одинаковое множество имен атрибутов и если соответствующие атрибуты определены на одном и том же домене. Если необходимо выполнить операцию объединения, пересечения или вычитания двух отношений, которые почти совместимы по типу, за исключением некоторых различий в именах атрибутов, можно использовать оператор переименования, чтобы сделать эти отношения полностью совместимыми по типу, прежде чем выполнить необходимую операцию.

Теперь можно приступить к определениям отдельных операций начальной алгебры. В дальнейшем изложении будем предполагать применение операции переименования во всех конфликтных случаях.

В дальнейшем рассмотрении будут приведены традиционное обозначение операции, например, как в [3, 8, 12, 16] (левое поле таблицы) и обозначение, принятое в книге Дейта [14] (правое поле). Для иллюстрирующих примеров используем базу данных поставщиков и деталей.

3.2.1. Операции присваивания и переименования

• ПРИСВАИВАНИЕ

Операция **присваивания** позволяет присвоить результат вычисления реляционного выражения некоторой переменной отношения БД.

←	:=
---	----

ПРИМЕР 3.1. Операция присваивания

Присвоить значение переменной отношения ПОСТАВЩИК переменной отношения R

R ← ПОСТАВЩИК **или**
R := ПОСТАВЩИК

Отношение ПОСТАВЩИК справа от знака операции присваивания является частным случаем реляционного выражения.

Операции реляционной алгебры могут в конечном счете стать чрезвычайно сложными. Для упрощения такие операции можно разбить на ряд меньших операций реляционной алгебры и присвоить имена результатам промежуточных выражений.

• ПЕРЕИМЕНОВАНИЕ

Операция **переименования** производит отношение, тело которого совпадает с телом операнда, но имена атрибутов изменены.

$R \leftarrow \rho_{b_1, b_2, \mathbf{K}, b_n}(A)$	$R := (A \text{ rename } a_1 \text{ as } b_1, \mathbf{K}, a_n \text{ as } b_n)$
--	---

ПРИМЕР 3.2. Операция переименования

Переименовать атрибуты отношения ПОСТАВЩИК: ПНОМ в П#, ГОРОД в ПГОРОД

$R \leftarrow \rho_{П\#, ПФАМ, СТАТУС, ПГОРОД}(ПОСТАВЩИК)$ или
 $R := (ПОСТАВЩИК \text{ rename } ПНОМ \text{ as } П\#, ГОРОД \text{ as } ПГОРОД)$

С помощью этих выражений (которые могут быть вложены в другие выражения) вычисляется отношение, имеющее то же самое тело, что и отношение ПОСТАВЩИК, но с именем атрибута П# вместо ПНОМЕР и ПГОРОД вместо ГОРОД (рис. 3.1).

R

П#	ПФАМ	СТАТУС	ПГОРОД
П1	Иванов	20	Воронеж
П2	Петров	15	Москва
П3	Сидоров	10	Москва
П4	Зайцев	30	Воронеж
П5	Волков	20	Киев

Рис. 3.1. Переименованное отношение ПОСТАВЩИК

3.2.2. Унарные операции

Начнем с изучения двух унарных операций: выборки и проекции.

- **ВЫБОРКА (ОГРАНИЧЕНИЕ, СЕЛЕКЦИЯ)**

Выборка – это сокращенное название θ -выборки, где θ обозначает любой скалярный оператор сравнения ($=, <>, >, >=$ и т.д.).

q-выборкой из отношения A по атрибутам X и Y (в этом порядке)

$A \text{ where } X \theta Y$

называется отношение, имеющее тот же заголовок, что и отношение A , и тело, содержащее множество всех кортежей t отношения A , для которых проверка условия “ $X \theta Y$ ” дает значение *истина*. Атрибуты X и Y

должны быть определены на одном и том же домене, а оператор должен иметь смысл для этого домена.

Выборка, как было определено, допускает простые сравнения в выражении where. Однако на основании свойства замкнутости можно однозначно расширить определение до формы, в которой условие в выражении where будет содержать произвольное число логических сочетаний таких простых сравнений (с использованием логических связок AND, OR, NOT и круглых скобок).

$R \leftarrow \sigma_{X\theta Y} (A)$	$R := A \text{ where } X \theta Y$
---	------------------------------------

ПРИМЕР 3.3. Операция выборки

Составить список всех поставщиков из Воронежа.

$$R \leftarrow \sigma_{ГОРОД = \text{Воронеж}} (\text{ПОСТАВЩИК}) \text{ или}$$

$$R := \text{ПОСТАВЩИК where ГОРОД} = \text{'Воронеж'}$$

R

ПНОМ	ПФАМ	СТАТУС	ГОРОД
П1	Иванов	20	Воронеж
П4	Зайцев	30	Воронеж

Рис. 3.2. Результат операции выборки

- **ПРОЕКЦИЯ**

Операция проекции применяется к одному отношению и определяет новое отношение, содержащее вертикальное подмножество отношения-операнда, т.е. подмножество, получаемое исключением всех атрибутов, не указанных в списке атрибутов, с последующим уничтожением кортежей-дубликатов из того, что осталось.

$R \leftarrow \Pi_{a_1, a_2, \dots, a_n} (A)$	$R := A[a1, a2, \mathbf{K}, an]$
---	----------------------------------

ПРИМЕР 3.4. Операция проекции

Составить список всех городов, в которых размещаются поставщики.

$$R \leftarrow \Pi_{\text{ГОРОД}}(\text{ПОСТАВЩИК}) \text{ или}$$

$$R := \text{ПОСТАВЩИК}[\text{ГОРОД}]$$

Результат операции показан на рис.3.3. Обратите внимание, что в этом примере (проекция поставщиков по атрибуту ГОРОД), хотя отношение ПОСТАВЩИК имеет пять кортежей и, следовательно, пять городов, в результате есть только три города, поскольку исключены дублирующие кортежи.

ГОРОД
Воронеж
Москва
Киев

Рис. 3.3. Результат операции проекции

3.2.3. Традиционные операции над множествами

- **ОБЪЕДИНЕНИЕ**

Объединением двух *совместимых по типу* отношений A и B называется отношение с тем же заголовком, как и в отношениях A и B , и с телом, состоящим из множества всех кортежей, принадлежащих A или B , или обоим отношениям за исключением повторяющихся.

$R \leftarrow A \cup B$	$R := A \text{ union } B$
-------------------------	---------------------------

ПРИМЕР 3.5. Операция объединения

Составить список городов, в которых или размещаются поставщики, или хранятся на складе детали (либо и то и другое).

$$R \leftarrow \Pi_{\text{ГОРОД}}(\text{ПОСТАВЩИК}) \cup \Pi_{\text{ГОРОД}}(\text{ДЕТАЛЬ})$$

или

$R := \text{ПОСТАВЩИК}[\text{ГОРОД}] \text{ union } \text{ДЕТАЛЬ}[\text{ГОРОД}]$

Для создания совместимых по типу отношений сначала следует применить операцию проекции, чтобы выделить из отношений ПОСТАВЩИК и ДЕТАЛЬ столбцы с атрибутами Город, исключая в случае необходимости дубликаты. Затем для комбинирования полученных промежуточных отношений следует использовать операцию объединения. Результат выполнения всех этих действий приведен на рис.3.4.

R
ГОРОД
Воронеж
Москва
Киев
Минск

Рис. 3.4. Результат операции объединения

- **РАЗНОСТЬ**

Разность двух *совместимых по типу* отношений A и B – это отношение с тем же заголовком, как и в отношениях A и B , и с телом, состоящим из множества всех кортежей, принадлежащих отношению A и не принадлежащих отношению B .

$R \leftarrow A - B$	$R := A \text{ minus } B$
----------------------	---------------------------

ПРИМЕР 3.6. Операция разности

Составить список городов, в которых детали хранятся на складе, но нет поставщиков.

$R \leftarrow \Pi_{\text{ГОРОД}}(\text{ДЕТАЛЬ}) - \Pi_{\text{ГОРОД}}(\text{ПОСТАВЩИК})$

или

$R := \text{ДЕТАЛЬ}[\text{ГОРОД}] \text{ minus } \text{ПОСТАВЩИК}[\text{ГОРОД}]$

В данном случае аналогично предыдущему примеру следует создать совместимые по типу отношения ДЕТАЛЬ и ПОСТАВЩИК, выполнив их проекцию по атрибуту Город. Затем для комбинирования новых получен-

ных отношений следует использовать операцию разности. Результат выполнения всех этих действий приведен на рис.3.5.

R
ГОРОД
Минск

Рис. 3.5. Результат операции разности

- **ПЕРЕСЕЧЕНИЕ**

Пересечением двух совместимых по типу отношений A и B называется отношение с тем же заголовком, как и в отношениях A и B , и с телом, состоящим из множества кортежей, принадлежащих одновременно обоим отношениям A и B .

$R \leftarrow A \text{ I } B$	$R := A \text{ intersect } B$
-------------------------------	-------------------------------

ПРИМЕР 3.7. Операция пересечения

Составить список городов, в которых детали хранятся на складе, а также крайней мере есть один поставщик.

$R \leftarrow \Pi_{\text{ГОРОД}}(\text{ПОСТАВЩИК}) \text{ I } \Pi_{\text{ГОРОД}}(\text{ДЕТАЛЬ})$

или

$R := \text{ПОСТАВЩИК}[\text{ГОРОД}] \text{ intersect } \text{ДЕТАЛЬ}[\text{ГОРОД}]$

Как и в предыдущем примере следует создать совместимые по типу отношения ДЕТАЛЬ и ПОСТАВЩИК, выполнив их проекцию по атрибуту Город. Затем для комбинирования новых полученных отношений следует использовать операцию пересечения. Результат выполнения всех этих действий приведен на рис.3.6.

R
ГОРОД
Воронеж
Москва

Рис. 3.6. Результат операции пересечения

• ДЕКАРТОВО ПРОИЗВЕДЕНИЕ

Декартово произведение двух отношений A и B , где A и B не имеют общих имен атрибутов, определяется как отношение с заголовком, который представляет собой сцепление двух заголовков исходных отношений A и B , и телом, состоящим из множества всех кортежей t , таких, что t представляет собой сцепление кортежа a , принадлежащего отношению A , и кортежа b , принадлежащего отношению B . Кроме того:

- o степень результирующего отношения равна сумме степеней исходных отношений;
- o мощность результирующего отношения равна произведению мощностей исходных отношений.

Исходные отношения могут содержать атрибуты с одинаковыми именами. В таком случае надо предварительно выполнить операцию переименования одного из атрибутов.

$R \leftarrow A \otimes B$	$R := A \text{ times } B$
----------------------------	---------------------------

Схема операции приведена на рис.3.7.

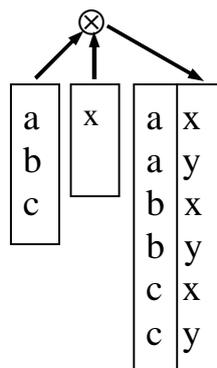


Рис. 3.7. ресечения

ПРИМЕР 8.8. Операция декартова произведения

Получить все такие пары «номера поставщиков-номера деталей», для которых выводимые поставщик и деталь размещены в одном городе.

$$R \leftarrow \Pi_{\text{ПНОМ, ДНОМ}} (\sigma_{\text{ПГОРОД} = \text{ГОРОД}} (\rho_{\text{ПНОМ, ПФАМ, СТАТУС, ПГОРОД}} (\text{ПОСТАВЩИК}) \otimes \text{ДЕТАЛЬ}))$$

или

$R = ((\text{ПОСТАВЩИК rename ГОРОД as ПГОРОД}) \text{ times ДЕТАЛЬ})$
 $\text{where ПГОРОД} = \text{ГОРОД}) [\text{ПНОМ}, \text{ДНОМ}]$

Здесь для того, чтобы в отношении, полученном в результате декартова произведения, не было атрибутов с одинаковым именем ГОРОД, в исходном отношении ПОСТАВЩИК этот атрибут переименован в ПГОРОД.

Иногда для обеспечения уникальности имен атрибутов в отношении, полученном как результат выполнения операции декартова произведения, используют названия отношений в качестве префиксов имен атрибутов. Тогда решение поставленной задачи может быть записано так:

$R := ((\text{ПОСТАВЩИК times ДЕТАЛЬ})$
 $\text{where ПОСТАВЩИК.ГОРОД} = \text{ДЕТАЛЬ.ГОРОД}) [\text{ПНОМ}, \text{ДНОМ}]$

или

$R \leftarrow \Pi_{\text{ПНОМ}, \text{ДНОМ}} (\sigma_{\text{ПОСТАВЩИК.ГОРОД} = \text{ДЕТАЛЬ.ГОРОД}} (\text{ПОСТАВЩИК} \otimes \text{ДЕТАЛЬ}))$

Результат этих действий приведен на рис. 3.8.

R

ПНОМ	ДНОМ
П1	Д1
П1	Д4
П1	Д6
П2	Д2
П2	Д5
П3	Д2
П3	Д5
П4	Д1
П4	Д4
П4	Д6

Рис. 3.8. Результат операции декартова произведения

3.2.4. Специальные реляционные операции

Как правило, пользователей интересует лишь некоторая часть всех комбинаций кортежей декартова произведения, коорая удовлетворяеи заданному условию. Поэтому вместо декартова произведения обычно используется одна из самых важных операций реляционной алгебры – операция соединения. В результате ее выполнения на базе двух исходных отношений создается некоторое новое отношение. Операция соединения является производной от операции декартова произведения, так как она эквивалентна операции выборки из декартова произведения тех кортежей, которые удовлетворяют условию, указанному в предикате соединения в качестве формулы выборки.

Рассмотрим различные типы операций соединения, которые несколько отличаются друг от друга и могут быть полезны в различных ситуациях.

- **ТЕТА-СОЕДИНЕНИЕ**

Пусть отношения A и B не имеют общих имен атрибутов и θ обозначает любой скалярный оператор сравнения ($=, <>, >, >=, <,<=$). Тогда тета-соединением отношения A по атрибуту X с отношением B по атрибуту Y называется отношение с тем же заголовком, что и при декартовом произведении отношений A и B , и с телом, содержащим множество кортежей t , таких что t принадлежит этому декартову произведению и вычисление условия " $X \theta Y$ " дает значение истина для этого кортежа. Атрибуты X и Y должны быть определены на одном и том же домене, а операция должна иметь смысл для этого домена.

$R \leftarrow A \underset{X\Theta Y}{><} B$	$R := A \text{ join } B \text{ on } X \Theta Y$
---	---

Обозначение тета-соединения можно переписать на основе базовых операций выборки и декартова произведения:

$$A \underset{X\Theta Y}{><} B = \sigma_{X\Theta Y}(A \otimes B)$$

Так же как и в случае с декартовым произведением, степень тета-соединения называется сумма степеней операндо-отношений A и B . Если θ обозначает "равно", то тета-соединение называется *эвисоедине-*

нием или соединением по эквивалентности. Еще раз обратимся к запросу, который рассматривался в примере 3.8 (см. пример 3.9).

ПРИМЕР 8.9. Операция эквисоединения

Получить все такие пары «номера поставщиков-номера деталей», для которых выводимые поставщик и деталь размещены в одном городе.

В примере 3.8 для получения этого списка использовались операции декартова произведения и выборки. Однако тот же самый результат можно получить с помощью операции эквисоединения.

$$R \leftarrow \Pi_{\text{ПНОМ, ДНОМ}} \left(\rho_{\text{ПНОМ, ПФАМ, СТАТУС, ПГОРОД}} (\text{ПОСТАВЩИК}) \right) \bowtie_{\text{ПГОРОД=ГОРОД}} \text{ДЕТАЛЬ}$$

или

$$R := ((\text{ПОСТАВЩИК} \text{ rename } \text{ГОРОД} \text{ as } \text{ПГОРОД}) \text{ join } \text{ДЕТАЛЬ} \text{ on } \text{ПГОРОД}=\text{ГОРОД}) [\text{ПНОМ, ДНОМ}]$$

Результат действий представлен на рис. 3.8.

Из определения следует, что результат эквисоединения должен включать два таких атрибута, значения которых должны быть равны в каждом кортеже отношения. Если один из этих атрибутов исключается (что может быть сделано операцией проекции), результатом будет просто естественное соединение.

• **ЕСТЕСТВЕННОЕ СОЕДИНЕНИЕ**

Естественным соединением называется соединение по эквивалентности двух отношений A и B, выполненное по всем общим атрибутам, из результатов которого исключается по одному экземпляру каждого общего атрибута.

Степенью естественного соединения называется сумма степеней операндов-отношений A и B за исключением количества общих атрибутов.

$R \leftarrow A \bowtie B$	$R := A \text{ join } B$
----------------------------	--------------------------

ПРИМЕР 3.10. Операция естественного соединения

Получить все такие пары «номера поставщиков-номера деталей», для которых выводимые поставщик и деталь размещены в одном городе.

Построить этот запрос, используя операцию естественного соединения, еще проще, чем в примерах 3.8, 3.9:

$$R \leftarrow \Pi_{\text{ПНОМ, ДНОМ}}(\text{ПОСТАВЩИК} \bowtie \text{ДЕТАЛЬ})$$

ИЛИ

$$R := (\text{ПОСТАВЩИК} \text{ join } \text{ДЕТАЛЬ})[\text{ПНОМ}, \text{ ДНОМ}]$$

Результат действий представлен на рис. 3.8.

• ДЕЛЕНИЕ

Пусть отношения A и B имеют заголовки

$\{X_1, X_2, \dots, X_m, Y_1, Y_2, \dots, Y_n\}$

и

$\{Y_1, Y_2, \dots, Y_n\}$

соответственно, т.е. атрибуты Y_j ($j=1, \dots, n$) - общие для двух отношений, и отношение A имеет дополнительные атрибуты X_i ($i=1, \dots, m$), а отношение B не имеет дополнительных атрибутов (отношения A и B представляют соответственно делимое и делитель). Предположим также, что соответствующие атрибуты (т.е. атрибуты с одинаковыми именами) определены на одном и том же домене. Пусть теперь выражения $\{X_i\}$ и $\{Y_j\}$ обозначают два составных атрибута X и Y соответственно. Тогда **делением** отношений A на B ($A \text{ divideby } B$) называется отношение с заголовком $\{X\}$ и телом, содержащим множество всех кортежей $\{X:x\}$, таких что существует кортеж $\{X:x, Y:y\}$, который принадлежит отношению A для всех кортежей $\{Y:y\}$, принадлежащих отношению B . Иначе это можно сформулировать так: результат содержит такие X -значения из отношения A , для которых соответствующие Y -значения из A включают все Y -значения из отношения B .

$R \leftarrow A \div B$	$R := A \text{ divideby } B$
-------------------------	------------------------------

ПРИМЕР 3.11. Операция деления отношений

Пусть

$$A(X_1, X_2, Y_1, Y_2) = \begin{bmatrix} i & o & r & k \\ o & r & c & a \\ c & k & r & k \\ i & o & c & a \\ c & k & c & a \\ i & o & k & c \end{bmatrix} \quad B(Y_1, Y_2) = \begin{bmatrix} c & a \\ r & k \end{bmatrix}$$

Получить **результат** $R := A \text{ divideby } B$

$$R(X_1, X_2) := \begin{bmatrix} i & o & r & k \\ o & r & c & a \\ c & k & r & k \\ i & o & c & a \\ c & k & c & a \\ i & o & k & c \end{bmatrix} \div \begin{bmatrix} c & a \\ r & k \end{bmatrix} = \begin{bmatrix} i & o \\ c & k \end{bmatrix}$$

Только выделенные кортежи в отношении A содержат в качестве значений атрибутов $\{Y_j\}$ все значения кортежей отношения B ($\langle i \ o \ r \ k \rangle$, $\langle i \ o \ c \ a \rangle$ и $\langle c \ k \ r \ k \rangle$, $\langle c \ k \ c \ a \rangle$). Поэтому частное содержит кортежи со значениями атрибутов $\{X_j\}$ $\langle i \ o \rangle$, $\langle c \ k \rangle$. Для кортежа $\langle o \ r \ c \ a \rangle$ в отношении-делителе нет соответствующего кортежа $\langle o \ r \ r \ k \rangle$ – поэтому значения $\langle o \ r \rangle$ не попали в частное.

Оператор деления полезен для запросов, которые на обычном языке включают слово «все».

ПРИМЕР 3.12. Операция деления отношений

Получить список имен поставщиков, поставляющих все детали.

$$R \leftarrow \Pi_{\text{ПФАМ}} \left(\left(\Pi_{\text{ПНОМ, ДНОМ}}(\text{ПД}) \div \Pi_{\text{ДНОМ}}(\text{ДЕТАЛЬ}) \right) \right) \succ \langle \text{ПОСТАВЩИК} \rangle$$

или

$$R := \left(\left(\text{ПД}[\text{ПНОМ, ДНОМ}] \text{ divideby } \text{ДЕТАЛЬ}[\text{ДНОМ}] \right) \right. \\ \left. \text{join } \text{ПОСТАВЩИК} \right) [\text{ПФАМ}]$$

Результат этих действий показан на рис. 3.9.

Рис. 3.9. Результат операции деления

3.2.5. Несколько замечаний по свойствам операций

Из рассмотренных в этом разделе операций реляционной алгебры к основным можно отнести объединение, разность, декартово произведение, проекцию и выборку. Остальные операции реляционной алгебры можно получить с помощью основных, но они имеют самостоятельное значение.

Пересечение отношений:

$$R \leftarrow A \text{ I } B = A - (A - B)$$

или

$$R := A \text{ intersect } B = A \text{ minus } (A \text{ minus } B)$$

Деление отношений:

$$R \leftarrow A \div B = \Pi_X(A) - \Pi_X((\Pi_X(A) \otimes B) - A)$$

или

$$R := A \text{ divideby } B = A[X] \text{ minus } ((A[X] \text{ times } B) \text{ minus } A) [X]$$

Естественное соединение:

$$R \leftarrow A \text{ >< } B = \Pi_{X,Y,Z} (\sigma_{A:Y=B:Y} (A \otimes B))$$

или

$$R := A \text{ join } B = (A \text{ times } B) \text{ where } (A : Y = B : Y) [X, Y, Z]$$

Здесь условие $A:Y=B:Y$ означает:

$$A:Y_1 = B:Y_1 \wedge A:Y_2 = B:Y_2 \wedge \dots \wedge A:Y_n = B:Y_n,$$

где $A:Y_1$ – имя столбца отношения $A \ddot{\wedge} B$, соответствующего столбцу Y_1 в отношении A ; $B:Y_1$ – имя столбца отношения $A \ddot{\wedge} B$, соответствующего столбцу Y_1 в отношении B .

Тета-соединение:

$$R \leftarrow A \underset{X \theta Y}{><} B = \sigma_{X \theta Y} (A \otimes B)$$

ИЛИ

$$R := A \text{ join } B \text{ on } (X\theta Y) = (A \text{ times } B) \text{ where } (X\theta Y)$$

3.3. Примеры

Приведем несколько примеров [14] использования реляционной алгебры для выражения словесных запросов в виде формулы. Они используют примерные данные базы поставщиков и деталей.

ПРИМЕР 3.13. Получить имена поставщиков, которые поставляют деталь Д2

$$R \leftarrow \Pi_{\text{ПФАМ}} (\sigma_{\text{ДНОМ} = \text{Д2}} (\text{ПД} \bowtie \text{ПОСТАВЩИК}))$$

ИЛИ

$$R := ((\text{ПД} \text{ join } \text{ПОСТАВЩИК}) \text{ where } \text{ДНОМ} = \text{Д2}') [\text{ПФАМ}]$$

ПРИМЕР 3.14. Получить имена поставщиков, которые поставляют по крайней мере одну красную деталь

$$R \leftarrow \Pi_{\text{ПФАМ}} (\Pi_{\text{ПНОМ}} (\sigma_{\text{ЦВЕТ} = \text{Красный}} (\text{ДЕТАЛЬ}) \bowtie \text{ПД}) \bowtie \text{ПОСТАВЩИК})$$

ИЛИ

$$R := (((\text{ДЕТАЛЬ} \text{ where } \text{ЦВЕТ} = \text{Красный}') \text{ join } \text{ПД}) [\text{ПНОМ}] \text{ join } \text{ПОСТАВЩИК}) [\text{ПФАМ}]$$

ПРИМЕР 3.15. Получить номера поставщиков, поставляющих по крайней мере все те детали, которые поставяет поставщик П2

$$R \leftarrow \Pi_{\text{ПНОМ}} (\Pi_{\text{ПНОМ}, \text{ДНОМ}} (\text{ПД}) \div \sigma_{\text{ПНОМ} = \text{П2}} (\text{ПД}))$$

ИЛИ

$$R := \text{ПД} [\text{ПНОМ}, \text{ДНОМ}] \text{ dividedby } (\text{ПД} \text{ where } \text{ПНОМ} = \text{П2}') [\text{ПНОМ}]$$

ПРИМЕР 3.16. Получить все пары номеров поставщиков, размещенных в одном городе

$$R \leftarrow \Pi_{\text{ПНОМ1, ПНОМ}} \left(\Pi_{\text{ПНОМ1, ГОРОД}} \left(\rho_{\text{ПНОМ1, ПФАМ, СТАТУС, ГОРОД}} (\text{ПОСТАВЩИК}) \right) \right) \bowtie_{\text{ПНОМ1 < ПНОМ}} \Pi_{\text{ПНОМ, ГОРОД}} (\text{ПОСТАВЩИК})$$

или

$$R := \left(\left(\text{ПОСТАВЩИК rename ПНОМ as ПНОМ1} \right) \left[\text{ПНОМ1, ГОРОД} \right] \right. \\ \left. \text{join} \quad \left(\text{ПОСТАВЩИК} \right) \left[\text{ПНОМ, ГОРОД} \right] \right) \\ \text{where} \\ \text{ПНОМ1 < ПНОМ} \left[\text{ПНОМ1, ПНОМ} \right]$$

ПРИМЕР 3.17. Получить имена поставщиков, которые не поставляют деталь Д2

$$R \leftarrow \Pi_{\text{ПФАМ}} \left(\left(\Pi_{\text{ПНОМ}} (\text{ПОСТАВЩИК}) - \Pi_{\text{ПНОМ}} (\sigma_{\text{ДНОМ} = \text{Д2}'} (\text{ПД})) \right) \right) \bowtie \text{ПОСТАВЩИК}$$

или

$$R := \left(\left(\text{ПОСТАВЩИК} \left[\text{ПНОМ} \right] \text{ minus} \left(\text{ПД where ДНОМ} = \text{Д2}' \right) \left[\text{ПНОМ} \right] \right) \right. \\ \left. \text{join} \text{ ПОСТАВЩИК} \right) \left[\text{ПФАМ} \right]$$

Разберем последний пример несколько подробнее, чтобы проиллюстрировать еще один момент. Не всегда просто представить себе, как выразить данный запрос в виде одной вложенной формулы (особенно если запрос сложный). Однако это и не обязательно. Вот пошаговое решение проблемы:

$$T1 \leftarrow \Pi_{\text{ПНОМ}} (\text{ПОСТАВЩИК}) \quad \text{или} \quad T1 := \text{ПОСТАВЩИК} \left[\text{ПНОМ} \right]$$

$$T2 \leftarrow \sigma_{\text{ДНОМ} = \text{Д2}'} (\text{ПД}) \quad \text{или} \quad T2 := \text{ПД where ДНОМ} = \text{'Д2}'$$

$$T3 \leftarrow \Pi_{\text{ПНОМ}} (T2) \quad \text{или} \quad T3 := T2 \left[\text{ПНОМ} \right]$$

$$T4 \leftarrow T1 - T3 \quad \text{или} \quad T4 := T1 \text{ minus } T3$$

$$T5 \leftarrow T4 \bowtie \text{ПОСТАВЩИК} \quad \text{или} \quad T5 := T4 \text{ join} \text{ ПОСТАВЩИК}$$

$$T6 \leftarrow \Pi_{\text{ПФАМ}} (T5) \quad \text{или} \quad T6 := T5 \left[\text{ПФАМ} \right]$$

Отношение $T6$ содержит необходимый результат.

4. Рекомендации студентам по подготовке к лабораторной работе с указанием литературы.

- изучить теоретические и справочно-информационные материалы;
- проделать приведенные примеры.

Литература.

- Дейт, К.Дж. Введение в системы баз данных / К.Дж.Дейт;пер.с англ.и ред.К.А.Птицына. — 8-е изд. — М. : Вильямс, 2005. — 1380с. : ил.
- Конноли Т., Бегг К. Базы данных. Проектирование, реализация и сопровождение. Теория и практика. – 3-е изд. – М.: Издательский дом "Вильямс", 2003. – 1440 с.: ил.
- Астахова, И.Ф. Информационные системы : Учеб.пособие / И.Ф.Астахова,А.С.Потапов,В.А.Чулюков,А.Н.Журбин;ВГПУ. — Воронеж : Б.и., 2002. — 148с.

5. Описание экспериментальных установок (лабораторного оборудования).

Лабораторная выполняется на компьютерах, объединенных в локальную сеть и имеющих доступ к Интранету каф. информатики.

На сервере должна быть установлена СУБД Oracle версии 8 или более поздней.

В Oracle должна быть создана учебная база данных:

ПРЕДПРИЯТИЕ

Пред#	Название	Рейтинг	Город
180	Электроника	230	Воронеж
230	Гормолзавод	300	Москва
150	Сельмаш	140	Воронеж
190	Хлебозавод	300	Курск
270	Рудгормаш	240	Москва

где

Пред# – номер предприятия, номер обций по некоторым группам городов;

Название – название предприятия;

Рейтинг – рейтинг предприятия по некоторым показателям;

Город – город, в котором находится предприятие.

ПРОДУКЦИЯ

Прод#	Наименование	Количество	Город
10	Магнитофоны	12000	Воронеж
20	Кровати	15000	Москва
30	Тракторы	20000	Воронеж
40	Кухни	30000	Орел
50	Продукты	10000	Воронеж

где

Прод# – номер продукции;

Наименование – наименование продукции;

Количество – стоимость продукции, выпускаемой в год в данном городе;

Город – город, в котором указанная продукция выпускается.

ЛИЧНОСТЬ

Лич#	Фамилия	Город	День_рожд	Пред#
55	Иванов	Воронеж	15.03.02	180
10	Петров	Москва	17.02.95	230
100	Сидоров	Воронеж	03.12.93	150
190	Иванов	Курск	18.04.91	190

где

Лич# – номер личности;

Фамилия – фамилия человека;

Город – город проживания;

День_рожд – дата рождения данного человека;

Пред# – номер предприятия, где работает данная личность.

ПРЕД_ПРОД

Пред#	Прод#	Год	Выработка
150	30	2000	150
180	10	2000	100
190	50	2001	50
230	50	2001	120
270	20	2002	50

где

Пред# – номер предприятия;

Прод# – номер продукции;

Год – год выпуска продукции;

Выработка (тыс.руб) – количество продукции данного предприятия.

6. Краткое содержание работы, выполняемой студентами в ходе занятия.

Студенты формируют на ЭВМ запросы к учебной базе данных на языке реляционной алгебры, используя программу по ссылке: <http://impm.vspu.ac.ru/~ghost/relop/>

Упражнения по запросам

В следующих упражнениях предлагается по словесному запросу к базе данных предприятий составить соответствующее алгебраическое выражение, необходимое для этого запроса.

- 1) Получить названия предприятий, производящих продукцию с номером 30.
- 2) Получить имена предприятий, производящих продукцию всех сортов.
- 3) Получить номера предприятий, производящих по крайней мере ту продукцию, которую выпускает предприятие с номером 190.
- 4) Получить пары предприятий, находящихся в одном городе.
- 5) Получить имена предприятий, не производящих продукцию с номером 50.
- 6) Выбрать информацию обо всех предприятиях.
- 7) Выбрать название продукции, у которой количество потребления в городе находится в диапазоне от 12000 до 15000.
- 8) Выбрать название предприятий, у которых выработка продукции в 2001 г. на единицу работающего составила 100 тыс. руб.
- 9) Выбрать название предприятий, которые производят заданную продукцию.
- 10) Выбрать все пары названий городов, для которых предприятие из первого города, а интересующая продукция во втором городе.
- 11) Выбрать фамилии людей, которые работают на предприятии с заданным названием.
- 12) Выбрать фамилии людей, у которых город проживания совпадает с городом нахождения предприятия.
- 13) Определить фамилии людей, начинающиеся на «И», работающих на предприятии с заданным названием предприятия.
- 14) Определить название предприятий, производящих продукцию с заданным номером в заданном году.
- 15) Определить название продукции с заданным номером, имеющей выработку на единицу работающего > 100 тыс. руб.
- 16) Определить номера предприятий из Воронежа с рейтингом выше 20.
- 17) Найти пары номеров личностей, в которых оба человека работает на одном предприятии.
- 18) Определить имена предприятий, производящих продукцию с заданным номером.
- 19) Определить номера предприятий, имеющих в списке работающих по крайней мере одного «Иванов».

- 20) Найти имена предприятий, производящих по крайней мере одну продукцию с номером 50.
- 21) Выбрать имена предприятий, производящих все виды продукции.
- 22) Определить название предприятий, которые не производят продукцию с номером 50.
- 23) Определить номера продукции, производящих по крайней мере все виды продукции, производимые предприятием с номером 270.
- 24) Получить номера продукции, которая имеет количество более 15000, либо производится предприятием с номером 270, либо то и другое.
- 25) Найти все пары номеров предприятий, в которых два предприятия находятся в одном городе.

Список и условия упражнений можно взять по ссылке:

http://impm.vspu.ac.ru/~chul/Inf_Syst/1.5.htm

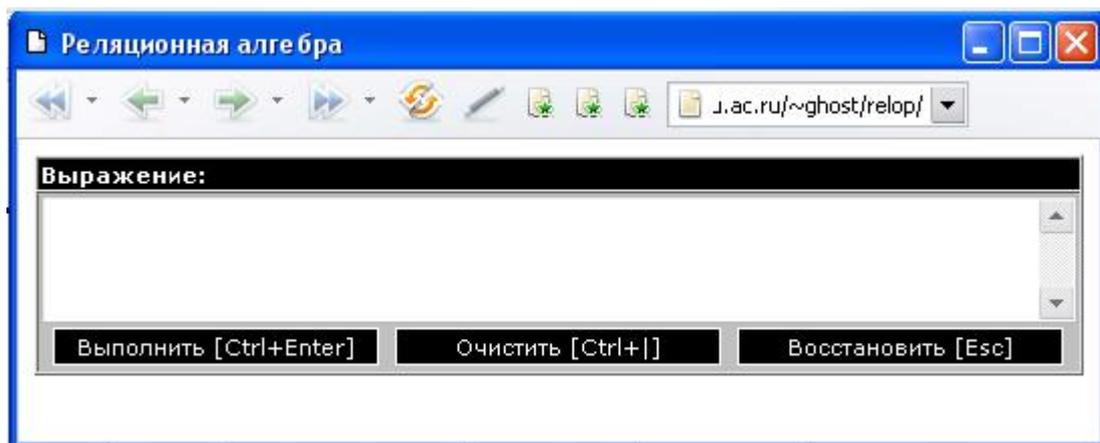
7. Порядок проведения эксперимента, постановки опыта, снятия замеров и обработки данных эксперимента.

Рассмотрим порядок работы на примере упражнения №1.

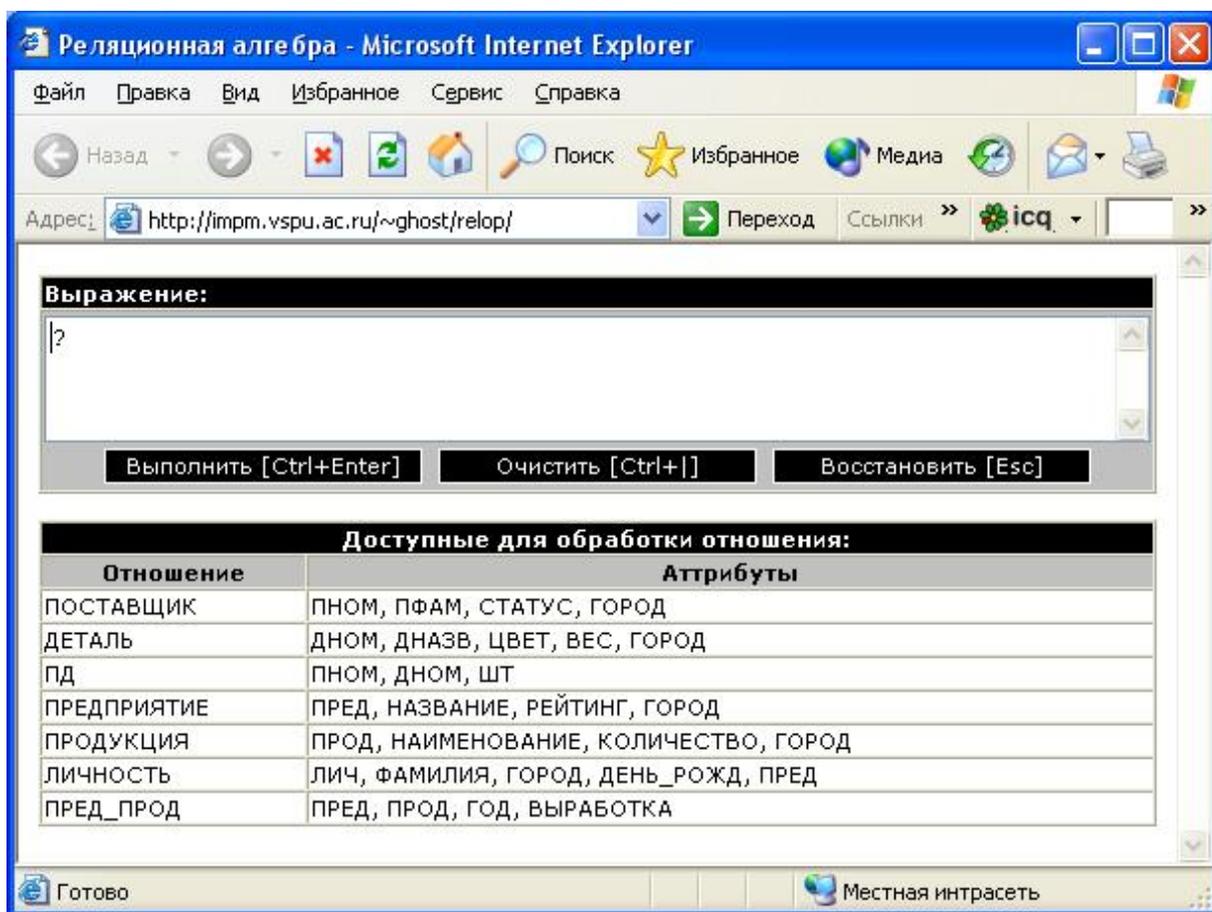
Получить названия предприятий, производящих продукцию с номером 30.

- 1) Зайти на страницу по ссылке

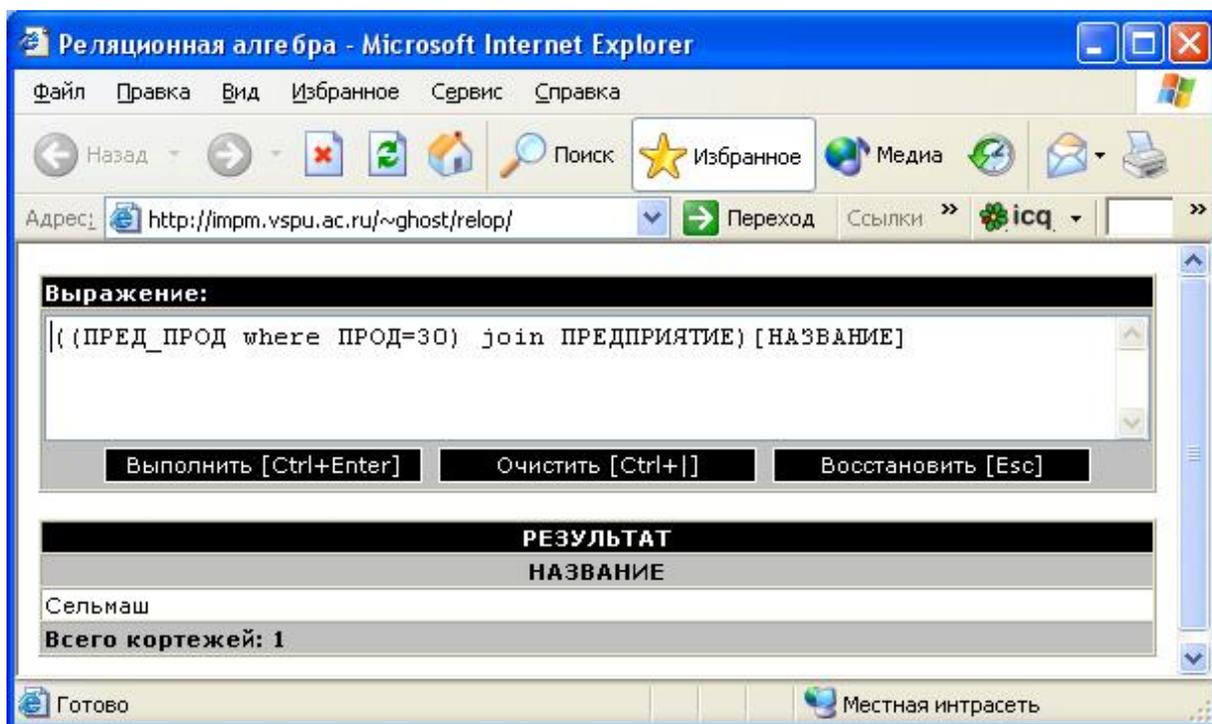
<http://impm.vspu.ac.ru/~ghost/relop/>



- 2) Ввести в поле «Выражение» знак вопроса и нажать Ctrl+Enter для получения состава отношений и их атрибутов:



- 3) Ввести в поле «Выражение» запрос вопроса и нажать Ctrl+Enter для получения ответа (имена отношений и атрибутов можно копировать из поля «Доступные для обработки отношения»):



8. Техника безопасности.

Определяется правилами техники безопасности при работе с компьютером.

9. Исходные данные для работы.

Учебная база данных (п.5).

10. Методика анализа полученных результатов.

Сравнить полученные результаты с исходными данными.

11. Порядок оформления отчета по лабораторной работе и его защиты.

Отчет должен содержать условия упражнений, запрос и результат запроса. При защите лабораторной надо уметь объяснить содержание запроса.

ЛАБОРАТОРНАЯ РАБОТА № 2

Тема: Реляционные операторы: реляционное исчисление

Продолжительность 7 часов

1. Целью лабораторных работ является знакомство с математической основой языков манипулирования данными в реляционных информационных системах, а именно – с реляционным исчислением.

2. Вопросы, подлежащие исследованию, связаны с изучением базовых теоретико-множественных операций реляционного исчисления доменов и реляционного исчисления кортежей.

3. Теоретические и справочно-информационные материалы.

Могут использоваться из Internet по ссылкам:

http://impm.vspu.ac.ru/~chul/Inf_Syst/index.htm

http://www.vspu.ac.ru/~chul/Inf_Syst/index.htm

3.1. Введение

Реляционное исчисление основывается на механизме исчисления предикатов первого порядка. *Реляционное исчисление* – это система обозначений для получения необходимого отношения в терминах данных отношений. Реляционная алгебра и реляционное исчисление отличаются только внешне, на самом деле они эквивалентны. Однако языки исчисления – это не процедурные языки, поскольку их средствами можно выразить все, что необходимо и необязательно указывать, как это получить. Выражение в исчислении описывает лишь свойства желаемого результата, фактически не указывая, как его получить. Выражения реляционной алгебры, напротив, определяют конкретный порядок выполнения операций.

Основными понятиями исчисления являются понятие *переменной* с некоторой областью допустимых значений и понятие *правильно построенной формулы* (WFF-well formulated formula), опирающейся на предикаты, переменные и кванторы.

В зависимости от области определения переменной различают *исчисление кортежей* и *исчисление доменов*. В **исчислении кортежей** областью определения переменных являются отношения базы данных, т.е. допустимым значением каждой переменной является кортеж некоторого отношения. В **исчислении доменов** областью определения переменных являются домены, на которых определены атрибуты отношений базы данных, то есть допустимым значением каждой переменной является значение некоторого домена.

3.2. Реляционное исчисление с переменными кортежей

Переменная кортежа (или области значений) определяется с помощью синтаксиса, предложенного в [14] и являющегося синтаксисом языка QUEL, следующим образом:

$$\text{RANGE OF } T \text{ IS } X_1, X_2, \dots, X_n.$$

Здесь T – определяемая переменная кортежа, а X_i ($i = 1, 2, \dots, n$) – либо имя отношения, либо выражение исчисления кортежей. Пусть X_i является отношением R_i ($i = 1, 2, \dots, n$). Отношения R_1, R_2, \dots, R_n должны быть совместимы по типу, т.е. они должны иметь идентичные заголовки.

Тогда переменная кортежа T изменяется на объединении этих отношений, т.е. ее значение в любое заданное время будет некоторым текущим кортежем по крайней мере одного из этих отношений. Конечно, если список идентификаторов выражений будет просто одним именованным отношением R (это обычный случай), то переменная кортежа будет просто принимать значения текущих кортежей одного такого отношения R .

При использовании кортежных переменных в формулах можно ссылаться на значение атрибута переменной (это аналогично тому, как, например, при программировании на языке Pascal можно сослаться на значение поля переменной типа записи). Например, для того, чтобы сослаться на значение атрибута A отношения, значения которого принимает переменная T , нужно употребить конструкцию $T.A$.

Если компоненты кортежа могут быть идентифицированы по их порядковой позиции, то можно ссылаться на них с помощью индексной ссылки $T[i]$.

Правильно построенная формула WFF служит для выражения условий, накладываемых на кортежные переменные. WFF состоит из простых сравнений скалярных значений (значений атрибутов переменных или литерально заданных констант). Более сложные варианты WFF строятся с помощью логических операций NOT, AND, OR, IF...THEN, и двух кванторов EXISTS и FORALL. Квантор EXISTS называется *квантором существования*, а квантор FORALL – *квантором общности*.

Если f – формула WFF, в которой участвует переменная x , то

$EXISTS x (f)$ и $FORALL x (f)$

являются допустимыми формулами WFF. Первая формула означает: «Существует **по крайней мере одно значение** переменной x , что вычисление формулы f для этого x дает значение истина». Вторая формула означает: «**Для всех значений** переменной x вычисление формулы f дает значение истина».

Квантор существования EXISTS определяется формально как повторяющееся OR (ИЛИ). То есть, если R – это отношение с кортежами T_1, T_2, \dots, T_m ; T – это переменная кортежа, которая изменяется на этом отношении; а $f(T)$ – это формула, в которой используется переменная T , то формула

$EXISTS T (f(T))$

определяется равносильно следующей формуле WFF:

$false \text{ OR } (f(T1)) \text{ OR } \dots \text{ OR } (f(Tm))$.

Пример. Пусть используется отношение R:

$R(D, F, W)$

2 3 4

2 3 5

2 4 5

Тогда следующие выражения будут иметь приведенные ниже значения:

EXISTS T ($T.W > 2$) : *true*

EXISTS T ($T.F > 4$) : *false*

EXISTS T ($T.D > 2$ OR $T.W \geq 5$) : *true*

Квантор существования FORALL определяется как повторяющееся AND (I). Другими словами, если R , T и $f(T)$ такие же, как рассматривались выше, то формула

$\text{FORALL } T(f(T))$

определяется равносильно следующей формуле:

$true \text{ AND } (f(T1)) \text{ AND } \dots \text{ AND } (f(Tm))$.

Пример. Если отношение R содержит те же самые кортежи, что и в предыдущем примере, то приведенные ниже выражения будут иметь такие значения:

FORALL T ($T.D > 2$) : *false*

FORALL T ($T.F > 2$) : *true*

FORALL T ($T.D = 2$ AND $T.W > 3$) : *true*

Переменные, входящие в WFF, могут быть *свободными* или *связанными*. Все переменные, входящие в WFF, при построении которой не использовались кванторы, являются **свободными**. Фактически, это означает, что если для какого-то набора значений свободных кортежных переменных при вычислении WFF получено значение *true*, то эти значения кортежных переменных могут входить в результирующее отношение. Пусть f – формула WFF, в которой переменная x свободна. Если имя переменной x

использовано сразу после квантора при построении WFF вида EXISTS $x (f)$) или FORALL $x (f)$, то в этой WFF и во всех WFF, построенных с ее участием, x – это **связанная** переменная. Это означает, что такая переменная не видна за пределами минимальной WFF, связавшей эту переменную. При вычислении значения такой WFF используется не одно значение связанной переменной, а вся ее область определения.

Пусть T и K – две кортежные переменные, определенные на отношении R . Тогда, WFF EXISTS $K (T.A > K.A)$ для текущего кортежа переменной T принимает значение *true* в том и только в том случае, если во всем отношении R найдется кортеж (связанный с переменной K) такой, что значение его атрибута A удовлетворяет внутреннему условию сравнения. Формула WFF FORALL $K (T.A > K.A)$ для текущего кортежа переменной T принимает значение *true* в том и только в том случае, если для всех кортежей отношения R (связанных с переменной K) значения атрибута A удовлетворяют условию сравнения.

Таким образом, кванторы в реляционном исчислении играют ту же роль, что декларации в языке программирования. Понятие свободной переменной аналогично понятию глобальной переменной, описанной вне текущей процедуры. Понятие связанной переменной аналогично понятию локальной переменной, описанной в текущей процедуре.

Итак, WFF обеспечивают средства формулировки условия выборки из отношений БД. Чтобы можно было использовать исчисление для реальной работы с БД, требуется еще один компонент, который определяет набор и имена столбцов результирующего отношения. Этот компонент называется *целевым списком (target_list)*.

Целевой список строится из целевых элементов, каждый из которых может иметь следующий вид:

- $T.A [AS X]$, где T – имя свободной переменной соответствующей WFF, а A – имя атрибута отношения, на котором определена переменная T , а X – это имя атрибута результата вычисления элементов целевого списка.
- T , что эквивалентно наличию подписка $T.A1, T.A2, \dots, T.An$, где $A1, A2, \dots, An$ включает имена всех атрибутов определяющего отношения;
- $N = T.A$; N – новое имя соответствующего атрибута результирующего отношения.

Последний вариант требуется в тех случаях, когда в WFF используются несколько свободных переменных с одинаковой областью определения.

Выражением реляционного исчисления кортежей называется конструкция вида TARGET_LIST WHERE WFF. Значением выражения является отношение, тело которого определяется WFF, а набор атрибутов и их имена – целевым списком.

Как указывалось выше, для описания реляционного исчисления кортежей использован синтаксис реального языка запросов QUEL.

Если использовать традиционный синтаксис языка предикатов, то описание реляционного исчисления с переменными кортежей выглядит следующим образом.

Выражение записывается в виде:

$$\{t \mid \Psi(t)\}$$

где t – единственная свободная переменная, обозначающая кортеж фиксированной длины (если необходимо указать арность кортежа, то используют запись $t^{(i)}$; i – арность кортежа t); Ψ - правильно построенная формула (WFF).

На рис. 3.1 представлен обзор рассмотренных элементов языка QUEL и их эквиваленты из языка предикатов.

Элемент языка	Язык QUEL	Язык предикатов
Конъюнкция	AND	\wedge
Дизъюнкция	OR	\vee
Отрицание	NOT	\neg
Импликация	IF...THEN	\rightarrow
Квантор существования	EXISTS	\exists
Квантор общности	FORALL	"

Рис. 3.1. Элементы синтаксиса языка QUEL и языка предикатов

Для основных операций реляционной алгебры укажем следующие соответствующие выражения (на языке QUEL и языке предикатов) реляционного исчисления на переменных кортежей.

1. Операции объединения ($A \cup B$ или A union B) соответствуют выражения:

$$\{t \mid A(t) \vee B(t)\} \quad \text{или} \quad \text{RANGE OF } t \text{ IS } A, B$$

Необходимо получить множество всех кортежей t , причем таких кортежей, которые принадлежат отношениям A или B .

2. Операции разности ($A - B$ или A minus B) соответствуют выражения:

$$\{t \mid A(t) \wedge (\neg B(t))\} \quad \text{или} \quad \begin{array}{l} \text{RANGE OF } t \text{ IS } A; \\ \text{RANGE OF } t \text{ IS } B; \\ t \text{ WHERE } t \neq r \end{array}$$

$$\{t \mid A(t) \wedge (\neg B(t))\} \quad \text{или} \quad \begin{array}{l} \text{RANGE OF } t \text{ IS } A; \\ \text{RANGE OF } r \text{ IS } B; \\ t \text{ WHERE } t \neq r \end{array}$$

$$\{t \mid A(t) \wedge (\neg B(t))\}$$

Рассматривается все множество кортежей t , такое, что t принадлежит A и не принадлежит B .

3. Операции декартового произведения ($A \otimes B$ или A times B) соответствуют выражения:

$$\{t^{(k+m)} \mid (\$u)(\$v)(A(u) \dot{\cup} B(v) \dot{\cup} \quad t[1]=u[1] \dot{\cup} \dots \dot{\cup} \\ t[k]=u[k] \dot{\cup} \\ \dot{\cup} t[k+1]=v[1] \dot{\cup} \dots \dot{\cup} t[k+m]=v[m])\}$$

ИЛИ

$$\begin{array}{l} \text{RANGE OF } t \text{ IS } A; \\ \text{RANGE OF } u \text{ IS } A; \\ \text{RANGE OF } r \text{ IS } B; \\ \text{RANGE OF } v \text{ IS } B; \\ (t,r) \text{ WHERE EXISTS } u \text{ EXISTS } v (t[1]=u[1] \text{ AND } \dots \\ \text{AND } t[k]=u[k] \text{ AND } r[1]=v[1] \text{ AND } \dots \text{ AND } r[m]=v[m]) \end{array}$$

Рассматривается все множество кортежей арности $k+m$ (k – арность отношения A , m – арность отношения B), таких, что существует кортеж u , принадлежащий A , и существует кортеж v , принадлежащий B , причем k первых компонентов каждого результирующего кортежа образуют компоненты кортежа u , а следующие m компонентов результирующего кортежа образуют компоненты кортежа v .

4. Операции проекции $(\pi_{i_1, i_2, \dots, i_n}(A))$ или $A[i_1, i_2, \dots, i_n]$ соответствуют выражения:

$$\{t^{(n)} \mid (\exists u)(A(u) \dot{\cup} t[1] = u[i_1] \dot{\cup} \dots \dot{\cup} t[n] = u[i_n])\}$$

или

RANGE OF u IS A ;

t WHERE EXISTS u ($t[1] = u[i_1]$ AND ... AND $t[n] = u[i_n]$)

5. Операции выборки $(\sigma_{X \theta Y}(A))$ или A where $X \theta Y$ соответствуют выражения:

$$\{t \mid A(t) \wedge X' \theta Y'\} \quad \text{или} \quad \text{RANGE OF } t \text{ IS } A;$$

t WHERE $X' \theta Y'$

где $X' \theta Y'$ – это условие $X \theta Y$, в котором каждый операнд, обозначающий компонент i , заменен на $t[i]$.

Как и в реляционной алгебре, остальные операции можно получить с помощью приведенных пяти основных операций.

3.3. Примеры

В примерах подразумевается, что действуют следующие переменные области значений:

RANGE OF ПА IS ПОСТАВЩИК;

RANGE OF ДА IS ДЕТАЛЬ;

RANGE OF ПДА IS ПД;

RANGE OF ПДБ IS ПД;

RANGE OF ПДВ IS ПД

ПРИМЕР 3.1. Получить номера поставщиков из Москвы со статусом больше 20

$$\{t^{(1)} \mid (\$ u) (\text{ПОСТАВЩИК}(u) \hat{\cup} t[1] = u[\text{ПНОМ}] \hat{\cup} u[\text{ГОРОД}] = \\ = \text{'Москва'} \hat{\cup} u[\text{СТАТУС}] > 20)\}$$

или

**(ПА.ПНОМ) WHERE ПА.ГОРОД='Москва' AND
ПА.СТАТУС>20**

ПРИМЕР 3.2. Получить имена поставщиков, которые поставляют деталь Д2

$$\{t^{(1)} \mid (\$ u) (\$ v) (\text{ПД}(u) \hat{\cup} \text{ПОСТАВЩИК}(v) \hat{\cup} t[1] = v[\text{ПФАМ}] \hat{\cup} \\ \hat{\cup} u[\text{ПНОМ}] = v[\text{ПНОМ}] \hat{\cup} u[\text{ДНОМ}] = \text{'Д2'})\}$$

или

**(ПА.ПФАМ) WHERE EXISTS ПДА (ПДА.ПНОМ =
ПА.ПНОМ
AND ПДА.ДНОМ = 'Д2')**

ПРИМЕР 3.3. Получить имена поставщиков, которые поставляют по крайней мере одну красную деталь

$$\{t^{(1)} \mid (\$ u) (\text{ПОСТАВЩИК}(u) \hat{\cup} t[1] = u[\text{ПФАМ}] \hat{\cup} (\$ v) (\text{ПД}(v) \hat{\cup} \\ \hat{\cup} u[\text{ПНОМ}] = v[\text{ПНОМ}] \hat{\cup} (\$ w) (\text{ДЕТАЛЬ}(w) \hat{\cup} \\ \hat{\cup} w[\text{ДНОМ}] = v[\text{ДНОМ}] \hat{\cup} w[\text{ЦВЕТ}] = \text{'Красный'}))\}$$

или

**(ПА.ПФАМ) WHERE EXISTS ПДА (ПА.ПНОМ =
ПДА.ПНОМ
AND EXISTS ДА(ДА.ДНОМ = ПДА.ДНОМ AND
ДА.ЦВЕТ = 'Красный'))**

ПРИМЕР 3.4. *Получить имена поставщиков, которые поставляют по крайней мере одну деталь, поставляемую поставщиком П2*

$$\{t^{(1)} \mid (\$ u) (\text{ПОСТАВЩИК}(u) \dot{\cup} t[1] = u[\text{ПФАМ}] \dot{\cup} (\$ v) (\$ w) \dot{\cup} (\text{ПД}(v) \dot{\cup} \text{ПД}(w) \dot{\cup} u[\text{ПНОМ}] = v[\text{ПНОМ}] \dot{\cup} v[\text{ДНОМ}] = w[\text{ДНОМ}] \dot{\cup} w[\text{ДНОМ}] = \text{'П2'})))\}$$

или

**(ПА.ПФАМ) WHERE EXISTS ПДА (EXISTS ПДБ
(ПА.ПНОМ = ПДА.ПНОМ AND
ПДА.ДНОМ = ПДБ.ДНОМ AND
ПДБ.ДНОМ = 'П2'))**

ПРИМЕР 3.5. *Получить имена поставщиков, которые поставляют все детали*

$$\{t^{(1)} \mid (\$ u) (\text{ПОСТАВЩИК}(u) \dot{\cup} t[1] = u[\text{ПФАМ}] \dot{\cup} (" v) (\$ w) \dot{\cup} (\text{ДЕТАЛЬ}(v) \dot{\cup} \text{ПД}(w) \dot{\cup} w[\text{ПНОМ}] = u[\text{ПНОМ}] \dot{\cup} w[\text{ДНОМ}] = v[\text{ДНОМ}]))\}$$

или

**(ПА.ПФАМ) WHERE FORALL ДА (EXISTS ПДА
(ПДА.ПНОМ = ПА.ПНОМ AND
ПДА.ДНОМ = ДА.ДНОМ))**

ПРИМЕР 3.6. *Получить имена поставщиков, которые не поставляют деталь Д2*

$$\{t^{(1)} \mid (\$ u) (\$ v) (\text{ПД}(u) \dot{\cup} \text{ПОСТАВЩИК}(v) \dot{\cup} t[1] = v[\text{ПФАМ}] \dot{\cup}$$

$$\{t^{(1)} \mid (\exists u) (\exists v) (\text{ПОСТАВЩИК}(u) \wedge \text{ПД}(v) \wedge t[1] = u[\text{ПНОМ}] \wedge$$

$$\wedge (\exists w) (v[\text{ПНОМ}] = \text{'П2'} \wedge (\exists w) (\text{ПД}(w) \wedge$$

$$\wedge w[\text{ПНОМ}] = u[\text{ПНОМ}] \wedge$$

$$\wedge w[\text{ДНОМ}] = v[\text{ДНОМ}])))) \}$$

или

(ПА.ПФАМ) WHERE NOT EXISTS ПДА
(ПДА.ПНОМ = ПА.ПНОМ AND ПДА.ДНОМ = 'Д2'
)

ПРИМЕР 3.7. *Получить номера поставщиков, поставляющих по крайней мере все те детали, которые поставяет поставщик П2*

$$\{t^{(1)} \mid (\exists u) (\exists v) (\text{ПОСТАВЩИК}(u) \wedge \text{ПД}(v) \wedge t[1] = u[\text{ПНОМ}] \wedge$$

$$\wedge (\exists w) (v[\text{ПНОМ}] = \text{'П2'} \wedge (\exists w) (\text{ПД}(w) \wedge$$

$$\wedge w[\text{ПНОМ}] = u[\text{ПНОМ}] \wedge$$

$$\wedge w[\text{ДНОМ}] = v[\text{ДНОМ}]))) \}$$

или

ПА.ПНОМ WHERE FORALL ПДБ (IF ПДБ.ПНОМ = 'П2'
THEN

EXISTS ПДВ (ПДВ.ПНОМ = ПА.ПНОМ AND
ПДВ.ДНОМ = ПДБ.ДНОМ))

ПРИМЕР 3.8. *Получить номера деталей, которые или весят более 16, или поставяются поставщиком П2, или и то и другое.*

$$\{t^{(1)} \mid (\exists u) (\text{ДЕТАЛЬ}(u) \wedge t[1] = u[\text{ДНОМ}] \wedge u[\text{ВЕС}] > 16 \vee$$

$$\vee (\exists v) (\text{ПД}(v) \wedge v[\text{ДНОМ}] = u[\text{ДНОМ}] \wedge$$

$$\wedge v[\text{ПНОМ}] = \text{'П2'})) \}$$

или

ДА.ДНОМ WHERE ДА.ВЕС > 16 OR

EXISTS ПДА (ПДА.ДНОМ = ДА.ДНОМ AND
ПДА.ПНОМ = 'П2')

3.4. Реляционное исчисление с переменными доменов

Реляционное исчисление, ориентированное на домены (или исчисление доменов), отличается от исчисления кортежей тем, что в нем используются переменные доменов вместо переменных кортежей, т.е. переменные, принимающие свои значения в пределах домена, а не отношения.

Основным формальным отличием исчисления доменов от исчисления кортежей является наличие дополнительного набора предикатов, позволяющих выражать так называемые **условия членства**. Если R - это n -арное отношение с атрибутами t_1, t_2, \dots, t_n , то условие членства имеет вид

$$R(\textit{pair}, \textit{pair}, \dots),$$

где каждая пара \textit{pair} имеет вид $t:v$, при этом v – это либо литерально задаваемая константа, либо имя доменной переменной. Условие членства принимает значение *true* в том и только в том случае, если в отношении R существует кортеж, содержащий значения указанных атрибутов. Если v – константа, то на атрибут t задается жесткое условие, не зависящее от текущих значений доменных переменных; если же v – имя доменной переменной, то условие членства может принимать разные значения при разных значениях этой переменной. Например, вычисление выражения

$$\text{ПД (ПНОМ:'П1', ДНОМ:'Д1')}$$

дает значение *true*, если и только если в отношении ПД существует кортеж со значением ПНОМ, равным П1, и значением ДНОМ, равным Д1. Аналогично, условие членства

$$\text{ПД (ПНОМ:ПНОМА, ДНОМ:ДНОМА)}$$

принимает значение *true*, если и только если в отношении ПД существует кортеж со значением ПНОМ, эквивалентным текущему значению переменной домена ПНОМА (какому бы то ни было), и значением ДНОМ, эквивалентным текущему значению переменной домена ДНОМА (опять же какому бы то ни было).

Во всех остальных отношениях формулы и выражения исчисления доменов выглядят похожими на формулы и выражения исчисления кортежей. В частности, конечно, различаются свободные и связанные вхождения доменных переменных.

Далее будем считать, что существуют переменные доменов с именами, образуемыми добавлением цифр 1, 2, 3, ... к соответствующим именам доменов. Кроме того предполагается, что в базе данных поставщиков и деталей каждый атрибут имеет такое же имя, как и соответствующий ему домен, за исключением атрибутов ПФАМ и ДНАЗВ, для которых соответствующий домен называется просто ИМЯ.

Например, выражение
 (ПНОМ1) WHERE ПОСТАВЩИК (ПНОМ:ПНОМА,
 ГОРОД: 'Воронеж')

означает подмножество всех номеров поставщиков из города Воронеж.

С использованием традиционного синтаксиса языка предикатов реляционное исчисление с переменными доменов имеет вид:

$$\{x_1x_2\dots x_k \mid \Psi(x_1, x_2, \dots, x_k)\},$$

где Ψ — формула, обладающая тем свойством, что только ее свободные переменные доменов являются различными переменными x_1, x_2, \dots, x_k .

Реляционное исчисление доменов является основой большинства языков запросов, основанных на использовании форм. В частности, на этом исчислении базируется известный язык QBE (Query-by-Example), который был первым (и наиболее интересным) языком в семействе языков, основанных на табличных формах.

3.5. Примеры

Приведем некоторые рассмотренные выше примеры, но выраженные в терминах исчисления доменов.

ПРИМЕР 3.9. *Получить номера поставщиков из Москвы со статусом больше 20*

{ПНОМ1 | (∃ СТАТУС1) (СТАТУС1>20 ∧
 ∧ ПОСТАВЩИК(ПНОМ1, СТАТУС1, ГОРОД) ∧
 ∧ ГОРОД='Москва')}

или

ПНОМ1 WHERE EXISTS СТАТУС1
 (СТАТУС1>20 AND ПОСТАВЩИК (ПНОМ:ПНОМ1,

СТАТУС:СТАТУС1, ГОРОД:'Москва'))

ПРИМЕР 3.10. *Получить имена поставщиков, которые поставляют по крайней мере одну красную деталь*

{ИМЯ1 | (\$ ПНОМ1) (\$ ДНОМ1)
(ПОСТАВЩИК(ПНОМ1, ИМЯ1) Ù
Ù ПД (ПНОМ1, ДНОМ1) Ù
Ù ДЕТАЛЬ (ДНОМ1, ЦВЕТ) Ù
Ù ЦВЕТ = 'Красный')}

ИЛИ

ИМЯ1 WHERE EXISTS ПНОМ1 EXISTS ДНОМ1
(ПОСТАВЩИК (ПНОМ:ПНОМ1, ПФАМ:ИМЯ1)
AND ПД (ПНОМ: ПНОМ1, ДНОМ: ДНОМ1)
AND ДЕТАЛЬ (ДНОМ:ДНОМ1, ЦВЕТ:
'Красный'))

ПРИМЕР 3.11. *Получить имена поставщиков, которые поставляют по крайней мере одну деталь, поставляемую поставщиком П2*

{ИМЯ1 | (\$ ПНОМ1) (\$ ДНОМ1)
(ПОСТАВЩИК(ПНОМ1, ИМЯ1) Ù
Ù ПД (ПНОМ1, ДНОМ1) Ù
Ù ПД (ПНОМ, ДНОМ1) Ù
Ù ПНОМ = 'П2')}

ИЛИ

ИМЯ1 WHERE EXISTS ПНОМ1 EXISTS ДНОМ1
(ПОСТАВЩИК (ПНОМ:ПНОМ1, ПФАМ:ИМЯ1)
AND ПД (ПНОМ:ПНОМ1, ДНОМ:ДНОМ1)

AND ПД (ПНОМ:'П2', ДНОМ:ДНОМ1))

ПРИМЕР 3.12. *Получить имена поставщиков, которые поставляют все детали*

{ИМЯ1 | (\$ ПНОМ1) (ПОСТАВЩИК (ПНОМ1, ИМЯ1) Ù Ù(" ДНОМ1)(ДЕТАЛЬ(ДНОМ1) ® ПД (ПНОМ1, ДНОМ1))))}

или

**ИМЯ1 WHERE EXISTS ПНОМ1 (ПОСТАВЩИК
(ПНОМ:ПНОМ1, ПФАМ: ИМЯ1)
AND FORALL ДНОМ1 (IF ДЕТАЛЬ (ДНОМ: ДНОМ1)
THEN ПД (ПНОМ:ПНОМ1, ДНОМ: ДНОМ1
)))**

ПРИМЕР 3.13. *Получить имена поставщиков, которые не поставляют деталь Д2*

{ИМЯ1 | (\$ ПНОМ1) (ПОСТАВЩИК (ПНОМ1, ИМЯ1) Ù Ù Ø (ПД (ПНОМ1, ДНОМ) Ù ДНОМ = 'Д2')) }

или

**ИМЯ1 WHERE EXISTS ПНОМ1 (ПОСТАВЩИК
(ПНОМ:ПНОМ1, ПФАМ:ИМЯ1)
AND NOT ПД (ПНОМ: ПНОМ1, ДНОМ: 'Д2'))**

ПРИМЕР 3.14. *Получить номера поставщиков, поставляющих по крайней мере все детали, поставляемые поставщиком П2*

{ПНОМ1 | (" ДНОМ1) ((ПД (ПНОМ, ДНОМ1) Ù Ù ПНОМ= 'П2') ® ПД (ПНОМ1, ДНОМ1))))}

или

**ПНОМ1 WHERE FORALL ДНОМ1 (IF ПД (ПНОМ: 'П2',
ДНОМ:ДНОМ1)
THEN ПД (ПНОМ:ПНОМ1,
ДНОМ:ДНОМ1))**

ПРИМЕР 3.15. Получить номера деталей, которые или весят более 16, или поставляются поставщиком П2, или и то и другое.

{ДНОМ1 | (\$ ВЕС1) (ДЕТАЛЬ (ДНОМ1,ВЕС1) Û Û ВЕС1>16)Û (ПД(ПНОМ,ДНОМ1)Û ПНОМ = 'П2')}

или

**ДНОМ1 WHERE EXISTS ВЕС1
(ДЕТАЛЬ (ДНОМ:ДНОМ1, ВЕС:ВЕС1) AND ВЕС1 > 16)
OR ПД (ПНОМ : 'П2', ДНОМ: ДНОМ1)**

4. Рекомендации студентам по подготовке к лабораторной работе с указанием литературы.

- изучить теоретические и справочно-информационные материалы;
- проделать приведенные примеры.

Литература.

- Дейт, К.Дж. Введение в системы баз данных / К.Дж.Дейт;пер.с англ.и ред.К.А.Птицына. — 8-е изд. — М. : Вильямс, 2005. — 1380с. : ил.
- Конноли Т., Бегг К. Базы данных. Проектирование, реализация и сопровождение. Теория и практика. – 3-е изд. – М.: Издательский дом "Вильямс", 2003. – 1440 с.: ил.
- Астахова, И.Ф. Информационные системы : Учеб.пособие / И.Ф.Астахова,А.С.Потапов,В.А.Чулюков,А.Н.Журбин;ВГПУ. — Воронеж : Б.и., 2002. — 148с.

5. Описание экспериментальных установок (лабораторного оборудования).

Упражнения выполняются в рабочей тетради.

6. Краткое содержание работы, выполняемой студентами в ходе занятия.

Требуется дать решение к упражнениям, перечисленным в п.6 лабораторной работы 1 в терминах исчисления доменов и в терминах исчисления кортежей.

7. Исходные данные для работы.

Исходными данными является учебная база данных, приведенная в п.5 лабораторной работы 1.

8. Методика анализа полученных результатов.

Сравнить полученные результаты с исходными данными.

11. Порядок оформления отчета по лабораторной работе и его защиты.

Отчет должен содержать условия упражнений, запросы и результаты запросов. При защите лабораторной надо уметь объяснить содержание запроса.

ЛАБОРАТОРНАЯ РАБОТА № 3

Тема: Реляционные операторы: язык SQL

Продолжительность 7 часов

1. Целью лабораторных работ является знакомство с языком манипулирования данными в реляционных информационных системах SQL.

2. Вопросы, подлежащие исследованию, связаны с изучением базовых конструкций языка SQL.

3. Теоретические и справочно-информационные материалы.

Берутся по ссылкам http://impm.vspu.ac.ru/~chul/sql_in_samples/ или http://www.vspu.ac.ru/~chul/sql_in_samples/ или из учебного пособия:

Астахова, Ирина Федоровна. SQL в примерах и задачах: Учеб. пособие для студ. вузов, обуч. по направлению 510200- "Приклад. математика и информатика" / И.Ф.Астахова, А.П.Толстобров, В.М.Мельников .— Минск: Новое знание, 2002 .— 175 с.: табл.

4. Рекомендации студентам по подготовке к лабораторной работе с указанием литературы.

- изучить теоретические и справочно-информационные материалы;
- проделать приведенные примеры.

Литература.

- Дейт, К.Дж. Введение в системы баз данных / К.Дж.Дейт;пер.с англ.и ред.К.А.Птицына. — 8-е изд. — М. : Вильямс, 2005. — 1380с. : ил.
- Конноли Т., Бегг К. Базы данных. Проектирование, реализация и сопровождение. Теория и практика. – 3-е изд. – М.: Издательский дом "Вильямс", 2003. – 1440 с.: ил.
- Астахова, Ирина Федоровна. SQL в примерах и задачах: Учеб. пособие для студ. вузов, обуч. по направлению 510200- "Приклад. математика и информатика" / И.Ф.Астахова, А.П.Толстобров, В.М.Мельников .— Минск: Новое знание, 2002 .— 175 с.: табл.

5. Описание экспериментальных установок (лабораторного оборудования).

Лабораторная выполняется на компьютерах, объединенных в локальную сеть и имеющих доступ к Интранету каф. информатики.

На сервере должна быть установлена СУБД Oracle версии 8 или более поздней.

В Oracle должна быть создана учебная база данных, содержимое которой описано в учебном пособии:

- Астахова, Ирина Федоровна. SQL в примерах и задачах: Учеб. пособие для студ. вузов, обуч. по направлению 510200- "Приклад. математика и информатика" / И.Ф.Астахова, А.П.Толстобров, В.М.Мельников .— Минск: Новое знание, 2002 .— 175 с.: табл.

на стр.19-22 или по ссылкам:

http://impm.vspu.ac.ru/~chul/sql_in_samples/

или http://www.vspu.ac.ru/~chul/sql_in_samples/

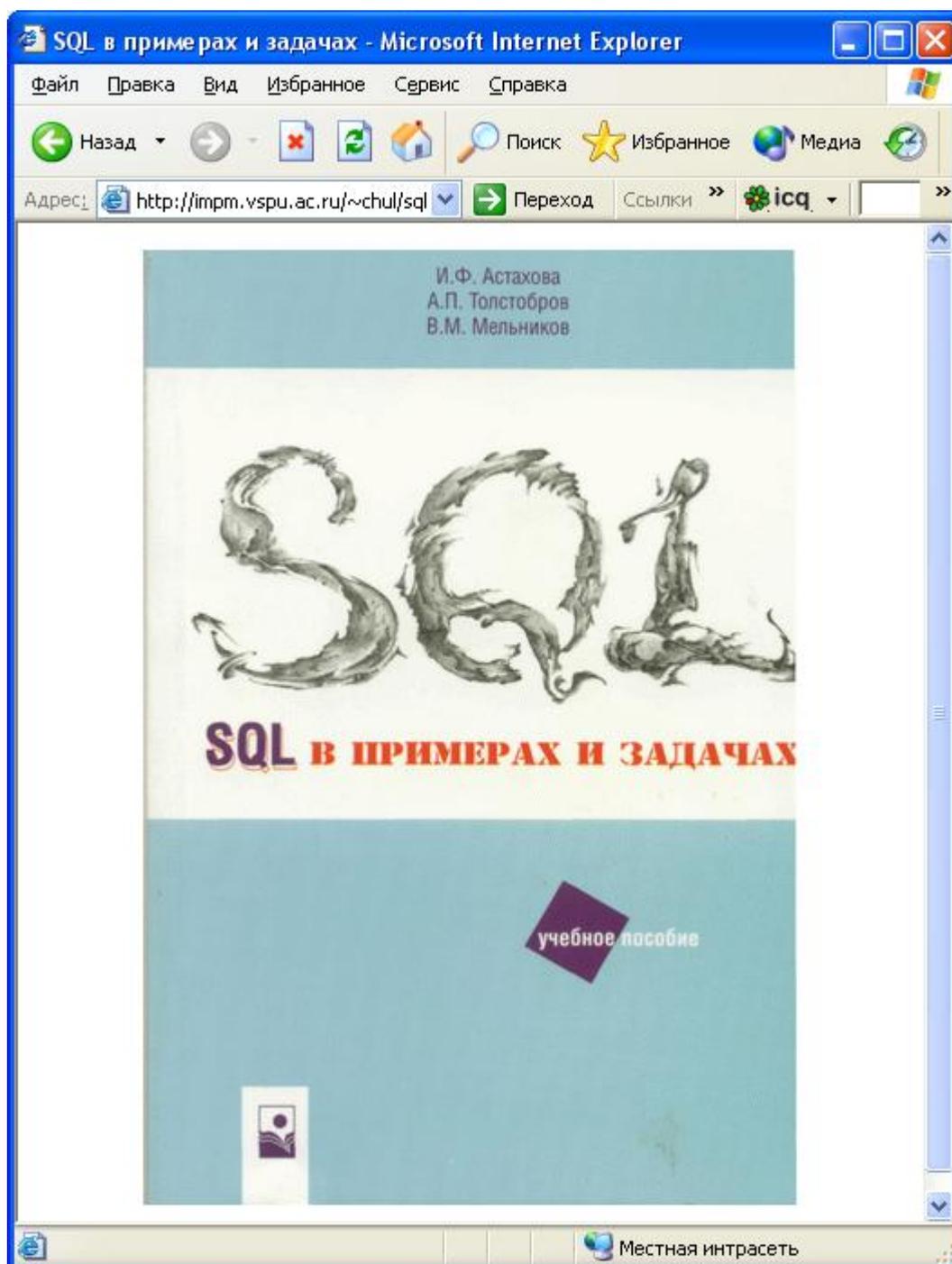
6. Краткое содержание работы, выполняемой студентами в ходе занятия.

Студенты формируют на ЭВМ запросы к учебной базе данных на языке SQL, используя программу по ссылке: <http://impm.vspu.ac.ru/~oracle/>

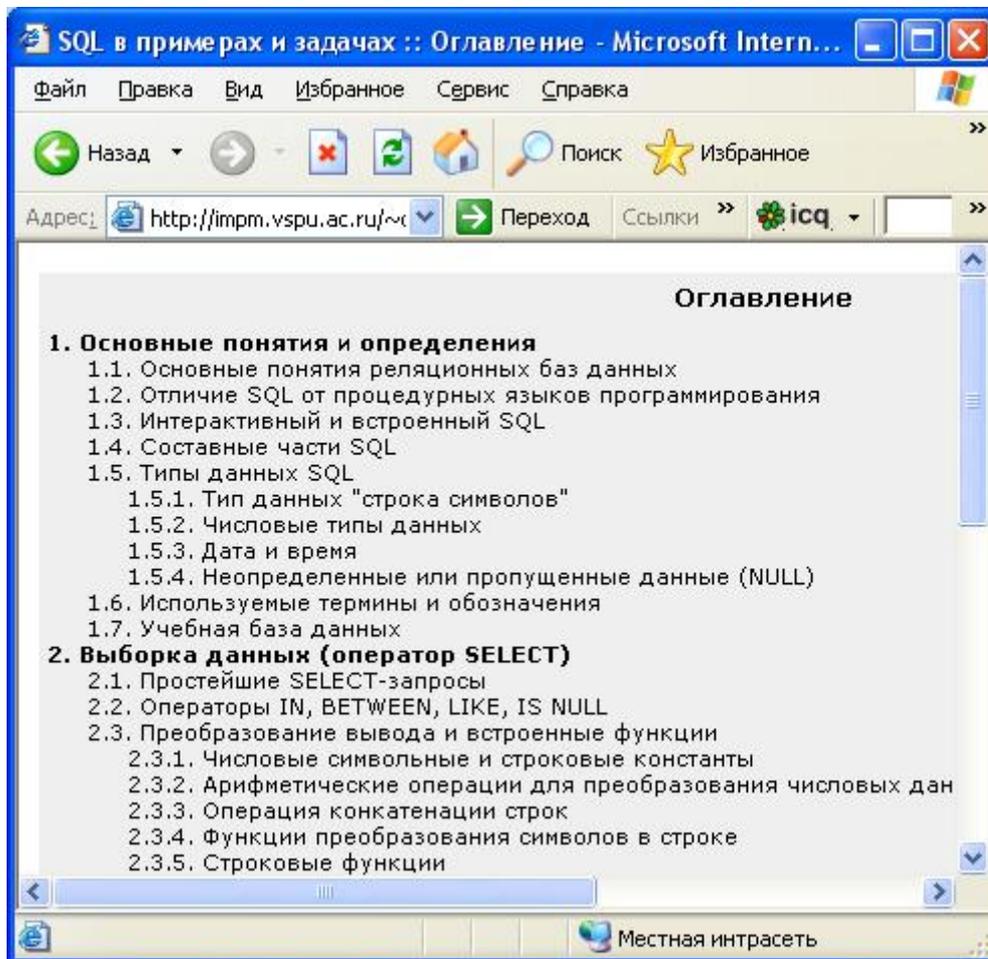
При этом выполняются все примеры данного теоретического раздела, а затем упражнения, указанные преподавателем.

7. Порядок проведения эксперимента, постановки опыта, снятия замеров и обработки данных эксперимента.

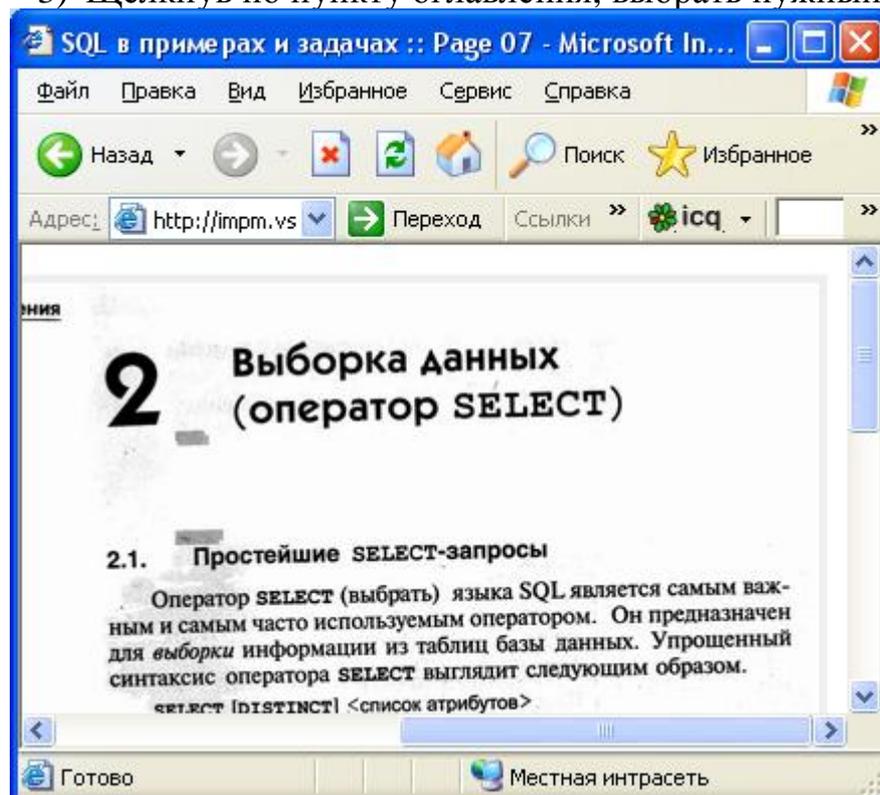
- 1) Зайти по ссылке http://impm.vspu.ac.ru/~chul/sql_in_samples/ или http://www.vspu.ac.ru/~chul/sql_in_samples/



- 2) Щелкнув по обложке, зайти в оглавление.



3) Щелкнув по пункту оглавления, выбрать нужный раздел.



Изучив теоретическую часть, выполнить примеры. Рассмотрим порядок выполнения примера на стр.26.

Написать запрос, выполняющий выборку имен (NAME) всех студентов с фамилией (SURNAME) Претров, сведения о которых находятся в таблице STUDENT.

- 4) Зайти на страницу по ссылке : <http://imp.m.vspu.ac.ru/~oracle/> (рис.1)

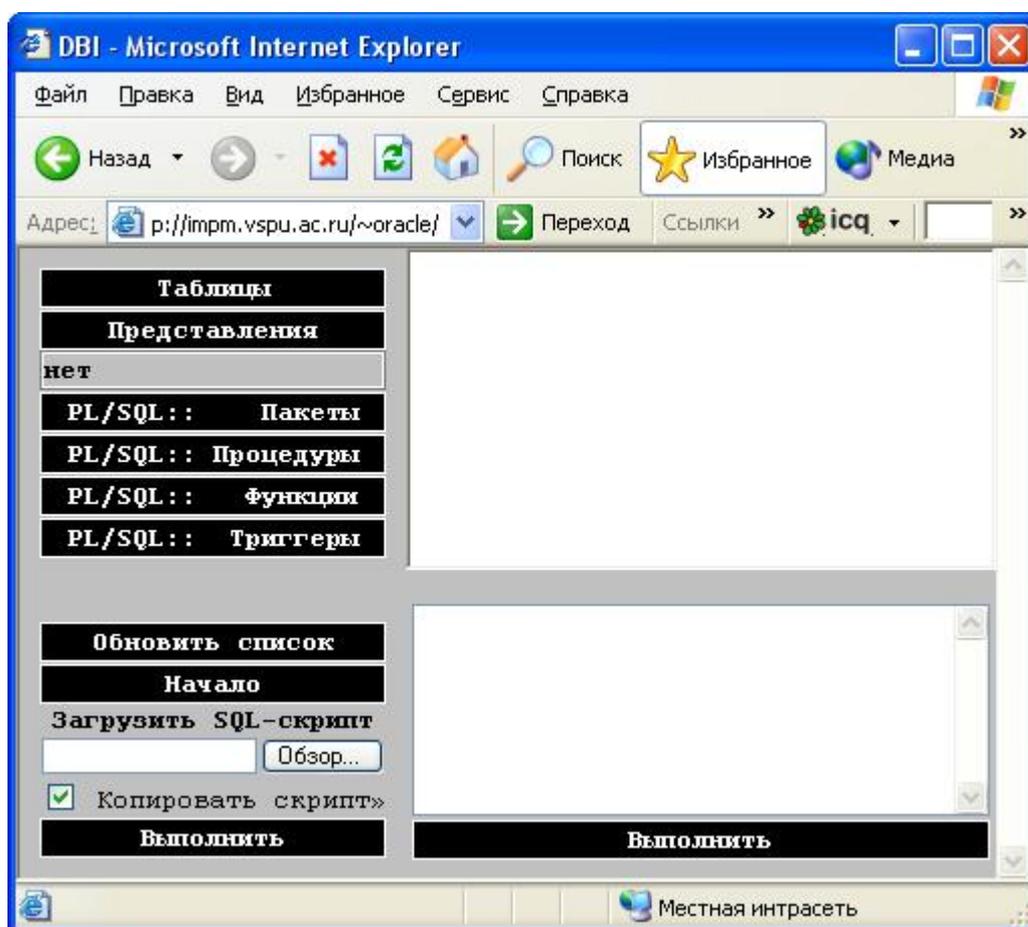


Рис. 1.

- 5) При необходимости щелкните по кнопке «Таблицы», выберите нужную таблицу для вывода структуры и содержимого этой таблицы (рис. 2).
- 6) Введите в нижнее поле запрос и нажмите Ctrl+Enter. В верхнем поле появится результат (рис. 3).

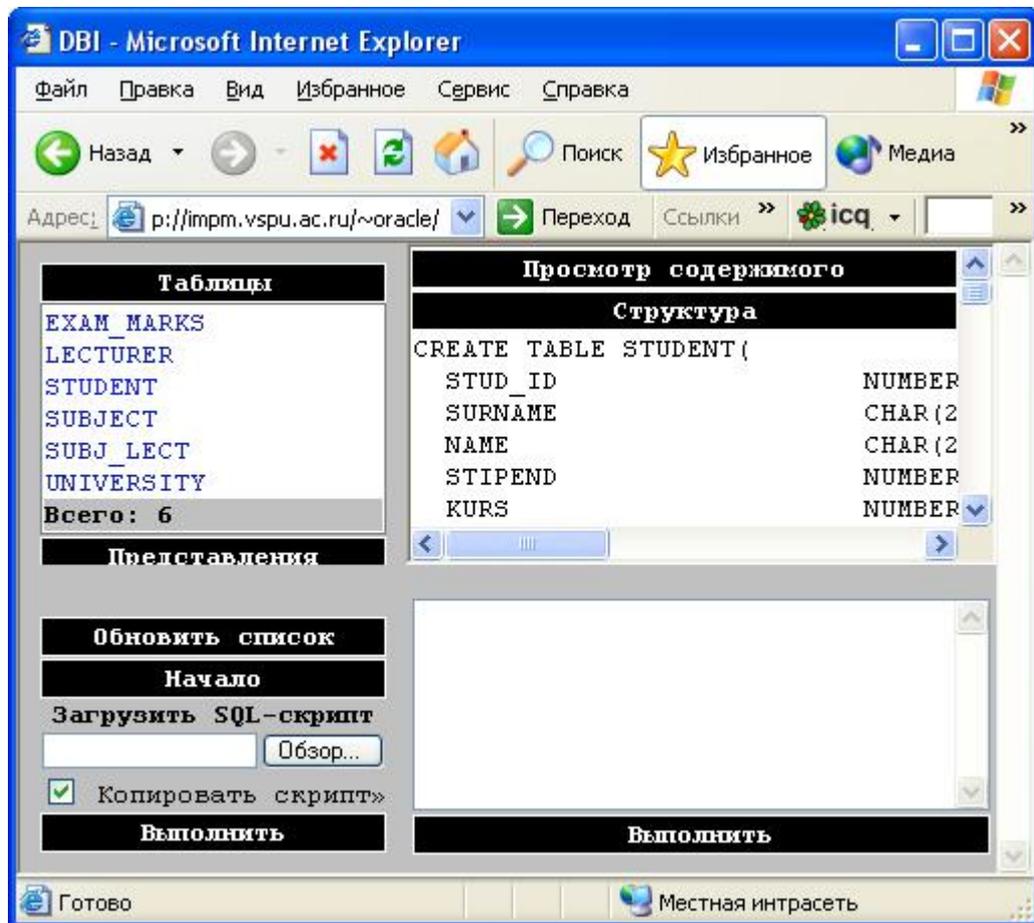


Рис. 2.

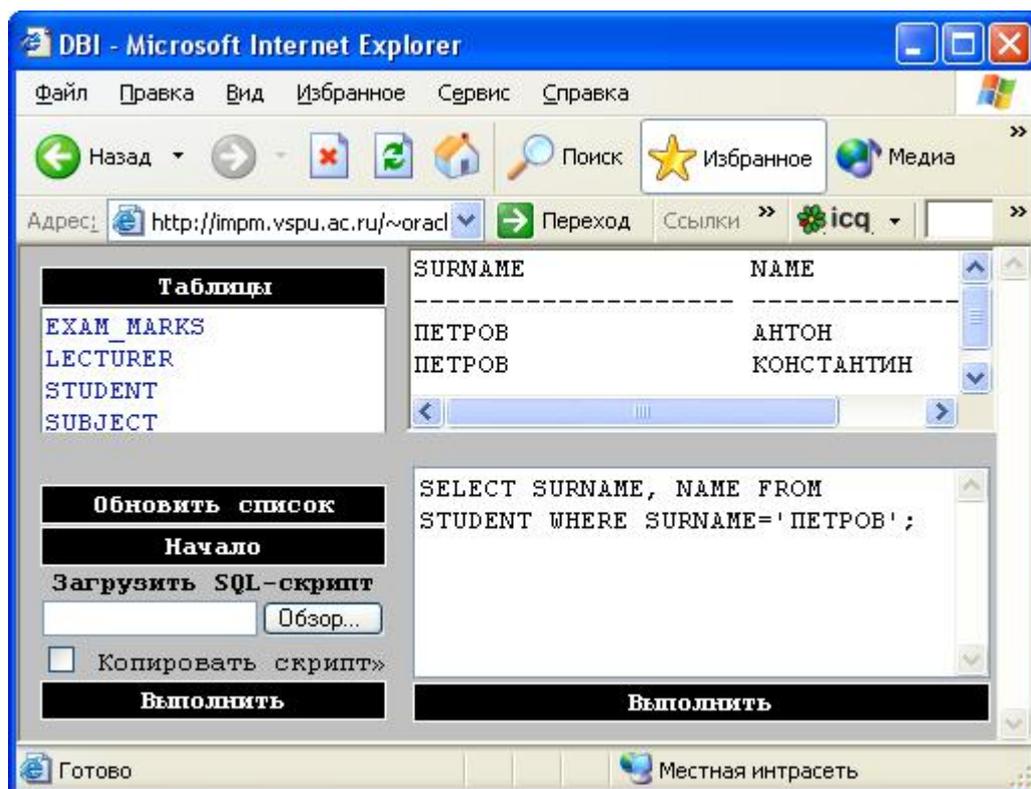


Рис. 3.

8. Техника безопасности.

Определяется правилами техники безопасности при работе с компьютером.

9. Исходные данные для работы.

Учебная база данных (п.5).

10. Методика анализа полученных результатов.

Сравнить полученные результаты с исходными данными.

11. Порядок оформления отчета по лабораторной работе и его защиты.

Отчет должен содержать условия упражнений, запрос и результат запроса. При защите лабораторной надо уметь объяснить содержание запроса.

ЛАБОРАТОРНАЯ РАБОТА № 4

Тема: Проектирование БД на основе концепции сущности-связи

Продолжительность 4 часа

1. Целью лабораторных работ является знакомство с основами проектирования баз данных, функциональными зависимостями, алгоритмом декомпозиции.

2. Вопросы, подлежащие исследованию, связаны с изучением различных способов проектирования баз данных.

3. Теоретические и справочно-информационные материалы.

Теоретические материалы необходимо изучать в лекциях 11,12,13 и 15 и в пособии:

Астахова, И.Ф. Информационные системы : Учеб.пособие / И.Ф.Астахова,А.С.Потапов,В.А.Чулюков,А.Н.Журбин;ВГПУ. — Воронеж : Б.и., 2002. — 148с.

4. Рекомендации студентам по подготовке к лабораторной работе с указанием литературы.

- изучить теоретические и справочно-информационные материалы;
- проделать приведенные упражнения.

Литература.

- Дейт, К.Дж. Введение в системы баз данных / К.Дж.Дейт;пер.с англ.и ред.К.А.Птицына. — 8-е изд. — М. : Вильямс, 2005. — 1380с. : ил.
- Конноли Т., Бегг К. Базы данных. Проектирование, реализация и сопровождение. Теория и практика. – 3-е изд. – М.: Издательский дом "Вильямс", 2003. – 1440 с.: ил.
- Астахова, И.Ф. Информационные системы : Учеб.пособие / И.Ф.Астахова,А.С.Потапов,В.А.Чулюков,А.Н.Журбин;ВГПУ. — Воронеж : Б.и., 2002. — 148с.

5. Описание экспериментальных установок (лабораторного оборудования).

Упражнения выполняются в рабочей тетради.

6. Краткое содержание работы, выполняемой студентами в ходе занятия.

Требуется дать решение следующих задач.

ЗАДАЧИ К ГЛАВЕ "ПРОЕКТИРОВАНИЕ БАЗ ДАННЫХ"

1. Найдите первичный ключ для отношения ФАКУЛЬТЕТ, приведенного на рис. 1:

Сном	Сфам	Кном	Тном	Курс	Семестр	Оценка
3297	Иванов	120	136	МТА	О97	4
3297	Иванов	120	136	ИНФ	О97	5
3297	Иванов	120	136	ФИЗ	398	2
3297	Иванов	120	136	МТА	398	4
3496	Петров	238	344	МТА	О96	3
3496	Петров	238	344	МТА	397	4
3496	Петров	238	344	ПСИ	398	5
3596	Сидоров	120	136	АЛГ	397	2
3596	Сидоров	120	136	ГЕО	О97	4
3596	Сидоров	120	136	ФИЗ	О97	3
4798	Николаев	345	321	ПЕД	399	4

Рис. 1. Отношение ФАКУЛЬТЕТ

2. Укажите примеры данных в отношении ФАКУЛЬТЕТ (рис. 1), являющиеся как дублированными, так и избыточными.
3. Указать примеры данных в отношении ФАКУЛЬТЕТ (рис. 1), являющиеся дублированными, но неизбыточными.

4. Отношение ТЕЛНОМ предназначено для хранения информации о фамилиях служащих, номерах домашних и служебных телефонов. Каждый служащий имеет уникальную фамилию, один домашний номер телефона и может иметь несколько служебных телефонных номеров. У нескольких служащих номер служебного телефона может быть общим. Типичный экземпляр отношения ТЕЛНОМ приведен на рис. 2.

ТЕЛНОМ		
Фамилия	Дтелном	Слж_тел
ИвановИИ	45-11-21	3167
ИвановИИ	45-11-21	3168
ПетровПП	45-98-98	4000
СидоровСС	48-45-12	3167
СидоровСС	48-45-12	3168
ЗайцевЗЗ	45-77-66	4001
ЗайцевЗЗ	45-77-66	4002

Рис.2. Экземпляр отношения ТЕЛНОМ

- а) укажите, какие данные в экземпляре отношения ТЕЛНОМ являются избыточными;
 б) как следует изменить отношение ТЕЛНОМ, чтобы исключить избыточные данные;
 в) приведите экземпляры отношений, которые Вы предлагаете использовать с целью исключения избыточности. Воспользуйтесь данными, приведенными на рис.2.

ЗАДАЧИ К ГЛАВЕ "ФУНКЦИОНАЛЬНЫЕ ЗАВИСИМОСТИ"

1. На рис. 1 приведены диаграммы функциональных зависимостей для нескольких отношений. Для каждого отношения требуется определить все детерминанты и все возможные ключи. Определите те отношения, которые находятся в НФБК. Если отношение не в НФБК, приведите его к нормальной форме, используя алгоритм декомпозиции.

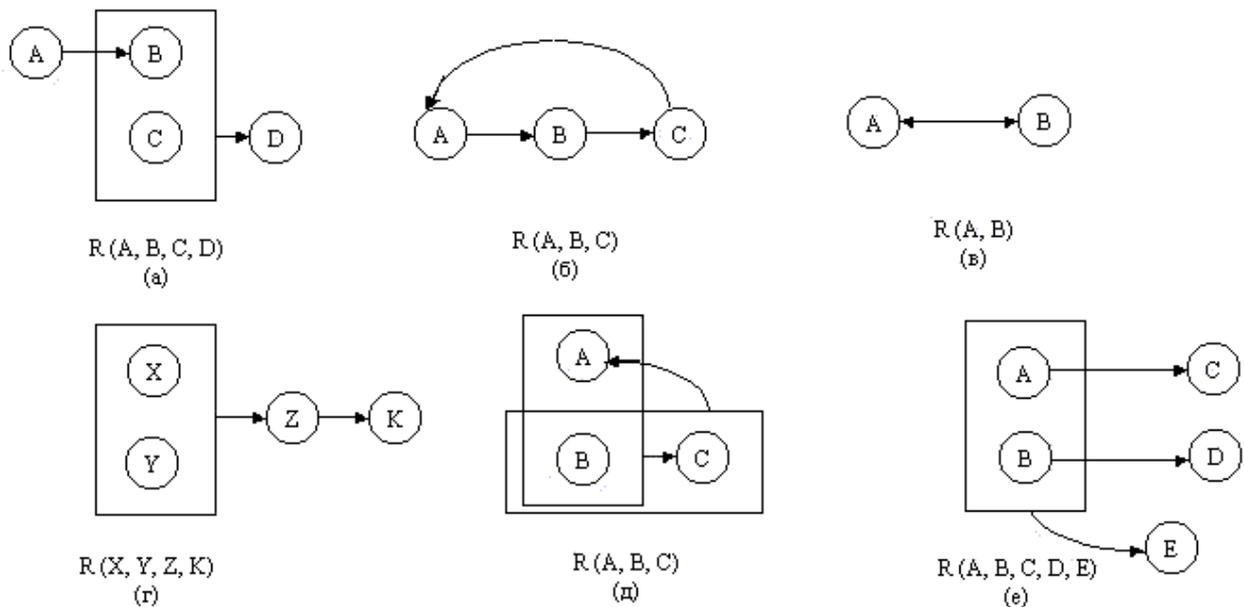


Рис. 1. Данные к задаче 1

2. Определите функциональные зависимости между атрибутами отношения ТЕЛ-НОМ, рассмотренного в задаче 4 предыдущего раздела.
3. Руководство супермаркета желает разработать БД, предназначенную для хранения информации о счетах покупателей. Информация, касающаяся каждого покупателя, должна включать в себя следующее: номер счета, фамилию, адрес, номер телефона, кредитоспособность (высокая, средняя, низкая, слабая) и сведения об уплате налогов. Изобразите диаграмму функциональных зависимостей, используя соответствующие атрибуты и выделяя сделанные предположения. Получите для БД отношения в НФБК.
4. Секретарь государственного учреждения проектирует БД для хранения информации обо всех автомобилях, зарегистрированных в штате. Информация, хранение которой предполагается, включает в себя регистрационный номер, номер лицензии, марку, адрес владельца, название страховой компании, номер страхового полиса, округ, в котором зарегистрирован автомобиль, и дату последней регистрации. Разработайте диаграммы функциональных зависимостей для соответствующих атрибутов.

ЗАДАЧИ К ГЛАВЕ "НЕКОТОРЫЕ УТОЧНЕНИЯ АЛГОРИТМА ДЕКОМПОЗИЦИИ"

1. Представьте каждую функциональную зависимость, показанную на рис. 1, в математической форме. Затем шаг за шагом редуцируйте диаграмму посредством удаления всех транзитивных зависимостей.

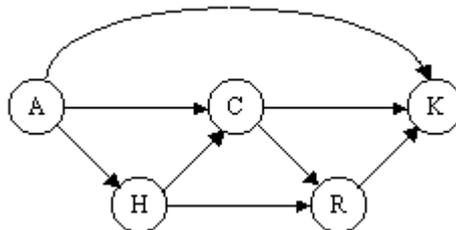


Рис. 1. Диаграмма ФЗ для задачи 1

2. На рис. 2 идентифицируйте все функциональные зависимости, избыточность которых является следствием добавления.

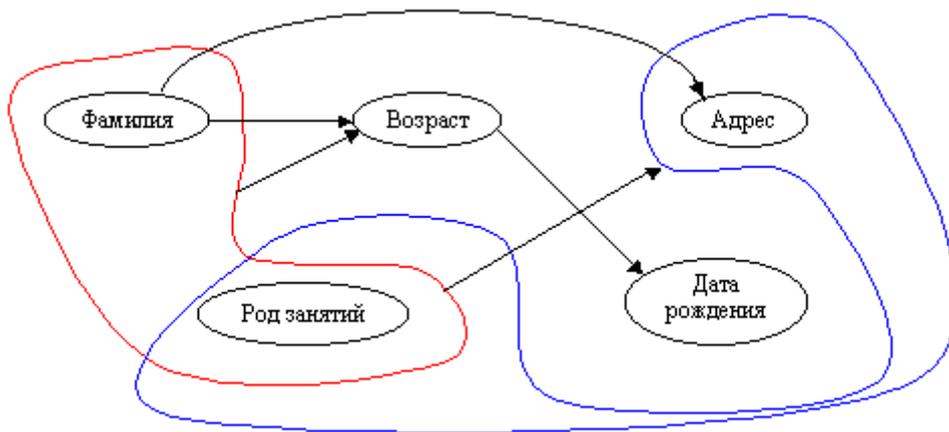


Рис. 2. Диаграмма ФЗ для задачи 2

3. Может ли диаграмма, представленная справа на рис. 3, быть получена из диаграммы, представленной в левой части того же рисунка, с помощью правил вывода?

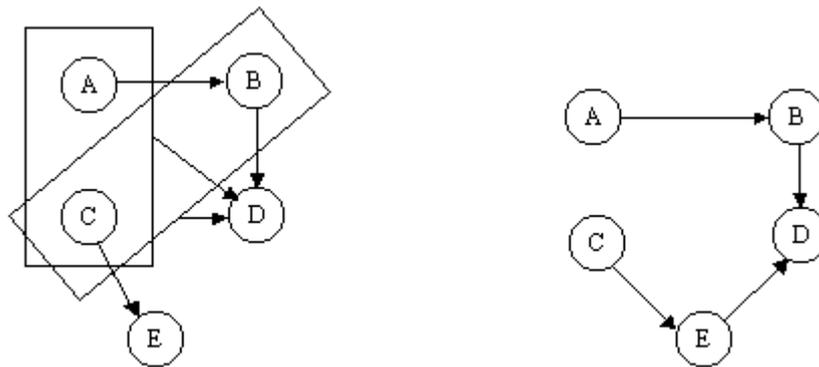


Рис. 3. Диаграмма ФЗ для задачи 3

4. Предположим, что в результате проектирования были получены следующие четыре НФБК-отношения:

R1 (Слж-фам, Слж-адр, Возраст, Пол, Начальник-фам)

R2 (Начальник-фам, Отдел)

R3 (Слж-фам, Отдел)

R4 (Отдел, Отдел-Тел-номер, Отдел-адрес)

Одно из этих отношений является избыточным. Укажите это отношение и поясните причину.

5. Если перечисленные в задаче 4 отношения находятся в НФБК, то какие из следующих утверждений являются истинными и какие ложными? Дайте обоснование сделанных выводов:

- служащий может иметь более одного руководителя;
- служащий может работать только в одном отделе;

- в) в каждом отделе имеется только один телефон;
- г) отделу выделяется только один телефонный номер;
- д) у каждого руководителя имеется телефон;
- е) служащие, не являющиеся руководителями, не имеют телефонов.

7. Порядок оформления отчета по лабораторной работе и его защиты.

Отчет должен содержать условия задач и их решение. При защите лабораторной надо уметь объяснить логику получения решения.

ЛАБОРАТОРНАЯ РАБОТА № 5

Тема: Получение отношений из диаграмм ER-типа

Продолжительность 3 часа

1. Целью лабораторных работ является знакомство с основами ER-проектирования баз данных.

2. Вопросы, подлежащие исследованию, связаны с изучением различных способов проектирования баз данных.

3. Теоретические и справочно-информационные материалы.

Теоретические материалы необходимо изучать в лекциях 11,12,13 и 15 и в пособии:

Астахова, И.Ф. Информационные системы : Учеб.пособие / И.Ф.Астахова,А.С.Потапов,В.А.Чулюков,А.Н.Журбин;ВГПУ. — Воронеж : Б.и., 2002. — 148с.

4. Рекомендации студентам по подготовке к лабораторной работе с указанием литературы.

- изучить теоретические и справочно-информационные материалы;
- проделать приведенные упражнения.

Литература.

- Дейт, К.Дж. Введение в системы баз данных / К.Дж.Дейт;пер.с англ.и ред.К.А.Птицына. — 8-е изд. — М. : Вильямс, 2005. — 1380с. : ил.
- Конноли Т., Бегг К. Базы данных. Проектирование, реализация и сопровождение. Теория и практика. — 3-е изд. — М.: Издательский дом "Вильямс", 2003. — 1440 с.: ил.

- Астахова, И.Ф. Информационные системы : Учеб.пособие / И.Ф.Астахова,А.С.Потапов,В.А.Чулюков,А.Н.Журбин;ВГПУ. — Воронеж : Б.и., 2002. — 148с.

5. Описание экспериментальных установок (лабораторного оборудования).

Упражнения выполняются в рабочей тетради.

6. Краткое содержание работы, выполняемой студентами в ходе занятия.

Требуется дать решение следующих задач.

ЗАДАЧИ К ГЛАВЕ ER – ПРОЕКТИРОВАНИЕ

1. Начертите примеры диаграмм ER-экземпляров для всех классов принадлежности и случая степени связи 1:1.
2. Начертите примеры диаграмм ER-экземпляров для всех классов принадлежности и случая степени связи 1:n.
3. Начертите примеры диаграмм ER-экземпляров для всех классов принадлежности и случая степени связи n:1.
4. Начертите примеры диаграмм ER-экземпляров для всех классов принадлежности и случая степени связи m:n.
5. Начертите типичные диаграммы экземпляров сущностей для каждой из следующих ситуаций. Перечислите предположения, сделанные в каждом случае:
 - а) руководству сети бакалейных магазинов требуется хранить информацию об отдельных магазинах и поставщиках, у которых эти магазины покупают продукцию; каждый магазин покупает продукцию у нескольких поставщиков, и каждый поставщик обеспечивает продукцией несколько магазинов;
 - б) службе окраски домов необходимо быть в курсе дел маляров, входящих в союз, и знать конкретно те дома, окраску которых они осуществляют; каждый маляр красит только один дом в любой момент времени, но один дом могут красить несколько маляров одновременно; некоторые маляры могут быть без работы; дома, не подлежащие окраске, не учитываются;
 - в) в гараже работают механики по ремонту автомобилей; каждый механик занят ремонтом сразу нескольких автомашин, при этом ремонт каждого автомобиля осуществляется только одним механиком.

Для каждой диаграммы экземпляров, полученной при решении задач из пункта 5, начертите диаграмму ER-типа. Укажите ключи сущностей в каждой диаграмме.

7. Порядок оформления отчета по лабораторной работе и его защиты.

Отчет должен содержать условия задач и их решение. При защите лабораторной надо уметь объяснить логику получения решения.

ЛАБОРАТОРНАЯ РАБОТА № 6

Тема: **Разработка пользовательских программ в среде баз данных**

Продолжительность 6 часов

1. Целью лабораторных работ является знакомство с основами проектирования пользовательских программ в среде баз данных.

2. Вопросы, подлежащие исследованию, связаны с изучением различных способов проектирования баз данных в различных СУБД.

3. Теоретические и справочно-информационные материалы.

Теоретические материалы необходимо изучать в лекциях 11,12,13 и 15 и в пособии:

Астахова, И.Ф. Информационные системы : Учеб.пособие / И.Ф.Астахова,А.С.Потапов,В.А.Чулюков,А.Н.Журбин;ВГПУ. — Воронеж : Б.и., 2002. — 148с.

4. Рекомендации студентам по подготовке к лабораторной работе с указанием литературы.

- изучить теоретические и справочно-информационные материалы;
- разработать проект базы данных и реализовать его в среде Oracle или MS Access.

Литература.

- Дейт, К.Дж. Введение в системы баз данных / К.Дж.Дейт;пер.с англ.и ред.К.А.Птицына. — 8-е изд. — М. : Вильямс, 2005. — 1380с. : ил.
- Конноли Т., Бегг К. Базы данных. Проектирование, реализация и сопровождение. Теория и практика. — 3-е изд. — М.: Издательский дом "Вильямс", 2003. — 1440 с.: ил.
- Астахова, И.Ф. Информационные системы : Учеб.пособие / И.Ф.Астахова,А.С.Потапов,В.А.Чулюков,А.Н.Журбин;ВГПУ. — Воронеж : Б.и., 2002. — 148с.

5. Описание экспериментальных установок (лабораторного оборудования).

Проект выполняется на компьютере в среде Oracle или Access.

6. Краткое содержание работы, выполняемой студентами в ходе занятия.

Создаваемый проект должен отвечать следующим условиям.

- 1) Информационная система должна обеспечивать выполнение перечисленных ниже операций над базами.
- 2) База состоит из нескольких связанных таблиц с использованием (по необходимости) связей «один к одному», «один ко многим», «много к одному», «много ко многим».
- 3) Для облегчения просмотра и поиска используется упорядоченность с помощью индексации или (в крайнем случае) сортировки.
- 4) Вся информация поддается просмотру и редактированию. Записи, содержащие много полей, можно просматривать как в табличном, так и в постраничном виде (в виде карточек).
- 5) Информация, поддающаяся стандартизации, хранится в отдельных таблицах-справочниках, также поддающихся корректировке (с соблюдением целостности).
- 6) Программа позволяет проводить поиск или выборку информации по произвольному запросу (фильтру). При неудачном задании условия запроса возможна его корректировка (с целью сужения или расширения зоны поиска).

7. Техника безопасности.

Соответствует правилам техники безопасности при работе с компьютером.

8. Исходные данные для работы.

Задания для проектирования и создания информационной системы.

1. Личная библиотека. Карточка домашней библиотеки: выходные данные книги (авторы, название, издательство и так далее), раздел библиотеки (специальная литература, хобби, домашнее хозяйство, беллетристика и так далее), происхождение и наличие книги в данный момент, субъективная оценка книги. Выбор книг по произвольному запросу; инвентаризация библиотеки.
2. Карточка Интерпола. Данные по каждому зарегистрированному преступнику: фамилия, имя, кличка, рост, цвет волос и глаз, особые приметы, гражданство, место и дата рождения, последнее место жительства, знание языков, преступная профессия, последнее дело и так далее. Преступные и мафиозные группировки (данные о подельщиках). Выборка по любому подмножеству признаков. Перенос «завязавших» в архив; удаление — только после смерти.

3. Бюро знакомств. База потенциальных женихов и невест: пол, регистрационный номер, дата регистрации, сведения о себе, требования к партнеру. Выбор подмножества подходящих кандидатур, подготовка встреч (формирование приглашения для знакомства). Перенос в архив пар, решивших свои семейные проблемы, удаление клиентов, отказавшихся от услуг.
4. Биржа труда. База безработных: анкетные данные, профессия, образование, место и должность последней работы, причина увольнения, семейное положение, жилищные условия, контактные координаты, требования к будущей работе. База вакансий: фирма, должность, условия труда и оплаты, жилищные условия, требования к специалисту. Поиск и регистрация вариантов с той и другой стороны; формирование объявлений для печати, удаление в архив после трудоустройства, полное удаление при отказе от услуг.
5. Записная книжка. Анкетные данные, адреса, телефоны, место работы или учебы, должность знакомых, коллег и родственников, характер знакомства, деловые качества и так далее. Автоматическое формирование поздравления с днем рождения (по текущей дате). Упорядочение по алфавиту и по дате последней корректировки. Поиск по произвольному шаблону.
6. Касса аэрофлота. Расписание: номер рейса, маршрут, пункты промежуточной посадки, время отправления, дни полета. Количество свободных мест на каждом рейсе. Выбор ближайшего рейса до заданного пункта (при наличии свободных мест), оформление заданного числа билетов по согласованию с пассажиром (с уменьшением числа свободных мест), оформление посадочной ведомости.
7. Справочник потребителя (служба быта). База предприятий бытового обслуживания города: название, разряд, адрес и телефоны, специализация, перечень оказываемых услуг, форма собственности, часы и дни работы. Поиск предприятий по заданной услуге и другим признакам.
8. Справочник покупателя. База торговых точек города: название, адрес и телефоны, специализация, форма собственности, время работы. Выбор магазинов по произвольному шаблону.
9. Магазин с одним продавцом. Компьютер вместо кассового аппарата. База наличия товаров: наименование, единица измерения, цена единицы, количество, дата последнего завоза. Регистрация поступления товара (как старых, так и новых наименований). Оформление покупки: выписка чека, корректировка базы. Проблема уценки и списания. Инвентаризация остатков товара с вычислением суммарной стоимости.
10. 17.10 Отдел кадров. База данных о сотрудниках фирмы: паспортные данные, образование, специальность, подразделение, должность, оклад, даты поступления в фирму и последнего назначения и т. д. Выбор по

произвольному шаблону. Сокращение штатов: выбор для увольнения лиц пенсионного и предпенсионного возраста, подготовка приказа.

11. Генеалогическое дерево. Паспортные данные членов некоторого родового клана; ссылки на детей (или на родителей). Поиск всех потомков или всех предков для указанного лица.
12. Склад. База товаров, хранящихся на складе: наименование, единица измерения, цена единицы, количество, дата последнего завоза. Регистрация поступления товара (формирование приходной накладной) и отгрузки (расходная накладная). Вывод инвентарной ведомости.
13. Касса автовокзала. Расписание автобусов: номер рейса, конечный и промежуточные пункты, время отправления. Количество свободных мест на каждом рейсе. Выбор ближайшего рейса до заданного пункта (при наличии свободных мест), оформление билетов, оформление посадочной ведомости. Предварительная продажа, возврат билетов.
14. Администратор гостиницы. Список номеров: класс, число мест. Список гостей: паспортные данные, даты приезда и отъезда, номер. Поселение гостей: выбор подходящего номера (при наличии свободных мест), регистрация, оформление квитанции. Отъезд: выбор всех постояльцев, отъезжающих сегодня, освобождение места или оформление задержки с выпиской дополнительной квитанции. Возможность досрочного отъезда с перерасчетом. Поиск гостя по произвольному признаку.
15. Справочник меломана. База групп и исполнителей; база песен; база дисков с перечнем песен (в виде ссылок). Выбор всех песен заданной группы; всех дисков, где встречается заданная песня.
16. Ежедневник. База намечаемых мероприятий — дата, время и продолжительность, место проведения. Автоматическое напоминание ближайшего дела: по текущей дате и времени; удаление вчерашних дел либо перенос на будущее. Анализ «накладок» — пересечений планируемых дел. Просмотр дел на завтра, послезавтра и так далее.
17. Терминология. База определений какой-либо науки: вводимый термин, его толкование (определение), ссылки на используемые термины. Возможность просмотра всей цепочки от заданного термина до первичных понятий.
18. Шеф-повар. База рецептов блюд: раскладка, рецепт приготовления. База продуктов на складе: наименование, цена, количество. Формирование меню на день (на заданное число персон); званый ужин. Проверка достаточности запасов; формирование расходной накладной на склад, корректировка запасов.
19. Справочник лекаря. База болезней: название, симптомы, процедуры, перечень рекомендуемых лекарств с указанием требуемого количества. База медикаментов на складе: название, количество, взаимозаменяемость. Формирование рецепта после осмотра больного, проверка наличия лекарств, корректировка запасов.

- 20.** Зачисление абитуриентов. База абитуриентов: анкетные данные, совокупность оценок на вступительных экзаменах, готовность учиться на договорной основе. Выбор для зачисления заданного количества абитуриентов; формирование для собеседования списка тех, кто набрал предельный проходной балл, но не может платить за образование.
- 21.** Обмен жилья. База предложений по обмену: район, площадь, планировка и т. д.; требования к вариантам обмена. Регистрация клиентов, выбор подходящих вариантов, удаление при состоявшемся обмене или отказе.
Развитие задачи. Возможность съезда или разъезда, в том числе «несколько на несколько»; «возможны варианты».
- 22.** Справочник абитуриента. База вузов: наименование, адрес, перечень специальностей, конкурс прошлого года по каждой специальности (дневной, вечерней, заочной форм), размер оплаты при договорном обучении. Выбор по разным критериям: все о данном вузе; все о данной специальности, поиск минимального конкурса по данной специальности или вообще.
- 23.** Справочник почтовой индексации. Республика, область (край), район, населенный пункт, почтовый индекс. Поиск по любой совокупности полей (кроме последнего); иерархическая связь между полями (обратите внимание, что, например, Павловск есть в Алтайском крае, Воронежской и Ленинградской областях).
- 24.** Сбербанк. Сведения о вкладчиках банка: номер лицевого счета, категория вклада, паспортные данные, текущая сумма вклада, дата последней операции. Операции приема и выдачи любой суммы, автоматическое начисление процентов.
- 25.** Ломбард. База хранимых товаров и недвижимости: анкетные данные клиента, наименование товара, оценочная стоимость; сумма, выданная под залог, дата сдачи, срок хранения. Операции приема товара, возврата, продажи по истечении срока хранения.
- 26.** Справочник селекционера. Наименование сорта какой-либо культуры, автор, родительские сорта, урожайность, характеристики плодов, морозоустойчивость, устойчивость к вредителям и болезням, наличие в том или ином селекционном фонде. Выбор сортов, обладающих заданными свойствами.
- 27.** Справочник работника ГИБДД. Марка, цвет, заводской и бортовой номера, дата выпуска, особенности конструкции и окраски, дата последнего техосмотра транспортного средства (автомобиля, мотоцикла, прицепа и т. д.), паспортные данные владельца. Выбор транспортных средств по произвольному шаблону. Формирование приглашений на техосмотр в соответствии со сроком.
- 28.** Справочник владельца видеотеки. База видеофильмов: название, студия, жанр, год выпуска, режиссер, исполнители главных ролей, краткое

содержание, субъективная оценка фильма. Факт наличия фильма в видеотеке. Оформление выдачи и возврата кассеты.

29. Купи-продай. База продавцов: наименование товара, объем партии при оптовой продаже, цена, условия продажи-отгрузки, форма оплаты, контактный адрес или телефон, примечание (например, «посредников прошу не беспокоить»). База покупателей: наименование товара, объем покупки, приемлемая цена и форма оплаты, контактный адрес или телефон, примечание. Поиск и регистрация вариантов с той и другой стороны; формирование объявлений для печати, удаление в архив после купли-продажи (возможно, один из клиентов остается неудовлетворенным), полное удаление при отказе от услуг.
30. Справочник фаната. База спортсменов: анкетные и антропологические данные, гражданство, происхождение, вид спорта, клуб или команда, данные о личном рекорде или победах и так далее. Выбор по произвольному признаку. Поиск рекордсмена в заданном виде спорта.
31. Справочник радиолюбителя. Базы паспортных данных транзисторов, диодов, тиристоров и так далее: марка, характеристики, предельно допустимые условия эксплуатации, цена, учет взаимозаменяемости и т. д. Подборка по заданным требованиям.
32. Справочник коммерческих банков. Наименование, адрес, статус (форма собственности), условия хранения средств на лицевом счете (годовые проценты на различных видах вкладов). Выбор банка с наибольшим процентом для заданного типа вклада.
33. Справочник начальника тюрьмы. Анкетные данные заключенных, статья, срок, дата заключения под стражу, место в тюремной иерархии, камера, сведения о родственниках, особенности характера. Формирование статистических сводок о составе, выбор по произвольному признаку.
34. Справочник командира. Список подчиненных военнослужащих: анкетные данные, адрес родителей, гражданская профессия, образование, звание и дата его получения, должность, подразделение, форма службы (срочная, кадровая, контрактная и так далее), период службы (для срочнослужащих), особенности характера и отношение к службе. Формирование списков: заданного подразделения, офицерского состава, новобранцев и т. д.
35. Риэлтерская контора (купля-продажа жилья). База предложений: район и адрес, характеристика дома и квартиры, запрашиваемая стоимость, координаты заявителя. База спроса: требования покупателя к жилью (возможно несколько вариантов, допустимые диапазоны), финансовые возможности, координаты заявителя. Подбор вариантов для той и другой стороны, автоматизированный поиск взаимоприемлемых вариантов. Пример запроса покупателя: однокомнатная, до 200 тыс. р., Поток и Новосиликатный не предлагать.

36. Очередь на жилье. Список очередников на получение и улучшение жилья: дата поступления в фирму, дата подачи заявления, состав семьи, жилищные условия на дату подачи заявления, льготы на дополнительную площадь, внеочередное и первоочередное улучшение, пожелания на район и другие. Реализация распределения получаемого муниципального жилья: удовлетворение подходящих очередников, переселение следующих в освободившиеся квартиры. Добавление заявителей и удаление выбывших и удовлетворенных.
37. Автосалон. База новых и подержанных отечественных и иностранных автомобилей: марка, год выпуска, технические характеристики, особенности исполнения, техническое состояние, запрашиваемая цена. База покупателей: контактные координаты, требования к марке, техническим характеристикам и техническому состоянию, финансовые возможности Автоматизация подбора вариантов для покупателя, формирование заявки для поставщиков и перегонщиков.
38. Справочник туриста. Турагентства и предлагаемые услуги: страна, город (или маршрут круиза), условия проживания и проезда, экскурсионное обслуживание, сервис принимающей стороны, стоимость путевки.
39. •*Купи-продай». База объявлений (радио-, газетных, в бегущей строке) по всем рубрикам. Поиск для любого спроса или предложения, включая контекстный поиск в МЕМО-полях (например «репетиторство & математика»).
40. Крылатые фразы. Справочник пословиц, поговорок, афоризмов, каламбуров, других словесных курьезов. Классификация по авторам и источникам, поиск по темам и ключевым словам.
41. Каталог запчастей автомобиля. В автомобиле насчитывается несколько тысяч деталей; некоторые используются в разных марках. Таблицы: страна, фирма-изготовитель, марка автомобиля; агрегат, узел, деталь. Учет взаимозаменяемости. Пользователи: работники автосервиса, магазинов запчастей; поставщики-оптовики.
42. Каталог радиодеталей (справочник радиомастера). Модели бытовой аудио- и видеотехники; для каждой модели — каталог радиодеталей, использованных в ней (резисторы, конденсаторы, катушки индуктивности, чипы и т. д.). Многие детали используются в разных моделях; некоторые детали односторонне (например, вместо резистора 10КО,125Вт можно поставить 10КО,25Вт, но не наоборот) или двусторонне (транзисторы с близкими характеристиками) заменяемы. Учет наличия деталей на складе, поиск подходящих деталей для ремонта.
43. Справочник нумизмата. Монеты: страна, номинал, год выпуска, количество выпущенных монет, особенности. Коллекционеры: страна, имя, контактные координаты, наличие редких монет в коллекции. Собственная коллекция.

44. Справочник филателиста. Марки: страна, нарицательная стоимость, год выпуска, тираж, особенности. Филателисты: страна, имя, контактные координаты, наличие редких марок в коллекции. Собственная коллекция.
45. Справочник любителя живописи. Художники с анкетными данными и стилями. Картины со ссылкой на художников, датой создания, жанром. Коллекционеры и музеи: наличие оригиналов картин и копий. Аукционы и комиссии: дата проведения, список выставленных шедевров и цены на них. Собственная коллекция.
46. Телепрограмма. Программа телепередач нескольких телекомпаний (на неделю по дням, часам). Разные жанры телепередач: новости, спорт, художественные фильмы, сериалы и т. д. Выбор совокупной программы по определенному запросу (вкусу). Программирование видеомагнитофона при временных «накладках» передач.
47. Справочник астронома. Для каждой из зарегистрированных звезд известны: название, созвездие, видимая звездная величина, расстояние, координаты на небосклоне. Поиск звезд заданного созвездия, самых ярких звезд, видимых звезд и созвездий в заданной точке земного шара в заданное время.
48. Справочник географа. Города (географические координаты, численность населения), регионы (области, провинции, штаты и т. д. — принадлежность стране, столица, численность населения), страны (площадь, численность населения, форма государственного правления, столица), материка. Поиск городов-«тезок», подсчет доли городского населения в странах и регионах, населенность материков и т. д.
49. Справочник гидролога. Реки мира: протяженность (от истока до впадения), куда впадает (в другую реку, море или озеро), годовой сток (км³), площадь бассейна (без крупных притоков). Вычисление годового стока и площади бассейна для заданной крупной реки (вместе с ее притоками), моря или озера.
50. Справочник астронома. Видимые звезды: название, созвездие, видимая звездная величина, расстояние от Земли, координаты на небосклоне: прямое восхождение (ч, мин) и склонение (град., мин). Поиск звезд, входящих в данное созвездие, самой яркой звезды созвездия, видимых созвездий и звезд в заданной точке земного шара в заданное время.

9. Методика анализа полученных результатов.

Прект проверяется на соответствие структуры нормальной формы Бойса-Кодда, на полноту и корректность выполнения запросов.