



**В. В. Кириллов,
Г. Ю. Громов**

Введение в реляционные базы данных

- Базы данных и управление ими
- Реляционная модель данных
- SQL – стандартный язык для работы с реляционными базами данных
- Основы проектирования реляционных баз данных
- Создание приложений на языке SQL и его процедурных расширениях
- Хранимые процедуры
- Примеры создания баз данных
- Инструментарий для создания баз данных и приложений к ним

+CD

bhv[®]



УДК 681.3.06
ББК 32.973.26-018.2
К43

Кириллов, В. В.

К43 Введение в реляционные базы данных / В. В. Кириллов, Г. Ю. Громов. — СПб.: БХВ-Петербург, 2009. — 464 с.: ил. + CD-ROM — (Учебная литература для вузов)

ISBN 978-5-94157-770-5

В книге рассматриваются основные понятия баз данных и систем управления ими (СУБД), моделей данных, положенных в основу баз данных и методов проектирования реляционных баз данных. Обсуждаются реляционные операции и основы теории нормализации отношений и приводятся примеры проектирования баз данных. Большое место уделено подробному описанию языка SQL — международного стандарта языка реляционных баз данных. Рассматриваются основные понятия, необходимые для изучения SQL и применения его на практике. Подробно рассмотрено манипулирование данными в интерактивном режиме, затронуты вопросы обеспечения безопасности хранимых данных, средств оптимизации запросов и создания прикладных программ. На прилагаемом к книге компакт-диске содержатся дистрибутивы СУБД OracleXE, SqlDeveloper, учебные базы данных и дополнительные материалы.

Для студентов и начинающих программистов

УДК 681.3.06
ББК 32.973.26-018.2

Рецензент — А. А. Бобцов, профессор, д. т. н., декан факультета компьютерных технологий и управления СПбГУ ИТМО

Группа подготовки издания:

Главный редактор	<i>Екатерина Кондукова</i>
Зам. главного редактора	<i>Игорь Шишигин</i>
Зав. редакцией	<i>Григорий Добин</i>
Редактор	<i>Екатерина Капальгина</i>
Компьютерная верстка	<i>Натальи Караваевой</i>
Корректор	<i>Виктория Пиотровская</i>
Дизайн серии	<i>Инны Тачиной</i>
Оформление обложки	<i>Елены Беляевой</i>
Фото	<i>Кирилла Сергеева</i>
Зав. производством	<i>Николай Тверских</i>

Лицензия ИД № 02429 от 24.07.00. Подписано в печать 01.09.08.

Формат 70×100^{1/16}. Печать офсетная. Усл. печ. л. 37,41.

Тираж 2000 экз. Заказ №

"БХВ-Петербург", 194354, Санкт-Петербург, ул. Есенина, 5Б.

Санитарно-эпидемиологическое заключение на продукцию № 77.99.02.953.Д.006421.11.04 от 11.11.2004 г. выдано Федеральной службой по надзору в сфере защиты прав потребителей и благополучия человека.

Отпечатано с готовых диапозитивов
в ГУП "Типография "Наука"
199034, Санкт-Петербург, 9 линия, 12

ISBN 978-5-94157-770-5

© Кириллов В. В., Громов Г. Ю., 2008
© Оформление, издательство "БХВ-Петербург", 2008

Оглавление

Введение.....	1
ЧАСТЬ I. ЧТО ТАКОЕ БАЗА ДАННЫХ И СУБД.....	5
Глава 1. Зачем нужны базы данных	7
1.1. Данные и ЭВМ	7
1.2. Концепция баз данных	9
1.3. Архитектура СУБД.....	12
Глава 2. Инфологическая модель данных "сущность-связь"	15
2.1. Основные понятия	15
2.2. Характеристика связей и язык моделирования.....	16
2.3. Классификация сущностей	22
2.4. О первичных и внешних ключах.....	26
2.5. Ограничения целостности.....	29
2.6. О построении инфологической модели	30
Глава 3. Реляционный подход.....	32
3.1. Реляционная структура данных.....	32
3.2. Реляционная база данных.....	35
3.3. Манипулирование реляционными данными	43
3.3.1. Обновление отношений.....	44
3.3.2. Реляционные операции.....	46
ЧАСТЬ II. ЯЗЫК SQL. ИЗВЛЕЧЕНИЕ ДАННЫХ.....	55
Глава 4. Основы SQL.....	57
4.1. Стандарты языка SQL.....	57
4.2. Почему SQL?.....	58
4.3. Таблицы SQL.....	61

4.4. Синтаксические конструкции SQL	67
4.4.1. Предложения SQL	67
4.4.2. Идентификаторы (имена)	69
4.4.3. Константы и <i>NULL</i> -значения	70
4.4.4. Операторы	72
4.4.5. Резервированные и ключевые слова	74
4.4.6. Псевдостолбцы, таблица <i>DUAL</i> и еще о словах, которые нежелательно использовать пользователям	75
4.5. Типы данных SQL	77
4.5.1. Символьные	78
4.5.2. Двоичные	79
4.5.3. Числовые	80
4.5.4. Дата/время	81
4.5.5. Связь с данными	82
4.5.6. Интервальные	82
4.5.7. XML	82
4.5.8. Данные, специфичные для СУБД Oracle	83
4.6. Функции SQL	83
4.6.1. Числовые функции	84
4.6.2. Символьные функции	85
4.6.3. Даты и время	87
4.6.4. Преобразование данных	90
4.6.5. Различные функции для работы с одиночной строкой	92
4.6.6. Агрегатные функции	94
4.6.7. Функции <i>CASE</i> , <i>CAST</i> и <i>DECODE</i>	95
Глава 5. Запросы с использованием единственной таблицы	99
5.1. О предложениях <i>SELECT</i> и <i>SUBQUERY</i>	99
5.2. Выборка без использования фразы <i>WHERE</i>	100
5.2.1. Простая выборка	102
5.2.2. Исключение дубликатов	103
5.2.3. Выборка вычисляемых значений	104
5.3. Выборка с использованием фразы <i>WHERE</i>	106
5.3.1. Использование операторов сравнения	107
5.3.2. Использование <i>BETWEEN</i>	108
5.3.3. Использование <i>IN</i>	110
5.3.4. Использование <i>LIKE</i>	111
5.4. Выборка с упорядочением (<i>ORDER BY</i>)	112
5.5. Агрегирование данных	115
5.5.1. Агрегатные SQL-функции	115
5.5.2. Функции без использования фразы <i>GROUP BY</i>	116

5.5.3. Фраза <i>GROUP BY</i>	119
5.5.4. Использование фразы <i>HAVING</i>	121
5.6. Иерархические запросы	122

Глава 6. Запросы с использованием нескольких таблиц 125

6.1. О средствах одновременной работы с множеством таблиц	125
6.1.1. Использование фразы <i>JOIN</i>	128
6.2. Запросы, использующие соединения	130
6.2.1. Декартово произведение таблиц.....	130
6.2.2. Эквисоединение таблиц.....	133
6.2.3. Естественное соединение таблиц	134
6.2.4. Композиция таблиц	135
6.2.5. Тета-соединение таблиц	135
6.2.6. Соединение таблицы со своей копией	136
6.2.7. Внешние соединения	138
6.3. Вложенные подзапросы	140
6.3.1. Виды вложенных подзапросов.....	140
6.3.2. Простые вложенные подзапросы.....	141
6.3.3. Использование одной и той же таблицы во внешнем и вложенном подзапросе	143
6.3.4. Вложенный подзапрос с оператором сравнения, отличным от <i>IN</i>	144
6.3.5. Коррелированные вложенные подзапросы.....	144
6.3.6. Запросы, использующие <i>EXISTS</i>	146
6.3.7. Функции в подзапросе	147
6.4. Фразы для работы с наборами: <i>EXCEPT (MINUS), INTERSECT, UNION</i>	148
6.5. Заключение	150

ЧАСТЬ III. ЯЗЫК SQL. ИЗМЕНЕНИЕ ДАННЫХ..... 157

Глава 7. Организация доступа к базе данных..... 159

7.1. О системе баз данных	159
7.1.1. Данные.....	159
7.1.2. Аппаратное обеспечение	160
7.1.3. Программное обеспечение	160
7.1.4. Пользователи	160
7.2. Защита данных	161
7.3. Средства языка SQL	163
7.3.1. Предложение <i>GRANT</i>	163
7.3.2. Предложение <i>REVOKE</i>	165

7.3.3. Синонимы	166
7.3.4. Представления	167
7.3.5. Разграничение доступа к записям таблицы	172
Глава 8. Внесение изменений в базу данных.....	175
8.1. Особенности и синтаксис предложений модификации	175
8.2. Предложение <i>DELETE</i>	177
8.2.1. Удаление единственной записи	177
8.2.2. Удаление множества записей.....	177
8.2.3. Удаление с вложенным подзапросом.....	178
8.3. Предложение <i>INSERT</i>	178
8.3.1. Вставка единственной записи в таблицу	179
8.3.2. Вставка множества записей.....	180
8.4. Предложение <i>UPDATE</i>	182
8.4.1. Обновление единственной записи	183
8.4.2. Обновление множества записей	183
8.4.3. Обновление с подзапросом	183
8.4.4. Обновление нескольких таблиц.....	183
Глава 9. Транзакции и параллелизм	185
9.1. Что такое транзакция.....	185
9.2. Предложения <i>COMMIT</i> , <i>ROLLBACK</i> и <i>SAVEPOINT</i>	187
9.3. Многопользовательский режим работы	188
9.3.1. Параллелизм транзакций	188
9.3.2. Блокировки	189
ЧАСТЬ IV. ОСНОВЫ ПРОЕКТИРОВАНИЯ БАЗ ДАННЫХ.....	193
Глава 10. Введение в проектирование.....	195
10.1. Цели проектирования	195
10.2. Универсальное отношение.....	198
10.3. Почему проект базы данных может быть плохим?	201
10.4. Процедура проектирования	204
10.4.1. Этапы проектирования базы данных	205
Глава 11. Нормализация	211
11.1. О нормализации, функциональных и многозначных зависимостях	211
11.2. Нормальные формы	213
11.3. Процедура нормализации.....	216
11.4. Построение даталогической (табличной) модели	218
11.5. Различные советы и рекомендации.....	222

Глава 12. Пример проектирования базы данных "LIBRARY"	224
12.1. Назначение и предметная область	224
12.2. Построение инфологической модели	228
12.3. Построение даталогической модели	230
ЧАСТЬ V. ЯЗЫК SQL. СОЗДАНИЕ БАЗЫ ДАННЫХ.....	239
Глава 13. Создание базы данных и ее основных объектов	241
13.1. О языке определения данных (DDL)	241
13.2. Создание базы данных и схем	242
13.3. Создание таблиц.....	243
13.3.1. Описание таблицы.....	245
13.3.2. Ограничения целостности	247
13.3.3. Комментарии к описанию таблицы.....	250
13.4. Изменение таблиц.....	251
13.5. Удаление таблиц	254
13.6. Создание последовательностей	254
Глава 14. Системный каталог (словарь данных).....	256
14.1. Что такое системный каталог	256
14.2. Словарь данных Oracle.....	257
14.2.1. Структура словаря данных	258
14.2.2. Краткое содержимое словаря данных	259
14.2.3. Примеры использования словаря данных.....	263
Глава 15. Оптимизация SQL-запросов	269
15.1. Введение	269
15.2. Использование индексов	271
15.2.1. Что такое индексы.....	271
15.2.2. Создание индексов	272
15.2.3. Необходимость использования индексов	273
ЧАСТЬ VI. СОЗДАНИЕ ПРИЛОЖЕНИЙ НА SQL.....	275
Глава 16. Программирование на SQL	277
16.1. Введение	277
16.2. Статический SQL	278
16.3. Динамический SQL.....	279
16.4. Интерфейс программирования приложений.....	281

Глава 17. Процедурные расширения SQL	283
17.1. Введение	283
17.2. Основы PL/SQL.....	283
17.2.1. Анонимный блок PL/SQL.....	284
17.3. Переменные, константы, записи PL/SQL	286
17.4. Команды управления ходом выполнения программы	289
17.4.1. Команды условного перехода (<i>IF</i>)	289
17.4.2. Метки и оператор безусловного перехода (<i>GOTO</i>)	290
17.4.3. Операторы цикла (<i>LOOP</i> , <i>WHILE...LOOP</i> и <i>FOR...LOOP</i>).....	291
17.4.4. Операторы <i>EXIT</i> , <i>EXIT-WHEN</i> и <i>NULL</i>	293
17.5. SQL-предложения в PL/SQL.....	294
17.5.1. <i>SELECT...INTO</i>	295
17.5.2. <i>INSERT</i> , <i>UPDATE</i> и <i>DELETE</i>	296
17.6. Обработка ошибок.....	296
17.6.1. Встроенные исключительные ситуации	297
17.6.2. Исключительные ситуации, определяемые пользователем	299
17.6.3. Обработчик <i>OTHERS</i>	300
17.7. Курсоры	300
17.7.1. Связь объектов PL/SQL с таблицами базы данных	300
17.7.2. Явный курсор.....	300
17.7.3. Неявный курсор (SQL-курсor).....	307
17.8. Динамический SQL в PL/SQL.....	308
Глава 18. Хранимые процедуры	315
18.1. Введение	315
18.2. Хранимые процедуры.....	316
18.2.1. Создание описания процедуры	316
18.2.2. Удаление описания процедуры.....	317
18.2.3. Перекомпиляция процедуры	317
18.2.4. Пример создания процедуры.....	318
18.3. Функции.....	320
18.3.1. Создание описания функции.....	320
18.3.2. Удаление описания функции	321
18.3.3. Перекомпиляция функции.....	321
18.3.4. Пример создания функции	321
18.4. Триггеры	323
18.4.1. Создание описания триггера	324
18.4.2. Изменение описания триггера	327
18.4.3. Удаление описания триггера.....	328
18.4.4. Использование триггера	328
18.4.5. Изменяющиеся (мутирующие) таблицы	332

18.5. Пакеты (модули)	338
18.5.1. Модули	338
18.5.2. Создание описания пакета.....	338
18.5.3. Изменение описания пакета	340
18.5.4. Удаление пакета	341
18.5.5. Примеры пакетов.....	342
18.6. Встроенные пакеты PL/SQL	345

ЧАСТЬ VII. ПРИМЕР СОЗДАНИЯ БАЗЫ ДАННЫХ "УСНЕВ" 347

Глава 19. Описание предметной области "Учебные планы" 349

19.1. О Государственных образовательных стандартах (ГОС)	349
19.2. Основные образовательные программы (ООП).....	353
19.2.1. Базовые учебные планы.....	353
19.2.2. Индивидуальные учебные планы и планы ускоренного обучения	358
19.2.3. Рабочие учебные планы.....	358
19.2.4. Дисциплины по выбору студента и индивидуальные рабочие планы	360
19.2.5. Студенческие потоки и группы	363
19.2.6. Еще о рабочих учебных планах	365

Глава 20. Построение инфологической модели "Учебные планы" 366

20.1. Первая попытка проектирования	366
20.2. Вторая попытка проектирования.....	369
20.2.1. Шапки планов.....	369
20.2.2. Дисциплины планов	376
20.3. Инфологическая модель "Учебные планы"	381

Глава 21. "Итоговая успеваемость" 382

21.1. Описание предметной области "Итоговая успеваемость"	382
21.2. Инфологическая модель "Итоговая успеваемость"	386
21.3. Объединенная инфологическая модель "УСНЕВ"	390

Глава 22. Работаем с SQL..... 393

22.1. Запросы	393
22.2. Ответы к некоторым запросам	400

Глава 23. Некоторые приложения базы данных "УСНЕВ".....	403
23.1. Функции <i>Человек</i> и <i>Decline</i>	403
23.2. Пакет для просмотра успеваемости	413
Литература.....	425
ПРИЛОЖЕНИЯ.....	427
Приложение А. Инструментальные средства разработки и выполнения	429
A1. Oracle Database Express Edition.....	429
A1.1. Общие сведения.....	429
A1.2. Состав Oracle Database XE	432
A1.3. Требования к программному обеспечению	434
A1.4. Взаимодействие с межсетевыми экранами.....	434
A1.5. Требования к надстройкам Oracle Database для платформы .NET	435
A1.6. Требования к Web-браузеру	435
A2. Установка сервера Oracle Database XE	436
A3. Русификация Oracle Database XE и создание баз данных "COOK" и "УСНЕВ"	442
A4. SqlDeveloper.....	443
A4.1. Введение.....	443
A4.2. Краткое руководство по установке и настройке	445
Приложение Б. Описание содержимого компакт-диска	451
Предметный указатель	453

Глава 3



Реляционный подход

3.1. Реляционная структура данных

В конце 60-х годов появились работы, в которых обсуждались возможности применения различных табличных даталогических моделей данных, т. е. возможности использования привычных и естественных способов представления данных. Наиболее значительной из них была статья исследователя фирмы IBM д-ра Эдгара Кодда [9], где, вероятно, впервые был применен термин "реляционная модель данных".

Будучи математиком по образованию Эдгар Кодд предложил использовать для обработки данных аппарат теории множеств (объединение, пересечение, разность, декартово произведение). Он показал, что любое представление данных сводится к совокупности двумерных таблиц особого вида, известного в математике как *отношение* — relation (англ.) [1, 4, 9].

Наименьшая единица данных реляционной модели — это отдельное *атомарное* (неразложимое) для данной модели значение данных. Так, в одной предметной области фамилия, имя и отчество могут рассматриваться как единое значение, а в другой — как три различных значения.

Доменом называется множество атомарных значений одного и того же типа. Так, на рис. 1.1 домен пунктов отправления (назначения) — множество названий населенных пунктов, а домен номеров рейса — множество целых положительных чисел.

Смысл доменов состоит в следующем. Если значения двух атрибутов берутся из одного и того же домена, то, вероятно, имеют смысл сравнения, использующие эти два атрибута (например, для организации транзитного рейса можно дать запрос "Выдать рейсы, в которых время вылета из Москвы в Сочи больше времени прибытия из Архангельска в Москву"). Если же значения

двух атрибутов берутся из различных доменов, то их сравнение, вероятно, лишено смысла: стоит ли сравнивать номер рейса со стоимостью билета?

Отношение на доменах D_1, D_2, \dots, D_n (не обязательно, чтобы все они были различны) состоит из заголовка и тела. На рис. 3.1 приведен пример отношения для расписания движения самолетов (см. рис. 1.1).

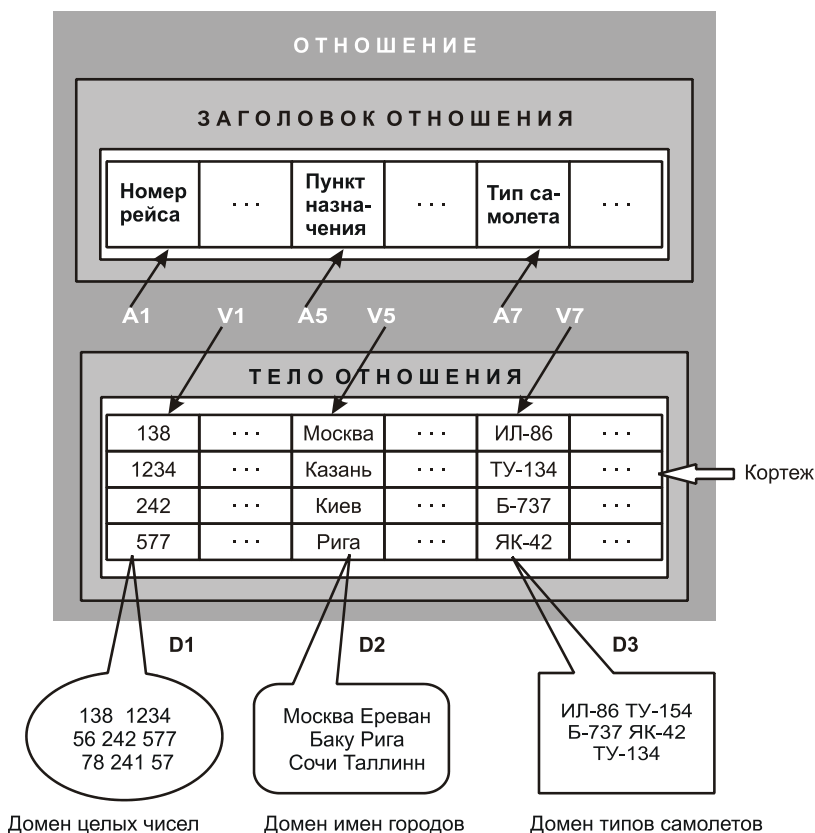


Рис. 3.1. Отношение с математической точки зрения (A_i — атрибуты, V_i — значения атрибутов)

Заголовок (на рис. 1.1 он назывался интерпретацией) состоит из такого фиксированного множества атрибутов A_1, A_2, \dots, A_n , что существует взаимно однозначное соответствие между этими атрибутами A_i и определяющими их доменами D_i ($i = 1, 2, \dots, n$).

Тело состоит из меняющегося во времени множества *кортежей*, где каждый кортеж состоит в свою очередь из множества пар атрибут-значение ($A_i:V_i$),

($i = 1, 2, \dots, n$), по одной такой паре для каждого атрибута A_i в заголовке. Для любой заданной пары атрибут-значение ($A_i:V_i$) V_i является значением из единственного домена D_i , который связан с атрибутом A_i .

Степень отношения — это число его атрибутов. Отношение степени один называют унарным, степени два — бинарным, степени три — тернарным, а степени n — n -арным. Степень отношения РЕЙС (см. рис. 1.1) — 8.

Кардинальное число или мощность отношения — это число его кортежей. Мощность отношения РЕЙС равна 10. Кардинальное число отношения изменяется во времени в отличие от его степени.

Поскольку отношение — это множество, а множества по определению не содержат совпадающих элементов, то никакие два кортежа отношения не могут быть дубликатами друг друга в любой произвольно-заданный момент времени.

Пусть R — отношение с атрибутами A_1, A_2, \dots, A_n . Говорят, что множество атрибутов $K = (A_i, A_j, \dots, A_k)$ отношения R является возможным ключом R тогда и только тогда, когда удовлетворяются два независимых от времени условия:

- уникальность — в произвольный заданный момент времени никакие два различных кортежа R не имеют одного и того же значения для A_i, A_j, \dots, A_k ;
- минимальность — ни один из атрибутов A_i, A_j, \dots, A_k не может быть исключен из K без нарушения уникальности.

Каждое отношение обладает хотя бы одним возможным ключом, поскольку по меньшей мере комбинация всех его атрибутов удовлетворяет условию уникальности. Один из возможных ключей (выбранный произвольным образом) принимается за его первичный ключ. Остальные возможные ключи, если они есть, называются альтернативными ключами.

Упомянутые ранее и некоторые другие математические понятия явились теоретической базой для создания реляционных СУБД, разработки соответствующих языковых средств и программных систем, обеспечивающих их высокую производительность, и создания основ теории проектирования баз данных. Однако для массового пользователя реляционных СУБД можно с успехом использовать неформальные эквиваленты этих понятий:

- отношение — *таблица* (иногда *файл*);
- кортеж — *строка* (иногда *запись*);
- атрибут — *столбец, поле*.

При этом принимается, что *запись* означает *экземпляр записи*, а *поле* означает *имя и тип поля*.

3.2. Реляционная база данных

Реляционная база данных — это совокупность отношений, содержащих всю информацию, которая должна храниться в БД. Однако пользователи могут воспринимать такую базу данных как совокупность таблиц. Так в табл. 3.1—3.10 показаны таблицы базы данных, построенные по инфологической модели базы данных "СООК" (рис. 2.8 и 2.9).

Отметим, что большинство примеров, связанных с изучением основ языка SQL (см. главы 5—7), используют информацию из базы данных "СООК". Поэтому представлялось целесообразным не только привести данные этой базы на прилагаемом к книге компакт-диске, но и в табл. 3.1—3.10, потратив шесть страниц текста. Урезаны лишь тексты рецептов (табл. 3.8) и число строк табл. 3.10.

Так как иллюстративная база данных создавалась для лекционного курса в 1989 году, когда существовали "смешные" цены, а также исчезнувшие названия статусов (коопторг) и городов (Ленинград), то авторы пытались несколько раз ее модифицировать. Однако поняв, что изменение цен, статусов и названий идет быстрее, чем подготовка и, тем более, выпуск издания, они решили сохранить в книге старые цены и названия.

Заметим также, что химический состав продуктов приведен для 1 кг их съедобной части: основные пищевые вещества (белки, жиры и углеводы) даны в граммах, а минеральные вещества (калий, кальций, натрий) и витамины (B2, PP, C) — в миллиграммах. Калорийность блюд определяется по массе и калорийности каждого из продуктов, входящих в это блюдо: для получения значения калорийности продукта исходят из того, что при окислении 1 г углеводов или белков в организме освобождается в среднем 4,1 ккал, а при окислении 1 г жиров — 9,3 ккал. Стоимость продуктов дана в рублях, а трудоемкость приготовления блюда — в копейках.

Таблица 3.1. Блюда

КОД_БЛЮДА	БЛЮДО	КОД_ВИДА	ОСНОВА	ВЫХОД	ТРУД
1	Салат летний	1	Овощи	200,0	3
2	Салат мясной	1	Мясо	200,0	4
3	Салат витаминный	1	Овощи	200,0	4
4	Салат рыбный	1	Рыба	200,0	4
5	Паштет из рыбы	1	Рыба	120,0	5
6	Мясо с гарниром	1	Мясо	250,0	3

7	Сметана	1	Молоко	140,0	1
8	Творог	1	Молоко	140,0	2
9	Суп харчо	2	Мясо	500,0	5
10	Суп-пюре из рыбы	2	Рыба	500,0	6
11	Уха из судака	2	Рыба	500,0	5
12	Суп молочный	2	Молоко	500,0	3
13	Бастурма	3	Мясо	300,0	5
14	Бефстроганов	3	Мясо	210,0	6
15	Судак по-польски	3	Рыба	160,0	5
16	Драчена	3	Яйца	180,0	4
17	Морковь с рисом	3	Овощи	260,0	3
18	Сырники	3	Молоко	220,0	4
19	Омлет с луком	3	Яйца	200,0	5
20	Каша рисовая	3	Крупа	210,0	4
21	Пудинг рисовый	3	Крупа	160,0	6
22	Вареники ленивые	3	Молоко	220,0	4
23	Помидоры с луком	3	Овощи	260,0	4
24	Суфле из творога	3	Молоко	280,0	6
25	Рулет с яблоками	4	Фрукты	200,0	5
26	Яблоки печеные	4	Фрукты	160,0	3
27	Суфле яблочное	4	Фрукты	220,0	6
28	Крем творожный	4	Молоко	160,0	4
29	"Утро"	5	Фрукты	200,0	5
30	Компот	5	Фрукты	200,0	2
31	Молочный напиток	5	Молоко	200,0	2
32	Кофе черный	5	Кофе	100,0	1
33	Кофе на молоке	5	Кофе	200,0	2

Таблица 3.2. ВИДЫ БЛЮД

КОД_ВИДА ВИД

- 1 Закуска
- 2 Суп
- 3 Горячее
- 4 Десерт
- 5 Напиток

Таблица 3.3. ПРОДУКТЫ

КОД_ПРОДУКТА	ПРОДУКТ	БЕЛКИ	ЖИРЫ	УГЛЕВ	К	СА	NA	B2	PP	C
1	Говядина	189,0	124,0	0,0	3150	90	600	1,5	28,0	0
2	Судак	190,0	80,0	0,0	1870	270	0	1,1	10,0	30
3	Масло	60,0	825,0	90,0	230	220	740	0,1	1,0	0
4	Майонез	31,0	670,0	26,0	480	280	0	0,0	0,0	0
5	Яйца	127,0	115,0	7,0	1530	550	710	4,4	1,9	0
6	Сметана	26,0	300,0	28,0	950	850	320	1,0	1,0	2
7	Молоко	28,0	32,0	47,0	1460	1210	500	1,3	1,0	10
8	Творог	167,0	90,0	13,0	1120	1640	410	2,7	4,0	5
9	Морковь	13,0	1,0	70,0	2000	510	210	0,7	9,9	50
10	Лук	17,0	0,0	95,0	1750	310	180	0,2	2,0	100
11	Помидоры	6,0	0,0	42,0	290	140	400	0,4	5,3	250
12	Зелень	9,0	0,0	20,0	340	275	75	1,2	4,0	380
13	Рис	70,0	6,0	773,0	540	240	260	0,4	16,0	0
14	Мука	106,0	13,0	732,0	1760	240	120	1,2	22,0	0
15	Яблоки	4,0	0,0	113,0	2480	160	260	0,3	3,0	130
16	Сахар	0,0	0,0	998,0	30	20	10	0,0	0,0	0
17	Кофе	127,0	36,0	9,0	9710	180	180	0,3	1,8	0

Таблица 3.4. ПОСТАВЩИКИ

КОД_ПОСТАВЩИКА	НАЗВАНИЕ	СТАТУС	ГОРОД	АДРЕС	ТЕЛЕФОН
1	СЫТНЫЙ	рынок	Ленинград	Сытнинская, 3	2329916
2	ПОРТОС	кооператив	Резекне	Садовая, 27	317664
3	ШУШАРЫ	совхоз	Пушкин	Новая, 17	4705038
4	ТУЛЬСКИЙ	универсам	Ленинград	Тульский, 5	2710837
5	УРОЖАЙ	коопторг	Луга	Песчаная, 19	789000
6	ЛЕТО	агрофирма	Ленинград	Пулковская, 8	2939729
7	ОГУРЕЧИК	ферма	Паневежис	Укмерге, 15	127331
8	КОРЮШКА	кооператив	Йыхви	Нарвское ш., 64	432123

Таблица 3.5. ПОСТАВКИ

КОД_ПОСТАВЩИКА	КОД_ПРОДУКТА	ЦЕНА	К_ВО	ДАТА
1	11	1,50	50	14.05.1989
1	12	3,00	10	10.05.1989
1	15	2,00	170	10.05.1989
2	1	3,60	300	14.05.1989
2	6	3,60	80	14.05.1989
2	5	1,80	100	14.05.1989
3	7	0,40	200	14.05.1989
3	12	2,50	20	14.05.1989
3	15	1,50	200	14.05.1989
4	4	2,04	50	09.05.1989
4	13	0,88	150	09.05.1989
4	16	0,94	200	09.05.1989
4	17	4,50	50	09.05.1989
5	4	3,00	50	14.05.1989
5	10	0,50	130	14.05.1989
5	13	1,20	40	14.05.1989
5	14	0,50	70	14.05.1989
5	16	1,00	50	14.05.1989
6	10	0,70	90	10.05.1989
7	1	4,20	70	10.05.1989
7	3	4,00	250	10.05.1989
7	6	2,20	140	10.05.1989
7	8	1,00	150	14.05.1989
8	5	2,00	70	10.05.1989
8	11	1,00	100	10.05.1989

Таблица 3.6. МЕНЮ

СТРОКА	КОД_ТРАПЕЗЫ	КОД_БЛЮДА	ДАТА
1	1	3	15.05.1989
2	1	6	15.05.1989
3	1	19	15.05.1989
4	1	21	15.05.1989

5	1	31 15.05.1989
6	1	32 15.05.1989
7	2	1 15.05.1989
8	2	6 15.05.1989
9	2	9 15.05.1989
10	2	12 15.05.1989
11	2	14 15.05.1989
12	2	16 15.05.1989
13	2	18 15.05.1989
14	2	26 15.05.1989
15	2	28 15.05.1989
16	3	6 15.05.1989
17	3	8 15.05.1989
18	3	20 15.05.1989
19	3	16 15.05.1989
20	3	30 15.05.1989
21	3	31 15.05.1989

Таблица 3.7. ТРАПЕЗЫ

КОД_	КОД_ ТРАПЕЗЫ	ТРАПЕЗА
-----	-----	-----
	1	Завтрак
	2	Обед
	3	Ужин

Таблица 3.8. РЕЦЕПТЫ

КОД_	ИД БЛЮДА	РЕЦЕПТ	ВА- РИАНТ
-----	-----	-----	-----
1	1	Помидоры и яблоки нарезать кружочками, положить помидоры в	
2	2	Вареное охлажденное мясо, свежую зелень и вареную очищенну	
3	3	Зелень мелко нарезать и положить горкой в салатник. Свежие	
4	4	Вареные рыбу и морковь нарезать небольшими ломтиками, а по	
5	5	Филе судака припустить до готовности. Морковь спассеровать	
6	6	Мясо вареное нарезать тонкими кусочками, гарнировать ломти	

7	7	Сметану положить в стакан конической формы. Подавать на ст	
8	8	Протертый творог положить в салатник или на мелкую тарелку	
9	9	Грудинку говядины нарезать на куски и варить до полуготовн	
10	10	Филе судака припустить с маслом и веточкой сельдерея. Гото	
11	11	Судак очистить, разделать на филе с кожей и ребрами, нарез	
12	12	Промытый рис варить в кипящей воде 5-6 минут, откинуть на	
13	13	Мясо нарезать кубиками (30 г), добавить рубленый лук, соль	
14	14	Говядину нарезать на широкие ломти толщиной до 20 мм, отби	
15	15	Подготовленную рыбу нарезать на порционные куски; кожу на	
16	16	Сырые яйца смешать с пшеничной мукой, добавить сметану и с	
17	17	Нарезать кружочками морковь, положить в посуду, добавить п	
18	18	В протертый творог добавить яйца, сахарный песок, соль и п	
19	19	К свежим яйцам добавляют холодное молоко, соль и тщательно	
20	20	Рис сварить в воде до полуготовности, добавить кипящее мол	
21	21	Готовую рисовую рассыпчатую кашу смешать с холодным молоко	
22	22	В протертый творог положить яйца, соль, сахар, размягченно	
23	23	Спассеровать на масле мелко нарезанный лук. В конце пассер	
24	24	В протертый творог положить сметану, яичные желтки, растер	
25	25	Очистить яблоки, разрезать каждое на 8 частей и каждую час	
26	26	Не прорезая насквозь, удалить из яблок сердцевину и семена	
27	27	Запеченные яблоки (см. БЛЮДО 26) охладить и протереть чере	
28	28	Яйца размешать с сахаром и, взбивая, протереть, не доводя	
29	29	Очищенную и промытую морковь натереть на терке, залить вод	
30	30	Яблоки очистить от кожицы, удалить сердцевину и нарезать.	
31	31	Яблоки натереть на терке и отжать из них сок. В стакан вли	
32	32	Вскипятить воду в кофейнике со щепоткой соли или 1/2 чайно	1
33	32	Кофеварку или кастрюлю сполоснуть горячей водой, положить	2
34	33	Сварить черный кофе, как указано в БЛЮДЕ 32, взяв меньшее	

Таблица 3.9. СОСТАВ

КОД_БЛЮДА	КОД_ПРОДУКТА	ВЕС	КОД_БЛЮДА	КОД_ПРОДУКТА	ВЕС	КОД_БЛЮДА	КОД_ПРОДУКТА	ВЕС
1	11	100	12	7	350	21	13	70
1	15	80	12	13	35	21	6	30
1	12	5	12	3	5	21	3	20
1	4	15	12	16	5	21	5	20
2	1	65	13	1	180	21	16	15
2	9	40	13	11	100	22	8	140
2	11	35	13	10	40	22	6	30

2	12	20	13	12	20	22	14	20
2	5	20	13	3	5	22	16	15
2	4	20	14	1	90	22	5	8
3	11	55	14	7	50	23	11	250
3	15	55	14	6	20	23	10	65
3	6	50	14	10	10	23	3	20
3	12	20	14	3	5	24	8	80
3	10	15	14	12	5	24	7	100
3	16	5	14	14	3	24	5	40
4	2	50	15	2	100	24	6	30
4	11	50	15	9	20	24	16	20
4	4	40	15	5	20	24	3	10
4	9	35	15	3	20	24	14	10
4	5	20	15	10	10	25	15	120
4	12	5	15	12	5	25	16	35
5	2	80	16	5	120	25	14	30
5	9	40	16	7	35	25	8	20
5	3	25	16	6	15	25	3	20
5	12	5	16	14	9	26	15	150
6	1	80	16	3	5	26	16	20
6	11	150	17	9	150	26	3	2
6	4	30	17	7	50	27	15	50
6	12	10	17	13	25	27	7	150
7	6	125	17	3	20	27	5	80
7	16	15	17	12	10	27	16	35
8	8	75	17	14	5	27	3	2
8	6	50	18	8	140	28	8	100
8	16	15	18	6	30	28	5	20
9	1	80	18	14	15	28	6	20
9	10	30	18	5	10	28	16	15
9	11	25	18	16	15	28	3	10
9	13	35	19	5	120	29	15	150
9	12	15	19	7	45	29	9	200
9	3	15	19	10	20	29	16	15
10	2	70	19	3	15	30	15	70
10	7	250	20	13	50	30	16	10
10	3	20	20	7	75	31	7	150
10	14	15	20	15	75	31	15	150
10	12	5	20	16	10	31	16	25
11	2	100	20	3	5	32	17	8
11	9	20				33	17	8
11	10	20				33	16	25
11	3	5				33	7	75
11	12	2						

Таблица 3.10. ВЫБОР

МЕСТО	СТРОКА	МЕСТО	СТРОКА	МЕСТО	СТРОКА	МЕСТО	СТРОКА	МЕСТО	СТРОКА
1	1	6	1	11	1	16	2	21	1
1	3	6	4	11	3	16	3	21	3
1	5	6	6	11	5	16	6	21	6
1	8	6	7	11	8	16	7	21	7

1	10	6	10	11	10	16	9	21	10
1	11	6	13	11	11	16	11	21	13
1	14	6	15	11	14	16	14	21	14
1	17	6	16	11	17	16	17	21	16
1	18	6	19	11	18	16	18	21	18
1	20	6	20	11	20	16	21	21	20
2	2	7	2	12	1	17	1	22	1
2	4	7	3	12	3	17	4	22	4
2	6	7	5	12	5	17	5	22	6
2	7	7	8	12	8	17	7	22	7
2	9	7	10	12	10	17	9	22	10
2	11	7	11	12	13	17	11	22	13
2	15	7	14	12	14	17	15	22	14
2	16	7	17	12	16	17	16	22	16
2	19	7	18	12	18	17	19	22	19
2	21	7	21	12	20	17	21	22	20
3	2	8	2	13	2	18	1	23	1
3	3	8	4	13	4	18	4	23	4
3	6	8	5	13	6	18	5	23	6
3	7	8	8	13	7	18	7	23	7
3	9	8	10	13	9	18	9	23	10
3	11	8	11	13	11	18	13	23	13
3	14	8	14	13	15	18	15	23	14
3	17	8	17	13	16	18	16	23	16
3	18	8	19	13	19	18	19	23	19
3	21	8	20	13	21	18	21	23	20
4	1	9	1	14	2	19	1	24	1
4	4	9	3	14	4	19	4	24	4
4	5	9	5	14	6	19	5	24	6
4	7	9	8	14	7	19	7	24	7
4	9	9	10	14	9	19	9	24	10
4	12	9	13	14	11	19	12	24	12
4	15	9	14	14	15	19	15	24	15
4	16	9	16	14	16	19	16	24	16
4	18	9	18	14	19	19	18	24	18
4	21	9	20	14	21	19	21	24	20
5	1	10	1	15	2	20	1	25	2
5	3	10	3	15	3	20	4	25	3
5	6	10	5	15	6	20	5	25	5
5	7	10	8	15	7	20	7	25	8
5	10	10	10	15	9	20	9	25	10
5	13	10	13	15	11	20	13	25	11
5	14	10	14	15	14	20	14	25	14
5	16	10	16	15	17	20	16	25	17
5	19	10	18	15	18	20	19	25	18
5	20	10	20	15	21	20	21	25	21

В рассмотренной базе данных, как и в любой другой реляционной базе данных:

- каждая таблица состоит из однотипных строк и имеет уникальное имя;
- строки имеют фиксированное число полей (столбцов) и значений (множественные поля и повторяющиеся группы недопустимы). Иначе говоря,

в каждой позиции таблицы на пересечении строки и столбца всегда имеется в точности одно значение или ничего;

- строки таблицы обязательно отличаются друг от друга хотя бы единственным значением, что позволяет однозначно идентифицировать любую строку такой таблицы;
- столбцам таблицы однозначно присваиваются имена, и в каждом из них размещаются однородные значения данных (даты, фамилии, целые числа или денежные суммы);
- полное информационное содержание базы данных представляется в виде явных значений данных, и такой метод представления является единственным. В частности, не существует каких-либо специальных "связей" или указателей, соединяющих одну таблицу с другой. Так, связи между строкой с код_блюда=9 табл. 3.1 (БЛЮДА) и строкой с код_продукта=13 табл. 3.3 (ПРОДУКТЫ) (для приготовления харчо нужен рис), представляются не с помощью указателей, а благодаря существованию в табл. 3.9 (СОСТАВ) строки, в которой код блюда равен 9, а код продукта — 13;
- при выполнении операций с таблицей ее строки и столбцы можно обрабатывать в любом порядке безотносительно к их информационному содержанию. Этому способствует наличие имен таблиц и их столбцов, а также возможность выделения любой строки или любого набора строк с указанными признаками (например, продуктов, не содержащих углеводов, но имеющих в своем составе витамин С).

3.3. Манипулирование реляционными данными

В главе 11 будет показано, что стремление к минимизации числа таблиц для хранения данных может привести к возникновению различных проблем при их обновлении, и будут даны рекомендации по разбиению некоторых больших таблиц на несколько маленьких. Но как сформировать требуемый ответ, если нужные для него данные хранятся в разных таблицах?

Предложив реляционную модель данных, Эдгар Кодд создал и инструмент для удобной работы с отношениями — реляционную алгебру. Каждая операция этой алгебры использует одну или несколько таблиц (отношений) в качестве ее операндов и продуцирует в результате новую таблицу, т. е. позволяет "разрезать" или "склеивать" таблицы (рис. 3.2).

Созданы языки манипулирования данными, позволяющие реализовать все операции реляционной алгебры и практически любые их сочетания. Среди

них наиболее распространены SQL (Structured Query Language — *структурированный язык запросов*) и QBE (Query-By-Example — *запросы по образцу*) [5, 6]. Оба относятся к языкам очень высокого уровня, с помощью которых пользователь указывает, какие данные необходимо получить, не уточняя процедуру их получения.

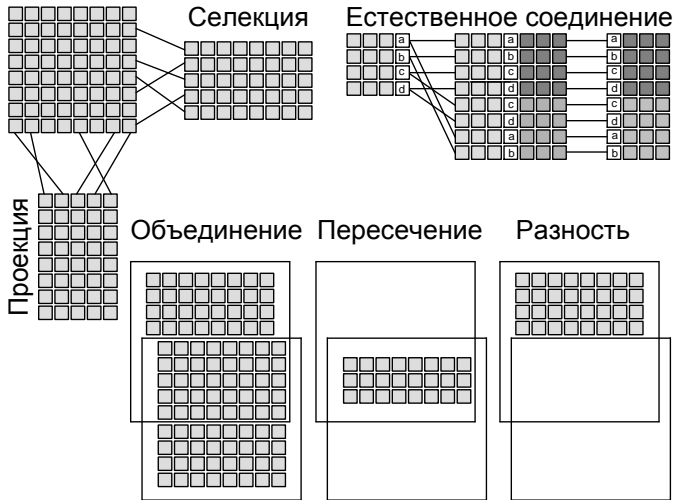


Рис. 3.2. Некоторые операции реляционной алгебры

С помощью единственного запроса на любом из этих языков можно соединить несколько таблиц во временную таблицу и вырезать из нее требуемые строки и столбцы (селекция и проекция).

3.3.1. Обновление отношений

Отношения (например, отношение R , с атрибутами A_1, A_2, \dots, A_n) могут дополняться, удаляться или изменяться.

Добавление

Если эту операцию обозначить ADD и применить к отношению R , то можно записать:

$ADD (R; A_1 = d_1, A_2 = d_2, \dots, A_n = d_n)$

или при фиксированном порядке имен атрибутов

ADD (R; d_1, d_2, \dots, d_n),

где d_i — значение i -го атрибута добавляемого кортежа.

Пример 3.1. Добавить в таблицу БЛЮДА (табл. 3.1) блюдо Шашлык:

ADD (Блюда; Код_блюда=35, Блюдо='Шашлык', Код_вида=3, Основа='Мясо', Выход=200, Труд=6)

или

ADD (Блюда; 35, 'Шашлык', 3, 'Мясо', 200, 6)

Цель операции — добавить указанный кортеж в определенное отношение. Результат операции может быть не согласован с целями операции по следующим причинам:

- добавляемый кортеж не соответствует описанию (схеме) определенного отношения;
- некоторые значения кортежа не принадлежат соответствующим доменам (например, по описанию значения атрибута должны быть целыми числами, а в операции указано текстовое значение);
- описанный кортеж совпадает по ключу с кортежем, уже находящимся в отношении.

В каждом из этих случаев операция ADD (R; d_1, d_2, \dots, d_n) оставляет отношение R неизменным и некоторым образом сообщает об ошибке.

Пример 3.2. Добавить в таблицу ПРОДУКТЫ (табл. 3.3) Шпик:

ADD (Продукты; Код_продукта=8, Продукт='Шпик', Белки=110, Жиры=850, Углеводы='Отсутствуют', К=2700, СА=120, НА=710, В2=1,2, РР=32, С=0)

Эта операция не допускается по всем из перечисленных причин (продукт с номером Код_продукта=8 уже существует, Углеводы именуется Углев и оцениваются в граммах, а не текстовой величиной).

Удаление

Эта операция, которую мы обозначим DEL, вводится для уничтожения сделанного. Для приведенного ранее отношения R она записывается в виде

DEL (R; $A_1= d_1, A_2= d_2, \dots, A_n= d_n$)

или при фиксированном порядке имен атрибутов

DEL (R; d_1, d_2, \dots, d_n).

Пример 3.3. Из таблицы ПОСТАВЩИКИ (табл. 3.5) удалить поставщика ПОРТОС:

DEL (Поставщики; Код_поставщика=3, Название='ПОРТОС', Город='Резекне')

или

DEL (Поставщики; 3, 'ПОРТОС', 'Резекне')

В действительности нет необходимости задавать так много информации, чтобы однозначно определить кортеж, который надо удалить. Достаточно определить значения некоторого ключа. Если $K = \{V_p, V_j, \dots, V_n\}$ является ключом, то можно использовать следующую форму записи:

DEL (R; $V_1 = e_1, V_2 = e_2, \dots, V_n = e_n$).

Краткая форма записи операции удаления:

DEL (Поставщики; 3)

Операция удаления не выполняется лишь в тех случаях, когда заданный кортеж отсутствует в отношении. Тогда отношение остается неизменным и сообщается об ошибке условия. Ограничения на удаление последнего кортежа из отношения не накладывается; пустое отношение допускается.

Изменение

Вместо того чтобы добавлять или удалять целый кортеж отношения, можно изменить лишь часть кортежа. Для R при $\{C_1, C_2, \dots, C_p\}$ из $\{A_1, A_2, \dots, A_n\}$ операция изменения имеет вид:

CH (R; $A_1 = d_1, A_2 = d_2, \dots, A_n = d_n; C_1 = e_1, C_2 = e_2, \dots, C_p = e_p$).

Или если $K = \{V_1 = e_1, V_2 = e_2, \dots, V_n = e_n\}$ является ключом, то

$C_1 = e_1, C_2 = e_2, \dots, C_p = e_p$.

Пример 3.4. Уменьшить количество лука в бастурме до 30 г.

CH (Состав; Код_блюда=13, Код_продукта=10, Вес=40; Вес=30)

или

CH (Состав; Код_блюда=13; Вес=30)

Операция изменения является наиболее удобной. Тот же результат может быть получен с помощью операции добавления, следующей за операцией удаления. Таким образом, все возможные ошибки операции добавления и удаления присущи и операции изменения: указанный в операции кортеж не существует, изменения имеют неправильный формат или используемые значения не принадлежат существующему домену или измененный кортеж имеет тот же ключ, что и кортеж, уже принадлежащий отношению.

3.3.2. Реляционные операции

Операции обновления — это операции над кортежами. В данном разделе мы будем рассматривать операторы, которые включают в себя целое отношение, т. е. позволяют "разрезать" и "склеивать" таблицы.